A

MINI PROJECT

REPORT ON

"A BLOCKCHAIN BASED DECENTRALIZED AUTONOMOUS ONLINE SOCIAL NETWORK"

This mini project Submitted in partial fulfillment of the requirements

For the award of the degree of

BACHELOR OF TECHNOLOGY
IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

M.SIDDU GANESH	**19TU1A0508**
J.YAMINI	**19TU1A0514**
P.JAI RAM	**19TU1A0510**

Under the guidance of

MRS. B RAMA LAKSHMI

(Asst.Prof)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ELLENKI INSTITUTE OF ENGINEERING & TECHNOLOGY

PATELGUDA, AMEENPUR, SANGAREDDY DISTRICT - 502319
Affiliated to JNTU, Hyderabad

2022-2023

ELLENKI INSTITUTE OF ENGINEERING & TECHNOLOGY

PATELGUDA, AMEENPUR, SANGAREDDY DISTRICT – 502319

Affiliated to JNTU Hyderabad.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the mini project entitled **"A BLOCKCHAIN BASED DECENTRALIZED AUTONOMOUS ONLINE SOCIAL NETWORK"** being submitted by **M.SIDDU GANESH, J.YAMINI & P.JAI RAM** bearing Roll No.**19TU1A0508, 19TU1A0514 & 19TU1A0510** in partial fulfillment of the requirements for the award of degree of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE AND ENGINEERING,** under **Jawaharlal Nehru Technological University,** Hyderabad, is a record of bonafide work carried out by them under by guidance and supervision. The results embodied in this Mini Project report have not been submitted to any other university or institute for the award of any degree.

INTERNAL GUIDE

B RAMA LAKSHMI
(Asst.Prof)

HEAD OF THE DEPARTMENT

Dr. SHAIK MAHABOOB SHARIEF
(Professor)

EXTERNAL EXAMINER

PRINCIPAL

Dr.P.JOHN PAUL
(Professor)

ACKNOWLEDGEMENT

We would like to thank **Sri.Dr E. SADASHIVA REDDY, Chairman, Ellenki Group of Institutions** for providing us the conducive environment for carrying through my academic schedules and Mini Projects with ease.

We express a whole hearted gratitude to **Sri M. SAMBHA SHIVA REDDY, Director, Ellenki Institute Of Engineering and Technology,** for providing us the conducive environment for carrying through my academic schedules and Mini projects with ease.

we would like to thank our esteemed secretary **Sri E.DAYAKAR REDDY** and Academic Director **Sri E.SAI KIRAN REDDY** for their invaluable supervision, support and guidance throughout our study in college.

We would like to thank our **Principal Dr. P. JOHN PAUL ,Ellenki Institute Of Engineering and Technology.** His insight during the Mini Project and regular guidance were invaluable to me and also for providing right suggestions at every phase of the development of my Mini Project.

We would like to thank our **Head of the Department (Computer Science and Engineering), Dr. SHAIK MAHABOOB SHARIEF** for providing seamless support and knowledge over the past one year and also for providing right suggestions at every phase of the development of my Mini Project.

We would like to thank our guide **B RAMA LAKSHMI, Department of Computer Science and Engineering, Ellenki Institute Of Engineering and Technology,** for providing seamless support and knowledge over the past one year and also for providing right suggestions at every phase of the development of my Mini Project.

Our sincere thanks to all our faculty members for their guidance and support throughout our study at college and we also thank all our administrative staff, examination branch staff for timely responding to our queries

There is definitely a need to thank my friends and parents for their patience and support in always leading to successful completion of the Mini Project.

M.SIDDU GANESH	**19TU1A0508**
J.YAMINI	**19TU1A0514**
P.JAI RAM	**19TU1A0510**

ABSTRACT

Online social networks (OSN) are becoming more important in people's daily life, however, all popular OSNs are centralized, and this raises a series of security, privacy and management issues. A decentralized architecture based on blockchain technology provides the ability to solve above issues. In this paper, an OSN service is developed based on blockchain technology in order to make it operate decentralized. Large volume of data normally required low-security requirements can be stored in Interplanetary Filesystem (IPFS) to make data decentralized. A decentralized autonomous organization is developed for user autonomy, users can self-manage the OSN in a democratic way.

Keywords - Blockchain, Online Social Network (OSN), Decentralized Autonomous Organization (DAO)

Contents

LIST OF FIGURES

LIST OF SCREENSHOTS

CHAPTER-01

INTRODUCTION

1.INTRODUCTION

Online Social Network (OSN) is a platform for people to build connections with each other via the Internet. It is a major platform that the public can obtain and disseminate information, exchange views and share their lives on it. Research from Chaffey [1] reveals the liveness of top used OSN in the world (Figure 1), therefore, it can be found that interacting with OSN is a very popular online activity for Internet users.

Nowadays, most of OSNs are centralized, which means the OSN companies often have full ownership of all user data and service. In general, users can only use the service after they agree to the agreements of OSN which are enacted by OSN companies. However, many agreements give the OSN companies right to use user data for personalized services such as advertisement. If users do not allow the companies to use their data and protect their privacy, they usually have to make a series of expatiatory applications or even give up using such OSN. Data and service centralization also caused all data of users is uploaded and stored in centralized servers which are controlled by OSN companies. Therefore, it is hard for users to protect their contents on the OSN when the servers crash down. To make matters worse, if the servers are hacked, security information includes password, security problems, address of users is possible to be leaked. For many users using the same password in kinds of sites, hackers can easily hack their accounts by using a method named credential stuffing attack [2]. This makes personal information of users at risk of leakage and abuse. Such problems of centralized OSNs boost researchers to consider develop an OSN based on the decentralization framework.

1.1 PURPOSE

Decentralized OSNs have the potential to provide a safer and more controllable social network environment for users where privacy and information are more controllable for their owners. Because the data is stored in a distributed way and service is no longer relied on centralized servers. In general, a decentralized OSN is usually operated by a peertopeer mechanism in which each node stores some parts of data and support the service. However, it is not binding on malicious acts, and lack of self-management and sustainable developing abilities.

In this paper, we proposed an autonomous decentralized online social network architecture based on block chain technology. Block chain is able to provide a safe and trusted peer-to-peer mechanism where participants have unique identities and private keys. The private key has the highest control right of the corresponding account and is stored in user's own device. Moreover, all transactions in block chain need to be signed by the private key, so cheating can be avoided.

1.2 SCOPE

In order to give the system abilities of self-management and sustainable development, a decentralized autonomous mechanism powered by block chain is embedded in the architecture. The rest of this paper is organized as follows. Firstly, we introduce the background of related technologies used in this architecture. Secondly, a detail description of the architecture is discussed. Thirdly, functions of this project are showed. Finally, a conclusion is made.

CHAPTER-02

LITERATURE SURVEY

2.1 SMART CONTRACTS IMPLEMENTATION, APPLICATIONS, BENEFITS, AND LIMITATIONS

Author - Silas Nzuva

A smart contract can be defined as a self-executing contract that utilizes blockchain technology to digitally enforce, verify, or facilitate the performance or negotiation of a contract. Owing to the security and decentralized system exhibited by blockchain technology, smart contracts can foster transaction credibility between contracting parties without the necessity of third parties as exhibited in traditional contracts. Any business organization that aims at achieving greater heights in management and production dimensions must consider utilizing robust technologies that are aimed at bolstering its competitive edge. Owing to the newness of smart contracts, characterized by very few studies on the same, this research reviews how smart contracts through blockchain technology can be implemented in an organization to enhance performance and outlines the applications, benefits, and limitations associated with such contracts.

2.2 BITCOIN: A PEER-TO-PEER ELECTRONIC CASH SYSTEM

Author - Satoshi Nakamoto

A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort

basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

2.3 CREDENTIAL STUFFING EXPLAINED: HOW TO PREVENT, DETECT, AND DEFEND AGAINST IT

Author – Lucian Constantin

Credential stuffing is the automated use of collected usernames and passwords to gain fraudulent access to user accounts. Billions of login credentials have landed in the hands of hackers over the past several years as a result of data breaches. These credentials fuel the underground economy and are used for everything from spam to phishing and account takeovers. Credential stuffing attacks are one of the most common ways cybercriminals abuse stolen usernames and passwords.

A literature survey of blockchain-based ADOSNs would involve reviewing and summarizing existing research on this topic. This might include studies on the technical aspects of building and maintaining a decentralized social network, as well as research on the social and economic implications of such networks.

Some potential areas of focus for a literature survey of blockchain-based ADOSNs might include:

- The use of blockchain technology to enable decentralized decision-making and governance within the network
- The potential for ADOSNs to provide greater privacy and security for users compared to traditional social networks
- The role of ADOSNs in promoting freedom of expression and combating censorship
- The challenges and limitations of building and maintaining a decentralized social network, including issues related to scalability and network effects
- The potential economic implications of ADOSNs, including the use of cryptocurrency or other blockchain-based incentives to encourage participation and contribution within the network.

CHAPTER-03

SYSTEM REQUIREMENTS

3.1 SYSTEM REQUIREMENTS

3.1.1 HARDWARE REQUIREMENTS

SYSTEM CONFIGURATION:

- Processor - Pentium –IV
- RAM - 4 GB (min)
- Hard Disk - 20 GB
- Key Board - Standard Windows Keyboard
- Mouse - Two or Three Button Mouse
- Monitor - SVGA

3.1.2 SOFTWARE REQUIREMENTS

SOFTWARE REQUIREMENTS:

- Operating System - Windows XP
- Coding Language - Java/J2EE(JSP, Servlet)
- Front End - J2EE
- Back End - MySQL

CHAPTER-04

THEORETICAL BACKGROUND

4.1 INTRODUCTION

OVER VIEW: CLIENT SERVER

With the varied topic in existence in the fields of computers, Client Server is one, which has generated more heat than light, and also more hype than reality. This technology has acquired a certain critical mass attention with its dedication conferences and magazines. Major computer vendors such as IBM and DEC, have declared that Client Servers is their main future market. A survey of DBMS magazine reveled that 76% of its readers were actively looking at the client server solution. The growth in the client server development tools from $200 million in 1992 to more than $1.2 billion in 1996.

Client server implementations are complex but the underlying concept is simple and powerful. A client is an application running with local resources but able to request the database and relate the services from separate remote server.

4.1.2 FRONT END OR USER INTERFACE DESIGN

The entire user interface is planned to be developed in browser specific environment with a touch of Intranet-Based Architecture for achieving the Distributed Concept.

The browser specific components are designed by using the HTML standards, and the dynamism of the designed by concentrating on the constructs of the Java Server Pages.

4.1.3 COMMUNICATION OR DATABASE CONNECTIVITY TIER

The Communication architecture is designed by concentrating on the Standards of Servlets and Enterprise Java Beans. The database connectivity is established by using the Java Data Base Connectivity.

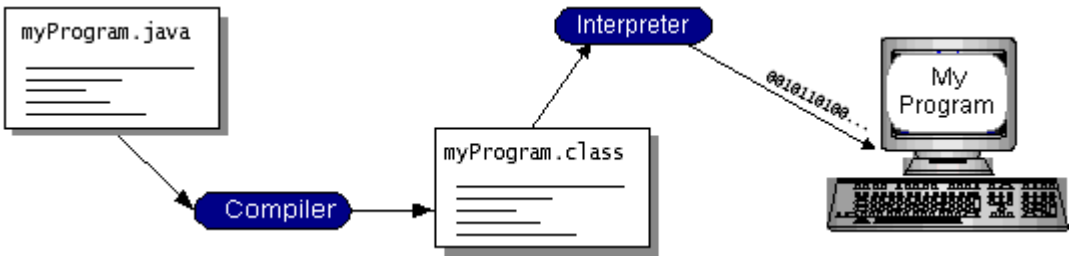
The standards of three-tier architecture are given major concentration to keep the standards of higher cohesion and limited coupling for effectiveness of the operations**.**

4.2 INTRODUCTION TO JAVA

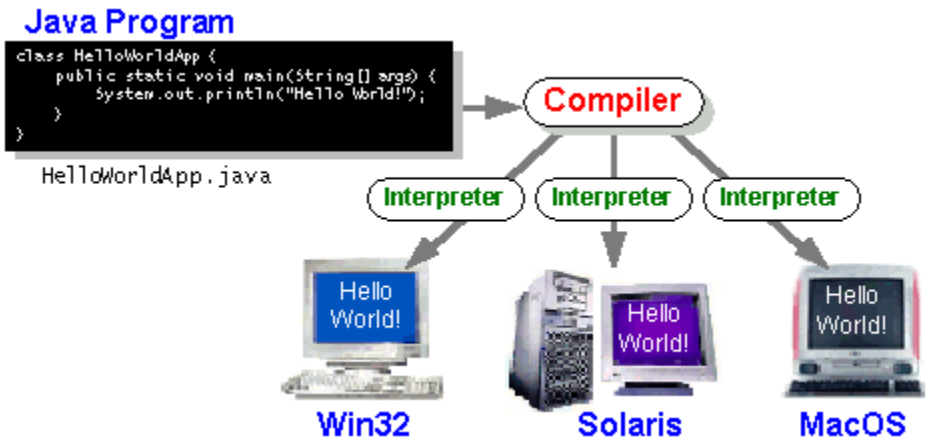
Java technology is both a programming language and a platform. The Java programming language is a high-level language that can be characterized by all of the following buzzwords

- Simple
- Architecture neutral
- Object oriented
- Portable
- Distributed
- High performance
- Interpreted
- Multithreaded
- Robust
- Dynamic
- Secure

With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted. With the compiler, first you translate a program into an intermediate language called *Java byte codes* —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed. The following figure illustrates how this works.



You can think of Java byte codes as the machine code instructions for the *Java Virtual Machine* (Java VM). Every Java interpreter, whether it's a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make "write once, run anywhere" possible. You can compile your program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. That means that as long as a computer has a Java VM, the same program written in the Java programming language can run on Windows 2000, a Solaris workstation, or on an iMac.



4.2.1THE JAVA PLATFORM

A *platform* is the hardware or software environment in which a program runs. We've already mentioned some of the most popular platforms like Windows 2000, Linux, Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.

The Java platform has two components:

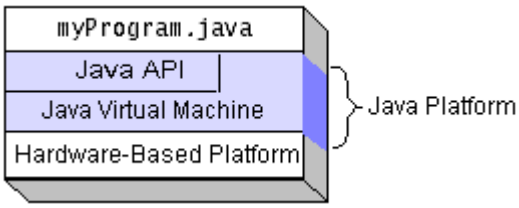
- The *Java Virtual Machine* (Java VM)
- The *Java Application Programming Interface* (Java API)

You've already been introduced to the Java VM. It's the base for the Java platform and is ported onto various hardware-based platforms.

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of

related classes and interfaces; these libraries are known as *packages*. The next section, What Can Java Technology Do? Highlights what functionality some of the packages in the Java API provide.

The following figure depicts a program that's running on the Java platform. As the figure shows, the Java API and the virtual machine insulate the program from the hardware.

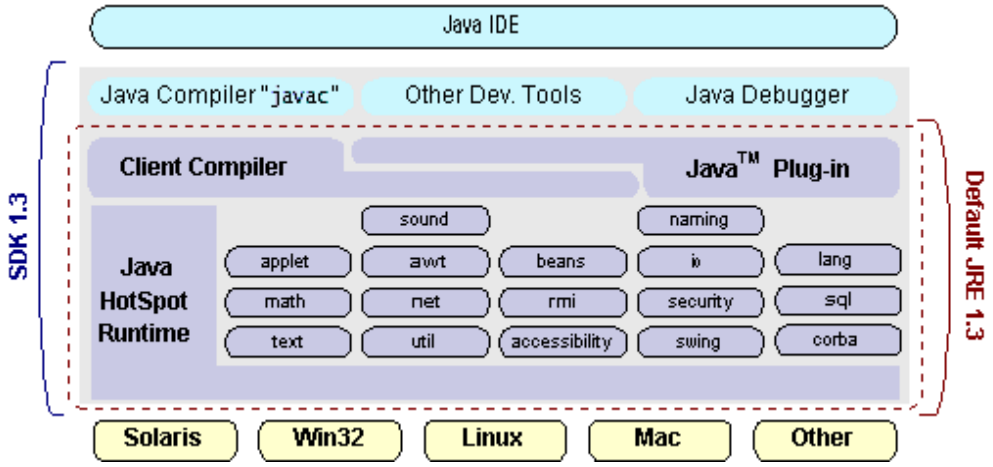


Native code is code that after you compile it, the compiled code runs on a specific hardware platform. As a platform-independent environment, the Java platform can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time byte code compilers can bring performance close to that of native code without threatening portability.

Every full implementation of the Java platform gives you the following features:

- **The essentials**: Objects, strings, threads, numbers, input and output, data structures, system properties, date and time, and so on.
- **Applets**: The set of conventions used by applets.
- **Networking**: URLs, TCP (Transmission Control Protocol), UDP (User Data gram Protocol) sockets, and IP (Internet Protocol) addresses.
- **Internationalization**: Help for writing programs that can be localized for users worldwide. Programs can automatically adapt to specific locales and be displayed in the appropriate language.
- **Security**: Both low level and high level, including electronic signatures, public and private key management, access control, and certificates.
- **Software components**: Known as JavaBeans™, can plug into existing component architectures.
- **Object serialization**: Allows lightweight persistence and communication via Remote Method Invocation (RMI).
- **Java Database Connectivity (JDBC™)**: Provides uniform access to a wide range of relational databases.

The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more. The following figure depicts what is included in the Java 2 SDK.



4.3 BENEFITS OF JAVA

FEATURES OF JAVA

- Simple: Java is designed to be easy to learn and use. It has a straightforward syntax that is similar to C++, and its object-oriented nature means that code is organized in a logical, easy-to-understand way.
- Portable: Java code can be run on any device that has a Java Virtual Machine (JVM) installed. This means that Java programs are portable and can run on any device that supports Java, regardless of the underlying hardware and operating system.
- Secure: Java was designed with security in mind. It includes features such as automatic memory management and secure programming constructs that make it difficult to introduce vulnerabilities into Java code.
- Robust: Java has a number of features that make it a robust programming language. For example, it has a strong type checking system and exception handling mechanism, which help to prevent errors and ensure that code is reliable.
- Multithreaded: Java supports multithreading, which allows developers to write programs that can perform multiple tasks concurrently. This can be useful for creating programs that need to

perform multiple tasks at the same time, such as a web server handling requests from multiple clients.

- High-performance: Despite its simplicity and portability, Java is a high-performance language that is capable of running complex programs efficiently. This is because the Java Virtual Machine (JVM) is designed to optimize code and make it run as efficiently as possible.
- Object-oriented: Java is an object-oriented programming language, which means that it is based on the concept of "objects" that represent real-world entities. This makes it easier to model real-world situations and design reusable code.
- Dynamic: Java is a dynamically-typed language, which means that the type of a variable is not determined at compile time, but rather at runtime. This makes it easier to write code that can adapt to changing conditions.
- Interpreted: Java code is compiled into an intermediate form called bytecode, which is then interpreted by the Java Virtual Machine (JVM). This makes it easier to run Java programs on different platforms, since the JVM is available on a wide range of devices.
- Network-centric: Java is designed to be used in networked environments, and it includes a number of features that make it well-suited for building networked applications. For example, it has built-in support for common network protocols like HTTP and TCP/IP, and it includes libraries for working with URLs and sockets.
- Extensible: Java can be extended through the use of libraries and frameworks. There is a wide range of third-party libraries available for Java, which can be used to add additional functionality to Java programs. Additionally, Java's modular design makes it easy to build and use frameworks that can be used to structure larger, more complex programs.
- Multilingual: Java is available in a number of languages, including English, Spanish, French, and Chinese. This makes it easier for developers around the world to use Java, regardless of their native language.
- Large developer community: Java has a large and active developer community, which means that there is a wealth of resources available for learning Java, as well as a large pool of developers who can provide support and help with issues.
- Widely used: Java is one of the most popular programming languages in the world, and it is used to build a wide variety of applications, including web, mobile, desktop, and backend applications. This means that there are many job opportunities available for Java developers.

- Strong documentation and support: Java has extensive documentation and support available, both from Oracle (the company that maintains Java) and from the larger Java developer community. This makes it easier for developers to get help with issues and learn how to use Java effectively.
- Supports functional programming: Java 8 and later versions introduced support for functional programming, through the inclusion of lambda expressions and other functional programming constructs. This makes it easier to write concise, expressive code using Java.
- Has a large standard library: Java has a large standard library that includes a wide range of utility classes, collection classes, and other useful tools. This makes it easier for developers to write programs without having to write everything from scratch.
- Supports annotation-based programming: Java supports the use of annotations, which are special types of metadata that can be added to Java code. Annotations can be used to provide additional information about code, such as how it should be used or how it should be processed.

CHAPTER-05

SYSTEM ANALYSIS

5.1 SYSTEM STUDY

5.1.1 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

5.2 EXISTING SYSTEM

Blockchain has made an impact on today's technology by revolutionizing the financial industry through utilization of cryptocurrencies using decentralized control. This has been followed by extending Blockchain to span several other industries and applications for its capabilities in verification. With the current trend of pursuing the decentralized Internet, many methods have been proposed to achieve decentralization considering different aspects of the current Internet model ranging from infrastructure and protocols to services and applications.

An existing system investigates Blockchain's capacities to provide a robust and secure decentralized model for Internet. The paper conducts a critical review on recent Blockchain-based methods capable for the decentralization of the future Internet. We identify and investigate two research aspects of Blockchain that provides high impact in realizing the decentralized Internet with respect to current Internet and Blockchain challenges while keeping various design in considerations. The first aspect is the consensus algorithms that are vital components for decentralization of the Blockchain.

The system identifies three key consensus algorithms including PoP, Paxos, and PoAH that are more adequate for reaching consensus for such tremendous scale Blockchain-enabled architecture for Internet. The second aspect that we investigated is the compliance of Blockchain with various emerging Internet technologies and the impact of Blockchain on those technologies. Such emerging Internet technologies in combinations with Blockchain would help to overcome Blockchain's established flaws in a way to be more optimized, efficient and applicable for Internet decentralization.

5.2.1 DISADVANTAGES OF EXISTING SYSTEM

- The system is less secured since all low-security requirement big volume data are stored in IPFS. When a file is stored in IPFS, the related address named CID is returned to block chain in order to get the file in IPFS when it is needed.
- The workflow of publishing comments is similar to publishing tweets. But users need to indicate the hash ID of the tweet that they want to comment on when they send a transaction for commenting.

5.3 PROPOSED SYSTEM

In the proposed system, the system proposed an autonomous decentralized online social network architecture based on block chain technology. Block chain is able to provide a safe and trusted peer-to-peer mechanism where participants have unique identities and private keys. The private key has the highest control right of the corresponding account and is stored in user's own device.

Moreover, all transactions in block chain need to be signed by the private key, so cheating can be avoided. In order to give the system abilities of self-management and sustainable development, a decentralized autonomous mechanism powered by block chain is embedded in the architecture.

The rest of the system is organized as follows. Firstly, we introduce the background of related technologies used in this architecture. Secondly, a detail description of the architecture is discussed. Thirdly, functions of this project are showed. Finally, a conclusion is made.

5.3.1 ADVANTAGES OF PROPOSED SYSTEM

- The system proposes Decentralized Autonomous Organization (DAO) which is more secure and efficient.
- The system designs a most DAOs operate rely on block chain and smart contracts, the code about a DAO plays like law, every operation must be expressed as a set of executable code, run on the block chain platform, to ensure the operation can be executed without any hindrance.

CHAPTER-06

SYSTEM DESIGN

6.1 INTRODUCTION

A blockchain-based online social networking platform is a type of social media platform that uses blockchain technology to store and manage user data and interactions. Blockchain technology is a decentralized, distributed ledger system that allows for secure and transparent record-keeping of transactions and data.The use of blockchain technology in online social networking offers several potential benefits, including enhanced security and privacy, improved transparency and accountability, and the ability to create decentralized, community-driven platforms.

However, designing a blockchain-based online social networking platform also presents a number of challenges, including the need to balance the benefits of decentralization with the scalability and performance requirements of a social networking platform, as well as the need to ensure that the platform is user-friendly and easy to use.

Overall, the introduction of a blockchain-based online social networking platform represents an exciting opportunity to explore new models for online social interaction and to address some of the challenges faced by traditional social media platforms.

6.2 MODULES

HOME PAGE

Introduction to blockchain: A module that provides an overview of what blockchain technology is and how it works. This could include information about the history of blockchain, the types of blockchain networks that exist, and the key concepts and technologies involved.

Use cases for blockchain: A module that showcases examples of how blockchain technology is being used in various industries and applications. This could include information about how blockchain is being used to facilitate financial transactions, supply chain management, voting systems, and more.

Community forum: A module that provides a platform for members of the blockchain community to connect and discuss topics of interest. This could be a discussion forum, chat room, or social media group.

ADMIN

An administrative module in a software application: This could be a part of a program that is used for managing and configuring the application, rather than for performing the primary functions of the program. For example, a content management system (CMS) might have an admin module that allows the user to manage and publish website content, set up user accounts, and configure various settings.

An administrative module in a course: This could refer to a part of an educational program that covers topics related to administration, such as financial management, personnel management, or policy development.

An administrative module in a computer system: This could be a hardware component or software program that is responsible for managing and controlling access to the system's resources, such as memory, processors, and storage.

REGISTER

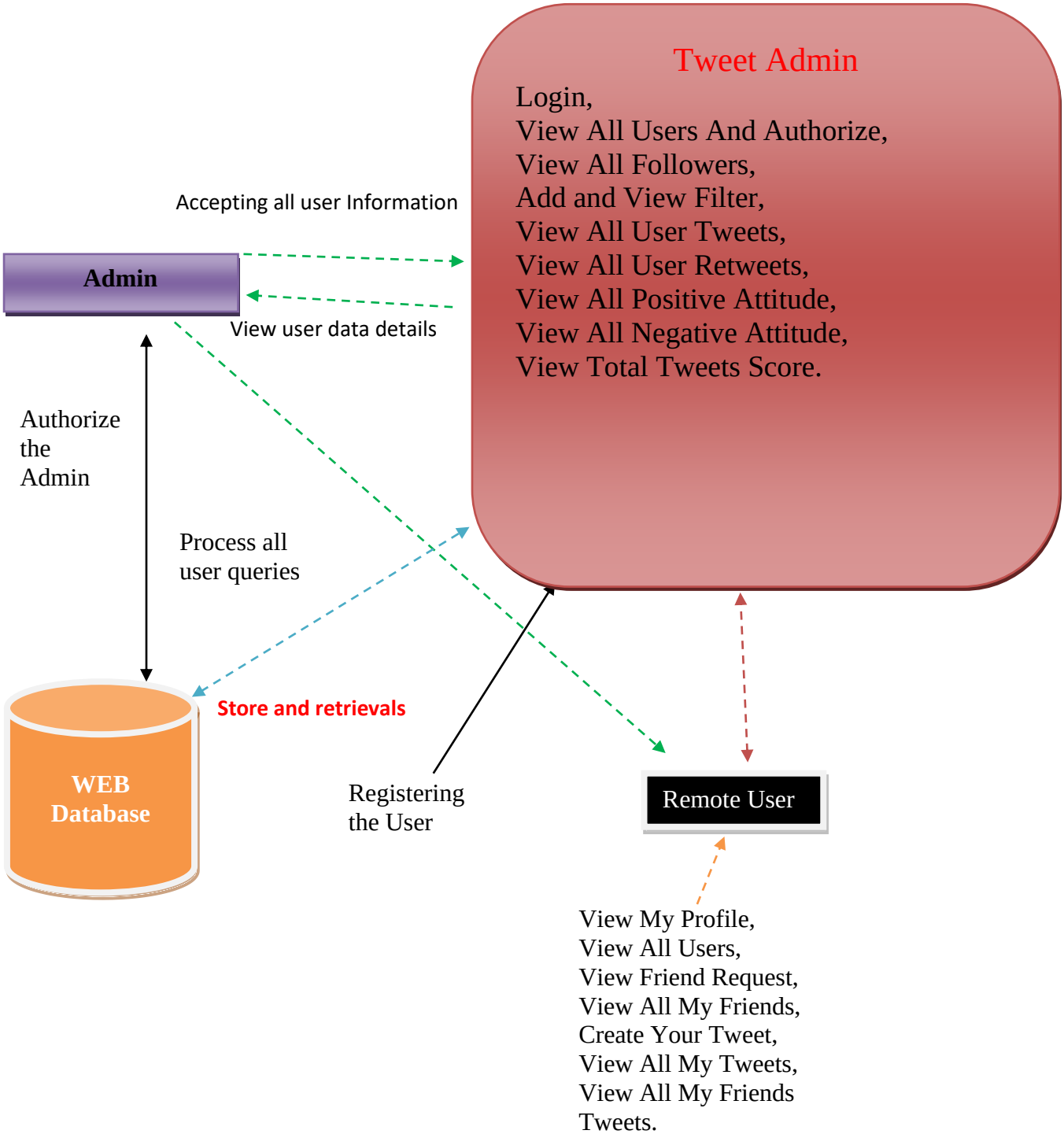
Personal information: This could include the user's name, email address, and date of birth. Some BASON platforms might also ask for other personal information, such as a phone number or physical address.

USER

Login credentials: Users will typically be required to create a username and password for their account. Some BASON platforms might also require users to set up additional security measures, such as two-factor authentication or a security question.

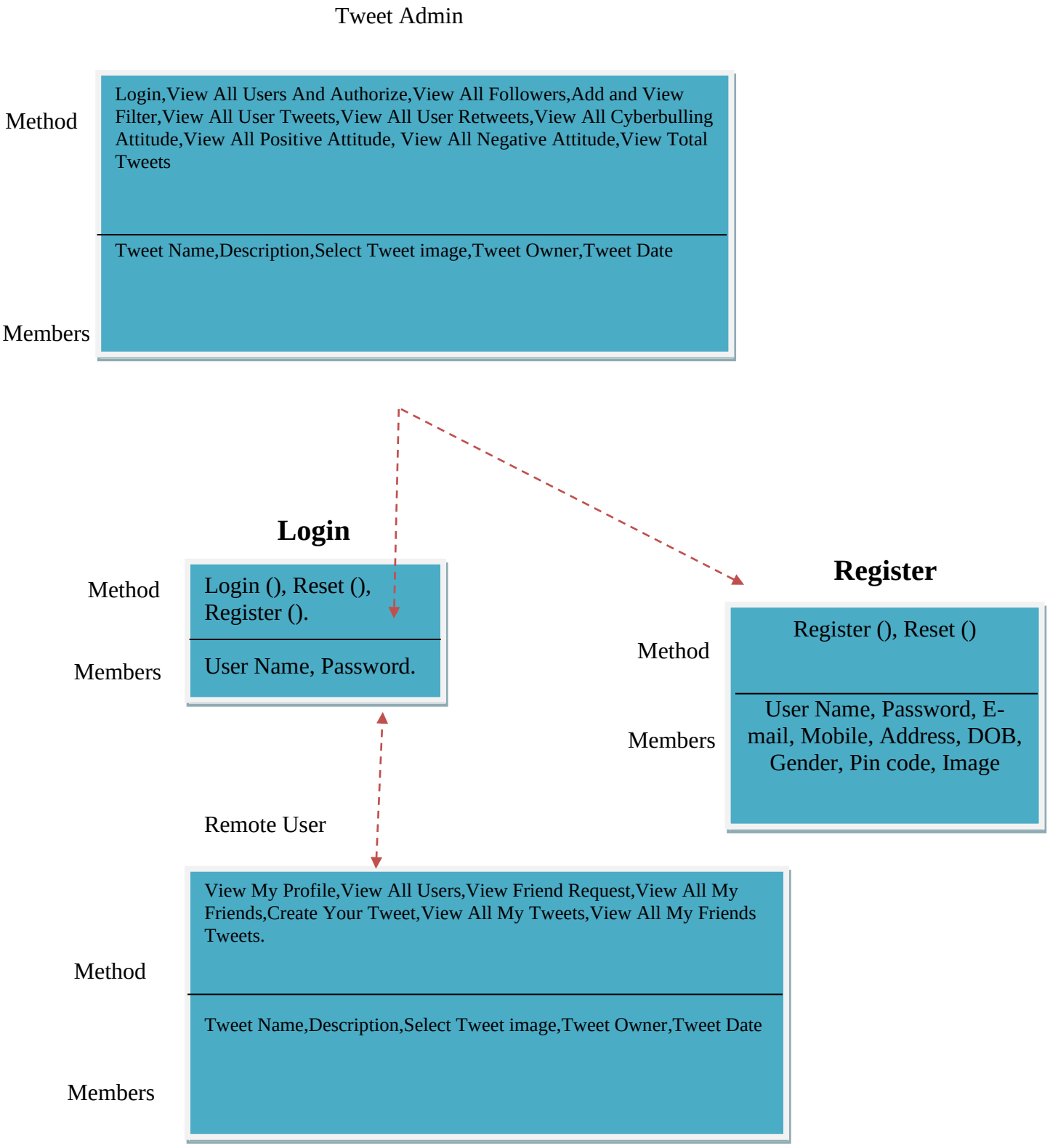
6.3 SYSTEM ARCHITECTURE

ARCHITECTURE DIAGRAM

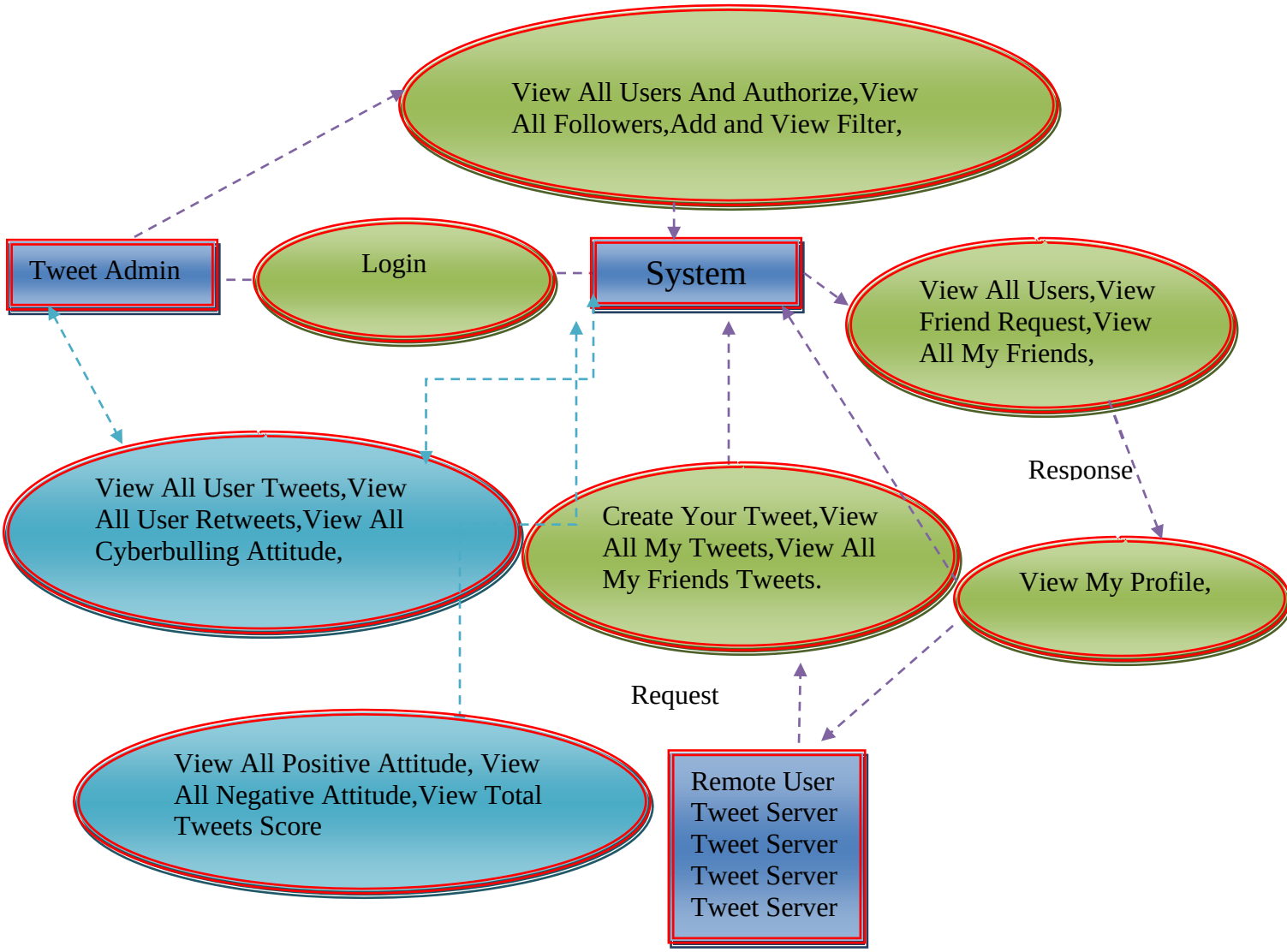


6.4 UML Diagram

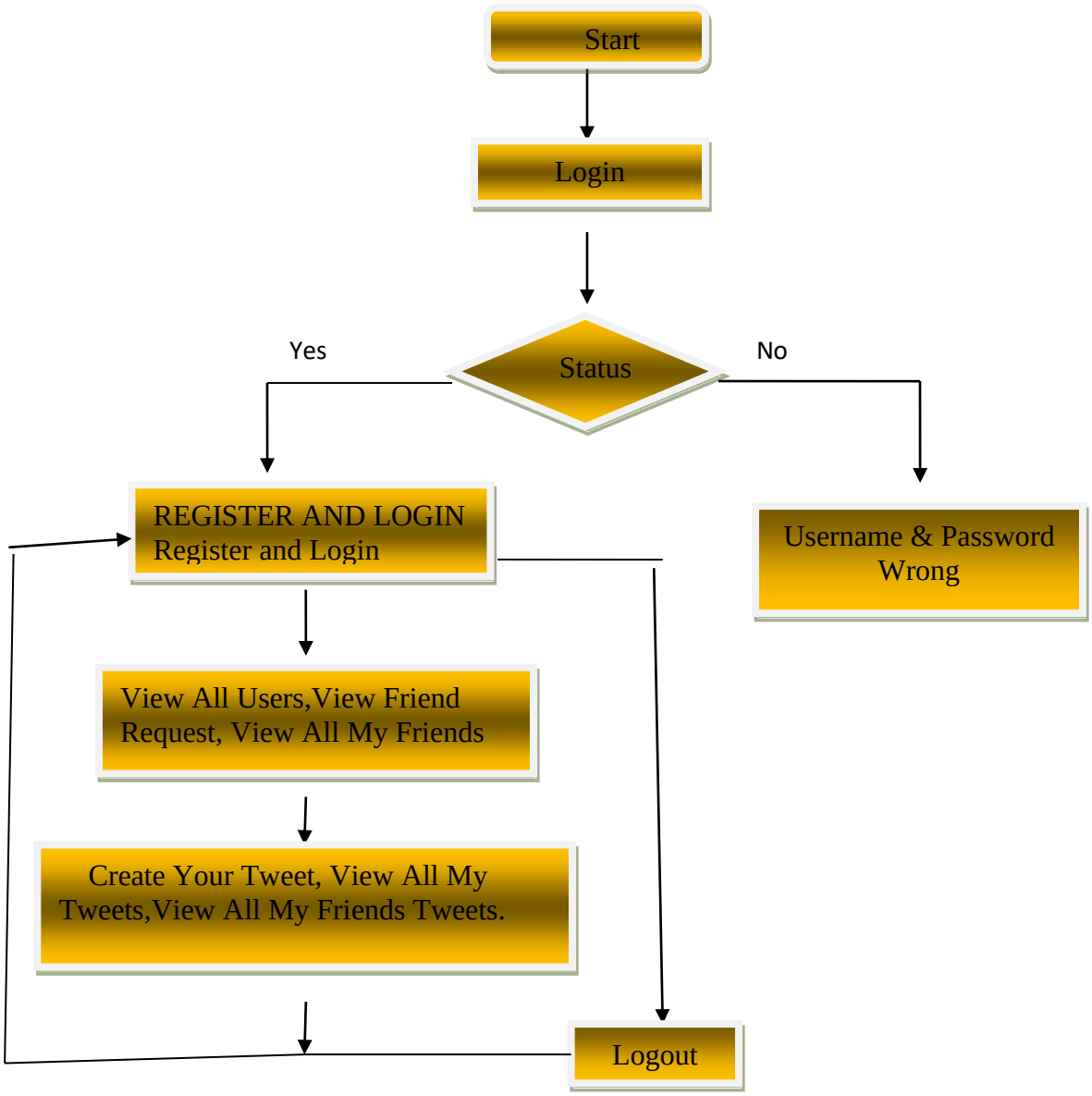
6.4.1 CLASS DIAGRAM:



6.4.2 DATA FLOW DIAGRAM:



6.4.3 FLOW CHART: REMOTE USER



CHAPTER-07

SYSTEM IMPLEMENTATION

INPUT AND OUTPUT REPRESENTATION

7.1 SYSTEM IMPLEMENTATION

Choosing a blockchain platform: The ADOSN will need to be built on top of a blockchain platform, such as Ethereum or Hyperledger Fabric. The choice of platform will depend on the specific requirements of the ADOSN, including its scalability, security, and functionality needs.

Designing the network architecture: The ADOSN will need to be designed with a decentralized network architecture, with nodes distributed across the network to validate and store data. The network will also need to have a mechanism in place for reaching consensus on the state of the network, such as proof-of-work or proof-of-stake.

- **Implementing data storage:** The ADOSN will need a way to store and manage data in a decentralized manner, such as using a distributed database or file system.
- **Implementing identity and authentication:** The ADOSN will need a system in place for verifying the identity of users and ensuring that only authorized users can access the network. This may involve the use of digital identities and cryptographic keys.
- **Implementing privacy and security measures:** The ADOSN will need to prioritize the privacy and security of its users, with measures in place to protect against cyber threats and unauthorized access to data. This may involve the use of encryption and other security measures.
- **Implementing incentives:** If the ADOSN has an incentive structure in place to encourage participation and contribution within the network, this will need to be implemented as part of the system. This may involve the use of cryptocurrency or other blockchain-based incentives.
- **Implementing governance mechanisms:** The ADOSN will need a system in place for making decisions and resolving disputes within the network, such as through the use of smart contracts or other decentralized governance mechanisms.

7.1.1 LOGICAL DESIGN

A logical design for a blockchain-based ADOSN might include the following elements:

- **Decentralized structure:** The ADOSN should be designed to operate without a central authority or intermediary, relying instead on a decentralized network of nodes to validate and store data.
- **Consensus mechanism:** The ADOSN should have a mechanism in place for reaching consensus on the state of the network, such as proof-of-work or proof-of-stake.
- **Data storage:** The ADOSN should have a way to store and manage data in a decentralized manner, such as using a distributed database or file system.
- **Identity and authentication:** The ADOSN should have a system in place for verifying the identity of users and ensuring that only authorized users can access the network.
- **Privacy and security:** The ADOSN should prioritize the privacy and security of its users, with measures in place to protect against cyber threats and unauthorized access to data.
- **Incentives:** The ADOSN may have an incentive structure in place to encourage participation and contribution within the network, such as through the use of cryptocurrency or other blockchain-based incentives.
- **Governance:** The ADOSN should have a system in place for making decisions and resolving disputes within the network, such as through the use of smart contracts or other decentralized governance mechanisms.

7.1.2 PHYSICAL DESIGN

A Physical design for a blockchain-based ADOSN might include the following elements:

- **Hardware:** The ADOSN will need a network of computers to run the blockchain software and store data. These computers, known as "nodes," can be run on a variety of hardware, including servers, personal computers, or even smartphones. The choice of hardware will depend on factors such as the scalability and performance requirements of the ADOSN.
- **Software:** The ADOSN will need to run software that implements the blockchain protocol and enables decentralized decision-making and data storage. This may include the blockchain platform itself, as well as any additional software required to support the specific functionality of the ADOSN.
- **Network architecture:** The ADOSN will need to be designed with a decentralized network architecture, with nodes distributed across the network to validate and store data. The network will also need to have a mechanism in place for reaching consensus on the state of the network, such as proof-of-work or proof-of-stake.

- **Data storage:** The ADOSN will need to have a way to store and manage data in a decentralized manner, such as using a distributed database or file system.
- **Identity and authentication:** The ADOSN will need a system in place for verifying the identity of users and ensuring that only authorized users can access the network. This may involve the use of digital identities and cryptographic keys.
- **Privacy and security:** The ADOSN will need to prioritize the privacy and security of its users, with measures in place to protect against cyber threats and unauthorized access to data. This may involve the use of encryption and other security measures.
- **Incentives:** If the ADOSN has an incentive structure in place to encourage participation and contribution within the network, this will need to be implemented as part of the system. This may involve the use of cryptocurrency or other blockchain-based incentives.
- **Governance mechanisms:** The ADOSN will need a system in place for making decisions and resolving disputes within the network, such as through the use of smart contracts or other decentralized governance mechanisms.

7.2 INPUT AND OUTPUT REPRESENTATION

7.2.1 INPUT DESIGN

Input Design plays a vital role in the life cycle of software development, it requires very careful attention of developers. The input design is to feed data to the application as accurate as possible. So, inputs are supposed to be designed effectively so that the errors occurring while feeding are minimized. According to Software Engineering Concepts, the input forms or screens are designed to provide to have a validation control over the input limit, range and other related validations.

This system has input screens in almost all the modules. Error messages are developed to alert the user whenever he commits some mistakes and guides him in the right way so that invalid entries are not made. Let us see deeply about this under module design.

Input design is the process of converting the user created input into a computer-based format. The goal of the input design is to make the data entry logical and free from errors. The error is in the input are controlled by the input design. The application has been developed in user-friendly manner. The forms have been designed in such a way during the processing the cursor is placed in the position

where must be entered. The user is also provided with in an option to select an appropriate input from various alternatives related to the field in certain cases.

Validations are required for each data entered. Whenever a user enters an erroneous data, error message is displayed and the user can move on to the subsequent pages after completing all the entries in the current page.

7.2.2 OUTPUT DESIGN

The Output from the computer is required to mainly create an efficient method of communication within the company primarily among the project leader and his team members, in other words, the administrator and the clients. The output of VPN is the system which allows the project leader to manage his clients in terms of creating new clients and assigning new projects to them, maintaining a record of the project validity and providing folder level access to each client on the user side depending on the projects allotted to him. After completion of a project, a new project may be assigned to the client. User authentication procedures are maintained at the initial stages itself. A new user may be created by the administrator himself or a user can himself register as a new user but the task of assigning projects and validating a new user rests with the administrator only.

The application starts running when it is executed for the first time. The server has to be started and then the internet explorer in used as the browser. The project will run on the local area network so the server machine will serve as the administrator while the other connected systems can act as the clients. The developed system is highly user friendly and can be easily understood by anyone using it even for the first time.

CHAPTER-08

SYSTEM TESTING

8.1 SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

8.2 TYPES OF TESTS

UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

FUNCTIONAL TEST

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centred on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

SYSTEM TEST

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

WHITE BOX TESTING

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

BLACK BOX TESTING

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a

black box. you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

8.2.1 UNIT TESTING:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

8.2.2 INTEGRATION TESTING

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

8.2.3 ACCEPTANCE TESTING

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER-09

SOURCE CODE

9.1 A_MAIN.JSP

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Admin Main</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-aller.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>

<style type="text/css">
<!--
.style1 {font-size: 30px}
.style2 {font-family: "Times New Roman", Times, serif}
.style4 {font-size: 25px}
.style7 {color: #FFFFFF}
.style8 {color: #00FFFF}
.style9 {font-size: 25px; color: #FFFFFF; }
.style10 {font-size: 25px; color: #FFFFFF; font-weight: bold; }
-->
</style>
</head>
<body>
<div class="main">
  <div class="header">
    <div class="header_resize">
      <div class="logo">
        <h1><a href="#" class="style1">A Blockchain based Autonomous Decentralized
Online Social Network</a></h1>
      </div>
      <div class="menu_nav">
        <ul>
          <li><a href="index.html"><span>Home Page</span></a></li>
          <li class="active"><a href="a_login.jsp"><span>Admin</span></a></li>
          <li><a href="u_login.jsp"><span>User</span></a></li>
        </ul>
```

```
      </div>
      <div class="clr"></div>
      <div class="slider">
        <div id="coin-slider"> <a href="#"><img src="images/slide1.jpg" width="935"
height="307" alt="" /> </a> <a href="#"><img src="images/slide2.jpg" width="935"
height="307" alt="" /> </a> <a href="#"><img src="images/slide3.jpg" width="935"
height="307" alt="" /> </a> </div>
        <div class="clr"></div>
      </div>
      <div class="clr"></div>
    </div>
  </div>
  <div class="content">
    <div class="content_resize">
      <div class="mainbar">
      <p align="center" class="style10">Welcome to Admin Main</p>
      <p align="center" class="style10"> </p>
      <p align="center" class="style10"><img src="images/Admin.jpg" width="485"
height="364" /></p>
      <p align="center" class="style10"> </p>
      </div>
      <div class="sidebar">
        <div class="gadget">
          <h2 class="star"><span>Admin</span> Menu</h2>
          <div class="clr"></div>
          <ul class="sb_menu">
            <li><a href="a_all_users.jsp">View All Users and Authorize</a></li>
            <li><a href="a_friend_req.jsp">View Friend Request and Response</a></li>
      <li><a href="a_user_on_com.jsp">View All Users Details Based on
Blockchain</a></li>
      <li><a href="a_all_posts.jsp">View All tweets Based on Blockchain</a></li>
      <li><a href="a_posts_recom.jsp">View Tweets Recommendations</a></li>

      <li><a href="a_chart2.jsp">View All tweets Rank in Chart</a></li>
      <li><a href="a_login.jsp">Log Out</a></li>
          </ul>
        </div>
      </div>
      <div class="clr"></div>
    </div>
  </div>
  <div class="fbg"></div>
  <div class="footer">
    <div class="footer_resize">
      <div style="clear:both;"></div>
```

```
    </div>
  </div>
</div>
<div align=center></div>
</body>
</html>
```

9.2 U_REGISTER.JSP

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>User Register</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-aller.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<script language="javascript" type="text/javascript">
function valid()
{


var na3=document.s.userid.value;
if(na3=="")

{
alert("Please Enter User Name");
document.s.userid.focus();
return false;
}
else
{

}
var na4=document.s.pass.value;
if(na4=="")

{
alert("Please Enter Password");
document.s.pass.focus();
```

```
return false;
}

var na29=document.s.com.value;
if(na29=="-Select-")

{
alert("Please Select Your Community");
document.s.com.focus();
return false;
}

var na6=document.s.email.value;
if(na6=="")

{
alert("Please Enter the Email");
document.s.email.focus();
return false;
}

if (na6.indexOf("@", 0) < 0)
{
alert("Please enter a valid e-mail address.");
document.s.T3.focus();
return false;
}

if (na6.indexOf(".", 0) < 0)
{
alert("Please enter a valid e-mail address.");
document.s.T3.focus();
return false;
}


var na7=document.s.mobile.value;
if(na7.length!=10)

{
alert("Please Enter Valid Mobile number or Enter 10 Digit number");
document.s.mobile.focus();
return false;
}
```

```
if(na7=="")

{
alert("Please Enter  Mobile number");
document.s.mobile.focus();
return false;
}

var na55=document.s.address.value;
if(na55=="")

{
alert("Please Enter Your Address");
document.s.address.focus();
return false;
}

var dob=document.s.dob.value;
if(dob=="")

{
alert("please Enter Your Date Of Birth");
document.s.dob.focus();
return false;
}

var na9=document.s.gender.value;
if(na9=="-Select-")

{
alert("Please Select Your Gender");
document.s.gender.focus();
return false;
}

var na11=document.s.pic.value;
if(na11=="")

{
alert("please Select Picture");
document.s.pic.focus();
return false;
}
```

```
var naloc=document.s.loc.value;
if(naloc=="")

{
alert("Please Enter Your location");
document.s.loc.focus();
return false;
}

var na25=document.s.lat.value;
if(na25=="")

{
alert("Please Enter Latitude");
document.s.lat.focus();
return false;
}
var na35=document.s.lon.value;
if(na35=="")

{
alert("Please Enter Longitude");
document.s.lon.focus();
return false;
}

}
</script>

<script
src="https://maps.googleapis.com/maps/api/js?v=3.exp&sensor=false&key=AIzaSyD0X4v7eqMFc
WCR-VZAJwEMfb47id9IZao"></script>
<script>
            var map;

            function initialize()
      {
                var mapOptions = {
                    zoom: 12,
                    center: new google.maps.LatLng(12.9716, 77.5946),
                    mapTypeId: google.maps.MapTypeId.ROADMAP
```

```
                                };
                map = new google.maps.Map(document.getElementById('map_canvas'),
                    mapOptions);

                google.maps.event.addListener(map, 'click', function(event)
        {
                    document.getElementById('latMap').value = event.latLng.lat();
                    document.getElementById('lngMap').value = event.latLng.lng();
                });
            }
            function mapDivClicked (event)
      {
                var target = document.getElementById('map_canvas'),
                    posx = event.pageX - target.offsetLeft,
                    posy = event.pageY - target.offsetTop,
                    bounds = map.getBounds(),
                    neLatlng = bounds.getNorthEast(),
                    swLatlng = bounds.getSouthWest(),
                    startLat = neLatlng.lat(),
                    endLng = neLatlng.lng(),
                    endLat = swLatlng.lat(),
                    startLng = swLatlng.lng();

                document.getElementById('posX').value = posx;
                document.getElementById('posY').value = posy;
                document.getElementById('lat').value = startLat + ((posy/350) * (endLat
- startLat));
                document.getElementById('lng').value = startLng + ((posx/500) * (endLng
- startLng));
            }
            google.maps.event.addDomListener(window, 'load', initialize);
</script>

<style type="text/css">
<!--
.style1 {font-size: 30px}
.style2 {
  font-family: "Times New Roman", Times, serif;
  font-size: 20px;
}
.style7 {color: #FFFFFF}
.style9 {
  font-size: 25px;
  color: #CCCCCC;
  font-family: "Times New Roman", Times, serif;
```

```
}
.style10 {font-size: 25px; color: #FFFFFF; font-weight: bold; }
.style11 {
  font-family: "Times New Roman", Times, serif;
  color: #CCCCCC;
}
.style12 {color: #CCCCCC}
.style14 {font-size: 20px}
.style15 {font-family: "Times New Roman", Times, serif}
-->
</style>
</head>
<body>
<div class="main">
  <div class="header">
    <div class="header_resize">
      <div class="logo">
        <h1><a href="#" class="style1">A Blockchain based Autonomous Decentralized
Online Social Network</a></h1>
      </div>
      <div class="menu_nav">
        <ul>
          <li><a href="index.html"><span>Home Page</span></a></li>
          <li><a href="a_login.jsp"><span>Admin</span></a></li>
          <li class="active"><a href="u_login.jsp"><span>User</span></a></li>
        </ul>
      </div>
      <div class="clr"></div>
      <div class="slider">
        <div id="coin-slider"> <a href="#"><img src="images/slide1.jpg" width="935"
height="307" alt="" /> </a> <a href="#"><img src="images/slide2.jpg" width="935"
height="307" alt="" /> </a> <a href="#"><img src="images/slide3.jpg" width="935"
height="307" alt="" /> </a> </div>
        <div class="clr"></div>
      </div>
      <div class="clr"></div>
    </div>
  </div>
  <div class="content">
    <div class="content_resize">
      <div class="mainbar">
      <p align="center" class="style10">User Registration </p>
<form name="s" action="u_RegIns.jsp" method="post" enctype="multipart/form-data"
onSubmit="return valid()"  ons target="_top">
```

```
            <span class="style12 style7 style9">
            <label for="name"> <br />
            </label>
            </span><span class="style7 style9">
            <label for="name"></label>
            </span><span class="style9">
            <label for="name">User Name (required)<br />
            </label>
            </span>
            <p class="style14 style5 style6 style11"><input id="name" name="userid"
class="text" />
            </p>

            <span class="style9 style14 style8 style20 style5 style6">
             <label for="password">Password (required)<br />
             </label>
            </span>
            <p class="style14 style5 style6 style11"><input type="password"
id="password" name="pass" class="text" /></p>

      <span class="style8 style9 style14 style20 style5 style6">
                  <label for="gender">Select Your Community (required)<br />
                  </label>
          </span>
                  <p class="style8 style9 style20 style5 style6">
                    <select id="s1" name="com" class="text" style="width:200px">
                      <option>-Select-</option>
                      <option>Scientiest</option>
                      <option>Engineer</option>
            <option>Doctor</option>
                    </select>
                  </p>

                  <span class="style9 style14 style8 style20 style5 style6">
                  <label for="email">Email Address (required)<br />
                  </label>
                  </span>
                  <p class="style14 style5 style6 style11">
                    <input id="email" name="email" class="text" />
                  </p>

                  <span class="style9 style14 style8 style20 style5 style6">
                  <label for="mobile">Mobile Number (required)<br />
                  </label>
                  </span>
```

```
    <p class="style14 style5 style6 style11">
      <input id="mobile" name="mobile" class="text" />
    </p>

    <span class="style9 style14 style8 style20 style5 style6">
    <label for="address">Your Address(required)<br />
    </label>
    </span>
    <p class="style14 style5 style6 style11">
      <textarea name="address" cols="50" id="address"></textarea>
    </p>

    <span class="style9 style14 style8 style20 style5 style6">
    <label for="dob">Date of Birth (required)<br />
    </label>
    </span>
    <p class="style14 style5 style6 style11">
      <input id="dob" name="dob" class="text" />
    </p>



    <span class="style8 style9 style14 style20 style5 style6">
    <label for="gender">Select Gender (required)<br />
    </label>
    </span>
    <p class="style8 style9 style20 style5 style6">
      <select id="s1" name="gender" class="text" style="width:150px">
        <option>-Select-</option>
        <option>Male</option>
        <option>Female</option>
      </select>
    </p>
    <span class="style19 style9 style14 style23 style5 style6">
    <label for="pic">Select Profile Picture (required)</label>
    </span>
    <span class="style8 style9 style20 style23 style5 style6">
    <label for="pic"><br />
    </label>
    </span>

    <p class="style15">
      <input type="file" id="pic" name="pic" class="text" />
</p>
```

```
      <span class="style9 style14 style8 style20 style5 style6">
          <label for="loc"></label>
          </span><span class="style9 style14 style8 style20 style5 style6"><label
for="lat"><br />
                  </label>
          </span>
          <p class="style15">
                    <input name="submit" type="submit" value="REGISTER"> <input
type="reset" value="RESET">
                  </p>
</form>
<p> </p>
   <p align="right"><a href="u_login.jsp" class="style2">Back</a></p>

      </div>
      <div class="sidebar">
        <div class="gadget">
          <h2 class="star"><span>Sidebar</span> Menu</h2>
          <div class="clr"></div>
          <ul class="sb_menu">
            <li><a href="index.html">Home</a></li>
            <li><a href="a_login.jsp">Admin</a></li>
            <li><a href="u_login.jsp">User</a></li>
          </ul>
        </div>
      </div>
      <div class="clr"></div>
    </div>
  </div>
  <div class="fbg"></div>
  <div class="footer">
    <div class="footer_resize">
      <div style="clear:both;"></div>
    </div>
  </div>
</div>
<div align=center></div>
</body>
</html>
```

9.3 A_LOGIN.JSP

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Admin Login</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-aller.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<script language="javascript" type="text/javascript">
function valid()
{
var na3=document.s.userid.value;
if(na3=="")

{
alert("Please Enter Name");
document.s.userid.focus();
return false;
}
else
{

}
var na4=document.s.pass.value;
if(na4=="")

{
alert("Please Enter Password");
document.s.pass.focus();
return false;
}


}
</script>
<style type="text/css">
<!--
.style1 {font-size: 30px}
.style2 {font-family: "Times New Roman", Times, serif}
.style4 {font-size: 25px}
.style7 {color: #FFFFFF}
.style8 {color: #00FFFF}
```

```
.style9 {font-size: 25px; color: #FFFFFF; }
.style10 {font-size: 25px; color: #FFFFFF; font-weight: bold; }
-->
</style>
</head>
<body>
<div class="main">
  <div class="header">
    <div class="header_resize">
      <div class="logo">
        <h1><a href="#" class="style1">A Blockchain based Autonomous Decentralized
Online Social Network</a></h1>
      </div>
      <div class="menu_nav">
        <ul>
          <li><a href="index.html"><span>Home Page</span></a></li>
          <li class="active"><a href="a_login.jsp"><span>Admin</span></a></li>
          <li><a href="u_login.jsp"><span>User</span></a></li>
        </ul>
      </div>
      <div class="clr"></div>
      <div class="slider">
        <div id="coin-slider"> <a href="#"><img src="images/slide1.jpg" width="935"
height="307" alt="" /> </a> <a href="#"><img src="images/slide2.jpg" width="935"
height="307" alt="" /> </a> <a href="#"><img src="images/slide3.jpg" width="935"
height="307" alt="" /> </a> </div>
        <div class="clr"></div>
      </div>
      <div class="clr"></div>
    </div>
  </div>
  <div class="content">
    <div class="content_resize">
      <div class="mainbar">
      <p align="center" class="style10">Admin LogIn</p>
  <form name="s" action="a_authentication.jsp" method="post" onSubmit="return valid()"
 ons target="_top">

    <table align="center" border="2" width="51%" height="146">

      <tr>
        <td width="48%" height="25" bgcolor="#FF3300" class="style4 style2"><span
class="style9 style2 style4 style7"> Name </span></td>
        <td width="55%" height="25" bgcolor="#FF3300"><input type="text"  name="userid"
size="15" /></td>
```

```
      </tr>
      <tr>
        <td width="48%" height="25" bgcolor="#FF3300" class="style5 style2"> <span
class="style4 style9 style7">Password</span></td>
        <td width="55%" height="25" bgcolor="#FF3300"><input type="password"
 name="pass" size="15" /></td>
      </tr>
      <tr>
        <td height="78" colspan="2"><p align="center">
            <input type="submit"  value="Login" name="B1" />
            <input type="reset" value="Reset" name="B2" />
        </td>
      </tr>
    </table>

</form>

      </div>
      <div class="sidebar">
        <div class="gadget">
          <h2 class="star"><span>Sidebar</span> Menu</h2>
          <div class="clr"></div>
          <ul class="sb_menu">
            <li><a href="index.html">Home</a></li>
            <li><a href="a_login.jsp">Admin</a></li>
            <li><a href="u_login.jsp">User</a></li>
          </ul>
        </div>
      </div>
      <div class="clr"></div>
    </div>
  </div>
  <div class="fbg"></div>
  <div class="footer">
    <div class="footer_resize">
      <div style="clear:both;"></div>
    </div>
  </div>
</div>
<div align=center></div>
</body>
</html>
```

9.4 INDEX.HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Home Page</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-aller.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style1 {font-size: 30px}
.style2 {
  color: #FFFFFF;
  font-weight: bold;
}
.style3 {
  color: #FF0000;
  font-style: italic;
  font-weight: bold;
}
.style4 {font-size: 14px}
.style5 {font-size: 18px}
-->
</style>
</head>
<body>
<div class="main">
  <div class="header">
    <div class="header_resize">
      <div class="logo">
        <h1><a href="#" class="style1">A Blockchain based Autonomous Decentralized
Online Social Network</a></h1>
      </div>
      <div class="menu_nav">
        <ul>
          <li class="active"><a href="index.html"><span>Home Page</span></a></li>
```

```
          <li><a href="a_login.jsp"><span>Admin</span></a></li>
          <li><a href="u_login.jsp"><span>User</span></a></li>
        </ul>
      </div>
      <div class="clr"></div>
      <div class="slider">
        <div id="coin-slider"> <a href="#"><img src="images/slide1.jpg" width="935"
height="307" alt="" /> </a> <a href="#"><img src="images/slide2.jpg" width="935"
height="307" alt="" /> </a> <a href="#"><img src="images/slide3.jpg" width="935"
height="307" alt="" /> </a> </div>
        <div class="clr"></div>
      </div>
      <div class="clr"></div>
    </div>
  </div>
  <div class="content">
    <div class="content_resize">
      <div class="mainbar">
        <div class="article">
          <h2 class="style3">A Blockchain based Autonomous Decentralized Online Social
Network</h2>
          <p class="infopost">Blockchain, Online Social Network (OSN), Decentralized
Autonomous Organization (DAO)</p>
          <div class="clr"></div>
          <div class="img"><img src="images/img1.jpg" width="200" height="349" alt=""
class="fl" /></div>
          <div class="post_content">
            <p align="justify" class="style2"> </p>
            <p align="justify" class="style2"> </p>
            <p align="justify" class="style2">Online social networks (OSN) are becoming
more important in people’s daily life, however, all popular OSNs are centralized, and
this raises a series of security, privacy and management issues. A decentralized
architecture based on blockchain technology provides the ability to solve above issues.
In this paper, an OSN service is developed based on blockchain technology in order to
make it operate decentralized. Large volume of data normally required low-security
requirements can be stored in Interplanetary Filesystem (IPFS) to make data
decentralized. A decentralized autonomous organization is developed for user autonomy,
users can self-manage the OSN in a democratic way..</p>
          </div>
          <div class="clr"></div>
        </div>
      </div>
      <div class="sidebar">
        <div class="searchform">
          <form id="formsearch" name="formsearch" method="post" action="#">
```

```
            <span>
            <input name="editbox_search" class="editbox_search" id="editbox_search"
maxlength="80" value="Search our ste:" type="text" />
            </span>
            <input name="button_search" src="images/search.gif" class="button_search"
type="image" />
          </form>
        </div>
        <div class="clr"></div>
        <div class="gadget">
          <h2 class="star"><span>Sidebar</span> Menu</h2>
          <div class="clr"></div>
          <ul class="sb_menu">
            <li><a href="index.html">Home</a></li>
            <li><a href="a_login.jsp">Admin</a></li>
            <li><a href="u_login.jsp">User</a></li>
          </ul>
        </div>
        <div class="gadget">
          <h2 class="star"><span>Concepts</span></h2>
          <div class="clr"></div>
          <ul class="ex_menu">
            <li><span class="infopost style4"><span class="infopost">Blockchain, <br />
              Online Social Network (OSN), <br />
            Decentralized Autonomous Organization (DAO)</span>, </span></li>
            <li></li>
          </ul>
        </div>
      </div>
      <div class="clr"></div>
    </div>
  </div>
  <div class="fbg"></div>
  <div class="footer">
    <div class="footer_resize">
      <div style="clear:both;"></div>
    </div>
  </div>
</div>
<div align=center></div>
</body>
</html>
```

9.5 STYLES.CSS

```
/* Design by http://www.dreamtemplate.com */
```

```
@charset "utf-8";
body {
    margin:0;
    padding:0;
    width:100%;
    color:#c4c3c3;
    font:normal 12px/1.5em "Liberation sans", Arial, Helvetica, sans-serif;
    background:#081421 url(../images/main_bg.jpg) repeat left top;
}
html, .main {
    padding:0;
    margin:0;
}
.main {
}
.clr {
    clear:both;
    padding:0;
    margin:0;
    width:100%;
    font-size:0;
    line-height:0;
}
h2 {
    margin:8px 0;
    padding:8px 0;
    font-size:24px;
    font-weight:normal;
    color:#c4c3c3;
}
p {
    margin:8px 0;
    padding:0 0 8px 0;
}
a {
    color:#6494c0;
    text-decoration:underline;
}
.header, .content, .menu_nav, .fbg, .footer, form, ol, ol li, ul, .content .mainbar,
.content .sidebar {
    margin:0;
    padding:0;
}
.header {
}
```

```
.header_resize {
    margin:0 auto;
    padding:0;
    width:960px;
}
.logo {
    padding:0;
    float:none;
    float:left;
    width:auto;
    height:114px;
}
h1 {
    margin:0;
    padding:32px 0 0;
    font-size:36px;
    font-weight:bold;
    line-height:1.2em;
    text-transform:uppercase;
}
h1 a, h1 a:hover {
    color:#e2e7ed;
    text-decoration:none;
}
h1 span {
}
h1 small {
    display:block;
    font-size:13px;
    line-height:1.2em;
    letter-spacing:normal;
    text-transform:none;
    color:#fff;
}
.slider {
    margin:0;
    padding:14px 12px 14px 13px;
    background:url(../images/slider_bg.jpg) no-repeat left top;
}
.rss {
    padding:18px 20px 0 0;
    float:right;
    width:auto;
}
.rss p {
```

```
    margin:0;
    padding:0;
    float:right;
    width:auto;
    color:#fff;
    font-size:13px;
    line-height:1.5em;
    font-weight:bold;
}
.rss a {
    color:#fff;
    text-decoration:none;
}
.rss a:hover {
    text-decoration:underline;
}
.rss img {
    margin:-3px 0 -3px 6px;
    border:none;
}
.menu_nav {
    margin:0 auto;
    padding:34px 0 0;
    height:48px;
    float:right;
}
.menu_nav ul {
    list-style:none;
    padding:0;
    height:48px;
    float:left;
}
.menu_nav ul li {
    margin:0;
    padding:0;
    float:left;
    background:url(../images/white_bg.jpg) repeat left top;
}
.menu_nav ul li a {
    display:block;
    margin:0;
    padding:15px 24px;
    font-size:13px;
    line-height:18px;
    font-weight:normal;
```

```
    color:#fff;
    text-decoration:none;
    text-transform:none;
    text-align:center;
}
.menu_nav ul li.active a, .menu_nav ul li a:hover {
    text-decoration:none;
    color:#6494c0;
}
.content {
    padding:0;
}
.content_resize {
    margin:0 auto;
    padding:24px 0 40px;
    width:960px;
}
.content .mainbar {
    margin:0;
    padding:0;
    float:right;
    width:684px;
}
.content .mainbar h2 {
    margin-bottom:0;
    padding-bottom:8px;
    font-size:24px;
    line-height:1.2em;
    color:#c4c3c3;
}
.content .mainbar div.img {
    padding:12px 0 0;
    float:left;
    width:179px;
}
.content .mainbar img {
}
.content .mainbar img.fl {
    margin:0;
    border:1px solid #ccd0d5;
}
.content .mainbar .article {
    margin:0 0 16px;
    padding:6px 20px;
    background:url(../images/white_bg.jpg) repeat left top;
```

```
}
.content .mainbar .post_content {
    float:right;
    width:440px;
}
.content .mainbar .post_content strong {
    color:#c4c3c3;
}
.content .sidebar {
    padding:0;
    float:left;
    width:260px;
}
.content .sidebar .gadget {
    margin:0 0 16px;
    padding:12px 20px 24px;
    background:url(../images/white_bg.jpg) repeat left top;
}
.content .sidebar h2 {
    color:#c4c3c3;
}
ul.sb_menu, ul.ex_menu {
    margin:0;
    padding:0;
    list-style:none;
    color:#c4c3c3;
}
ul.sb_menu li, ul.ex_menu li {
    margin:0;
}
ul.sb_menu li, ul.ex_menu li {
    padding:8px 0;
    width:220px;
}
ul.sb_menu li a {
    color:#c4c3c3;
    text-decoration:none;
    margin-left:-16px;
    padding:4px 8px 4px 16px;
}
ul.ex_menu li a {
    font-weight:bold;
    color:#c4c3c3;
    text-decoration:none;
}
```

```
ul.sb_menu li a:hover, ul.ex_menu li a:hover {
    color:#6494c0;
    font-weight:bold;
    text-decoration:none;
}
p.spec {
    padding:0 0 16px;
}
p.infopost {
    margin:0 0 8px;
    padding:6px 12px;
    color:#c4c3c3;
    border-top:1px solid #10151b;
    border-bottom:1px solid #10151b;
}
p.infopost a {
    padding:0 2px;
    text-decoration:none;
    font-weight:bold;
    color:#6494c0;
}
p.infopost a:hover {
    text-decoration:underline;
}
p.infopost span.date {
    color:#949494;
}
.article a.com {
    display:block;
    float:right;
    margin:0;
    font-weight:normal;
    text-decoration:none;
    text-align:center;
    color:#c4c3c3;
}
.article a.com:hover {
    text-decoration:none;
}
.article a.com span {
    font-weight:normal;
    color:#6494c0;
    font-weight:bold;
}
.post_content a.rm {
```

```
    display:block;
    float:left;
    margin:0 0 16px;
    padding:10px 16px;
    font-size:13px;
    line-height:16px;
    font-weight:bold;
    text-transform:none;
    color:#6494c0;
    text-decoration:none;
    text-align:center;
    background:#172029;
}
.post_content a.rm:hover {
    text-decoration:underline;
}
.content p.pages {
    margin:0;
    padding:24px 20px;
    height:20px;
    font-size:11px;
    color:#6e6e6e;
    text-align:left;
    background:url(../images/white_bg.jpg) repeat left top;
}
.content p.pages span, .content p.pages a {
    display:block;
    float:left;
    margin:0 4px 0 0;
    padding:4px 0;
    width:26px;
    text-decoration:none;
    text-align:center;
}
.content p.pages span, .content p.pages a:hover {
    color:#fff;
    background:#405b74;
}
.content p.pages a {
    color:#6d6d6d;
    border:none;
}
.content p.pages small {
    font-size:12px;
    float:right;
```

```
}
.content .mainbar .comment {
    margin:0;
    padding:16px 0 0 0;
}
.content .mainbar .comment img.userpic {
    border:1px solid #dedede;
    margin:10px 16px 0 0;
    padding:0;
    float:left;
}
.fbg {
    padding:0;
}
.fbg_resize {
    margin:0 auto;
    width:920px;
    padding:16px 20px;
    color:#c4c3c3;
    background:url(../images/white_bg.jpg) repeat left top;
}
.fbg h2 {
    color:#fff;
    padding-bottom:24px;
}
.fbg img.gal {
    margin:0 8px 8px 0;
    padding:0;
    border:4px solid #ece7e7;
}
.fbg .col {
    margin:0;
    float:left;
}
.fbg .c1 {
    padding:0 16px 0 0;
    width:280px;
}
.fbg .c2 {
    padding:0 36px 0 16px;
    width:280px;
}
.fbg .c3 {
    padding:0 0 0 16px;
    width:274px;
```

```
}
.fbg .fbg_ul {
    margin:0;
    padding:0;
    list-style:none;
}
.fbg .fbg_ul li {
    margin:0;
    padding:0;
    list-style:none;
}
.fbg .fbg_ul li a {
    display:block;
    margin:0;
    padding:2px 0 2px 12px;
    color:#c4c3c3;
    text-decoration:none;
    background:url(../images/li.gif) no-repeat left center;
}
.fbg .fbg_ul li a:hover {
    color:#6494c0;
    text-decoration:underline;
}
.fbg p.contact_info {
    line-height:1.8em;
}
.fbg p.contact_info a {
    color:#6494c0;
    text-decoration:underline;
    font-weight:normal;
}
.fbg p.contact_info span {
    display:block;
    float:left;
    width:74px;
    font-weight:bold;
}
ol {
    list-style:none;
}
ol li {
    display:block;
    clear:both;
}
ol li label {
```

```
    display:block;
    margin:0;
    padding:16px 0 0 0;
}
ol li input.text {
    width:480px;
    border:1px solid #c0c0c0;
    margin:0;
    padding:5px 2px;
    height:16px;
    background-color:#fff;
}
ol li textarea {
    width:480px;
    border:1px solid #c0c0c0;
    margin:0;
    padding:2px;
    background-color:#fff;
}
ol li .send {
    margin:16px 0 0 0;
}
.searchform {
    margin:0 0 16px;
    padding:0;
    float:left;
    background:url(../images/white_bg.jpg) repeat left top;
}
#formsearch {
    margin:0;
    padding:24px 20px;
    width:auto;
    height:33px;
}
#formsearch span {
    display:block;
    margin:0;
    padding:0;
    float:left;
    background:url(../images/search_bg.gif) no-repeat left top;
}
#formsearch input.editbox_search {
    margin:0;
    padding:7px 6px 7px 10px;
    float:left;
```

```
    width:175px;
    font-size:12px;
    line-height:16px;
    color:#a8acb2;
    background:none;
    outline:none;
    border:none;
}
#formsearch input.button_search {
    margin:0;
    padding:0;
    border:none;
    float:left;
}
.footer {
}
.footer_resize {
    margin:0 auto;
    padding:0 20px;
    width:920px;
    background:url(../images/white_bg.jpg) repeat left top;
    border-top:1px solid #10151b;
    line-height:1.5em;
    color:#c4c3c3;
}
.footer_resize p {
    margin:0;
    padding:25px 0;
    line-height:normal;
    white-space:nowrap;
    text-indent:inherit;
}
.footer_resize a {
    color:#6494c0;
    font-weight:normal;
    margin:0;
    padding:0;
    border:none;
    text-decoration:underline;
    background-color:transparent;
}
.footer_resize a:hover {
    color:#6494c0;
    background-color:transparent;
    text-decoration:none;
```

```
}
.footer_resize .lf {
    float:left;
}
.footer_resize .rf {
    float:right;
}
a {
    outline:none;
}
```

9.6 SCRIPT.JS

```
// <![CDATA[
$(function() {

  // Slider
  $('#coin-slider').coinslider({width:935,height:307,opacity:1});

  // Radius Box
  //$('p.infopost').css({"border-radius":"16px", "-moz-border-radius":"16px", "-webkit-
border-radius":"16px"});
  //$('.content .sidebar .gadget, .fbg_resize').css({"border-radius":"12px", "-moz-
border-radius":"12px", "-webkit-border-radius":"12px"});
  //$('.content p.pages span, .content p.pages a').css({"border-radius":"16px", "-moz-
border-radius":"16px", "-webkit-border-radius":"16px"});
  //$('.menu_nav').css({"border-bottom-left-radius":"16px", "border-bottom-right-
radius":"16px", "-moz-border-radius-bottomleft":"16px", "-moz-border-radius-
bottomright":"16px", "-webkit-border-bottom-left-radius":"16px", "-webkit-border-
bottom-right-radius":"16px"});

});

// Cufon
Cufon.replace('h1, h2, h3, h4, h5, h6', { hover: true });
//Cufon.replace('h1', { color: '-linear-gradient(#fff, #ffaf02)'});
//Cufon.replace('h1 small', { color: '#8a98a5'});

// ]]>
```

9.7 DATABASE.SQL

```
create database bcba;

use bcba;
```

```
SET FOREIGN_KEY_CHECKS=0;
-- ----------------------------
-- Table structure for admin
-- ----------------------------
DROP TABLE IF EXISTS `admin`;
CREATE TABLE `admin` (
  `name` text NOT NULL,
  `pass` text NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for com_vel
-- ----------------------------
DROP TABLE IF EXISTS `com_vel`;
CREATE TABLE `com_vel` (
  `id` int(11) NOT NULL auto_increment,
  `community` text,
  `c_movs` int(11) default NULL,
  `t_movs` int(11) default NULL,
  `c_vel` int(11) default NULL,
  PRIMARY KEY  (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- ----------------------------
-- Table structure for friend_req
-- ----------------------------
DROP TABLE IF EXISTS `friend_req`;
CREATE TABLE `friend_req` (
  `id` int(11) NOT NULL auto_increment,
  `rfrom` text,
  `rto` text,
  `status` text,
  `req_dt` text,
  `res_dt` text,
  PRIMARY KEY  (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- ----------------------------
-- Table structure for posts
-- ----------------------------
DROP TABLE IF EXISTS `posts`;
CREATE TABLE `posts` (
  `id` int(11) NOT NULL auto_increment,
  `title` text,
```

```
  `image` blob,
  `description` text,
  `uses` text,
  `date` text,
  `rank` int(11) default NULL,
  `user` text,
  `hash_sign` text,
  PRIMARY KEY  (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- ----------------------------
-- Table structure for recommend
-- ----------------------------
DROP TABLE IF EXISTS `recommend`;
CREATE TABLE `recommend` (
  `id` int(11) NOT NULL auto_increment,
  `title` text,
  `rfrom` text,
  `rto` text,
  `dt` text,
  PRIMARY KEY  (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- ----------------------------
-- Table structure for user
-- ----------------------------
DROP TABLE IF EXISTS `user`;
CREATE TABLE `user` (
  `id` int(11) NOT NULL auto_increment,
  `name` text,
  `pass` text,
  `Community` text,
  `email` text,
  `mobile` text,
  `dob` text,
  `gender` text,
  `addr` text,
  `status` text,
  `image` longblob,
  PRIMARY KEY  (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- ----------------------------
-- Records
-- ----------------------------
```

```
INSERT INTO `admin` VALUES ('admin', 'admin');
INSERT INTO `com_vel` VALUES ('4', 'Scientiest', '6', '7', '85');
INSERT INTO `com_vel` VALUES ('6', 'Engineer', '1', '7', '14');
INSERT INTO `friend_req` VALUES ('1', 'Omkar', 'Mahesh', 'Friends', '22/08/2021
10:28:58', '22/08/2021   11:11:42');
INSERT INTO `friend_req` VALUES ('3', 'Ganesh', 'omkar', 'Friends', '22/08/2021
13:59:33', '22/08/2021   14:00:14');
INSERT INTO `friend_req` VALUES ('4', 'Manjunath', 'Omkar', 'Friends', '22/08/2021
18:26:48', '22/08/2021   18:27:00');
INSERT INTO `friend_req` VALUES ('5', 'Manjunath', 'ganesh', 'Friends', '22/08/2021
18:32:30', '22/08/2021   18:32:42');
INSERT INTO `friend_req` VALUES ('6', 'Manjunath', 'mahesh', 'Friends', '23/08/2021
17:44:52', '23/08/2021   17:45:29');
INSERT INTO `friend_req` VALUES ('7', 'Gokul', 'Manjunath', 'Friends', '23/08/2021
18:02:01', '23/08/2021   18:02:22');
```

CHAPTER-10

OUTPUT SCREENSHOTS

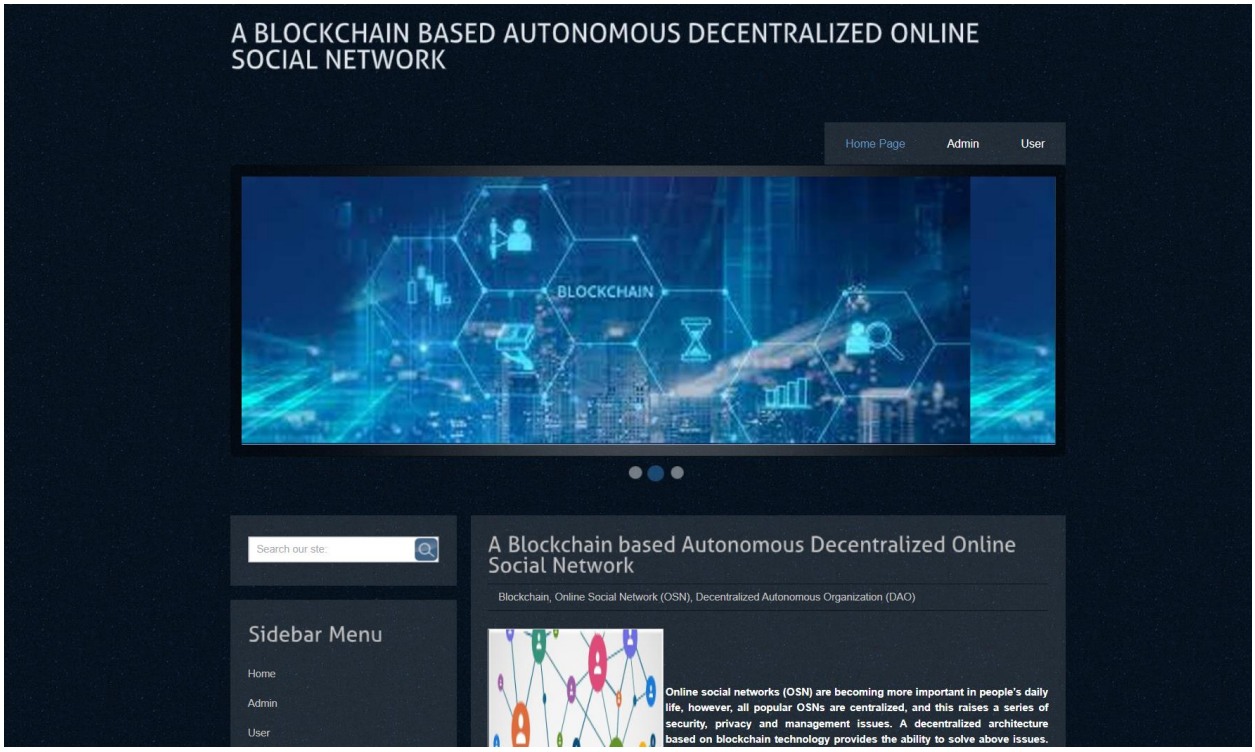


Fig 1: Home Page

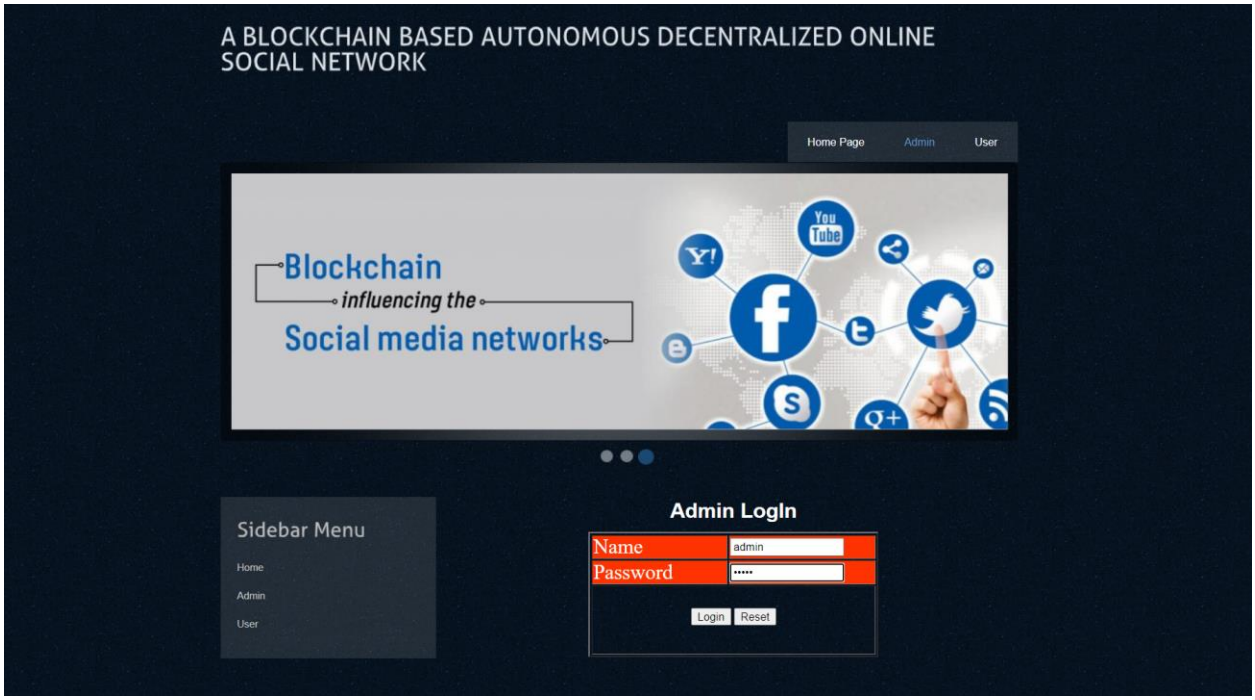


Fig 2: Admin Login

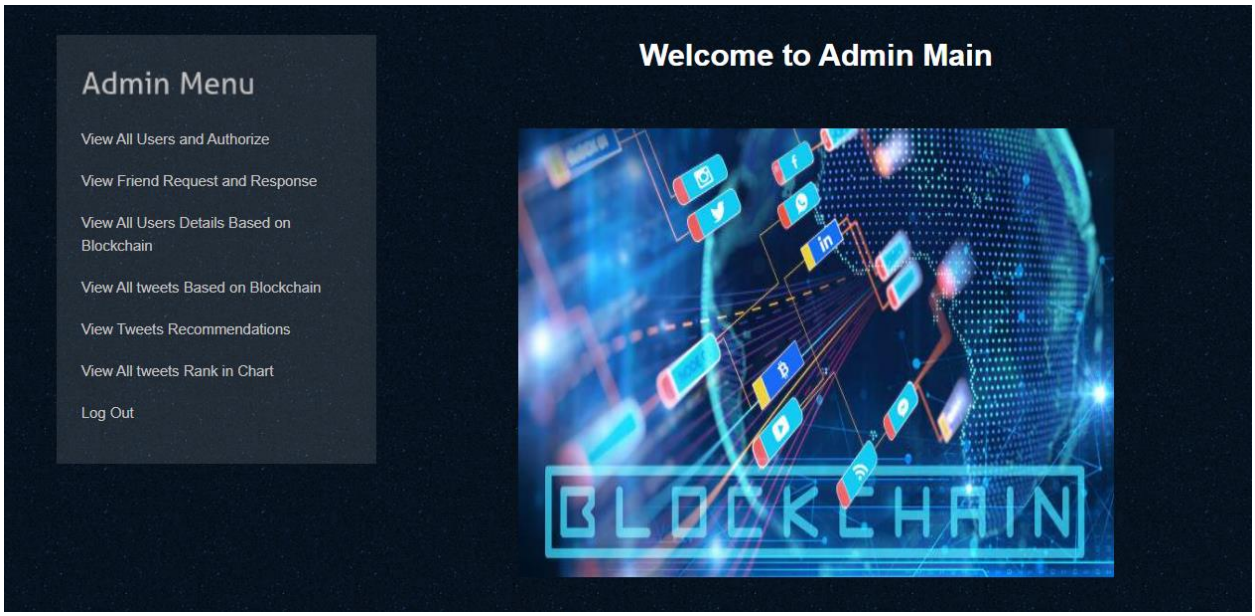


Fig 3: Admin Menu

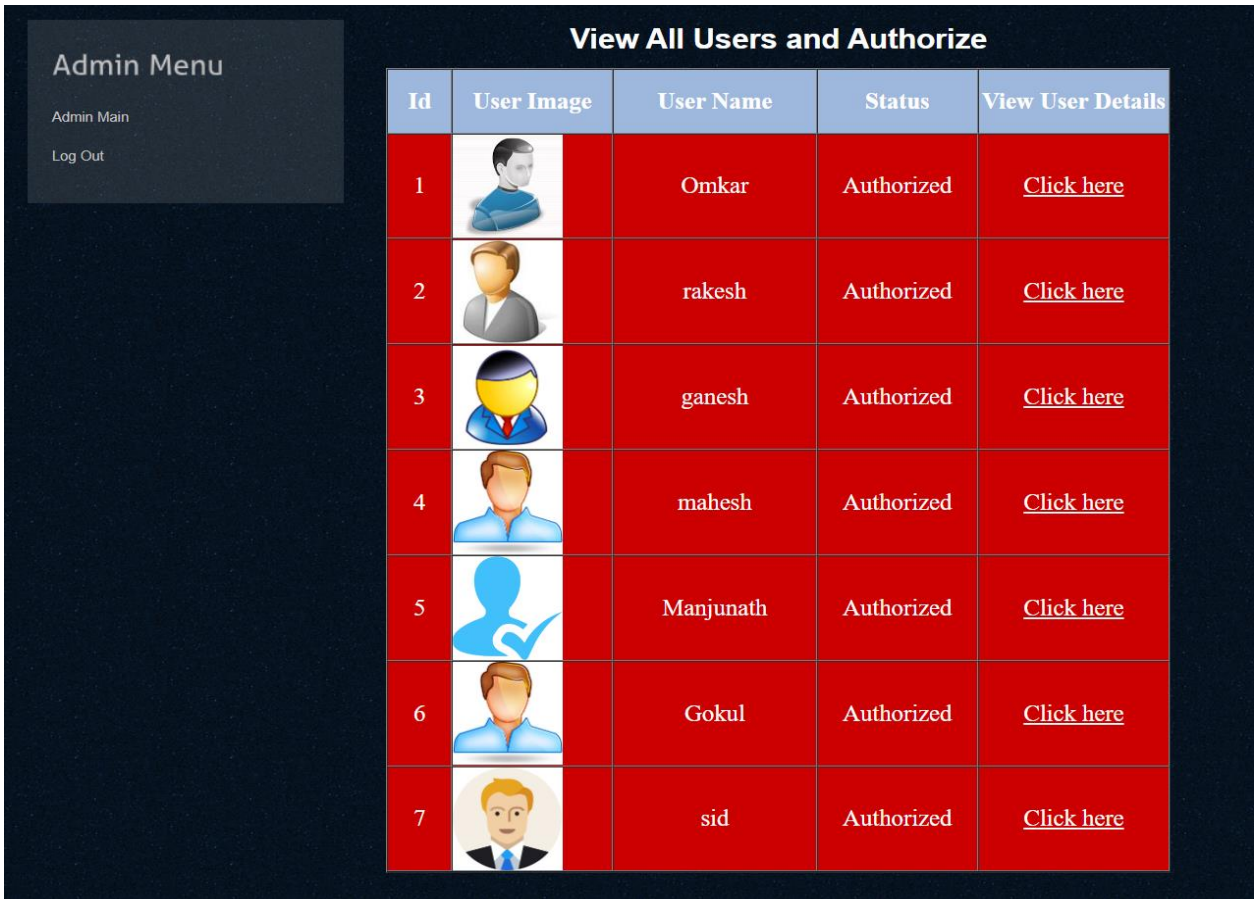


Fig 4: View All Users and Authorize

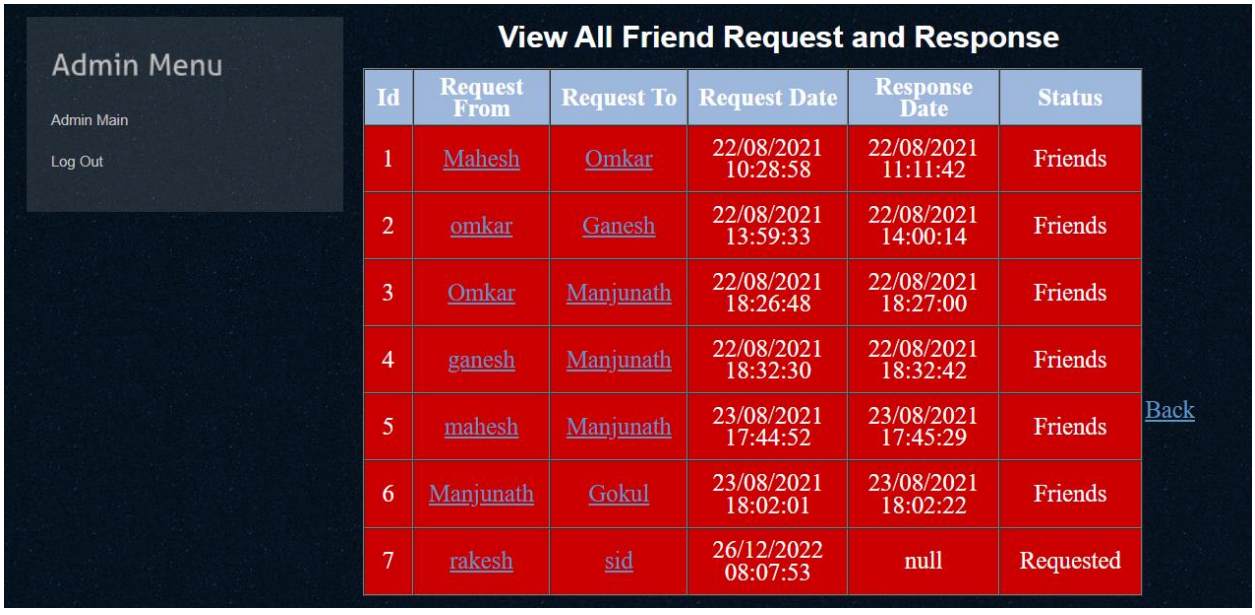


Fig 5: View Users Details Based on Community Blockchain

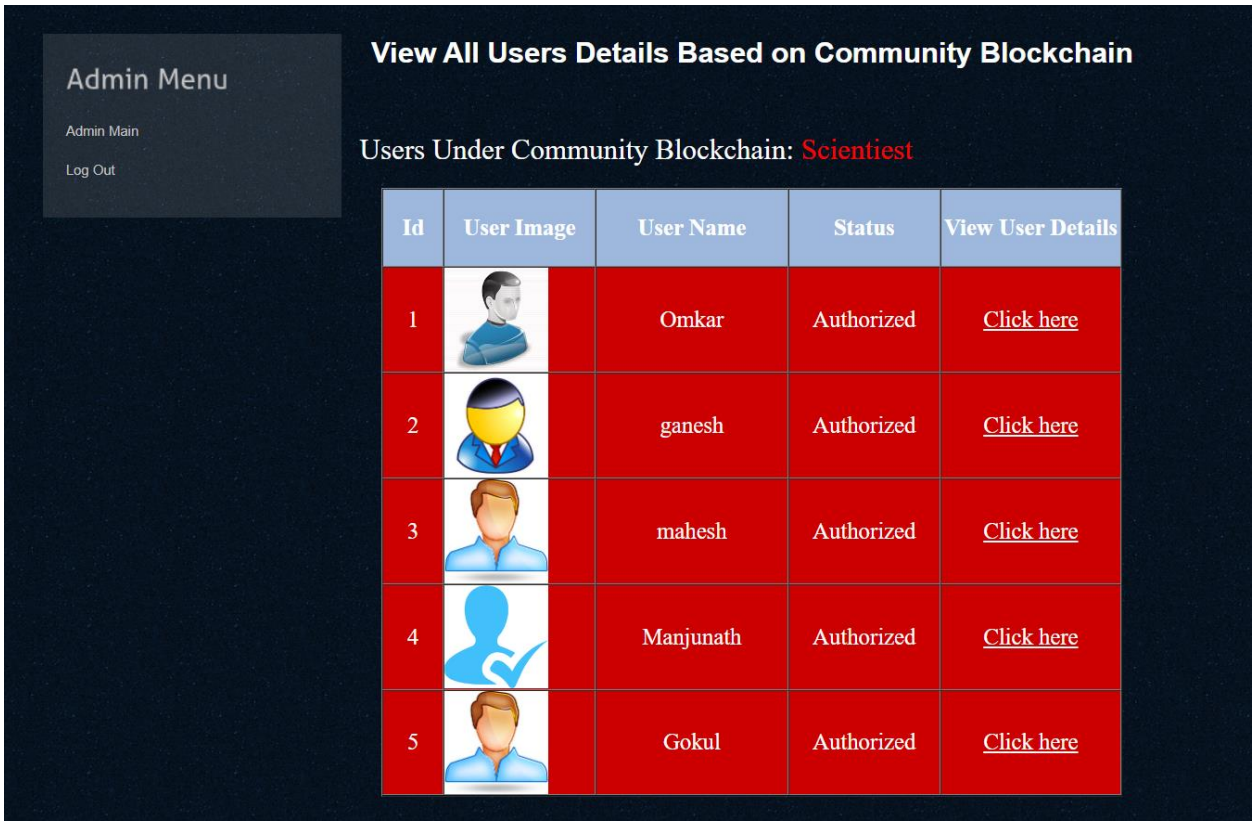


Fig 6: View All Users Details Based on Community Blockchain

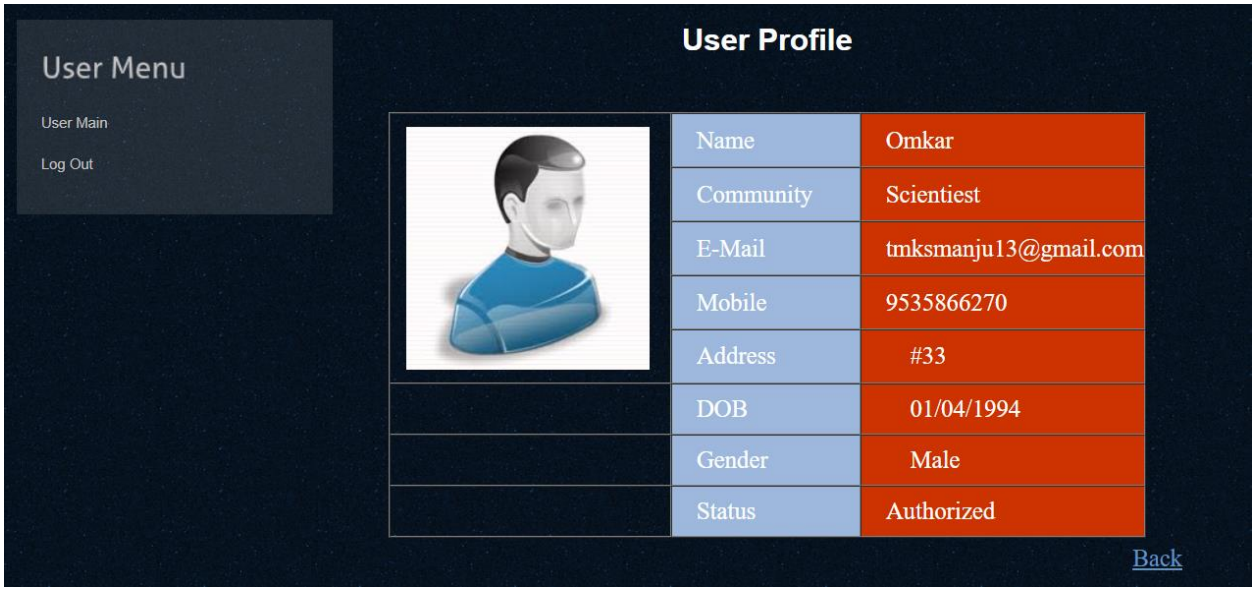


Fig 7: User Profile

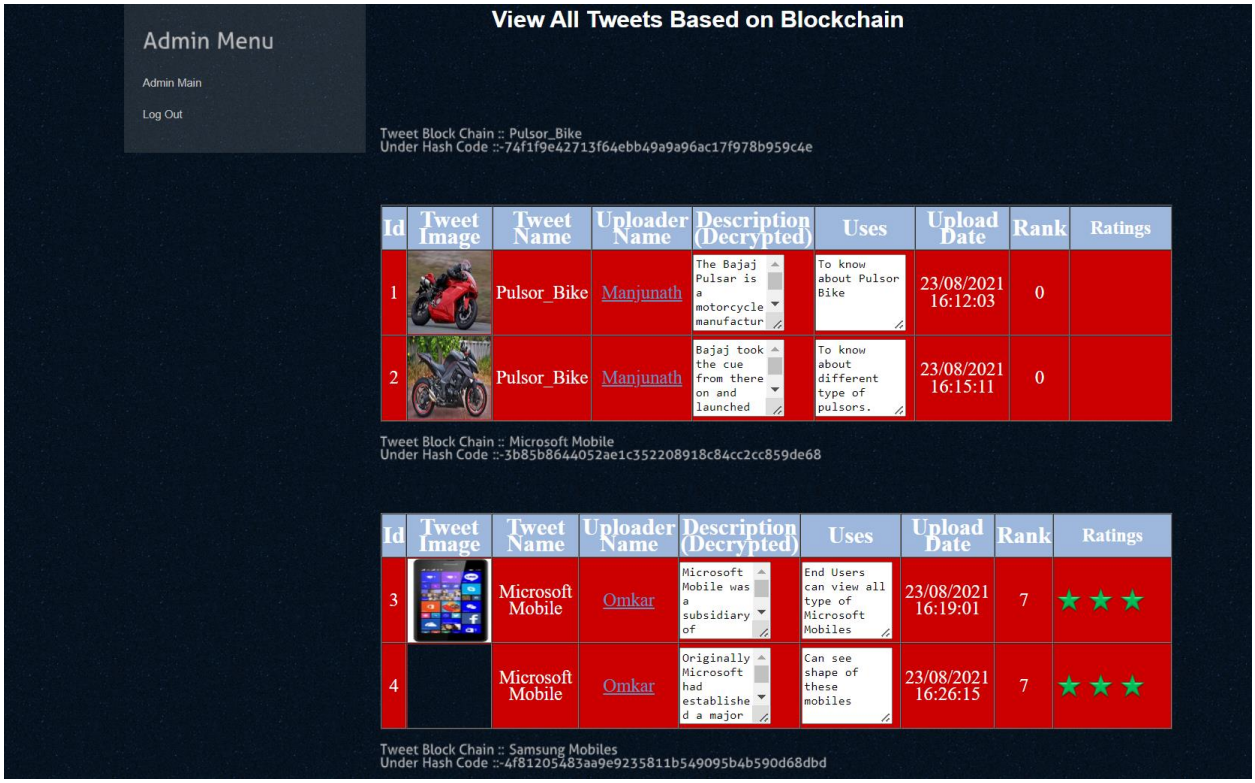


Fig 8: View All Tweets Based on Blockchain

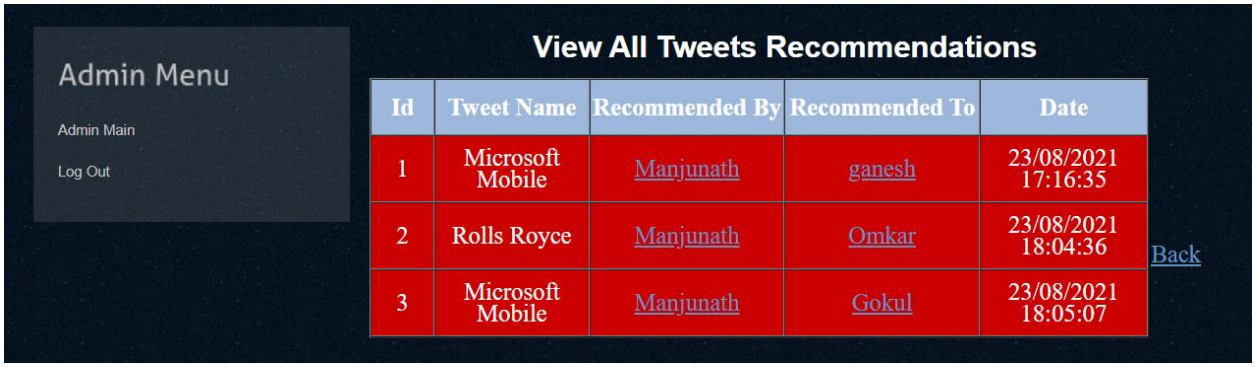


Fig 9: View All Tweets Recommendations

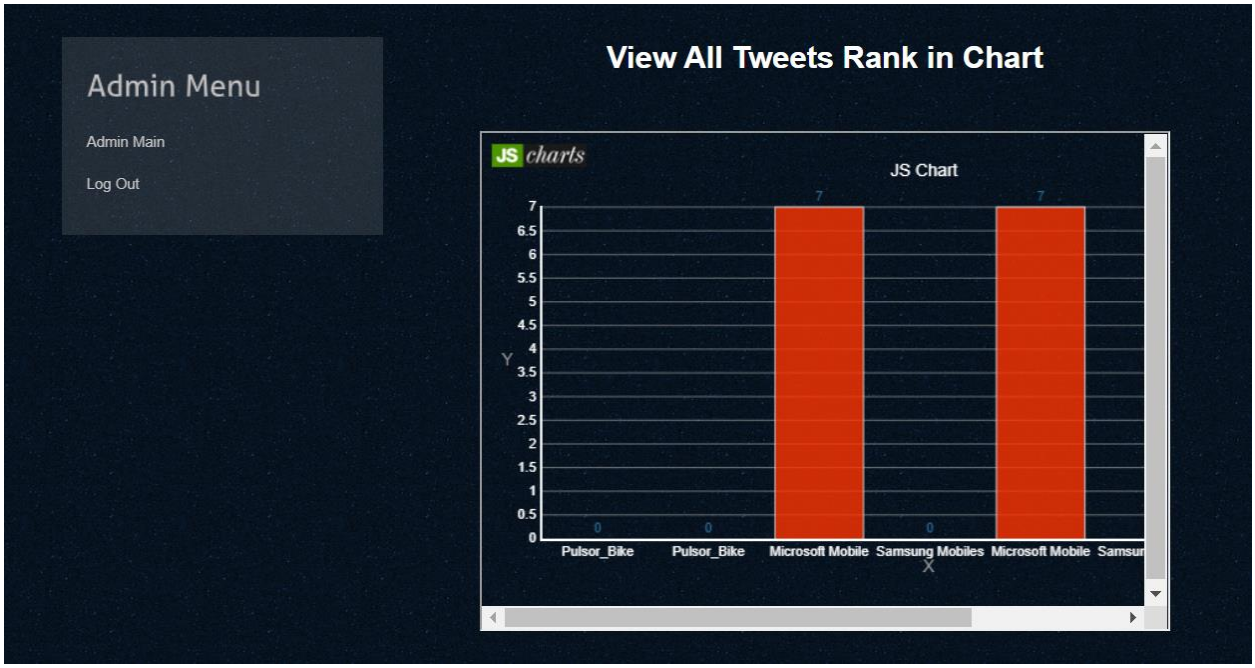


Fig 10: View All Tweets Ranks in Chart

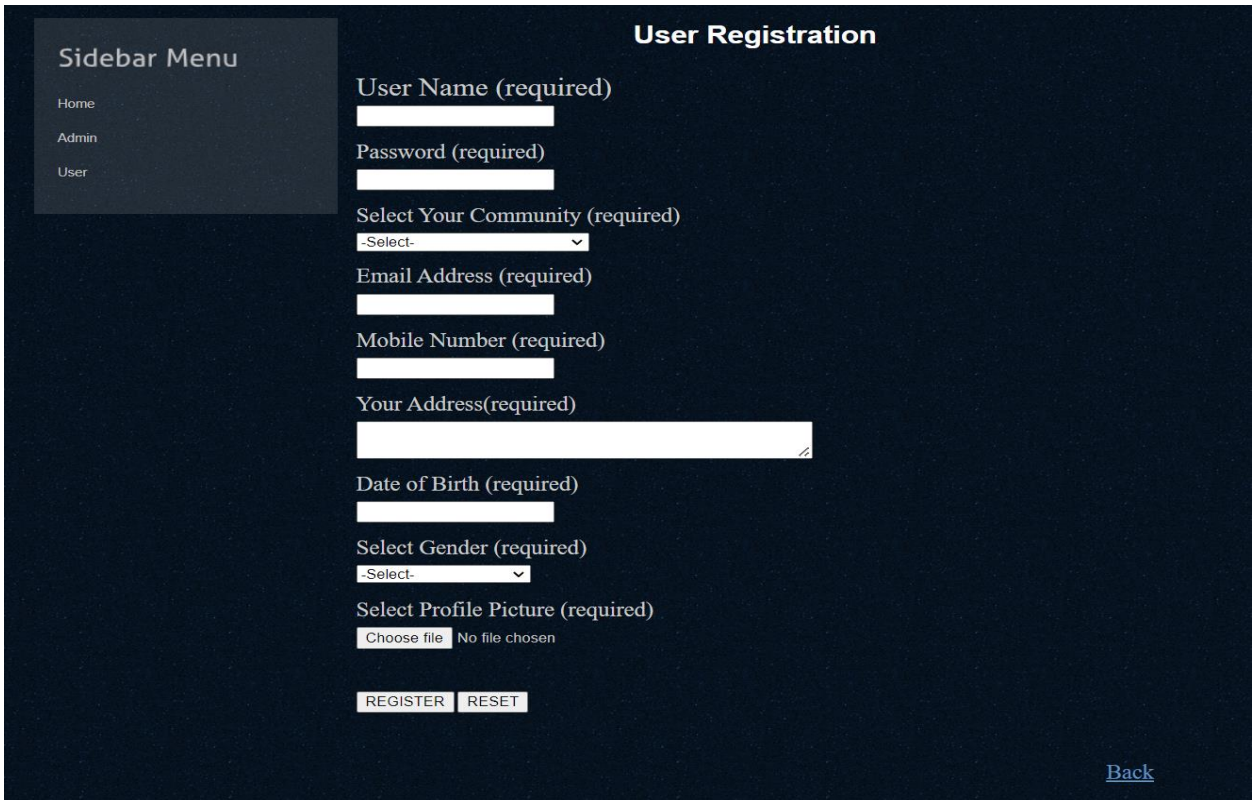


Fig 11: Registration

You are not Authorized by the Admin, Please wait!!!

Back

Fig 12: Before Admin authorizing(if user login)

Fig 13: Waiting State

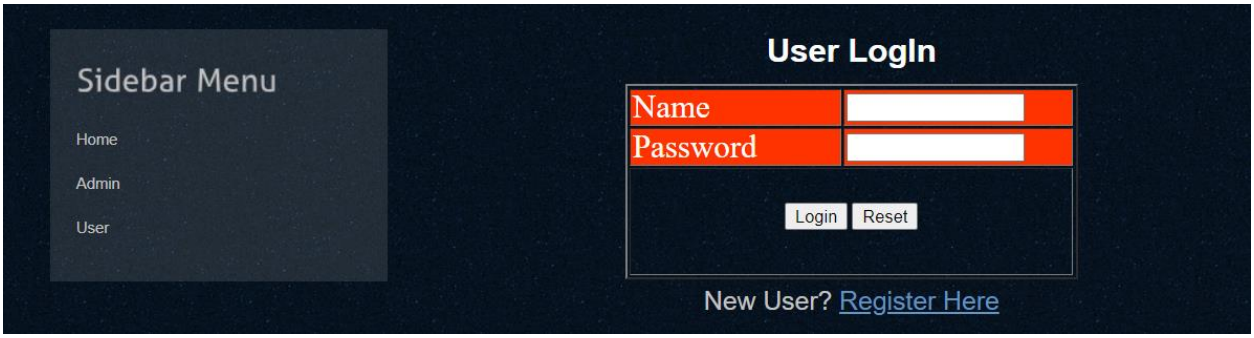


Fig 14: User Login

Fig 15: User Page

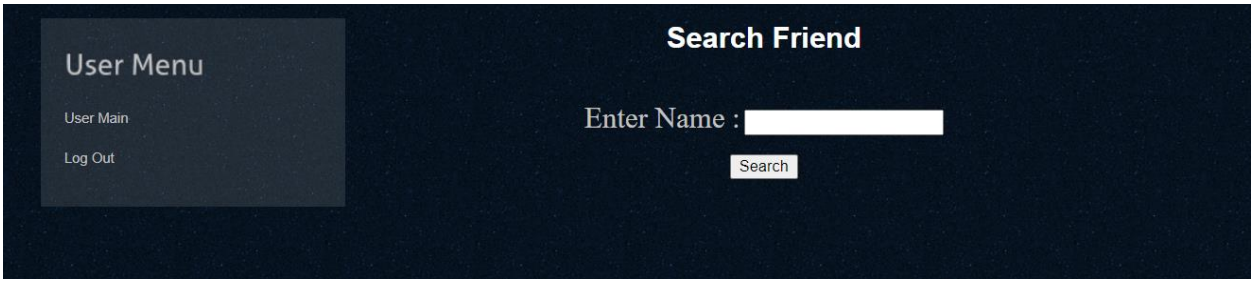


Fig 16: Search for Friend

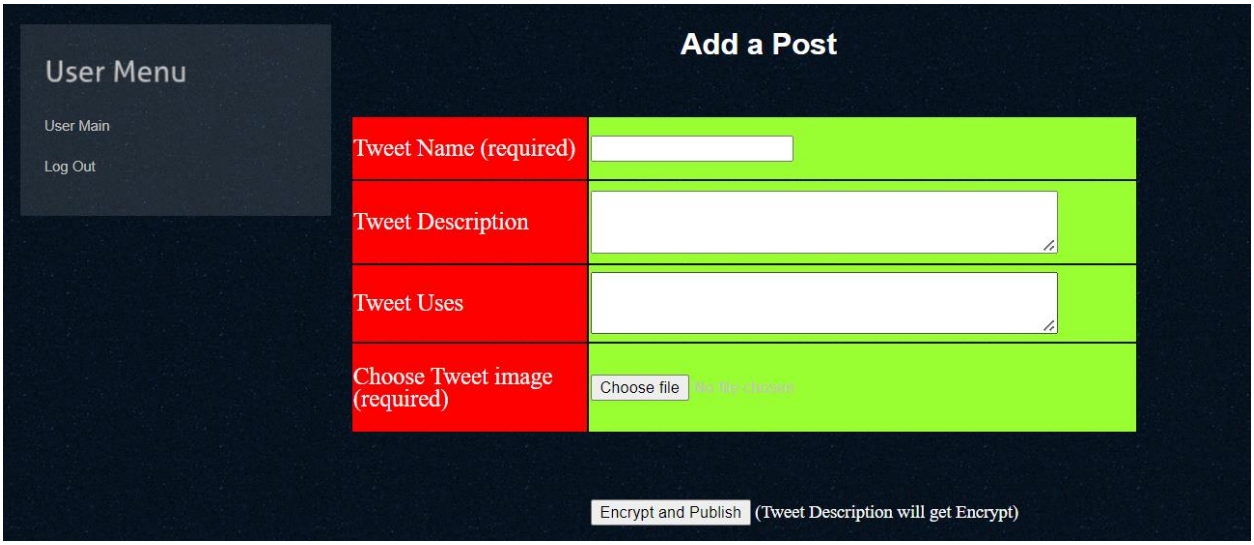


Fig 17: Add a Post

CONCLUSION

This paper presents an implementation of block chain used in OSN. Users keep their security information under their control, in order to avoid security information leakage from centralized servers. Additionally, since the social network service is decentralized, users do not need to worry about service crash down by centralized entity. Furthermore, there is a DAO for the whole users to self-manage their social network. It is possible for an OSN to develop sustainably without a centralized leader. The block chain implemented in this project not only provides a decentralized environment for OSN, but also make it possible for users to manage their social network in a decentralized way.

For future work, a user-friendly interface will be developed in order to replace the CLI clients since they are not very suitable for normal users. As a public IPFS network is used in this project, in order to improve the data privacy level, a private IPFS network will be developed. In autonomy part, the simulate plan will be needed in further development, the simulate plan can use tokens to motivate users to create more high-quality content in OSN and pay their effort in the autonomy part.

REFERENCES

[1] D. Chaffey, "Global social media research summary 2020 | Smart Insights", Smart Insights, 2020. [Online]. Available: https://www.smartinsights.com/social-media-marketing/social-media-strategy/new-global-social-media-research/. [Accessed: 03- May- 2020].

[2] L. Constantin, "Credential stuffing explained: How to prevent, detect and mitigate", *CSO Online*, 2019. [Online]. Available: https://www.csoonline.com/article/3448558/credential-stuffing-explained-how-to-prevent-detect-and-defend-against-it.html. [Accessed: 30- Apr- 2020].

[3] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System", *Bitcoin.org*, 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf. [Accessed: 16- Feb- 2020].

[4] "Cryptocurrency Prices: Coins Market Cap Live Coin Prices for All Coins", *Blockonomi*. [Online]. Available: https://blockonomi.com/market-cap/. [Accessed: 24- Aug- 2020].

[5] M. Di Silvestre, P. Gallo, M. Ippolito, E. Sanseverino, G. Sciume and G. Zizzo, "An Energy Blockchain, a Use Case on Tendermint", *2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe)*, 2018. Available: 10.1109/eeeic.2018.8493919.

[6] "What is Tendermint | Tendermint Core", *Docs.tendermint.com*, 2020. [Online]. Available: https://docs.tendermint.com/master/introduction/what-is-tendermint.html. [Accessed: 26- Aug- 2020].

[7] V. Buterin, *Ethereum White Paper - A NEXT GENERATION SMART CONTRACT & DECENTRALIZED APPLICATION PLATFORM*. 2014, p. 23.

[8] S. Henningsen, M. Florian and S. Rust, "Mapping the Interplanetary Filesystem", *arXiv preprint*, arXiv:2002.07747, 2020