International Journal of Computer Applications (0975 - 8887)
Volume 187 - No.26, July 2025

Carp: An Easy Solution to Avoid Parking Hazards

Ankana Chakraborty
Sister Nivedita University
DG Block, Action Area I, 1/2 New Town, Chakpachuria, West Bengal-700156

Moumita Shee
Sister Nivedita University
DG Block, Action Area I, 1/2 New Town, Chakpachuria, West Bengal-700156

Aishwaria Roy
Sister Nivedita University
DG Block, Action Area I, 1/2 New Town, Chakpachuria, West Bengal-700156

Tanni Saha
Sister Nivedita University
DG Block, Action Area I, 1/2 New Town, Chakpachuria, West Bengal-700156

Rima Bachar
Sister Nivedita University
1DG Block, Action Area I, 1/2 New Town, Chakpachuria, West Bengal-700156

Soma Datta
Sister Nivedita University
DG Block, Action Area I, 1/2 New Town, Chakpachuria, West Bengal-700156

ABSTRACT

Even though there are plenty of parking spaces available, the growing number of cars has made parking a major concern in urban areas these days. It occurs because people are unaware of the parking spaces that are close by and available. The Car Parking Management System that the authors have suggested may be a clever and effective way to deal with this issue. This system offers users a smooth parking experience by utilizing mobile applications and contemporary technologies. It makes it simple for drivers to find open parking spaces, make reservations, and pay. It also helps to save time and lessen traffic. Additionally, it will assist parking space suppliers in making efficient use of their available space and generating revenue. Those who wish to manually reserve a spot in real-time will also find the authors' system useful. The parking data in this system will either be updated manually or automatically. A shared web service will be used for the authorization and authentication process, and the system will be linked to a mobile server so that users can access pertinent data via the mobile app. The design, app flow, main features, and advantages of the suggested parking system are covered in this article, which also emphasizes how it transforms conventional parking management into a more efficient and user-friendly solution. The accuracy of the proposed technique is 95.78%.

Keywords

Car Parking, Pre-booking, Real-time reservation, Haversine distance measuring, Location, LPN, Google Maps API, Flutter

1. INTRODUCTION

Now everyone is living in a fast-paced world where everyone is moving forward. Almost every household in busy urban areas owns a vehicle that saves time and provides a hassle-free journey. But finding a place to park in the busy overcrowded areas, such as when going to watch a football match of Cristiano Ronaldo vs. Lionel Messi in a stadium with thousands of people, finding a place to park can be a hassle and a stressful experience. In a metropolitan area, on a Monday morning while going to work or office meetings, finding a place to park can also pose a serious challenge, with loss of time, energy, and resources. Lack of knowledge about availability of car parking zones in the city of Shillong and its suburbs is a real-time problem scenario of the car parking hassle, leading to traffic congestion. It is estimated that in urban cities, nearly 30% of vehicle drivers are looking for parking places and the average search time for each vehicle is 8.1 minutes (Shoup, 2006). The inefficient search for parking significantly adds up to the traffic congestion as well as wastes the driver's time. It increases the use of gas that can lead to increased air pollution. In the Chicago metropolitan area, the search for parking contributes annually to approximately 39 million extra vehicle miles, the consumption of 1.9 million gallons of fuel, and the release of over 29,000 tons of CO2 emissions (Ayala et al., 2012).

Meanwhile, a large number of parking resources remain underutilized due to fluctuations in supply and demand across different times and locations. A considerate percentage of private parking spaces are found to be vacant during the daytime like the Beijing metropolitan area (Yan et al., 2021). Keeping this in mind, the authors have come up with a solution ParkCle to tackle the search for parking places in real time. ParkCle offers real-time insights

into parking availability at your preferred location or locations near you. It will reduce your search time and save your time, energy and resources. You can either prebook your parking space or book it at an instant for a stipulated point of time. Payment method is hassle free, using our payment gateway. A minimal extra charges need to be paid for extra time than booked hours as per the rules and regulations of that particular zone.
Reservation for parking is a new generation parking feature that ensures pre-booked parking spaces for drivers and helps to reduce or eliminate competitions for parking spots effectively. By minimizing the need to search for a parking space, this system not only decreases traffic and emissions but also gains driver support by providing perks like assured parking space, shorter queues and minimized waiting times. A series of decision-making and optimization challenges along with various approaches and considerations constitute this process. Drawing from the present published works, the authors examine every research area—stock management, distribution strategies, booking rates, and efficiency evaluation—individually in the following sections.

1.1 Major Contributions

This research work aims to reduce your search time, save your energy and resources during car parking as well as giving facilities such as pre-booking or real time booking with hassle free payment method. The authors' contribution can be summarized as follows-

—The proposed method has a separate sign up or login option for the parking spot providers. Parking spot providers can offer spaces for the duration of their choice and make the best use of their free spaces. The parking fee will be determined based on the standard fees of that particular area.

—It has another option to sign up or log in as a user. Users must register on the platform with proper documents such as Name, E-Mail, Contact number, and Driving License number. The documents are verified for their authenticity and only after proper and successful verification can the user sign in. Then the user can find parking spots and pre-book spaces according to their needs as well as book the spot manually. They have to verify their booking details while parking their car by showing the verification code. If users park their car for longer than the booked time, they have to pay the extra according to the standard rules. They can cancel their booking within a particular time. Otherwise, they have to pay a minimal amount for cancellation.

—In the proposed system, a server has been established to provide business customer services within the application environment. The system will access parking data which can be updated automatically or manually by an administrator, ensuring synchronization with the database. The application will also be connected to a mobile server to access smartphone data, providing users with relevant information through the mobile app. It will also synchronize with the backend system of our smart parking finder app. A shared web service will take care of the authorization and authentication process.

These contributions highlight their efforts to reduce or eliminate parking hassles and make car parking easy, less time consuming and safer.

1.2 Section Organizations

This paper contains mainly four sections. Section 1 describes the introduction of the proposed work. Section 2 contains the literature survey along with its limitations. Section 3 describes the detail design of the proposed technique and Section 4 shows the performance analysis and the comparative study between the existing method and the proposed technique.

2. LITERATURE SURVEY OF THE EXISTING METHODS ON SAME DOMAINS

Numerous approaches have been developed to enhance car parking systems. An analysis of these existing solutions reveals that they often require human intervention to operate more effectively. One of these existing systems is- Hilal Al-Kharusi and Ibrahim Al-Bahadly developed an intelligent parking management system based on image processing. This study develops a clever image processing-based system for detecting parking spaces. The images of the parking space are used to find the vacant spaces in the parking lot by capturing the images from a particular camera which helps in finding several cars at the same time. Evaluating different parking areas is also easy. After these images have been processed, the system is able to inform the user through displayed information on the available parking bays. It reduces the time spent by the user in looking for a vacant parking bay. The system has been developed on both software and hardware platforms [1]. Another Existing method is -Intelligent Parking Management System by Madhavi K.R., Amrutha S., Nagamani N., Anusha J., Prof. Ruckmani Divakaran. This paper introduces an intelligent parking space detection system that employs a color detection algorithm for image processing. The system is equipped with a camera that acts as a sensor and captures photographs of a particular area of interest. The available spaces are marked with green circular dots. The nearest available parking slot number is displayed to drivers on an LCD panel at the entrance [9]. Another reference is- Determining car-park occupancy from single images by S. Funck, N. Mohler, W. Oertel. This document proposes an approach to estimate car park occupancy using only one image taken by a single camera. Surveillance CCTV cameras are already fitted in most parking lots, which can be used for automatic detection. The proposal is most suitable for situations where rough estimates of occupancy can be provided. The imaging technique used considers the input image and first creates a reference image of the car park when it is known to be empty before subtracting the two. Occupancy estimate is performed by computing the ratio of vehicle pixels to the total area of the car park after correcting for perspective and blockage [6]. In [12], researchers have focused on a wireless vehicle management system, providing a comprehensive account of the design and construction of the system integrating embedded systems, GPRS, and vehicle number plates recognition technology developed with the aid of the ARM9 microprocessor. The hardware comprise an ARM based embedded system, a video capture and compression module, a GPRS Module. The software part consists of an Embedded Linux, MPEG4 encoding, CGI programming and automatic license recognition algorithms. Images are captured, vehicle plates are recognized, and the data are sent to the remote center via GPRS for efficient vehicle monitoring and control system over a long range. Applications incorporate accessing vehicles, management of parking zones and supervision of traffic . A prototype of the system was implemented using the notes of crossbow [11]. Another example of a parking system is the Intelligent Parking Space Detection System Based on Image Processing developed by R. Yusnita et. al. This study presents an enhanced parking spot detection system that recognizes the brown rounded markings on the parking lot floor using a combination of computer vision and image processing techniques. The information processed by the system display availability of free parking spaces which is shown on real time on the

seven-segment display unit. This proposed system has been implemented in software and hardware designs to ensure effective working of the system [13]. Realtime car parking system using image processing by Ms. Sayanti Banerjee, Ms. Pallavi Choudekar and Prof. M. K. Muju. The system counts the number of cars parked and stores the images of empty bays captured by a camera installed at the entrance of the parking lot. The system captures images in a chronological order and processes the images using the Prewitt operator for edge detection technique to determine if it contains the image of a vehicle similar to the reference image of the vehicle. Then it matches the succeeding images based on the percentage of image overlap and provides incoming motorists with guidance and information based on which images model the parking area [3].Further research includes- Wang et al.'s exploration of vehicle automation in parking[8] and Geng and Cassandras' infrastructure-based system for optimizing parking space utilization[7]. A RESTful Java based application hosted by a Central Server is enhanced with Google Cloud Messaging (GCM) to monitor alerts related to violations such as inappropriate use of reserved spaces, exceeding allowable parking duration, which in turn alerts the traffic police through a designated mobile application[10].Tayo Fabusuyi Tayo Fabusuyi, Robert C. Hampshire, and Victoria Hill evaluated the efficiency of parking systems in urban areas, emphasizing user satisfaction and reduced congestion[5].Aaesha, AlRefaei's iValet system highlighted real-time reservation features [2], while Mathias Gabriel Diaz Ogás, Ramon Fabregat, Silvana Aciar provided a comprehensive survey on the technologies and frameworks of smart parking systems[4]. A proof-of-concept has been developed and tested to confirm that the proposed solution can meet the real requirements Smart Parking System.

3. DETAILED DESIGN

The authors have designed a workflow to accurately evaluate key data points of parking spot availability, parking durations, secure payment method to ensure efficient real-time parking management. False detection could lead to incorrect availability updates, it needs to be minimized. Figure 1 illustrates a clear and detailed illustration of these sequential steps, highlighting how each operation contributes to the overall functionality of the system. Moreover, by alerting owners or governments to infractions and non-authorized parking, traffic flow is made stable and optimal space utilization.

3.1 Flowchart

Fig. 1 represents the flowchart of the authors' proposed parking management system app. The flowchart illustrates the process of the parking spot finder system named "ParkCle". The process starts with the registration and login process. Parking providers need to sign in using Proper data(Name, Email id, Phone Number). After successful verification of the data, providers need to give their car parking spot details with mentioning the charges of the spot. New users must simultaneously register with accurate information (user name, DL number, and phone number), which will be verified and stored in our database. Users can browse parking spaces, make reservations in advance, and locate spots instantly after logging in. The user must finish the process by choosing the type of vehicle, entering the LPN number, and entering the parking length after locating a site. If users go over the allotted time, they will be charged more for additional time. They can also take use of the hassle-free parking facilities and cancel the reservation within a certain amount of time.

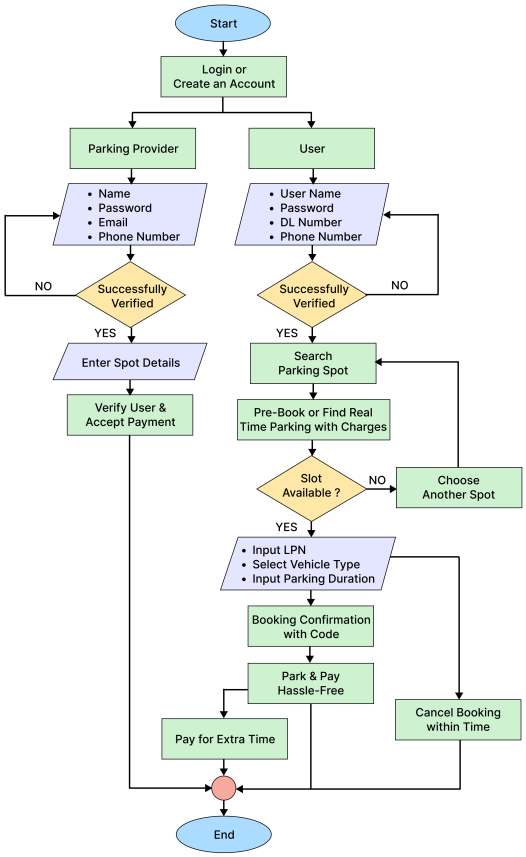

Fig. 1: Flowchart of Parking Management System

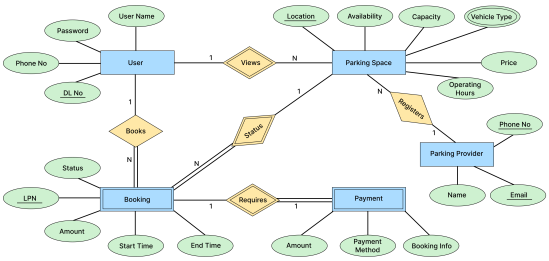

Fig. 2: ER Diagram

3.2 Entity Relationship Diagram

This ER diagram illustrates a system designed to find available Parking space. User, Parking Space, Booking and Payment are four entities. The User is identified with attributes by Username, Password, Phone No, and uniquely identified by a Driving License Number (key attribute). Users can view multiple Parking Spaces (View) which hold location (key attribute), availability, capacity, price, operating hours and vehicle type (multi-valued attribute) and make multiple Bookings (Books). The Booking entity records License Plate Number (key attribute), booking status, amount, stating and ending time linking users to Parking places (Status). Booking needs to pay parking charges (Requires), the Payment en-

Table 1. : Limitations associated with specific existing methods. [10, 13, 6, 3, 11, 12]

Sl No.	Used Techniques	Accuracy	Limitation
1	IEEE 802.15.4 Wireless Sensor Network (WSN) technology and Ultra-High Frequency (UHF) Radio Frequency Identification (RFID) are combined in the parking system.	A reliable level of performance.	The system's performance may be affected by environmental conditions such as weather, which can impact RFID and WSN signals.
2	Morphological processing, HSV conversion for background removal, and black-and-white image conversion.	Accuracy level is moderate.	The ability of current processing techniques to handle photos in low visibility conditions may be limited.
3	Image subtraction technique is used to compare current and reference images to estimate occupancy.	Reasonable occupancy estimation with CCTV cameras.	Assumes initial empty state of the parking lot; limited by image quality and perspective distortions.
4	Prewitt operator is used for edge detection to identify vehicles and vacant spaces.	Moderate accuracy obtained in counting cars and detecting vacancies.	Sensitive to image quality and vehicle overlap.

This table highlights the limitations of several existing vehicle detection and parking management techniques.

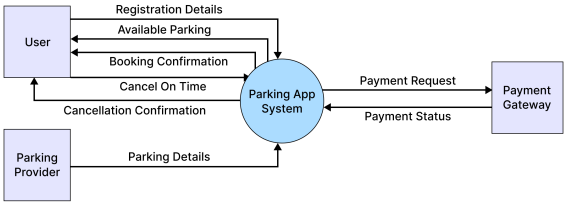

Fig. 3: DFD Level 0

tity includes amount, booking info and payment method. Parking Provider entity with details such as name, email (key attribute), phone no enter the Parking Space details through the (Registers) relationship. This structured design aids in functionalities such as structured database supporting real time user interaction, efficient booking management and secure payment processing thereby making it suitable for smart city applications.

3.3 Data Flow Diagram

The parking app system DFD Diagram provides a high level overview of the system showing the main flow in a time effective way.

DFD Level 0 : The diagram represents the parking app system as a single process interacting with three external entities named User, Parking Provider and Payment Gateway. It shows the data flow focusing on the inputs and outputs between the system and external entities. Parking providers send parking spot details to the system. The system takes the registration details from the User and sends the available parking details to the User to book a spot. After booking process completion, System sends confirmation notification to User. If the user cancels the booking within time, the system sends a cancellation confirmation notification. During payment, the system generates a payment request and receives payment status details from designed payment gateway.

DFD Level 1 : Level 1 Data Flow Diagram for the parking system describes how the User interacts with processes such as Registration/Login, Viewing Parking, Booking, Cancel Booking, Payment, and Update Parking. These processes interact with four data stores such as User Database, Parking Database, Booking Database and Payment Database to handle user details, parking availability, booking information, and payments. The Parking Provider updates parking availability and the Payment Gateway processes transactions. These data flow indicates smooth functioning across all constituents allowing effective parking management.

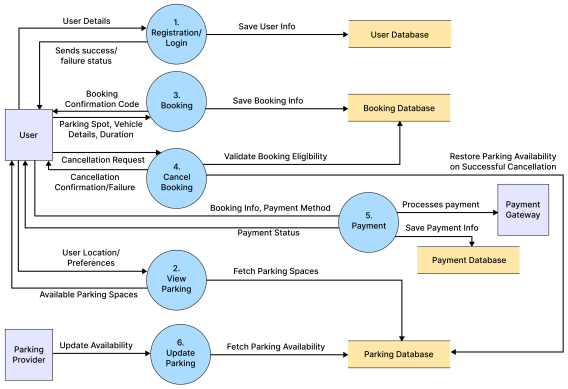

Fig. 4: DFD Level 1

DFD Level 2 : The Parking App System Level 2 diagram describes the Figure of parking app system process, showing the subprocesses - Validate user, Register new user, Verify space availability, Save booking details, Cancel booking on a time, Process payment, Save payment details, Update parking availability which all have the input and output data flows. It shows the full data stream function between external entities and databases. The diagram shows how each operation contributes to the overall functionality of the system and provides clear and detailed information.

3.4 Technologies Used

1. Flutter - used to create cross platform apps that work for both android and iOS.
2. Firebase - used for authentication and login database.
3. Google Maps API - with Places, maps for android and iOS.

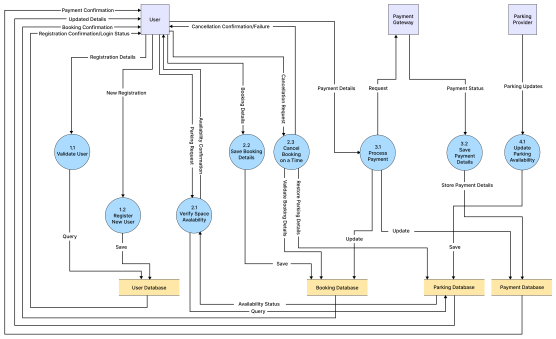

Fig. 5: DFD Level 2

4. Dependencies and flutter packages : used to create cross platform apps that work for both android and iOS.
a. Geolocator - It is used to determine the device's last known location, current location, and continuous location updates; to determine whether location services are enabled on the device; to determine the bearing between two geo-coordinates; and to determine the distance (in meters) between two geo-coordinates. **b. HTTP** - Used to make individual HTTP requests with minimal hassle and make it easy to consume HTTP resources.
c. Google Maps Flutter - For using Google Maps in the application.
d. Provider - Used for in-app notifications for any changes in space.
e. URL Launcher - Used to handle external services such as payment gateways, ticket creation, and so on.

3.5 Approach

1. App Flow : The flow of the app begins with the registration and login process. The new users need to register with a proper kyc and driving license which is then saved in our database. Returning users can sign in directly. On successful registration or login the user is taken to the main menu where they can either view and select the available parking space in their vicinity or search for parking spots in the location of their choice and book the place. For booking the place the user needs to first select the type of vehicle, enter its number plate, and select the amount of time for which they want to park. After which a code will be generated to ensure successful booking and the code is sent to the phone with which the app is registered. The verification code is present both in the app as well as sent to the phone as a message. The user needs to show and verify the code during the time of parking. Booking fees will be the same as the standard fees of that particular place. In case the user needs to park for more than the given time, while exiting they need to pay the extra amount as per the standard rules. The application integrates with web services and APIs (Application Programming Interfaces) to interpret protocols effectively. A server has been established to deliver business customer services within the application environment. The system will then access parking data, which can be automatically updated or manually modified by an administrator, ensuring synchronization with the database. Additionally, the application will connect to a mobile server to access smartphone data, providing users with relevant information through the mobile app. It will also synchronize with the backend system of the smart parking finder, allowing conventional transactions to continue for manual users who book parking without using the app.
A shared web service will take care of authorization and authentication, providing security access to all users dealing with the parking finder system. The backend server will play a critical role in enabling the exchange of parking space information between customers and parking providers to enable the uploading and downloading of data.
2. Elements Present : The allocation of reservations focuses on pairing drivers with lots or spaces to minimize repeated cruise. The auctioning of reservations must contend with several allocations, which reflect varying optimization objectives. Some of these considerations-their drivers, their vehicles, and the parking lots-are demand and supply variables for the reserved spaces. Further, traffic conditions greatly impact travel costs for drivers, and hence the details of the traffic situation must be considered with care. Key aspects of reservation allocation can be broadly categorized into three areas, which are reviewed below. On the driver/vehicle side, parking requests typically include fundamental elements such as origin, destination, arrival/start time, departure/end time, and the duration of parking. Along with the basic requirements, many individual preferences can be validly included to suit particular requirements. Typical considerations would include driver type (commuter or occasional traveler), reservation techniques (real-time or shared-time reservations), needed level of security, highest permissible driving limit or walking distances, highest waiting times, and charged price ceilings. Such preferences are often brought out in the literature as key leverage points to enable better parking reservation systems to be personalized and made more effective. Driver type is determined by the purpose of the trip, while reservations are dictated by the kind of allocation. A dynamic form of resource allocation is possible by Real-time reservation (RTR), as drivers get closer to their destination, they are continuously assigned an optimal parking space. On the other hand, share-time reservation (STR) is a static allocation method that uses a precard of reactivation times: drivers select an unoccupied parking space preferred by them in their respective duration of time in the future. Parking lot elements are associated with attributes such as location, capacity, cost, and other features. Factors related to location, such as proximity to popular destinations, influence the distances drivers need to walk or drive. Key factors in the procedure of creating and refining a model include the capacity of the parking area, its instantaneous accessibility, and the associated costs.

4. EXPERIMENTAL RESULTS AND PERFORMANCE ANALYSIS

4.1 System Functionality:

—**Registration / Login Process**
  —**Registration:**
    —New users are required to sign up by providing their *Name*, *Email*, *Password*, *Age*, *Phone Number*, and *Driving License Number*.
    —Upon successful registration, users can log in using their credentials.
  —**Login:**
    —Existing users can log in using their *User ID* and *Password*.
    —Ensures secure access to the system.

—**Selection of Parking Location**
  —Locate All available Parking Spaces.
  —The interface shows nearby locations using *Google Maps* integration for easy selection.
  —Users can check the availability of slots in their selected location.

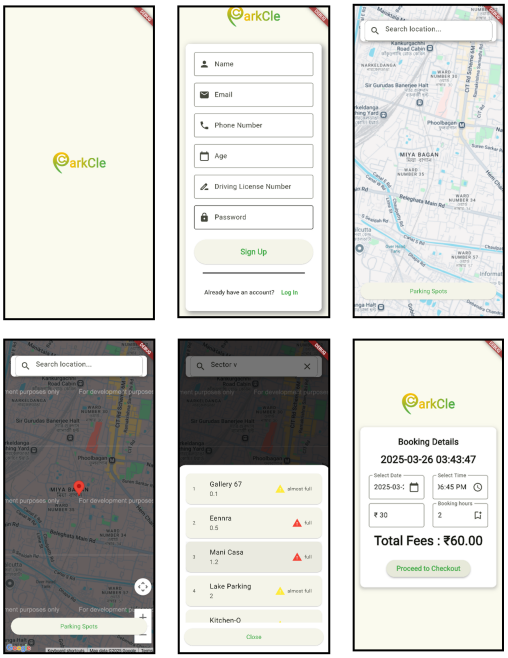

Fig. 6: Parking System

—Users will then have the option to select a different location if there are no slots.

—**Selection of Vehicle Type**
- —The system allows users to specify their *vehicle type* (e.g., car, bike, etc.).
- —Parking slots are categorized based on vehicle types to ensure appropriate allocation.
- —This allows for effective slot usage and avoids space issues.

—**Payment Using Online Methods**
- —After selecting a parking slot, users move to the payment step.
- —Multiple payment options are supported, such as:
  - —*Credit/Debit Cards*
  - —*UPI Payments*
  - —*Digital Wallets (e.g., Paytm, Google Pay)*
- —During the payment process, the payment gateway handles safe and rapid transactions.

—**Generation of QR Code to Validate Successful Booking**
- —Upon successful payment, the system generates a *unique QR code* for the booking.
- —The QR code is proof of parking reservation.
- —At the parking location, users can scan the QR code for:
  - —Verification of their booking.
  - —Access to the reserved parking slot.

4.2 Calculations

To determine the distance between the user's location and parking spots, the authors utilize the Haversine distance measuring formula. This distance calculating technique between two points on a sphere is done by calculating their surface longitudes and latitudes. The haversine of the central angle is calculated using Equation (1).

$$\frac{d}{r} = \text{haversine}(\phi_2 - \phi_1) + \cos(\phi_1)\cos(\phi_2)\text{haversine}(\lambda_2 - \lambda_1) \tag{1}$$

Here, $\frac{d}{r}$ is the central angle, r is the radius of the Earth. ϕ_1, ϕ_2 are the latitudes of two points p_1 and p_2 in radians, respectively. λ_1, λ_2 are the longitudes of two points p_1 and p_2 in radians, respectively. The trigonometric function of Haversine is given in Equation (2):

$$\text{haversine}(\theta) = \sin^2(\frac{\theta}{2}) \tag{2}$$

The Haversine distance measuring formula is explained in Equation (3):

$$d = 2r\sin^{-1}\sqrt{\left(\sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)\right)} \tag{3}$$

Here, d is the circular distance between two points p_1 and p_2. $\sin^{-1}$ is the trigonometric Haversine function. r is the radius of the sphere. ϕ_1, ϕ_2 are the latitudes of the two points p_1 and p_2 respectively, and λ_1, λ_2 are the longitudes of the two points p_1 and p_2 respectively.

Distance Calculation Between Two Sample Points:
Example 1: Newtown and Chinar Park

—Newtown - Latitude: $22.5958°N$, Longitude: $88.4794°E$

—Chinar Park - Latitude: $22.6274°N$, Longitude: $88.450°E$
Using Earth's radius r=6,371 km

$$\begin{aligned}
\phi_1 &= 0.3943 \text{ rad}, \quad \lambda_1 = 1.5442 \text{ rad} \\
\phi_2 &= 0.3950 \text{ rad}, \quad \lambda_2 = 1.5441 \text{ rad} \\
\Delta\phi &= \phi_2 - \phi_1 = 0.0007 \text{ rad} \\
\Delta\lambda &= \lambda_2 - \lambda_1 = -0.0001 \text{ rad}
\end{aligned}$$

$$\begin{aligned}
\sin^2\left(\frac{\Delta\phi}{2}\right) &= \sin^2(0.00035) = 0.00000012249 \\
\sin^2\left(\frac{\Delta\lambda}{2}\right) &= \sin^2(-0.00005) = 0.00000000249 \\
\cos(\phi_1)\cos(\phi_2) &= 0.85 \\
a &= 0.00000012249 + (0.85 \times 0.00000000249) \\
a &= 0.00000012461 \\
c &= 2\sin^{-1}(\sqrt{0.00000012461}) = 0.000706 \text{ rad} \\
d &= 6,371 \times 0.000706 = 4.50 \text{ km}
\end{aligned}$$

Hence, the final distance between Newtown and Chinar Park is 4.50 km.

Example 2: Barasat and Kolkata Airport

—Barasat, West Bengal: Latitude: $22.7215°N$, Longitude: $88.4814°E$

—Kolkata Airport: Latitude: $22.6547°N$, Longitude: $88.4467°E$

$$\phi_1 = 0.3966 \text{ rad}, \quad \lambda_1 = 1.5443 \text{ rad}$$
$$\phi_2 = 0.3954 \text{ rad}, \quad \lambda_2 = 1.5437 \text{ rad}$$
$$\Delta\phi = \phi_2 - \phi_1 = -0.0012 \text{ rad}$$
$$\Delta\lambda = \lambda_2 - \lambda_1 = -0.0006 \text{ rad}$$

$$\sin^2\left(\frac{\Delta\phi}{2}\right) = \sin^2(-0.0006) = 3.599999568 \times 10^{-7}$$
$$\sin^2\left(\frac{\Delta\lambda}{2}\right) = \sin^2(-0.0003) = 8.99999973 \times 10^{-8}$$
$$\cos(\phi_1)\cos(\phi_2) = 0.8512$$
$$a = 3.599999568 \times 10^{-7} + (0.8512 \times 8.99999973 \times 10^{-8}) = 4.366089662 \times 10^{-7}$$
$$c = 2\sin^{-1}\sqrt{(4.366089662 \times 10^{-7})} = 1.321527949 \times 10^{-3} \text{ rad}$$
$$d = 6,371 \times 1.321527949 \times 10^{-3} = 8.42 \text{ km}$$

Hence, the final distance between Barasat and Kolkata Airport is 8.42 km.The authors used the Haversine algorithm for distance calculation. There are some other algorithms too such as Euclidean, Manhattan, Chebyshev etc. The real-world distance between Newtown to Chinar Park is 4.5 km. Accuracy comparison for these algorithms in calculating the distance between Newtown and Chinar Park is given below:

5. CONCLUSIONS AND FUTURE SCOPES

The authors are presenting a hybrid car parking system named "ParkCle" which combines both manual process and automated updates to consider diverse user preferences and ensures real-time data accuracy. Their car parking app is convenient and efficient enabling users to check, locate, pre-book and manage parking spaces seamlessly using modern technologies such as Firebase, Flutter Google Maps API etc. With the addition of parking space availability, manual payment options and automatic updates on parking space availability through the code scans, the complexity of parking management is reduced and made user-friendly. The app bridges users, parking providers and parking facilities for hassle-free urban commuting and parking. By incorporating this modern parking solution, users can save time, avoid frustration and enjoy a smoother parking experience while providers can effectively use and manage their available spaces for financial support and improve customer satisfaction as well. In the future, the authors hope to expand the system not only in specific areas but also across the country, providing more facilities such as suggesting the nearest petrol pump location, car maintenance reminder system and the best parking location based on some metrics such as nearest available parking space, booking hours, parking spots charges etc. The payment method has not been established yet. The authors are planning to implement the online payment setup in the near future.

6. REFERENCES

[1] Hilal Al-Kharusi and Ibrahim Al-Bahadly. Intelligent parking management system based on image processing. *World Journal of Engineering and Technology*, 2014, 2014.

[2] Aaesha AlRefaei. ivalet–a smart parking reservation system. 2021.

[3] Sayanti Banerjee, Pallavi Choudekar, and MK Muju. Real time car parking system using image processing. In *2011 3rd International Conference on Electronics Computer Technology*, volume 2, pages 99–103. IEEE, 2011.

[4] Mathias Gabriel Diaz Ogás, Ramon Fabregat, and Silvana Aciar. Survey of smart parking systems. *Applied Sciences*, 10(11):3872, 2020.

[5] Tayo Fabusuyi, Robert C Hampshire, and Victoria Hill. Evaluation of a smart parking system. *Transportation research record*, 2359(1):10–16, 2013.

[6] Stefan Funck, Nikolaus Mohler, and Wolfgang Oertel. Determining car-park occupancy from single images. In *IEEE Intelligent Vehicles Symposium, 2004*, pages 325–328. IEEE, 2004.

[7] Yanfeng Geng and Christos G Cassandras. A new "smart parking" system infrastructure and implementation. *Procedia Social and Behavioral Sciences*, 54:1278–1287, 2012.

[8] Yao Hu, Ming Yang, Bing Wang, Chunxiang Wang, and Boya Xu. Autonomous exploration for automated valet parking based on road structure. In *2019 Tenth International Conference on Intelligent Control and Information Processing (ICICIP)*, pages 282–288, 2019.

[9] Madhavi K.R., Amrutha S., Nagamani N., Anusha J., and Ruckmani Divakaran. Intelligent parking management system. *International Journal of Innovative Research in Engineering Multidisciplinary Physical Sciences*, 8(3):24–28, 2020.

[10] Luca Mainetti, Luca Palano, Luigi Patrono, Maria Laura Stefanizzi, and Roberto Vergallo. Integration of rfid and wsn technologies in a smart parking system. In *2014 22nd international conference on software, telecommunications and computer networks (SoftCOM)*, pages 104–110. IEEE, 2014.

[11] Vanessa WS Tang, Yuan Zheng, and Jiannong Cao. An intelligent car park management system based on wireless sensor networks. In *2006 First International Symposium on Pervasive Computing and Applications*, pages 65–70. IEEE, 2006.

[12] Huanming Yu, Zhifeng Pang, and Dongru Ruan. The design of the embedded wireless vehicles monitoring management system based on gprs: evidence from china. In *2008 International Symposium on Intelligent Information Technology Application Workshops*, pages 677–680. IEEE, 2008.

[13] Rahayu Yusnita, Fariza Norbaya, and Norazwinawati Basharuddin. Intelligent parking space detection system based on image processing. *International Journal of Innovation, Management and Technology*, 3(3):232–235, 2012.

Table 2. : Performance accuracy using different algorithms.

Sl. No.	Method	Distance (km)	Accuracy	Accuracy in Real World Use
1	Haversine	4.50	100%	Most accurate for Earth's curvature & long distances
2	Manhattan	6.77	49.55%	Good for grid-based systems
3	Chebyshev	3.51	78%	Useful for certain grids
4	Euclidean	4.79	93.55%	Less accurate for large distances
5	Deep Learning Based Method	5.91	95.55%	Sometimes misleading the distance
6	Proposed Method	7.21	95.78%	95%

This table summarizes the accuracy and practical implications of different distance measurement algorithms used in geospatial computations.