

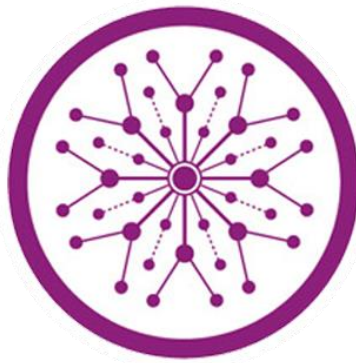
FundMe Crowdfunding

Final Year Project

Session 2019-2023

A project submitted in partial fulfillment of the degree of

BS in Computer Science



Department of Computer Science

Faculty of Computer Science & Information Technology

The Superior University, Lahore

Spring 2023

Type (Nature of project)	[☒] Development [☐] Research [☐] R&D			
Area of specialization	Blockchain			
FYP ID	FYP-BCSM-F22-069			
Project Group Members				
Sr.#	Reg. #	Student Name	Email ID	*Signature
(i)	BCSM-F19-122	Faisal Hussain	Bcsm-f19-122@superior.edu.pk	
(ii)	BCSM-F19-054	M. Naveed	Bcsm-f19-054@superior.edu.pk	
(iii)	BCSM-F19-168	M. Adnan	Bcsm-f19-168@superior.edu.pk	

*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to the work of others.

Plagiarism Free Certificate

This is to certify that, I am **Faisal Hussain** S/D of **Muhammad Hussain**, group leader of FYP under registration no **BCSM-F19-122** at Computer Science Department, The Superior University, Lahore. I declare that my FYP report is checked by my supervisor.

Date: _____ Name of Group Leader: _____ Signature: _____

Name of Supervisor: Mr. Rizwan

Designation: Lecturer

Signature: _____

HOD: Dr. Irfan Ud din

Signature: _____

FundMe Crowdfunding

Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
	1.0		<Original Draft>	
			<Changes Based on Feedback from Supervisor>	
			<Changes Based on Feedback From Faculty>	
			<Added Project Plan>	
			<Changes Based on Feedback from Supervisor>	

APPROVAL

PROJECT SUPERVISOR

Comments: _____

Name: _____

Date: _____ Signature: _____

PROJECT MANAGER

Comments: _____

Date: _____ Signature: _____

HEAD OF THE DEPARTMENT

Comments: _____

Date: _____ Signature: _____

Dedication

This work is dedicated to our supervisor, Mr. Rizwan. He has been with us for over a year and during this time he has shown himself to be a great leader, mentor, and friend, He has always been willing to help out and provide support, no matter the situation.

Acknowledgments

I am really thankful to my supervisor and Falcon IT Consulting. The team of Falcon IT Consulting who helped me was extremely helpful and supportive. They made the process of competing for this project much easier. Without their assistance, this project would not have been possible. I would like to thank my team for their invaluable support and cooperation. I would like to acknowledge everyone who was involved in the process of developing this project.

Executive Summary

In the old crowdfunding days, people would post different projects on Kickstarter or GoFundMe and ask for money from friends and family. Crowdfunding companies charge heavily for their services, and investors, the donator, and governments are often unaware of how their money is being used. Blockchain-based crowdfunding is a new way of raising funds for new projects, start-ups, and medical treatment in a decentralized way using cryptocurrencies such as Bitcoin or Ethereum. It works in a way that is similar to Kickstarter except that there is no middle man and the transactions are all recorded on a blockchain ledger. It also ensures that investors know exactly where their money is going and how they will benefit from it. In addition, transactions are faster and investor funds are secure and protected from fraud. An open-source platform called Ethereum allows developers to create decentralized applications on top of its blockchain network which they can use to launch their own crowd-funded projects.

Table of Contents

Plagiarism Free Certificate	2
Dedication	5
Acknowledgments.....	6
Executive Summary	7
Table of Contents	8
List of Figures	10
List of Tables	11
Chapter 1	12
Introduction.....	12
1.1. Background	13
1.2. Motivations and Challenges	13
1.3. Goals and Objectives.....	14
1.4. Literature Review/Existing Solutions	14
1.5. Gap Analysis	15
1.6. Proposed Solution	16
1.7. Project Plan	16
1.7.1. Work Breakdown Structure	16
1.7.2. Roles & Responsibility Matrix	17
1.7.3. Gantt Chart	17
1.8. Report Outline	18
1.9. Empathy Map	18
Chapter 2.....	19
Software Requirement Specifications	19
2.1. Introduction	20
2.1.1. Purpose	20
2.1.2. Document Conventions	20
2.1.3. Intended Audience and Reading Suggestions	20
2.1.4. Product Scope	21
2.2. Overall Description	21
2.2.1. Product Perspective	21
2.2.2. User Classes and Characteristics	22
2.2.3. Operating Environment	22
2.2.4. Design and Implementation Constraints.....	23
2.2.5. Assumptions and Dependencies	23
2.3. External Interface Requirements.....	24
2.3.1. User Interfaces.....	24
2.3.2. Hardware Interfaces.....	24
2.3.3. Software Interfaces	25
2.3.4. Communications Interfaces	25
2.4. System Features.....	26
2.4.1. System Feature 1	26
2.4.1.1. Description and Priority.....	26
2.4.1.2. Stimulus/Response Sequences	27
2.5. Functional Requirements.....	27

2.5.1.	System Feature 2	28
2.5.1.1.	Description and Priority.....	28
2.5.1.2.	Stimulus/Response Sequences	28
2.5.1.3.	Functional Requirements	28
2.5.2.	System Feature 3 (and so on)	29
2.6.	Nonfunctional Requirements.....	29
2.6.1.	Performance Requirements.....	29
2.6.2.	Safety Requirements	29
2.6.3.	Security Requirements.....	29
2.6.4.	Usability Requirements	30
2.6.5.	Software Quality Attributes.....	30
2.6.6.	Maintainability/Supportability Requirements	30
2.7.	Domain Requirements.....	31
Chapter 3.....		32
Use Case Analysis.....		32
3.1.	Use Case Model	34
3.2.	Use Cases Description.....	35
Chapter 4.....		38
System Design		38
4.1.	Architecture Diagram	39
4.2.	Domain Model.....	40
4.3.	Entity Relationship Diagram with data dictionary	41
4.4.	Class Diagram	42
4.5.	Sequence / Collaboration Diagram	43
4.6.	Operation contracts	44
4.7.	Activity Diagram.....	45
4.8.	State Transition Diagram	48
4.9.	Component Diagram	49
4.10.	Deployment Diagram	50
4.11.	Data Flow diagram [<i>only if structured approach is used - Level 0 and 1</i>].....	51
Chapter 5.....		54
Implementation		54
5.1.	Important Flow Control/Pseudo codes	55
5.2.	Components, Libraries, Web Services and stubs	56
5.3.	Deployment Environment	56
5.4.	Tools and Techniques.....	56
Back End:.....		57
5.5.	Best Practices / Coding Standards.....	57
5.6.	Version Control	57
Chapter 6		58
Testing and Evaluation		58
Chapter 7.....		68
Summary, Conclusion & Future Enhancements.....		68
Appendix A: User Manual		72
2.7.1.	References	75

List of Figures

1.7.1	WBS	5
1.7.1	Gantt Chart	6
1.9	Empathy Map	7
2.2	Caption of second figure of second chapter	22
2.3	Caption of third figure of second chapter	26
5.1	Caption of first figure of fifth chapter	49
5.2	Caption of second figure of fifth chapter	49

List of Tables

1.1	label of first table of first chapter	6
1.2	label of second table of first chapter	7
2.1	label of first table of second chapter	14
2.2	label of second table of second chapter	22
2.3	label of third table of second chapter	26
5.1	label of first table of fifth chapter	49
5.2	label of second table of fifth chapter	49

Chapter 1

Introduction

Chapter 1: Introduction

This platform allows companies, investors, and creators to raise funds by creating decentralized crowdfunding campaigns. This application will provide a track of the records, funds, and transactions that take place on the blockchain. By giving control of invested money to the investors, we will bring transparency and trust to the crowdfunding industry. By blockchain technology, we will be able to create a secure and transparent system for all transactions and contracts that need to be carried out on the platform. Blockchain technology offers a reliable approach to tracing transactions between anonymous participants. Fraud and malfunctions can thus be detected quickly.

1.1. Background

This project aims to develop a decentralized website platform that will decentralize fundraising and democratize the investment market. The platform will provide a global audience with an easy way to invest in early-stage companies. This project will have a great impact on the lives of millions of people around the globe who are in desperate need of resources for their non-profit organizations and businesses. Our goal is to make fundraising easy and secure for everyone involved in the crowdfunding process.

1.2. Motivations and Challenges

The motivation of crowdfunding platforms using blockchain technology increases the integrity of several projects and ventures and therefore attracts huge funds from investors and donors.

Challenges:

Provide safe money to those who need it and donors trust that your money is in safe hands.

1.3. Goals and Objectives

A new campaign is created by making an instance of the Campaign factory. The user will access the campaign factory through the user interface to create a new campaign. The donors will donate and contribute to the campaign after creating a new campaign. The user cannot use the money directly; to use it, they need to provide the request for use. The campaign creator will request spending money to buy some accessories or anything with proof. Then all investors will be notified that the creator needs to spend some money. So, the investor needs to approve the request if he wants. When the investor voted, all the approvals were recorded. If the approval exceeds the specified condition, such as two-thirds, the money will be automatically transferred to the vendor to whom the creator must send the money.

1.4. Literature Review/Existing Solutions

Major crowdfunding platforms, Kickstarter and Indiegogo, as well as Amazon have launched several projects consistent with this trend. A higher platform fee may lead to higher firm profits in equilibrium. Firms can use crowdfunding strategically to signal high demand for their products. Kickstarter and Indiegogo are two of the most popular crowdfunding platforms, but success there isn't always guaranteed: For every successful campaign, there are probably several failed campaigns as well. Competition increases the chances of using crowdfunding compared to the monopoly case. A non-monotonic relationship exists between the risk of crowdfunding campaign failure and firm profit.

1.5. Gap Analysis

	CURRENT STATUS	TARGET	GAP ANALYSIS
NETWORK SERVICES	This website will be available on smartphones, Desktop PCs, Laptops, and the required internet connection.	Provide safe money to those who need it and donors trust that your money is in safe hands.	There are no specific applications that provide the secure finding.
USER INTERFACE	The user interface will be interactive with the user so that he can easily access and request to get money.	Our target is to make a website that connects with blockchain and that holds the record of all transactions.	There are few such websites that will provide a complete record of unchanged. Blockchain based websites provides the complete information
DATABASE	1- Blockchain 2- Smart Contract	1- Blockchain 2- Smart Contract	We use blockchain for making smart contracts that hold the record of all nodes by the decentralized approach.
SOURCE CODE	DAPP website using the blockchain technology and React.js, solidity as our programming languages. Web3.js is used for integration of front end and backend.	DAPP website using the blockchain technology and React.js, solidity as our programming languages. Web3.js is used for integration of front end and backend.	There are a few websites that provide a solution but no one is provide a safe funding and donors trust that your money is in safe hands.

1.6. Proposed Solution

To meet the proposed solution, we allow companies, investors, and creators to raise funds for their projects by creating decentralized crowdfunding campaigns on Ethereum networks. We will use the Ethereum network that will allow users to create smart contracts. This application will provide a track of the records, funds, and transactions that take place on the Ethereum network, and provide a decentralized dispute resolution service in case of any issues that may arise during the crowdfunding campaign. By giving control of invested money to the investors, we will bring transparency and trust to the crowdfunding industry. By blockchain technology, we will be able to create a secure and transparent system for all transactions and contracts that need to be carried out on the platform. We believe that this will revolutionize the crowdfunding industry and allow people to raise funds for any project or idea they may have.

1.7. Project Plan

The project is going to be managed and developed according to the described WBS below and the responsibilities assigned.

1.7.1. Work Breakdown Structure

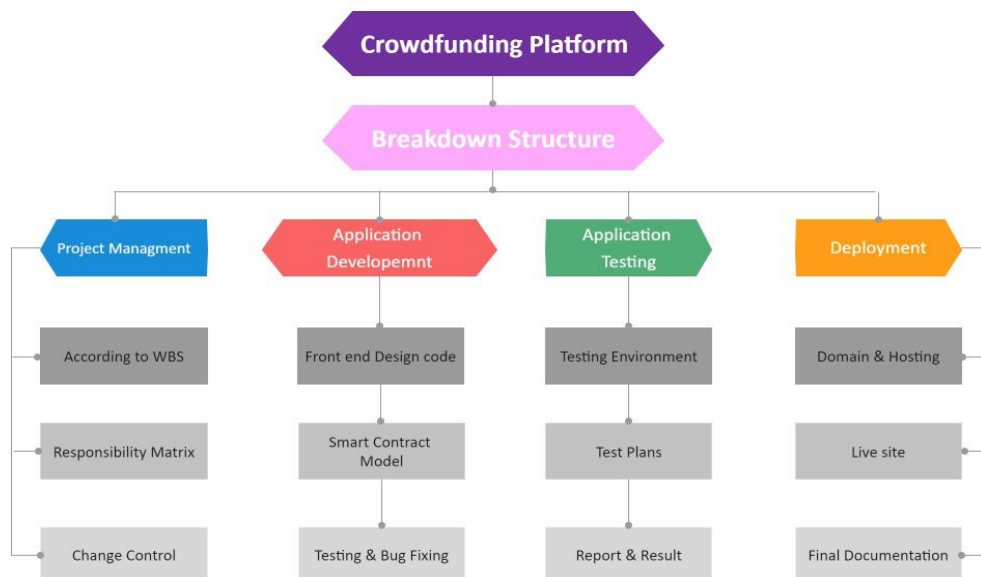


Figure 1: WBS

1.7.2. Roles & Responsibility Matrix

WBS #	WBS Deliverable	Activity to Complete the Deliverable	Duration (# of Days)	Responsible Team Member(s) & Role(s)
1	Project Management	Create WBS and assign responsibilities	2	Faisal
2	Research	Gather all the information	5	All Members
3	Planning	How the process starts	2	All Members
4	Hosting Services	Hosting Servers	1	M. Naveed
5	UI Design	Decide design	10	All Members
6	Front End Code	Now start coding	--	M. Adnan, M. Naveed
8	Smart Contract	Develop a smart contract	--	Faisal Hussain
9	Integration	Interaction of front end and smart contract	--	M. Naveed
10	Testing / Debug	Testing / Debug	--	All Members
11	Deployment website	Live website	2	M. Naveed
12	Final Documentation	Final document and manual of the project	--	All Members

1.7.3. Gantt Chart

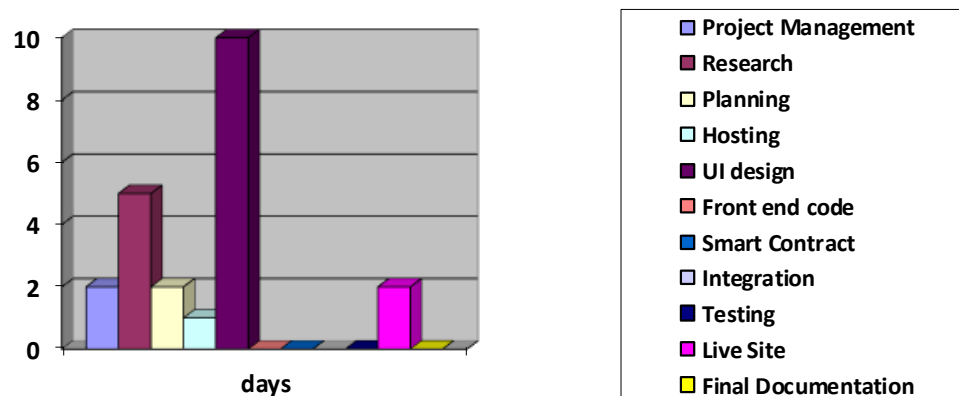


Figure 2: Gantt

1.8. Report Outline

This reports above chapter describes the phases of requirement gathering, documentation, development tools and techniques, implementation, and expected deliverables, as well as clearly defining the roles and responsibilities of each member during each phase for the project's successful development.

1.9. Empathy Map

Say: No Satisfaction Scams Charge Heavily	Do: Satisfy the customer Improving the system Earn more profit
Think: Customer satisfaction Ease of customer Time Saving	Feel: Exhausted Disappointed Excited

Figure 3: Empathy Map

Chapter 2

Software Requirement Specifications

Chapter 2: Software Requirement Specifications

2.1. Introduction

2.1.1. Purpose

The purpose of this software requirement specification document is to provide a detailed description of the functional and working of the website. This document will cover each and every feature of the application including campaign creation, request for money and all the interfaces. This document will cover all intended feature and technical dependencies.

2.1.2. Document Conventions

Standard rules of documentation have been followed in order to standardize the work. This software requirement document uses the following conventions.

Description	Appearance
Website	FundMe
Smart Contract	Solidity
Database	Ethereum Blockchain
SRS	Software Requirements Specification

2.1.3. Intended Audience and Reading Suggestions

This document is intended for developers, project managers and users of this website who wish to read about what the website can do. Rest of SRS contains work flow of the system, how the system works, flow of data at the hosting and other technical functionalities of the software.

- For developers there is a need to start from introduction and understand workflow diagrams.
- The project manager sequence of reading is from introduction to class diagram and use cases.
- For users of a website there is a need to read from introduction and read all descriptions including scope and functionalities.

2.1.4. Product Scope

- This website will be used as a platform where you can get secure funding.
- This project will facilitate people around the globe who are in desperate need of resources for their non-profit organizations and businesses
- The website will be globally used.
- Module of this website includes all the funding menu, a simple interface that covers the whole user experience for the users.

For getting started, we need to get some resources to take our first step. Those resources are listed below

1. Domain
2. Hosting
3. Proper marketing Strategy (We've made our own)

2.2. Overall Description

2.2.1. Product Perspective

The website is designed and developed to provide the best services to the users who surf the website to get secure crowdfunding. The website is an online platform that connects lenders and borrowers with the help of a secure digital smart contract. This website serves as a platform for users who are looking to avail of secure funding so that they can fulfill their needs and lead a comfortable life. It helps users by providing reliable and safe financial solutions to borrowers who need quick money to meet their various requirements.

This website is the replacement of all the websites that provide unsecure crowdfunding. All these services will be provided on a single platform to facilitate its users.

2.2.2. User Classes and Characteristics

Classes	Characteristics
User	The user will access the campaign factory through the user interface to create a new campaign. The user cannot use the money directly; to use it, they need to provide the request for use.
Contributor	The donors will donate and contribute to the campaign after creating a new campaign. All the investors or contributor will be notified that the creator needs to spend some money. So, the investor needs to approve the request if he wants.
Developer	Responsible for optimization, navigation setup and ad-placement.

2.2.3. Operating Environment

- The website can be accessed by using any type of device.
- Meta mask crypto wallet is necessary
- Its design is responsive with every screen. It's dimensions and its content are responsive along every device.
- No user's personal information will be collected apart from its website surfing behavior.

2.2.4. Design and Implementation Constraints

Managerial of Gas Fee

As our mainstream is to provide donators and investors for fundraiser, many of our donors are asking how much the minimum amount we can pay after calculate the gas fee for their transaction. So, we've to manage the gas fee for individual doner if it donates amount is lower than the gas fee. For that we've to manage the smart contract that help us doner can donate funds. No third-party technologies or plugins are present that we can implement at our website through third party APIs that we can implement.

Resources: The fund raise campaigns and other resources that we present on our website can cause an issue due to low availability.

Block Memory: A block is a container data structure. One block contains more than 500 transactions on average. The average size of a block seems to be 1MB (source). There's been a bit of an uproar on the Ethereum subreddit and elsewhere lately about the block size issue.

2.2.5. Assumptions and Dependencies

All the dependencies and assumptions that this project will have been listed below.

1. Smart Contract deployment for Doner and Investor at units present in the website.
2. The third-party plugins that we use can cause performance issues related to the website's design.
3. The website will be customized and manual designed according to Smart Contract Development.
4. The root and sub folders for website code will be done according to Smart Contract Policies so we can have a dynamic use.
5. Hosting server's bandwidth will be a challenge but it will be analyzed and combated between stress testing of websites.
6. A smart contract is a solidity language code that facilitates, records, and enforce the terms of an agreement. When a person or entity signs up for an account of Ethereum blockchain, they are also agreeing to put their trust.

2.3. External Interface Requirements

2.3.1. User Interfaces

The website is developed for making the user navigation through pages and campaign posts as simple as possible. The font family which we used in our website is **Poppins, Rubik**. The UI of the website also contains special units for displaying. Let me mention some of the units in the list that represent website.

1. Header
2. Categories Section
3. Mile Stone Section
4. Explore Project Section
5. Accordion Section
6. Footer

The user will have a navigation menu consisting of categories and search bar. Through which they can navigate through websites and get to their desired campaign project.

Responsiveness: The website is responsive across all the devices and screens that are manufactured across the world. I use Google recommended policy guidelines for navigation along with responsiveness recommendations.

2.3.2. Hardware Interfaces

A user can access this website by using the following devices.

- Mobile Phone
- Tablet
- Computer
- Laptop

Any other device which supports browser installation. Users must be connected to the internet for using the real time enrollment and downloading of courses.

2.3.3. Software Interfaces

Website is an easy-to-use platform designed and developed for native usage. It is developed in React for designing interfaces. Website's running scripts are coded with JavaScript and design is managed by CSS, bootstrap and MUI. The backend of the website is done in node version **18.12.1** hosted on a Linux server provided by Hostinger and the database of website is managed in Smart Contract of Ethereum Blockchain and MongoDB. The components we are using in MongoDB are:

1. Structure
2. Unstructured data
3. Real time database
4. Create and Automate
5. Analytics.

Security: We use Let's Encrypt Authority X3 Secure Sockets Layer (**SSL**) securing the hypertext protocol. That helps us secure the website's domain from different attacks including SQL injection which can cause a very bad effect on website's content and stored databases.

2.3.4. Communications Interfaces

The website is hosted on a Linux server that uses a MongoDB database to host the website's database. It uses a HTTPS connection and the major data network is optimized by Amazon Web Services. There's no use of cookies, the user behavior and other analytics will be controlled and analyzed by a third-party application provided by Google. i.e. **Google Analytics**. The connection of the internet is must for using the website otherwise you'll see a cached version and can't avail the proper usage that is promised by team FundMe.

2.4. System Features

FundMe is a user-friendly website that provides all the features to its users at a single place with. The website's features are as follow:

- Navigation through Header Menu
- Project Campaign Creator
- Donator Donate their Money
- Investor Invest their Money
- Immutable Ledger of Fund Raise Transactions
- Contact Us (Email, WhatsApp)
- Chatbot for Live Q&A.
- Notification for Latest Project Creator.
- Subscription to Latest Updates.

2.4.1. System Feature 1

The system feature is named as the Top most important feature of our website. i.e

“Create Project for Fund Raising” & “Donators and Investors Panel”

2.4.1.1. Description and Priority

The major stakeholder of our project is the Donators and Investors with a high-level priority rated as 10/10. We've the website provides Fund Raising Campaign using blockchain technology. Blockchain technology makes the immutable ledger for secure transaction record but as our user's some data will be saved on server side. So, we've to maintain security and make sure it doesn't get affected with any type of malware or ransomware. When a large flow of traffic will be at our domain, we've to manage the backdoors and other accessible pages like the login of our CMS (Content Management System) to make sure that no one gets access to the data flow and behavior stats of users surfing at our website in real time as well as old users.

2.4.1.2. Stimulus/Response Sequences

- At first, when the user will come to our website's landing page. The website will load the fonts, images and already store cache files for faster loading. After that the user will go to a specific campaign for requesting to raising fund for their project using the navigation menu. On other hand, Donators and Investors will also go to a specific campaign for donating or investing their fund for project using the navigation menu or search bar.
- If the user, donator and investor are use the search bar, the search will be transferred to the hosted server and database will respond with a relative query according to the user's input and the related information with that keywords will appear at the user's screen. In case of no result found at those keywords the users will get a 404 page with a message of no campaign create and request this kind of project approve from admin or upload.
- When a user clicks on create project after fill the required data completely then will be accessed and comes "you have created project for fund raising" from the server whose link will be pointed to that certain button. On the other side donator and investor will transfer their funds that a smart contract inherits with that button present there. All of this process will be done using a custom script made by the developer.

2.5. Functional Requirements

REQ-SF1-1: Menu Navigation

REQ-SF1-2: Request for Fund Raising

REQ-SF1-3: Donators transfer their funds

REQ-SF1-4: Investors transfer their investment

REQ-SF1-5: Donation Immutable Ledger available publicly

REQ-SF1-6: Connect to Meta Mask wallet

2.5.1. System Feature 2

This system feature describes the developer and admin side that how a developer can develop, optimize and make smart contract at the website as well as how an admin can audit the website's condition and optimization needs. It's named as **"Website & Smart Contract Optimization Panel"**

2.5.1.1. Description and Priority

As this feature directly links to our revenue generation and our policies which can indirectly get our website penalized. So, it holds a high priority at scale of 7 out 10.

2.5.1.2. Stimulus/Response Sequences

At first, the developer and admin have to login into WSO Panel. It requires him to have a username and email that is provided by the host manager or admin. When the developer will enter his credentials an automated system of authentication will authenticate the user from the host and in response provide the access to the developer. The core responsibilities of a developer are to check the user behavior from analytics and provide the best solution for optimization in website and smart contract with respect to that. In case of forgetting the password, the developer can request the forgot password by pressing the forgot password function but the admin needs to reset his/her password through the host i.e AWS.

2.5.1.3. Functional Requirements

REQ-SF2-1: Set Menu

REQ-SF2-2: Add Content at website.

REQ-SF2-3: Approve fundraising project by admin.

REQ-SF2-4: Reject fundraising project by admin.

REQ-SF2-5: Stop fake fundraising project by admin.

REQ-SF2-6: Changing in Smart Contract by developer.

REQ-SF2-12: This feature will have two input fields for credentials.

REQ-SF2-13: This feature will have a button for verification.

REQ-SF2-14: This feature will have a forgot password button that'll navigate to the forgot password page.

2.5.2. System Feature 3 (and so on)

All the system features are discussed at System feature 1 and 2.

2.6. Nonfunctional Requirements

2.6.1. Performance Requirements

Performance of this site is very efficient and responsive immediately. It takes 1 sec on average in response to the backend server response quickly and finds objects very fast. It loads in just 1.5s that is less than the google recommended loading time. But the transaction of funds will take up to 5 sec or more time because of mining hash code. All functions are responsive and take less time than a traditional e-learning website because the closest data network is hosted at Amazon website services that is a cloud-based service.

2.6.2. Safety Requirements

Our safety depends on some factors, all of them are listed below with proper explanation.

- **Use of cookies** is the most important data that we need to manage because of this we'll get user analytics and behavioral data about the users.
- **Malware** - The main focus will be preventing malware from getting into our core files that can harm our indexes as well the usage of users. So, we'll use a third-party word fence to combat that and manage all our scanning.
- **Smart Contract**- are a new technology that allow for the execution of agreements between parties without the need for a third party or traditional contracts. This can save time and money, as well as help to ensure that all parties are held accountable.

2.6.3. Security Requirements

- Authentication and password management.
- Authorization and role management.
- Audit logging and analysis.
- Network and data security.
- Code integrity and validation testing.

- Cryptography and key management.
- Data validation and sanitization

2.6.4. Usability Requirements

We see that all the usability criteria are subjective. As a consequence of this, the tests will also have a strong component of subjectivism.

- **User-Friendly**- all the things of website are delightful to use.
- **User-Engagement**- is an interaction between an external donators and investors with fundraisers.
- **Efficiency of use**- goals are easy to accomplish quickly and with few or no user errors
- **Intuitiveness**- interface is easy to learn and navigate; buttons, headings, and help/error messages are simple to understand
- **Low perceived workload**- the interface appears easy to use, rather than intimidating, demanding and frustrating

2.6.5. Software Quality Attributes

- The website and application will be available at any browser.
- The website and application will be available using any type of internet connection
- The user interface will be responsive to enhance user experience.
- Stress testing of websites is done and it can manage up to 10000+ users at a time.
- The application and website will run at any device supporting browsers (In case of website).
- Meta Mask is connected for each page for transactions.
- Immutable ledger available for all donators and investors transaction

2.6.6. Maintainability/Supportability Requirements

- At the admin side, each user has their level for verification, level will define what kind of stakeholder he is and what kind of functionalities he will perform.
- At the admin side, managing the website roles will be assigned to each stakeholder. For example, if he's a developer he'll have access to the Smart Contracts and customization in website.
- No stake holder can access the user data or activities except the admin.

2.7. Domain Requirements

The requirements that are must to fulfill voting system approval factors are most important for funds transfers and I've listed some of others as follows:

1. Proper Navigation to the pages.
2. Proper indexing should be done.
3. There should be almost 30 posts on the website for approval.
4. All the posts and pages should be SEO optimized and in perfect with respect to readability.
5. There should be a proper structure data for your website's search engine presence.
6. Smart Contract should be testing well before deploying because after deploying costly to change in it.
7. Meta Mast should be contacted with each page.

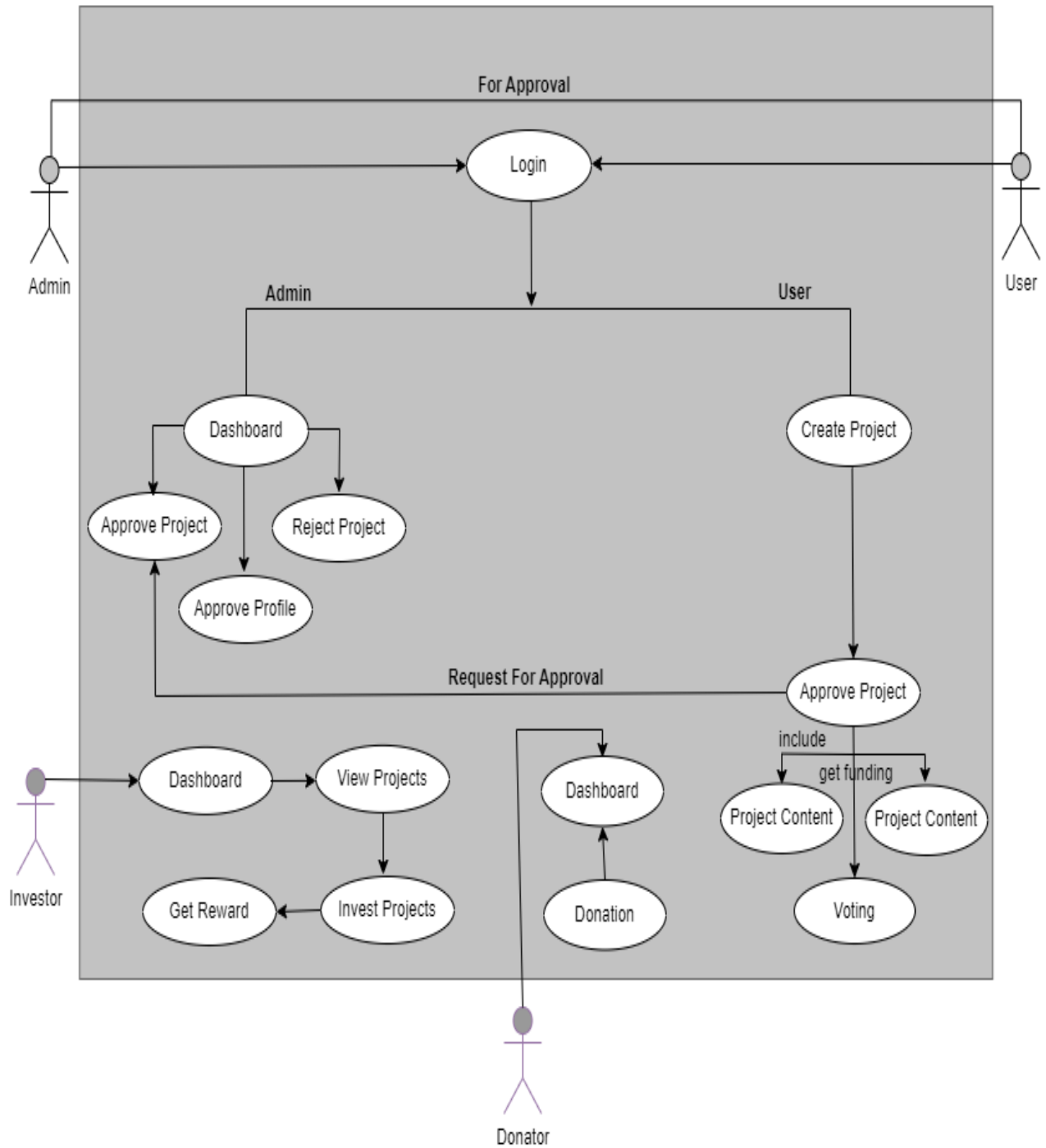
Chapter 3

Use Case Analysis

Chapter 3: Use Case Analysis

This chapter is all about the roles and functionalities of the system and its actors in the use case form. All the primary and secondary actors with respect to their responsibilities. We have made a use case diagram in which we've properly defined each and every functional and nonfunctional requirement according to the respected id or levels. In the use case given below explains the relation between actors and their functionalities.

3.1. Use Case Model



3.2. Use Cases Description

USE CASE NAME	Fund Me
MADE/CREATED BY	Faisal Hussain Muhammad Naveed Muhammad Adnan
MADE/CREATION DATE	2-12-2022
ACTORS	Admin, Smart Contracts, Users, Donators, Investors
DESCRIPTION	The website is designed and developed to provide the best services to the users who surf the website to get funds for their project or to request for fund raising on the specific category to get response. A user can use the navigation menu to go to a specific category according to his/her choice. Website's other users (Investors and Donators) can donate and invest their funds for a specific project. All user's behavior, analytics, bounce rate and all the other factors will be managed in the Smart Contract. The approval for projects and profiles, optimization will be done at admin or developer end. This website is the replacement of all the websites that provide crowdfunding through third party by centralized based system. We've developed the website with Smart Contract of Ethereum Blockchain that will provide immutable ledger of all transactions and latest UI standards.
DATA	Data is gathered from different crowdfunding website which are related to fundraising, donation and investment. User data at the website is also gathered for analytics purposes.
PRE-CONDITION	All Users should have to visit the website no login no sign-up just a connection to the internet.
POST-CONDITION	Create Project for Fundraising, Donation, Investment, Viewing
PRIMARY ACTOR	User/Smart Contract/ Investors / Donators
SECONDARY ACTOR	Admin, Developer

Use Case Name	Create Project for Fund Raising / Get Funds
Priority level	1
Primary actor	Users
Pre-Condition	User just needs to visit the website Fund Me
Post-Condition	Users can create the project for fund raising and get funds for their project under the roles which define in the Smart Contract.
Main success Scenario	The User just needs to visit our website Fund Me and can request for the funds and get funds by crowdfunding.
Extensions	If the users face any type of error in create project for fund raising module, he/she can contact our customer support and can get help through get in touch from for his queries.

Use Case Name	Donate Funds
Priority level	1
Primary actor	Donators
Pre-Condition	Needs to visit the website and go to dashboard
Post-Condition	They can see the created request for fundraising and can donate their donations by Smart Contract and can also get help from our automated chat bot.
Main success Scenario	Donators need to visit the website and can see their donations.
Extensions	If the donation of donator is not transfer or transaction failed, they will contact our customer service for resolve this the issues.

Use Case Name	Investor's Investment
Priority level	2
Primary actor	Investors
Pre-Condition	Need to visit the website and go to dashboard panel
Post-Condition	They can see the created request for capital and can Invest their funds by Smart Contract and can also get help from directly admin in case of any suspicious acts.
Main success Scenario	Investors need to visit the website and can see their investment.
Extensions	If the investment of investor is not transfer or transaction failed, they will contact our customer service for resolve this the issues.

Use Case Name	Developing
Priority level	4
Primary actor	Developer
Pre-Condition	Needs full website access
Post-Condition	Handles the full website designing and developing handling databases, Smart Contract and all other
Main success Scenario	Can log in to the website and can make necessary changes if they are critical and solve all technical issues if they get in, hosting renewal, Smart Contract deploying, and all developing related work
Extensions	Successfully resolving and managing the website if facing some problems can ask the admin before taking any critical steps.

Use Case Name	Administration
Priority level	4
Primary actor	Admin
Pre-Condition	Whole the control on the website and hosting
Post-Condition	Manages all the operations and directs other members to do certain tasks. Perform check-ins on the website, always looking for getting more traffic and growth of the website, full control over all the resources, perform all kinds of CRUD operations related to resources.
Main Success Scenario	Manages all the work related to our website and its resources successfully
Extensions	All things done accurately if face some issues can the meetings and consult with members

Use Case Name	Ethereum Tech
Priority level	1
Primary actor	Smart Contract
Pre-Condition	Whole the control on funding and transaction between users
Post-Condition	Manages all the operations of transaction and record to do for a certain purpose.
Main Success Scenario	Manages all the record of transaction ledger related to our users funds and its resources successfully.
Extensions	All things done accurately if face some issues in smart contract then developer need to maintain it

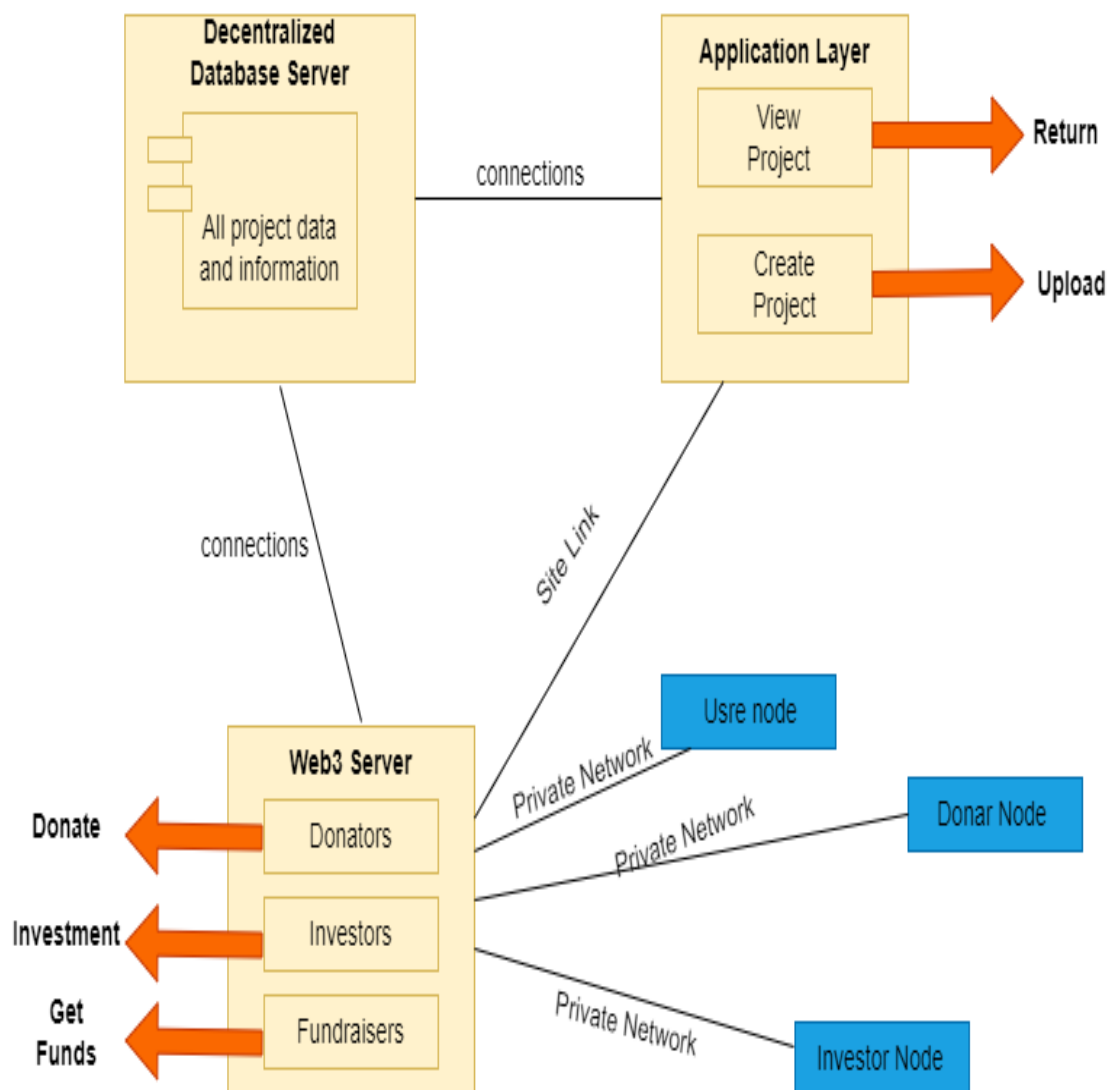
Chapter 4

System Design

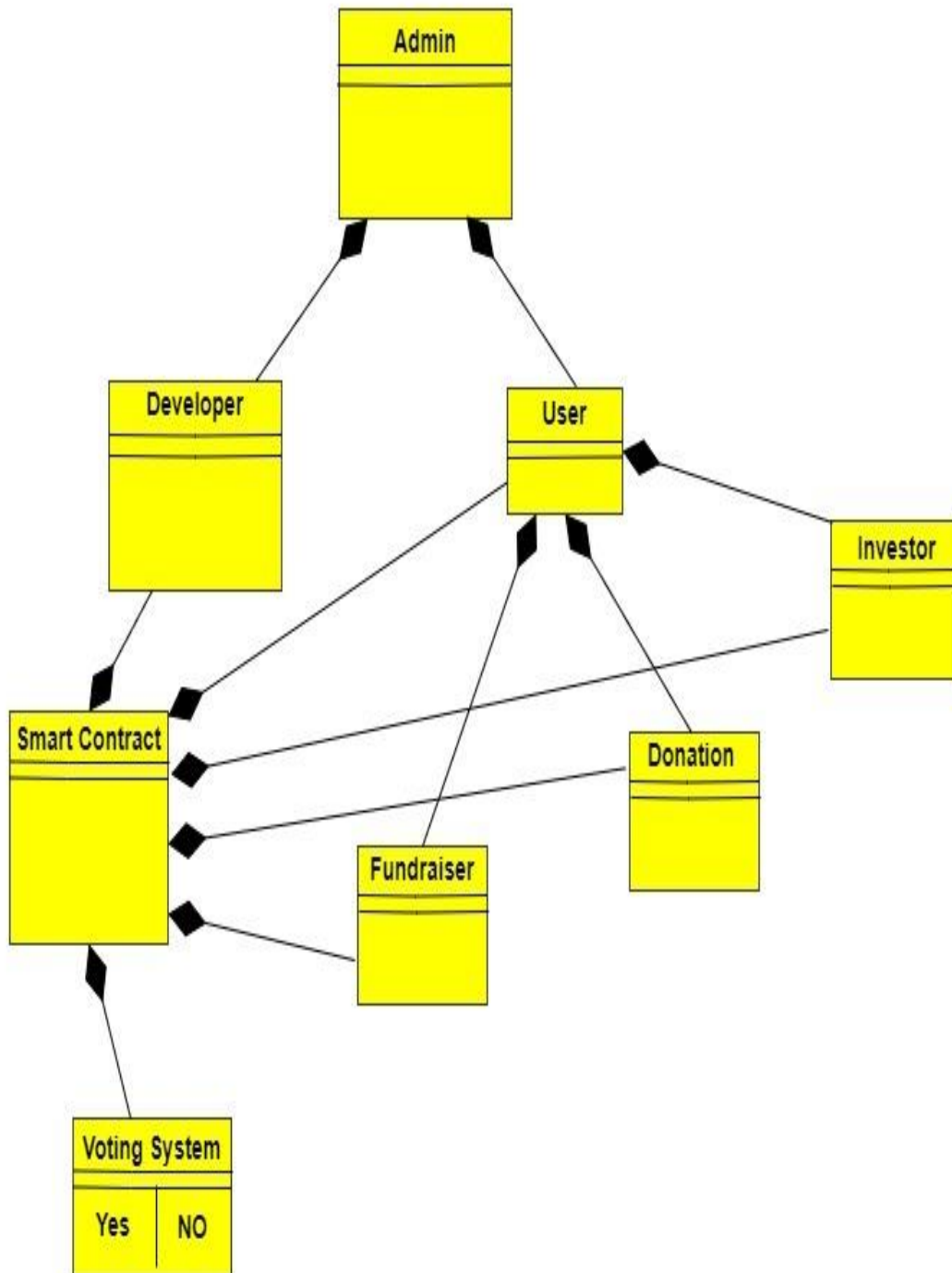
Chapter 4: System Design

In this chapter all the diagrams are given so you can clearly check out the overall working graphically.

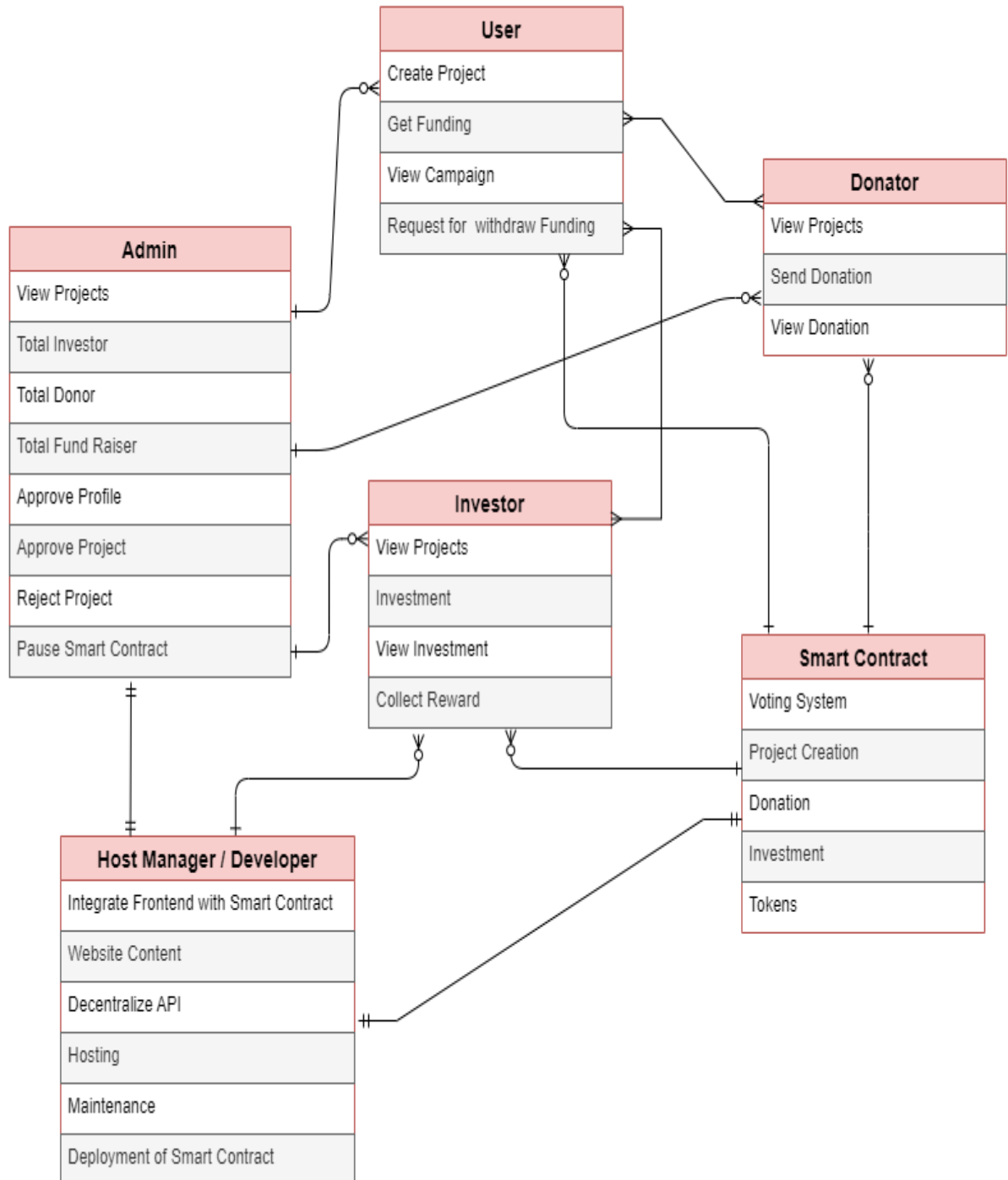
4.1. Architecture Diagram



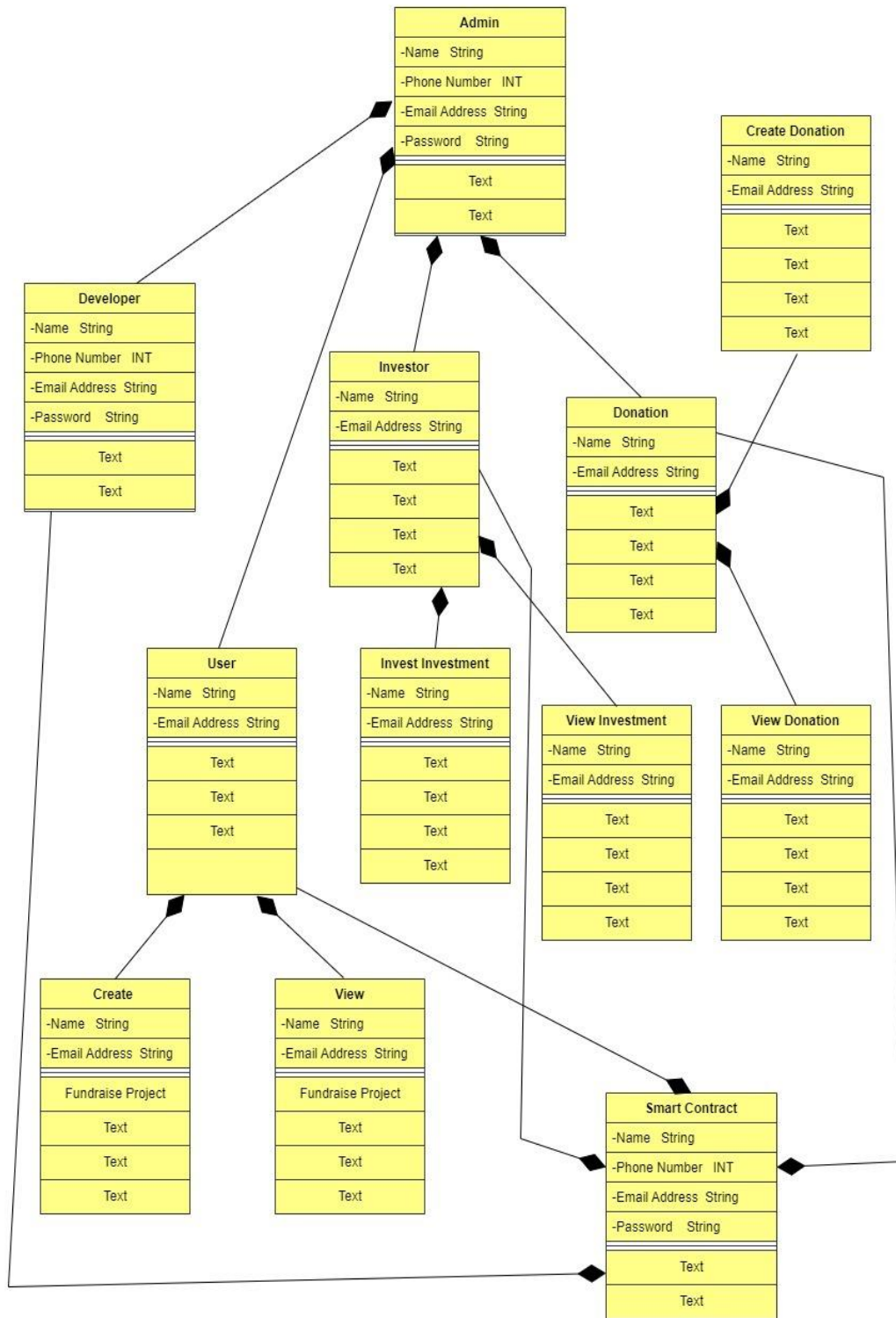
4.2. Domain Model



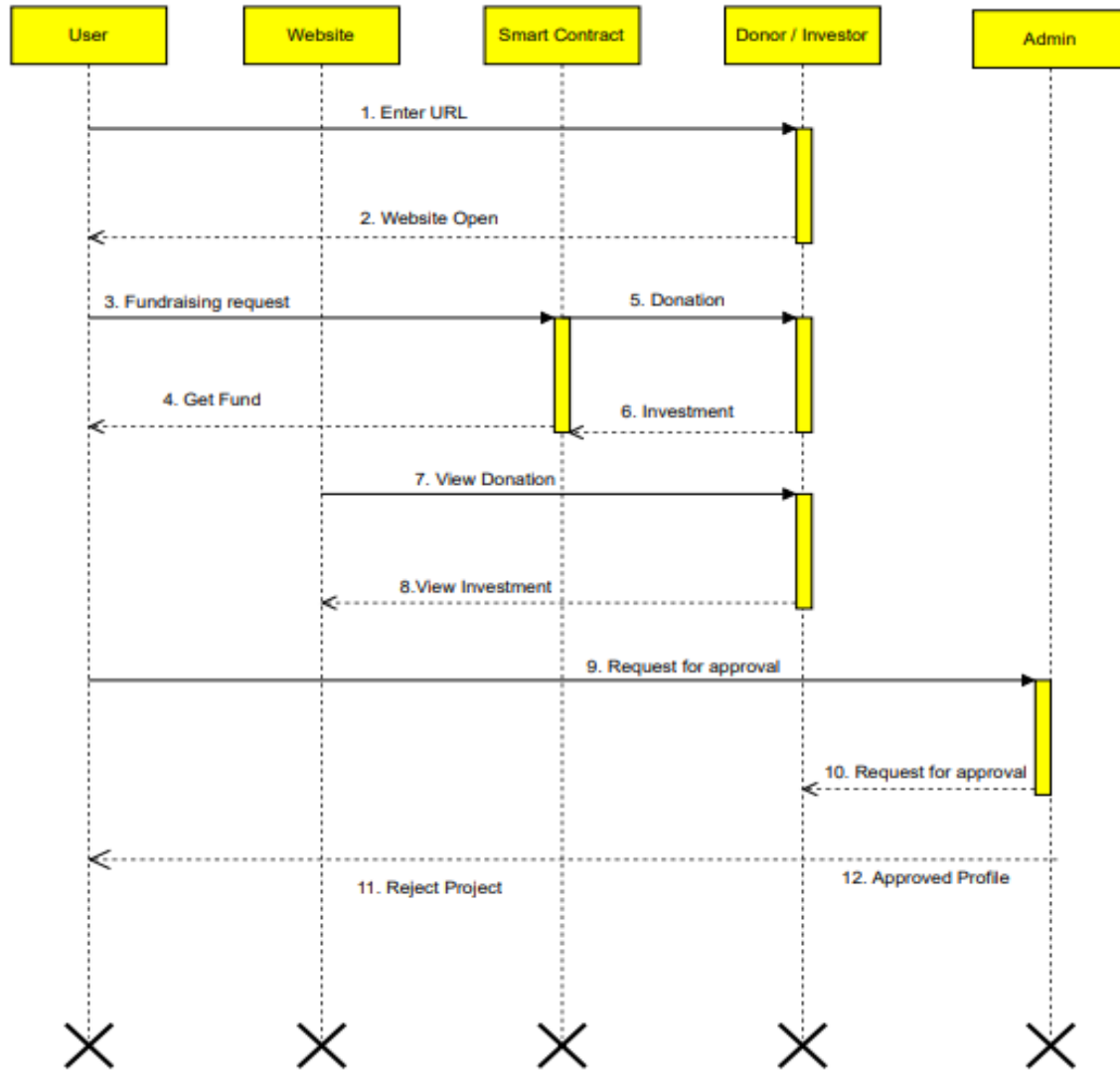
4.3. Entity Relationship Diagram with data dictionary



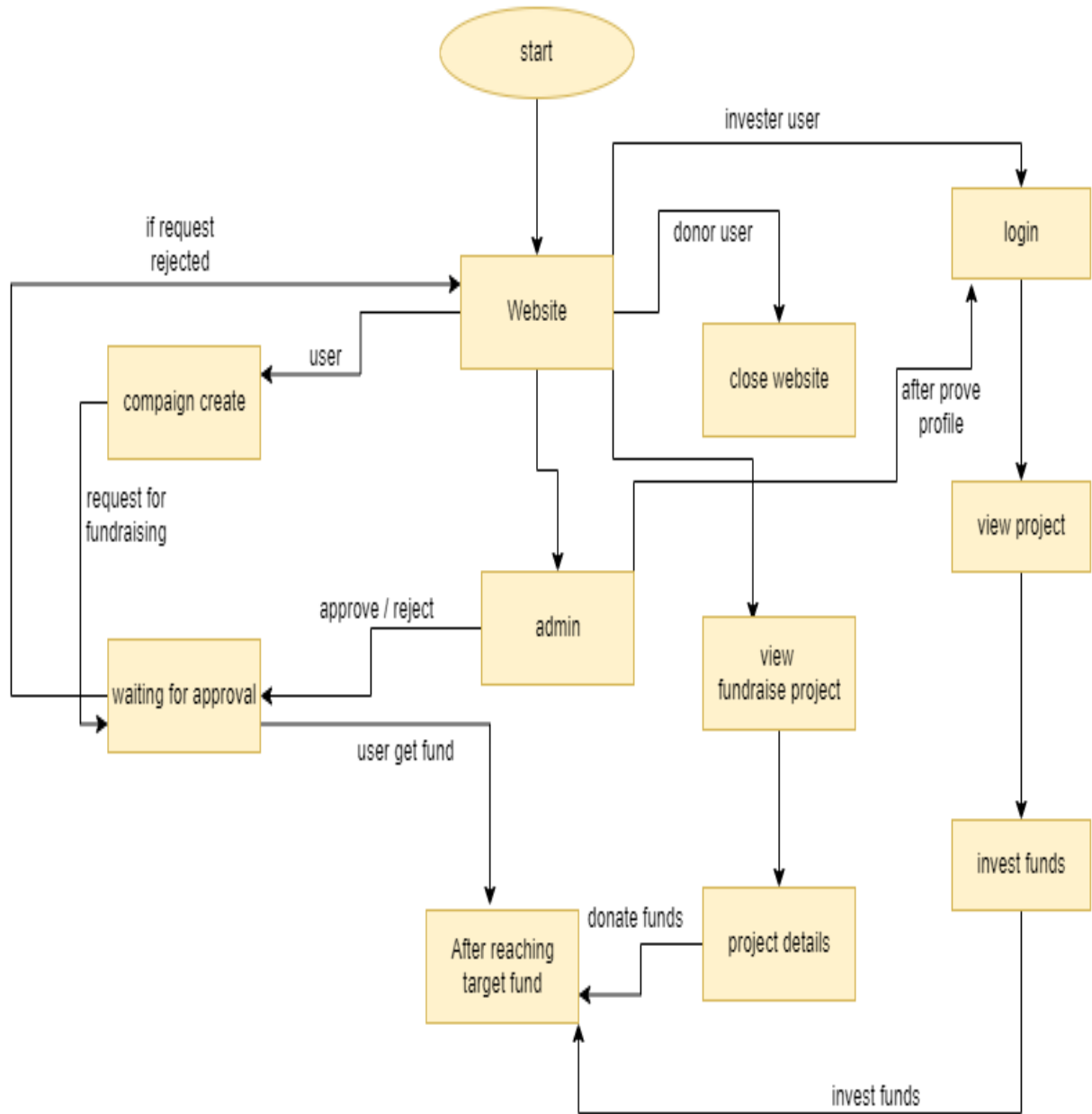
4.4. Class Diagram



4.5. Sequence / Collaboration Diagram

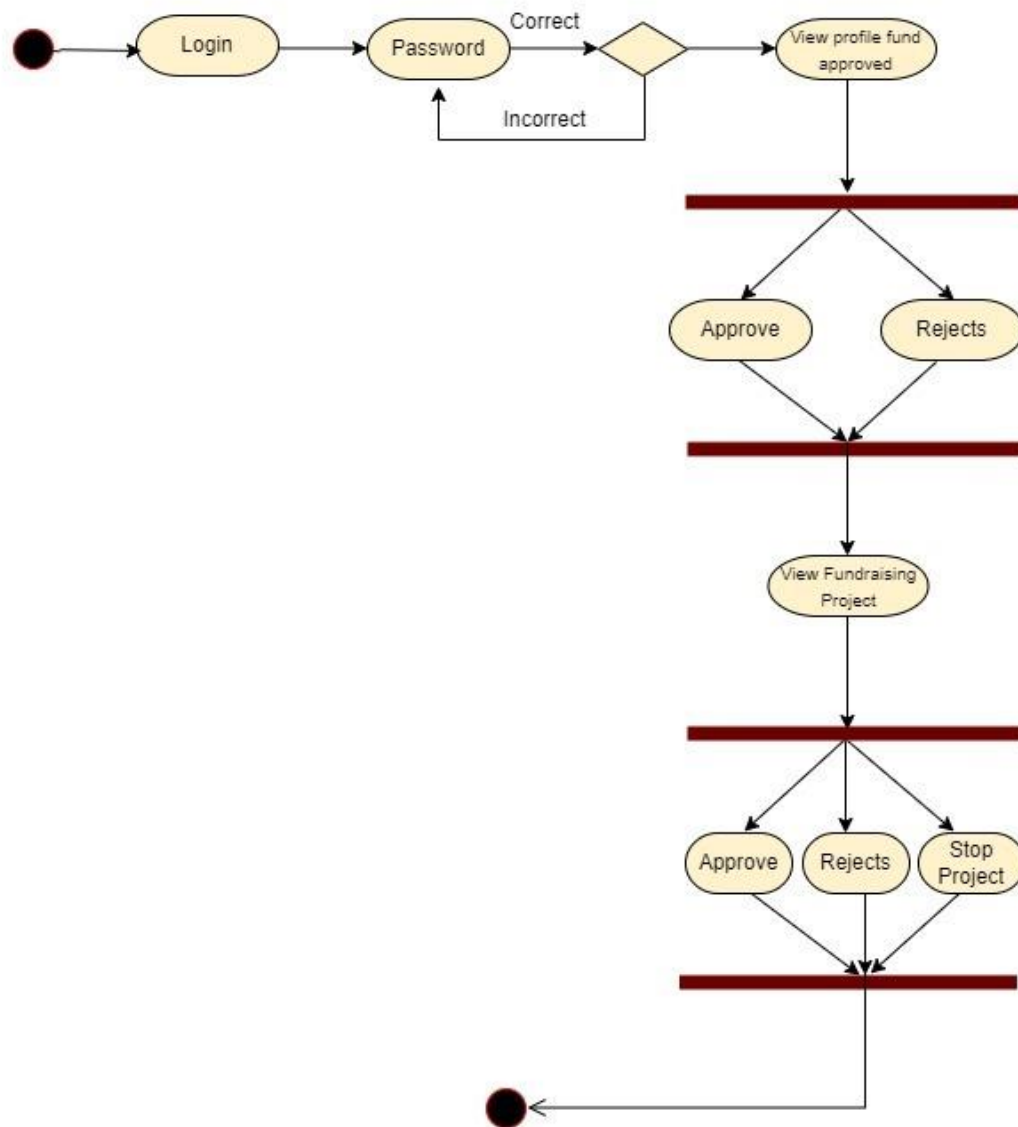


4.6. Operation contracts

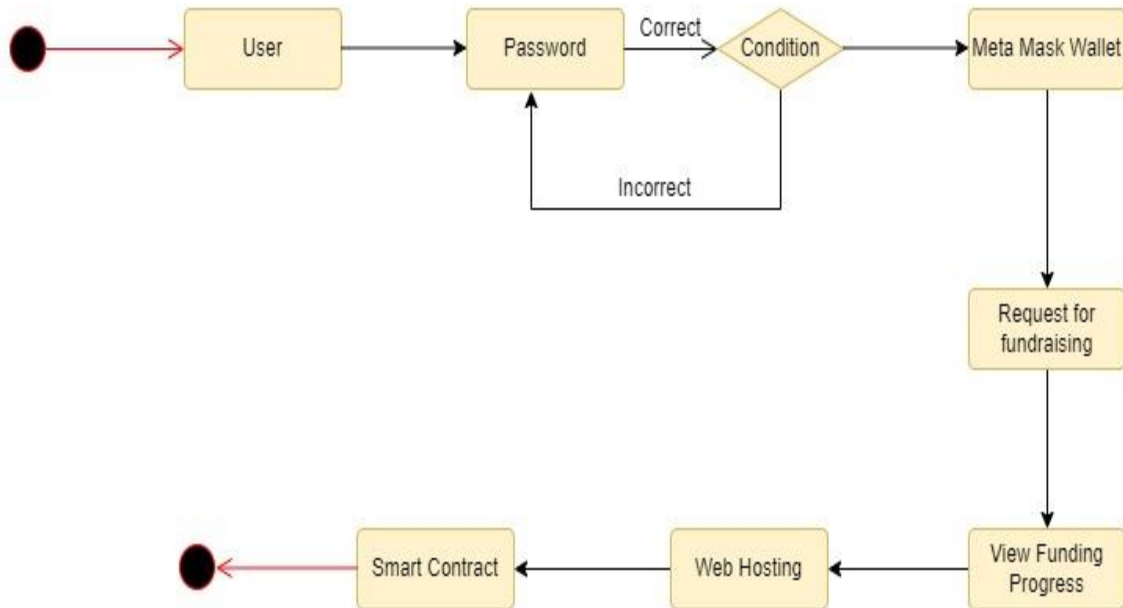


4.7. Activity Diagram

Admin Activity Diagram

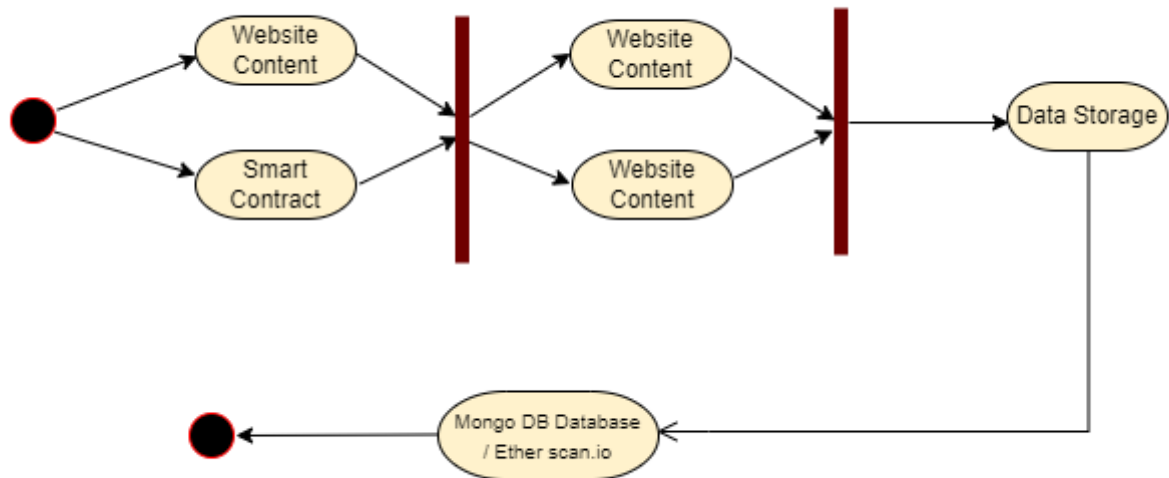


User Activity Diagram

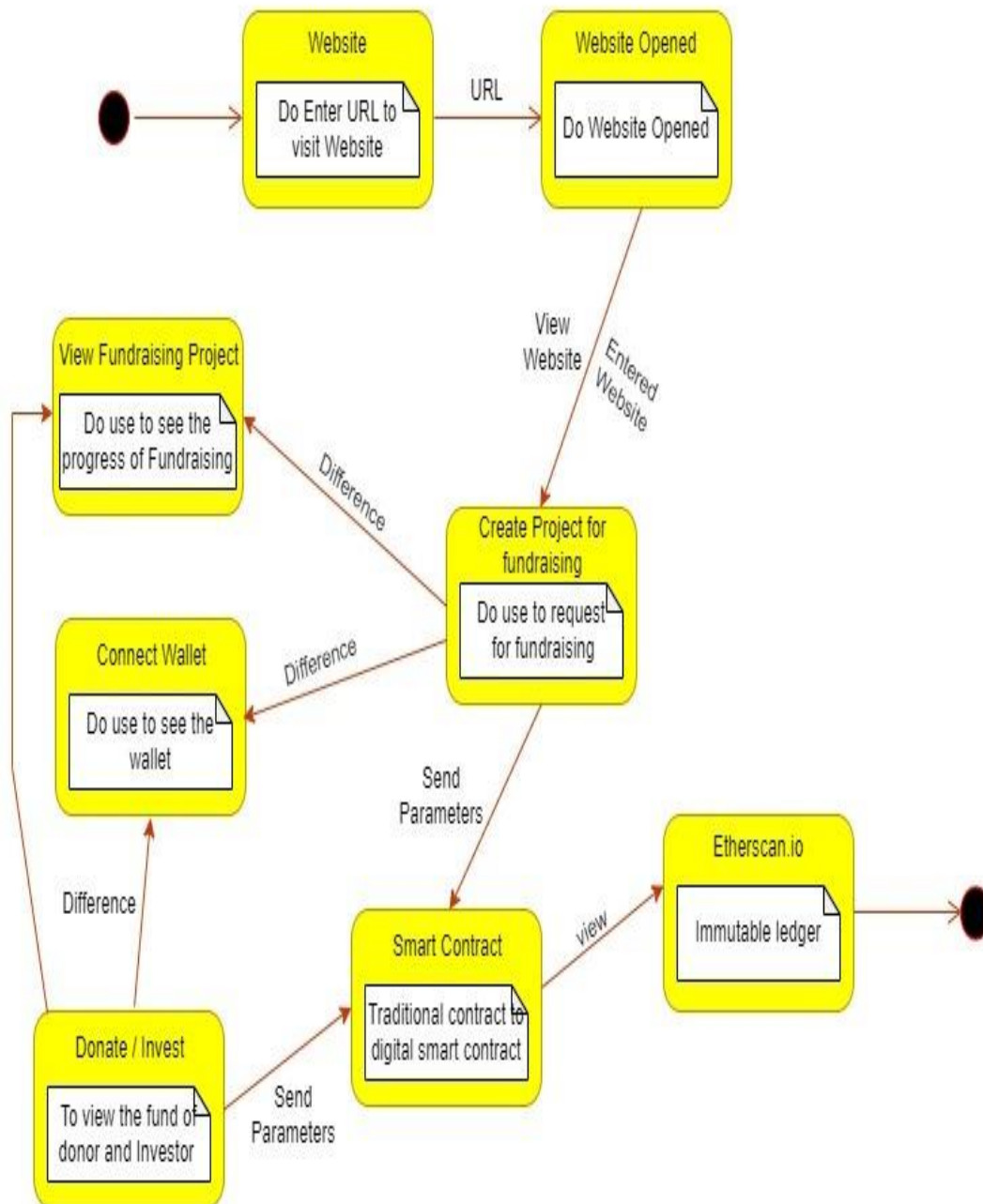


Smart Contract Activity Diagram

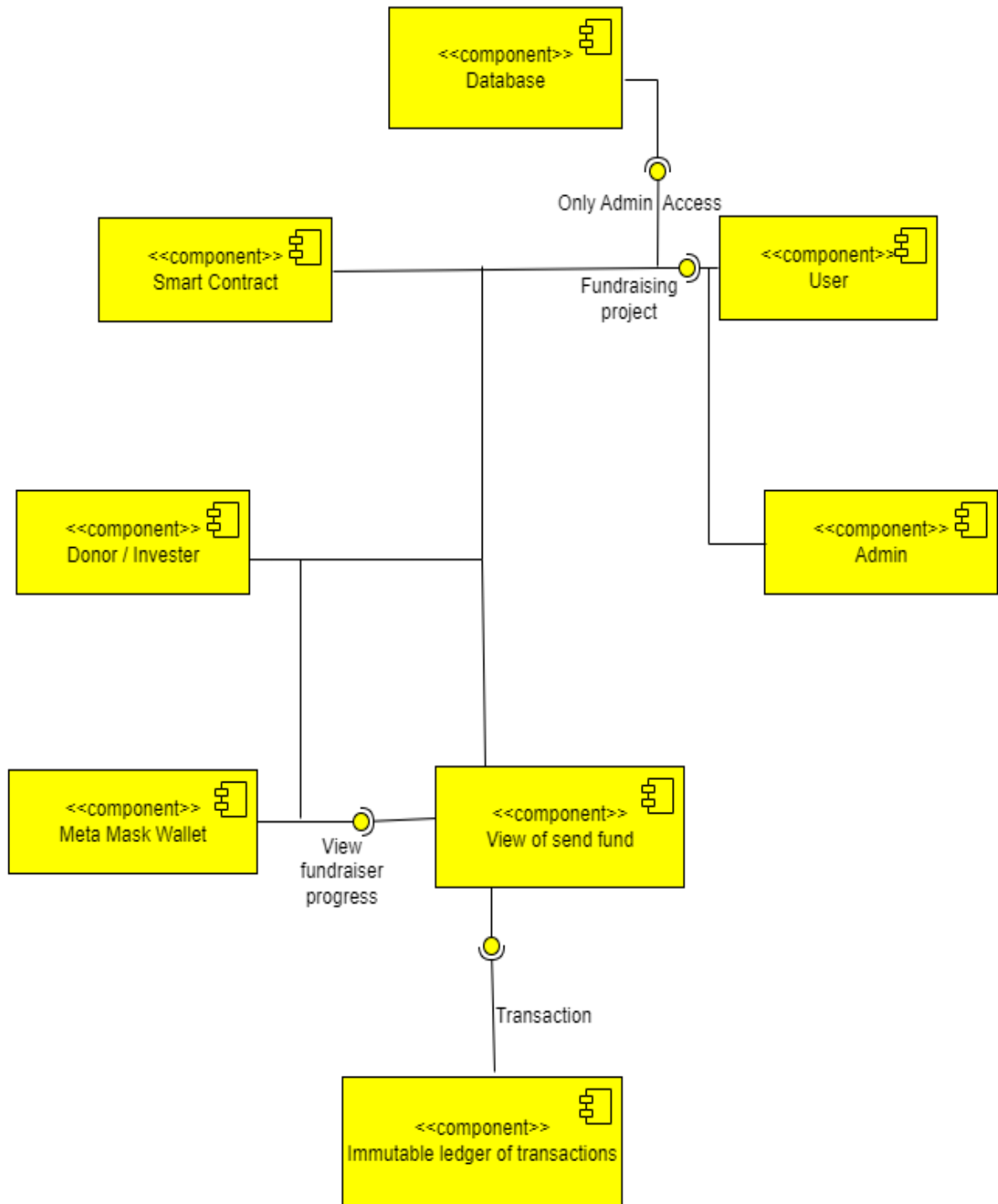


Host Manger/ Developer Activity Diagram

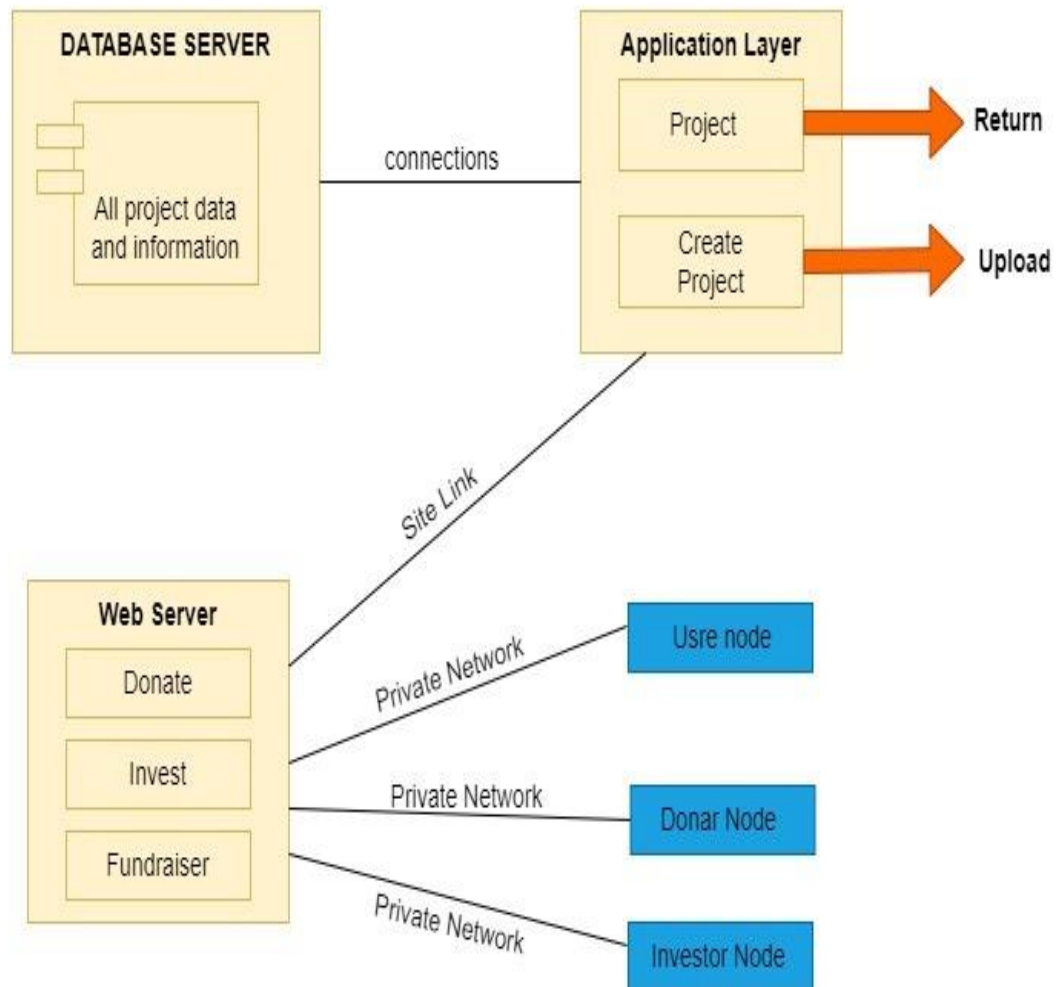
4.8. State Transition Diagram



4.9. Component Diagram

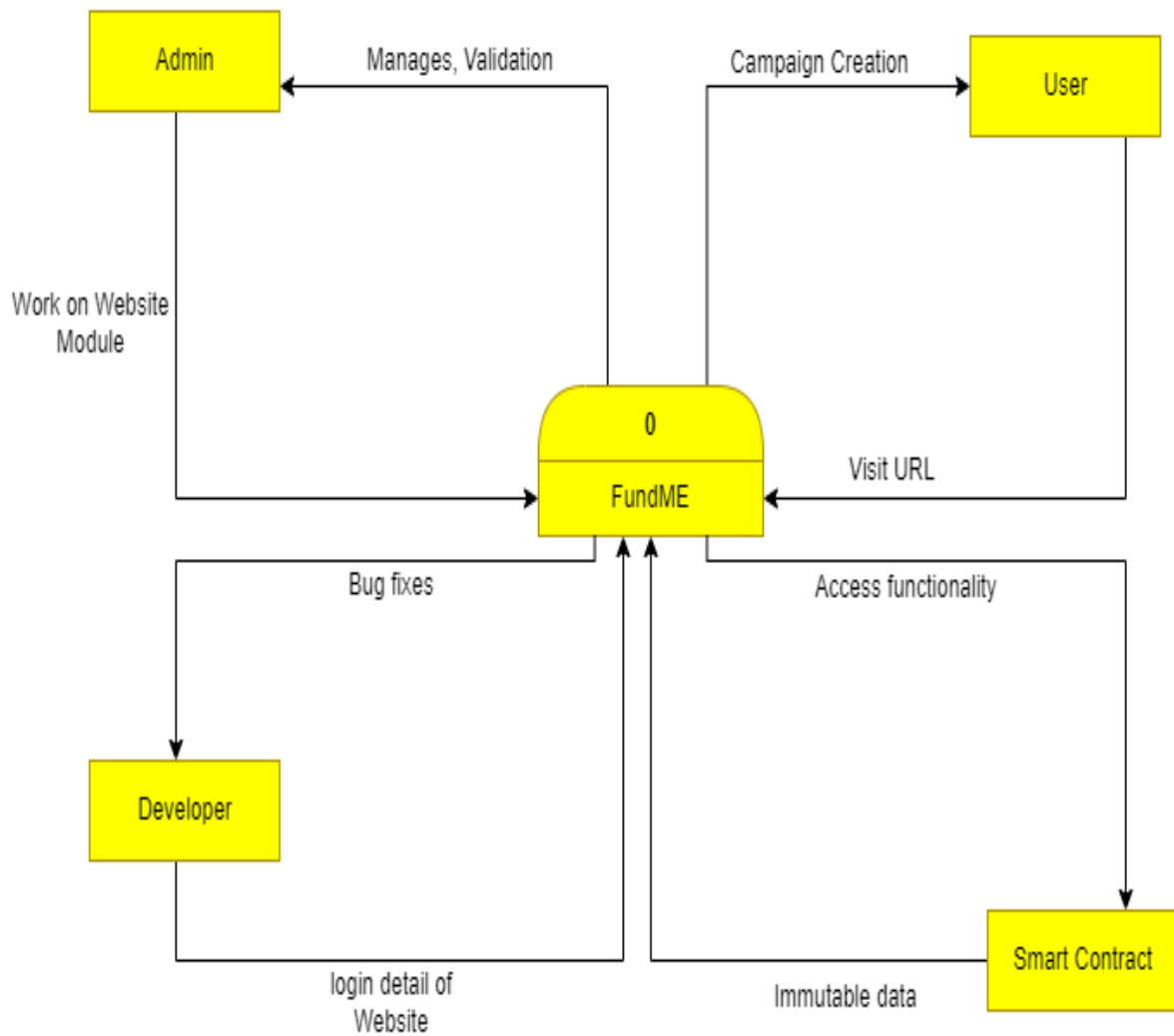


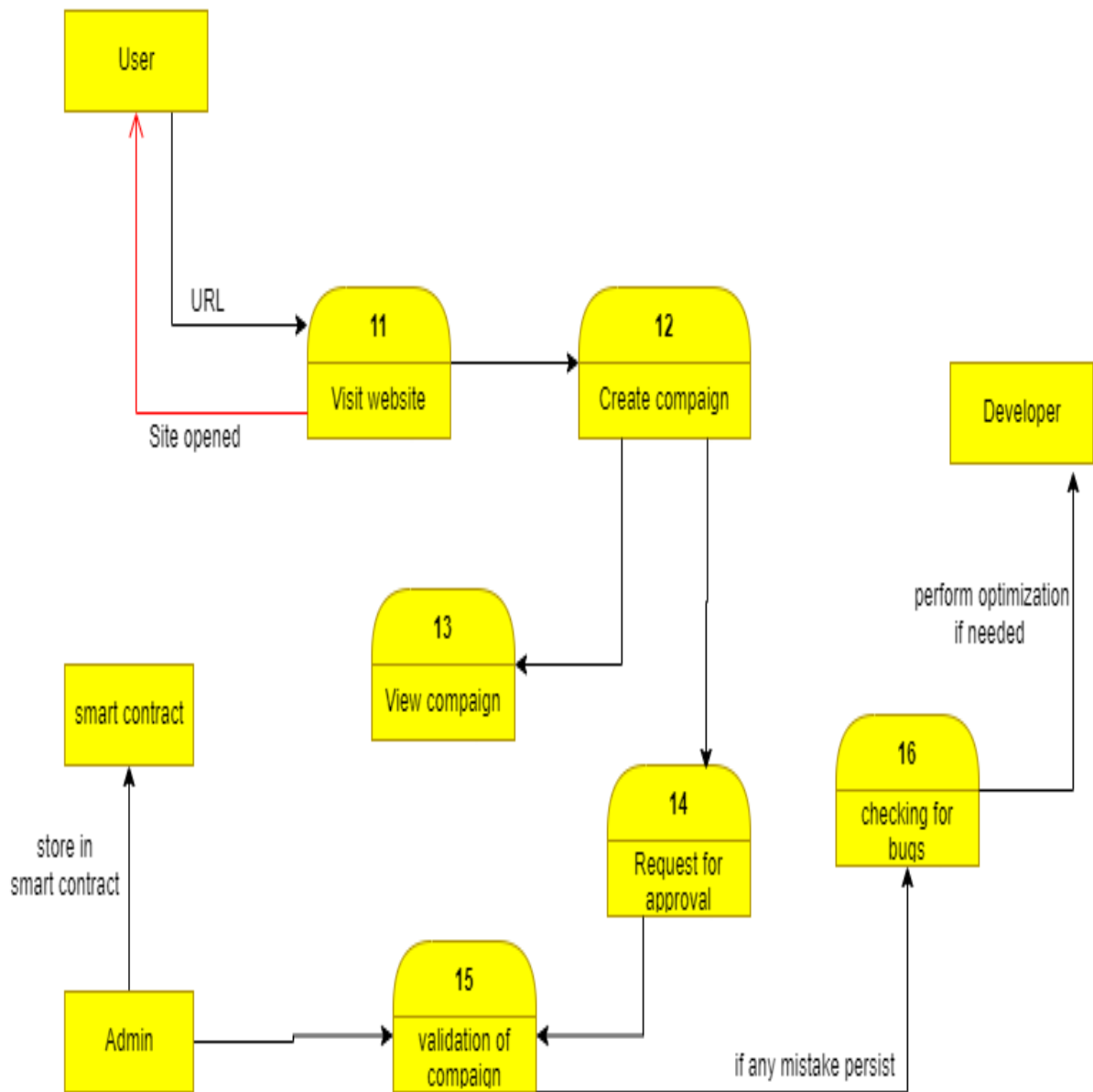
4.10. Deployment Diagram

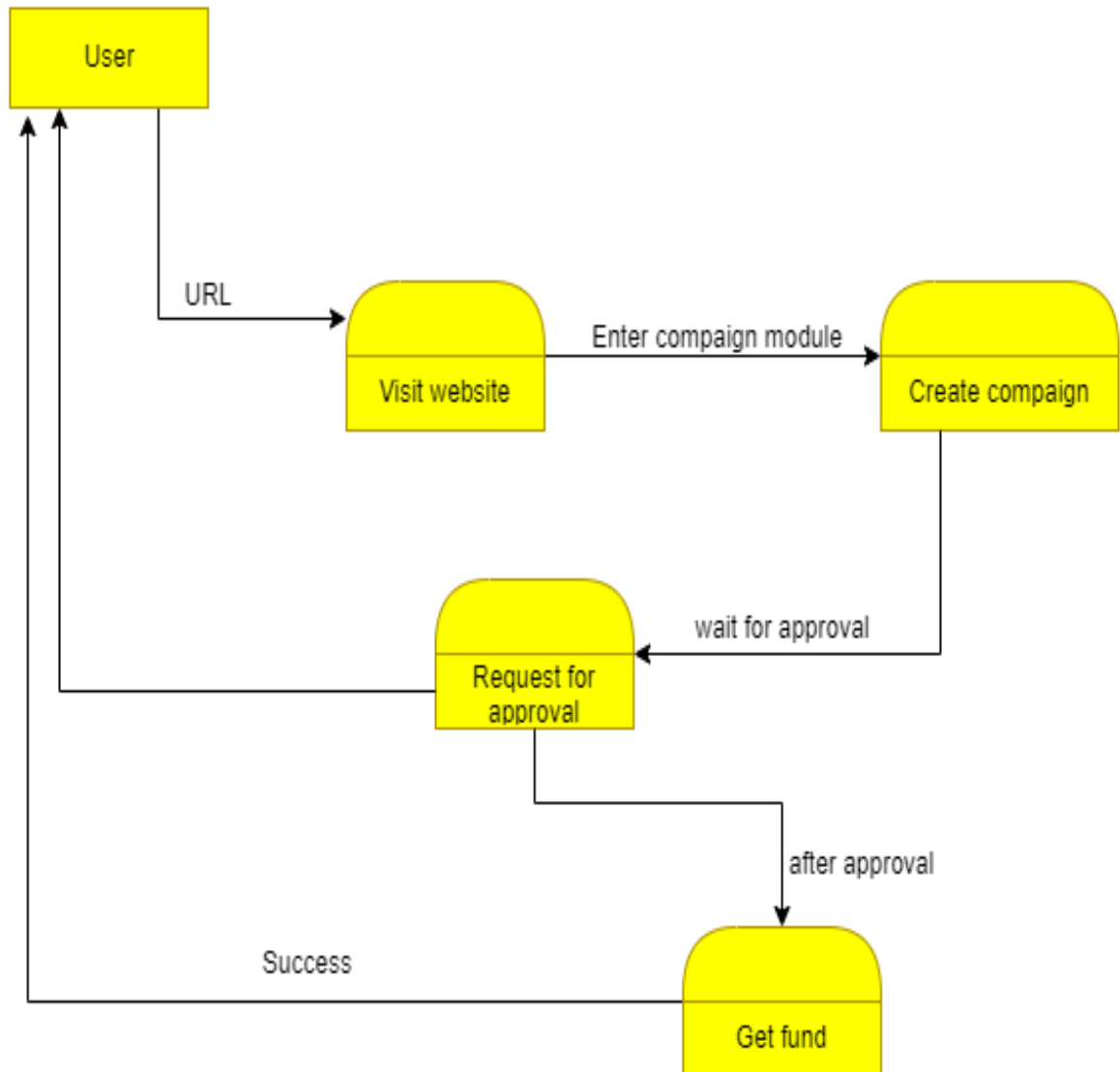


4.11. Data Flow diagram [only if structured approach is used - Level 0 and 1]

Level 0



Level 1

Level 2

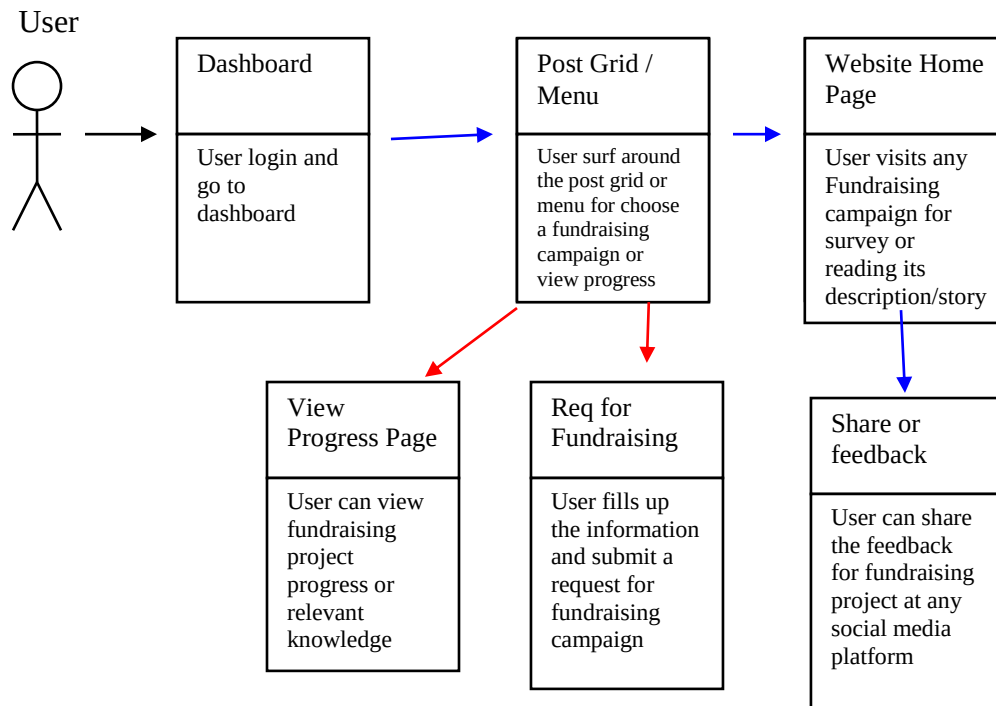
Chapter 5

Implementation

Chapter 5: Implementation

This document describes the project implementation for developing our project Fundme's website. The project implements HTML, CSS, React, MUI, JavaScript, MongoDB, Solidity and other technologies. The project will be capable of running on phones, laptops and other devices. The interface for the project will provide a platform where you can get funds for the project or startup by crowdfunding and donators/investor can transfer their funds and immutable ledgers available for all the transaction publicly.

5.1. Important Flow Control/Pseudo codes



5.2. Components, Libraries, Web Services and stubs

Amazon Web Services is a cloud computing platform that provides hosting and other optimization services to websites and systems. We will be using Cloudflare that is presented by Amazon for enabling CDN and other services like Lazy Load for websites. For web development and services, we are using React.js , Web3.js, Smart Contract and Blockchain, IPFS services respectively. For getting users' analytics and to understand the behavior of users we will using data mining is a technology that is used for identifying patterns and ways from large quantities of data or other repositories. Then with the help of algorithms, it extracts useful information.

5.3. Deployment Environment

We will deploy this project on blockchain but we've build dashboard for our users and setting up a frontend for our website's theme and we are also building a smart contract for backend on blockchain.

5.4. Tools and Techniques

The tools that are required are Listed below

Development Tools

1. Visual Studio Code
2. Hardhat Framework
3. Ganache and Goerli Test Net
4. React.js
5. Smart Contracts

Auditing and Analytics Tools

6. SEO ToolKit
7. Rapid Miner (Data mining)
8. Social Media Marketing
9. Search Console

Front End:

For our website we've used comprehensive languages like HTML5, CSS, JavaScript, and React.js for frontend development. It is responsive for every browser, device and operating system. With increasing end user expectations in relation to the quality, expressiveness and performance of web-based applications, we have given more attention to satisfy the customer. Major campaign creator and donation is done using JavaScript and smart contract that is responsive for every device as well.

Back End:

For our website we've used hosting with Cloudflare CDN for live our website. We will make our database with smart contract on blockchain that is secured with immutable ledger. The CDN serves the purpose of loading media files from the closest data network.

5.5. Best Practices / Coding Standards

- Defined Life Cycle and Milestones
- Defined Organization, Systems, and Roles
- Quality Assurance
- Planned Commitments
- Scope and mission
- Search Engine Optimization – On-Page
- Smart Contract deployment on Blockchain
- Off chain and on chain techniques on Blockchain

5.6. Version Control

We have this complete documentation that completely represents all the factors that are related to our project. In addition, we've used technical pages in our website and the application we've developed almost 60 % that consists of web links to them too. That will consist of all the privacy policies, sitemap and about us pages. The name of our document that we are currently writing is **Project Report Documentation**.

Chapter 6

Testing and Evaluation

Chapter 6: Testing and Evaluation

The testing and evaluation phase is a crucial step in ensuring the quality, functionality, and performance of FundMe Crowdfunding Platform project.

6.1. Use Case Testing

Use Case: 1
User Registration on FundMe Crowdfunding Platform (Connect Wallet)
Preconditions:
The user is not registered on FundMe Crowdfunding Platform.
Steps:
User navigates to the FundMe Crowdfunding Platform registration page.
User fills in the required registration information.
User submits the registration form.
Expected Result:
User receives a registration confirmation from FundMe Crowdfunding Platform.
User can now log in to the FundMe Crowdfunding Platform

Use Case: 2
Creating a Campaign on FundMe Crowdfunding Platform.
Preconditions:
User is logged in to the FundMe Crowdfunding Platform.
Steps:
User navigates to the "Campaign Creation" page on FundMe Crowdfunding Platform.
User fills in the details of the campaign (e.g., title, description, images, target money, dead-line.) on
FundMe Crowdfunding Platform.

User uploads any necessary information FundMe Crowdfunding Platform.
User submits the creation form on FundMe Crowdfunding Platform.
Expected Result:
The campaign is created and visible on the FundMe Crowdfunding Platform. (Anyone can see)
User receives a confirmation of the successful campaign creation from FundMe platform.

Use Case: 3
Donor donates money to campaigns on FundMe platform.
Preconditions:
User is logged in to the FundMe platform.
Sufficient funds are available in the user's account on LDRS.
Steps:
User selects a campaign on FundMe platform.
User reviews the campaign details and image on FundMe platform.
User clicks on the "Donate" button on FundMe platform.
User confirms the donation on FundMe platform.
Expected Result:
The user's account balance on FundMe platform is debited with the purchase amount.
Donor receives a confirmation of the successful donation from FundMe platform.
Donor checks their donations campaigns

Use Case: 4
User checks their donations
Preconditions:
User is logged in to the FundMe platform.
Expected Result:
User can see their donations collected on campaigns

Use Case: 5
Withdrawal of Funds from FundMe platform.
Preconditions:
User is logged in to the FundMe platform.
User has a positive account balance on FundMe platform.
Steps:
User navigates to the "Withdraw Funds" section on FundMe platform.
User enters the withdrawal amount on FundMe platform.
User initiates the withdrawal on FundMe platform.
Expected Result:
The user's account balance on FundMe platform is reduced by the withdrawal amount.
The withdrawn funds are transferred to the user's designated external wallet.
User receives a confirmation of the successful withdrawal from FundMe platform.

6.2. Equivalence partitioning

Equivalence partitioning is a technique used in software testing to divide the input data into groups or partitions that are expected to exhibit similar behavior. Here's equivalence partitioning for our project:

User Registration:

Valid Equivalence Partition:

User provides all required registration information correctly.

Invalid Equivalence Partitions:

- User provides an invalid email address format.
- User leaves a mandatory field blank.
- User enters a password below the required length.
- User enters a password with weak complexity.

Creating a Campaign Creation:

Valid Equivalence Partition:

User provides accurate details, images, and necessary documents for the campaign creation.

Invalid Equivalence Partitions:

User enters a price below the minimum allowed value.

User exceeds the maximum character limit for the description field.

Donates on Campaign:

Valid Equivalence Partition:

User has sufficient funds in their account to donate the selected campaign.

Invalid Equivalence Partitions:

User has insufficient funds in their account to complete the purchase.

The selected campaign is no longer available for donation.

Withdrawal of Funds:**Valid Equivalence Partition:**

Campaign creator requests to withdraw an amount within their available account balance.

Invalid Equivalence Partitions:

User requests to withdraw an amount exceeding their available campaign balance.

6.3. Boundary value analysis

Boundary value analysis is a testing technique that focuses on testing the boundary or edge values of input data. Here's boundary value analysis for our project:

- ***User Registration:***

Boundary Values:

Valid: Test with the minimum and maximum allowed lengths for fields like username, email, and password.

Invalid: Test with values below and above the minimum and maximum lengths for fields.

- ***Creating Campaign Creation:***

Boundary Values:

Valid: Test with the target money allowed price values for the campaign creation.

Invalid: Test with values below and above the target price values.

- ***Donate to Campaign:***

Boundary Values:

Valid: Test with the minimum and target allowed transaction amounts.

Invalid: Test with values below and above the minimum and target allowed transaction amounts.

- ***Withdrawal of Funds:***

Boundary Values:

Valid: Test with the minimum and target withdrawal amounts allowed.

Invalid: Test with values below and above the minimum and target withdrawal amounts.

Data flow testing

Data flow testing is a technique that focuses on the flow of data within a system and ensures that the data is correctly transformed and processed. Here's data flow testing for our project:

User Registration:
Test the flow of user registration data from the registration form to the user database.
Verify that the entered user data is correctly stored in the database.
Verify that any data transformations or validations during the registration process are applied correctly.

Creating Campaign Creation:
Test the flow of land properties data from the listing form to the campaign creation.
Verify that the entered data is correctly used to create the campaign.
Verify that any data transformations or validations during the listing creation process are applied correctly.

Donates to campaign:
Test the flow of transaction data from the user interface to the payment gateway and money transfer process.
Verify that the transaction data is correctly processed, and the payment is successfully completed.
Verify that the transaction is executed accurately based on the transaction data.

Withdrawal of Funds:
Test the flow of withdrawal data from the user interface to the withdrawal process.
Verify that the withdrawal data is correctly processed, and the requested funds are transferred to the user's external wallet.
Verify that any data transformations or validations during the withdrawal process are applied correctly.

6.1. Unit testing

Unit: 1

User Registration Component:

Test that the user registration component correctly validates and stores user data in the database. Test different scenarios, such as valid inputs, invalid inputs, and edge cases for fields like username, email, and password. Test that appropriate error messages are returned for invalid inputs.

Unit: 2

Campaign creation Component:

Test that the campaign creation component accurately creates campaign on the provided properties data. Test different scenarios, such as valid inputs, invalid inputs, and edge cases for fields like title, target money, and other details. Test that the campaign creation process handles errors or exceptions gracefully.

Unit: 4

Withdrawal Component:

Test the withdrawal component to ensure correct processing of fund withdrawals. Test scenarios such as successful withdrawals, insufficient funds, invalid withdrawal amounts, and edge cases for withdrawal limits. Verify that the component updates the relevant data structures (e.g., account balances, transaction records) accurately.

Unit: 5

Error Handling and Exception Handling:

Test the error handling and exception handling mechanisms of your system. Test scenarios that might lead to errors or exceptions and verify that the system handles them gracefully.

6.1. Integration testing

Integration testing for our project involves testing the interaction and seamless integration between components such as user registration, campaign creation, search functionality, donation processes, and fund withdrawal. This testing phase focuses on ensuring that data flows correctly between these components, dependencies are resolved, and the system functions as a cohesive whole. By testing the integration between different modules and verifying the proper exchange of data, you can identify and address any issues or inconsistencies, ensuring that your FundMe platform operates smoothly and delivers a seamless user experience.

6.1. Performance testing

Performance testing is crucial for our project in the context of campaigns. This testing focuses on evaluating the system's speed, scalability, and stability under various conditions. It involves assessing the responsiveness and resource usage of the platform, such as its ability to handle a high volume of users, concurrent transactions, and heavy data loads.

6.2. Stress Testing

Stress testing is an important aspect of ensuring the robustness and reliability of our FundMe platform project for campaign creation. This type of testing involves pushing the system to its limits by subjecting it to extreme conditions, such as high user loads, excessive data volume, or prolonged usage. The goal of stress testing is to identify the system's breaking point and understand its behavior under overwhelming stress. By conducting stress testing, you can determine the system's stability, responsiveness, and ability to recover gracefully from failure. This helps you identify potential weaknesses, optimize performance, and ensure that your NFT marketplace can withstand challenging scenarios without compromising functionality or user experience.

Chapter 7

Summary, Conclusion & Future Enhancements

7.1. Project Summary

The project aims to develop an innovative Funding platform, named FundMe. This platform allows companies, investors, and creators to raise funds by creating decentralized crowdfunding campaigns. This application will provide a track of the records, funds, and transactions that take place on the blockchain. By giving control of invested money to the investors, we will bring transparency and trust to the crowdfunding industry. By blockchain technology, we will be able to create a secure and transparent system for all transactions and contracts that need to be carried out on the platform. Blockchain technology offers a reliable approach to tracing transactions between anonymous participants. Fraud and malfunctions can thus be detected quickly.

7.2. Achievements and Improvements

Achievements:

Successful development and launch of the FundMe platform, providing a secure and efficient platform for get donations. Implementation of user registration, campaign creation, enabling seamless interactions for users. Integration of blockchain technology to ensure transparent and tamper-proof transactions, enhancing trust and security within the marketplace. Establishment of a user-friendly interface and intuitive user experience, facilitating smooth navigation and engagement with the platform.

Improvements:

Enhancing scalability and performance of the platform to accommodate increasing user demand and large transaction volumes. Further refining the search functionality to provide more comprehensive and accurate results based on specific campaign criteria. Continuous improvement of the user interface and experience to ensure an intuitive and seamless donation experience for users. Strengthening the blockchain integration and security measures to fortify the platform against potential vulnerabilities and attacks. Expanding the platform reach and user base through targeted marketing and strategic partnerships with relevant stakeholders in the land industry. Incorporating

user feedback and suggestions to continually enhance and refine the features and functionalities of the FundMe, ensuring it remains at the forefront of innovation in the digital donation space.

7.3. Critical Review

The development and launch of the FundMe platform for getting donations as non- digital assets mark a significant achievement. The platform successfully addresses the challenges of unclear campaigns and fraudulent transactions by leveraging blockchain technology and implementing robust authentication mechanisms. The user-friendly interface and intuitive user experience contribute to a seamless trading experience. The integration of advanced search functionality and secure transactions further enhance the platform's usability. However, there are areas that require improvement. Enhancing scalability and performance is crucial to handle increasing user demand and transaction volumes effectively. Refining the search functionality to provide more accurate and comprehensive results based on specific campaign criteria will improve user satisfaction. Strengthening blockchain integration and security measures is vital to ensure the platform's resilience against potential vulnerabilities. Expanding the platform reach and user base through targeted marketing and strategic partnerships will contribute to its long-term success. Continual incorporation of user feedback and suggestions is necessary to refine and enhance the platform's features and functionalities. Overall, the FundMe platform demonstrates promising potential but should focus on addressing these critical areas to maximize its impact and value proposition in the digital land asset space.

7.4. Lessons Learnt

Importance of Robust Security: The project highlighted the criticality of implementing strong security measures, especially when dealing with sensitive transactions and valuable assets like land properties. Lessons were learned about the significance of blockchain technology in ensuring transparency, immutability, and secure transactions.

User-Centric Design: The emphasis on user experience and intuitive design proved to be a valuable lesson. Understanding user needs, preferences, and pain points helped in creating a user-friendly interface and seamless navigation, enhancing user engagement and satisfaction.

Continuous Iteration and Improvement: The project emphasized the importance of an iterative development approach. Regularly seeking user feedback, analyzing market trends, and incorporating improvements based on insights enabled the project to stay relevant, competitive, and responsive to evolving user demands.

7.5. Future Enhancements/Recommendations

Integration with Land Data Providers: Enhancing the marketplace by integrating with land data providers can provide real-time and accurate information about campaign creators, including legal status, ownership records, and historical data. This integration would enrich the user experience and increase trust in the platform.

Enhanced Collaboration with Regulatory Bodies: Strengthening partnerships and collaboration with regulatory bodies, land administration agencies, and legal entities can help streamline the process of verifying campaign ownership, resolving disputes, and ensuring compliance within the platform. This collaboration would foster credibility and trust among users.

Expansion to Global Markets: Considering the potential of getting campaign donations globally, expanding the marketplace's reach to international markets would unlock new opportunities and attract a wider user base. Tailoring the platform to cater to different regions' legal frameworks and market dynamics is essential for successful expansion.

Integration of Smart Contracts: Implementing smart contracts within the FundMe place can automate and enforce contractual agreements, such as donations, royalties, and revenue sharing. This

would increase efficiency, reduce reliance on intermediaries, and provide enhanced transparency for users.

Scalability and Performance Optimization: Continuous focus on enhancing the platform's scalability and performance is crucial to accommodate increased user traffic, large transaction volumes, and growing demand. Regular stress testing, optimization of infrastructure, and leveraging technological advancements can ensure a seamless user experience.

Appendix A: User Manual

This appendix provides detailed instructions and guidance for users on how to effectively navigate and utilize the LDRS (Land Properties NFTs Marketplace) platform. It includes step-by-step procedures, explanations of key features, and tips for optimizing user experience. The user manual aims to empower users with the knowledge and understanding needed to make the most of the platform's functionalities and ensure a seamless trading experience.

A.1. First Level Heading

This section within the user manual delves into specific topics or areas of interest related to the FundMe platform. It provides comprehensive information, guidelines, and best practices for users to follow when engaging with the marketplace. The content is presented in a clear and concise manner, with appropriate headings and subheadings to facilitate easy navigation and comprehension.

A.1.1. Second Level Heading

Within each section, there are subheadings that further divide the content into more specific areas. These subheadings help users find the relevant information they are looking for quickly and efficiently. The instructions and explanations provided are accompanied by supporting visuals, where applicable, to enhance clarity and understanding.

Appendix B: Administrator Manual

This appendix is dedicated to providing detailed instructions and guidelines specifically for administrators of the FundMe platform. It covers topics such as user management, system configuration, security settings, and other administrative tasks. The administrator manual equips individuals responsible for overseeing the platform with the knowledge and tools necessary to effectively manage and maintain the system.

Appendix C: Information / Promotional Material

This section includes various promotional materials, such as brochures, flyers, standees, banners, and other informative resources related to the FundMe platform. These materials are designed to create awareness, generate interest, and promote the features and benefits of the platform. The appendix provides a collection of marketing collateral that can be used to effectively communicate the value proposition of the LDRS platform to potential users and stakeholders.

Reference and Bibliography

1. [2] Journal of King Saud University - Computer and Information Sciences, 34(9), 6855-6871. Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system
2. Musamih, Ahmad & Dirir, Ahmed & Yaqoob, Ibrar & Salah, Khaled & Jayaraman, Raja & Puthal, Deepak. (2022). NFTs in Smart Cities: Vision, Applications, and Challenges. IEEE Consumer Electronics Magazine. PP. 1-14. 10.1109/MCE.2022.3217660.
3. Elmessiry, A., Elmessiry, M., & Bridgesmith, L. (2021). NFT Student Teacher Incentive System (NFT-STIS). In Proceedings of the EDULEARN21 Conference (pp. [page numbers]). ISBN: 978-84-09-31267-2.
4. Regner, Ferdinand & Schweizer, André & Urbach, Nils. (2019). NFTs in Practice – NonFungible Tokens as Core Component of a Blockchain-based Event Ticketing Application.
5. Serrano, Will. (2022). Real Estate Tokenisation via Non Fungible Tokens. 81-87. 10.1145/3532640.3532651.
6. Hafsa Rasheed, Habib Ahmad, Attiya Yasmin Javid & Idrees Khawaja (2022) Is real estate a real diversifier in Pakistan? An ARDL approach, Cogent Business & Management, 9:1, 2082013, DOI: 10.1080/23311975.2022.2082013
7. Vazquez-Abad, F., Vidal-Carreras, P. I., & O'Leary, J. (2022). CryptoKitties and NBA Top Shot: From kittens to LeBron James. Frontiers in Blockchain, 5, 129.

2.7.1. References

Article Title: *Smart Contract and Blockchain for Crowdfunding Platform*

URL Link: <https://www.warse.org/IJATCSE/static/pdf/file/ijatcse83932020.pdf>

URL Link: <https://www.kickstarter.com>

URL Link: <https://www.indiegogo.com>

URL Link: <https://www.makeuseof.com/decentralized-platforms-crypto-crowdfunding>

URL Link: <https://republic.co>

URL Link: <https://www.seedinvest.com>

URL Link: <https://daomaker.com>

URL Link: <https://wefunder.com>

URL Link: <https://blog.cryptostars.is/crypto-crowdfunding-platform-development-54cf96ff3e4>

URL Link: <https://nevonprojects.com/blockchain-based-crowdfunding-platform>

URL Link: <https://www.polkastarter.com>

URL Link: <https://www.gofundme.com>

URL Link: <https://www.gemini.com/cryptopedia/blockchain-crowdfunding-blockchain-startups>

URL Link: <https://www.fundedbyme.com>

URL Link: <https://www.coinlist.co>

URL Link: <https://www.geeksforgeeks.org/how-does-blockchain-support-crowdfunding>

URL Link: <https://ieeexplore.ieee.org/document/9640888>

URL Link: <https://www.startengine.com>

URL Link: <https://icoholder.com>

URL Link: <https://www.fundable.com>