DECENTRALIZED WEB-HOSTING SYSTEM

A Project Report

Submitted by

Aditya Ingle	**112103050**
Vedang Khedekar	**112103070**
Anagha Mahure	**112103082**

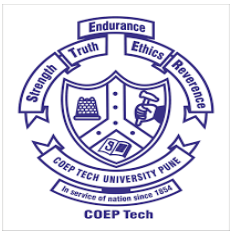
of

TY (Computer Engineering)

Under the guidance of

Dr. Tanuja R. Pattanshetti

COEP Technological University



DEPARTMENT OF COMPUTER ENGINEERING

COEP Technological University

April, 2024

DEPARTMENT OF COMPUTER ENGINEERING

COEP Technological University

CERTIFICATE

Certified that this project, titled "RETAIL E-COMMERCE PRICE TRACKER" has been successfully completed by

Aditya Ingle	**112103050**
Vedang Khedekar	**112103070**
Anagha Mahure	**112103082**

and is approved for the fulfilment of the requirements of "Software Engineering Mini Project- Stage II".

SIGNATURE

Dr. Tanuja R. Pattanshetti

Project Guide

Department of Computer Engineering

COEP Technological University,

Shivajinagar, Pune - 5.

Abstract

Hosting websites is traditionally done in a centralized network where a web server distributes the website's files to it's clients. This enables larger companies or governments to choose what content is shown and remove the freedom that the world wide web initially brought. However, using a decentralized network to host websites might be favorable in some situations due to the decentralized networks' high scalability and protection against censorship. This study aims to determine how a website can be published in a decentralized network and if it can be preferred for some types of websites. The study also aims to find out if there exist any security vulnerabilities in the decentralized networks. This was done by a website hosted on a traditional web server and the decentralized solution IPFS. The proposed system utilizes blockchain as the underlying framework for hosting and accessing web content. Smart contracts facilitate the allocation of resources and enforce rules governing content distribution and access. Through a peer-to-peer network, users can host and access websites without relying on traditional hosting providers, thereby mitigating the risk of single points of failure and censorship.

Contents

List of Figures

Chapter 1

Synopsis

1.1 Project Title

DECENTRALIZED WEB-HOSTING SYSTEM

1.2 Internal Guide

Dr. Tanuja R. Pattanshetti

1.3 Problem Statement

In an increasingly centralized digital landscape, the reliance on traditional web hosting solutions has resulted in vulnerabilities such as single points of failure, data breaches, and censorship. To address these challenges and foster a more resilient and secure internet infrastructure, there is a pressing need to develop a decentralized web hosting ecosystem leveraging the power of blockchain technology.

1.4 Plan of Project Execution

Task	Start Date	End Date	Duration	Resources
Problem statement finalization	08-Jan-24	15-Jan-24	8	Vedang,Aditya, Anagha
Project Plan	16-Jan-24	31-Jan-24	16	Vedang
Research	01-Feb-24	05-Feb-24	5	Vedang, Anagha
Requirement Analysis	06-Feb-24	09-Feb-24	4	Aditya,Vedang
Architectural Design	10-Feb-24	13-Feb-24	4	Aditya
User-Interface Design	13-Feb-24	18-Feb-24	6	Anagha
Security Design	19-Feb-24	29-Feb-24	10	Vedang
Prototyping	01-Mar-24	10-Mar-24	10	Vedang,Aditya
Backend Development	11-Mar-24	20-Mar-24	10	Vedang,Aditya
Frontend	21-Mar-24	31-Mar-24	10	Anagha,Aditya
Testing	01-Apr-24	10-Apr-24	10	Vedang,Anagha,Aditya
Documentation and project report	11-Apr-24	15-Apr-24	5	Vedang,Anagha,Aditya

Chapter 2

Problem Definition and scope

2.1 Problem Definition

2.1.1 Goals and objectives

Goal and Objectives:

- Goals
  - Develop a decentralized architecture for web hosting to enhance reliability, security, and censorship resistance.
- Objectives
  - Enable users to host websites without relying on traditional, centralized hosting providers.
  - Foster a community-driven approach to web hosting, promoting decentralization and democratization of the internet.
  - Enable users to modify their files in an online IDE.

2.1.2 Statement of scope

- **Target Products:** Blockchain-based hosting services, Decentralized web hosting platforms
- **Target Websites:** E-commerce platforms, Social networking sites, Portfolio websites
- **Target Audience:** This application is ideal for small to medium-sized businesses, content creators, individuals, developers building dApps.

2.2 Software context

The Decentralized Web Hosting using Blockchain project operates within the context of blockchain technology and decentralized systems. It involves the integration of blockchain protocols for distributed storage and consensus mechanisms. The project utilizes programming languages such as Solidity for smart contract development and JavaScript for frontend interfaces. Smart contracts are self-executing contracts with the terms of the agreement directly written into code. Solidity is a programming language specifically designed for writing smart contracts on blockchain platforms like Ethereum. Smart contracts enable automation of tasks and the execution of predefined actions without the need for intermediaries. Additionally, it leverages blockchain platforms like IPFS for decentralized storage and hosting. The system employs a decentralized storage protocol, such as IPFS (InterPlanetary File System), to store and retrieve web content.

2.3 Major Constraints

- – **Blockchain scalability:** Addressing scalability issues inherent in blockchain networks to ensure efficient hosting of websites and web applications.
  - – **User adoption:** Overcoming barriers to user adoption and ensuring that the platform is accessible and intuitive for both developers and end-users.
  - – **Regulatory compliance:** Navigating legal and regulatory challenges associated with decentralized hosting, including data protection and compliance with relevant regulations.
  - – **Interoperability:** Ensuring interoperability with existing web infrastructure and standards to facilitate seamless integration with the broader internet ecosystem.

2.4 Outcome

The primary outcome of the Decentralized Web Hosting using Blockchain project is the development of a robust and user-friendly platform for hosting websites and web applications in a decentralized manner. Key components of the outcome include:

- - Decentralized storage infrastructure
- - Secure and censorship-resistant hosting environment
- - User-friendly deployment tools and interfaces
- - Scalable and resilient hosting network
- - Active community participation and governance mechanisms

2.5 Applications

- **Decentralized Website Hosting:** Enables users to host websites and web applications in a decentralized manner, ensuring censorship resistance and data integrity.
- **Content Distribution:** Facilitates the distribution of content across a decentralized network, reducing reliance on centralized hosting providers.
- **Blockchain-Based Applications:** Supports the deployment of blockchain-based applications and decentralized services, leveraging the security and immutability of blockchain technology.
- **Online Freedom:** Promotes online freedom and digital sovereignty by providing users with alternative hosting solutions that are resistant to censorship and control.

2.6 Software Resources Required

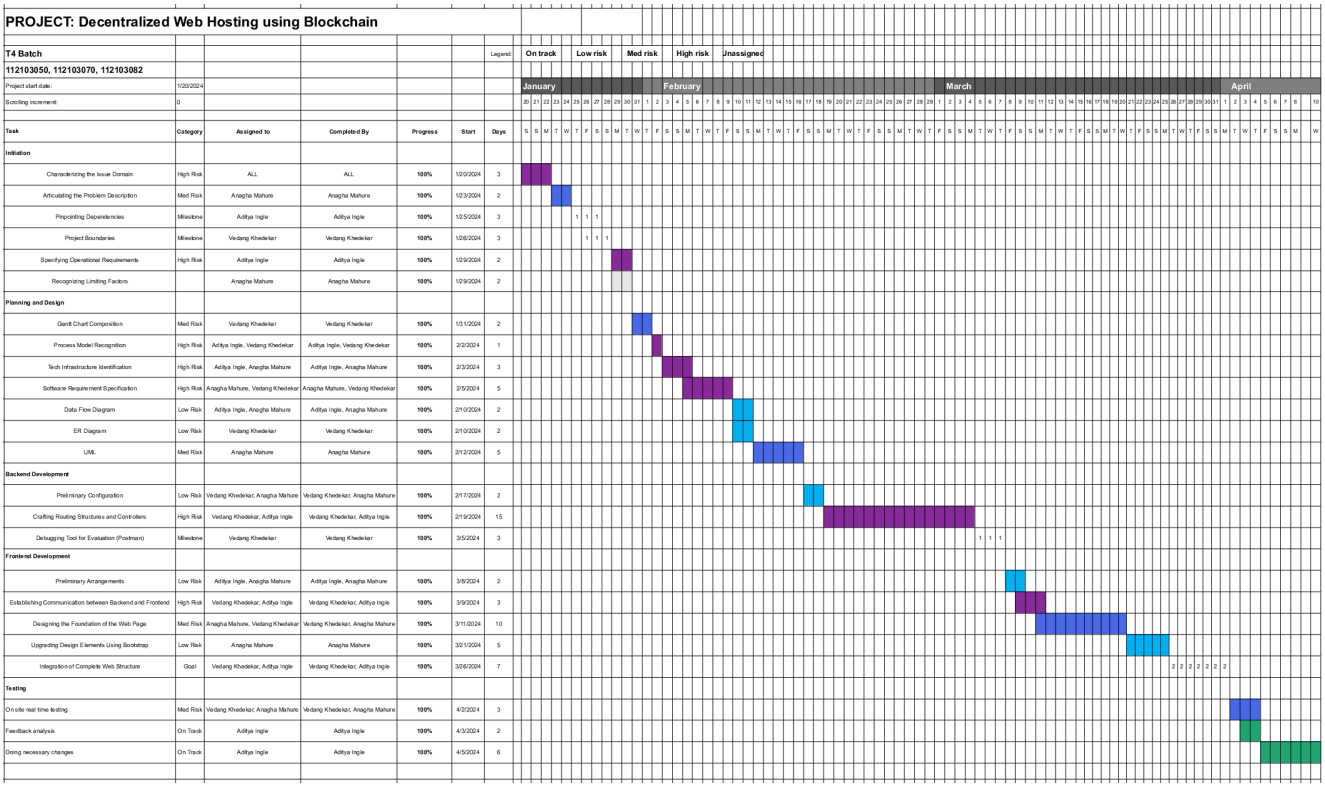
1. **Blockchain Platform:** Utilize blockchain platforms such as Ethereum or IPFS for decentralized storage and consensus mechanisms.
2. **Smart Contract Development Tools:** Tools for developing smart contracts, including languages like Solidity.
3. **Frontend Development Tools:** JavaScript frameworks like nextjs for developing user interfaces and frontend components.
4. **Decentralized Storage Solutions:** Integration with decentralized storage protocols such as IPFS or Filecoin for storing website data.
5. **Version Control System:** Version control software such as Git is necessary for managing the application's source code, tracking changes, and collaborating with team members.

Chapter 3

Project Plan

3.1 Project Schedule

3.1.1 Gantt Chart



PROJECT: Decentralized Web Hosting using Blockchain

T4 Batch

112103050, 112103070, 112103082

Project start date: 1/20/2024

Scrolling increment: 0

Legend: On track | Low risk | Med risk | High risk | Unassigned

Task	Category	Assigned to	Completed By	Progress	Start	Days
Initiation						
Characterizing the Issue Domain	High Risk	ALL	ALL	100%	1/20/2024	3
Articulating the Problem Description	Med Risk	Anagha Mahure	Anagha Mahure	100%	1/23/2024	2
Pinpointing Dependencies	Milestone	Aditya Ingle	Aditya Ingle	100%	1/25/2024	3
Project Boundaries	Milestone	Vedang Khedekar	Vedang Khedekar	100%	1/26/2024	3
Specifying Operational Requirements	High Risk	Aditya Ingle	Aditya Ingle	100%	1/29/2024	2
Recognizing Limiting Factors		Anagha Mahure	Anagha Mahure	100%	1/29/2024	2
Planning and Design						
Gantt Chart Composition	Med Risk	Vedang Khedekar	Vedang Khedekar	100%	1/31/2024	2
Process Model Recognition	High Risk	Aditya Ingle, Vedang Khedekar	Aditya Ingle, Vedang Khedekar	100%	2/2/2024	1
Tech Infrastructure Identification	High Risk	Aditya Ingle, Anagha Mahure	Aditya Ingle, Anagha Mahure	100%	2/3/2024	3
Software Requirement Specification	High Risk	Anagha Mahure, Vedang Khedekar	Anagha Mahure, Vedang Khedekar	100%	2/5/2024	5
Data Flow Diagram	Low Risk	Aditya Ingle, Anagha Mahure	Aditya Ingle, Anagha Mahure	100%	2/10/2024	2
ER Diagram	Low Risk	Vedang Khedekar	Vedang Khedekar	100%	2/12/2024	2
UML	Med Risk	Anagha Mahure	Anagha Mahure	100%	2/12/2024	5
Backend Development						
Preliminary Configuration	Low Risk	Vedang Khedekar, Anagha Mahure	Vedang Khedekar, Anagha Mahure	100%	2/17/2024	2
Crafting Routing Structures and Controllers	High Risk	Vedang Khedekar, Aditya Ingle	Vedang Khedekar, Aditya Ingle	100%	2/19/2024	15
Debugging Tool for Evaluation (Postman)	Milestone	Vedang Khedekar	Vedang Khedekar	100%	3/5/2024	3
Frontend Development						
Preliminary Arrangements	Low Risk	Aditya Ingle, Anagha Mahure	Aditya Ingle, Anagha Mahure	100%	3/8/2024	2
Establishing Communication between Backend and Frontend	High Risk	Vedang Khedekar, Aditya Ingle	Vedang Khedekar, Aditya Ingle	100%	3/9/2024	3
Designing the Foundation of the Web Page	Med Risk	Anagha Mahure, Vedang Khedekar	Vedang Khedekar, Anagha Mahure	100%	3/11/2024	10
Upgrading Design Elements Using Bootstrap	Low Risk	Anagha Mahure	Anagha Mahure	100%	3/21/2024	5
Integration of Complete Web Structure	Goal	Vedang Khedekar, Aditya Ingle	Vedang Khedekar, Aditya Ingle	100%	3/26/2024	7
Testing						
On-site real time testing	Med Risk	Vedang Khedekar, Anagha Mahure	Vedang Khedekar, Anagha Mahure	100%	4/2/2024	3
Feedback analysis	On Track	Aditya Ingle	Aditya Ingle	100%	4/3/2024	2
Doing necessary changes	On Track	Aditya Ingle	Aditya Ingle	100%	4/5/2024	6

Chapter 4

Software requirement specification

4.1 Introduction

4.1.1 Use-cases

- **Sign Up (Node Registration):** Users create an account on the decentralized web hosting platform by providing necessary information like email, and password. Upon submission, the platform generates a unique identifier (public key) for the user's node, allowing them to participate in the network.
- **Reset Password:** Users can reset their password by initiating a password reset request through the platform.
- **Smart Contracts:** Smart contracts within the decentralized web hosting system facilitate resource allocation, content verification, user authentication, website deployment, file management
- **File Upload:**Users upload files to the decentralized web hosting platform using a file upload interface.
- **File Updates Using Online IDE:** Users can update their files using an online Integrated Development Environment (IDE) provided by the platform.

- **Website Hosting:** Users deploy their websites on the decentralized network by uploading website files and associated resources.
- **Website Access (User Authentication):** Users authenticate themselves to access their hosted websites securely. Once authenticated, users can interact with the website and access its content.

4.1.2 Use Case View

Use Case Diagram:

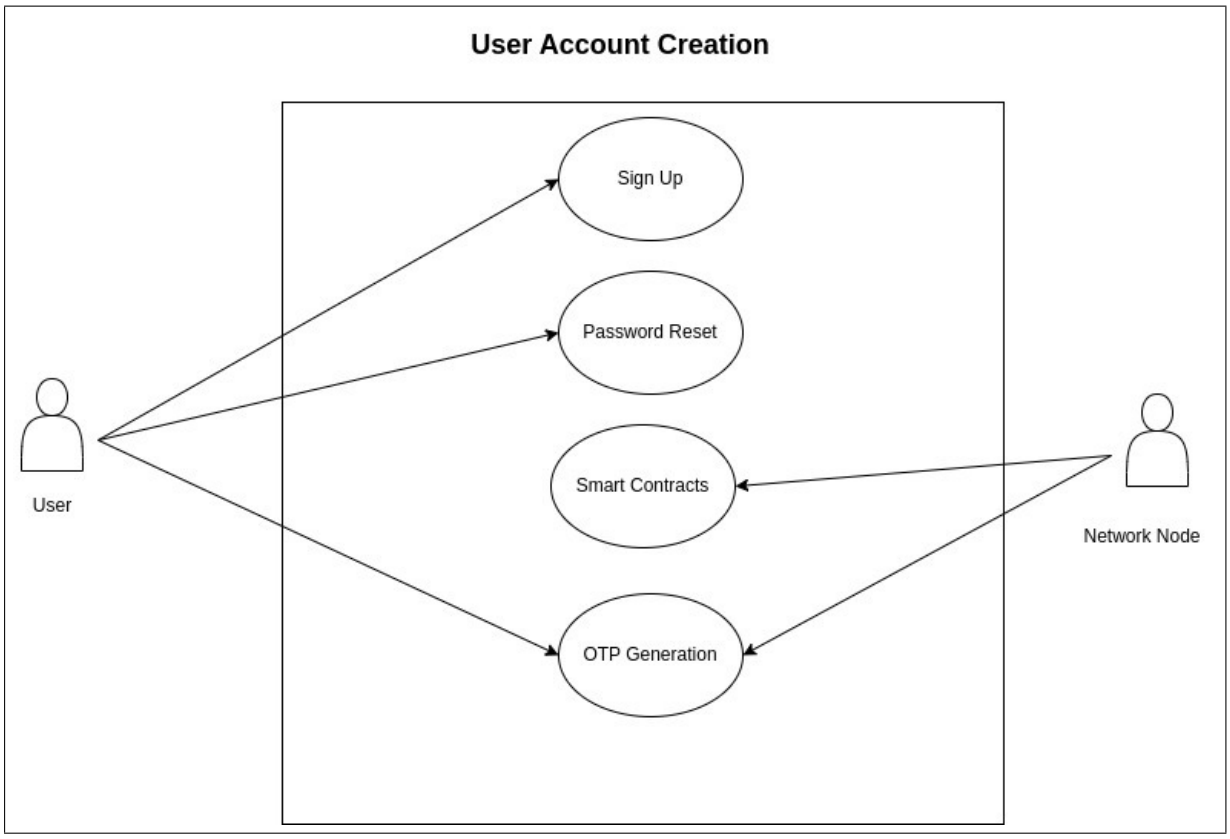


Figure 4.1: Use case diagram - User Account Creation

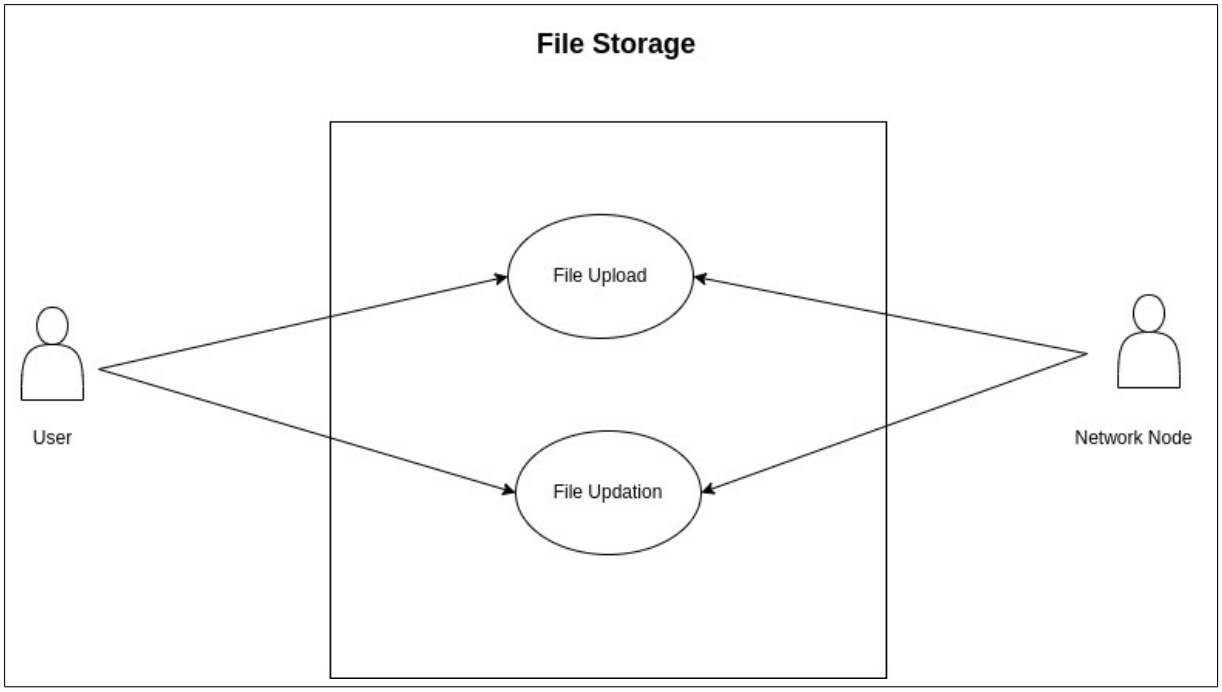


Figure 4.2: User Case Diagram - File Storage

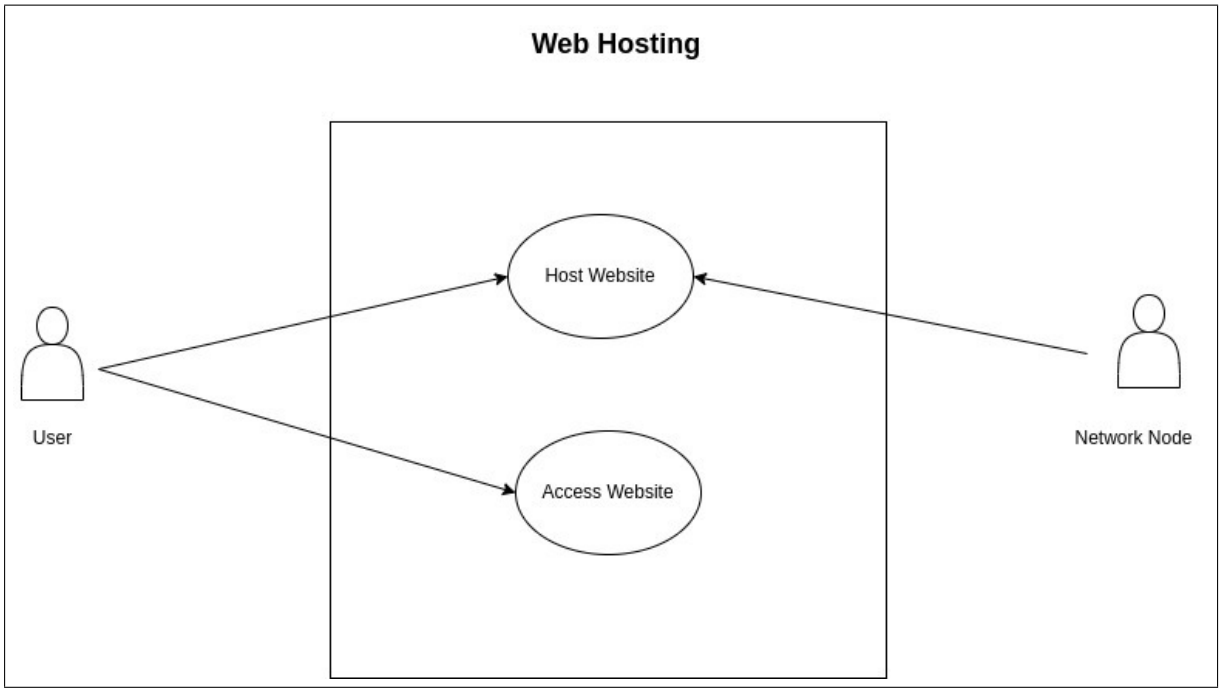


Figure 4.3: User Case Diagram - Web Hosting

4.2 Data Model and Description

4.2.1 Data objects and Relationships

- **User:** Represents an individual or entity participating in the decentralized web hosting network. Users register, authenticate, and engage in transactions within the network.
- **Website:**Represents a hosted website on the decentralized network. Users deploy and manage their websites through the platform.
- **Node:** Represents a node in the decentralized network, which provides hosting resources. Nodes register, authenticate, and allocate hosting resources to websites.
- **Transaction:**Records the exchanges of hosting services, payments, or other interactions within the decentralized network.

Entity Relationship Diagram:

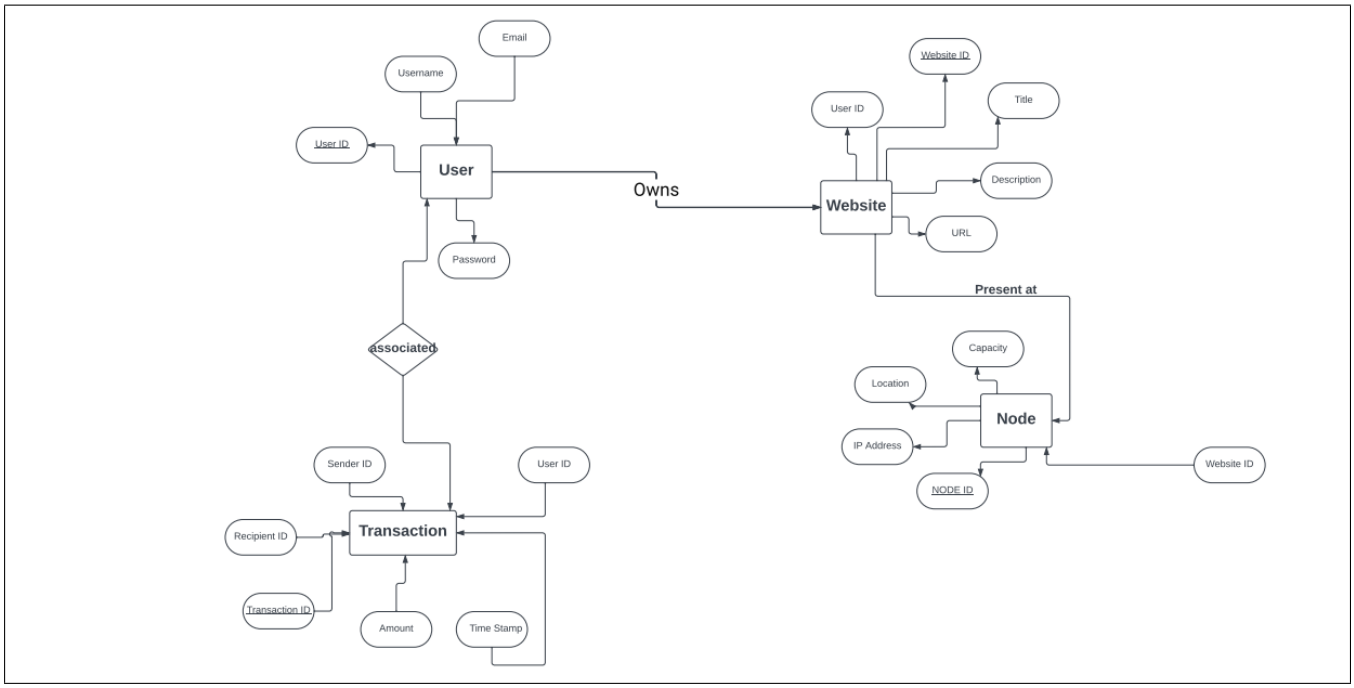


Figure 4.4: Entity Relationship diagram

4.3 Functional Model and Description

4.3.1 Functional Requirements:

- **Node Registration:** Nodes register on the decentralized network by providing necessary information and authenticating their identity.
- **Website Deployment:** Users deploy their websites on the decentralized network, specifying hosting requirements and preferences.
- **Content Verification:**Nodes verify the integrity of hosted content to prevent tampering or unauthorized modifications, ensuring the accuracy and reliability of hosted websites.

- **User Authentication:**Users authenticate themselves securely to access hosting services and manage their websites on the decentralized network.

4.3.2 Data Flow Diagram

Level 0 Data Flow Diagram

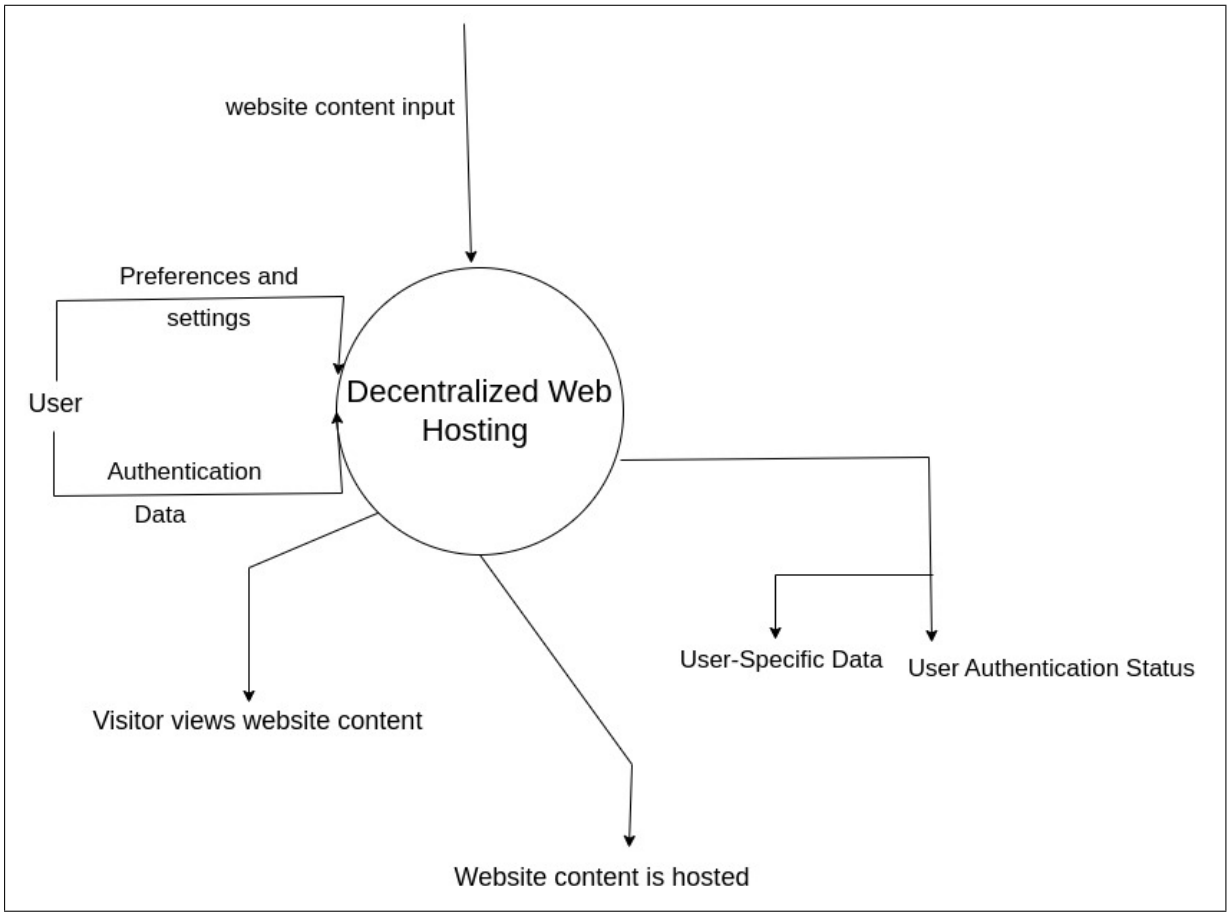


Figure 4.5: DFD Level0

Level 1 Data Flow Diagram

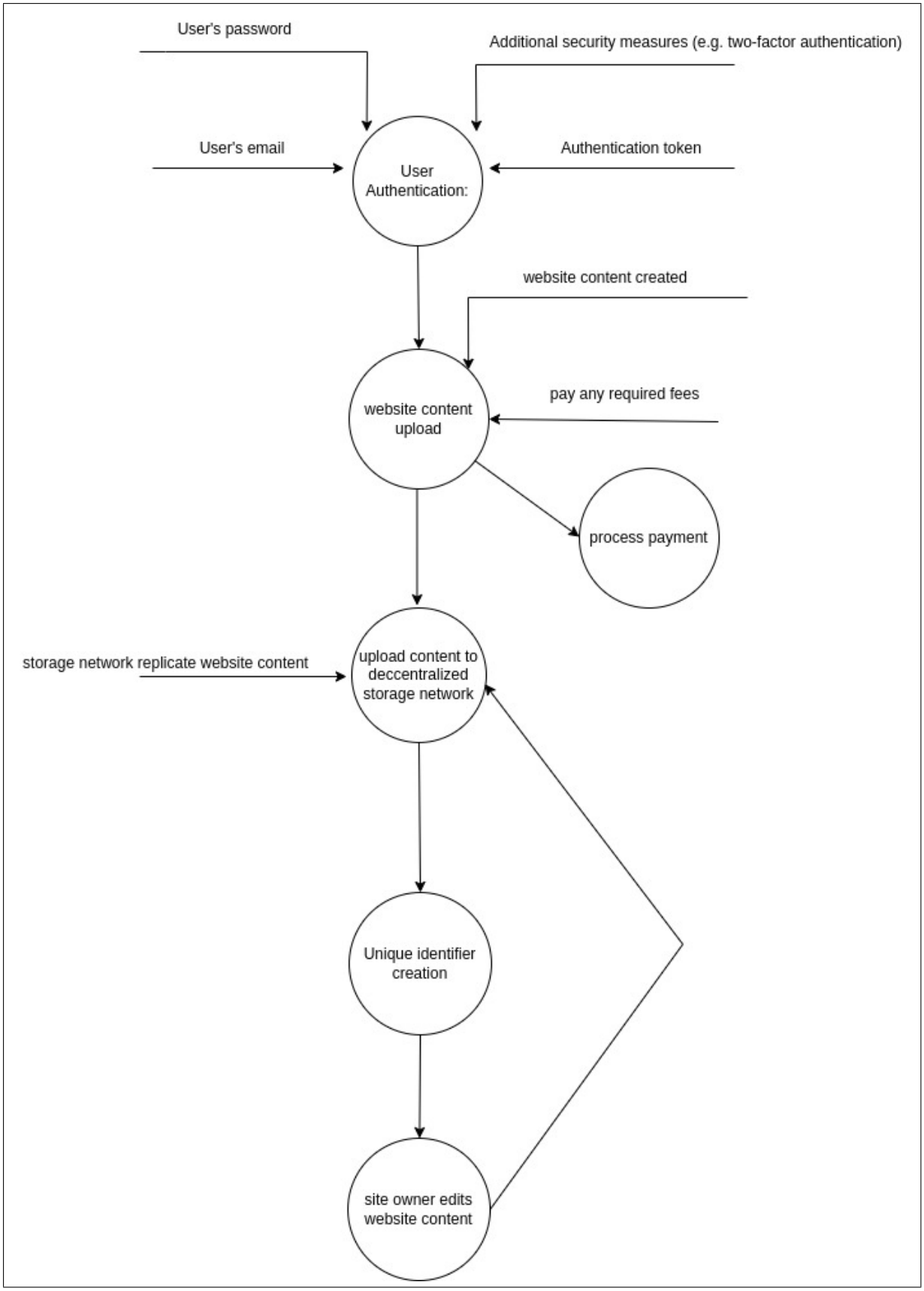


Figure 4.6: DFDLevel1

4.3.3 Description of functions

4.3.4 Activity Diagram:

- The Activity diagram represents the steps taken.

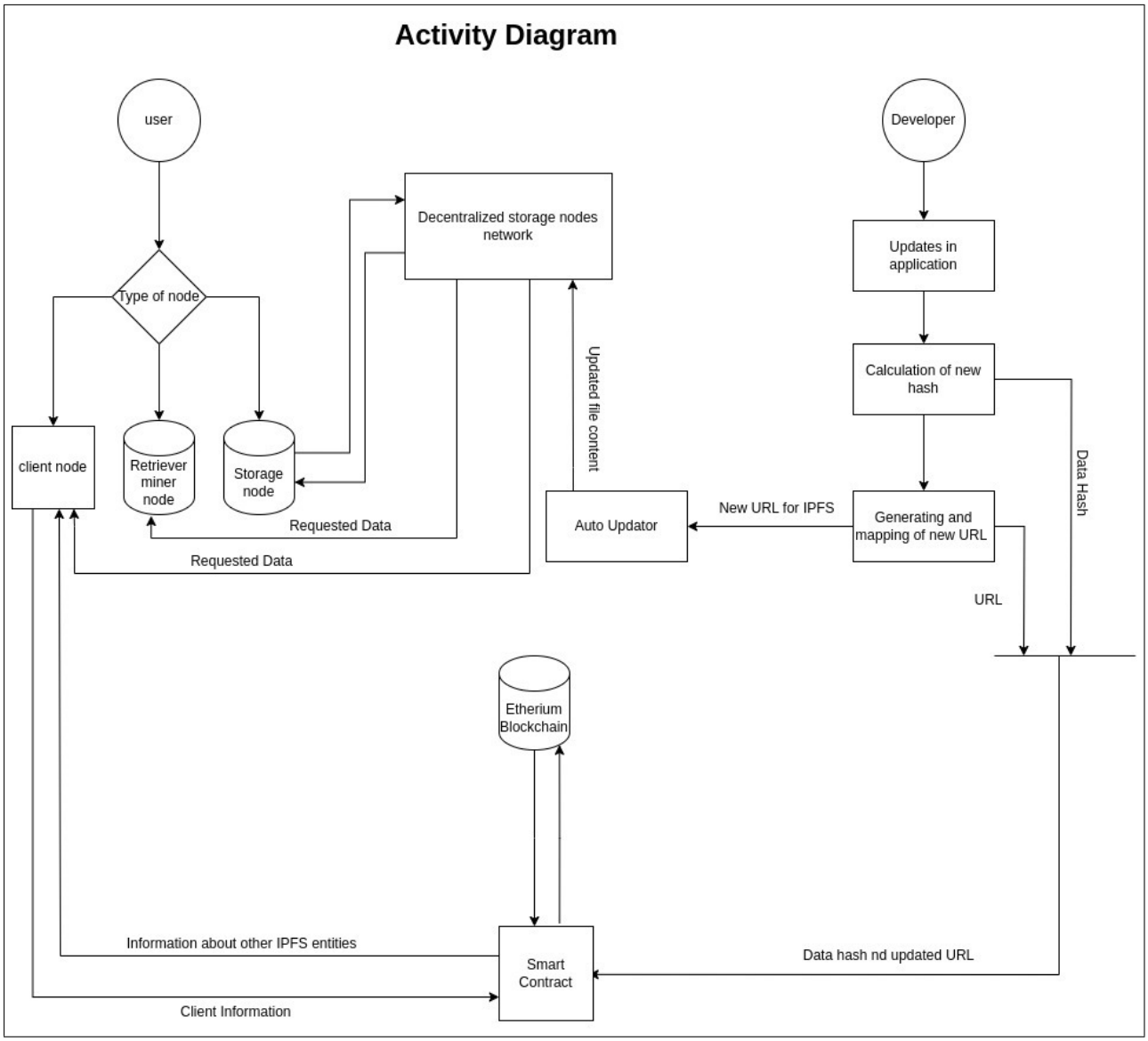


Figure 4.7: Activity diagram

4.3.5 Non Functional Requirements:

Performance Requirements

- **Response Time:**The decentralized web hosting system should exhibit low latency in responding to user actions such as website deployment, updates, and content retrieval.
- **Scalability:** The network infrastructure should be capable of expanding seamlessly to accommodate a growing user base and increasing demand for hosting resources without compromising performance.
- **Concurrency:** The system should effectively handle multiple simultaneous interactions, allowing users to deploy, manage, and access websites concurrently without experiencing delays or resource contention issues.
- **Error Handling:**The system should be robust in handling errors, providing clear and informative feedback to users and administrators in cases of failed deployments, transaction errors, or other system anomalies.

Safety and Security Requirements:

- **Data Privacy:** - User data and transaction details should be securely stored and encrypted to prevent unauthorized access.
- **Content Integrity:** Hosted content should be verified to ensure accuracy and reliability, minimizing the risk of fraudulent or misleading information.

4.3.6 Design Constraints

- **Data acquisition**

- **Adaptation to Network Changes:**The system should adapt to changes in network topology and node availability.
- **Compliance with Ethical Guidelines:** Data acquisition techniques must comply with ethical standards and regulations.

- **Client-side constraints**
  - **Offline Access:** Users should be able to access basic functionalities even when offline.
  - **Synchronization of Data:** Client-side and server-side data should be synchronized to maintain consistency.

- **Data processing**
  - **Optimization of performance:** Data processing should be optimized to handle large volumes of hosting requests efficiently without compromising performance.

Chapter 5

Detailed Design Document

5.1 Component Design

5.1.1 Class Diagram

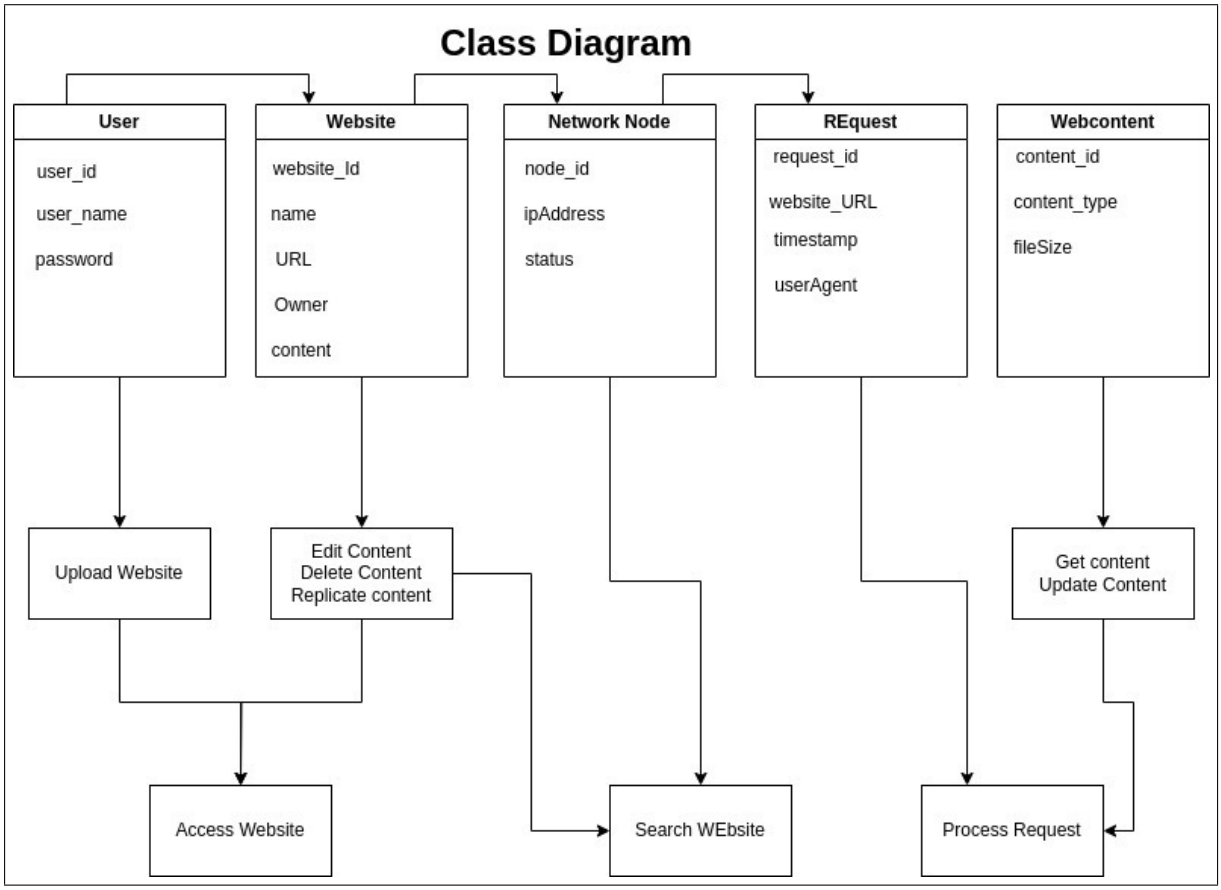


Figure 5.1: Class Diagram

5.1.2 Sequence Diagram

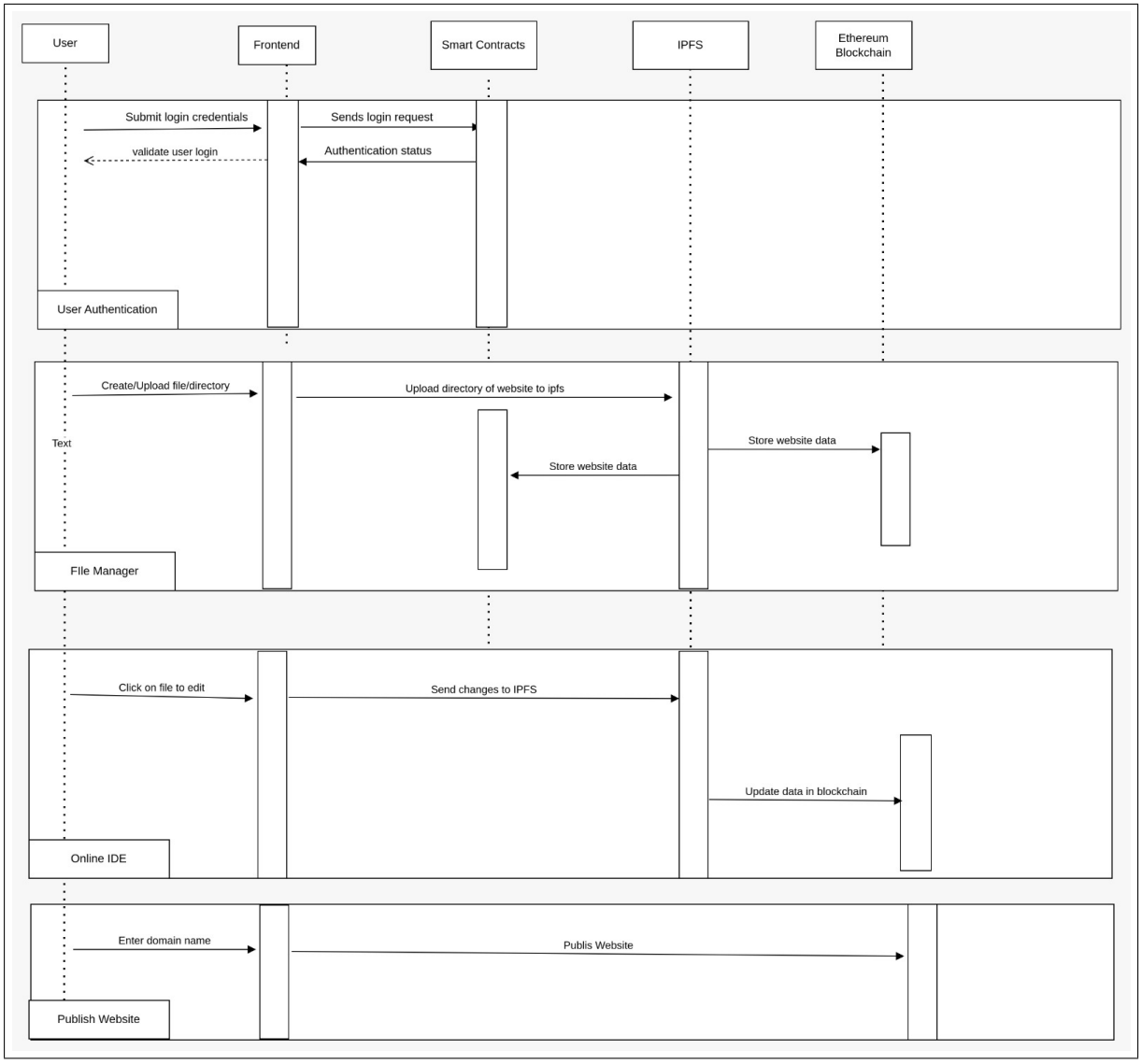


Figure 5.2: Sequence Diagram

5.1.3 Component Diagram

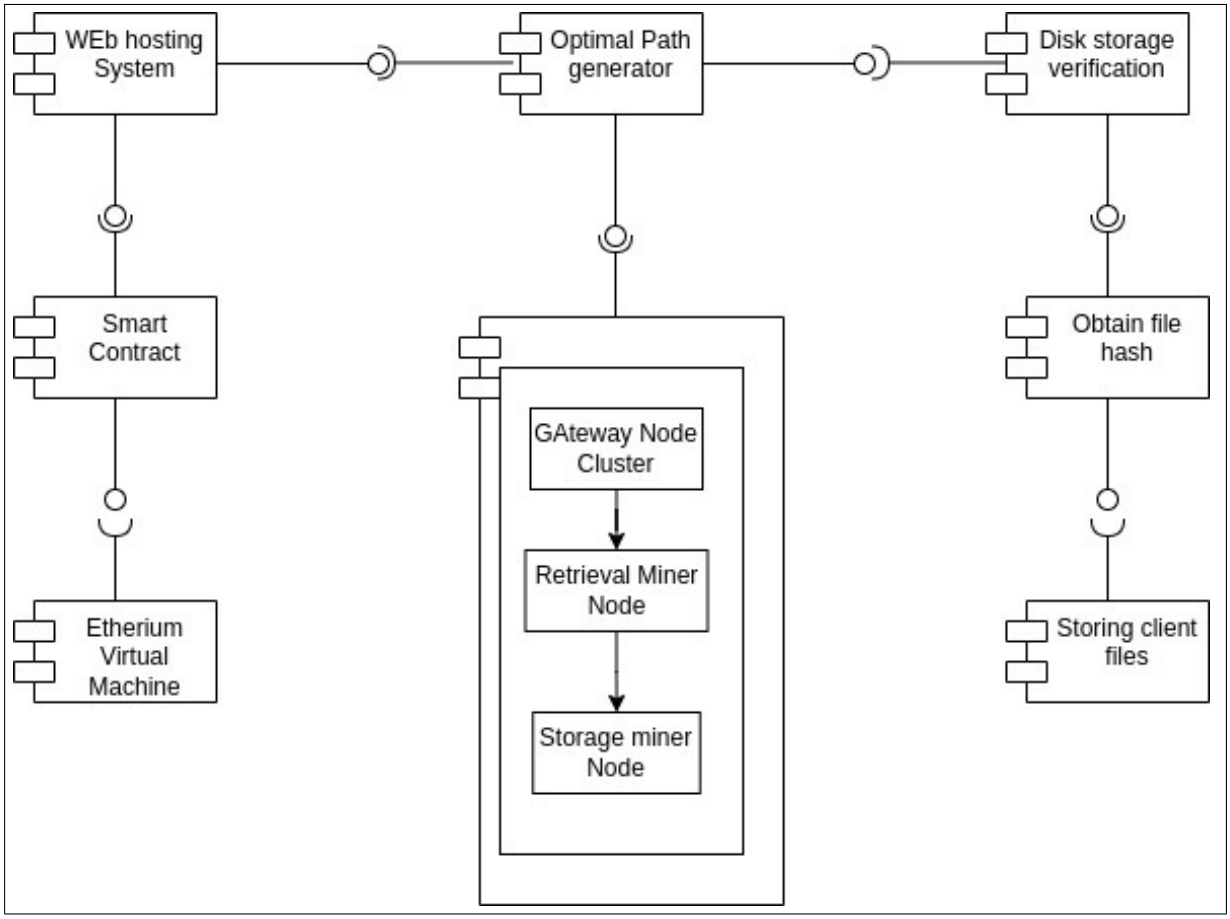


Figure 5.3: Component Diagram

5.1.4 Deployment Diagram

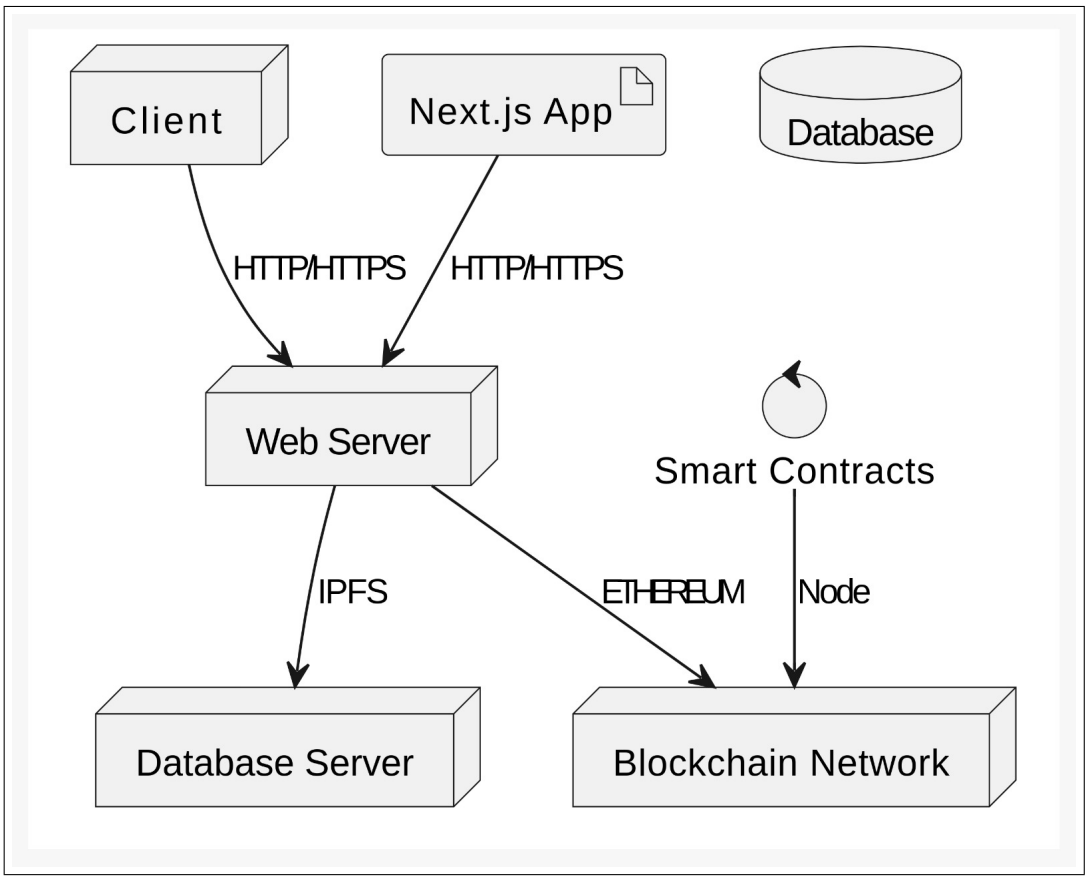


Figure 5.4: Deployment Diagram

5.2 Navigation Flow

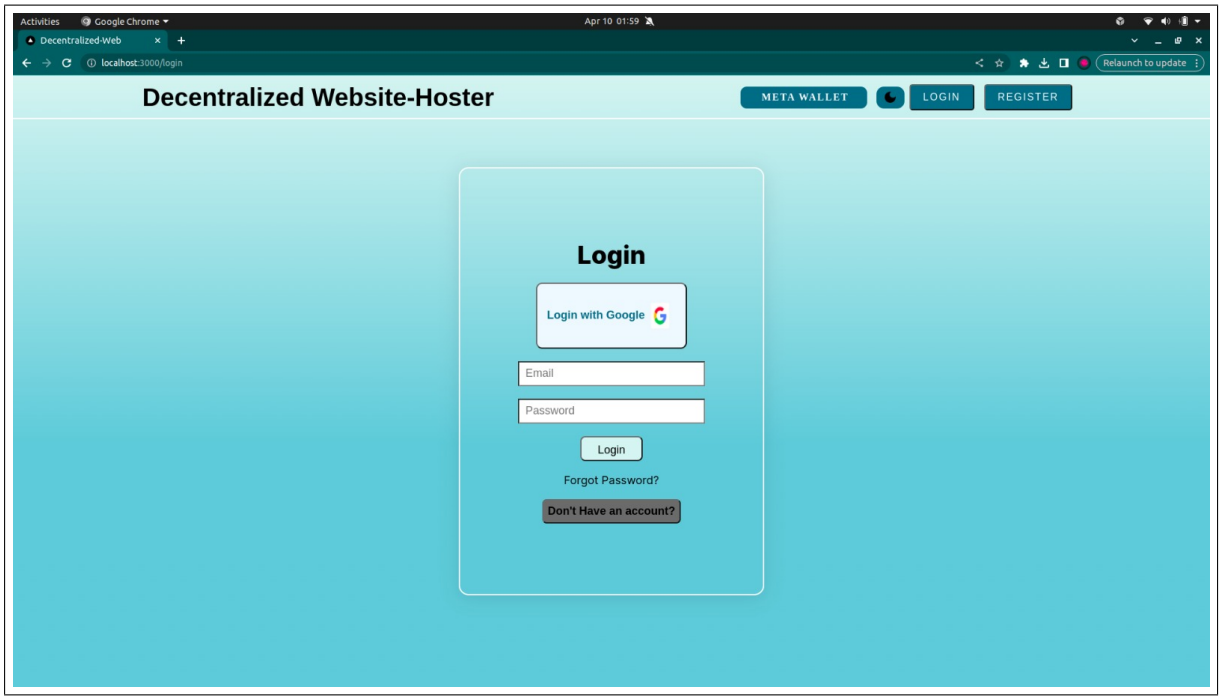


Figure 5.5: Login

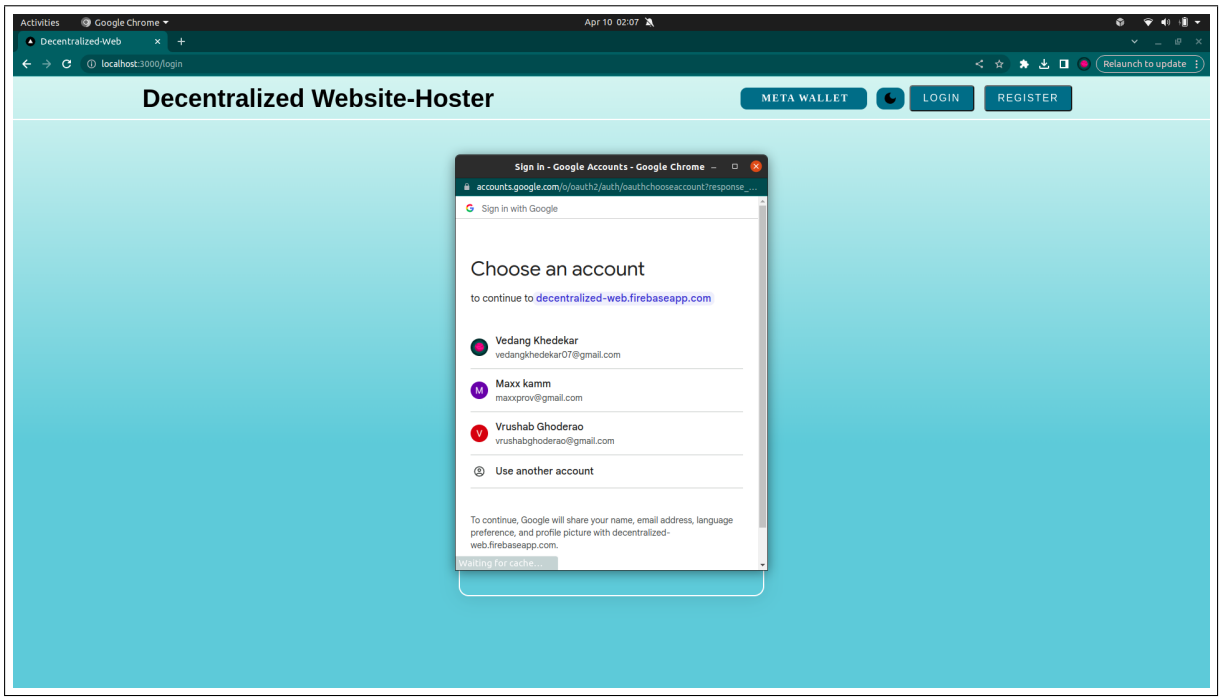


Figure 5.6: User Authentication

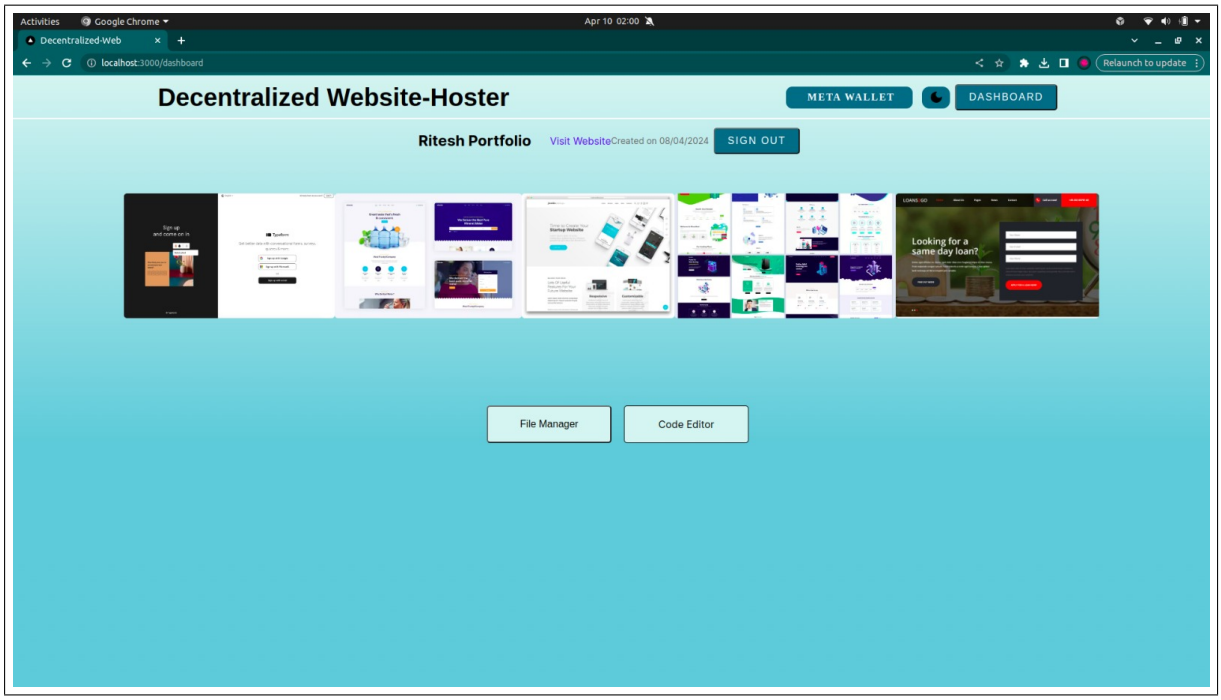


Figure 5.7: Dashboard

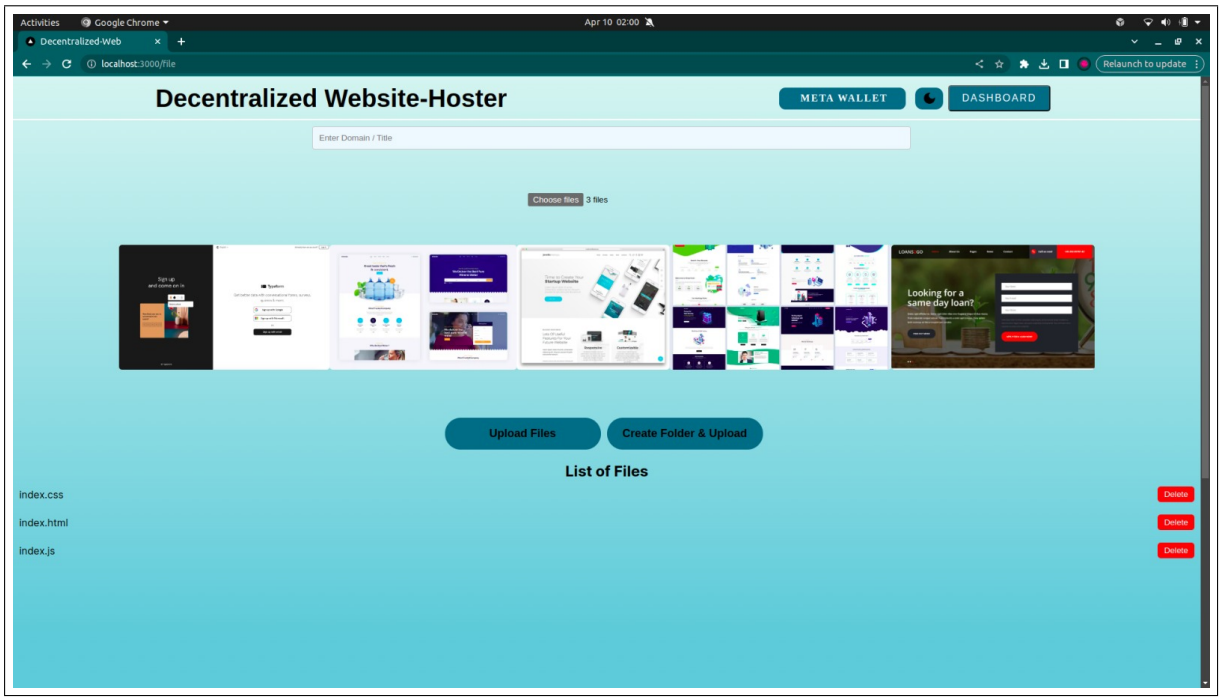


Figure 5.8: File Upload

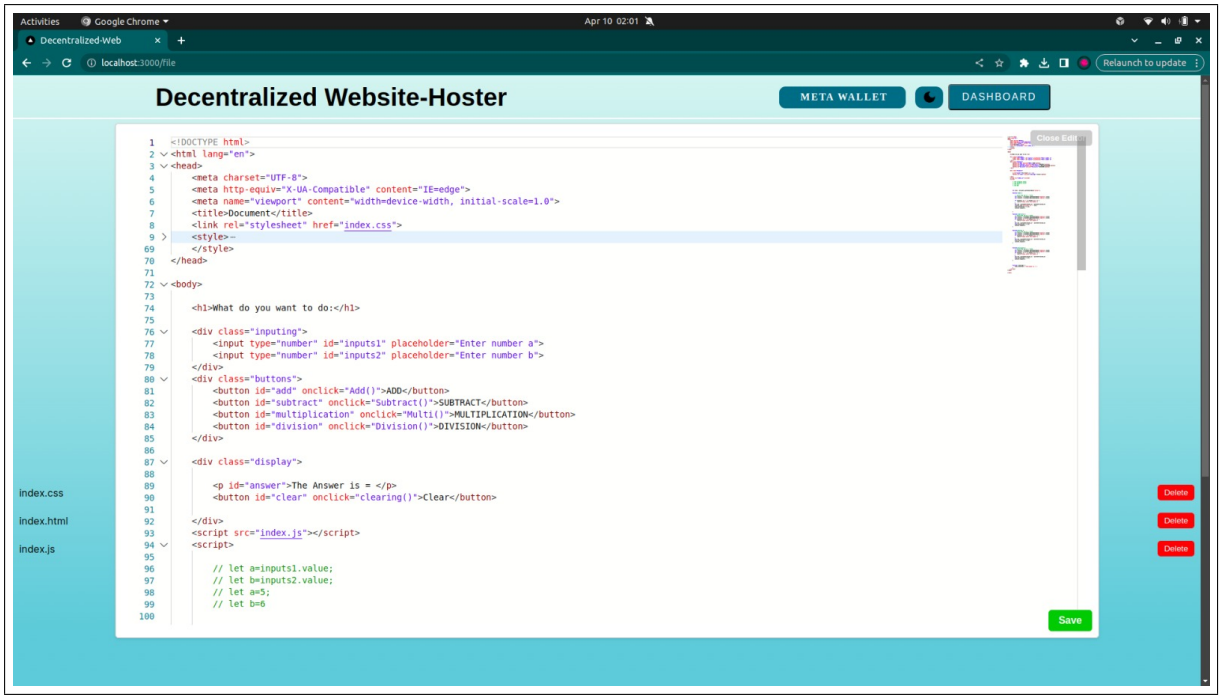


Figure 5.9: File Modification

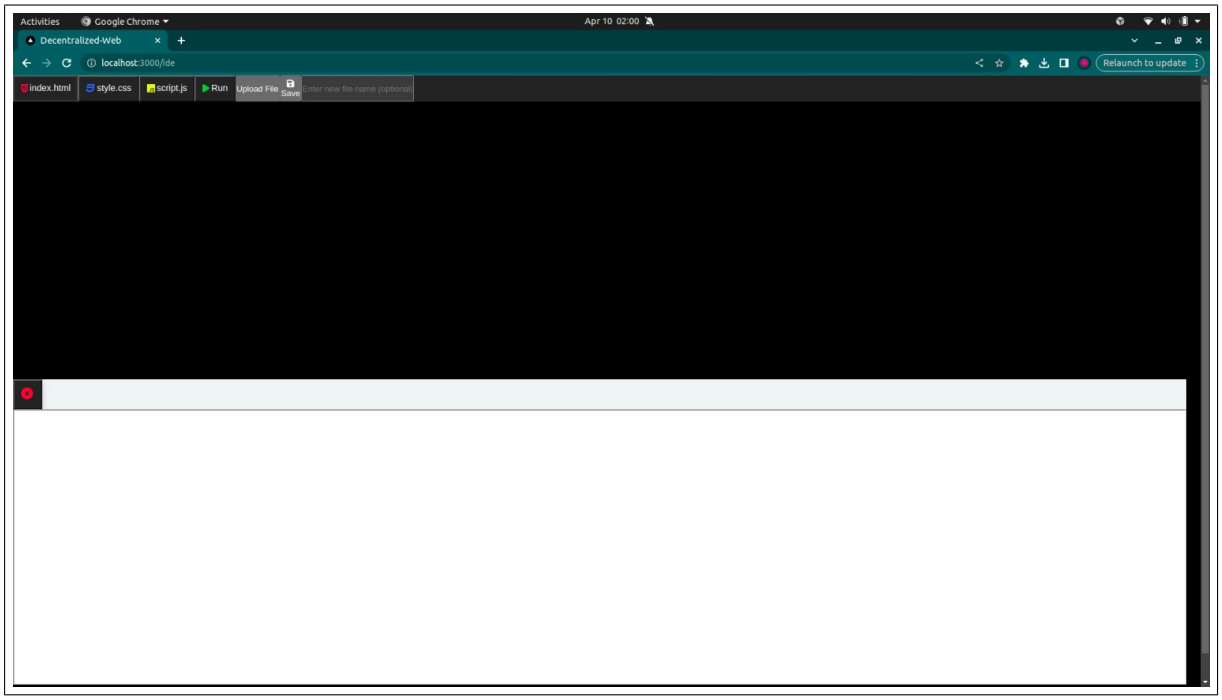


Figure 5.10: Online IDE

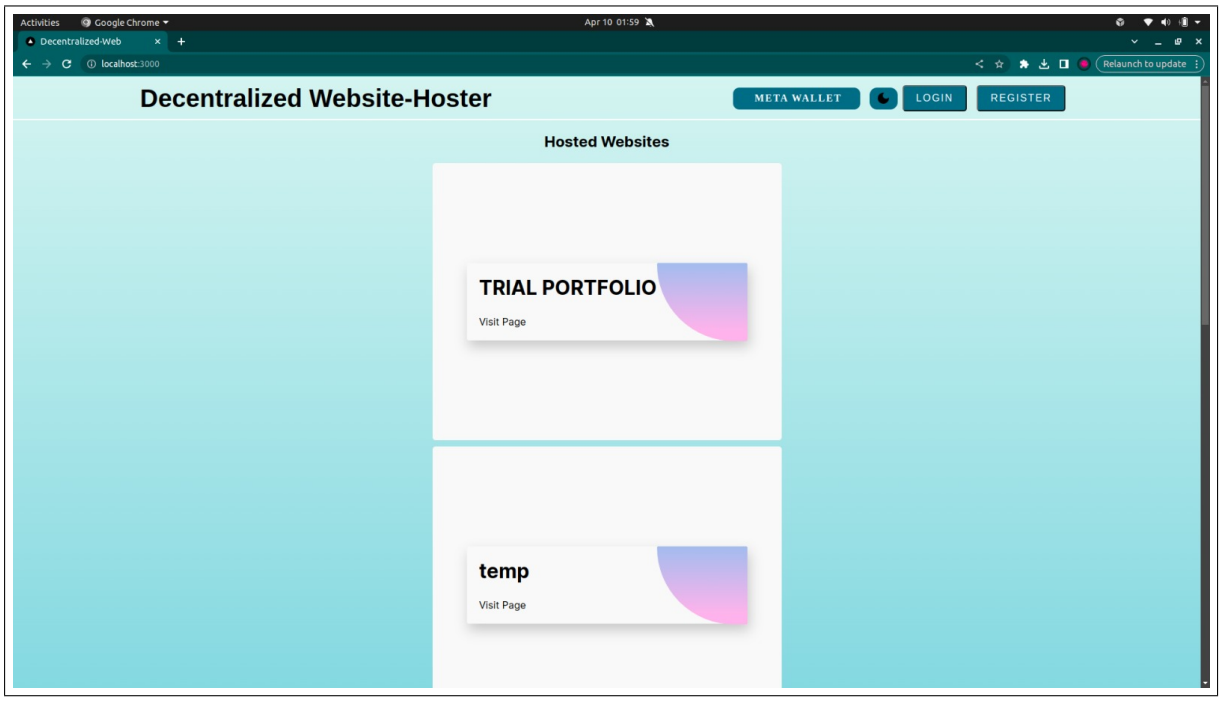


Figure 5.11: main

Figure 5.12: Website Hosted

Chapter 6

Summary and Conclusion

The developed decentralized web hosting system offers users the capability to host various websites on a decentralized network. Utilizing the InterPlanetary File System (IPFS), users can upload their websites, ensuring content availability and resilience through decentralized storage. Key components of the system include nodes, which provide hosting resources, and smart contracts that govern hosting agreements between users and hosting nodes. The system emphasizes performance, scalability, concurrency, and robust error handling to provide a seamless experience for users deploying and managing their websites.

Through rigorous development and testing, the system demonstrates efficient response times, scalability to accommodate increasing user and website demands, support for multiple concurrent interactions, and graceful error handling mechanisms. Leveraging decentralized technologies, the system offers users a decentralized and resilient hosting solution that ensures website availability and reliability.

In conclusion, the developed decentralized web hosting system represents a significant advancement in hosting solutions, offering users a decentralized alternative to traditional hosting services. By harnessing the power of IPFS and smart contracts, the system provides a robust, scalable, and fault-tolerant platform for hosting various websites. The

system's emphasis on performance, scalability, concurrency, and error handling ensures a seamless user experience while maintaining the integrity and availability of hosted content.

Moving forward, further enhancements and optimizations can be explored to enhance system capabilities and address emerging challenges in decentralized web hosting. Additionally, ongoing monitoring and maintenance will be essential to uphold system performance and reliability as the user base and hosted content continue to grow. Overall, the developed decentralized web hosting system represents a promising solution in the decentralized web landscape, offering users a reliable and resilient platform for hosting their websites.