DAYANANDA SAGAR COLLEGE OF ENGINEERING
(An Autonomous Institute affiliated to VTU, Belagavi, Approved by AICTE & ISO 9001:2008 Certified)
Accredited by National Assessment & Accreditation Council (NAAC) with 'A' grade, Shavige
Malleshwara Hills, Kumaraswamy Layout, Bengaluru-560078

Final Project Report

on

"Decentralized Finance (DeFi) Banking Application Using Blockchain Technology"

Submitted By
Abhinav Chettri [1DS19CS002]
Sai Prithvi Adapa[1DS19CS138]
Siddharth Shivam[1DS19CS159]
Ubaid Ul Rehman [1DS19CS755]

8th Semester B.E. (CSE)

Under the guidance of
Anupama Girish
Assistant Professor
Dept. of CSE
DSCE, Bangalore

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
DAYANANDA SAGAR COLLEGE OF ENGINEERING
Bangalore-78

VISVESVARAYA TECHNOLOGICAL UNIVERSITY
DAYANANDA SAGAR COLLEGE OF ENGINEERING

Shavige Malleshwara Hills, Kumaraswamy Layout, Bangalore - 560078
Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the project entitled "**Decentralized Finance (DeFi) Banking Application Using Blockchain Technology**" is a bonafide work carried out by Abhinav Chettri [1DS19CS002], Sai Prithvi Adapa [1DS19CS138], Siddharth Shivam [1DS19CS159], Ubaid Ul Rehman [1DS19CS755] in partial fulfillment of 8th semester, Bachelor of Engineering in Computer Science and Engineering under Visvesvaraya Technological University, Belgaum during the year 2022-23.

Dr. Ramesh Babu D R
Vice Principal & Head of Department,
CSE, DSCE

Signature: . . . . . .

Name of the Examiners:

1: . . . . . .

2: . . . . . .

Dr. B G Prasad
Principal,
DSCE

Signature: . . . . . .

Signature with date:

. . . . . .

. . . . . .

Abstract

Typically, traditional or current banking systems require workers during different parts of transactions leaving room for mistakes such as alteration of bank balance amounts or corrupting transaction records by malpractices like tampering etc. That's not ideal nor is the slow-speed process involved when depositing and withdrawing money in these cases satisfactory. We are introducing a Defi Banking App that solely relies on automated processes featuring exclusive Ethereum Blockchain Technology - providing assurance with zero human intervention. Within our concept, users appreciate integrated Smart Contracts controlling every fragment of the service accordingly. Depositing cryptocurrency, earning rewards through it comes along with swift, instant & comfortable features.

Keywords: Blockchain, decentralization, cryptocurrency, Ethereum, banking system, smart contract.

Acknowledgement

We have taken efforts in this project on “Decentralized Finance (DeFi) Banking Application System Using Blockchain Technology” However, it would not have been possible without the kind support and help of many individuals. We would like to extend my sincere thanks to all of them.

We would like to take this opportunity to express our gratitude to Dr. B G Prasad, Principal of DSCE, for providing us with an opportunity to showcase our best work throughout the course.

We owe our deep gratitude to our project guide, Anupama Girish, Assistant Professor, CSE, DSCE for her constant support and encouragement. Her guidance and supervision have helped us achieve the goals of the project.

We are also very grateful to our respected Vice Principal, HOD of Computer Science Engineering, DSCE, Bangalore, Dr. Ramesh Babu D R, for his support and guidance.

We would like to thank our project coordinators Dr. Ramya RS Associate Professor, CSE, DSCE for believing in us. We are also thankful to all the other faculty and staff members of our department for their kind cooperation and help.

Lastly, Our thanks and appreciation also go to our classmates and friends in developing the project and people who have willingly helped us out with their abilities.

Abhinav Chettri [1DS19CS002]
Sai Prithvi Adapa[1DS19CS138]
Siddharth Shivam[1DS19CS159]
Ubaid Ul Rehman [1DS19CS755]

Contents

Chapter 1: Introduction

Introduction

Welcome to DeFi Bank Web Application, where traditional banking is transformed through blockchain technology, providing unmatched financial freedom and security. Our platform empowers individuals and businesses to take control of their assets without intermediaries. With our intuitive web application, you can seamlessly access decentralized financial services from any device.

Privacy and security are our top priorities. Built on smart contracts and encryption protocols, our web application ensures the protection of your funds and personal information. With robust security measures in place, you can confidently manage your finances, transact, and invest.
Our comprehensive services include:

1. Wallet Management: Create and manage your digital wallet securely, storing various cryptocurrencies and tokens. Maintain complete control over your funds, effortlessly sending, receiving, and tracking transactions.

2. Lending and Borrowing: Experience decentralized lending and borrowing, accessing loans without traditional credit checks. Earn interest by lending your assets to borrowers worldwide while retaining control of your funds.

3. Yield Farming and Staking: Amplify your crypto holdings through yield farming and staking. Earn passive income by providing liquidity to decentralized exchanges or locking assets in staking pools, maximizing your returns.

4. Asset Exchange: Trade a wide variety of cryptocurrencies and tokens seamlessly on our platform. Benefit from low fees, fast transactions, and a secure trading environment. Advanced features like limit orders and trading charts are available to execute precise strategies.

Governance Participation: Engage in decentralized governance within the DeFi Bank ecosystem. Influence decision- making, propose improvements, and vote on essential matters. Your participation shapes the future of decentralized finance. Join the DeFi Bank Web Application today and embrace a new era of financial sovereignty. Experience limitless potential and take charge of your financial destiny in the decentralized finance landscape.

1.1 Overview

DeFi, short for Decentralized Finance, is a groundbreaking movement within the blockchain and cryptocurrency space. It aims to revolutionize traditional financial systems by leveraging blockchain technology and smart contracts to create an open, accessible, and transparent ecosystem of financial applications and services. By eliminating intermediaries and enabling peer-to-peer transactions, DeFi empowers individuals to have greater control over their finances, access a wide range of decentralized financial instruments, such as lending, borrowing, trading, and yield farming, and participate in a global financial system without the need for traditional banks. With its potential to enhance financial inclusion, reduce costs, and increase transparency, DeFi has emerged as a disruptive force that has garnered significant attention and investment, driving innovation in the world of finance.

1.2 Motivation

The motivation behind our proposed DeFi banking application stems from the limitations and shortcomings of traditional banking systems. Currently, these systems heavily rely on manual interventions, which can introduce various risks, including the potential for tampering with transaction history or bank balances. Such vulnerabilities compromise the security and trustworthiness of the financial ecosystem, leaving users exposed to potential fraud or unauthorized modifications. To address these challenges, we propose a DeFi banking application utilizing Ethereum Blockchain technology. By leveraging the decentralized and immutable nature of the blockchain, we aim to automate the banking system and eliminate the need for human interceptions. Through the implementation of smart contracts, we can ensure transparent and secure procedures, reducing the possibility of manipulation and fraudulent activities.

Our proposed model empowers users by allowing them to deposit and withdraw cryptocurrencies seamlessly, providing a faster and more efficient alternative to traditional banking methods. Furthermore, users have the opportunity to earn rewards for their participation in the system, incentivizing their engagement and promoting the adoption of decentralized finance. The key

motivation driving our solution is to establish a fully automated banking system that offers enhanced security, transparency, and speed. By eliminating human interceptions and enabling instant transactions, users can enjoy the convenience of conducting financial operations from the comfort of their homes or workplaces

1.3 Objective

- To provide a small scale organization, who cannot afford expensive solutions, with a scalable and easy-to-use threat intel platform that will help them in keeping up with cyber attacks and successfully mitigating them through due process.
- Our goal is to provide a faster, more efficient alternative to traditional banking systems, allowing users to perform instant transactions from the convenience of their homes or workplaces. By leveraging decentralization and blockchain technology, we aim to revolutionize the banking industry and promote the adoption of decentralized finance.

1.4 Scope

- Our proposed DeFi banking application aims to automate the banking system, eliminating human interceptions and ensuring the integrity of transactions. Users can deposit, withdraw, and earn rewards using cryptocurrencies. The scope includes instant transactions, enhanced security through the use of Ethereum Blockchain and smart contracts, and the inability to modify financial transactions.

- The system provides convenience and accessibility, allowing users to perform transactions from home or work. By leveraging decentralization and blockchain technology, our scope is to revolutionize traditional banking systems and promote the adoption of decentralized finance.

Chapter 2: Problem Statement and Proposed solution

2.1 Problem Statement

The problem at hand is to develop a decentralized banking application using blockchain technology that overcomes the limitations of traditional banking systems and provides an inclusive, transparent, and accessible financial ecosystem. The goal is to create a platform that offers open and permissionless financial services to individuals worldwide, regardless of their location or financial background. The application should address the challenges of centralized control, lack of transparency, and restricted access to financial services while reducing costs, streamlining processes, and ensuring the security of user funds. By leveraging blockchain technology, the aim is to create a decentralized finance (DeFi) banking solution that empowers users, eliminates intermediaries, and promotes financial inclusion on a global scale.

2.2 Existing Systems

The existing banking system, known as the traditional or centralized banking system, operates through a network of physical branches, ATMs, and online platforms. Traditional banks act as intermediaries, holding custody of customer funds and facilitating financial transactions. They offer a range of services, including deposit accounts, loans, credit cards, payment processing, and investment products. While banks provide security measures and regulatory compliance, customer privacy can be compromised due to data collection and reporting requirements. Accessibility to banking services may be limited in underdeveloped areas. The traditional banking system lacks transparency, and customers have limited visibility into how their funds are utilized. Moreover, financial inclusion can be challenging, as certain individuals or communities may face barriers to accessing banking services. These limitations have spurred the emergence of decentralized finance (DeFi) as an alternative ecosystem that leverages blockchain technology to provide an inclusive, transparent, and accessible financial system.

2.3 Proposed Solution

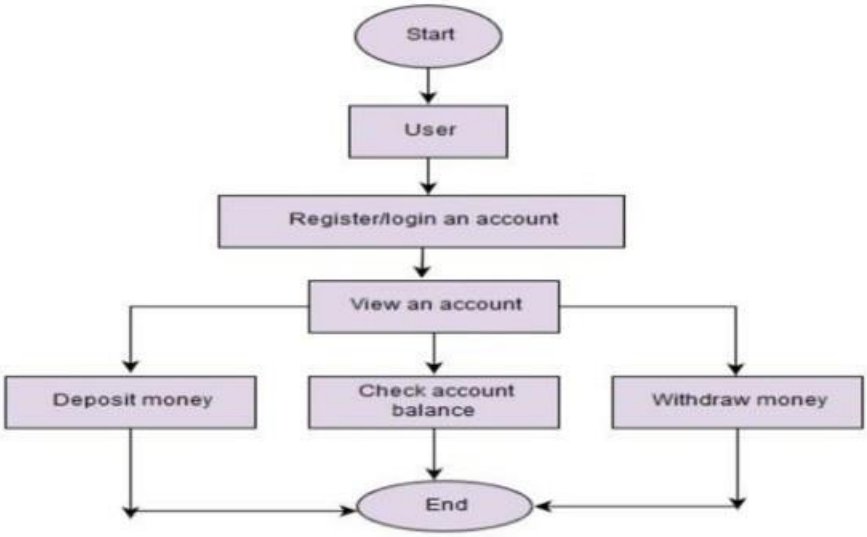


Figure 1.1 flow chart of proposed system

As you can see in the figure, a user will run the application through a terminal, which will start a web application on the specified port. This web app will then be accessible from any browser.

The web application is dashboard-based and it will show everything that a bank user needs. The flow of procedure is as follows:

- Register/login into an account.
- Deposit Money.
- Withdraw Money.
- Internet Banking.

The proposed approach comprises putting in place an accounting system designed exclusively for ether transactions inside a banking establishment. People will go through a smart contract enrollment procedure via web or mobile banking platforms, providing identity information. The smart contract then creates an account for the user, allowing them to utilise that account to conduct a range of

transactions with the bank or smart contract. By engaging with the smart contract, users may add ethers to their accounts and take money from their ethereum wallets. Users may see their account status and balances using internet banking or mobile banking systems, made possible through a user interface that serves as a client, creating a connection between the system's smart contract and user.

2.4 Work Plan:

Project Initiation:

- Define project objectives, scope, and deliverables.
- Identify stakeholders and establish communication channels.
- Form the project team and assign roles and responsibilities.
- Develop a project timeline and set milestones.

Research and Requirements Gathering:

- Conduct market research on the DeFi ecosystem and user needs.
- Analyze regulatory requirements and compliance considerations.
- Gather requirements from stakeholders for features and functionalities.

Architecture and Design:

- Design the overall application architecture.
- Define user interface (UI) and user experience (UX) design.
- Determine the blockchain platform and smart contract architecture.

Development:

- Develop the frontend with UI/UX implementation.
- Implement backend logic, including smart contracts and transaction processing.
- Integrate external APIs and conduct thorough testing.

- Deployment and Infrastructure Setup:

- Set up necessary infrastructure and deploy to a test environment.
- Perform load testing and ensure scalability.
- Plan deployment strategy for the production environment.

User Testing and Feedback:

- Conduct user acceptance testing and gather feedback.
- Incorporate user feedback and iterate on design and features.
- Perform security audits and address vulnerabilities.

Launch and Post-Launch Activities:

- Prepare documentation and user guides.
- Execute marketing and promotional strategy.
- Monitor performance, address issues, and update the application.

Ongoing Maintenance and Support:

- Provide ongoing maintenance, bug fixes, and security updates.
- Offer customer support and address inquiries.
- Stay updated with regulations and compliance requirements.

Chapter 3 Literature Survey

DeFi banking application using blockchain technology would involve researching and analyzing existing academic papers, articles, and publications related to decentralized finance, blockchain technology, and its applications in the banking sector.

A literature survey for a DeFi banking application using blockchain technology would involve researching and analyzing existing academic papers, articles, and publications related to decentralized finance, blockchain technology, and its applications in the banking sector. Here are some key areas to focus on during the literature survey:

3.1 Cryptocurrencies and Decentralize[defi]:

In conclusion, this paper explores the DeFi (Decentralised Finance) system's possible benefits and drawbacks in comparison to conventional financial systems. DeFi has a lot of potential and has the potential to be quite helpful, according to the analysis. It has a permissionless design and cheap transaction costs, however taxation might be a problem. DeFi can decrease capital efficiency even though it addresses problems with the fractional reserve system by overcollateralizing each loan. Investors taking on leveraged positions and the use of stablecoins are two issues with DeFi. DeFi has a bright future overall because of its capacity to cut costs, but it does not automatically resolve the banking industry's issue with rent and might pose other difficulties.

3.2 Overview of Blockchain Implementation on Islamic Finance:

This paper provides an overview of Saadiqin, a project that explores the use of blockchain technology in Islamic Finance. Saadiqin aims to develop Islamic banking solutions based on Muamara financial contracts, which follow the principles of profit and loss sharing. The paper discusses the advantages of integrating blockchain into the financial sector, emphasizing its ability to securely record and link transactions while maintaining transparency. The diagram illustrates the overall structure of Saadiqin and its integration with blockchain, utilizing Hyperledger and MuleSoft. The integration process required minimal modifications to the existing financial services system, with a focus on enhancing transaction transparency and openness.

3.3 Decentralized Finance on Blockchain and Smart Contract Based Financial Markets:

DeFi is a ground-breaking financial system that runs through smart contracts on the blockchain that are built on code, doing away with the need for middlemen. This article discusses potential hazards in addition to beneficial DeFi use cases. While Ethereum's smart contracts offer safe cryptocurrency storage, they don't offer any further advantages. The post suggests transferring bitcoins to a decentralised money market like Compound so they can earn interest in order to increase their temporal worth. DeFi's protocols and admin keys, which provide smart contract renewal and emergency shutdowns, ensure reliability. DeFi offers secure cryptocurrency exchanges, easy configuration, speedy transfers, and openness. For DeFi applications, the Ethereum blockchain enables these characteristics.

3.4 Decentralized Finance: The Case for a Holistic Approach to Regulating the Crypto Industry:

The need for cryptocurrency regulation is discussed in the essay, especially in light of financial crime and how cryptocurrencies interact with conventional finance. It emphasises the significance of applying laws everywhere, especially FATF recommendation number 16. The growth of decentralised finance (DeFi), which is outside the purview of the FATF, presents regulatory concerns. Increasing borrowing alternatives and the capacity to use loans as leverage for additional investments are two advantages of bitcoin lending and DeFi. Along with the openness and interoperability of DApps, the effectiveness and affordability of blockchain-based systems are also emphasised. In order to overcome regulatory loopholes and create complete strategies for centralised and decentralised cryptocurrency trading platforms, the essay urges cooperation between regulators and the sector.

3.5 Blockchain (Distributed Ledger Technology) Solves VAT Fraud:

According to a survey at the World Economic Forum, a significant tipping point is expected to occur by 2023, as stated by business executives and technology experts. The potential of blockchain technology is highlighted in various contexts, such as improving tax collection and facilitating centralized data centers in regions like the East African Community and Gulf Cooperation Council. The introduction of blockchain-based systems, including the Digital Invoice Customs Exchange (DICE), aims to enhance domestic data collection, accuracy, and collaboration. The decentralized nature of blockchain enables trustless interactions,

eliminating the need for intermediaries and promoting efficiency in various sectors, including finance and taxation.

3.6. KyberNetwork:A trustless decentralized exchange and payment service:

The architecture and development of KyberNetwork, an on-chain protocol that enables quick trades and conversions between digital assets and extremely liquid cryptocurrencies, are presented in this paper. As the first decentralised exchange system to accomplish dependable execution, instant trading, and high liquidity, KyberNetwork intends to set the bar high. Along with its exchange features, KyberNetwork provides a Payment API that enables Ethereum accounts to quickly process payments made with a variety of cryptocurrencies, with merchants receiving payment in either Ether or their preferred tokens. Plans for cross-chain transactions using other cryptocurrencies via relays and protocols like Polkadot and Cosmos are included in the roadmap. Derivatives are being introduced in an effort to lower user risk and enable cryptocurrency selection utilising KyberNetwork Crystals (KNC).

3.7 RBI Distributed Ledger Technology and Blockchain- A Future of Decentralized India:

The adoption of cryptocurrencies and blockchain technology in emerging markets like India is the main topic of this essay. Developing nations recognise the benefits that these technologies bring, even though wealthy ones have already adopted them for banking and business. The essay examines how cryptocurrencies, blockchain technology, and distributed ledger technology (DLT) could compete with fiat central banks. It also covers how cryptocurrencies can enhance the transactions and operations of central banks. With the assistance of the Indian government, the Reserve Bank of India (RBI) is deploying strengthened regulatory sandboxes and other technologies to pave the way for tech-centric growth. The paper explores how DLT is being used in India using a SWOC framework, looking at various central bank initiatives and efforts to pick up new skills. Despite the fact that the majority of central banks have not yet.Although the majority of central banks have not officially announced their plans for DLT-based software, initiatives like central bank digital currencies (CBDCs) can help the expansion of the financial markets.

3.8 Conclusion

We have explored the world of decentralised finance (DeFi) throughout this study and have developed a thorough grasp of its many facets. We started by defining DeFi, discussing its relevance, and looking at how it transforms conventional financial systems and gives people more control over their possessions. Then, we looked at how DeFi functions, including smart contracts and decentralised apps (DApps), emphasising how these tools make transactions easier and do away with middlemen. This survey has given a thorough overview of DeFi, highlighting the importance of blockchain and smart contracts in maintaining the security and stability of decentralised financial systems. The DeFi ecosystem can continue to develop and flourish, protecting the interests of its players in a world that is becoming more digital and networked, by remaining alert, utilising cutting-edge technology, and encouraging collaboration.

These areas include:

- The impact of DeFi on traditional banking systems: Explore how the principles and concepts of DeFi can disrupt and transform traditional banking models.
- Technical aspects of DeFi banking apps: Investigate the underlying technologies, such as blockchain, smart contracts, and decentralized protocols, that power DeFi banking applications. Examine the challenges and opportunities associated with building secure and scalable infrastructure.
- Security and privacy considerations: Analyze the unique security risks and privacy concerns inherent in DeFi banking apps. Explore potential solutions and best practices to protect user assets and personal information.
- User experience and adoption: Examine user behavior, motivations, and barriers to adoption of DeFi banking apps. Explore strategies for improving user experience, interface design, and increasing user adoption and retention.
- Regulatory and legal challenges: Investigate the evolving regulatory landscape for DeFi banking apps. Examine compliance requirements, regulatory frameworks, and legal implications of decentralized systems in the context of banking.
- Financial inclusion and accessibility: Assess the potential of DeFi banking apps in promoting financial inclusion and providing access to financial services for underserved populations. Explore the impact on unbanked or underbanked communities and potential strategies for bridging the gap.
- Risk management and governance: Examine risk management strategies, decentralized governancmodels, and mechanisms for addressing disputes or failures in DeFi banking apps.

- To conduct a thorough literature survey, it is recommended to consult academic databases, research repositories, and relevant journals focused on blockchain, decentralized finance, and banking. These sources will provide access to peer-reviewed articles, research papers, and industry reports that can enhance your understanding of the topic.
- The architecture and design of a DeFi banking app can vary depending on the platform and its specific requirements. However, there are common components and considerations involved in the system architecture and design.
- The user interface (UI) is the front-end component that allows users to interact with the app. It should be user- friendly and provide access to various functionalities such as wallet management, lending, borrowing, asset exchange, and portfolio tracking.
- The backend infrastructure consists of servers, databases, and networking components that process and store data. It may include cloud services, APIs, and third-party integrations.
- Blockchain integration is essential for DeFi banking apps, as they leverage blockchain technology. This integration allows users to interact with smart contracts, execute transactions, and access decentralized financial services.
- Smart contracts, which are self-executing contracts on the blockchain, automate transactions and financial operations within the app. They need to be carefully designed and audited for security and functionality.
- Wallet management ensures the secure storage and management of users' cryptocurrency holdings. Integration with hardware wallets or secure key management solutions enhances security.
- Lending and borrowing functionality requires a lending pool mechanism to facilitate transactions between users. Considerations include interest rate calculation, collateral management, and loan repayment mechanisms.
- Asset exchange functionality can be achieved through integration with decentralized exchanges (DEXs) and liquidity pools. This enables users to trade assets at competitive prices.

- Data analytics and reporting features provide users with insights into their portfolio performance and transaction history, aiding decision-making and tracking financial activities.
- Security and compliance are crucial. Robust security measures protect user funds, personal information, and smart contracts. Compliance with regulatory requirements, KYC, and AML procedures may also be necessary.
- Scalability and performance considerations ensure that the architecture can handle increasing user demand. Load balancing, caching mechanisms, and database optimization contribute to optimal performance.
- It's important to note that the actual architecture and design of a DeFi banking app may vary depending on the specific platform and project requirements. Consulting with experienced blockchain developers and architects is advisable to create a tailored solution for a DeFi banking app

Chapter 4: Architectural and System Design

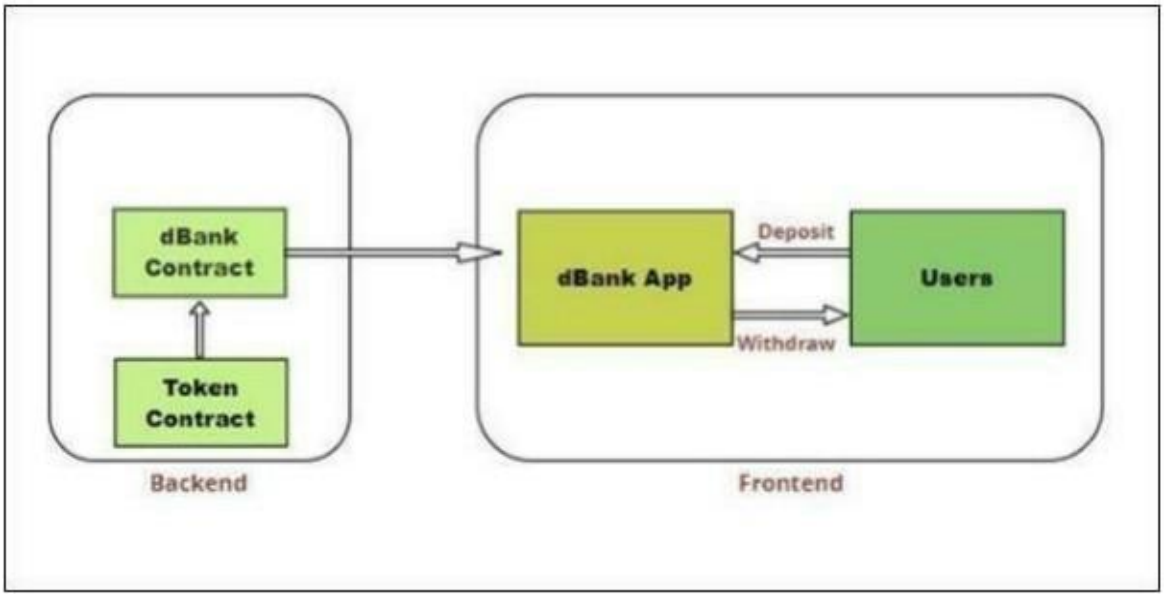
4.1 System Overview:

The following programming languages were used to construct this financial system:

- Solidity in the design of smart contracts.
- For communicating with the smart contract, use JavaScript.

The following frameworks and libraries are used:

- Ganache library to build the local test network.
- React.js is a front-end development framework.
- Using Node.js as a Runtime Environment.
- The Mocha JavaScript framework for testing is used for this purpose.
- Web3 collection for libraries.



4.2System Architecture:

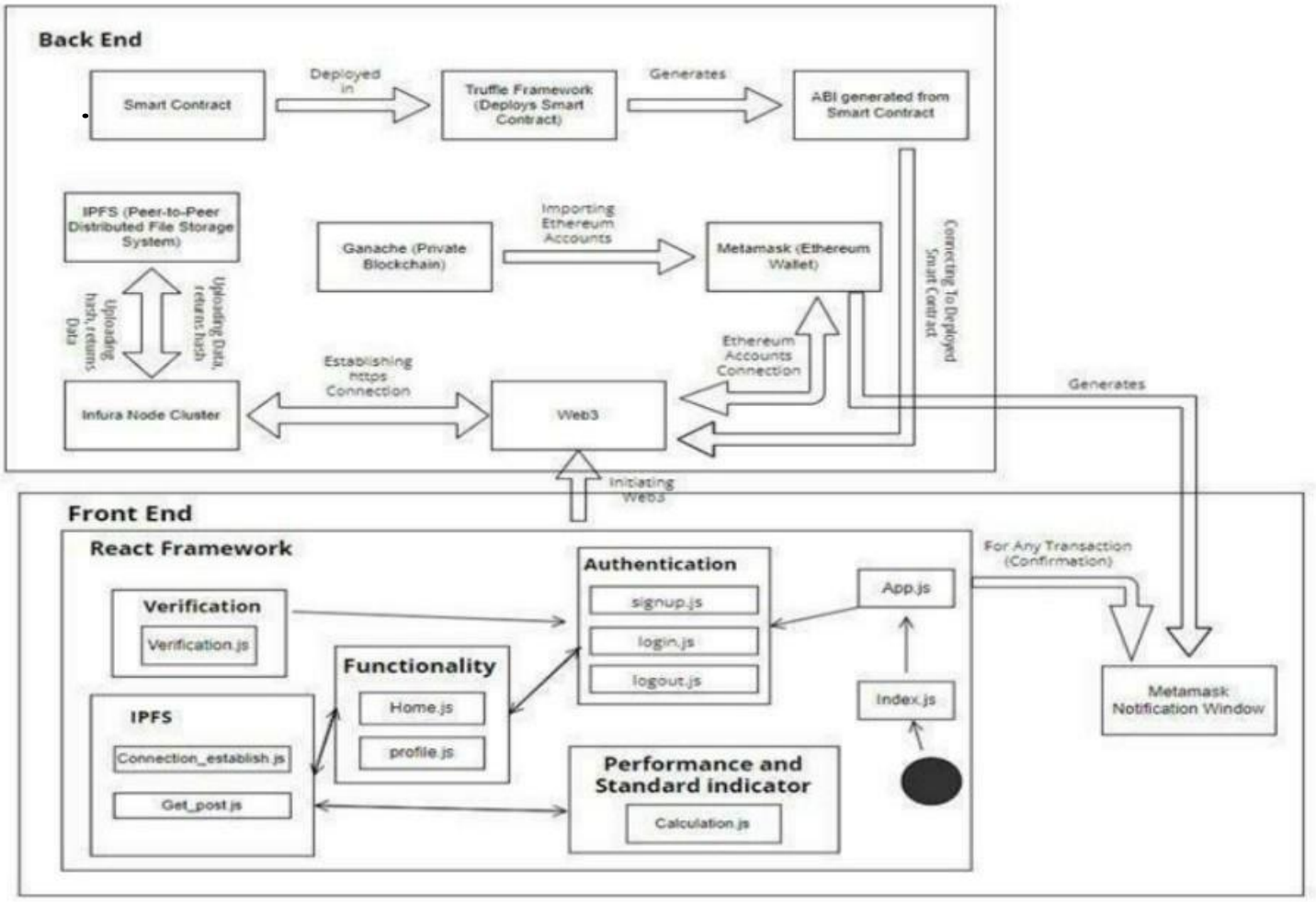


Figure 3.1: System architecture

Layer of the user interface (UI):

- interface that consumers use to communicate with the application.
- includes screens, forms, and user input validation from web or mobile applications.

Applied Layer:

- handles the DeFi banking application's essential business logic and features.
- contains modules for a variety of functionalities, including governance, token swapping, account management, lending and borrowing, staking and yield farming, and more.
- regulates the flow of data and transactions and coordinates the interaction of various components.

Layer for Smart Contracts:

- consists of Ethereum smart contracts or other smart contracts built on the blockchain.
- carries out the logic and business rules for numerous capabilities.
- manages operations including as voting, staking, lending/borrowing agreements, and token transfers.
- leverages Solidity and other programming languages for Etherucontracts.

Cryptocurrency Layer:

- represents the blockchain network(s) underneath which the DeFi application runs.
- Ethereum is the most popular blockchain platform, however others like Binance Smart Chain, Polkadot, and others may also be used.
- records all transactions and state changes, stores the execution of smart contracts, and confirms that it happened.

External Integrations:

- integrates with external services and APIs to retrieve real-time data from oracles, liquidity pools, market data, and token pricing.
- Integration with wallets, identity providers, KYC/AML services, and other third-party systems, as well as decentralised exchanges (DEXs), is possible.
- gives the programme the ability to retrieve and handle outside data needed for a variety of activities.

Infrastructure and Security Layer:

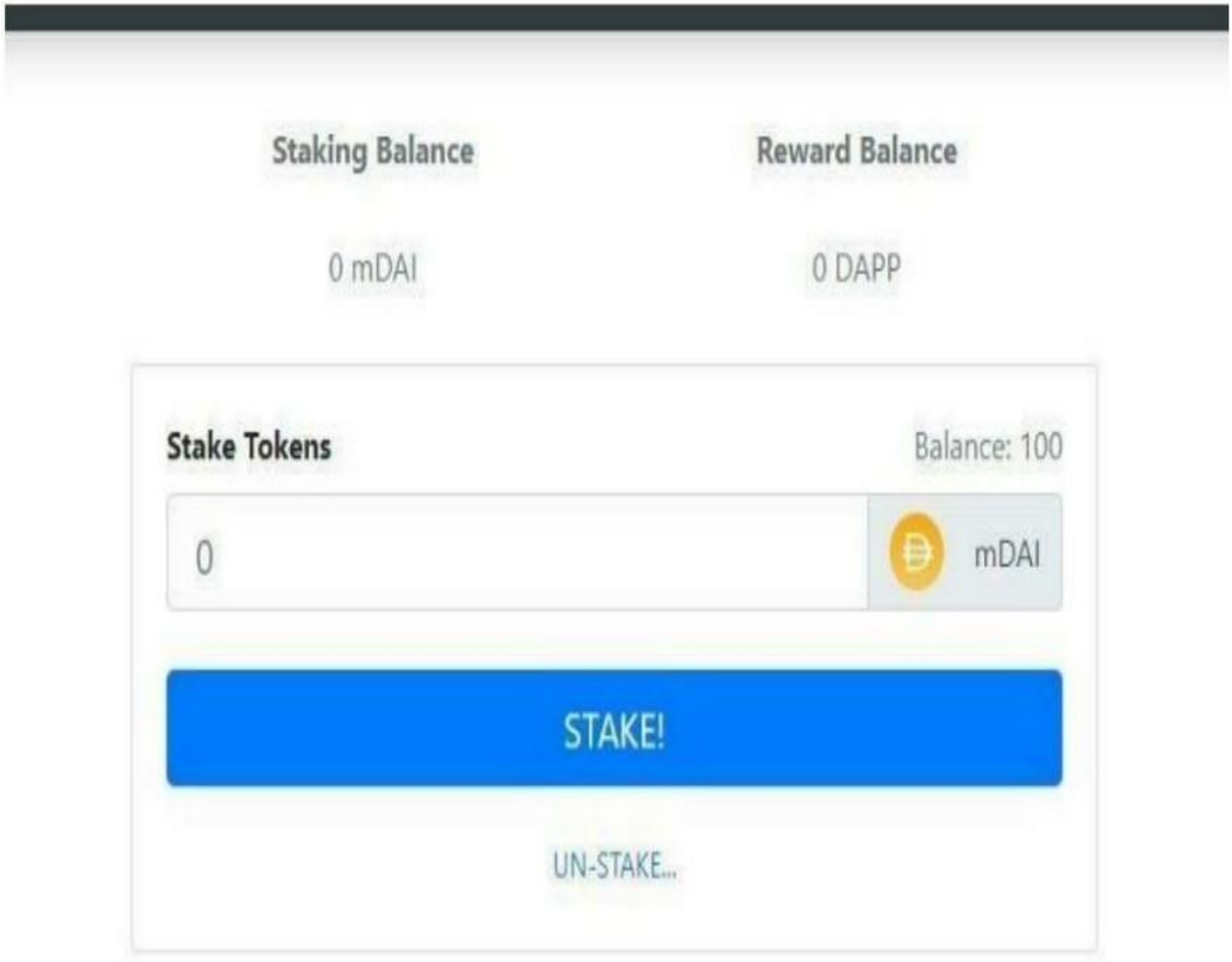
- focuses on guaranteeing the application's performance, reliability, and security.
- includes safeguards including access controls, encryption, authentication, and auditing procedures.
- uses monitoring, logging, and alerting systems to find security events and address system problems.
- takes advantage of infrastructure elements like as servers, cloud services, databases, and cache layers.

External Linkage:

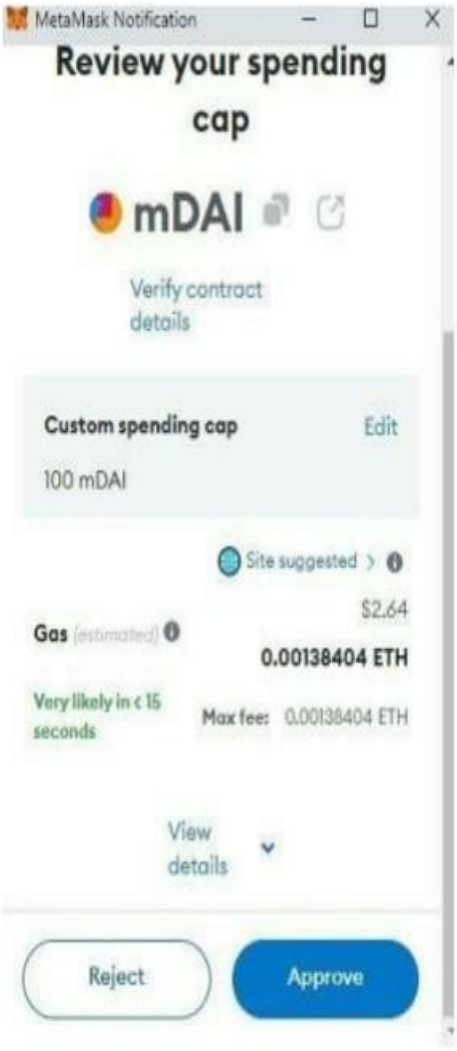
- connects users to external networks and protocols, such the internet, blockchain networks, oracles, and outside data sources.
- guarantees the efficient exchange of data and transactions between the application and outside parties.

4.2.1 Interface Design

The user after logging into the account gets directed to this screen where he/she can stake the tokens. The user can also unstake the tokens here and also can check his/her balance of an account. All the details of the user are shown here. The user can check the staking balance and also the reward balance.




Localhost7545
Connected
Account 2
0xEb5...B378
99.9891 ETH
$190,692.22 USD
Buy
Send
Swap
Bridge
Assets
NFTs
Activity
Unstake Tokens
-0 ETH


MetaMask Notification
Review your spending cap
mDAI
Verify contract details
Custom spending cap
Edit
100 mDAI
Site suggested
Gas (estimated)
$2.64
0.00138404 ETH
Very likely in < 15 seconds
Max fee: 0.00138404 ETH
View details
Reject
Approve

Explanation:

User: The User is an individual who interacts with the Application. The User can perform actions like depositing tokens, making payments, and checking the wallet balance.

Application: The Application serves as the front-end interface that allows the User to interact with the system. It provides functionalities such as depositing tokens, making payments, and checking the wallet balance. The Application communicates with the Smart Contract to execute transactions and retrieve wallet balance information.

Smart Contract: The Smart Contract is a piece of code deployed on a blockchain network (e.g., Ethereum) that holds the payment logic and manages user wallets. It interacts with the Application to receive payment requests and updates the wallet balances accordingly. The Smart Contract also provides functions to query the wallet balance when requested by the User.

Deposit Tokens: The User initiates a token deposit request through the Application. The Application communicates with the Smart Contract, which verifies the transaction and updates the User's wallet balance accordingly.

Make Payment: The User requests a payment through the Application, specifying the recipient and the amount. The Application sends the payment request to the Smart Contract, which deducts the payment amount from the User's wallet balance and transfers it to the recipient's wallet.

Check Wallet Balance: The User can request to check the wallet balance through the Application. The Application communicates with the Smart Contract to retrieve the wallet balance information, which is then displayed to the User.

In this block diagram, the User interacts with the Application, which serves as the user interface. The Application, in turn, communicates with the Smart Contract to execute transactions, update wallet balances, and retrieve balance information. The Smart Contract holds the payment logic and manages the wallet balances on the blockchain.

```
Compiling your contracts...

===========================

> Compiling .\src\contracts\DaiToken.sol

> Compiling .\src\contracts\DappToken.sol
> Compiling .\src\contracts\Migrations.sol
> Compiling .\src\contracts\TokenFarm.sol

> Artifacts written to C:\Users\siddh\OneDrive\Desktop\defi_app\src\abis
> Compiled successfully using:

   - solc: 0.5.16+commit.9c3226ce.Emscripten.clang

Network up to date.

truffle(development)> truffle migrate --reset

Compiling your contracts...

===========================

> Compiling .\src\contracts\DaiToken.sol

> Compiling .\src\contracts\DappToken.sol
> Compiling .\src\contracts\Migrations.sol
> Compiling .\src\contracts\TokenFarm.sol

> Artifacts written to C:\Users\siddh\OneDrive\Desktop\defi_app\src\abis
> Compiled successfully using:

   - solc: 0.5.16+commit.9c3226ce.Emscripten.clang



Starting migrations...
======================
> Network name:    'development'
> Network id:      5777
> Block gas limit: 6721975 (0x6691b7)
```

The "***truffle compile***" command is used in the Truffle development framework for compiling Solidity smart contracts into bytecode that can be deployed on the Ethereum blockchain. Here's how the "***truffle compile***" command works:

- Project Setup: Ensure that you have a Truffle project set up with the necessary project directories and configurations.
- Solidity Contracts: Write your Solidity smart contracts or place them in the appropriate directory within the Truffle project structure.
- Running the Command: Open a terminal or command prompt and navigate to the root directory of your Truffle project.
- Compilation: Run the "truffle compile" command in the terminal. Truffle will automatically detect the Solidity contracts within your project and compile them.
- Compilation Output: Truffle will compile each Solidity contract into bytecode, which represents the low-level instructions that can be executed on the Ethereum Virtual Machine (EVM).
- Artifact Generation: Truffle generates JSON artifacts for each compiled contract. These artifacts contain metadata about the contract, including the contract's ABI (Application Binary Interface) and deployment-related information.
- Output Directory: By default, the compiled contracts and their corresponding JSON artifacts are stored in the "build/contracts" directory within your Truffle project.
- Errors and Warnings: If there are any compilation errors or warnings, Truffle will display them in the terminal, allowing you to identify and address any issues in your Solidity code.
- Verification: After a successful compilation, you can verify the compiled contracts by checking the generated bytecode, ABIs, and other metadata in the JSON artifacts.

The "***truffle compile***" command simplifies the process of compiling Solidity contracts and prepares them for deployment on the Ethereum blockchain. It helps ensure the correctness and integrity of your smart contracts before moving on to the deployment phase.

```
siddh@LAPTOP-2CQS113F MINGW64 ~/OneDrive/Desktop/defi_app (master)
$ truffle compile

Compiling your contracts...
===========================
> Compiling .\src\contracts\DaiToken.sol
> Compiling .\src\contracts\DappToken.sol
> Compiling .\src\contracts\Migrations.sol
> Compiling .\src\contracts\TokenFarm.sol
> Artifacts written to C:\Users\siddh\OneDrive\Desktop\defi_app\src\abis
> Compiled successfully using:
   - solc: 0.5.16+commit.9c3226ce.Emscripten.clang

siddh@LAPTOP-2CQS113F MINGW64 ~/OneDrive/Desktop/defi_app (master)
$ truffle migrate

Compiling your contracts...
===========================
> Compiling .\src\contracts\DaiToken.sol
> Compiling .\src\contracts\DappToken.sol
> Compiling .\src\contracts\Migrations.sol
> Compiling .\src\contracts\TokenFarm.sol
> Artifacts written to C:\Users\siddh\OneDrive\Desktop\defi_app\src\abis
> Compiled successfully using:
   - solc: 0.5.16+commit.9c3226ce.Emscripten.clang


Starting migrations...
======================
> Network name:    'development'
> Network id:      5777
> Block gas limit: 6721975 (0x6691b7)
```

```
   > gas used:            750796 (0xb74cc)
   > gas price:           2.736501189 gwei
   > value sent:          0 ETH
   > total cost:          0.002054554146696444 ETH



  Replacing 'TokenFarm'
  ---------------------

   > transaction hash:    0x618e47c7d2aed1722301f355f7c9255b15d6a5a6cd0dc85a62b2eddf91759a0f

   > Blocks: 0            Seconds: 0
   > contract address:    0x20[illegible]
   > block number:        13
   > block timestamp:     1681232636
   > account:             0x9EdCaa9979946D397834314caA3eb1A2889A265b
   > balance:             99.9838504685972543
   > gas used:            908681 (0xddd89)
   > gas price:           2.71354241 gwei
   > value sent:          0 ETH
   > total cost:          0.00246574443066121 ETH
Compiled successfully!

You can now view defi-tutorial in the browser.

  Local:            http://localhost:3000/
  On Your Network:  http://192.168.56.1:3000/
```

The "***truffle migrate***" command is used in the Truffle development framework to deploy smart contracts onto the Ethereum blockchain. Here's how the "***truffle migrate***" command works:

- Project Setup: Ensure that you have a Truffle project set up with the necessary project directories and configurations.
- Configuration Files: In your Truffle project, you should have a "***truffle-config.js" or "truffle.js"*** file that contains the configuration settings for your deployment.

- Migration Scripts: Truffle uses migration scripts to manage the deployment of smart contracts. Migration scripts are JavaScript files that define the deployment steps for each contract.

- Migration Execution: Open a terminal or command prompt and navigate to the root directory of your Truffle project.

- Network Selection: Specify the desired network for deployment by using the ***"--network***" flag followed by the network name. For example, "***truffle migrate --network development***" deploys the contracts on the local development network.

- Migration Execution: Run the "***truffle migrate***" command in the terminal. Truffle will execute the migration scripts one by one, deploying each contract in the defined order.

- Deployment Output: Truffle will display the deployment progress in the terminal, indicating the contract being deployed and its deployment status.

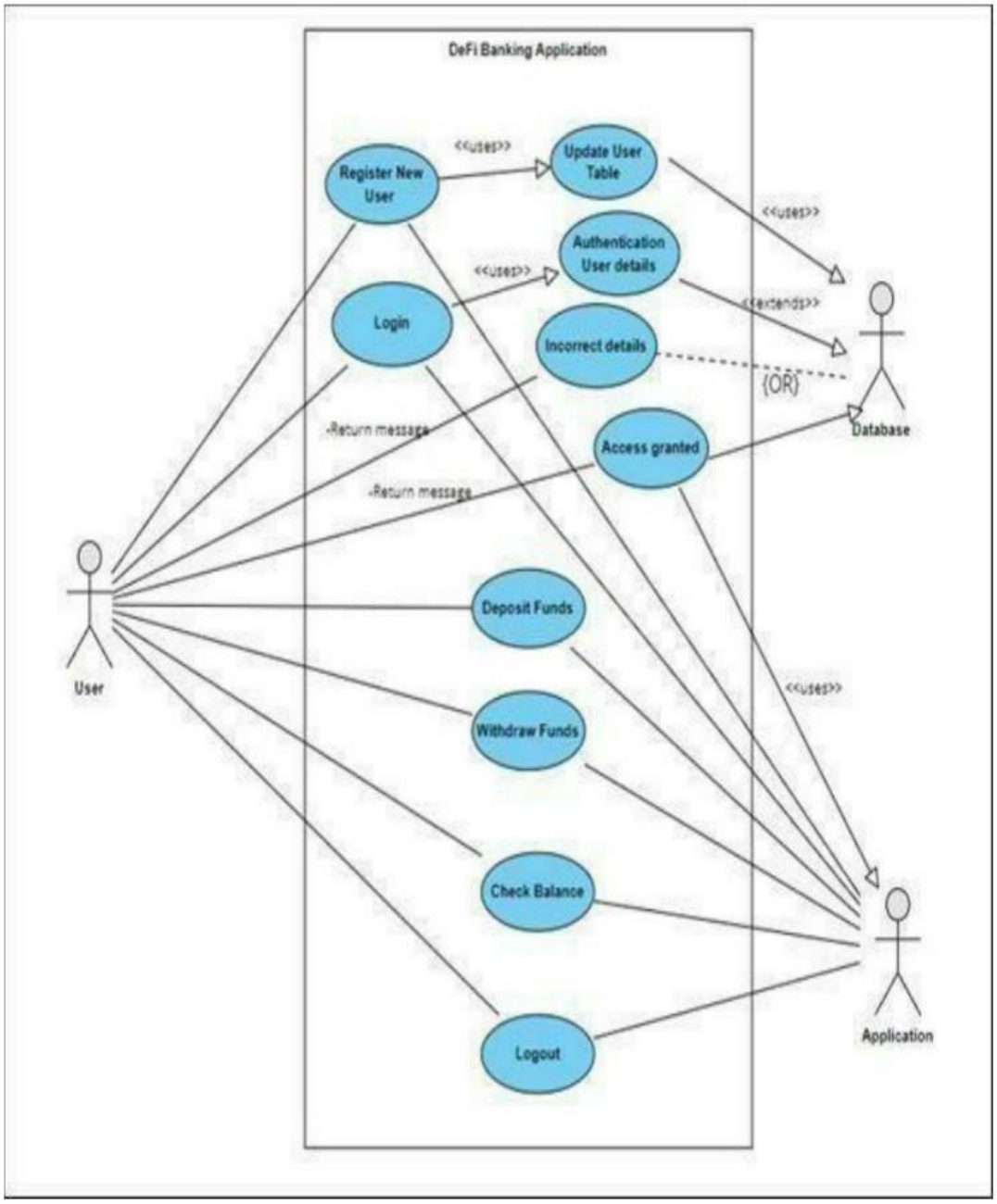
- Migration Status: Truffle keeps track of the deployed contracts using migration status files. It records the successfully deployed contracts, allowing for easy tracking of the deployment history.

- Contract Address: After successful deployment, Truffle displays the contract addresses for each deployed contract, allowing you to interact with them.

- Contract Interaction: Once the contracts are deployed, you can interact with them using Truffle's JavaScript console or by integrating them into your DApp or other applications.

The "***truffle migrate***" command simplifies the process of deploying smart contracts onto the Ethereum blockchain. It ensures the proper execution and tracking of deployment steps defined in the migration scripts, providing a straightforward way to manage and deploy your contracts.

4.2.2 Use Case Diagram:



User: The User is an actor in the system and represents an individual who interacts with the Application. The User can perform various actions such as logging in, depositing funds, withdrawing funds, and checking their balance.

Database: The Database is another actor in the system and is responsible for handling authentication. It stores user credentials and verifies the login credentials provided by the User.

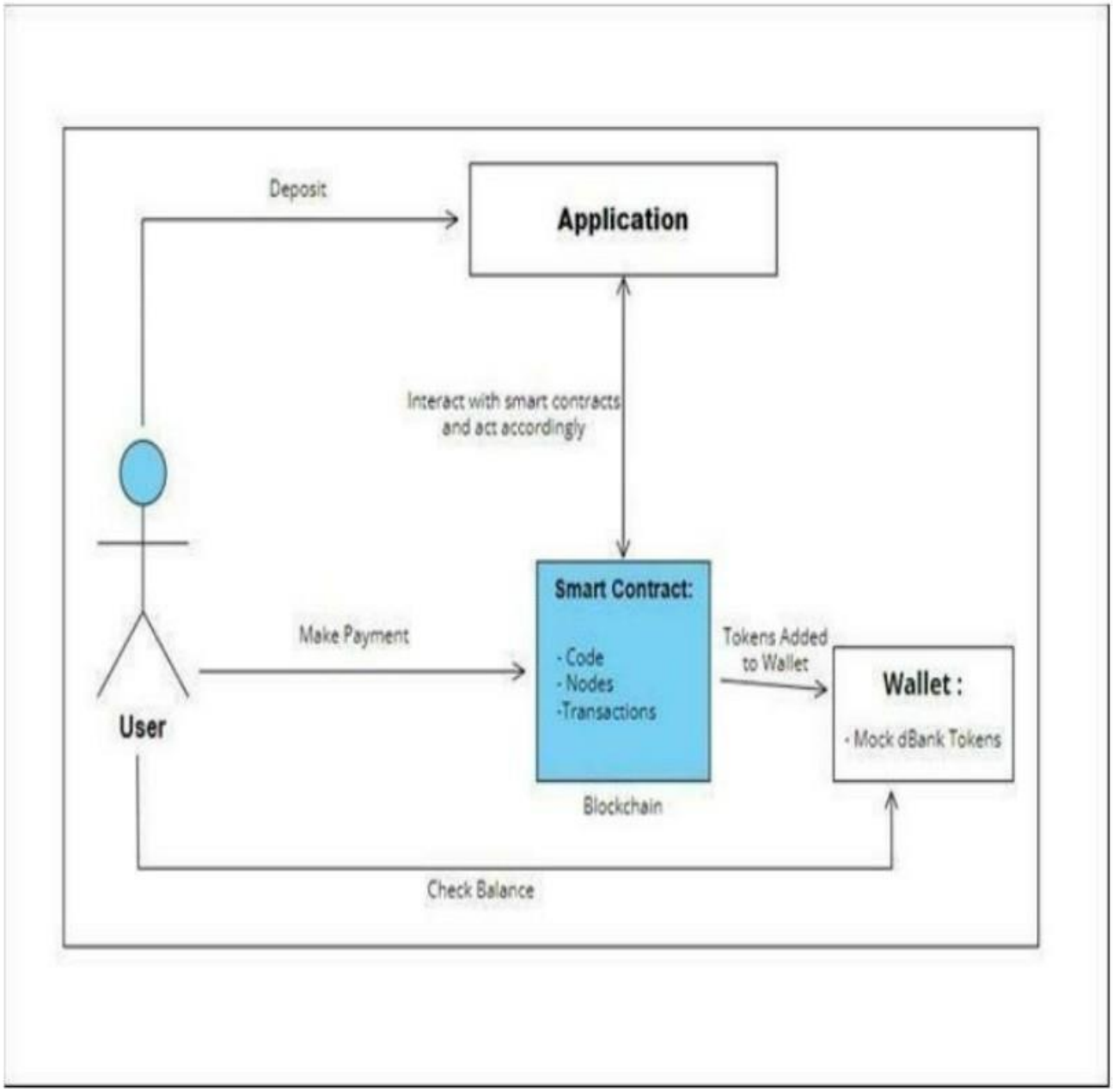
Application: The Application is the core component that facilitates user interactions and performs the necessary operations. It serves as an interface between the User and the Database, managing the different use cases.

Login through MetaMask: This use case allows the User to log in to the Application using the MetaMask wallet. MetaMask is a popular cryptocurrency wallet that enables secure interaction with decentralized applications (dApps) through a browser extension.

Deposit Funds: This use case enables the User to deposit funds into their account within the Application. The User initiates a deposit transaction, specifying the amount of funds they want to deposit. The Application handles the interaction with the Database to update the user's balance accordingly.

Withdraw Funds: This use case allows the User to withdraw funds from their account. The User initiates a withdrawal request, specifying the amount they want to withdraw. The Application verifies the availability of funds in the user's account and updates the balance accordingly.

Check Balance: This use case enables the User to view their account balance within the Application. The User can request the current balance, and the Application retrieves the information from the Database to display it to the User. In this use case diagram, the User interacts with the Application, and the Application communicates with the Database for authentication purposes. The Application handles the business logic and user interactions, while the Database manages user credentials and other relevant data.


Deposit
Application
Interact with smart contracts
and act accordingly
Smart Contract:
- Code
- Nodes
-Transactions
Make Payment
Tokens Added
to Wallet
Wallet :
- Mock dBank Tokens
User
Blockchain
Check Balance

The architecture and design of a DeFi banking app can vary depending on the platform and its specific requirements. However, there are common components and considerations involved in the system architecture and design.

The user interface (UI) is the front-end component that allows users to interact with the app. It should be user-friendly and provide access to various functionalities such as wallet management, lending, borrowing, asset exchange, and portfolio tracking.The backend infrastructure consists of servers, databases, and networking components that process and store data. It may include cloud services, APIs, and third-party integrations.

Blockchain integration is essential for DeFi banking apps, as they leverage blockchain technology. This integration allows users to interact with smart contracts, execute transactions, and access decentralized financial services.Smart contracts, which are self-executing contracts on the blockchain, automate transactions and financial operations within the app. They need to be carefully designed and audited for security and functionality. Wallet management ensures the secure storage and management of users' cryptocurrency holdings.

Integration with hardware wallets or secure key management solutions enhances security.

Lending and borrowing functionality requires a lending pool mechanism to facilitate transactions between users. Considerations include interest rate calculation, collateral management, and loan repayment mechanisms.Asset exchange functionality can be achieved through integration with decentralized exchanges (DEXs) and liquidity pools. This enables users to trade assets at competitive prices.

Data analytics and reporting features provide users with insights into their portfolio performance and transaction history, aiding decision-making and tracking financial activities.

Security and compliance are crucial. Robust security measures protect user funds, personal information, and smart contracts. Compliance with regulatory requirements, KYC, and AML procedures may also be necessary.

Scalability and performance considerations ensure that the architecture can handle increasing user demand. Load balancing, caching mechanisms, and database optimization contribute to optimal performance.

It's important to note that the actual architecture and design of a DeFi banking app may vary depending on the specific platform and project requirements. Consulting with experienced blockchain developers and architects is advisable to create a tailored solution for a DeFi banking app.

Command Prompt Implementation

Certainly! Let's explore in more detail the commands you mentioned: ***`truffle`, `npm`, and `ganache`***, and their usage in developing decentralized applications (DApps) with the Truffle framework.

1. Truffle:

Truffle is a widely-used development framework for building Ethereum-based DApps. It provides a comprehensive set of tools and utilities that simplify the process of developing, testing, and deploying smart contracts. Truffle automates tasks such as contract compilation, migration, testing, and asset management.

To install Truffle globally on your system, you can use the following command:

npm install -g truffle

By running this command, Truffle will be installed using npm (Node Package Manager), making it available as a command-line tool. Once installed, you can use the ***`truffle`*** command to interact with your DApp project. Truffle offers various subcommands, including ***`init`, `compile`, `migrate`, and `test`***, which enable you to perform different development tasks efficiently.

2. NPM (Node Package Manager):

NPM is a package manager specifically designed for managing JavaScript libraries and packages for Node.js applications. It simplifies dependency management and package installation for your projects. Truffle itself is installed and managed using NPM.

To initialize a new project and manage its dependencies with NPM, navigate to your project directory in the command prompt and use the following command:

npm init

This command guides you through a series of prompts to set up your project and generate a ***`package.json`*** file. The ***`package.json`*** file lists the project's dependencies and provides configuration details. You can use NPM to install additional packages or libraries required for your DApp development.

For example, if you need a specific Ethereum client library, you can use the following command:

npm install ethereumjs-util

This command installs the ***`ethereumjs-util`*** package and updates the ***`package.json`*** file accordingly.

3. Ganache:

Ganache is a personal Ethereum blockchain that allows you to create a local development environment for testing your DApps. It provides a simulated blockchain network with predefined accounts and test Ether (ETH) for development and testing purposes.

To install Ganache globally, you can use the following command:

npm install -g ganache-cli

Once installed, you can start a Ganache blockchain instance by running the following command in your command prompt:

ganache-cli

This command starts a local Ethereum network with a set of accounts and their associated private keys, along with preloaded test Ether balances. Ganache also provides HTTP and WebSocket endpoints for interacting with the blockchain from your DApp.

By utilizing Truffle, npm, and Ganache, developers can effectively develop, test, and deploy smart contracts for their DApps. Truffle simplifies the development workflow, npm manages dependencies, and Ganache provides a local blockchain environment for testing and debugging.

Chapter 5: Implementation

5.1 Overview of technology used.

Blockchain is a decentralised ledger that is used to conduct transactions and securely exchange virtual currency. The most recent version of the encrypted ledger is available to every member of the network, allowing them to validate fresh transactions. The blockchain ledger is a database of all previously completed Bitcoin transactions. In essence, it's a distributed database that keeps track of ever-expanding tamper-proof data structure blocks that include batches of individual transactions. A linear and chronological order is maintained as the completed blocks are added. Each block has a timestamp and an information link that refers to the block before it. Every user can join to the peer-to-peer, permission-less Bitcoin network and send new transactions to be verified and added to existing blocks.

In a 2008 research paper that was posted to a cryptography listserv, Satoshi Nakamoto explained the layout of the Bitcoin digital money. Nakamoto's idea provided a long-needed solution for cryptographers and created the groundwork for digital currency. This chapter describes the idea behind blockchain technology, its benefits, and how Bitcoin functions. It makes an effort to emphasize how important Blockchain is to the development of the Internet of Things (IoT), banking, and financial institutions. Although the cost of cybercrime doubled between 2013 and 2015, a sizable amount of it remains uncovered. According to Gartner research, by 2019 the cost of cybercrime is predicted to reach $2 trillion.

At the IBM Security Summit, IBM CEO Ginni Rometty declared that cybercrime poses the greatest danger to all businesses worldwide. Approximately two years ago, a fraud at China's Qingdao port cost Standard Chartered about $200 million. Blockchain-based technology is being used by banking and organisations to lower risk and stop cybercrime. For instance, Nasdaq has declared its intention to introduce Blockchain-based digital ledger technology, which will enhance their capacity for equity management. Together with DBS Group, Standard Chartered is creating an electronic invoice ledger that uses Blockchain. The Bitcoin Core client stores the Blockchain metadata in Google's Level DB.

Blockchain can be pictured as a vertical stack of blocks that are stacked one on top of the other, with the bottom block serving as the stack's foundation. The individual blocks are connected by links and make references to the preceding blocks in the chain. The secure hash algorithm (SHA-256) cryptographic hash algorithm is used to create a hash on the block header that uniquely identifies each individual block. A block will always have one parent, but it may have several children, all of whom will refer to the same parent block and hence have the same hash in the previous block hash field. Every block has a hash of its parent block in its own header, and the series of hashes connecting each block to its parent block forms a large chain pointing to the parent block called Genesis block.

HTML (Hypertext Markup Language) is the standard markup language used for creating web pages and web applications. It provides the structure and content of a webpage, defining the elements, layout, and semantic meaning of the page's content. HTML is the backbone of the World Wide Web, allowing developers to create interactive and accessible web pages.

Key features and characteristics of HTML:

- Structure and Elements
- Tags and Attributes
- Semantic Markup
- Links and Navigation

CSS, a style sheet language called Cascading Style Sheets is used to describe the presentation and aesthetic appeal of an HTML or XML document. It is a crucial part of creating websites and is used to manage the design, formatting, and organization of web pages.

Key features and concepts related to CSS:

- Selectors
- Properties and Values
- Stylesheets

React.js, an open-source JavaScript package called React.js is used to create user interfaces (UI) for online applications. As the application's data changes, it enables developers to efficiently design reusable UI components and update and render them. React.js is well-known for being used to create dynamic and interactive online apps since it adheres to a component-based architecture.

Key features and concepts related to React.js:

- Components
- Virtual DOM (Document Object Model)
- State Management:
- React Router

Node.js code can run outside of a web browser thanks to the open-source server-side runtime environment Node.js. It is created using the V8 JavaScript engine from Chrome and offers an event-driven, non-blocking I/O mechanism, making it extremely effective and scalable for creating server-side applications.

- JavaScript on server
- Event-driven Architecture
- Asynchronous
- Non-blocking I/O

Web3.js, a JavaScript package called Web3.js enables programmers to communicate with the Ethereum blockchain and create decentralized apps (DApps). In addition to reading from and writing to smart contracts, managing user accounts and processing transactions, it offers a set of APIs that make it possible to communicate with the Ethereum network.

Key features and concepts related to Web3.js:

- Ethereum Interaction
- Smart Contract Integration
- Account Management
- Transaction Handling
- Ethereum Standards and Utilities

Solidity, a high-level programming language called Solidity was created expressly for creating smart contracts for the Ethereum network. It is ideal for creating decentralized applications (DApps) and integrating complicated logic into smart contracts since it is statically typed and supports object-oriented ideas.

Key features and characteristics of Solidity:

- Smart Contract Development
- Ethereum Compatibility
- Object-Oriented Design

Truffle, a building, testing, and deploying smart contracts are made simpler by the widely used Ethereum development framework truffle. It offers a collection of utilities and tools that simplify the development process.

- Project Structure
- Smart Contract Compilation
- Contract Migration
- Testing Framework
- Integration with External Tools

MetaMask, users can interact with the Ethereum blockchain and decentralised applications (DApps) directly from their web browser by using the popular browser plugin and cryptocurrency wallet MetaMask. In order to manage Ethereum accounts, sign transactions, and safely store private keys, it acts as a bridge between the browser and the Ethereum network. Key features and functionalities of MetaMask:

- Ethereum Wallet
- Browser Integration
- Secure Key Management
- DApp Interaction

5.2 Implementation details of modules

The project can be divided roughly into two modules:

Frontend:

- The application's user interface and user experience are handled by the front-end module.
- The components that users interact with, as well as the design and layout, are included.
- utilizes web technologies such as HTML, CSS, and JavaScript to implement client-side logic.
- reaches out to the backend to retrieve data and start transactions.
- gives users the ability to register, log in, manage their accounts, view their transaction history, and access other DeFi services.

Backend:

- The DeFi banking application's backend module manages the server-side functionality and administrative tasks.
- handles queries that are received from the front end.
- Manages user authentication and authorization.
- Utilizes programming languages such as Node.js.
- implements business logic, including staking, yield farming, transaction management, token lending and borrowing, and connection with third-party services and APIs.

The purpose of this project is to build a complete blockchain application, specifically a decentralized finance (DeFi) app, using Ethereum, Solidity, Web3.js, and Truffle. The project entails setting up a blockchain, creating Ethereum smart contracts, writing tests, and developing a client-side website that communicates with the blockchain. Traditional web applications face limitations in securely storing value, but blockchain technology overcomes these limitations by offering immutable smart contracts and public ledgers. In this project, we aim to address these limitations by building a DeFi app that allows users to deposit the stablecoin "Dai" and earn a new token called "Dap". This aligns with the popular trend in DeFi known as yield farming, where users are incentivized to use the application and earn tokens without the need to make purchases.

To achieve this, we will create a website using React.js and Solidity programming language. The website will serve as the client-side interface, interacting with the blockchain. We will develop a smart contract that facilitates the staking of tokens and rewards users with new tokens. In order to complete the development, we will install necessary dependencies such as Node.js and Ganache.

The initial step involves creating smart contracts for two ERC20 tokens: a mock version of the stable cryptocurrency "Dai" and a fictional token called "Dap". These tokens will be utilized within a digital bank, enabling users to deposit Dai and earn Dap as interest. Furthermore, we will implement a smart contract for a token farm, allowing users to earn interest and receive new tokens. To deploy the smart contracts onto the blockchain, we will utilize Truffle. We will create migration files to facilitate the deployment process and generate JSON files that contain a description of the smart contract functionality. The "truffle migrate" command will be executed to deploy the smart contracts, accompanied by a gas fee payment in Ether. Additionally, we will leverage the Truffle console, a JavaScript runtime environment, to interact with the blockchain and retrieve smart contracts using the "await" keyword, which returns a promise requiring resolution. Thorough testing of smart contracts is crucial to ensure their proper functioning on the blockchain. We will employ JavaScript libraries such as Chai and Mocha to write comprehensive tests. These tests will cover various scenarios, including token staking, token transfers, and the maintenance of staking balances and stakers using mappings and arrays in Solidity. Throughout the development process, we will encounter specific concepts and techniques inherent to Solidity. Solidity allows us to retrieve the value of mappings using functions, enabling us to determine ownership and token amounts. Additionally, we will utilize web3 utils to convert values into human-readable decimals, facilitating interactions with smart contracts. By undertaking this project, we aim to equip developers with the necessary knowledge and skills to create and test smart contracts for token staking, transfers, and the management of staking balances. The project report provides a comprehensive overview of the development process, highlighting the significance of Ethereum, Solidity, Web3.js, and Truffle in building DeFi applications.

Chapter 6: Conclusion and Future Enhancements

Conclusion

Every industry now considers information technology to be a crucial innovation. In order to upend the status quo in a leadership position, organisations or teams that can use technology correctly and successfully play a vital role. People who do not adapt to technology do not live. We see the Blockchain technology as a driver for new applications in both the financial and non-financial sectors, including supply chains, manufacturing, and banking. By enabling safe trust frameworks, fostering agile value chain production, and fostering closer interaction with technologies like cloud computing and IoT, blockchain can play a critical role in revolutionising the digitization of industries and applications.

The researchers have shown their ability to use professional engineering principles, a DevOps approach to iterative development and management, and integration of cyber security, distributed computing, and blockchain technologies by creating a cloud-based application called banking. We believe that the banking industry is just one of many examples that show how Blockchain may be transformational.

Future Enhancements

Future enhancements for a DeFi banking application could include:

- Interoperability between blockchain networks that is cross-chain: This enables users to access a greater variety of DeFi protocols and assets.

- Implementing layer 2 solutions or integrating with scalable blockchain platforms will strengthen the application's ability to handle growing transaction volumes.

- Added privacy features: Implementing cutting-edge privacy-preserving technology, including secure multiparty computing or zero-knowledge proofs, to safeguard user information and transaction privacy.

- Interaction with traditional banking systems: Enabling interoperability between DeFi and traditional banking services, bridging the gap between the two, by facilitating interaction with legacy financial systems.

- Decentralized loan and insurance services will be added to the application's capabilities, giving customers access to more complete financial solutions.

- Enhancing the user interface and shortening the user onboarding procedures would improve the user experience and make the application more intuitive, user-friendly, and available to a larger range of users.

To meet the ever-changing needs of our users, we have a range of future enhancements planned for the DeFi Banking App:

- Improved User Interface: We are committed to delivering an intuitive and user-friendly experience. Our upcoming enhancements will focus on refining the interface, making it more visually appealing and easier to navigate. We will also enhance the application's responsiveness and speed.

- Expanded Cryptocurrency Selection: Recognizing the importance of offering diverse investment options, we plan to expand the range of supported cryptocurrencies and tokens. This will enable users to access and manage a broader array of digital assets through our app.

- Seamless Integration with Decentralized Exchanges: Our goal is to integrate with popular decentralized exchanges (DEXs), facilitating easy access to larger liquidity pools and enabling users to execute trades directly within the DeFi Banking App. This integration will enhance the overall trading experience.

- Advanced Portfolio Management: We understand the need for comprehensive portfolio management. tools. In our future enhancements, we will introduce features such as performance tracking, portfolio

- Compatibility Across Multiple Blockchains: To enhance accessibility, we are exploring support for multiple blockchain networks. This will allow users to interact with different blockchain ecosystems and access decentralized financial services across various chains, promoting interoperability.

- Integration of Additional DeFi Protocols: We are actively working on integrating more decentralized finance protocols into our platform. Through collaborations with reputable DeFi projects, we aim to expand the range of lending, borrowing, yield farming, and staking opportunities available to our users.

- Enhanced Security Features: Security is a top priority for us. In our future enhancements, we will implement additional security measures such as advanced encryption, two-factor authentication, and robust fraud detection mechanisms. These measures will ensure the utmost protection for user funds and personal information.

Our dedication to delivering cutting-edge DeFi banking experience drives us to continuously innovate. We are committed to providing users with the best possible tools and services to navigate the decentralized financial landscape effectively.

References

[1] Makarov I, Schoar A. (2022) Cryptocurrencies and decentralized finance (DeFi). National Bureau of Economic Research. No. 30006

[2] Alidin, A. A., Ali-Wosabi, A. A. A., & Yusoff, Z. (2018, November). Overview of blockchain implementation on Islamic finance: Saadiqin experience. In 2018 Cyber Resilience Conference (CRC) (pp.

1–2). IEEE.

[3] Schär F (2021) Decentralized finance: on blockchain-and smart contract-based financial markets. FRB of

St Louis Review Federal Reserve Bank of St Louis Review 103(2):153–174

[4] Bhuvana R, Aithal PS (2020) RBI distributed ledger technology and blockchain-a future of decentralized

India. Int J Manage Technol Social Sci (IJMTS) 5(1):227–237

[5] Salami I (2020) Decentralised finance: the case for a holistic approach to regulating the crypto industry. J Int Bank Financial Law 35(7):496–499

[6] Avgouleas E, Kiayias A. (2020) The architecture of decentralised finance platforms: a new open finance paradigm. Edinburgh School of Law Research Paper. No.16

[7] Ozili PK (2022) Decentralized finance and cryptocurrency activity in Africa. CSEF 109:1–13

[8] Arantes, G. M., D'Almeida, J. N., Onodera, M. T., Moreno, S. M. D. B. M., & Almeida, V. D. R. S.

(2018, July). Improving the process of lending, monitoring, and evaluating through Blockchain Technologies: An application of Blockchain in the Brazilian Development Bank (BNDES). In 2018 IEEE

International Conference on Internet of Things (iThings) and IEEE Green Computing anCommunications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) (pp. 1181– 188). IEEE.

[9] Luu, Loi and Velner, Yaron. “KyberNetwork: A Trustless Decentralized Exchange and Payment Service.”2017; https://whitepaper.io/document/43/kybernetwork- whitepaper.

[10] R. T. Ainsworth and A. B. Shact, “Blockchain (Distributed Ledger Technology) Solves VAT Fraud”,

Boston Univ. School of Law, Law and Economics Research Paper No. 16-41. Oct. 17, 2016. [Online] Available at SSRN: https://ssrn.c

[11] Shao-Ku Tien, Yu-Ting Wang, Yun-Zhan Cai, Meng-Hsun Tsai. Maximizing the Time Value of Cryptocurrency in Smart Contracts with Decentralized Money Markets 2020 IEEE International Conference on Blockchain (Blockchain)

[12] Peterson K. Ozili1Journal of Banking and Financial Technology https://doi.org/10.1007/s42786-022-00044-x Decentralized finance research and developments around the world Received: 12 October 2021 /

Accepted: 8 August 2022.

[14] Isabell M. Welpe, Eva Andrea Meyer Decentralized Finance – A Systematic Review and Research

Directions ECIS 2022 Research Papers ECIS 2022 Proceedings 6-18-2022 [17] Jessica Steinberger, Anna Sperotto, Mario Golling, and Harald Baier How to exchange security events? Overview and evaluation of formats and protocols 2015.

[18] https://cybersecurity.att.com/blogs/security-essentials/blackhat-2016- threat-intelligence-deja-vu

[19] US Joint Chiefs of Staff. Joint Publication 2-0 Joint Intelligence. Jt Publ. 2013;(October):144

[20] Ali Ahmed, Tabassum Ara, and Akash Kumar Threat Intelligence Sharing: A Survey 2018.

[21] AlienVault Open Threat Exchange (OTX) https://cybersecurity.att.com/open-threat-exchange

[22] Elevating What a TIP Can Be – The ThreatQ Platform https://www.threatq.com/elevating-tip-threatq-platform/

[23] ThreatQ https://www.threatview.ca/

[24] Muyowa Mutemwa, Jabu Mtsweni, and Njabulo Mkhonto Developing a cyber threat intelligence sharing platform for South African organisations 2017.

[25] Onur Catakoglu, Marco Balduzzi and Davide Balzarotti Automatic Extraction of Indicators of Compromise for Web Applications

[26] How to identify IoCs https://www.crowdstrike.com/cybersecurity101/indicators-of-compromise/

[27] The OpenIOC Framework. http://www.openioc.org

[28] OpenIOC – Sharing Threat Intelligence

https://www.darknet.org.uk/2016/06/openioc-sharing-threatintelligence/

[29] M. Faiella, G. G. Granadillo, Ibéria Medeiros, Rui Azevedo, and S. Zarzosa Enriching Threat Intelligence Platforms Capabilities 2019.

[30] A problem shared is a problem halved: A survey on the dimensions of collective cyber defense through security information sharing Florian Skopik, Giuseppe Settanni and Roman Fiedler

[31] Thomas D. Wagner, Esther Palomar, Khaled Mahbub, and Ali E. Abdallah A Novel Trust Taxonomy for Shared Cyber Threat Intelligence 2018.

[32] Exploring the opportunities and limitations of current Banking systems.

[33] Sander, T, and Halpern, J.: Ux aspects of threat information sharing platforms: An examination & lessons learned using personas. In: 2nd ACM Workshop on Information Sharing and Collaborative Security, pp. 51-59. ACM, New York (2015)

[34] Ibid at 29

[35] Ibid at 29

[36] Alessandra de Melo e Silva, João José Costa Gondim, Robson de Oliveira Albuquerque and Luis Javier

García Villalba A Methodology to Evaluate Standards and Platforms within Cyber Threat Intelligence

[37] Threat Intelligence Sharing Platforms: An Exploratory Study of Software Vendors and Research Perspectives Clemens Sauerwein, Christian Sillaber, Andrea Mussmann, and Ruth Bre

DrillBit

The Report is Generated by DrillBit Plagiarism Detection Software

Submission Information

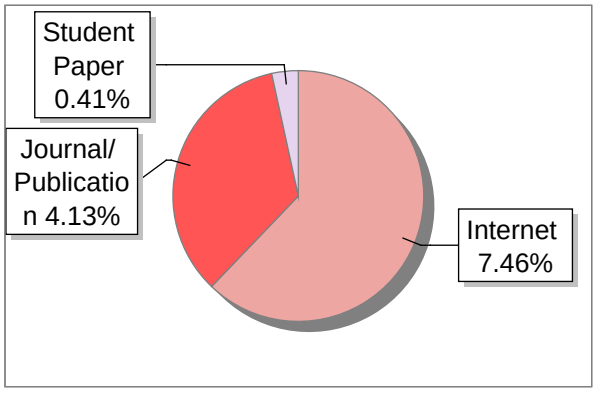
Author Name	Abhinav Chettri
Title	“Decentralized Finance (DeFi) Banking Applicati..
Paper/Submission ID	771856
Submission Date	2023-06-12 15:18:04
Total Pages	47
Document type	Project Work

Result Information

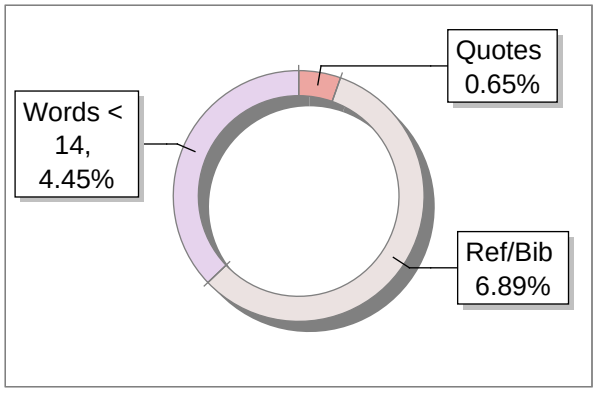
Similarity **12 %**

1 10 20 30 40 50 60 70 80 90

Sources Type



Report Content



Exclude Information

Quotes	Not Excluded
References/Bibliography	Not Excluded
Sources: Less than 14 Words Similarity	Not Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

A Unique QR Code use to View/Download/Share Pdf File

DrillBit

DrillBit Similarity Report

12	58	B
SIMILARITY %	MATCHED SOURCES	GRADE

A-Satisfactory (0-10%)
B-Upgrade (11-40%)
C-Poor (41-60%)
D-Unacceptable (61-100%)

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
1	www.readbag.com	1	Internet Data
2	www.researchgate.net	1	Internet Data
3	www.slideshare.net	1	Internet Data
4	qdoc.tips	1	Internet Data
5	qdoc.tips	<1	Internet Data
6	Analysis of Blockchain technology pros, cons and SWOT by Niranjanamurthy-2018	<1	Publication
7	www.mdpi.com	<1	Publication
8	www.theseus.fi	<1	Publication
9	moam.info	<1	Internet Data
10	docplayer.net	<1	Internet Data
11	IEEE 2020 IEEE International Conference on Blockchain (Blockchain) - Rhodes Is	<1	Publication
12	dayanandasagar.edu	<1	Internet Data
13	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper

14	docshare.tips	<1	Internet Data
15	ipindia.gov.in	<1	Publication
16	IEEE 2018 International Conference On Cyber Situational Awareness, D	<1	Publication
17	dspace.knust.edu.gh	<1	Publication
18	Distributed Security Framework for Reliable Threat Intelligence Sharin by Preuveneers-2020	<1	Publication
19	Discussion on The performance of noise meters in terms of the primary standard by Turka-1937	<1	Publication
20	docplayer.net	<1	Internet Data
21	pure.mpg.de	<1	Publication
22	qdoc.tips	<1	Internet Data
23	research.stlouisfed.org	<1	Publication
24	slate.com	<1	Internet Data
25	Thesis Submitted to Shodhganga Repository	<1	Publication
26	eprints.lmu.edu.ng	<1	Internet Data
27	researchspace.csir.co.zadspace	<1	Internet Data
28	Advanced Networked Technologies in the UK Voluntary Sector by Eleano-2001	<1	Publication
29	docplayer.net	<1	Internet Data
30	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper
31	www.slideshare.net	<1	Internet Data

32	Analyzing Consumer Protection for Gamblers Across Different Online Gambling Oper by Bonello-2017	<1	Publication
33	laimardownpho.ddns.net	<1	Internet Data
34	Article Published by AmeriQuests, Narrative, Law and Society Journal - www.ameriquests.org	<1	Publication
35	Central Bank Digital Cash and Cryptocurrencies Insights from a New B, by Masciandaro, Donato- 2018	<1	Publication
36	docplayer.net	<1	Internet Data
37	eproceeding.undiksha.ac.id	<1	Internet Data
38	www.devexperts.com	<1	Internet Data
39	www.devexperts.com	<1	Internet Data
40	byjus.com	<1	Internet Data
41	mdpi.com	<1	Internet Data
42	www.gemini.com	<1	Internet Data
43	adoc.pub	<1	Internet Data
44	egyankosh.ac.in	<1	Publication
45	Jawaharlal Nehru Technological University, Anantapuram Thesis Published in inflibnet - www.inflibnet.ac.in	<1	Publication
46	thecyberwire.com	<1	Internet Data
47	adage.com	<1	Internet Data
48	bambooinnovator.com	<1	Internet Data
49	brandbaron.com	<1	Internet Data

50	documents.mx	<1	Internet Data
51	moam.info	<1	Internet Data
52	moam.info	<1	Internet Data
53	publications.drdo.gov.in	<1	Publication
54	qdoc.tips	<1	Internet Data
55	slate.com	<1	Internet Data
56	www.dx.doi.org	<1	Publication
57	www.netce.com	<1	Internet Data
58	www.netce.com	<1	Internet Data