

**A
PROJECT REPORT
ON
Crypto Backer**

Submitted by
**Jay Butani(21IT407)
Shubham Vanani(22IT603)
Prince Beladiya(22IT605)
Sujal Vadgama (22IT608)**

**For Partial Fulfillment of the Requirements for Bachelor of Technology in
Information Technology**

**Guided by
Prof. Vishal Polara**

May, 2024



**Information Technology Department
Birla Vishvakarma Mahavidyalaya Engineering College
(An Autonomous Institution)
Vallabh Vidyanagar – 388120**



Birla Vishvakarma Mahavidyalaya Engineering College
(An Autonomous Institution)
Information Technology Department
AY: 2023-24, Semester II

CERTIFICATE

This is to certify that the project work entitled **Crypto Backer** has been successfully carried out by **Jay Butani(21IT407), Shubham Vanani(22IT603), Prince Beladiya(22IT605), Sujal Vadgama(22IT608)** for the subject **Major Project (4IT33)** during the academic year 2023-24, Semester-II for partial fulfilment of Bachelor of Technology in Information Technology. The work carried out during the semester is satisfactory.

Prof. Vishal Polara
IT Department
BVM

Dr. Keyur Brahmhatt.
Head, IT Department
BVM

Acknowledgement

I would like to express my sincere gratitude to everyone who contributed to the successful completion of this project and the preparation of the Software Requirements Specification (SRS) document for **CryptoBacker**. My deepest appreciation goes to my academic advisor, Professor Vishal Polara, for their invaluable guidance, support, and encouragement throughout this project. I am also grateful to the faculty and staff at Birla Vishvakarma Mahavidyalaya for providing the necessary resources and fostering an environment conducive to research and innovation. Special thanks to my peers and colleagues for their insightful feedback and collaboration. Lastly, I extend my heartfelt thanks to my friends for their unwavering support and understanding, which motivated me to persevere through the challenges. This project would not have been possible without the collective effort and support of all these individuals.

Abstract

The CryptoBacker project aims to create a secure and transparent crowdfunding platform leveraging the Ethereum blockchain. This platform enables users to create and fund campaigns with Ethereum, ensuring minimized fraud risks through smart contract functionality. Campaign creators undergo a rigorous verification process, with campaigns classified as either full or partial fund release based on their provided documentation and prototype. Contributions are held in smart contracts and converted to stablecoins (USDT/USDC) to mitigate price volatility. Funds are released according to the campaign's status, ensuring that backer's contributions are used appropriately. This platform combines the decentralization and security of blockchain technology with the innovative model of crowdfunding, providing a reliable and efficient solution for both campaign creators and backers.

List of Figures

Figure 3.1: Timeline chart.....	14
Figure 3.3: Level 0 DFD Diagram	22
Figure 3.4: Level 1 DFD Diagram	22
Figure 3.5: Level 2 DFD Diagram	23
Figure 3.6: Class Diagram.....	24
Figure 3.7: ER Diagram	25
Figure 3.8: Sequence Diagram	26
Figure 3.9: Activity Diagram	27

List of Tables

Table 1: Table of Abbreviations.....	VI
Table 3.1: User Table	28
Table 3.2: Campaign Table	29
Table 3.3: Admin Table	30
Table 3.4: Fund Contribution Table.....	31
Table 3.5: Transaction Table.....	32

List of Abbreviations

Abbreviation	Full Form
CPU	Central Processing Unit
RAM	Random Access Memory
SSD	Solid-State Drive
UI	User Interface
API	Application Programming Interface
DFD	Data Flow Diagram
SRS	Software Requirements Specification
i5	Intel Core i5 Processor
DApp	Decentralized Application
EVM	Ethereum Virtual Machine
DeFi	Decentralized Finance
USDT	Tether (a type of stablecoin)
USDC	USD Coin (a type of stablecoin)
DAO	Decentralized Autonomous Organization
CryptoBacker	Crypto Backer

Table 1: Table of Abbreviations

Table of Contents

Chapter 1: Introduction	2
1.1 Brief Overview of the Work:	2
1.2 Project Objective:.....	3
1.3 Project Scope:	3
1.4 Project Modules:.....	4
1.5 Project Hardware/Software Requirement:.....	6
Chapter 2: Literature Review	7
2.1 Literature Review	7
Chapter 3: System Analysis & Design.....	11
3.1 Comparison of Existing Applications with Crypto Backer:.....	11
3.2 Project Feasibility Study:.....	13
3.3 Project Timeline chart:.....	14
3.4 Detailed Modules Description:	14
3.5 Project SRS:	21
3.5.1 Use Case Diagram:.....	21
3.5.2 Data Flow Diagram:.....	22
3.5.3 Class Diagram:.....	24
3.5.4 Entity Relationship Diagram:.....	25
3.5.5 Sequence Diagram:	26
3.5.6 Activity Diagram:	27
3.6 Data Dictionary:	28

1.1 Brief Overview of the Work:

- The CryptoBacker project is designed to develop a decentralized crowdfunding platform using the Ethereum blockchain. The platform facilitates campaign creation and funding through Ethereum, with built-in mechanisms to ensure security and minimize fraud.
- ❖ **User Registration and Verification:**
 - Users sign up by providing personal details and government-authorized proof. Admins verify these details to approve or reject the user profiles, ensuring only legitimate users can create campaigns.
- ❖ **Campaign Creation and Approval:**
 - Verified users can create campaigns by submitting detailed descriptions, goals, rewards, deadlines, and supporting documents like patents. Admins then review and verify these campaigns, marking them as either full or partial fund release based on the provided documentation and prototype.
- ❖ **Funding Mechanism:**
 - Approved campaigns can receive funds from backers in Ethereum. These funds are held in a smart contract and swapped to stablecoins (USDT/USDC) to prevent value fluctuations.
- ❖ **Fund Release:**
 - For full fund release campaigns, funds are released after the campaign deadline. For partial fund release campaigns, funds are disbursed in stages, contingent on the campaign creator's progress and further verification by the admin.
- ❖ **Fraud Prevention and Fund Redistribution:**
 - The platform includes measures to handle suspicious activity. If fraud is detected, funds stored in the campaign contract are redistributed to backers, ensuring a fair and secure environment for all participants.
- This project utilizes the strengths of block chain technology - transparency, security, and decentralization to create a robust and reliable crowdfunding platform.

1.2 Project Objective:

❖ Create a Secure and Transparent Crowdfunding Platform:

- Develop a user-friendly platform leveraging Ethereum blockchain technology to ensure transparency, security, and trust in all funding transactions.

❖ Enable Campaign Creation and Management:

- Allow users to create detailed fundraising campaigns, including titles, descriptions, goals, deadlines, and supporting documents, with a seamless submission and management process.

❖ Implement Rigorous Verification Processes:

- Establish a thorough verification system for user profiles and campaigns, reducing the risk of fraudulent activities and ensuring only credible campaigns receive funding.

❖ Ensure Secure Fund Handling:

- Utilize smart contracts to hold contributed funds securely and convert them to stable coins (USDT/USDC) to avoid price fluctuations, ensuring that funds are managed and released appropriately.

❖ Facilitate Partial and Full Fund Releases:

- Design a flexible fund release mechanism where funds can be disbursed fully or partially based on campaign progress and milestones, as verified by the platform admins.

❖ Develop an Admin Dashboard for Effective Oversight:

- Create a comprehensive admin interface to manage user verifications, campaign approvals, and fund release requests, ensuring efficient and effective oversight of platform activities.

1.3 Project Scope:

❖ Startup Funding:

- Entrepreneurs and startups can use the platform to raise capital for new business ideas and product development, providing detailed plans and prototypes to attract backers.

❖ Creative Projects:

- Artists, filmmakers, musicians, and other creatives can seek funding for their projects, offering rewards and exclusive content to backers in exchange for their support.

❖ Community and Social Initiatives:

- Non-profit organizations and community groups can raise funds for social causes, community projects, and charitable initiatives, ensuring transparency and accountability to their donors.

❖ Innovation and Research:

- Researchers and innovators can secure funding for scientific research, technological advancements, and innovative solutions, providing progress updates and outcomes to backers.

❖ Education and Training:

- Educators and institutions can launch campaigns to fund educational programs, scholarships, and training workshops, engaging the community in their efforts to provide quality education.

1.4 Project Modules:

❖ User Management Module

- **User Registration and Authentication:** Handles the user signup process, including capturing personal details, email verification, and secure login.
- **Profile Management:** Allows users to view and update their profile information, including personal details and verification documents.
- **User Verification:** Admins review and approve user profiles, ensuring authenticity and compliance with platform standards.

❖ Campaign Management Module

- **Campaign Creation:** Enables users to create campaigns by providing necessary details like title, description, rewards, goal, deadline, images, and verification documents.
- **Campaign Submission:** Handles the submission process, setting the campaign status to pending until verified by the admin.

- **Campaign Verification and Approval:** Admins review campaign details and verification documents, approving or rejecting campaigns based on compliance with platform guidelines.

❖ Campaign Discovery Module

- **Campaign Browsing:** Allows users to browse and search for campaigns based on various filters like category, status, and popularity.
- **Campaign Details:** Provides detailed information about each campaign, including its current status (pending or approved), funding progress, and rewards.

❖ Funding Management Module

- **Fund Contribution:** Enables backers to contribute to approved campaigns using Ethereum, with the option to convert contributions to stablecoins (USDT/USDC) to avoid price fluctuations.
- **Smart Contract Integration:** Manages the creation and interaction with smart contracts to securely hold and manage contributed funds.
- **Fund Release:** Handles the process of releasing funds to campaign creators based on admin approval, either fully or partially, as per the campaign's progress and milestones.

❖ Admin Dashboard Module

- **User Verification Management:** Allows admins to review and approve or reject user verification requests.
- **Campaign Approval Management:** Admins can review campaign submissions, approve or request changes, and manage the overall campaign approval process.
- **Fund Release Management:** Admins can review and approve fund release requests from campaign creators, ensuring compliance with platform rules.
- **Campaign and User Monitoring:** Provides tools for admins to monitor active campaigns and user activities, ensuring platform integrity and security.

❖ Payment and Transaction Module

- **Payment Processing:** Manages the processing of Ethereum payments and the conversion to stable coins.

- **Transaction Tracking:** Keeps a detailed record of all transactions, including contributions, fund releases, and refunds, ensuring transparency and traceability.
- **Refund Management:** Handles the process of refunding contributions to backers in case of campaign cancellation or failure to meet verification standards.

1.5 Project Hardware/Software Requirement:

❖ Software:

- **Frontend:** ReactJS with Redux, Axios, web3.js, and a UI framework like Tailwind and Flowbite.
- **Backend:** Node.js with Express.js, MongoDB, and RESTful APIs.
- **Smart Contracts:** Solidity with Hardhat, and OpenZeppelin libraries for security.
- **Development Tools:** GitHub, Visual Studio Code, npm, Chai for testing.

❖ Hardware:

- **Development Machines:** Intel i5 or higher, 8GB RAM (16GB recommended), 256GB SSD (512GB recommended), Windows/macOS/Linux.
- **Blockchain Nodes:** Multi-core CPU, 8GB RAM (16GB recommended), 1TB SSD, high-speed internet.

2.1 Literature Review

1. Ethereum Whitepaper: A Next-Generation Smart Contract and Decentralized Application Platform

- **Author:** Vitalik Buterin
- **Summary:** The Ethereum Whitepaper, authored by Vitalik Buterin, provides a foundational understanding of Ethereum, a decentralized platform that enables smart contracts and decentralized applications (DApps). The whitepaper outlines the architecture of Ethereum, its consensus mechanism, and the Ethereum Virtual Machine (EVM). It emphasizes the flexibility of Ethereum's scripting language, which allows developers to create various decentralized applications, including crowdfunding platforms like CryptoBacker. This literature served as a crucial resource for understanding the technical underpinnings and potential applications of Ethereum in building secure and transparent crowdfunding mechanisms.
- **Citation:** Buterin, V. (2013). Ethereum Whitepaper: A Next-Generation Smart Contract and Decentralized Application Platform. Retrieved from <https://ethereum.org/en/whitepaper/>

2. Solidity Documentation

- **Author:** Ethereum Foundation
- **Summary:** The Solidity documentation provided by the Ethereum Foundation offers an in-depth guide to Solidity, the primary programming language used for writing smart contracts on the Ethereum blockchain. It covers the syntax, features, and best practices for writing secure and efficient smart contracts. This resource was indispensable in the development of the smart contracts for CryptoBacker, particularly for implementing the logic for holding and releasing funds based on campaign verification and milestone achievements.
- **Citation:** Ethereum Foundation. (n.d.). Solidity Documentation. Retrieved from <https://docs.soliditylang.org/en/v0.8.6/>

3. Decentralized Finance: On Blockchain- and Smart Contract-based Financial Markets

- **Authors:** Fabian Schär, Aleksander Berentsen
- **Summary:** This paper explores the concepts of decentralized finance (DeFi), focusing on blockchain-based financial systems and the role of smart contracts in enabling trustless transactions. The authors discuss various DeFi applications, including decentralized exchanges and lending platforms, highlighting the benefits of transparency, security, and accessibility. The insights from this paper informed the design of CryptoBacker's funding mechanism, ensuring that contributions are securely managed and disbursed using smart contracts to minimize fraud risks.
- **Citation:** Schär, F., & Berentsen, A. (2020). Decentralized Finance: On Blockchain- and Smart Contract-based Financial Markets. Federal Reserve Bank of St. Louis Review, Second Quarter 2020, 99-112. Retrieved from <https://doi.org/10.20955/r.102.99-112>

4. Crowdfunding: A Successful Way of New Venture Financing?

- **Authors:** Lambert, Thomas, Schwienbacher, Armin
- **Summary:** This research paper examines the effectiveness of crowdfunding as a financing method for new ventures. It analyzes various crowdfunding models, the motivations of backers, and the success factors for campaigns. The paper provides valuable insights into the challenges and opportunities of crowdfunding, such as the importance of trust and transparency. These findings were crucial in designing CryptoBacker's verification processes and the implementation of fund release mechanisms to build trust and ensure the platform's credibility.
- **Citation:** Lambert, T., & Schwienbacher, A. (2010). Crowdfunding: A Successful Way of New Venture Financing? Journal of Business Venturing Insights. Retrieved from https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1578175

5. Stablecoins: Risks, Potential and Regulation

- **Authors:** Raphael Auer, Stijn Claessens
- **Summary:** This paper provides an analysis of stablecoins, digital currencies that aim to maintain a stable value relative to a reference currency. The authors discuss the different types of stablecoins, their use cases, and the associated risks and regulatory challenges. The insights from this paper were essential in designing CryptoBacker's fund management system, where contributions are converted to stablecoins (USDT/USDC) to mitigate the impact of cryptocurrency price volatility. This ensures that the value of the funds raised remains stable and predictable.
- **Citation:** Auer, R., & Claessens, S. (2018). Stablecoins: Risks, Potential and Regulation. Bank for International Settlements. Retrieved from <https://www.bis.org/publ/work905.htm>

6. The DAO Hack and the Future of Ethereum

- **Author:** Matt Levine
- **Summary:** This article delves into the infamous DAO hack, a significant event in Ethereum's history where a vulnerability in a smart contract was exploited, leading to substantial financial losses. The author discusses the implications of the hack for Ethereum's security and governance. Understanding the DAO hack was crucial in the development of CryptoBacker, guiding the implementation of rigorous security measures and thorough code audits to prevent similar vulnerabilities in the platform's smart contracts.
- **Citation:** Levine, M. (2016). The DAO Hack and the Future of Ethereum. Bloomberg. Retrieved from <https://www.bloomberg.com/opinion/articles/2016-06-17/the-dao-hack-and-the-future-of-ethereum>

7. Blockchain Basics: A Non-Technical Introduction in 25 Steps

- **Author:** Daniel Drescher
- **Summary:** This book provides a comprehensive introduction to blockchain technology in a non-technical manner, covering the fundamental concepts and mechanisms that underpin blockchain systems. It includes explanations of cryptographic principles, consensus mechanisms, and the functioning of smart contracts. The book served as a foundational resource for understanding the broader context of blockchain technology and its application in building a secure and transparent crowdfunding platform like CryptoBacker.
- **Citation:** Drescher, D. (2017). Blockchain Basics: A Non-Technical Introduction in 25 Steps. Apress. Retrieved from <https://link.springer.com/book/10.1007/978-1-4842-2604-9>

3.1 Comparison of Existing Applications with Crypto Backer:

❖ Existing Applications

1. Kickstarter

- **Platform:** Centralized crowdfunding platform.
- **Campaign Approval:** Projects are reviewed by Kickstarter staff.
- **Funding Model:** All-or-nothing funding (if the goal is not met, funds are returned to backers).
- **Currency:** Traditional fiat currencies.
- **Fees:** Kickstarter takes a percentage fee from the funds raised.
- **Payment Processing:** Traditional payment methods (credit cards, bank transfers).
- **Fund Release:** Funds are released to project creators after the campaign ends successfully.

2. GoFundMe

- **Platform:** Centralized crowdfunding platform, primarily for personal causes and charity.
- **Campaign Approval:** Less stringent review process.
- **Funding Model:** Flexible funding.
- **Currency:** Traditional fiat currencies.
- **Fees:** GoFundMe takes a lower percentage fee compared to other platforms.
- **Payment Processing:** Traditional payment methods (credit cards, bank transfers).
- **Fund Release:** Funds are released to campaign creators on an ongoing basis.

❖ Crypto Backer

1. Platform: Decentralized crowdfunding platform on the Ethereum blockchain.

- **Advantage:** Enhanced transparency and security due to the decentralized nature of blockchain technology.

2. Campaign Approval: Verification process for business ideas and models, with campaigns requiring admin approval before going live.

- **Advantage:** Minimizes the risk of fraudulent campaigns and ensures higher quality projects.
3. **Funding Model:** Funds are held in a smart contract and released fully or partially based on campaign progress and admin approval.
- **Advantage:** Reduces the risk of misuse of funds and ensures that creators meet their milestones.
4. **Currency:** Contributions in Ethereum, converted to stablecoins like USDT/USDC to avoid price volatility.
- **Advantage:** Leverages the benefits of cryptocurrency while mitigating risks associated with price fluctuations.
5. **Fees:** Transparent and potentially lower transaction fees due to the use of blockchain technology.
- **Advantage:** Lower fees can attract more backers and creators to the platform.
6. **Payment Processing:** Ethereum blockchain transactions with smart contract integration.
- **Advantage:** Ensures secure and transparent handling of funds, with immutable records of all transactions.
7. **Fund Release:** Funds are released based on the progress and verification of project milestones, with partial disbursements possible.
- **Advantage:** Provides an additional layer of security and accountability, ensuring that funds are used as intended.

3.2 Project Feasibility Study:

❖ Technical Feasibility

- **Technology Stack:** Using ReactJS for the frontend and Node.js for the backend, combined with Ethereum blockchain for transactions and smart contracts.
- **Skills Required:** Developers need expertise in web development, blockchain technology, and smart contract programming (Solidity).

❖ Operational Feasibility

- **User Base:** Aimed at entrepreneurs seeking funding and backers looking to support innovative projects.
- **Verification Process:** Implementing a thorough verification process for campaign approval to ensure quality and reduce fraud.
- **Fund Management:** Using smart contracts to hold and release funds based on project milestones, ensuring accountability.

❖ Market Feasibility

- **Demand:** Growing interest in blockchain technology and decentralized applications, especially in the crowdfunding sector.
- **Competition:** Competing with established platforms like Kickstarter and Indiegogo, but offering unique advantages like transparency and security through blockchain.
- **Target Audience:** Blockchain enthusiasts, and traditional crowdfunding users seeking more transparency.

❖ Risk Analysis

- **Technical Issues:** Integration with Ethereum and smart contract vulnerabilities could cause problems.
 - ✓ **Mitigation:** Perform rigorous testing and security audits, use established security practices.
- **Security Concerns:** Risks of hacking and data breaches.
 - ✓ **Mitigation:** Implement strong security measures and regular audits.

- **User Adoption:** Attracting users to a new blockchain-based platform may be challenging.
 - ✓ **Mitigation:** Invest in marketing and offer incentives for early adopters.
- **Market Competition:** Competing with established platforms like Kickstarter and Indiegogo.
 - ✓ **Mitigation:** Highlight unique blockchain benefits and differentiate the platform.
- **Revenue Uncertainty:** Risk of not achieving profitability.
 - ✓ **Mitigation:** Develop a clear revenue model and diversify income sources.

3.3 Project Timeline chart:



Figure 3.1: Timeline chart

3.4 Detailed Modules Description:

1. User Management Module

1.1 User Registration and Authentication:

❖ Working Methodology:

- Users sign up by filling out a registration form with personal details such as name, email, birthdate, and address.
- The system sends a verification email to the user's provided email address.

- Upon clicking the verification link, the user's email is confirmed, and they can log in using their credentials.
- The login process involves secure password handling, with passwords hashed and stored securely.

❖ **Expected Output:**

- Successfully registered and authenticated users with verified email addresses.
- Secure storage of user credentials and personal details.

1.2 Profile Management

❖ **Working Methodology:**

- Users can access their profile section to view and update personal information.
- Users can upload and manage verification documents for identity proof.
- Changes made to the profile are subject to re-verification if necessary.

❖ **Expected Output:**

- Updated user profiles with current information and verification status.
- Secure management of user documents and personal details.

1.3 User Verification

❖ **Working Methodology:**

- Admins access a dashboard listing all users pending verification.
- Admins review the submitted documents and personal details to ensure authenticity.
- Approved users receive notifications, and their profiles are marked as verified. Rejected users receive feedback on required changes.

❖ **Expected Output:**

- Verified user profiles ready for campaign creation.
- Notifications sent to users about their verification status.

2. Campaign Management Module

2.1 Campaign Creation

❖ **Working Methodology:**

- Verified users access the campaign creation form to provide campaign details, including title, description, rewards, funding goal, deadline, images, and supporting documents.
 - The system captures and stores all provided information securely.
- ❖ **Expected Output:**
- Detailed campaign entries stored in the database.
 - Campaigns ready for admin review and approval.

2.2 Campaign Submission

- ❖ **Working Methodology:**
- Upon submission, the campaign status is set to “pending.”
 - The system notifies admins of new pending campaigns awaiting review.
- ❖ **Expected Output:**
- Campaigns in pending status awaiting admin approval.
 - Notifications sent to admins for review tasks.

2.3 Campaign Verification and Approval

- ❖ **Working Methodology:**
- Admins review campaign details and supporting documents.
 - Campaigns are marked as either “approved” or “rejected” based on compliance with platform guidelines.
 - Approved campaigns are classified as either full fund release or partial fund release.
 - Creators are notified of the approval status and any required changes if rejected.
- ❖ **Expected Output:**
- Approved campaigns ready for public viewing and funding.
 - Notifications sent to campaign creators regarding their campaign status.

3. Campaign Discovery Module

3.1 Campaign Browsing

- ❖ **Working Methodology:**

- Users can browse and search for campaigns using various filters such as category, status, and popularity.
- The system retrieves and displays relevant campaigns based on user-selected filters.

❖ **Expected Output:**

- A user-friendly interface for discovering campaigns.
- Filtered and sorted campaign listings tailored to user preferences.

3.2 Campaign Details

❖ **Working Methodology:**

- Detailed information about each campaign is displayed, including title, description, rewards, current status, funding progress, and deadline.
- Users can view images and supporting documents submitted by campaign creators.

❖ **Expected Output:**

- Comprehensive campaign details available for potential backers.
- Enhanced transparency and trust through detailed information display.

4. Funding Management Module

4.1 Fund Contribution

❖ **Working Methodology:**

- Users select an approved campaign to contribute to.
- Contributions are made in Ethereum, with an option to convert to stablecoins (USDT/USDC) to avoid price fluctuations.
- The system securely handles the transaction and updates the campaign's funding progress.

❖ **Expected Output:**

- Successful contribution transactions recorded in the system.
- Updated funding progress for each campaign.

4.2 Smart Contract Integration

❖ **Working Methodology:**

- Smart contracts are created to securely hold and manage contributed funds.

- The system interacts with smart contracts to store contributions and handle fund releases.

❖ **Expected Output:**

- Secure storage of contributions in smart contracts.
- Transparent and tamper-proof handling of funds.

4.3 Fund Release

❖ **Working Methodology:**

- For full fund release campaigns, the smart contract releases funds to the creator after the campaign deadline.
- For partial fund release campaigns, creators submit fund release requests with progress updates and supporting documents.
- Admins review fund release requests and approve or reject them based on compliance with platform rules.
- Approved requests trigger the smart contract to release the specified amount of funds to the creator.

❖ **Expected Output:**

- Timely and secure release of funds to campaign creators.
- Reduced risk of misuse through staged fund releases and admin oversight.

5. Admin Dashboard Module

5.1 User Verification Management

❖ **Working Methodology:**

- Admins access a dashboard to review and manage user verification requests.
- Admins approve or reject requests based on the authenticity of submitted documents.

❖ **Expected Output:**

- Efficient and secure user verification process.
- Clear and actionable feedback provided to users.

5.2 Campaign Approval Management

❖ Working Methodology:

- Admins review pending campaigns, approve or request changes, and manage the overall campaign approval process.
- Admins mark campaigns as full or partial fund release based on verification documents.

❖ Expected Output:

- Streamlined campaign approval process.
- Clear guidelines and decisions on campaign statuses.

5.3 Fund Release Management

❖ Working Methodology:

- Admins review fund release requests from campaign creators.
- Admins approve or reject requests based on compliance with platform rules and submitted progress updates.

❖ Expected Output:

- Secure and controlled fund release process.
- Accountability and transparency in fund management.

5.4 Campaign and User Monitoring

❖ Working Methodology:

- Admins monitor active campaigns and user activities through the dashboard.
- Admins use monitoring tools to detect and respond to suspicious activities.

❖ Expected Output:

- Enhanced platform security and integrity.
- Prompt detection and response to potential fraud.

6. Payment and Transaction Module

6.1 Payment Processing

❖ Working Methodology:

- The system processes Ethereum payments and converts them to stable coins (USDT/USDC) using a stable coin exchange.

- Secure handling of transactions ensures the integrity and reliability of the payment process.

❖ **Expected Output:**

- Efficient and secure processing of contributions.
- Stable value of contributions through conversion to stable coins.

6.2 Transaction Tracking

❖ **Working Methodology:**

- The system records all transactions, including contributions, fund releases, and refunds.
- Detailed transaction logs are maintained for transparency and traceability.

❖ **Expected Output:**

- Comprehensive transaction records.
- Enhanced transparency and accountability.

6.3 Refund Management

❖ **Working Methodology:**

- The system handles refund requests in case of campaign cancellation or failure to meet verification standards.
- Funds are redistributed to backers based on their contribution amounts, minus any released amounts and fees.

❖ **Expected Output:**

- Fair and transparent refund process.
- Backers receive appropriate refunds in a timely manner.

3.5 Project SRS:

3.5.1 Use Case Diagram:

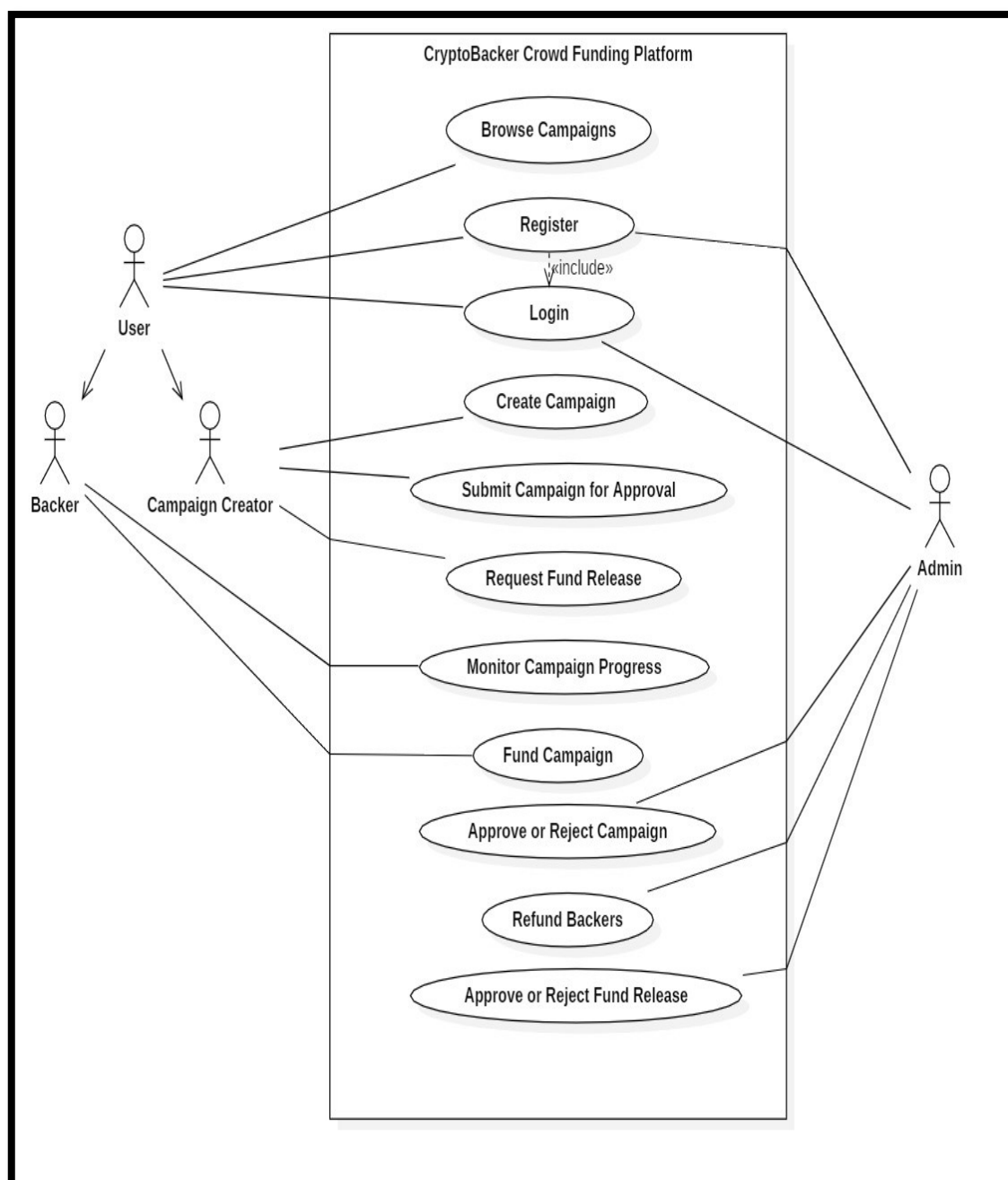


Figure 3.2: Use Case Diagram

3.5.2 Data Flow Diagram:

❖ Level 0:

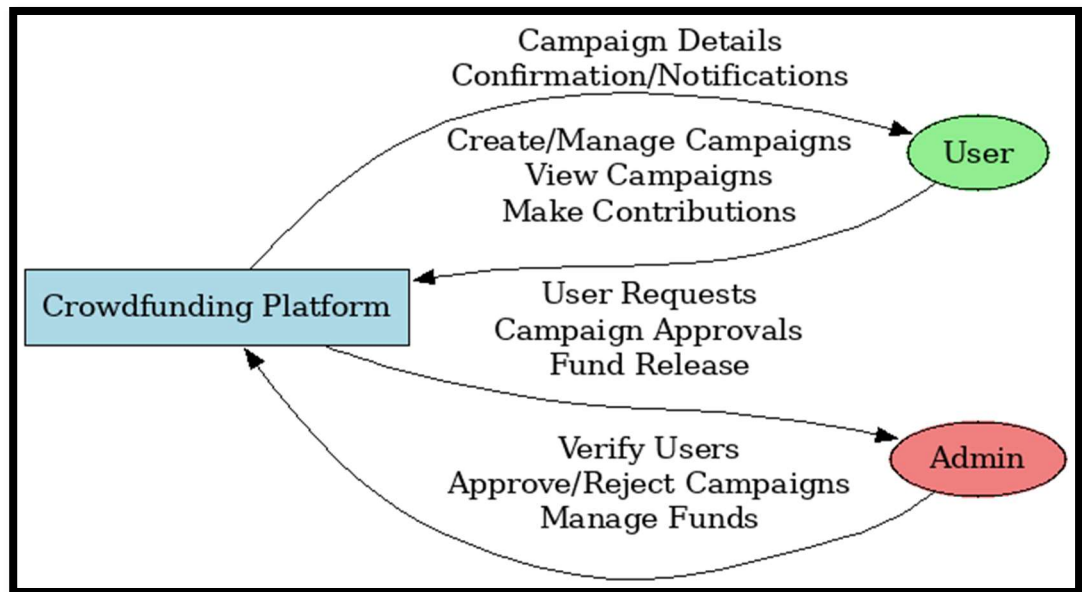


Figure 3.3: Level 0 DFD Diagram

❖ Level 1:

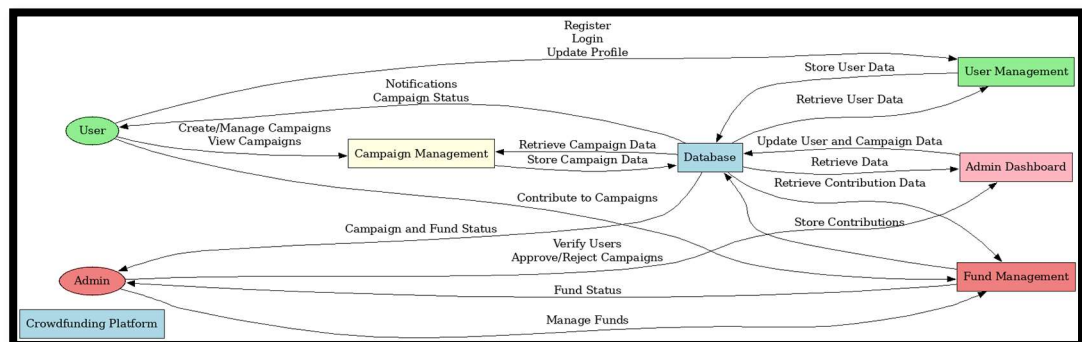


Figure 3.4: Level 1 DFD Diagram

❖ Level 2:

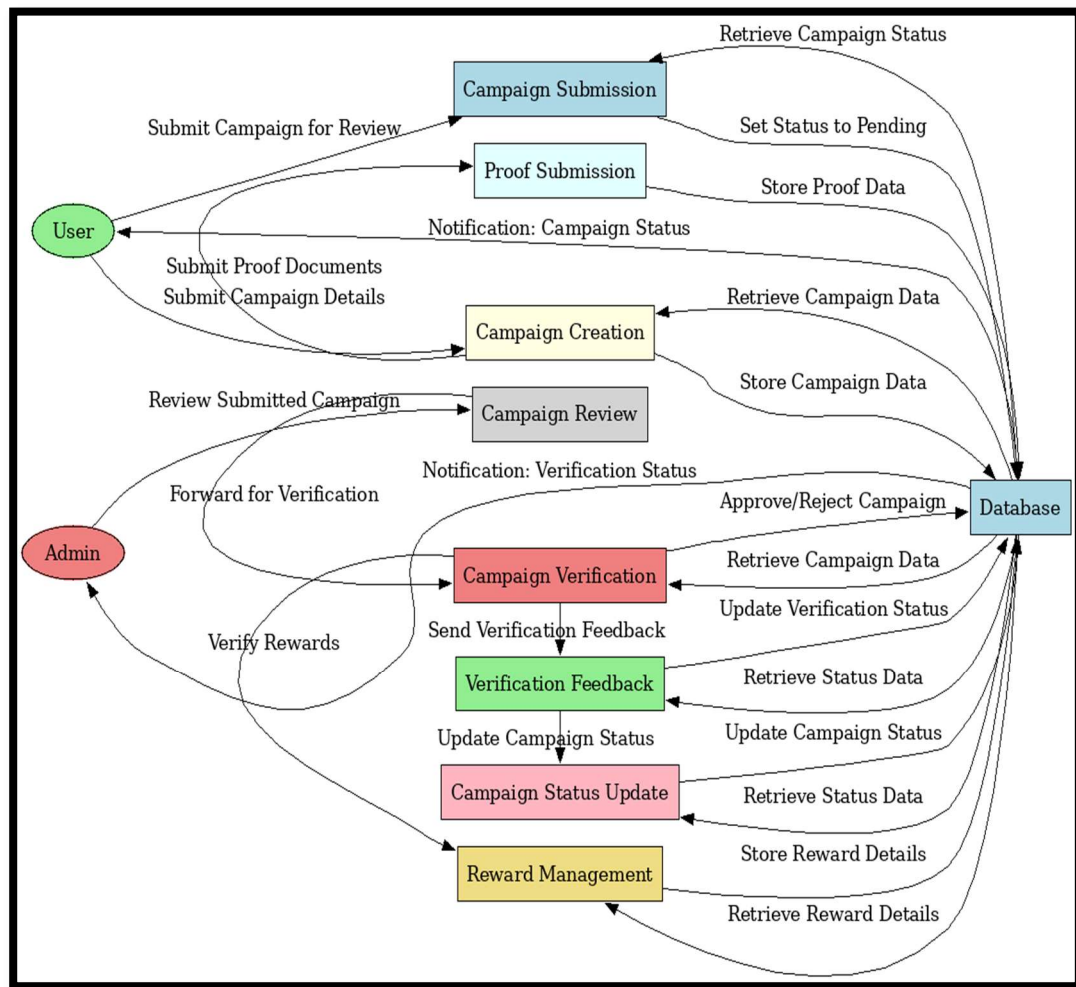


Figure 3.5: Level 2 DFD Diagram

3.5.3 Class Diagram:

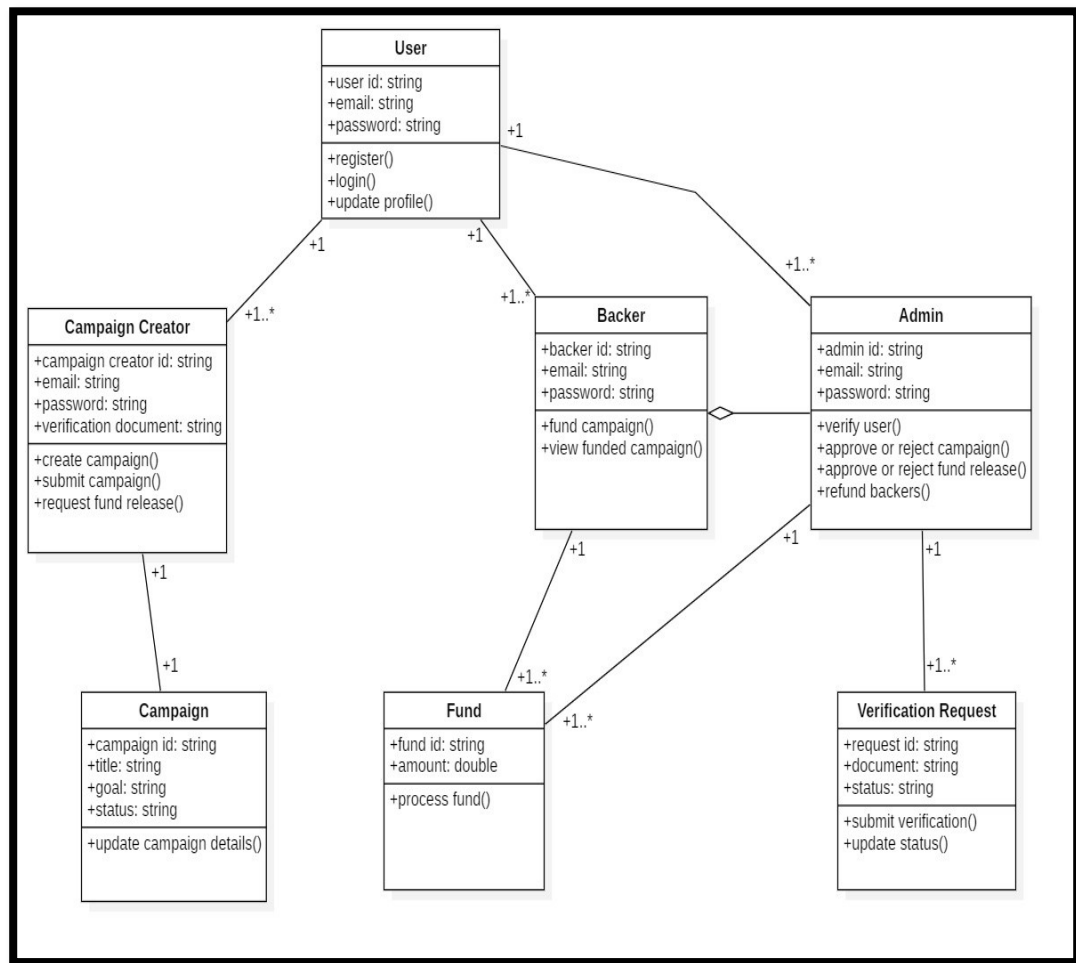


Figure 3.6: Class Diagram

3.5.4 Entity Relationship Diagram:

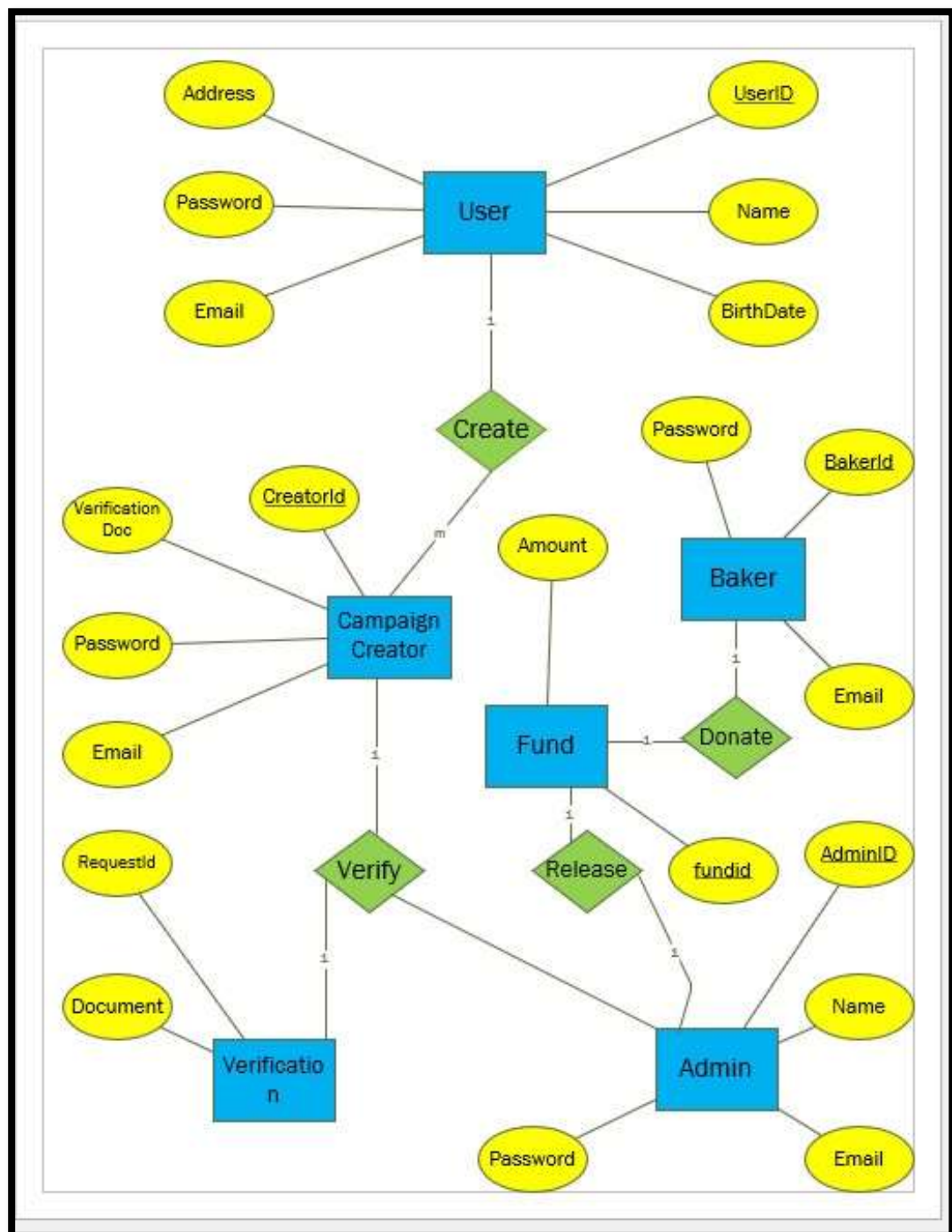


Figure 3.7: ER Diagram

3.5.5 Sequence Diagram:

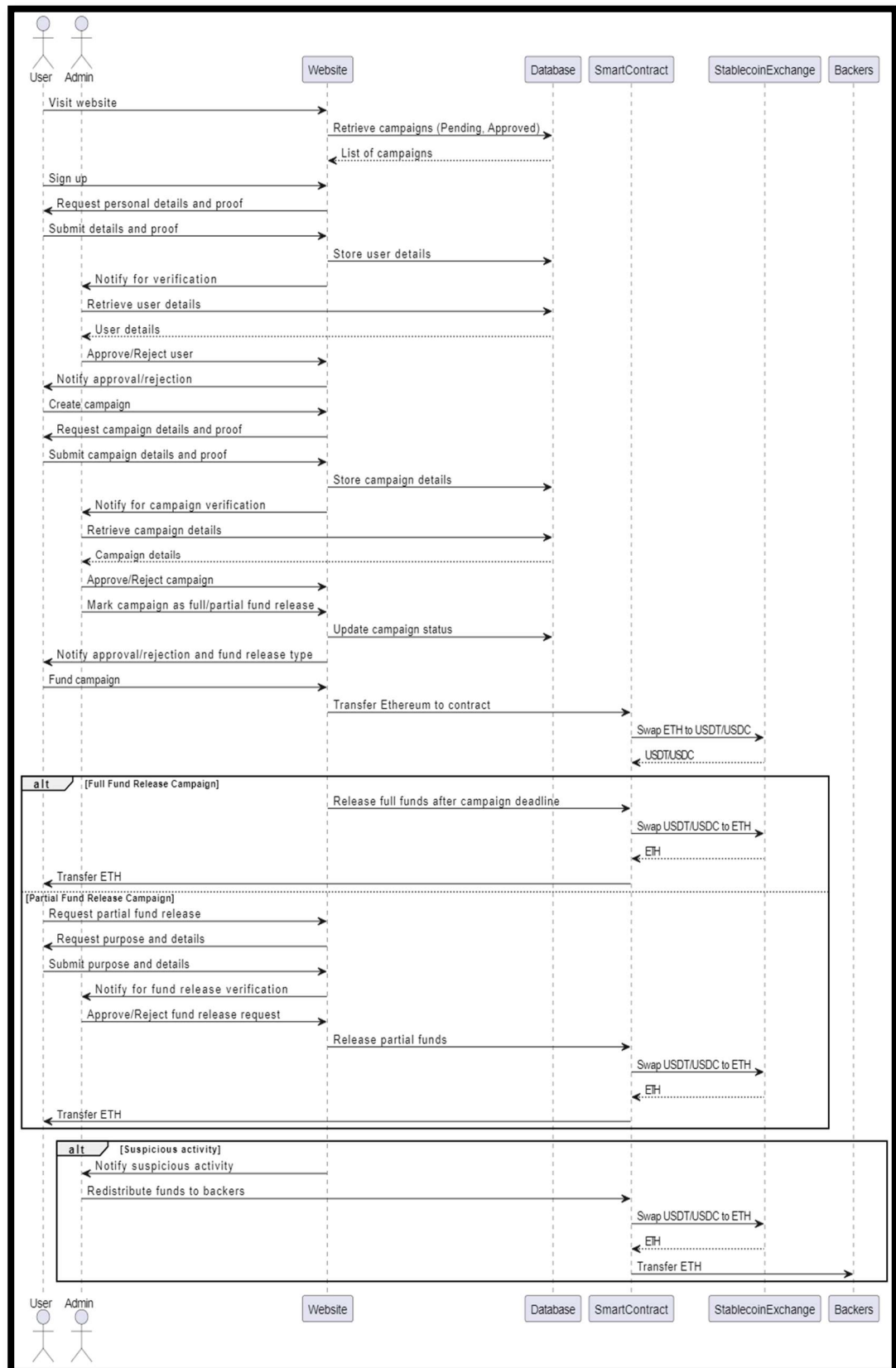


Figure 3.8: Sequence Diagram

3.5.6 Activity Diagram:

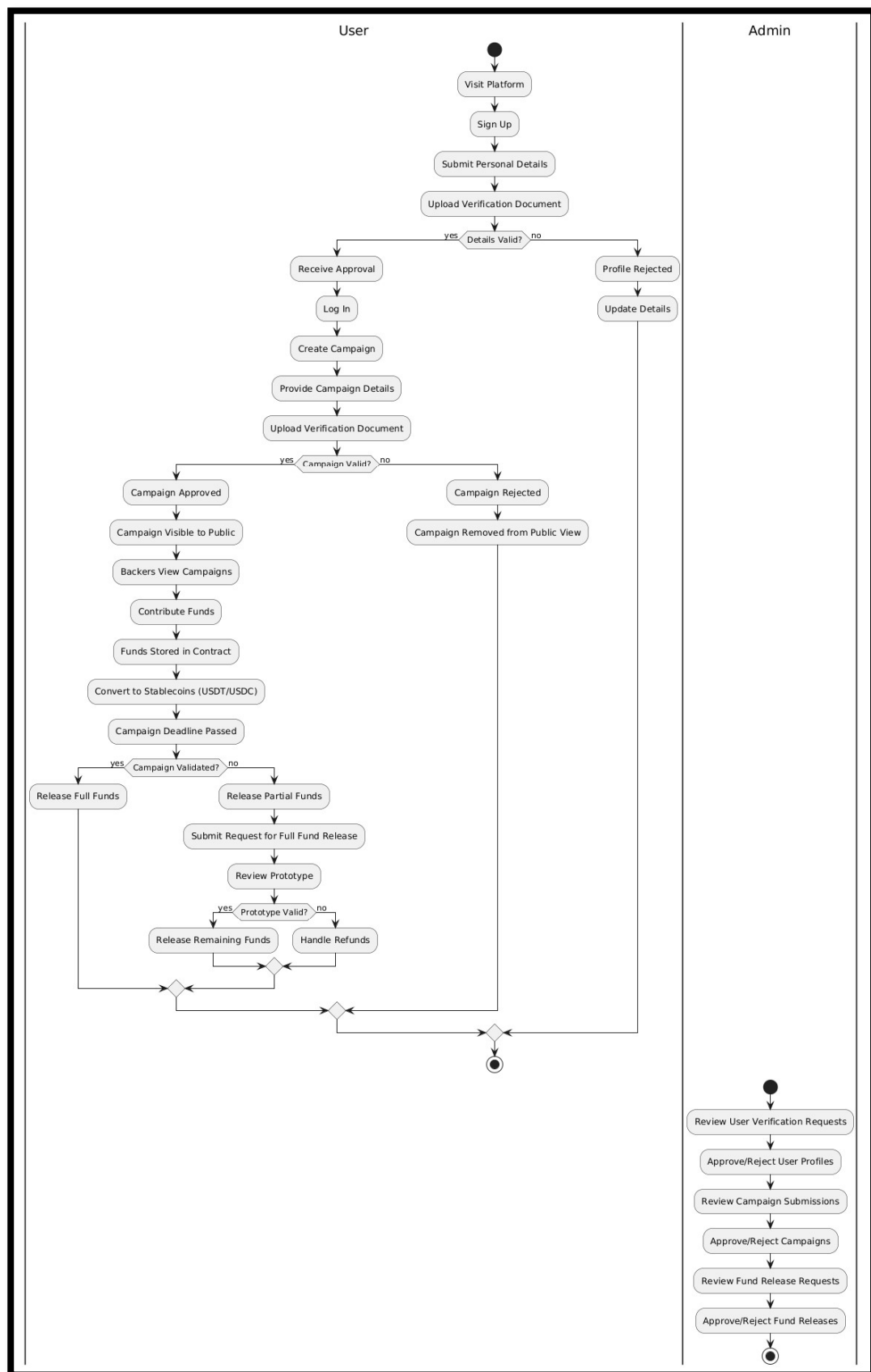


Figure 3.9: Activity Diagram

3.6 Data Dictionary:

❖ **User:**

Attribute	Description	Type	Format	Constraints
UserID	Unique identifier for the user	Integer	Numeric	Primary Key, Auto-incremented
Name	Full name of the user	String	Text (Up to 100 characters)	Required
Email	Email address of the user	String	Email format	Required, Unique
Birthdate	Date of birth of the user	Date	YYYY-MM-DD	Required
Address	Residential address of the user	String	Text (Up to 255 characters)	Required
ProofDocument	Document proving the identity of the user	String	File path or URL	Required
Status	Current verification status of the user	String	Enum (e.g., Pending, Approved, Rejected)	Required

Table 3.1: User Table

❖ Campaign:

Attribute	Description	Type	Format	Constraints
CampaignID	Unique identifier for the campaign	Integer	Numeric	Primary Key, Auto-incremented
Title	Title of the campaign	String	Text (Up to 150 characters)	Required
Description	Detailed description of the campaign	Text	Long text	Required
Rewards	Rewards offered for backing the campaign	Text	Long text	Optional
Goal	Funding goal of the campaign	Decimal	Numeric (with two decimal places)	Required
Deadline	Deadline date for the campaign	Date	YYYY-MM-DD	Required
Image	Image associated with the campaign	String	URL or file path	Optional
Status	Status Current status of the	String	Enum (e.g., Pending, Approved , Rejected)	Required
Verification Document	Document proving the authenticity of the campaign	String	File path or URL	Required

Table 3.2: Campaign Table

❖ **Admin:**

Attribute	Description	Type	Format	Constraints
AdminID	Unique identifier for the admin	Integer	Numeric	Primary Key, Auto-incremented
Name	Full name of the admin	String	Text (Up to 100 characters)	Required
Email	Email address of the admin	String	Email format	Required, Unique

Table 3.3: Admin Table

❖ **Fund Contribution:**

Attribute	Description	Type	Format	Constraints
ContributionID	Unique identifier for the fund contribution	Integer	Numeric	Primary Key, Auto-incremented
Amount	Amount contributed to the campaign	Decimal	Numeric (with two decimal places)	Required
Date	Date when the contribution was made	Date	YYYY-MM-DD	Required
ContributorID	Identifier of the user who made the contribution	Integer	Numeric	Foreign Key (references UserID)
CampaignID	Identifier of the campaign being funded	Integer	Numeric	Foreign Key (references CampaignID)

Table 3.4: Fund Contribution Table

❖ **Transaction:**

Attribute	Description	Type	Format	Constraints
TransactionID	Unique identifier for the transaction	Integer	Numeric	Primary Key, Auto-incremented
Amount	Amount involved in the transaction	Decimal	Numeric (with two decimal places)	Required
Type	Type of the transaction (e.g., Contribution, Refund)	String	Enum (e.g., Contribution, Refund)	Required
Date	Date of the transaction	Date	YYYY-MM-DD	Required
Status	Current status of the transaction	String	Enum (e.g., Completed, Pending, Failed)	Required

Table 3.5: Transaction Table