Project Title

Blockchain Powered Parking Solution
for Smart Cities

CSE400C
Capstone Project

Project Members

Fahamidul Hasan (2018-3-60-089)
Adnan Rahman Tushar (2019-3-60-099)
Sihama Ramjana Khadija (2020-1-60-019)
Tanjela Rahman Rosni (2020-1-60-018)

Project Supervisor

Dr. Md Mostofa Kamal Rasel (Ph.D.)
Assistant Professor,
Department of Computer Science and Engineering,
East West University

Project Title:

Blockchain Powered Parking Solution for Smart Cities

Abstract:

This research investigates the monetization and optimization of privately-owned empty parking spots to alleviate street parking congestion. It aims to develop seamless rental platforms using advanced technologies to connect parking spot owners with potential renters efficiently. The study explores effective monetization strategies, including dynamic pricing models that reflect varying demand and supply, while ensuring affordability for users. Additionally, it examines criteria for optimal parking spot selection, focusing on cost, distance, security, availability and emphasizes designing user-friendly interfaces for easy navigation. The research also addresses security measures for traceable transactions, and cost reduction strategies to minimize operational expenses. The goal is to provide a comprehensive, efficient, and secure solution that benefits both parking spot owners and users, enhancing urban parking management.

Introduction:

Background

The traditional approach to parking, characterized by manual processes and static infrastructure, has proven increasingly inadequate in meeting the demands of a rapidly urbanizing world. In response, a new era of automated parking management, underpinned by modern technologies, is emerging as a beacon of efficiency and innovation.This change is not just convenient—it's essential as cities work to maximize available space, lessen traffic, and give drivers a smooth, uninterrupted experience navigating the complex urban landscape.

Modern Technology-Based Automated Parking Management in Urban Areas: Modern technology can play a crucial role in improving parking management in urban areas. This likely involves the use of digital solutions to make parking more efficient and user-friendly.

Real-Time Availability Tracking:

Real-time availability tracking is a fundamental aspect of modern parking management. It allows drivers to check the availability of parking spaces in real time, reducing the time spent searching for a parking spot.

Peer-to-Peer (P2P) Parking Sharing:

Peer-to-peer parking sharing implies that individuals can share their parking spaces with others. This can be a collaborative approach to solving parking issues, where those with available spaces can rent them out to those in need.

Integration with Navigation Apps:

Integrating parking information with navigation apps can help drivers find parking options more easily. Navigation apps can provide directions to available parking spots.

Secure Decentralized Environment:

The use of secure decentralized environments, possibly based on blockchain technology, can enhance trust and security in parking management systems. Decentralization can help ensure transparency and prevent data manipulation.

Secure Payment Using Ethereum (ETH): Secure payment methods using Ethereum (ETH) can streamline transactions in automated parking management systems. ETH can provide secure, decentralized payment solutions, offering benefits such as transparency, lower transaction costs, and enhanced security.

Research Questions/Problem Statements:

Main Question: How to monetize and optimize the utilization of privately-owned empty parking spots to alleviate street parking congestion?

Sub-Questions:

a. **Facilitating Parking Spot Rentals:**What platform or system can be developed to facilitate the seamless renting of empty parking spots?How can technology be leveraged to connect parking spot owners with potential renters efficiently?

b. **Monetization Management**:What strategies can be employed to effectively monetize private parking spaces while ensuring affordability for users?How can dynamic pricing models be implemented to reflect varying demand and supply of parking spaces?

c. **Optimal Parking Spot Selection:**What criteria should be considered when evaluating the suitability of a parking spot, such as cost, distance, security and availability?

d. **Security Measures:**What security measures can be implemented to safeguard both the parked vehicles and the parking spot itself?How can transactions and payments be secured to protect the financial interests of both parking spot owners and renters?

e. **Cost Reduction Strategies**:How can operational costs associated with managing and maintaining parking spots be minimized?What technologies or methods can be employed to streamline transaction processes and reduce resource utilization?

The research aims to address the growing issue of street parking congestion by exploring the effective monetization and utilization of privately-owned empty parking spots. It involves investigating the development of platforms or systems that facilitate easy renting, implementing dynamic pricing for optimal monetization, and designing user-friendly interfaces to assist in spot selection. Additionally, the study explores security measures to protect both vehicles and transactions, as well as strategies to reduce overall operational costs. The goal is to provide a comprehensive solution that benefits both parking spot owners and users, ultimately contributing to improved urban parking management.

Research Objectives:

The outlined objectives align with the overarching goal of monetizing and optimizing the usage of private parking spots to alleviate street parking congestion.

1.Digital Payment System (ETH based) :The implementation of secure ETH based digital payment systems is crucial for streamlining transactions and improving user convenience in the context of monetizing parking spots. These systems enable efficient and secure processing of payments, reducing dependence on traditional payment methods.

3.Parking Spot Recommendation System: The system aims to assist users in finding the most suitable parking spot by implementing a recommendation system. Factors such as cost, distance, security, and availability are considered to enhance the overall user experience.Essentially, it guides users to the most suitable parking spots based on their preferences and requirements.

4.Monetizing Empty Parking Spots: The primary goal is to generate revenue and enhance the utilization of privately-owned parking spots, ultimately reducing street parking congestion. This involves developing strategies to effectively market and make these spaces available for rent.

5.Cost Reduction and Optimization with Blockchain and DB: The integration of blockchain and efficient database management is targeted at reducing maintenance costs and optimizing transactions and resources. This strategic combination aims to enhance operational efficiency and minimize associated expenses.

6.Blockchain for Security and Transaction Tracking: Blockchain technology is leveraged to ensure the security of parking spots and cars. It provides a transparent and traceable framework for transactions, enhancing overall security measures within the parking management system.

These objectives collectively ensure that the parking management system is not only technologically sound but also socially responsible, economically feasible, environmentally conscious, and adaptable to various user needs and regulatory contexts.

Focus:

This research focuses on the urban parking ecosystem, specifically on privately-owned parking spots that are currently underutilized. The study aims to bridge the gap between supply (parking spot owners) and demand (drivers looking for parking) through innovative technological solutions. The primary geographical focus is on urban areas with high parking congestion, although the findings and solutions may be applicable to a wider range of contexts.

Research Limitations:

- The study relies on the availability and willingness of parking spot owners to participate in the rental platforms.
- The effectiveness of dynamic pricing models may vary based on specific urban contexts and user behaviors.
- The security measures proposed need to be robust enough to address potential fraud and misuse, which may evolve over time.
- Technological solutions require user adoption, which can be influenced by factors such as familiarity with technology and trust in the platform.
- The research is limited by the availability of data on parking spot usage and user preferences.

By addressing these limitations and focusing on the outlined objectives, this research aims to provide valuable insights and practical solutions for urban parking management. The following chapters will delve deeper into the methodologies, analysis, results, and implications of the study.

Related Works:

Literature review:

[1]The paper discusses the challenges of finding parking spaces in urban areas due to increased urbanization and the growing number of vehicles.The proposed solution aims to use blockchain technology to create a transparent and efficient platform for renting out unused land as smart parking spaces.A Blockchain-based solution for creating smart parking spaces, with key contributions and services such as a Blockchain ecosystem, non-fungible parking tokens (NFTs), smart contracts, real-time surveillance, and a user-friendly mobile application for parking-related activities. Some noted inefficiencies include privacy concerns and security deposits. The main stakeholders involved are the certified landowners, contractors, and government agencies. Rental packages are available on an hourly, daily, and monthly basis, targeting both individuals and companies. However, the paper lacks specific technical details on the implementation of the system's searching/recommendation functionality for available parking spaces, offering only a high-level overview of the system's features.

[2]The paper mentions the development of a parking management system that leverages blockchain technology to enhance privacy ,transaction security and transparency to improve the overall parking experience.The information describes an Ethereum-based consortium blockchain parking system with contributions and services including the use of a consortium blockchain, privacy-preserving parking management, smart contracts for rentals, and user anonymity. Some identified inefficiencies are related to the storage of all data in the blockchain and user adoption. The main stakeholders involved in this system are vehicle users (drivers), parking space owners, government departments, certification authorities, blockchain developers, and service providers. Rental packages include hourly, daily, monthly, flexible duration, and dynamic pricing options,

targeting both individuals and companies.The paper, however, lacks specific technical details regarding the implementation of the searching and recommendation algorithms. The recommendation algorithm is presumed to match user preferences with available parking spaces while ensuring user privacy and security, but the precise workings are not detailed.

[3]The authors of this paper developed a model to provide smart parking solutions based on infrastructure integration mechanisms using blockchain technology. The integrated smart parking system described leverages blockchain technology to address trust and data integrity challenges in a smart city context. The contributions and services provided by the system include scalability through a layered architecture and a seamless user experience. However, there are identified inefficiencies related to high transaction costs, specifically gas fees associated with blockchain transactions.The key stakeholders involved in this system include individual clients, commercial users (enterprises), owners/operators of parking garages, developers of blockchain technology, local authorities, and data analytics and insights providers. The parking spaces can be lent for various durations, including hourly rentals, daily rentals, monthly rentals, and custom durations, targeting both individuals and companies.The specific technical details of the implementation for searching and recommending parking spaces are not provided in the text. However, it is mentioned that the system involves the integration of various sensors, data processing, user interfaces, APIs, and real-time data analysis to offer accurate and up-to-date information to drivers.

[4]The authors proposed Blockchain-based Smart parking with Fairness, reliability and Privacy protection which is called BSFP. The paper discusses an integrated smart parking system that utilizes blockchain technology to address trust and data integrity challenges in smart cities. It focuses on privacy protection through cryptographic techniques, fairness with rewards for honest miners and penalties for dishonest ones, efficiency with constant time complexity for most operations, and the handling of parking operations. It identifies computation overhead and complex registration and payment procedures as inefficiencies. The system involves stakeholders like payment service providers, network operators, sensor providers, parking facility owners, local governments, and personal vehicle users. It offers parking spaces for various durations, targeting both individuals and companies. While it mentions the inclusion of fair allocation algorithms and a recommendation system, specific technical implementation details are not provided

[5]The authors proposed a privacy-preserving blockchain-based smart parking system which is an advanced solution that not only addresses the efficiency and transparency of traditional parking systems but also focuses on protecting user's privacy and ensuring trust through reputation management.The paper outlines an integrated smart parking system that uses blockchain technology to tackle trust and data integrity challenges in smart cities. It focuses on providing privacy-preserving smart parking with anonymity, reputation management, and fair parking rates. It highlights blockchain overheads, accuracy of reputations, and user learning curve as inefficiencies. The primary stakeholders are drivers, parking lots, and parking authorities, and the system offers parking packages on an hourly, daily, and monthly basis.Individuals and companies are targeted.Implementation of efficient searching/recommendation is not discussed.

Limitations in Current Parking Solutions:

1. Blockchain Overhead (HIGH GAS USAGE)
GAS fees, or transaction processing expenses on the blockchain, are frequently incurred by blockchain transactions. The total cost-effectiveness of the system is impacted by these fees, which can vary and occasionally be rather high.
Impact: High GAS fees can deter users from utilizing blockchain-based systems for everyday transactions, such as parking payments, due to the added expense.

2. Mainstream Recommendation (Determined by Rating and Availability)
Traditional parking recommendation systems normally base their suggestions for parking locations on fundamental elements like user ratings and availability. Although helpful, this method ignores individual needs and does not always offer the most practical or economical solutions.
Impact: Users may end up with suboptimal parking choices that don't fully meet their specific requirements, such as proximity to their destination, security, features, and cost.

3.**Privacy vs. Transparency**
Balancing privacy and transparency is crucial. Public blockchains pose privacy concerns due to their transparent nature, while consortium blockchains offer better privacy but at the cost of scalability and trust among nodes.

Solutions:

1.Personalized Parking Recommendations

Summary: To address these limitations, we have developed a system that generates personalized parking recommendations based on analyzed data.
Variables Taken Into Account: We take into account a number of variables while making suggestions, such as cost, distance, availability and security. This guarantees that recommendations are sent to consumers based on their unique requirements and preferences.
Benefit: This personalized approach enhances user satisfaction by providing more relevant and convenient parking options.

2.Blockchain integration with effective database management

Blockchain Use: To cut maintenance costs and maximize transactions and resources, we combine blockchain technology with effective database administration.
Optimization: By reducing transaction fees and related costs, this combination seeks to improve operational efficiency.
Benefit: We are able to provide a parking system that is more dependable and affordable by utilizing blockchain's security and transparency in conjunction with good data processing.
Security: The use of private blockchain allows access control , reduced attack surface , instant mining as well as enhanced privacy.

Materials and Method:

Data Collection:

In our project, we utilized Google Maps to identify potential parking spaces. We randomly selected some locations, aiming to create a diverse and comprehensive dataset. For each selected location, we collected the geographical coordinates - latitude and longitude - which provide precise information about the location of these potential parking spaces.

Along with the geographical coordinates, we also gathered the address information for each location as provided by Google Maps. This includes details such as the street name, city which offer additional context about the location.

However, it's important to note that while the geographical coordinates and address information are accurate as per Google Maps, other details associated with these locations are not real data. These include the owner details, parking capacity, facilities available at the parking space, and other related information.

These additional details are hypothetical and have been created for the purpose of this project. They do not reflect the actual characteristics of the parking spaces. Therefore, while the dataset provides a realistic representation of potential parking spaces in terms of their geographical locations and addresses, it does not offer accurate information about the actual capacity, facilities, or ownership of these spaces.

We have also conducted surveys among stakeholders in order to gain insight about their preferences and usage pattern of parking services.

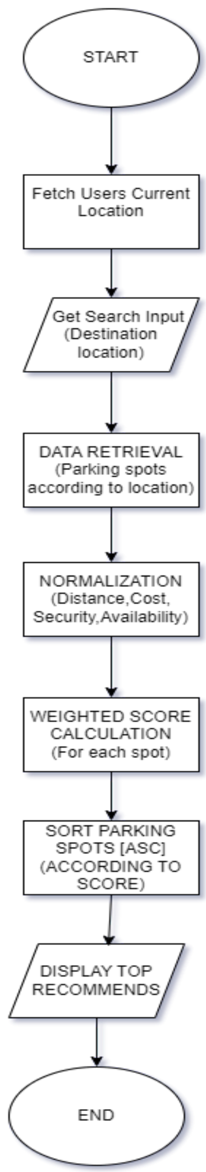
Expected Outcome:

1.Efficient Parking Recommendations: The system will generate personalized parking recommendations based on analyzed data, considering factors such as distance, cost, security and availability.

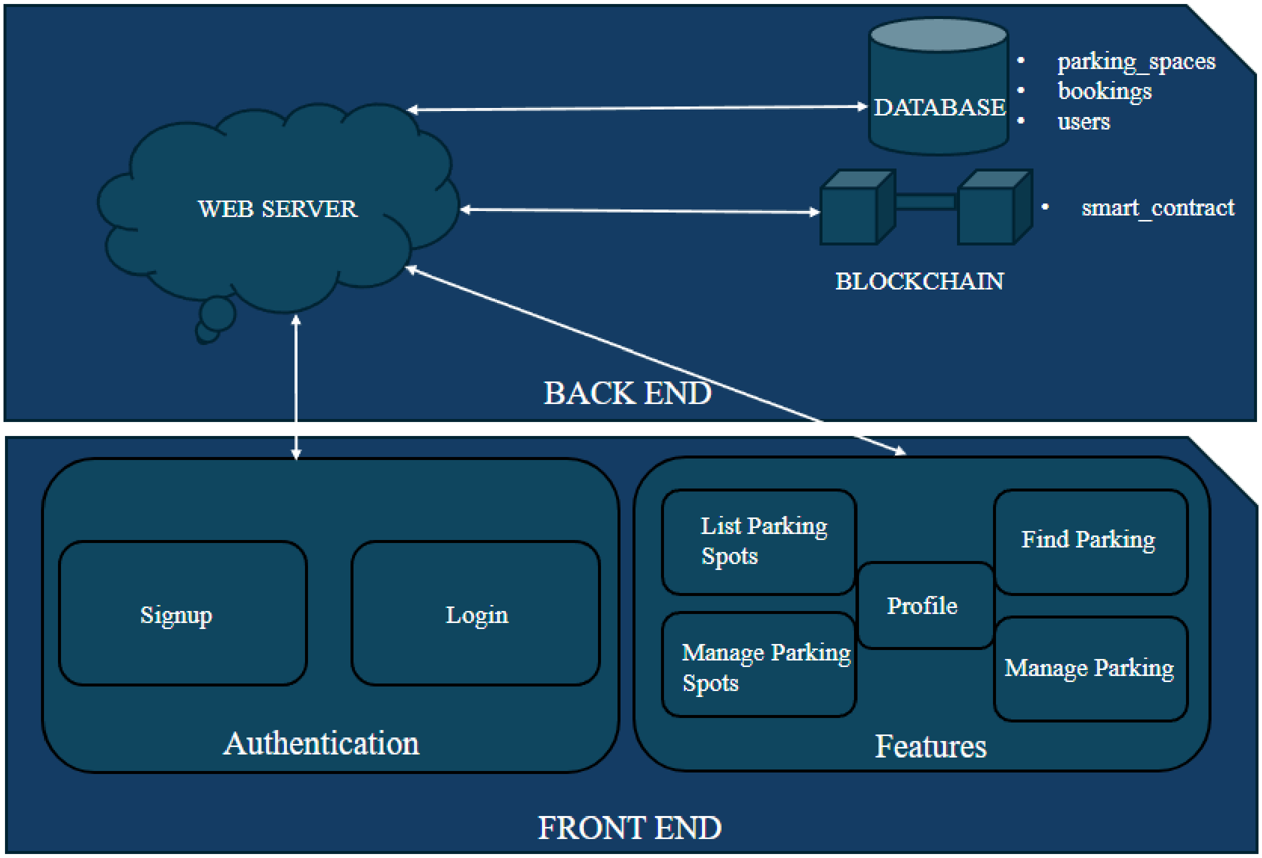
2.Secure Transaction and Payment Processing: The system will leverage WEB3.0 technologies to ensure secure data transmission and payment processing. WEB3.0, characterized by decentralized applications (dApps) and blockchain integration, offers robust security features ideal for handling transactions in automated parking management systems.

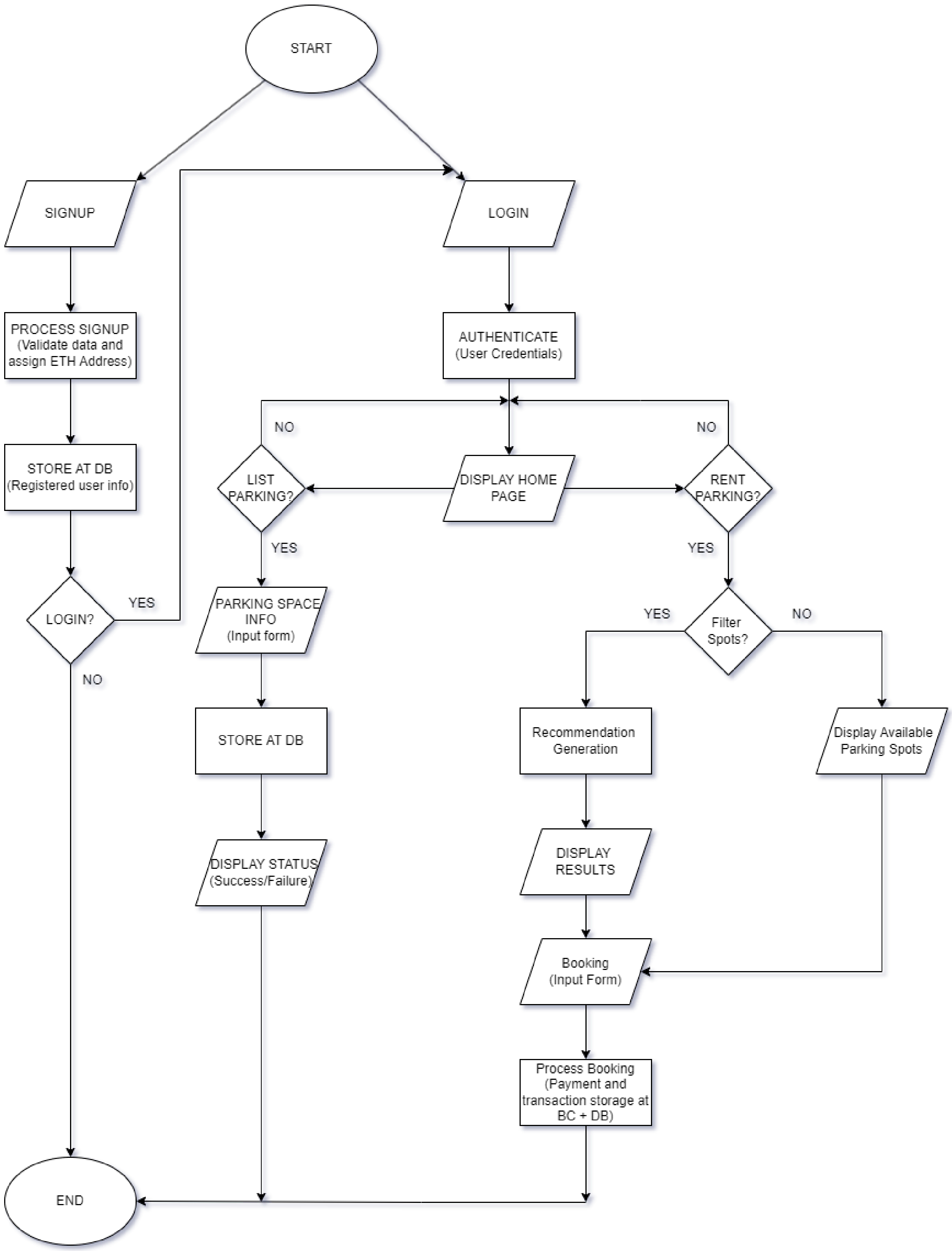
2.Enhanced User Experience: Data analysis will contribute to an improved user experience by providing accurate and timely information about parking options, empowering users to make informed decisions as well as providing them with the features that they desire.

Proposed Recommendation Model :



Proposed System :




START
SIGNUP
LOGIN
PROCESS SIGNUP (Validate data and assign ETH Address)
AUTHENTICATE (User Credentials)
STORE AT DB (Registered user info)
NO
NO
LIST PARKING?
DISPLAY HOME PAGE
RENT PARKING?
YES
YES
LOGIN?
YES
PARKING SPACE INFO (Input form)
YES
Filter Spots?
NO
NO
STORE AT DB
Recommendation Generation
Display Available Parking Spots
DISPLAY STATUS (Success/Failure)
DISPLAY RESULTS
Booking (Input Form)
Process Booking (Payment and transaction storage at BC + DB)
END

Implementation :

Frontend Design :

- **User-Friendly Forms :** The signup and login pages boast intuitive forms that are easy to navigate and understand. Clear labels, appropriate input fields and well-placed buttons guide users through the process effortlessly.

- **Interactive Home Page :**

Find parking : This section provides users with a clear and organized list of available parking spots. Information might include location details, parking type and pricing.

Manage parkings**:** A dedicated section allows users to view parking history in details as well as allow them to cancel their existing parking reservations. This empowers users to manage their parking needs efficiently.

List your parking spot **:** This section allows users to offer their parking spots for rent by filling up a spot information form.

Manage parking spots**:** A dedicated section allows users to view all the parking spots offered by them and allows them to update the status (active/inactive) of specific parking spots as well as allowing them to accept/reject any pending parking requests.

User Profile**:** A comprehensive user profile section enables users to view their personal information as well as ethereum address and balance.

Frontend :

HTML : The foundation of our web app's structure and content is built with HTML. This ensures the proper layout and organization of all elements on the page.

CSS : Cascading Style Sheets (CSS) is used to style our web app's visual appearance. CSS allows for customization of fonts, colors, backgrounds, and overall layout. This creates a visually appealing and user-friendly and responsive interface.

JavaScript (JS) : JavaScript adds interactivity and dynamic behavior to our web app.

Additional Considerations :

Responsive Design **:** Our web app is designed to adapt seamlessly to different screen sizes, ensuring optimal viewing experience on desktops, tablets, and smartphones.

Backend :

The backend of this web application utilizes a powerful combination of MySQL and PHP to ensure efficient data storage, retrieval, and manipulation.

Relational Database with MySQL :

Data Storage **:** MySQL serves as the central repository for all application data. It creates a structured and organized environment for storing information like:

- bookings (booking_id, user_id, space_id, start_time, end_time, total_cost,payment_method, license_plate, drivers_license, status,transaction_hash)
- parking_spaces(space_id, city, area, location, lat, lon, capacity, available_capacity, price, facilities, status, reg_timestamp, user_id, owner_name, owner_mobile)
- users (user_id, username, dob, gender, residential_address, city, zip, contact, email, nid, password, eth_address)

Relational Structure **:** Tables are created within the database to store related data. These tables are linked through unique identifiers, enabling efficient data retrieval and manipulation. For example, a "bookings" table is linked to a "users" table using unique user_id and booking_id references.

Dynamic Processing with PHP :

Server-Side Scripting **:** PHP acts as the bridge between the user interface (frontend) and the database (backend). It processes user requests submitted through our web app. Data Interaction **:** PHP scripts handles tasks like

- CRUD Operations on DB
- User authentication (login/signup)
- Processing listings
- Processing bookings
- Recommendation generation

Dynamic Content Generation **:** PHP scripts can dynamically generate web page content based on user input or retrieved data.So this allows in our web app for features like displaying personalized user information, updating parking availability in real-time, and presenting dynamic search results.

Secure and Traceable Transactions with Smart Contracts :

This web application takes a leap forward in security and transparency by incorporating Solidity smart contracts for managing parking transactions.

Immutable Code : Once deployed, the code of a smart contract cannot be altered. This ensures all transactions operate according to the predetermined rules established within the contract, fostering immutability and tamper-proof records.

Traceable Transaction History : All transactions involving our smart contract will permanently record on our private blockchain, providing a transparent trail. This allows for easy verification of transaction details and dispute resolution if necessary.

Attributes stored : bookingID,userID,spaceID,startTime,endTime,totalCost,hashValue,owner

Blockchain Structure :

This web application leverages a powerful combination of technologies to seamlessly integrate with a local blockchain network for development and testing purposes.

Technologies used **:** TRUFFLE , GANACHE , WEB3.JS

Truffle : Truffle acts as a development framework specifically designed for building and testing Ethereum applications. It offers features like:

- Smart contract compilation and deployment.
- Automated testing of smart contracts with a built-in testing suite.

Ganache : Ganache is a popular local blockchain simulator. It provides a user-friendly interface for managing the local blockchain network, including **:**

- Creating and managing accounts for testing purposes.
- Exporting and importing blockchain data for testing scenarios.

Web3.js : This JavaScript library serves as the bridge between the web application and the local blockchain. It allows the web app to interact with smart contracts deployed on the local chain, enabling features like **:**

- Reading data from the blockchain.
- Executing transactions on deployed smart contracts.
- Managing user accounts and signing transactions.

Solidity Code :

```
// SPDX-License-Identifier: MIT

pragma solidity ^0.8.0;

contract Parking {

    struct Booking {
```

```
    uint256 bookingId;

    uint256 userID;

    uint256 spaceId;

    uint256 startTime;

    uint256 endTime;

    uint256 totalCost;

    bytes32 hashValue;

    address payable owner;

}



mapping(uint256 => Booking) public bookings;



function addBooking(
```

```
    uint256 _bookingId,

    uint256 _userID,

    uint256 _spaceId,

    uint256 _startTime,

    uint256 _endTime,

    uint256 _totalCost,

    bytes32 _hashValue,

    address payable _owner

) public payable {

    require(_owner != address(0), "Invalid Address");

    require(msg.value >= _totalCost, "Insufficient ETH sent");



    bookings[_bookingId] = Booking(
```

```
        _bookingId,

        _userID,

        _spaceId,

        _startTime,

        _endTime,

        _totalCost,

        _hashValue,

        _owner

    );



    _owner.transfer(msg.value);

}
```

```
    function getBooking(

        uint256 _bookingId

    ) public view returns (Booking memory) {

        return bookings[_bookingId];

    }

}
```

Recommendation Algorithm:

Sorting, filtering, and weighted scoring are all incorporated into the recommendation system to give consumers customized recommendations. During the suggestion process, factors including price, security, distance and availability are taken into account. The relative weights assigned to each criterion in the recommendation process are determined from the pre launch survey.
The system uses the Min-Max normalization scale, which scales each parameter to a range between 0 and 1, to guarantee a fair comparison across many criteria. Values can be standardized by this method, which also guarantees that parameters with various scales or units can be compared successfully.

The normalized value for each parameter is calculated using the Min-Max normalization formula:

$$\textit{Normalized value} = (\textit{Cost} - \textit{Min Cost}) / (\textit{Max Cost} - \textit{Min Cost})$$

The final score for each recommendation is determined by calculating the weighted sum of these parameters after they have been normalized.

Math formulations:

The total score for each recommendation is calculated using the formula:

$$totalScore = (distanceWeight \times DistanceNorm) + (costWeight \times costNorm) + (securityWeight \times securityNorm) + (availabilityWeight \times availabilityNorm)$$

The algorithm may produce recommendations that are in line with user priorities and preferences thanks to this total score, which represents the combined effect of all the parameters on the recommendation process.

Algorithm Complexity:

Complexity of Time:

O(N x M) [For N parking places to be normalized with M features]: This complexity results from the normalization process of each parking spot's features. The algorithm's time complexity is O(N x M) since it must normalize M characteristics for each of the N parking spaces.

O(N Log N) [For List Sorting Using finalScore]: The list must be ordered in accordance with these scores once each parking spot's final score has been determined. Most sorting algorithms, such as quicksort or mergesort, have an O(N Log N) time complexity when sorting N elements.

As a result, for the recommendation algorithm, the dominating factor, which is O(N log N), determines the overall time complexity.

Complexity of Space:

O(N x M) [The total space complexity varies according to the quantity of parking spaces]: The recommendation algorithm's amount of parking spots taken into account largely determines the space complexity. O(N x M) space complexity results from the algorithm's need to store information for each parking spot, which makes the space complexity linearly dependent on the number of parking spots and the features for each parking spot.

Results And Discussion:

Obtained Result:

Final Product:

The final product is a user-friendly blockchain-powered parking system designed to revolutionize the way parking is managed and experienced. This innovative solution harnesses the power of technology to address various challenges associated with traditional parking systems, offering a seamless and secure experience for both parking providers and users.

Outcome:

1. Improved Efficiency and Reduced Waiting Times:

The implementation of real-time availability tracking and seamless integration with navigation apps significantly improves the efficiency of the parking system. Users can quickly identify and navigate to available parking spaces, reducing the time spent searching for parking. This streamlined process contributes to a more efficient overall urban mobility system, minimizing congestion and waiting times.

2. Improved Recommendation process:

The implementation of the recommendation algorithm considers various factors that are crucial for optimal spot recommendations such as distance, cost, security and availability.

3. Enhanced Security in Parking Transactions:

The use of blockchain technology ensures a secure and tamper-resistant environment for parking transactions. Each transaction is recorded on an immutable smart contract. This enhanced security reduces the risk of unauthorized access, fraud, and manipulation, instilling confidence in both parking providers and users.

4. Enhanced User Experiences, Leading to Higher Satisfaction:

The user-centric design,coupled with real time availability information and flexible rental packages, contributes to an overall enhanced user experience. Drivers can easily find, reserve, and pay for parking spaces, creating a hassle-free and satisfying parking journey. The user-friendly interface, coupled with the system's efficiency, promotes positive interactions and fosters higher satisfaction among both individual and corporate users.

In conclusion, the outcomes of the blockchain-based parking system directly impact the efficiency, security, and user satisfaction within the urban parking landscape, aligning with the broader goals of creating a smarter, more secure, and user-friendly city infrastructure.

In-depth analysis of various recommendation algorithms:

1. Rank Aggregation (Borda Count)

- **Description:** This method combines the points of each criterion into a single aggregated rating.

- **Time Complexity:** O(N×M)
  - Explanation: Here, N is the number of spots and M is the number of features. Each rating needs to be processed for every spot, leading to the time complexity of O(N×M)).
- **Space Complexity:** O(N)
  - Explanation: The aggregated scores for each spot need to be stored along with the features leading to space complexity of O(NxM).

2. Decision Rules

- **Description**: This method ranks the spots in each criterion based on specified constraints. It can have multiple rules for making decisions.

- **Time Complexity**: O(N×M)
  - Explanation: Each spot needs to be evaluated against all rules, leading to the time complexity of O(N×M)where N is the number of spots and M is the number of rules.
- **Space Complexity**: O(N)
  - Explanation: The space needed is proportional to the number of spots that meet the criteria, hence linear in terms of the number of spots.

3. Min-Max Weighted Sum

- **Description**: This method calculates normalized values and assigns weights for the calculation of the final weighted sum.

- **Time Complexity**: O(N×M)
  - Explanation: Similar to the previous methods, each spot is evaluated based on multiple criteria (parameters), leading to the time complexity of O(N×M).
- **Space Complexity**: O(N)
  - Explanation: Scores for each spot need to be stored, leading to linear space complexity in terms of the number of spots.

Comparison of results generated by different algorithms:

Example Data set

Spots	Distance	Cost	Availability	Security
1	1	8	2	CCTV
2	1.5	6	4	GUARD
3	2	12	1	INDOOR,GUARD
4	0.8	10	3	CCTV,GUARD,INDOOR
5	1.2	9	2	CCTV,INDOOR
6	1.8	7	5	GUARD
7	1.3	11	1	CCTV
8	0.9	8	4	INDOOR,GUARD

Borda Count Recommendation

1. Rank Each Spot for Each Criterion

Distance (Lower is Better)

1. Spot 4 (0.8)
2. Spot 8 (0.9)
3. Spot 1 (1.0)
4. Spot 5 (1.2)
5. Spot 7 (1.3)
6. Spot 2 (1.5)
7. Spot 6 (1.8)
8. Spot 3 (2.0)

Cost (Lower is Better)

1. Spot 2 ($6)
2. Spot 6 ($7)
3. Spot 1 and Spot 8 ($8) - tied
4. Spot 5 ($9)
5. Spot 4 ($10)
6. Spot 7 ($11)
7. Spot 3 ($12)

Availability (Higher is Better)

1. Spot 6 (5 spots)
2. Spot 8 and Spot 2 (4 spots) - tied
3. Spot 4 (3 spots)
4. Spot 1 and Spot 5 (2 spots) - tied
5. Spot 3 and Spot 7 (1 spot) - tied

Security (More Features is Better)

1. Spot 4 (CCTV, GUARD, INDOOR)
2. Spot 3 (INDOOR, GUARD)
3. Spot 5 (CCTV, INDOOR)
4. Spot 8 (INDOOR, GUARD)
5. Spot 7 (CCTV)
6. Spot 1 (CCTV)
7. Spot 6 and Spot 2 (GUARD) - tied

2. Assign Points Based on Ranks

For N spots, the points are assigned as follows:

points=N−rank+1

where N= Number of spots

Distance Points

1. Spot 4: 8−1+1=8
2. Spot 8: 8−2+1=7
3. Spot 5: 8−4+1=5
4. Spot 7: 8−5+1=4
5. Spot 2: 8−6+1=3
6. Spot 1: 8−3+1=6
7. Spot 6: 8−7+1=2
8. Spot 3: 8−8+1=1

Cost Points

1. Spot 2: 8−1+1=8
2. Spot 6: 8−2+1=7
3. Spot 1: 8−3+1=6
4. Spot 8: 8−3+1=6
5. Spot 5: 8−5+1=4
6. Spot 4: 8−6+1=3
7. Spot 7: 8−7+1=2
8. Spot 3: 8−8+1=1

Availability Points

1. Spot 6: $8-1+1=8$
2. Spot 8: $8-2+1=7$
3. Spot 2: $8-2+1=7$
4. Spot 4: $8-4+1=5$
5. Spot 1: $8-5+1=4$
6. Spot 5: $8-5+1=4$
7. Spot 3: $8-7+1=2$
8. Spot 7: $8-7+1=2$

Security Points

1. Spot 4: $8-1+1=8$
2. Spot 3: $8-2+1=7$
3. Spot 5: $8-3+1=6$
4. Spot 8: $8-4+1=5$
5. Spot 7: $8-5+1=4$
6. Spot 1: $8-6+1=3$
7. Spot 6: $8-7+1=2$
8. Spot 2: $8-7+1=2$

3. Sum Points Across All Criteria

Summing the points for each spot across all criteria:

Spot 1

- Distance: 6
- Cost: 6
- Availability: 4
- Security: 3
- **Total: 19**

Spot 2

- Distance: 3
- Cost: 8
- Availability: 7
- Security: 2
- **Total: 20**

Spot 3

- Distance: 1
- Cost: 1
- Availability: 2
- Security: 7
- **Total: 11**

Spot 4

- Distance: 8
- Cost: 3
- Availability: 5
- Security: 8
- **Total: 24**

Spot 5

- Distance: 5
- Cost: 4
- Availability: 4
- Security: 6
- **Total: 19**

Spot 6

- Distance: 2
- Cost: 7
- Availability: 8
- Security: 2
- **Total: 19**

Spot 7

- Distance: 4
- Cost: 2
- Availability: 2
- Security: 4
- **Total: 12**

Spot 8

- Distance: 7
- Cost: 6
- Availability: 7
- Security: 5
- **Total: 25**

Final Ranking (Borda Count):

The final Borda count ranking from best to worst is:

1. **Spot 8**: 25 points
2. **Spot 4**: 24 points
3. **Spot 2**: 20 points
4. **Spot 5**: 19 points
5. **Spot 6**: 19 points
6. **Spot 1**: 19 points
7. **Spot 7**: 12 points

8. **Spot 3**: 11 points

Decision Rules Recommendation

Distance-first Decision Rule:

- Spot 4: 1st
- Spot 8: 2nd
- Spot 5: 3rd
- Spot 7: 4th
- Spot 2: 5th
- Spot 6: 6th
- Spot 1: 7th
- Spot 3: 8th

Cost-first Decision Rule:

- Spot 2: 1st
- Spot 6: 2nd
- Spot 1 and Spot 8: 3rd (tied)
- Spot 5: 5th
- Spot 4: 6th
- Spot 7: 7th
- Spot 3: 8th

Availability-first Decision Rule:

- Spot 6: 1st
- Spot 8 and Spot 2: 2nd (tied)
- Spot 4: 4th
- Spot 1 and Spot 5: 5th (tied)
- Spot 3 and Spot 7: 7th (tied)

Security-first Decision Rule:

- Spot 4: 1st
- Spot 3: 2nd
- Spot 5: 3rd
- Spot 8: 4th
- Spot 7: 5th
- Spot 1: 6th
- Spot 6 and Spot 2: 7th (tied)

Aggregating Scores:

Now, summing up the rankings for each spot:

- Spot 4: 1 + 6 + 4 + 1 = 12
- Spot 8: 2 + 3 + 2 + 4 = 11
- Spot 2: 5 + 1 + 2 + 7 = 15
- Spot 6: 6 + 2 + 1 + 7 = 16
- Spot 5: 3 + 5 + 5 + 3 = 16
- Spot 1: 7 + 6 + 5 + 6 = 24
- Spot 7: 4 + 7 + 7 + 5 = 23
- Spot 3: 8 + 8 + 7 + 2 = 25

Final Ranking (Decision rules):

The decision rules ranking from best to worse is:

1. Spot 8 (Total points: 11)
2. Spot 4 (Total points: 12)
3. Spot 2 (Total points: 15)
4. Spot 5 (Total points: 16)
5. Spot 6 (Total points: 16)
6. Spot 7 (Total points: 23)
7. Spot 1 (Total points: 24)
8. Spot 3 (Total points: 25)

Min-Max Weighted Sum Recommendation

<u>Given Weights:</u>

- Distance Weight: distance=0.3
- Cost (Price) Weight: cost=0.3
- Availability Weight: availability=0.2
- Facilities (Security) Weight: facilities=0.2

<u>Formulas:</u>

- Norm : (value-min)/(max-min)
- Score : summation(Norm x Weight)

<u>Step-by-Step Calculation for Each Spot:</u>

Spot 1:

- Distance: 1
- Cost: 8
- Availability: 2
- Security: CCTV

Normalized Distance: = (1-0.8)/(2-0.8) = 0.25

Normalized Cost: (8-6)/(12-6) = 0.333

Normalized Availability: (2-1)/(5-1) = 0.25

Facilities (Security) Norm: 1/3

<u>Total Score:</u> (0.3×0.25)+(0.3×0.333)+(0.2×0.25)+(0.2×31) = 0.291

Spot 2:

- Distance: 1.5
- Cost: 6
- Availability: 4
- Security: GUARD

Normalized Distance: 0.75

Normalized Cost: 0

Normalized Availability: 0.75

Facilities (Security) Norm: 1/3

<u>Total Score:</u> 0.441

Spot 3:

- Distance: 2
- Cost: 12
- Availability: 1
- Security: INDOOR, GUARD

Normalized Distance: 1

Normalized Cost: 1

Normalized Availability: 0

Facilities (Security) Norm: 2/3

<u>Total Score:</u> 0.733

Spot 4:

- Distance: 0.8
- Cost: 10
- Availability: 3
- Security: CCTV, GUARD, INDOOR

Normalized Distance: = 0

Normalized Cost: 0.666

Normalized Availability: 0.5

Facilities (Security) Norm: 1

Total Score: 0.500

Spot 5:

- Distance: 1.2
- Cost: 9
- Availability: 2
- Security: CCTV,INDOOR

Normalized Distance: 0.5

Normalized Cost: 0.5

Normalized Availability: 0.25

Facilities (Security) Norm: 2/3

Total Score: 0.483

Spot 6:

- Distance: 1.8
- Cost: 7
- Availability: 5
- Security: GUARD

Normalized Distance: 0.833

Normalized Cost: 0.166

Normalized Availability: 1

Facilities (Security) Norm: 1/3

Total Score: 0.6916

Spot 7:

- Distance: 1.3
- Cost: 11
- Availability: 1
- Security: CCTV

Normalized Distance: 0.625

Normalized Cost: 0.833

Normalized Availability: 0

Facilities (Security) Norm: 1/3

Total Score: 0.504

Spot 8:

- Distance: 0.9
- Cost: 8
- Availability: 4
- Security: INDOOR, GUARD

Normalized Distance: 0.125

Normalized Cost: 0.333

Normalized Availability: 0.75

Facilities (Security) Norm: 2/3

Total Score: 0.420

Final Ranking (Min-Max Weighted Sum):

Spot 1: Total Score = 0.291

Spot 8: Total Score = 0.420

Spot 2: Total Score = 0.441

Spot 5: Total Score = 0.483

Spot 4: Total Score = 0.500

Spot 7: Total Score = 0.504

Spot 6: Total Score = 0.691

Spot 3: Total Score = 0.733

These spots have been ranked based on a detailed analysis using normalized values and assigned weights for factors such as distance, cost, availability, and security. This strategic approach ensures the most efficient and optimal parking recommendation to users.

Conclusion:

This project successfully addresses the challenge of monetizing and optimizing the utilization of privately-owned empty parking spots to reduce street parking congestion. By developing a blockchain-powered parking platform with dynamic pricing and secure transaction methods, the project improved efficiency and user satisfaction in urban parking management. Key contributions include enhanced security through blockchain technology, reduced time spent searching for parking(recommendation system), and a user-friendly interface that caters to user needs.The significance of these results lies in the potential application of this system in cities facing severe parking issues, offering a scalable and secure solution. Furthermore, the project also identified limitations such as high blockchain transaction costs and the need for more personalized recommendation systems.

Future work will focus on reducing these transaction costs, enhancing recommendation algorithms, expanding system integration, integration of various payment methods and ensuring scalability to different urban environments. This project demonstrates a promising step towards smarter, more efficient, and secure urban parking solutions.

Appendix

CO	CO Descriptions	Knowledge Profile (K)	Complex Engineering Problem (EP)
CO7	**Assess** and **address** the sustainability and impact of the capstone project in societal and environmental contexts	**Sustainability and Impact**: The project reduces traffic congestion and emissions by optimizing parking space usage, contributing to environmental sustainability. It provides economic benefits by enabling monetization of unused spaces and improves quality of life by reducing parking search times. **Ethical Responsibility**: Ensures secure and transparent transactions using blockchain, protecting user data and maintaining public trust.**[K7]**	**EP2 : Range of Conflicting Requirements :** The project addresses conflicts between security/transparency (blockchain) and user-friendliness, as well as balancing dynamic pricing for affordability and profitability. **EP5: Extent of Applicable Codes :** The project operates beyond traditional engineering codes due to its innovative use of blockchain, adhering to blockchain security and data privacy principles while complying with local parking regulations. **EP6: Extent of Stakeholder Involvement and Conflicting Requirements :** Involves diverse stakeholders such as parking spot owners, renters, urban planners, and authorities. Addresses their conflicting needs through features like dynamic pricing, real-time tracking, and navigation app integration. Stakeholder surveys ensure the platform meets varied requirements effectively.
CO8	**Apply** professional and engineering ethical principles and practices for the implementation of the capstone project	The project applies professional and engineering ethical principles by ensuring user data privacy and security through blockchain technology, which provides secure, tamper-proof transactions. Transparency and trust are fostered via smart contracts that guarantee clear, immutable terms. Dynamic pricing models balance affordability for users with profitability for owners, ensuring fairness. The platform enhances public safety by reducing traffic congestion and potential accidents through optimized parking usage and real-time tracking. It supports sustainability by minimizing the need for new parking infrastructure and lowering emissions, adhering to environmental standards. Additionally, the project complies with local parking regulations, demonstrating professional integrity while innovatively implementing blockchain technology.**[K7]**	

CO9	**Work** effectively as an individual and a team member for successful completion of the capstone project	The project upholds ethical principles by ensuring data privacy and security through blockchain technology, which guarantees tamper-proof transactions. Transparency and trust are maintained with smart contracts, while dynamic pricing ensures fairness between users and parking spot owners. By reducing traffic congestion and accidents through real-time parking optimization, the project enhances public safety. It supports sustainability by minimizing new infrastructure needs and lowering emissions, aligning with environmental standards. Furthermore, the project complies with local regulations, showcasing professional integrity in the innovative use of blockchain technology.	
CO10	**Write** effective reports and design documentation, and **make** effective presentations of the outcome of the capstone project		**EA1: Range of Resources :** Utilizes diverse resources including Google Maps, MySQL, PHP, HTML, CSS, JavaScript, Solidity, and stakeholder surveys. **EA2: Level of Interaction :** Addresses complex interactions between technical, engineering, and societal issues by integrating blockchain with web development and ensuring regulatory compliance. **EA3: Innovation :** Showcases innovative use of blockchain for decentralized parking solutions with features like smart contracts, real-time tracking, dynamic pricing, and peer-to-peer sharing. **EA4: Consequences to Society and the Environment :** Reduces traffic congestion and emissions, promotes efficient use of urban space, and enhances public safety, addressing significant societal and environmental impacts. **EA5: Familiarity :** Operates beyond traditional engineering standards by using blockchain, while ensuring compliance with legal and regulatory

			standards in parking and urban management.
CO11	**Conduct** economic analysis and cost estimation; and **apply** appropriate project management processes in the development life cycle of the capstone project	In the development life cycle of the capstone project, an economic analysis and cost estimation were conducted to ensure financial viability. The project utilized cost-effective resources such as open-source software (MySQL, PHP, Solidity) and free data sources (Google Maps). Dynamic pricing models were analyzed to maximize profitability for parking spot owners while maintaining affordability for users. Project management processes, including agile methodologies, were applied to streamline development, ensure timely delivery, and manage resources efficiently. Regular stakeholder meetings and iterative feedback loops were employed to align the project with user needs and market demands, ensuring a cost-effective and well-managed development process.	
CO12	**Prepare** to take part in independent and life-long learning for adapting emerging technologies for the solution of the complex computer science and engineering problems	The capstone project emphasizes the importance of independent and lifelong learning to adapt to emerging technologies. By integrating blockchain, smart contracts, and real-time data tracking into the parking solution, the project showcases a commitment to staying updated with cutting-edge advancements in computer science and engineering. Continuous learning was facilitated through research, online courses, and practical experimentation with new technologies. This approach not only solved complex engineering problems but also prepared the team for future challenges, fostering an environment of ongoing education and technological adaptation.	

References:

[1] S. Jennath et al.,S.Adarsh,Nikhil V.Chandran,R.Ananthan,A.Sabir and S.Asharaf "Parkchain: A blockchain powered parking solution for Smart Cities," Frontiers, vol. 2,2023, doi:10.3389/fbloc.2019.00006

[2] J. Hu, D. He, Q. Zhao and K. -K. R. Choo, "Parking Management: A Blockchain-Based Privacy-Preserving System," in IEEE Consumer Electronics Magazine, vol. 8, no. 4, pp. 45-49, July 2019, doi: 10.1109/MCE.2019.2905490.

[3] S. Ahmed, Soaibuzzaman, M. S. Rahman and M. S. Rahaman, "A Blockchain-Based Architecture for Integrated Smart Parking Systems," 2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), Kyoto, Japan, 2019, pp. 177-182, doi: 10.1109/PERCOMW.2019.8730772.

[4] C. Zhang et al., "BSFP: Blockchain-Enabled Smart Parking With Fairness, Reliability and Privacy Protection," in IEEE Transactions on Vehicular Technology, vol. 69, no. 6, pp. 6578-6591, June 2020

[5] 1.M. M. Badr, W. A. Amiri, M. M. Fouda, M. M. E. A. Mahmoud, A. J. Aljohani and W. Alasmary, "Smart Parking System With Privacy Preservation and Reputation Management Using Blockchain," in IEEE Access, vol. 8, pp. 150823-150843, 2020, doi: 10.1109/ACCESS.2020.3016945.