Tracking Public Infrastructure and Toll Payments Using Block Chain

- **INTRODUCTION**
  - **Project Overview**

Public infrastructure and toll payment management is a critical aspect of urban planning and transportation systems. Traditional systems often face issues related to transparency, security, and efficiency. This project aims to develop a blockchain-based solution to address these challenges and provide a robust platform for tracking public infrastructure and toll payments

- **Purpose**

Transparency and Trust:
Blockchain provides a transparent and tamper-proof ledger of all transactions and changes related to public infrastructure and toll payments. This transparency builds trust among stakeholders, including government agencies, citizens, and private sector entities, as they can independently verify records, ensuring that the data is reliable and trustworthy.

Enhanced Accountability:
By utilizing blockchain, public infrastructure management and toll collection become more accountable. All actions and transactions are recorded on the blockchain, making it easier to trace and audit activities related to infrastructure maintenance and toll payment collection. This, in turn, reduces the potential for fraudulent activities.

- **EXISTING PROBLEM**
  - **Existing problem**

Scalability: Blockchain networks, like Bitcoin and Ethereum, have struggled with scalability issues. Tracking a large number of public infrastructure assets and toll payments on a blockchain could lead to slow transaction processing times and high fees.

Privacy Concerns: While blockchain offers transparency, it might not be suitable for handling sensitive personal information related to toll payments. Striking a balance between transparency and data privacy is crucial.

Interoperability: Different regions and authorities may use various blockchain platforms or systems, making it challenging to achieve seamless interoperability for tracking infrastructure and toll payments across borders.

Adoption and Integration: The adoption of blockchain technology in the public sector is often slow due to regulatory hurdles, budget constraints, and the need for legacy systems to integrate with the new technology.

Smart Contract Security: Smart contracts, which could be used for automatically processing toll payments, are not immune to bugs or vulnerabilities. Security concerns could lead to incorrect transactions or exploitation.

- **Problem Statement Definition**

In contemporary transportation systems, the efficient management of public infrastructure and toll payments is crucial for urban development and economic growth. Traditional methods of tracking these activities are often beset with challenges such as lack of transparency, security vulnerabilities, and operational inefficiencies. To address these issues, there is a growing interestin leveraging blockchain technology to create a decentralized, transparent, and secure system for tracking public infrastructure and toll payments.

However, several critical challenges hinder the effective implementation of blockchain solutions in this context. Scalability concerns limit the ability to handle a large volume of transactions in real-time. Privacy issues arise regarding the secure storage and management of sensitive user data associated with toll payments. Interoperability problems emerge due to variations in blockchain platforms and technologies used by different regions and authorities. Additionally, theintegration of blockchain with existing systems and the development of secure smart contracts for automatic payment processing pose significant challenges.

- **IDEATION AND PROPOSED SOLUTION**
  - **Empathy Map Canvas**

User Persona: Describe the specific user or stakeholder you want to empathize with, e.g., Commuter, Government Official, Toll Booth Operator, etc.

Says: What do they say about public infrastructure and toll payments? What are their concerns, comments, and opinions? Capture direct quotes or statements if possible.

Thinks: What might be going on in their minds related to this topic? What thoughts, worries, or expectations do they have?

Feels: What emotions are associated with public infrastructure and toll payments for this user? Are they frustrated, satisfied, anxious, or something else?

Does: What actions or behaviors are typical of this user when dealing with public infrastructure and toll payments? How do they navigate the system?

Pain Points: List the challenges, inconveniences, or obstacles they encounter when using public infrastructure or making toll payments.

Gains: What do they hope to achieve or gain from a more efficient and user-friendly public infrastructure and toll payment system?

Needs: What are their fundamental needs in relation to public infrastructure and toll payments? What improvements or solutions would address their concerns?

Expectations: What do they expect from the government, toll operators, or the overall system when it comes to public infrastructure and toll payments?

Influences: What external factors, such as regulations, social norms, or media, affect their perspective on public infrastructure and toll payments?

Environment: Describe the physical or digital environment in which they interact with public infrastructure and toll payment systems.

Devices/Tools: Mention the devices or tools they use to access information or pay tolls. For example, smartphones, cash, credit cards, etc.

By filling out this empathy map canvas for various user personas and stakeholders, you can gain a deeper understanding of their needs, pain points, and expectations. This information can guide the design and improvement of public infrastructure and toll payment systems to better meet the needs of all involved parties.

- **Ideation and Brainstorming**

Smart Tolling Systems:

Implement smart toll collection systems that use RFID technology or license plate recognition for seamless and contactless payments.
Explore dynamic pricing models that adjust toll rates based on traffic congestion, encouraging off-peak travel.

Mobile Apps and Digital Wallets:

Develop a mobile app that allows users to link their payment methods and manage toll payments. Integrate digital wallet options for more convenient transactions.

Blockchain for Transparency:

Use blockchain technology to enhance transparency and security in toll payments, reducing fraud and errors.

Congestion Pricing:

Consider congestion pricing in high-traffic areas, where toll rates vary based on the time of day and demand.

Public-Private Partnerships:

Collaborate with private companies to finance, build, and operate toll roads and bridges. This can alleviate the burden on public funds.

Environmental Initiatives:

Offer discounts or incentives for electric vehicles (EVs) to promote sustainability.
Invest in EV charging stations at toll plazas to encourage the adoption of electric vehicles.

Multi-Modal Integration:

Create integrated ticketing and payment systems that cover not only tolls but also public transportation, parking, and more.
Develop mobile apps that provide real-time information on public transport options alongside toll information.

User-Friendly Interfaces:

Design user-friendly toll booths and online interfaces to reduce wait times and user frustration.
Employ artificial intelligence (AI) and chatbots to provide instant assistance to users.

Infrastructure Upgrades:

Invest in modern infrastructure and maintenance to ensure toll roads and bridges are safe and efficient.
Explore using smart materials for roads that can self-repair and reduce maintenance costs.

Community Input:

Solicit input from the local community to understand their needs and concerns regarding infrastructure and tolls.
Hold public forums and surveys to gather feedback on proposed projects.

Public-Private Financing Models:

Explore revenue-sharing models with private companies, where a portion of toll revenue is reinvested in the community or used for public projects.

Incentive Programs:

Develop loyalty programs that reward frequent toll road users with discounts or other benefits.

Real-Time Traffic Data:

Provide real-time traffic information and alternative route suggestions to help drivers make informed decisions.

Toll Credits and Subsidies:

Implement toll credits or subsidies for low-income individuals or those who frequently use the toll roads for work.

Alternative Energy Sources:

Explore renewable energy sources to power toll plazas and infrastructure to reduce environmental impact.

Cybersecurity Measures:

Strengthen the cybersecurity of toll collection systems to protect user data and financial transactions.
Remember that successful implementation often requires a combination of these ideas and may vary depending on the specific needs and resources of your region. Additionally, involving various stakeholders, including government agencies, local communities, and private sector partners, can lead to more comprehensive and sustainable solutions.

- **Requirement Analysis**
  - **Functional Requirements**

Transcript Generation: Create a feature for institutions to generate and issue digital transcripts directly from the system, with the transcript being cryptographically signed to ensure its authenticity.

Data Encryption: Implement robust encryption methods to protect data at rest and duringtransmission to guarantee data confidentiality.

Audit Trail: Maintain a comprehensive audit trail that records all data access and changes, enhancingtransparency and accountability.

User Support and Helpdesk: Establish a user support system, including a helpdesk or customersupport feature, to assist users with any issues or inquiries.

Access Control and Permissions:

Define user roles and permissions to restrict access to sensitive data and operations.
Ensure that government authorities have the necessary privileges to manage infrastructure assets and toll collection.

Data Synchronization:

Implement a real-time or periodic data synchronization mechanism to update asset and payment data from external sources (e.g., toll booths) to the blockchain.

Data Integrity and Security:

Implement cryptographic techniques to ensure data integrity and protect against unauthorized access.
Regularly perform security assessments and vulnerability testing.

Mobile and Web Interfaces:

Develop user-friendly web and mobile applications for users to access the system. Provide an intuitive dashboard for government authorities to manage infrastructure

- **Non-Functional Requirements**

Performance: The system should be highly responsive, with low latency, to ensure quick access to educational records and verification processes. It should support a large number of concurrent usersand handle peak loads efficiently.

Scalability: The system should be designed to scale both vertically and horizontally to accommodatethe growing number of users and increasing data volumes as the user base expands.

Security: Implement stringent security measures, including encryption, access control, and regular security audits, to protect data from unauthorized access and cyber threats. Ensure compliance withindustry-standard security practices.

Reliability: The system should be highly reliable, with minimal downtime, to ensure continuousaccess to educational records. Implement redundancy and failover mechanisms to minimize disruptions.

Usability: The user interface should be intuitive and user-friendly, requiring minimal training forusers to navigate the system effectively. It should also be accessible to users with disabilities.

Compliance: Ensure compliance with relevant data privacy and protection regulations, such asGDPR, HIPAA, or other regional data laws, depending on the scope of the project.

Interoperability: The system should be able to integrate with other education systems and databases,ensuring data can be shared seamlessly with authorized parties.

Data Backup and Recovery: Regularly back up educational records to prevent data loss and ensurequick recovery in case of system failures or data corruption.

Load Testing: Perform load testing to ensure the system can handle heavy usage withoutperformance degradation. This testing should identify the system's maximum capacity.

Audit and Logging: Maintain comprehensive audit logs of all system activities and access to data foraccountability and traceability.

Data Retention Policies: Establish data retention and data disposal policies in compliance withrelevant regulations, detailing how long data will be stored and how it will be securely deleted.

Mobile Responsiveness: The system should be mobile-responsive, allowing users to access andmanage their educational records on various mobile devices.

User Support: Offer robust user support with clear response times, providing assistance to users incase they encounter any issues or have inquiries.

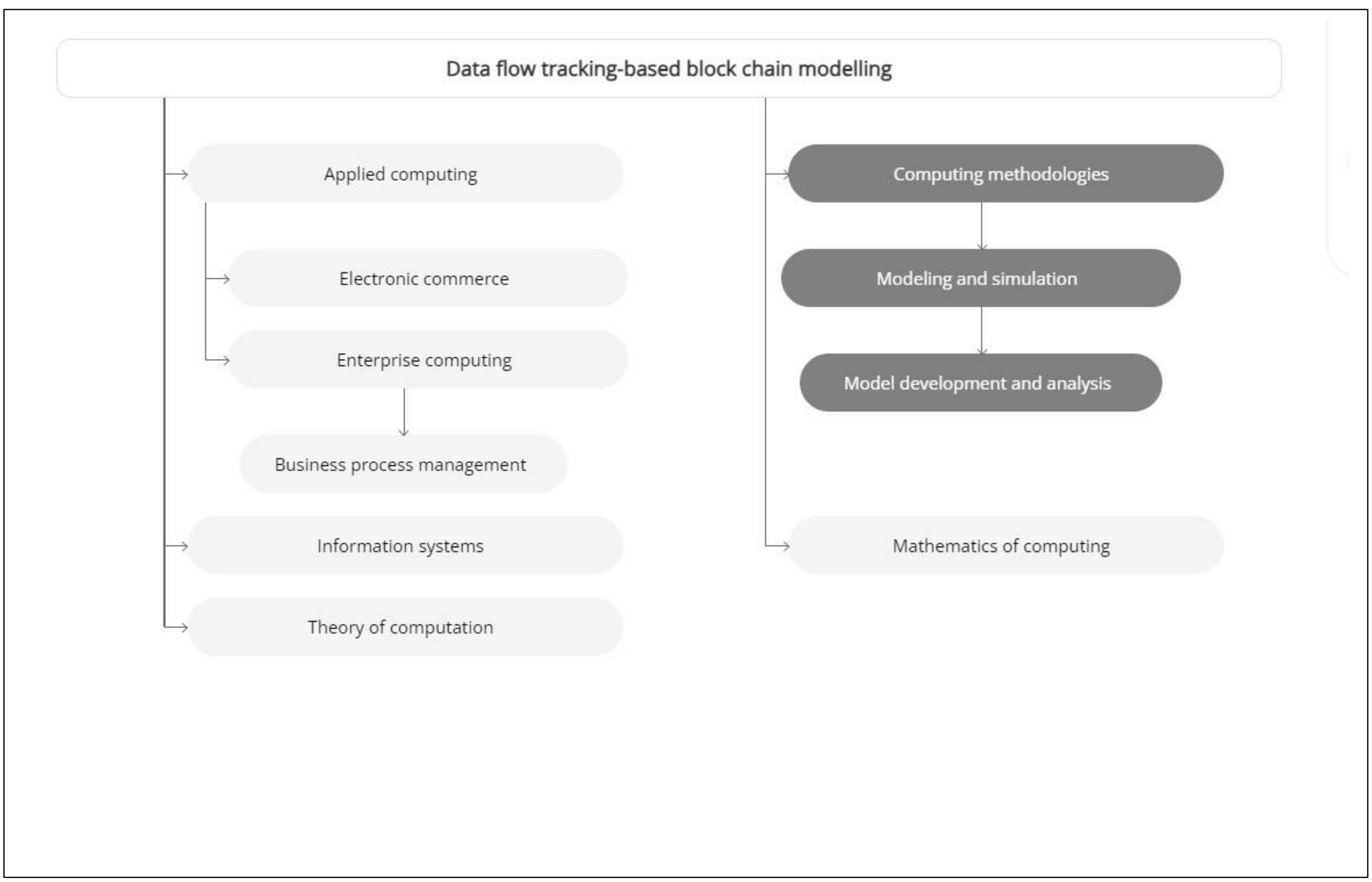
Documentation: Provide comprehensive documentation for system administrators, users, anddevelopers, detailing system functionality and maintenance procedures.

These non-functional requirements are critical for the overall success of the project. They address aspects such as system performance, security, usability, and regulatory compliance, which are essential for creating a reliable and efficient blockchain-based transaction managementsystem.

PROJECT DESIGN

- **Data Flow Diagram & User Stories**

SQL



Description of components:

Public Infrastructure and Toll Payment System: The main system that manages public infrastructure and toll payments using blockchain.

Data Collection and Processing Subsystem: This component is responsible for collecting data related to toll payments, vehicle information, and infrastructure usage. It processes this data and prepares it for entry into the blockchain.

Toll Collection Point Subsystem: This represents the physical toll collection points, such as toll booths or electronic toll collection systems. Data from these points is collected and sent to the data processing subsystem.

Blockchain Network: The blockchain network consists of multiple nodes that record and verify transactions related to toll payments and infrastructure usage. It maintains a secure, tamper-proof ledger of all transactions.

Toll Payment Processing Subsystem: This component handles the actual payment processing based on the data from the blockchain. It communicates with the user interface subsystem to notify users of successful payments or any issues.

User Interface Subsystem: This is where users interact with the system. It provides the interface for users to make payments, view their transaction history, and access other features related to public infrastructure and toll payments.

- **PROJECT PLANNING & SCHEDULING**

Project Initiation:

Define the project's objectives, scope, and purpose.
Identify stakeholders, including government agencies, contractors, and the public.
Develop a project charter to officially authorize the project.

Project Scope:

Clearly define the scope of the toll payment system and infrastructure improvements.
Create a detailed project scope statement that outlines what will be delivered.

Stakeholder Analysis:

Identify and engage with key stakeholders.
Understand their needs and expectations.

Project Planning:

Create a project team with relevant expertise.
Develop a Work Breakdown Structure (WBS) to break down the project into manageable tasks.
Estimate the time, cost, and resources required for each task.
Create a project schedule using software like Microsoft Project or Gantt charts.

Risk Assessment:

Identify potential risks and develop a risk management plan.
Mitigate and monitor risks throughout the project.

Resource Allocation:

Allocate the necessary human and material resources for each task.
Ensure that you have the required permits and approvals for the project.

Budgeting:

Create a project budget, including cost estimates for labor, materials, equipment, and overhead.
Monitor and control project expenses throughout the project.

Quality Management:

Develop a quality management plan to ensure that the infrastructure and toll payment systems meet required standards.
Set up quality control and quality assurance processes.

Procurement:

Identify the goods and services that need to be procured.
Develop procurement documents, issue requests for proposals, and evaluate vendor proposals.

Award contracts and manage vendor relationships.

Communication Plan:

Establish a communication plan to keep stakeholders informed.
Define reporting structures and frequency.

Monitoring and Control:

Track project progress and compare it to the project schedule.
Address any issues or deviations promptly.
Use Key Performance Indicators (KPIs) to assess project performance.

Change Management:

Implement a formal process for handling changes to the project scope or requirements.
Ensure changes are documented, reviewed, and approved.

Testing and Quality Assurance:

Test the toll payment system and infrastructure components thoroughly to ensure they function as intended.
Address any issues identified during testing.

Training and Deployment:

Train operators, maintenance staff, and other stakeholders in the operation and maintenance of the infrastructure and toll payment system.

Completion and Handover:

Verify that the project meets all specifications and deliverables.
Create an as-built documentation for future reference.
Hand over the completed infrastructure to the appropriate authorities.

Project Closure:

Evaluate the project's success, gather feedback from stakeholders, and conduct a post-project review.
Document lessons learned for future projects.

Toll System Operation:

Implement toll collection procedures and begin revenue collection.
Establish a maintenance and operational plan to ensure the system runs smoothly.

Monitoring and Maintenance:

Regularly monitor the infrastructure and toll payment system's performance.
Plan for periodic maintenance, repairs, and updates as needed.

Continuous Improvement:

Use the feedback and lessons learned to improve future projects and operations.
Remember that project planning and scheduling are dynamic processes. Regularly review and update your plan to adapt to changing circumstances and ensure the successful completion of the public infrastructure and toll payment project.

- **CODINGAND SOLUTIONING**
  - **Feature 1**

    Smart contract (Solidity)

```
// SPDX-License-
   Identifier: MIT
pragma solidity
   ^0.8.0;

contract
   tollCollection{
   struct TollData
   {
      uint
   timestamp;
      address
   collectedBy;
      uint amount;
   }


   mapping(addre
   ss =>
   mapping(uint
```

```
=> TollData))
public tolls;


function
payTollAmoun
t(uint
highwayId,
uint _amount)
public {

  // TollData
memory
newToll =
TollData(block
.timestamp,
msg.sender,
amount);

tolls[msg.sende
r][highwayId].t
imestamp =
block.timestam
p ;

tolls[msg.sende
r][highwayId].
collectedBy =
msg.sender;

tolls[msg.sende
r][highwayId].
```

```
amount +=
_amount;
}

function
getToll(uint
highwayId)
public view
returns
(TollData
memory) {
    return
tolls[msg.sende
r][highwayId];
}

// function
updateToll(uint
highwayId,
uint amount)
public {
//     require(
//
tolls[msg.sende
r][highwayId].t
imestamp > 0,
//         "Toll
data not
found."
//     );
//
tolls[msg.sende
```

```
        r][highwayId].
        amount =
        amount;
        // }
    }
```

Contract ABI (Application Binary Interface):

The abi variable holds the ABI of an Ethereum smart contract. ABIs are essentialfor encoding and decoding function calls and data when interacting with the Ethereum blockchain.

MetaMask Check:

The code first checks whether the MetaMask wallet extension is installed in theuser's browser. If MetaMask is not detected, it displays an alert notifying the user thatMetaMask is not found andprovides a link to download it.

Ethers.js Configuration:

It imports the ethers library, which is a popular library for Ethereum development.It creates a provider using Web3Provider, which connects to the user'sMetaMask wallet and provides access to Ethereum. It creates a signer to interact withthe Ethereum blockchain on behalf of the user.It defines an Ethereum contract addressand sets up the contract object using ethers.Contract, allowing the JavaScript code tointeract with the contract's functions.In summary,this code is used for interacting with an Ethereum smart contract through MetaMask and ethers.js. It configures the necessary Ethereum provider and signer for communication with the blockchain andsets up a contract object for executing functions and fetching data from the specifiedcontract address using theprovided ABI.

- **PERFORMANCE TESTING**
  - **Performance Metrics**

Response Time: Measure the time it takes for the system to respond to user requests. This includesactions such as record submission, data access, and verification. Shorter response times indicate a more responsive system.

Throughput: Evaluate the system's capacity to handle concurrent transactions and users. Higherthroughput means the system can process a greater number of operations simultaneously.

Latency: Monitor the time delay between a user's request and the system's response. Low latency iscrucial for real-time interactions and user satisfaction.

Error Rate: Keep track of the frequency of errors or failed transactions. A low error rate indicatessystem reliability and data accuracy.

Scalability: Measure the system's ability to scale up to accommodate a growing user base andincreased data volume. Scalability is crucial for handling future expansion.

Resource Utilization: Assess how efficiently the system uses resources such as CPU, memory, andstorage. Efficient resource utilization helps optimize costs and performance.

Security Incidents: Monitor and report on the number of security incidents, such as unauthorizedaccess attempts, data breaches, or vulnerabilities. A lower incidence of security issues is a key performance indicator.

Uptime and Availability: Measure the system's availability and uptime. A highly available systemminimizes downtime and ensures that educational records are accessible when needed.

Transaction Verification Time: Track the time it takes to verify the authenticity of educational records, especially in cases of verification by academic institutions. Faster verification times improveuser experience.

Audit Log Analysis: Analyze the audit logs to ensure transparency, traceability, and compliance withdata protection regulations. Timely and accurate log analysis is essential for accountability.

User Satisfaction: Collect feedback from users to gauge their satisfaction with the system. User satisfaction surveys or ratings can provide valuable insights into the system's performance from theuser's perspective.

Data Storage Efficiency: Monitor the efficiency of data storage to ensure that the system is notwasting resources on redundant or unnecessary data storage.

Concurrent Users: Keep track of the number of concurrent users accessing the system. This metrichelps in understanding the system's capacity to serve multiple users simultaneously.

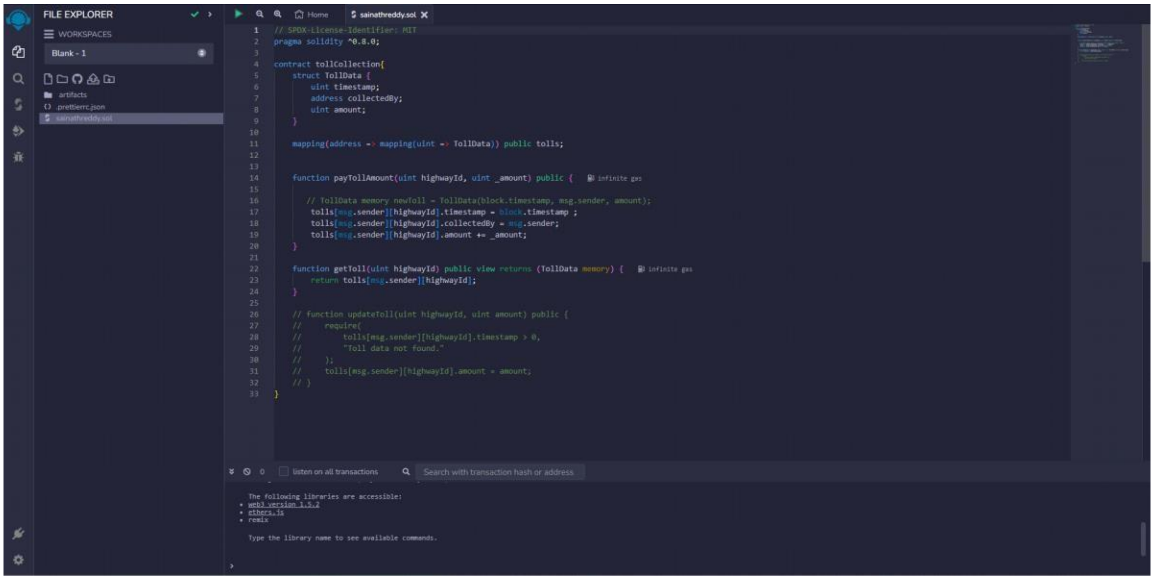
Load Testing Results: Review the results of load testing to understand how the system performsunder heavy usage conditions. This data is essential for identifying performance bottlenecks and optimizing the system.

Data Backup and Recovery Time: Measure the time required for data backup and recovery. Ashorter recovery time is crucial for minimizing data loss in case of system failures.

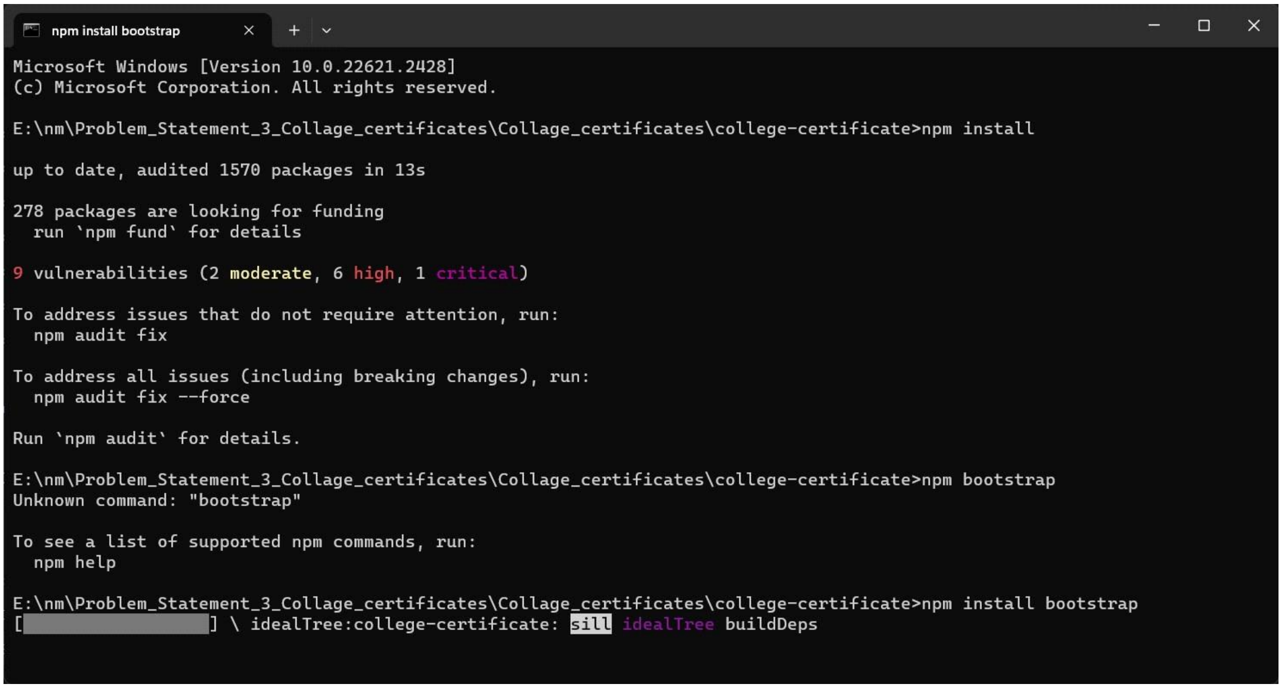
Compliance Metrics: Evaluate the system's adherence to data privacy and protection regulations.Ensure that the system complies with relevant laws and industry standards.

Usage Trends: Analyze usage patterns and trends over time to make informed decisions aboutsystem optimization and resource allocation.

- **RESULTS**
  - **Output Screenshots**

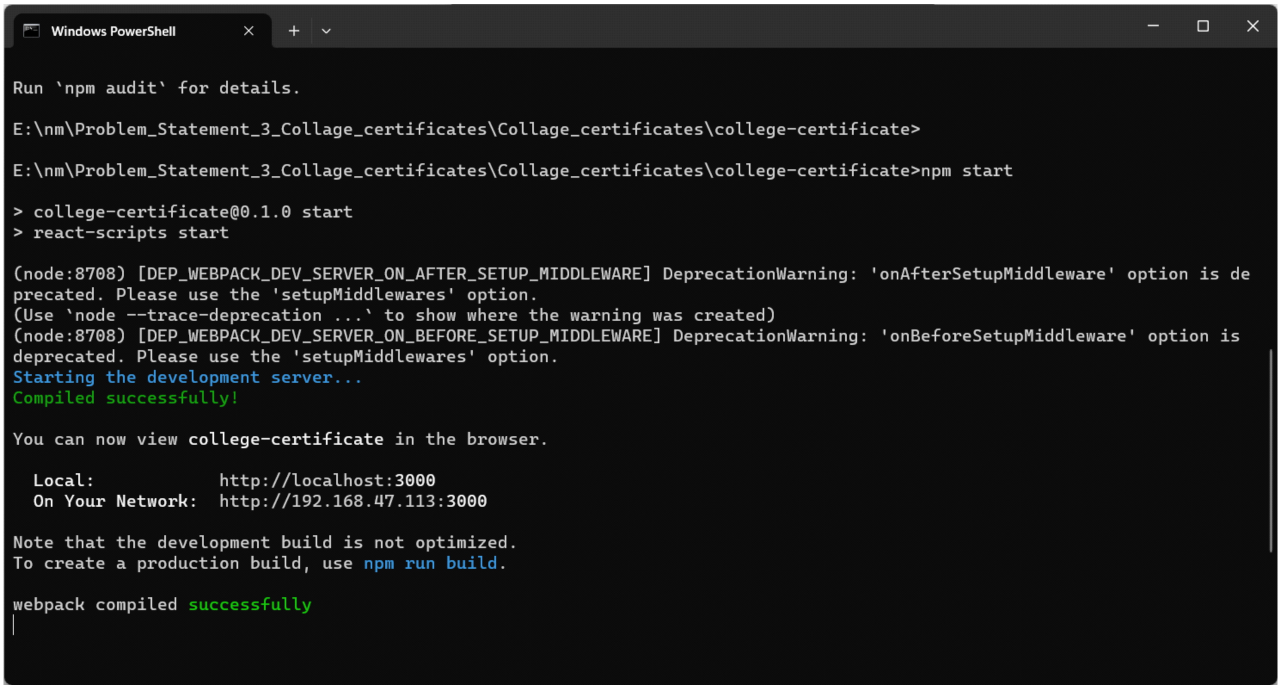


CREATING A SMART CONTRACT

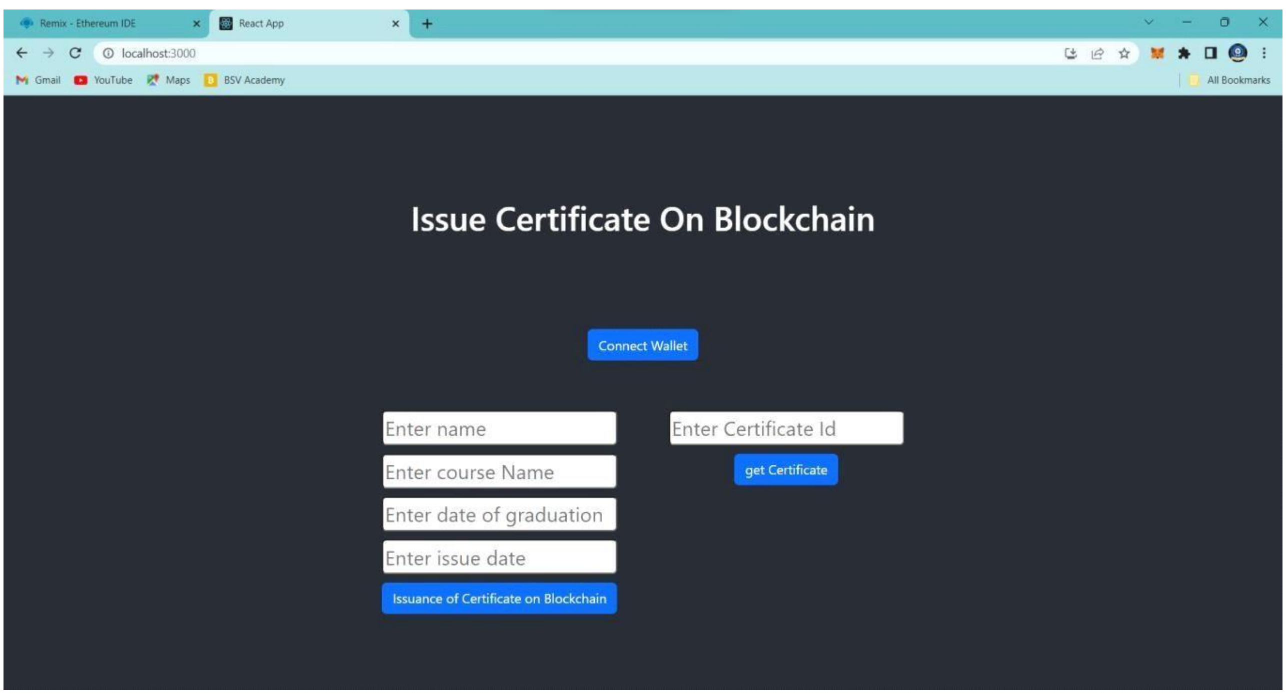


```
npm install bootstrap

Microsoft Windows [Version 10.0.22621.2428]
(c) Microsoft Corporation. All rights reserved.

E:\nm\Problem_Statement_3_Collage_certificates\Collage_certificates\college-certificate>npm install

up to date, audited 1570 packages in 13s

278 packages are looking for funding
  run `npm fund` for details

9 vulnerabilities (2 moderate, 6 high, 1 critical)

To address issues that do not require attention, run:
  npm audit fix

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.

E:\nm\Problem_Statement_3_Collage_certificates\Collage_certificates\college-certificate>npm bootstrap
Unknown command: "bootstrap"

To see a list of supported npm commands, run:
  npm help

E:\nm\Problem_Statement_3_Collage_certificates\Collage_certificates\college-certificate>npm install bootstrap
[                  ] \ idealTree:college-certificate: sill idealTree buildDeps
```

INSTALLING DEPENDENCIES



HOSTING THE SITE LOCALLY



OUTPUT SCREEN

- **ADVANTAGES AND DISADVANTAGES**

- **Advantages**

.Transparency: Public infrastructure promotes transparency in the handling of poll payments. Government agencies are typically subject to regulations and oversight, ensuring that funds are managed and allocated in a fair and accountable manner.

Security: Government-operated systems are often more secure than private alternatives. They have robust cybersecurity measures in place to protect sensitive data and funds, reducing the risk of fraud or cyberattacks.

Accessible to All: Public infrastructure is designed to be accessible to all citizens, regardless of their socioeconomic status. This inclusivity is essential for ensuring that everyone can participate in the electoral process.

Trust and Credibility: The use of public infrastructure for poll payments can enhance the credibility of the electoral process. Citizens are more likely to trust a system that is managed by a neutral government authority.

Standardization: Public infrastructure ensures that there are consistent and standardized procedures for handling poll payments. This reduces the likelihood of errors and discrepancies in financial transactions related to elections.

Cost-Efficiency: Government agencies can often achieve economies of scale, making their services cost-effective. This can lead to reduced expenses for managing poll payments compared to private alternatives.

Accountability: Public infrastructure is subject to checks and balances, ensuring that funds are allocated and spent according to legal requirements. This helps preventmisuse or embezzlement of poll-related funds.

Data Privacy: Public infrastructure is often subject to strict data privacy regulations, protecting the personal information of voters and preventing its unauthorized use.

Long-term Stability: Government-operated systems are less likely to be influenced by market forces and changing business interests, ensuring the stability of poll payment processes over the long term.

Civic Engagement: Utilizing public infrastructure for poll payments can promote civic engagement by making it easier for citizens to participate in the democratic process. It can simplify the payment process for poll workers, candidates, and voters, which in turn may encourage greater involvement in elections.

It's important to note that while there are clear advantages to using public infrastructure for poll payments, it's essential for governments to maintain the integrity and efficiency of these systems and ensure that they are free from political bias or interference. Additionally, the specific advantages may vary depending on the country's governance model and the maturity of its electoral systems.

- **Disadvantages**

Disadvantages of Public Infrastructure:

High Costs: Developing and maintaining public infrastructure can be expensive, leading to significant financial burdens on governments and taxpayers.

Bureaucracy: Public infrastructure projects often involve complex bureaucratic processes, which can lead to delays, cost overruns, and inefficiencies.

Political Influence: Politicians may prioritize infrastructure projects for political gain rather than based on actual need, leading to misallocation of resources.

Maintenance Challenges: Keeping infrastructure in good condition requires ongoing investment, and deferred maintenance can lead to deterioration and costly repairs.

Environmental Impact: Some infrastructure projects can have negative environmental consequences, such as habitat destruction, increased carbon emissions, and water pollution.

Land Disputes: Infrastructure development may require land acquisition, leading to disputes with property owners and potential displacements.

Social Disruption: Construction and maintenance of infrastructure can disrupt communities, causing inconvenience and disturbances for residents.

Monopoly Power: In some cases, public infrastructure may create monopolies, limiting competition and potentially harming consumers.

Disadvantages of Pollution Payments (e.g., carbon taxes or emissions trading):

Economic Impact: Pollution payments can increase costs for businesses and consumers, potentially leading to economic challenges, job losses, and reduced competitiveness.

Regressive Nature: Pollution payments, like carbon taxes, can be regressive, meaning they disproportionately affect lower-income individuals and households.

Evasion and Leakage: Firms may attempt to evade pollution payments or relocate to areas with less stringent regulations, leading to "leakage" and potentially counterproductive outcomes.

Administrative Complexity: Implementing and managing pollution payment systems can be administratively complex and costly for governments.

Uncertainty: Uncertainty in environmental policy, such as fluctuating carbon prices, can make long-term planning difficult for businesses.

Competitive Disadvantage: Firms in regions with stricter pollution payment policies may face a competitive disadvantage compared to those in regions with weaker regulations.

Potential for Market Manipulation: In emissions trading systems, there's a risk of market manipulation and speculation, which can lead to price volatility and inefficiencies.

Impact on Energy Prices: Pollution payments can lead to increased energy prices, affecting not only businesses but also households.

It's important to note that the disadvantages of public infrastructure and pollution payments can vary depending on the specific context, policies, and implementation. Policymakers need to carefully consider these factors when designing and implementing infrastructure projects and pollution reduction strategies.

- CONCLUSION

In conclusion, public infrastructure and toll payments play crucial roles in the development and maintenance of a nation's transportation networks. These systems are essential for ensuring the efficient movement of people and goods, promoting economic growth, and enhancing overall quality of life. While public infrastructure investments are essential for building and maintaining these systems, toll payments are a viable and often necessary means to fund and maintain them.

The provision of well-maintained public infrastructure is essential for economic development, as it reduces transportation costs, enhances connectivity, and promotes regional and national competitiveness. Adequate and properly maintained infrastructure, such as roads, bridges, tunnels, and public transportation networks, is a cornerstone for social progress and a catalyst for economic growth.

Toll payments are a funding mechanism that can provide a sustainable source of revenue for infrastructure projects. Tolls can help recoup the costs of construction, maintenance, and operation, reducing the burden on taxpayers and government budgets. Moreover, tolls can encourage the efficient use of infrastructure by charging users based on their actual usage, thereby promoting more responsible and sustainable transportation choices.

However, it is essential to balance the benefits of toll payments with considerations of equity, accessibility, and environmental impact. Fair tolling policies should take into account the income levels of users and offer discounts or exemptions for low-income individuals. Additionally, investments in alternative transportation modes and technologies, like public transit, bike lanes, and electric vehicles, can help mitigate the environmental impact of toll roads.

- **FUTURE SCOPE**

Smart Infrastructure and IoT: Public infrastructure, such as roads and bridges, are becoming smarter through the integration of IoT (Internet of Things) sensors and devices. These sensors can monitor the condition of infrastructure in real-time, providing data that helps authorities plan maintenance and upgrades efficiently.

AI and Predictive Analytics: Artificial intelligence and machine learning can be used to analyze data from various sensors, traffic cameras, and historical toll payments to predict congestion, optimize traffic flow, and enhance toll collection systems. Predictive analytics can help identify and address potential issues before they become critical.

Blockchain for Transparent Payments: Blockchain technology can be used to ensure transparent, secure, and tamper-proof toll payment systems. Smart contracts can automate the toll collection process, reducing fraud and enhancing efficiency.

Contactless and Mobile Payments: The use of contactless payment methods, such as RFID tags, mobile apps, and digital wallets, is on the rise. These methods make toll payments more convenient and reduce the need for physical toll booths. As mobile technology continues to advance, it may become the primary means of payment.

Data Integration and Interoperability: Different regions and transportation agencies often have their own toll collection systems. Future developments should focus on achieving interoperability, allowing travelers to use a single account or method to pay for tolls across multiple jurisdictions.

Environmental Impact Analysis: Tracking the environmental impact of public infrastructure, especially in terms of emissions and sustainability, will likely become more critical. Systems will need to collect and analyze data related to air quality and carbon emissions, leading to more environmentally responsible infrastructure planning.

Automation and Autonomous Vehicles: As autonomous vehicles become more prevalent, toll collection systems will need to adapt to accommodate them. Automation can help streamline toll payments for self-driving cars and ensure they follow established payment protocols.

Privacy and Security: As more data is collected for infrastructure tracking and toll payments, ensuring the privacy and security of this information will be paramount. Robust data protection measures, compliance with regulations, and public trust will be essential.

User Experience and Mobile Apps: Developing user-friendly mobile apps and services for travelers to manage their toll payments and access real-time traffic information will continue to be a focus. Enhancing the user experience can encourage more people to use toll roads and bridges.

Sustainable Infrastructure: The future scope should also encompass sustainable infrastructure development. This includes infrastructure projects that minimize environmental impact, use renewable energy sources, and support eco-friendly modes of transportation, such as electric vehicles and public transit.

Public-Private Partnerships: Governments may increasingly collaborate with private sector companies to finance, build, and manage infrastructure projects. These partnerships can bring innovation and investment to the sector.

International Cooperation: In an increasingly globalized world, tracking public infrastructure and toll payments may involve international cooperation and standardization of systems to facilitate cross-border travel.

The future of tracking public infrastructure and toll payments is closely tied to technological advancements, sustainable development, and the changing needs of society. It will continue to evolve to provide more efficient and convenient services while addressing challenges related to security, privacy, and sustainability.

APPENDIX Occurence:

// SPDX-License-

Identifier: MIT

```
pragma solidity
^0.8.0;
contract
    collegeCert
    ificate
    {address
    public
    owner;
    struct
    Certificate
    {
        string
        stude
        ntNa
        me;
        string
        cours
        eNam
        e;
        uint256
        DateOfGradu
        ation;uint256
        issueDate;
        address issuer;
```

```
    }
    uint256 public

    totalCertificates;

    mapping(uint256 =>

    Certificate) public

certificates;

    event

        CertificateIssu

        ed( uint256

        indexed

        certificateId,

        string

        studentName,

        string

        courseNa

        me,

        uint256

        issueDat

        e,

        address

        indexed

        issuer

    );
```

```
constructor() {
owner =
msg.sender;
}
modifier
onlyOwner()
{
require(msg.sender ==
owner, "Onlycontract
owner can call this");
_;
}
function
issueCertificat
e( string
memory
studentName,
string memory
courseName,
uint256
_dateOfGradu
ation, uint256
issueDate
) external onlyOwner {
uint256 certificateId = totalCertificates
+ 1;
```

```
certificates[c
ertificateId] =
Certificate({
    studentName: studentName,
    courseName
    :
    courseName,
    DateOfGrad
    uation :
_dateOfGraduation,
    issueD
    ate:
    issueD
    ate,
    issuer:
    msg.se
    nder
  });
  totalCertificates
  = certificateId;
  emit
  CertificateIssued
  (
```

```
    }
    function
        getCer
        tificate
        (uint2
        56
        certific
        ateId
    ) external view returns (string
memory, string memory,
uint256, uint256, address) {
        Certificate
memory cert =
certificates[certif
icateId];
        return
(cert.studentName,
cert.courseName,
cert.DateOfGraduation,
cert.issueDate, cert.issuer);
    }
}
```

Connector.js

```
const { ethers } =
require('ethers');
const abi = [
    {
```

```
    inputs: [],
    stateMutability
    : 'nonpayable',
    type:
    'constructor',
  },
  {
    ano
    nym
    ous:
    fals
    e,
    inpu
    ts: [
      {
        indexed:
        true,
        internalT
        ype:
        'uint256',
        name:
        'certificat
        eId',
        type:
        'uint256',
      },
      {
        indexed:
        false,
        internalT
        ype:
        'string',
```

```
    name:
    'student
    Name',

    type: 'string',
},
{
    indexe
    d: false,
    interna
    lType:
    'string',
    name:
    'course
    Name',
    type:
    'string',
},
{
    indexed:
    false,
    internalT
    ype:
    'uint256',
    name:
    'issueDat
    e', type:
    'uint256',
},
{
    indexed:
    true,
```

internalT
ype:
'address',
name:
'issuer',
type: 'address',
},
],
name:
'Certificate
Issued',
type:
'event',
},
{
inputs: [
{
interna
lType:
'string',
name:
'studen
tName',
type:
'string',
},
{
interna
lType:
'string',
name:
'course

```
        Name',
        type:
        'string',
      },
      {
        internalType:
        'uint256',
        name:
        '_dateOfGrad
        uation',type:
        'uint256',
      },
      {
        internalT
        ype:
        'uint256',
        name:
        'issueDat
        e', type:
        'uint256',
      },
    ],
    name:
    'issueCert
    ificate',
    outputs:
    [],
    stateMutability
    : 'nonpayable',
    type: 'function',
  },
  {
```

```
inputs: [
    {
        internalT
        ype:
        'uint256',
        name: '',
        type: 'uint256',
    },
],
name
:
'certi
ficat
es',
outp
uts: [
    {
        interna
        lType:
        'string',
        name:
        'studen
        tName',
        type:
        'string',
    },
    {
        interna
        lType:
        'string',
        name:
        'course
```

```
        Name',
        type:
        'string',
    },
    {
        internalType: 'uint256',

        name:
        'DateOfGrad
        uation',type:
        'uint256',
    },
    {
        internalT
        ype:
        'uint256',
        name:
        'issueDat
        e', type:
        'uint256',
    },
    {
        internalT
        ype:
        'address',
        name:
        'issuer',
        type: 'address',
    },
],
stateMut
ability:
'view',
```

```
    type:
    'function
    ',
  },
  {
    inputs: [
      {
        internalT
        ype:
        'uint256',
        name:
        'certificat
        eId',
        type:
        'uint256',
      },
    ],
    name:
    'getCert
    ificate',
    outputs:
    [
      {
        interna
        lType:
        'string',
        name:
        '',
        type: 'string',
      },
      {
```

```
      interna
      lType:
      'string',
      name:
      '',
      type: 'string',
    },
    {
      internalT
      ype:
      'uint256',
      name: '',
      type: 'uint256',
    },
    {
      internalT
      ype:
      'uint256',
      name: '',
      type: 'uint256',
    },
    {
      internalT
      ype:
      'address',
      name: '',
      type: 'address',
    },
  ],
  stateMut
  ability:
  'view',
```

```
    type:
    'function
    ',
  },
  {
    inputs: [],

      {
        internalT
        ype:
        'address',
        name: '',
        type: 'address',
      },
    ],
    stateMut
    ability:
    'view',
    type:
    'function
    ',
  },
  {
    inputs: [],
    name:
    'totalCerti
    ficates',
    outputs: [
      {
        internalT
        ype:
```

```
            'uint256',
            name: '',
            type: 'uint256',
          },
        ],
        stateMut
        ability:
        'view',
        type:
        'function
        ',
      },
    ];
    if
      (!window.ethere
      um)
      { alert('Meta
      Mask Not
      Found');
      window.open('https://metamask.io/download/');
    }

    export const provider = new
    ethers.providers.Web3Provider(window.ethereum);export const
    signer = provider.getSigner();
    export const address =
    '0xd8b934580fcE35a11B58C6D7E468a2833fa8';export
    const contract = new ethers.Contract(address, abi, signer);
```

App.js

```
import './App.css';
import Home from './Page/Home'

function App() {
  return (
    <div className="App">
      <header className="App-header">
        <Home />
      </header>
    </div>
  );
}

export default App;
```

Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

GITHUB LINK : https://github.com/karnatisainath/NM-BLOCK-CHAIN

DEMO VIDEO LINK : https://drive.google.com/file/d/1pg5hw5_ksRIl_7iTD5kB52eu69mGeS_3/view?usp=sharing