**Crowd Coin**



**S. R. S. Report – I**

**Institute of Engineering & Technology**

**Members**

Hritik Tiwari (161500245)

Jharna Agarwal (161500250)

Nainka Jain (161500339)

Nikhil Varshney (161500359)

Table of Contents

Revision History ii

Document Approval ii

1. Introduction 1

1.1 Purpose 1

1.2 Scope 1

1.3 Definitions, Acronyms, and Abbreviations 1

1.4 References 1

1.5 Overview 1

2. General Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Characteristics 2

2.4 General Constraints 2

2.5 Assumptions and Dependencies 2

3. Specific Requirements 2

3.1 External Interface Requirements 3

3.1.1 User Interfaces 3

3.1.2 Hardware Interfaces 3

3.1.3 Software Interfaces 3

3.1.4 Communications Interfaces 3

3.2 Functional Requirements 3

3.3 Use Cases 3

3.3.1 Use Cases 3

3.5 Non-Functional Requirements 4

3.5.1 Performance 4

3.5.2 Reliability 4

3.5.3 Availability 4

3.5.4 Security 4

3.5.5 Maintainability 4

3.5.6 Portability 4

3.6 Inverse Requirements 4

3.7 Design Constraints 4

3.8 Logical Database Requirements 4

3.9 Other Requirements 4

4. Analysis Models 4

4.1 Sequence Diagrams 5

4.2Data Flow Diagrams (DFD) 5

**1. Introduction**

## 1.1 Purpose

Crowdcoin is an ethereum based, real time funding decentralized application for creative projects.. Every project creator sets their project's funding goal and deadline. If people like the project, they can pledge money to make it happen. If the project succeeds in reaching its funding goal, all backers' credit cards are charged when time expires .It helps artists, musicians ,filmmakers , designers ,and others creators they need to make their ideas as a reality. To date ,tens of thousands of creative projects-big and small-have come to life .Our project focus on to help bring creative projects to life.

**1.2 Scope**

* There can be multiple creators who want to get fund for their project.
* Each creator sets their funding goal and deadline.
* If people like the project, they can pledge money to make it happen.
* If creator get funded, all backers’ credit cards get charged.
* Creator must have project idea to get funded.
* Money or fund will be taken back if they are unable to submit their project.

**1.3 Definitions, acronyms and abbreviations**

**Crowdcoin-** Crowdcoin is an application that allows the creator to complete their projects by getting funded from an external sources and after completion of their project they have to submit their project to the source from whom they get funded.

**Creator-** It is the person or team behind the project idea, working to bring it to life.

**Project-** It is the finite work with a clear goal that you’d like to bring to life.

**UML-**Unified Modeling Language is a standard language for writing software blueprints. The UML may be used to visualize, specify, construct and document.

**HTTP-** It stands for Hypertext Transfer Protocol and it is a service protocol.

**HTML-** It stands for Hyper Text Markup Language used to create the GUI.

**Solidity-** It is a contract-oriented, high-level language for implementing smart contracts.It was influenced by C++, python and JavaScript and is designed to target the Ethereum Virtual Machine (EVM).

**Truffle-**It is a Node based development framework which is currently the most used and actively maintained in the space.

**Truffle HDWallet provider** : It is a convenient and easy to configure network connection to ethereum through infura.io.

**Ganache-**Ganache is a personal for ethereum development you can use to deploy contracts, develop your applications and run test .It is available as both a desktop application as well as a command-line tool .It is available for Windows, Mac, and linux. **Ganache CLI:** Ganache CLI is the latest version of TestRPC; a fast and customizable blockcahin emulator. It allows you to make calls to the blockchain without the overheads of running an actual Ethereum node.

**ReactJS-**It is an open-source JavaScript library which is used for building user interfaces especially for single page applications .It is used for handling view layer for web and mobile apps. React also allows us to create reusable UI components. It allows developers to create large web applications which can change data without reloading page. The main purpose of is to be fast, scalable and simple.

**1.4 Tools Used**

**Visual Code Studio-**Visual Code Studio is a source code editor developed by Microsoft for Windows, Linux and macOS. It includes support for debugging  embedded Git control, [syntax highlighting](https://en.wikipedia.org/wiki/Syntax_highlighting), [intelligent code completion](https://en.wikipedia.org/wiki/Intelligent_code_completion), [snippets](https://en.wikipedia.org/wiki/Snippet_(programming)), and [code refactoring](https://en.wikipedia.org/wiki/Code_refactoring). It is also customizable, so users can change the editor's [theme](https://en.wikipedia.org/wiki/Theme_(computing)), [keyboard shortcuts](https://en.wikipedia.org/wiki/Keyboard_shortcut), and preferences. It is [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source), although the official download is under a [proprietary license](https://en.wikipedia.org/wiki/Proprietary_software). Visual Studio Code is based on Electron, a framework which is used to deploy Node.js applications for the desktop running on the Blink layout engine.

**Web3js-**It is a collection of libraries which allow you to interact with a local or remote ethereum node, using a HTTP or IPC connection. This is the Ethereum compatible javascript API which implements the generic JSON RPC spec .Its available on npm as a node module , for bower and component as an embedded.js and as a meteor.js package.

**MetaMask-**Metamask is a web browser plug-in that works with Google Chrome, Opera, and firefox.It can be installed the same way as any other Google Chrome extension or Firefox add-on. MetaMask is capable of interacting with any and all ERC-20 assets as it allows for custom tokens to be added. It can also interact with the various Ethereum test networks .So it is useful tool for developers ,as well as those that want to preview various Dapps.It allows its user to use web browser based apps..

**1.6 References**

* International Journal of Scientific and Research Publications, Volume 5, Issue 1, January 2015 1 ISSN 2250-3153.
* Software Engineering, seventh edition, Roger S. Pressman.
* Software Engineering, Seventh Edition, Ian Somerville.
* Hans van Vliet. Software Engineering: Principles and Practice (Second Edition). Wiley, 1999.
* Encyclopedia of Software Engineering” by LaPlante.
* Wikipedia -www.wikipedia.com
* Database Management Systems – Navathe.
* Object Oriented Modeling and Design with UML-Michael Blaha, James Ram Baugh.
* C. J. Date, A. Kannan and S. Swamynathan, *An Introduction to Database Systems*, Pearson Education, Eighth Edition, 2009.
* Shio Kumar Singh, *Database Systems Concepts, Designs and Application*, Pearson Education, Second Edition, 2011.

## 1.6 Overview

The rest of the SRS examines the specifications of the Crowdcoin in detail. Section 2 of the SRS presents the general factors that affect the Crowdcoin and its requirements such as user characteristics assumptions, dependencies and project constraints.

Section 3 outlines the detailed, specific functional, performance, system, non-functional, inverse and other related requirements along with design constraint of the Crowdcoin

The final section i.e. Section 4 contains Data Flow Diagrams (D.F. D), Sequence Diagrams, Entity Relationship Diagrams (E.R.). This section is highly useful in analysis phase of model.

* There can be multiple creators who want to get fund for their project..
* Each creator sets their funding goal and deadline.
* If people like the project, they can pledge money to make it happen.
* If creator gets funded, all backers’ account get charged.
* Creator must have project idea to get funded.
* Money or fund will be taken back if they are unable to submit their project.

**2. General Description**

This document contains the problem statement that the current system is facing which is

Frauds in the money given for startups by different.

Crowd coin is a funding platform for creative projects. Everything from film, games, and music to art, design, and technology. It is full of ambitious, innovative, and imaginative projects that are brought to life through the direct support of others.

Every project creator sets their project's funding goal and deadline. If people like the project, they can pledge money to make it happen. If the project succeeds in reaching its funding goal, all backers' credit cards are charged when time expires. Funding on all-or-nothing. If the project falls short of its funding goal, no one is charged.

Backers pledge money for different reasons. Some backers are rallying around their friends' projects. Some are supporting people they've long admired. Many are just inspired by a new idea. Others are inspired by a project's rewards — a copy of what's being made, a limited edition, or a custom experience related to the project.

But there are certain malicious campaign creators who are just for getting the money and at last say that we are out of money and we cannot return any product. In this case all the backers looses all of their money and get nothing in return of the money. so, this is the fraud happening in Kickstarter.

This decentralised application helps in managing the fraud by giving the backers or investors the authority to approve the request of money for the certain purpose and if all of the backers or investors do not agree to the request then there will be no transaction and it will also take care of that the money will not go in any of the managers or campaign creator's personal account.

Not all of the campaign creators are malicious or frauds but for the investors it is a very good applications as it saves the money from being fraud and they can also see where their money is being invested.

This section consists of five subsections, as follows:

1) Product perspective;

2) Product functions;

3) User characteristics;

4) General Constraints;

5) Assumptions and dependencies.

**2.1 Product Perspective**

This product is for the people who have the idea for any product and they wanted their idea to be funded for its development and progress. This real life solution is very helpful and has the following features-

**1. Easy to use-** It is very easy to use as it provides a very nice and interactive interface for the users.

**2.Secure-** This solution is very safe and secure as it uses block chain that provides 128 bit keys that can't be guessed by the humans and smart contracts for the overall mechanism of the funding of the campaign.

**3. Number of users supported-** The numbers of users supported by the solution is one i.e. the person who is creating the campaign or applying for funding of his/her idea. There can be as many numbers of persons who can contribute money for the idea.

**4. Tranceparency to the funders-**This solution provides transparency to the users as the contributor can see that how much money is used and where it is used and even approve or disapprove a request.

**2.2 Product Functions**

The main purpose of this solution is to provide funds to the new ideas and also keeps the fraud resistance capabilities in it.

The user that have the idea and they want funding for their idea can use this application and is called the manager. For the user to be manager he or she must have sufficient amount of gas.

The users who contribute for the idea or give money are called contributors. The contributors must also pay sufficient amount of ethers to be contributors. If the ethers paid are less than a specified limit then it will show error. Then the manager can make the request to the funders in whom the manager will fill the description of the request, amount required for the request and the account number in which the amount has to be transferred. All the contributors can see the requests and can approve it. A request can only be considered as approved only if more than 50 percent contributors mark the request as accepted else it will be denied. Till

time request is not accepted all the money will go in the contract. When the request is approved then only the money will be transferred to the respective account.

**2.3 User Characteristics**

There are two types of user i.e. the campaign creator or idea person and the another one is the amount payer or the contributors.

The manager is the person who has sufficient amount of gas and it can generate the request for funding of the idea by filling in the description, amount requested and the account number in which the amount is to be transferred i.e. the account number of the vendor.

The second user is the contributor who can give funds for the idea in exchange of the products. For the user to be contributor it must have certain amount of the ethers and also more than the specified limit for the user to be the contributor. The request gets approved only after the approval from 50% of the contributors.

**2.4 General Constraints**

**1**. The current constraint in this application is that the account that is used for the transfer of money is not verified i.e. the account can be of any other person.

2. The following work is going on the rinkeby test network.

3. The user must have Metamask (a chrome extension) must be added to make any sort of transaction.

### 2.4.1 Environmental and Technology Constraints

### 2.4.1.1 Software Constraints

The team has recommended that the System be constructed using Blockchain Technology for reasons of security and decentralization to choose a variety of vendor products. It uses Solidity, Metamask.

### 2.5 Assumptions and Dependencies

A number of factors that may affect the requirements specified in the SRS include:

1. All the transaction and transfer of amount is in ethers.
2. Assuming that the user already has the basic knowledge of English language.
3. It is assumed that we are working on rinkeby test Network.

# 3. Specific Requirements

# 3.1 External Interface Requirements

### 3.1.1 User Interfaces

1. **Home screen**

* Dashboard

**3.1.2 Hardware Interfaces**

1) Screen resolution of at least 800X600 is required for proper and complete viewing of screens. Higher resolution will be accepted.

2) Minimum 4 GB RAM.

3) Any type of processor.

### 3.1.3 Software Interfaces

1. Any windows based operating system.
2. Visual code Studio
3. ReactJS
4. HTML
5. Ganache cli
6. Node.js
7. MetaMask

### 3.3.4 Communications Interfaces

None

## 3.2 Functional Requirements

### 3.2.1 Provides fund to projects

Users are required to perform some functions.

**3.2.1.1 Introduction**

There are creator who demands for funds for their projects. If people like the project, they can pledge money to make it happen. If the project succeeds in reaching its funding goal, all backers'account are charged when time expires.

**3.2.1.2 Inputs**

The input in this is the creator need to login into their metamask account. After login creator has to enter ethers they want as their fund of project.

**3.2.1.3 Processing**

If the demanding ethers is present in the metamask then it gets transfer.

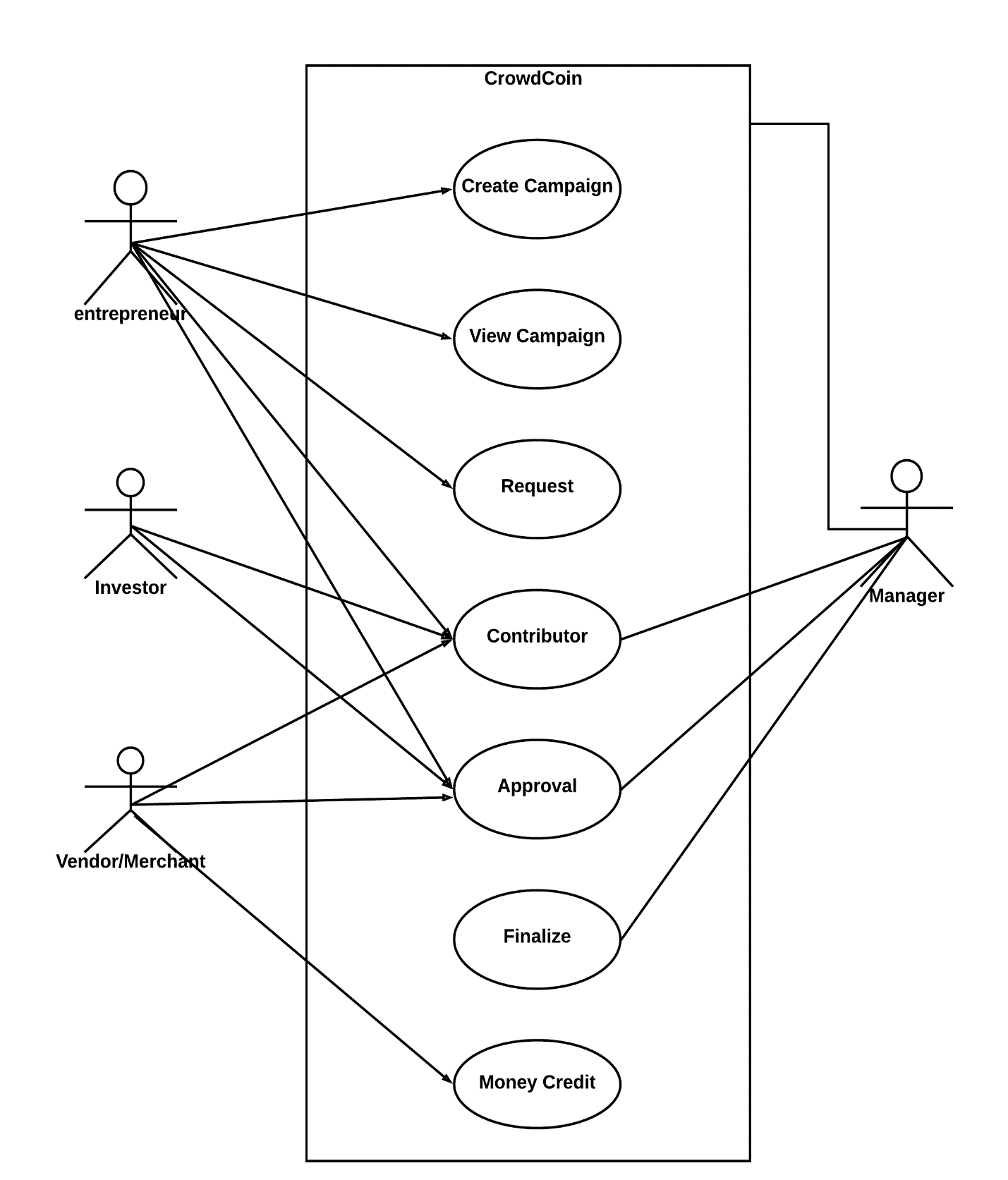
**3.2.1.4 Outputs**

It shows contract is successfully deployed.

**3.2.1.5 Error Handling**

Creator must have to enter more than .001 for successfully deployment of contracts and getting funds.

## 3.3 USE -CASE DIAGRAM



**Fig 3.2 Use- Cases Diagram**

## 3.4 Non-Functional Requirements

Non-functional requirements cover all the remaining requirements which are not covered by the functional requirements.

### Performance: For example, Response Time, Throughput, Utilization, and Static Volumetric. There is need for the internet connection.

* **Scalability:** There is no limit of data i.e., no constraint related to size of the data.

### Capacity: It can provides facilities to avoid fraud among the funds transfer.

* **Availability:** It is online software; software needs internet connection.
* **Maintainability:** There is maintenance needed.
* **Security:** There is constraint in Metamask account, i.e., only those can login who have account on MetaMask.
* **Environmental**: Ethereum Framework is used.
* **Portability:** The application can be run on any operating system and on any browser provided that there is an internet connection.

## 3.5 Design Constraints

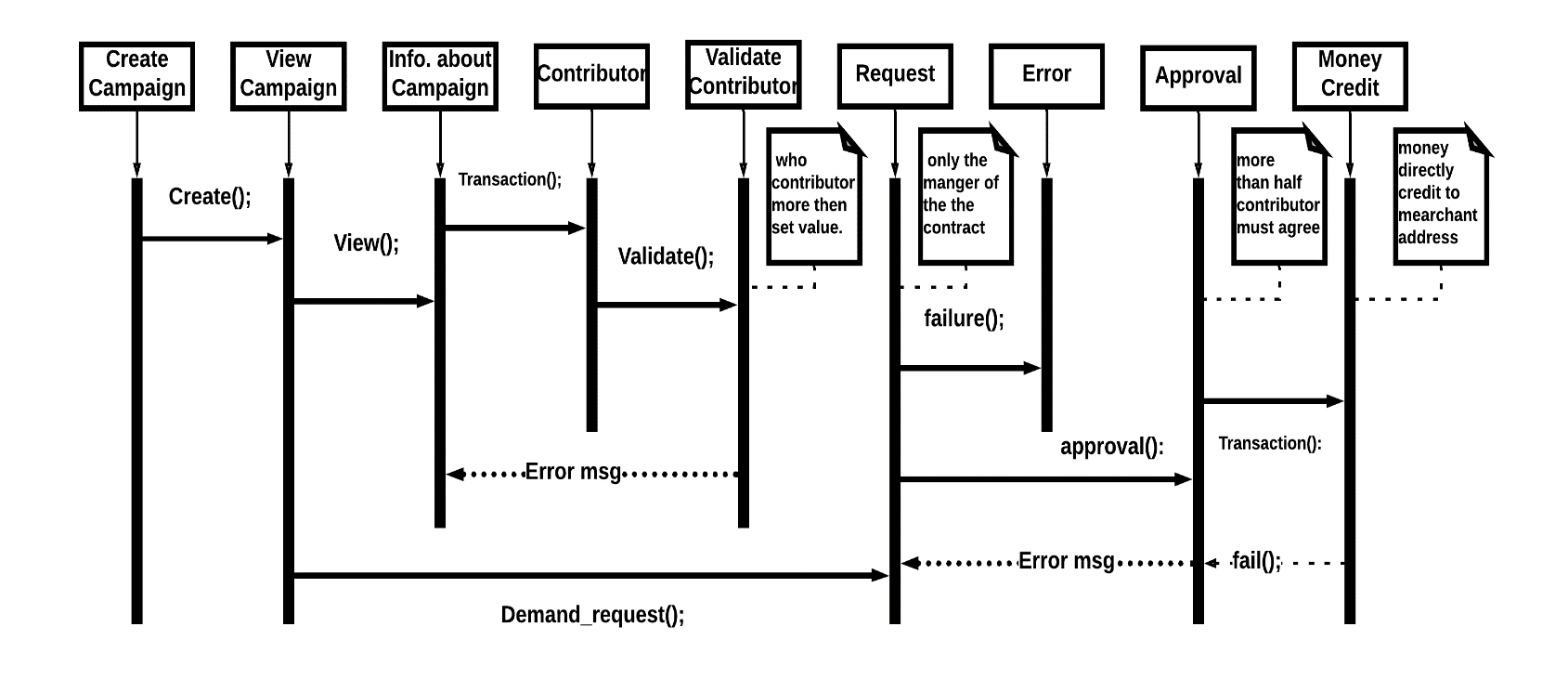
* No design Constraints

## 3.6 Logical Database Requirements

* No requirements of Database

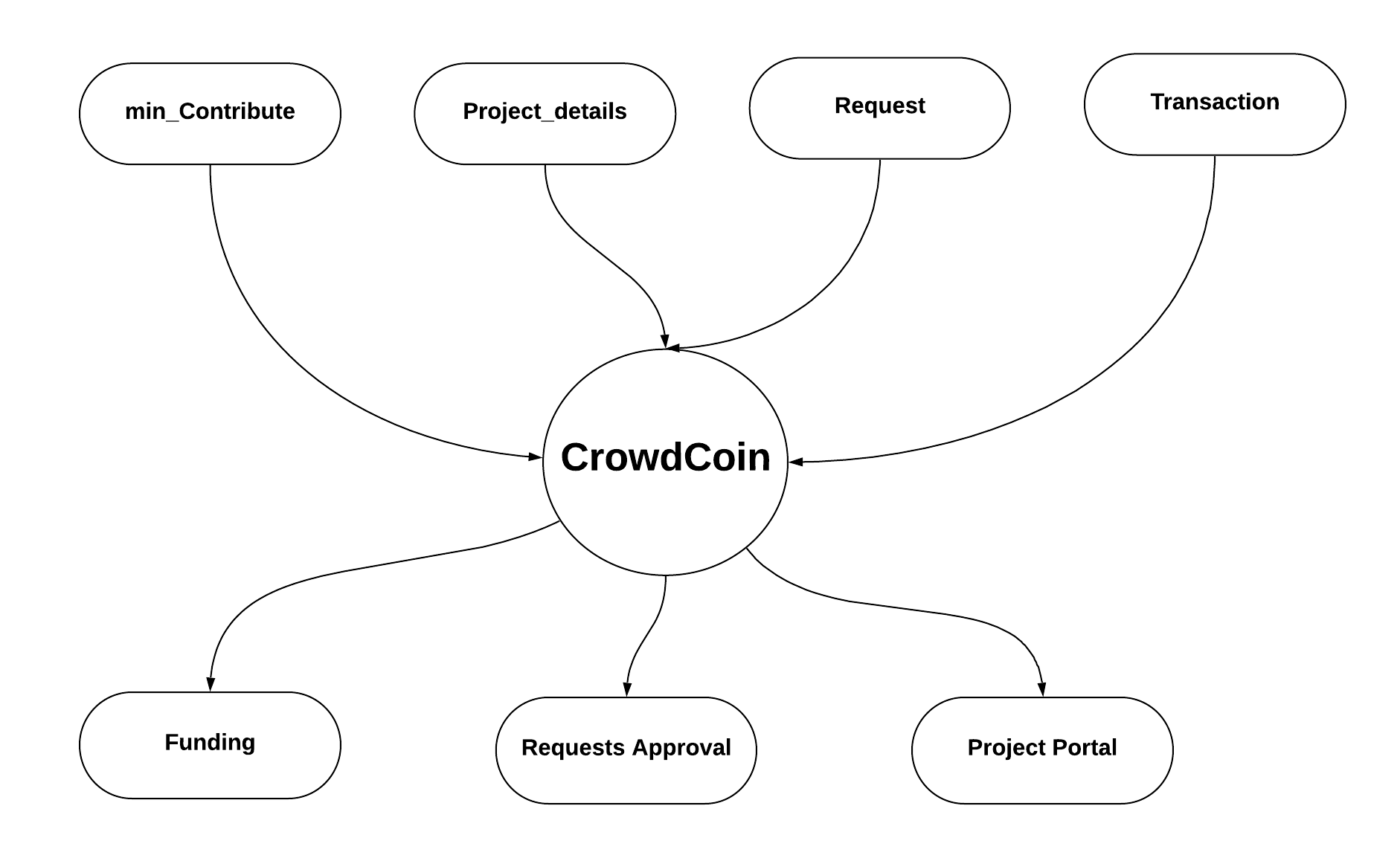
# 4. Analysis Models

## 4.1 Sequence Diagrams

****

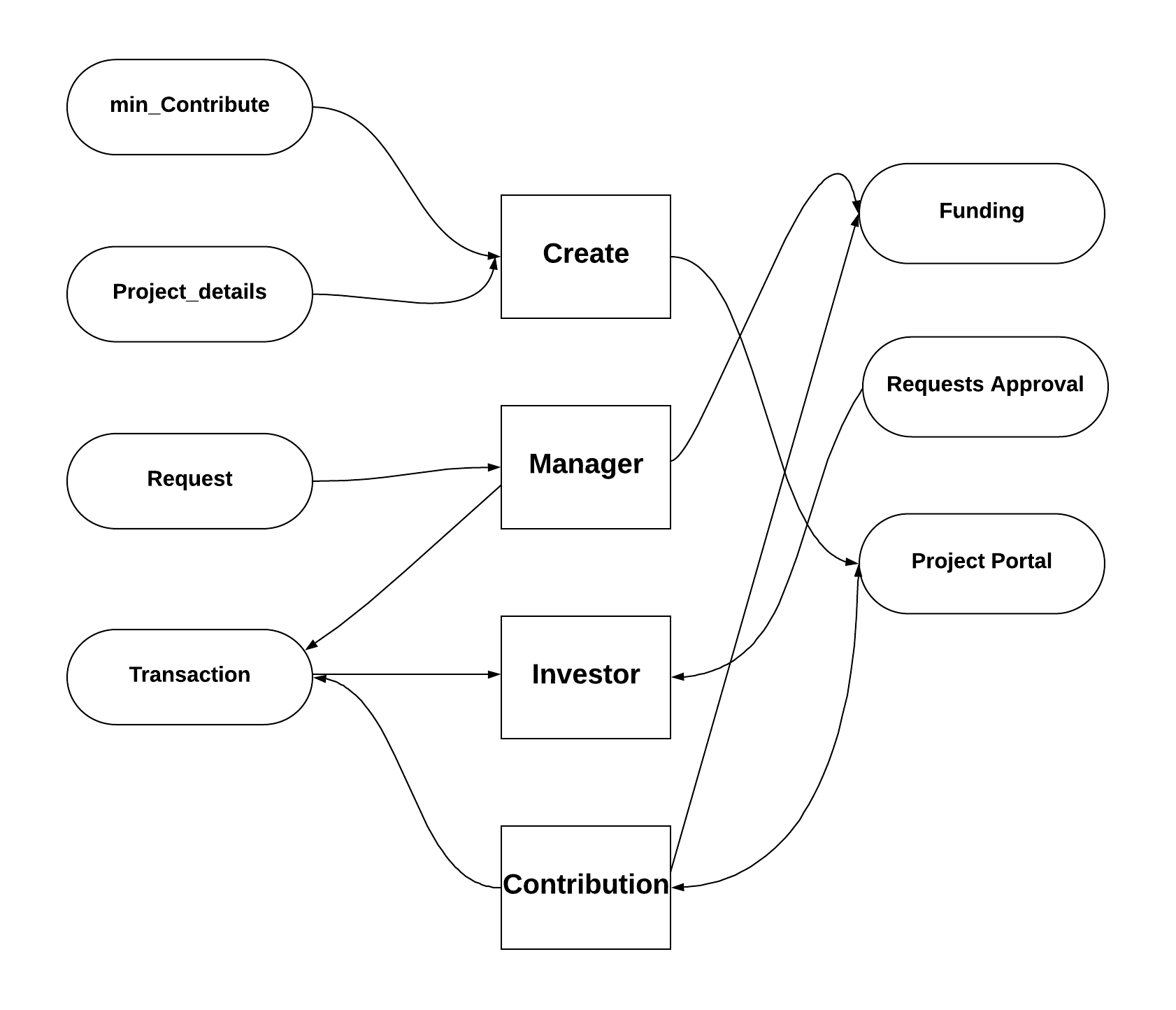
**Fig. 4.1 Sequence Diagram**

## 4.2 Data Flow Diagrams (DFD)

**0-LEVEL DFD:**

**Fig. 4.2.1 0-Level DFD**

**1-LEVEL DFD:**



**Fig. 4.2.2 Level 1 DFD**