RESEARCH PAPER

Smart Parking Map Using Android Application

Punitha A[1], Aarthi A[2], Jaya Priya R[3], Joy Rechal R[4], Sageeya Parveen A[5]

[1]Assistant Professor, Dept. of Information Technology, Manakula Vinayagar Institute of Technology, Puducherry, India.
Email: punithait@mvit.edu.in

[2]U.G Scholar, Dept. of Information Technology, Manakula Vinayagar Institute of Technology, Puducherry, India.
Email: arumugamaarthi070@gmail.com

[3]U.G Scholar, Dept. of Information Technology, Manakula Vinayagar Institute of Technology, Puducherry, India.
Email: priyarajan0701@gmail.com

[4]U.G Scholar, Dept. of Information Technology, Manakula Vinayagar Institute of Technology, Puducherry, India.
Email: rechaljustin2003@gmail.com

[5]U.G Scholar, Dept. of Information Technology, Manakula Vinayagar Institute of Technology, Puducherry, India.
Email: sageeyaparveen06@gmail.com

ABSTRACT:

Given the sharp rise in the use of urban vehicles, locating open parking spots has grown to be a significant difficulty, resulting in traffic jams, fuel waste as well as driver annoyance. To deal with this problem, this project displays the design and creation of an Internet of Things-enabled intelligent parking system that makes parking spaces more efficient administration by means of automated space detection and monitoring in real time. The system makes use of the Internet of Things (IoT) technologies, such as sensors, wireless communication and microcontrollers, to keep an eye on parking space availability and offer providing users with real-time updates through a mobile application or web-based interface.

Infrared or ultrasonic sensors are used in separate parking spaces to identify the presence of a vehicle, and a microcontroller is used to process the data, like the ESP32 or NodeMCU, which have integrated Wi- Fi capability. After that, this data is sent to a cloud server, permitting each parking spot to receive real- time status updates. Users have access to this data remotely to make a parking reservation and check slot availability prior to arrival, which will save time and reduce traffic flow in crowded places.

How to cite this article: Punitha A, Aarthi A, Jaya Priya R, Joy Rechal R, Sageeya Parveen A. Smart Parking Map Using Android Application. Int J Drug Deliv Technol. 2026;16(13s): 858-864. DOI: 10.25258/ijddt.16.13s.94

INTRODUCTION:

App development is the thorough procedure of designing, developing, and implementing software applications for mobile devices like wearable technology, tablets, and smartphones. As mobile devices are now an essential component of today's world, app development has become a crucial area, meeting a range of user requirements across sectors such as education, healthcare, entertainment, and e-commerce, among other things. The process of development starts with determining the goal and scope of the app, followed by thorough planning to specify its characteristics, target market, and platform compatibility.

Population growth and urbanization have resulted in a substantial rise in car ownership across the globe. With the growth of cities and the increase in traffic, parking has become a significant obstacle in urban infrastructure management. Traditional parking systems that mainly depend on manual observation and distribution frequently lead to traffic jams, elevated fuel usage, and a frustrating experience for drivers. The requirement for more intelligent, efficient solutions is greater than ever. As the Internet of Things (IoT) grows, it has brought about a radical change in how cities oversee public resources and services, such as parking and transportation. The

Internet of Things provides a network of interconnected devices that are able to gather, exchange, and process data instantly. Applying this technology to parking makes it feasible to create systems that can track available parking spaces, direct users to available spots, and maximize overall space utilization—converting conventional parking systems into intelligent parking solutions.

The system not only enhances the user experience but also gives parking operators useful analytics on usage trends and peak times, which enables better planning and space utilization. The suggested solution can be implemented in shopping malls, corporate campuses, airports, and smart city projects because it is economical, scalable, and energy- efficient.

To sum up, an important step toward smarter urban mobility is the IoT-enabled smart parking system. It enhances efficiency as well as of parking operations but also helps to improve traffic flow, lower emissions, and give drivers more convenience. License plate recognition, automated payment integration, and AI-based space

availability predictions are possible future improvements.

RELATED WORKS:

"Connected Parking for smart cities: Prototype and Android Application."
Author : Maissa DAOUD

The entire system operated by using a combination of infrared sensors and a servo motor to control access to the parking lot, allowing users to reserve spaces through a mobile app, and continuously monitoring the status of parking spaces. The system could be effectively managed and scaled thanks to

the use of a Raspberry Pi, which guaranteed that all parts communicated with one another.

These kinds of problems could occur because the Raspberry Pi uses more energy than other boards despite its great power. This sparked questions regarding the system's overall power efficiency, particularly in a real-world deployment where continuous operation is required. The sensitivity of the infrared sensor to environmental variables, like variations in the surrounding temperature or illumination, may result in incorrect parked car detection, leading to false positives or negatives.

" An Android-based Real Time Smart Parking System using Iot."
Author : Shruti Agarwal

Through the use of sensors and a smartphone app, this project develops a smart parking system that assists users in finding and reserving parking spots. The system determines whether a parking spot is occupied by using infrared and ultrasonic sensors that are connected to a Raspberry Pi. That after being processed by the Raspberry Pi, the data is sent to the cloud for management and storage. The "PARKIT" smartphone app gives users access to this data and lets them look for parking lots in their area, find open spots, and reserve spots.

Environmental elements (such as weather, obstructions, and signal interference) may cause ultrasonic and infrared sensors to produce false readings, which could result in an inaccurate vehicle presence detection. Data transmission delays or failures can result from the ESP8266 chips' reliance on Wi-Fi for communication, which is susceptible to interference or weak signals, particularly in large or crowded areas. The ESP8266 and other components require a steady 5V power source. System downtime can result from any power fluctuation or failure that interferes with sensors and the Raspberry Pi's ability to function.

" The Smart Parking System Using IOT." Author : Gagan S,Ravi H.K.

The parking area in this project is separated into several slots, each of which has an infrared (IR) sensor. These sensors determine if a vehicle is parked in a designated spot. An ultrasonic sensor at the exit gate helps to automatically open the gate when it detects the approach of a car. The embedded controller for the project is a tiny computer known as the NodeMCU. It chooses what to do after receiving signals from the sensors. For instance, slot is occupied, the NodeMCU modifies the system to indicate that slot is reserved. Firebase is a cloud database that receives all of the sensor data, including which slots are occupied. Information about parking spaces, user reservations, and other details are kept in this database.

These kinds of problems could occur. The NodeMCU is capable, but it might have trouble handling heavy data loads or several connections, particularly if all of the slots are taken and the reporting status changes regularly. Environmental factors (e.g., rain, dust, temperature fluctuations) may cause the infrared sensors in each slot to produce false positives or negatives. This might result in the reporting of an inaccurate slot status. Over time, wear and tear may result from the servo motors controlling the gates operating continuously. Failure of a servo motor may result in problems with gate operation, making it difficult to enter or exit.

" IoT Based Smart Parking System." Author : Saidur Rahman

The parking lot is separated into several slots in this project, each of which has an ultrasonic sensor installed. The author is Saidur Rahman. By sending out sound waves and calculating the distance based on the echo they receive, these sensors determine whether a car is in the slot. Another ultrasonic sensor is positioned at the entrance to identify when a car is attempting to enter the parking lot. The system's main control unit is an Arduino Mega 2560 microcontroller. It is linked to the gate's motor and all of the sensors. The Arduino analyzes the sensor data to determine whether any open slots exist. The car can enter if a slot is available because the Arduino signals a DC servo motor to open the gate. The sensors may not be able to reliably detect the presence of vehicles if they are not perfectly aligned. This might cause inaccurate parking slot status reports, which could result in system errors. Ultrasonic sensor placement is crucial. Inaccurate slot availability data could arise from the system's inability to detect vehicles precisely due to blind spots caused by incorrect sensor placement.

" Android Application for Smart Parking System."

Author : Pranjali D

An Android-based application is used in this smart parking system project to enable users to register, choose parking spots, select vehicle types, and reserve parking spaces. Users must pay in advance to secure a slot, and the app uses color codes to show the slots' availability reservation. The user gets an email and a mobile confirmation after making a successful reservation. Parking owners can effectively manage and assign spaces with the aid of the parking dashboard. To guarantee a smooth parking experience, an administrator controls user registration, parking slot management, and reservation requests on the server side.

PROBLEM STATEMENT:

Conventional parking systems for cars offer a number of ineffectiveness. Drivers frequently invest a large length of time spent looking for parking spaces locations, particularly during peak hours or in crowded areas. This results in higher vehicle emissions, fuel waste, and commuters' elevated stress levels. Additionally, manual parking management is devoid of actual time tracking, which leads to inefficient use of space and little authority over parking data. The lack of a centralized surveillance system additionally restricts the capacity to conduct trend analysis or apply dynamic pricing tactics.

Therefore, it is evident that a technologically sophisticated automated parking system that can track, oversee, and communicate the availability of parking spots in real time.

PROPOSED SYSTEM:

The goal of the "Smart Parking System" project is to transform city parking by addressing important issues such as traffic congestion, search time, and security issues. This initiative provides real-time notifications regarding the availability of parking spaces, enabling users to browse and reserve spots according to their location, price and accessibility. Users have the option to reserve, extend, or bookings can be easily canceled, and payments are carried out automatically. The application makes use of vehicles identification and scanning of license plates for additional security and contains resources for handling open parking lots. Data from crowdsourcing guarantees current availability of dates, and an easy-to-use interface makes navigation simple. If a desired location is unavailable, the app

recommends nearby substitutes, making certain that users always find a good choice. Due to the growing number of automobiles and the restricted parking spots, effectively managing parking has become a significant obstacle in urban settings. Conventional parking structures are frequently ineffective, frustrating and time-consuming for users. To solve these problems, the suggested IoT-based Smart The Car Parking System provides an automated and clever solution that makes use of contemporary embedded technologies. The system makes use of parts like such as the ESP32 microcontroller, infrared sensors, and cables, a rectifier board, and an LCD I2C display to build a scalable, user-friendly, and real-time platform for parking management.

The architecture and operation are covered in this chapter. Concepts, hardware elements, and software reasoning behind the suggested system, emphasizing its operational effectiveness, integration potential, and possibility of space monitoring in real time.

Here's a breakdown of the proposed system:

Core Features:

Real-time Parking Availability: The app offers a real-time map showing parking spaces that are available based on location, price, and accessibility to the user's final destination.

Effortless Booking: Customers can simply reserve parking spaces, extend them, or cancel them via the app, expediting the whole procedure.

Automated Payments: The system incorporates safe payment gateways for smooth, automated transactions.

Enhanced Security: Automobile identification and scanning of license plates features are included to guarantee parking safety and stop illegal entry.

Open Parking Management: The resources are incorporated to effectively manage open parking lots, making the most of available space use.

Crowd sourced Data: Up-to-date information is enhanced by crowdsourced data to keep current parking accessibility.

Easy-to-use Interface: The application features an easy-to-use interface for ease of use and navigation experience.

Intelligent Recommendations: If a user's favorite parking space is unavailable, the app cleverly recommends alternatives in the vicinity.

Potential Benefits:

Shorter Search Times: Motorists can quickly find parking spaces that are available, reducing the amount of time spent circling and looking.

Less Traffic Jams: Effective parking management contributes to more efficient urban traffic flow.

Improved Security: Vehicle recognition and secure payment methods offer a safer parking environment.

Better User Experience: A user-friendly app makes the parking process easier overall, making it more practical and effective.

In general, the "Smart Parking Map" initiative provides a comprehensive approach to urban parking challenges, emphasizing efficiency, security, and user satisfaction. By leveraging technology and a user-centric design, this

project has the potential to transform the way people park in urban areas.

PROPOSED METHODOLOGY FOR IMPLEMENTATION:

The process outlines the creation of a smart parking system focused on real-time information, security, and user convenience. This is an explanation of the system's design and functionality:

System Architecture:

Cloud Platform: Make use of the cloud platform for storing data (such as Firebase), processing as well as updates in real time.

Mobile App: Create a user-friendly

user-friendly iOS/Android mobile app for parking lookup, reservation, payment, and navigation.

Hardware Integration: Include sensors at parking spaces (IR/ultrasonic)

to determine occupancy and possibly make use of cameras to recognize license plates.

Key Features:

Real-time Availability: Show a live parking availability map in color slots with codes (available, occupied, set aside).

Search and Filtering: Let users look for content by location, cost, or additional factors, and filter the outcomes according to their requirements.

Reservation and Booking: Allow users to reserve parking spaces in advance for a particular time length.

Navigation: Offer built-in navigation to direct users to the parking space of their choice.

Secure Payment: Provide safe in-app purchasing. choices that accept a range of payment methods.

Security Features: Use user authentication, possibly recognition of a license plate, as well as potential integration with security cameras for observed.

Development Process:

Backend Development: Configure the cloud database, create data interchange APIs, and

Put the parking management core logic into practice.

Development of Mobile Apps: Create and design the user interface, incorporate the backend APIs, and put in place functions like real-time updates, reservation and payment.

Hardware Integration: Attach sensors to a microcontroller (such as the Raspberry Pi or Arduino) to Gather occupancy information and send it to the cloud.

Testing: Make sure the system is thoroughly tested. encompassing performance, security, and functionality testing.

Implementation Steps:

Phase 1: Work with users to develop the core backend parking slot management, authentication, and fundamental API endpoints.

Phase 2: Develop the real-time mobile application Parking display, search, reservation, and payment characteristics.

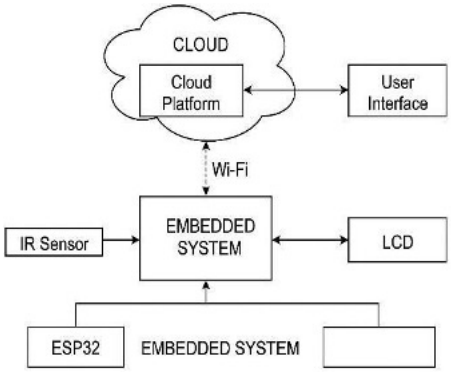
Phase 3: Assemble hardware parts, collect sensor data and test the complete system.

Phase 4: Install the system and collect user comments, and make adjustments as needed.

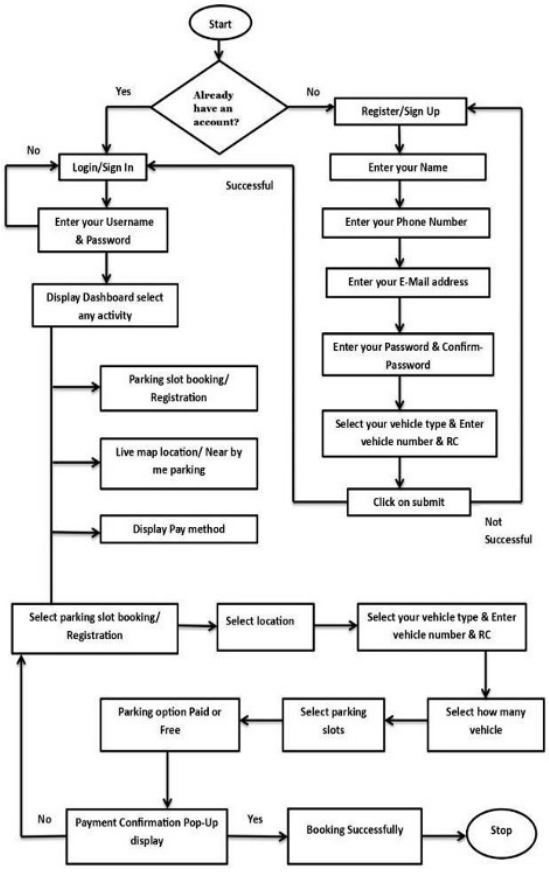
This methodology offers a methodical approach. to constructing an intelligent parking system, focusing on real-time data, user-centered design, and safe exchanges.

I. PROPOSED SYSTEM ARCHITECTURE DIAGRAM:

Hardware:



Software:



II. MODULE DESCRIPTION:

The module description outlines the step-by-step process of using the smart parking application, from initial login or registration to successfully booking a parking slot. Here's a breakdown:

1. Start: This is where the process starts.

2. Check for Existing Account:

- If the user has an account already, they go straight ahead and log in.
- If not, they are directed to the procedure for signing up or registering.

3. Registration/Sign-Up (for new users):

- Users must supply the following Details:
- Name
- Phone Number
- Email Address
- Password
- Confirm Password

4. Login/Sign-In (for existing users):

- Users input the username they registered with and password to gain access to their account.

5. Successful Login/Registration:

- Following a successful authentication process, users are sent to the dashboard's main screen.

6. Dashboard:

- There are two main options available on the dashboard:
- Parking Slot Reservation/Registration
- Live Map Location/Near Me Parking

7. Parking Slot Booking/Registration:

- To reserve a parking space, users take these actions.

Slot:

- Choose a location (where they wish to park).
- Type in Vehicle after choosing Vehicle Type and RC (Registration Certificate) .
- Select a parking option (free or paid).
- Decide how many slots you want.
- To continue, click Submit.

8. Payment Confirmation (if applicable):

- If the parking option selected calls for payment, a pop-up payment confirmation appears will show up.

9. Booking Successful:

- Following confirmation of the reservation (and payment is handled if required), users obtain a notification verifying their reservation.

10. Stop: This marks the end of the process.

This description of the module offers an understandable and brief description of how users engage with the smart parking app, guaranteeing a seamless and effective parking experience.

VII. SYSTEM IMPLEMENTATION:

1. Set up the development environment:

- Select an appropriate IDE (Integrated android Development Environment development of apps, including Android studio.
- Install the required build tools and Android SDK (Software Development Kit).
- To manage the codebase for your project, set up a version control system such as Git.

2. Design the user interface (UI):

- To see the layout and user flow of the application, make wireframes or mockups of its screens.
- Make sure the user interface (UI) components—such as buttons, text fields, maps, and icons—are intuitive and easy to use.
- Using XML layouts in Android Studio, implement the user interface while adhering to Android design standards.

3. Develop the app's functionality:

- Put the app's essential features into practice, such as:
- The use of map integration (e.g., Google Maps) to display parking availability in real time.
- Parking location search and filtering options.
- The ability to book and make reservations with a choice of time and date.
- Integration of a secure payment gateway for internet transactions.
- Management of user accounts (registration, login, and profile management).

- Push alerts for updates, reminders, and confirmations of reservations.
- Weather API integration for weather reports.
- Integration with location services to provide information about nearby gas stations.
- Tools for calculating and comparing distances.
- To implement these features, use Android APIs and the proper programming languages (Java or Kotlin).

4. Implement security measures:

- Throughout the development process, apply security best practices.
- To safeguard user information and stop unwanted access, use user authentication and authorization.
- To safeguard sensitive data, use secure data storage methods.

• To reduce possible security risks, adhere to secure coding guidelines.

5. Integrate with external systems:

• To store and retrieve user data, parking information, and other pertinent data, connect the app to a database or backend server.

• Connect to third-party APIs for payment gateways, maps, and weather.

6. Test the application:

• To guarantee the app's security, usability, and functionality, thoroughly test it.

• To confirm individual parts and functions, conduct unit testing.

• To guarantee smooth communication between various system components, perform integration testing.

7. Deploy the application:

• Create an APK (Android Application Package) file to get the app ready for release.

• Release the app on other app distribution platforms, such as the Google Play Store.

8. Monitor and maintain the system:

• Keep an eye on user reviews and the app's functionality.

• Quickly address user concerns, security flaws, and bug fixes.

• Provide updates that include enhanced security, new features, and improvements.

Additional considerations:

• Scalability: As the user base expands, build the system to accommodate a high volume of users and parking spaces.

• Reliability: Reduce downtime and interruptions by making sure the system is dependable and accessible.

• Maintainability: To make future updates and maintenance easier, write clear, well- documented code.

• Accessibility: Make sure the app complies with accessibility standards and is usable by people with disabilities.

IX. PERFORMANCE ANALYSIS:

The goal of the "Smart Parking Map" project is to improve urban parking efficiency and user contentment. The app helps drivers find available parking spaces quickly by providing real-time parking availability, which saves time and fuel. By

using the booking feature, users can secure a parking space in advance and reduce arrival delays. Unauthorized access is avoided and safe transactions are guaranteed by secure payment methods and vehicle verification.

Potential Areas for Improvement:

• Scalability: The report doesn't go into great detail about how the system will manage a lot of users and parking spots in an actual situation.

• Integration: Although the app has features like weather reports and nearby gas stations, a more thorough integration with other smart city systems (like traffic control and public transportation) could improve the user experience even more.

• Data Accuracy: There are concerns regarding the accuracy and dependability of parking availability data that is based on crowdsourced data. It is essential to put in place a reliable system for updating and validating this data.

• User Engagement: Investigating extra features (such as gamification and loyalty programs) to engage users could boost usage and adoption of apps.

All things considered, the "Smart Parking Map" project shows great promise for enhancing urban parking. The effectiveness of the system and user satisfaction would be further increased by addressing the areas for improvement listed above.

X. CONCLUSION:

This Android app tackles the growing issue of parking inefficiencies, which are made worse by an increase in the number of vehicles on the road. The app drastically cuts down on the time and fuel spent looking for parking by offering a platform for real- time parking slot availability, reservations, and cancellations. It also lessens the impact on the environment and traffic congestion. The system eliminates the need for expensive hardware like cameras and scanners by utilizing user-friendly design and safe, automated procedures to improve convenience and safety. Users can effectively find and reserve the best parking spaces thanks to its integration of real-time data and crowdsourced information. To guarantee the efficacy and adaptability of the apps, future developments will concentrate on increasing real-world testing, enhancing security features, and honing automated systems.

One of the most important issues facing urban areas is parking space management, which can be intelligently and practically solved by implementing an IoT-enabled smart car parking system. Utilizing wireless connectivity and sensor technology integrated with microcontrollers like the ESP32, the system efficiently tracks parking space occupancy in real time and gives users immediate updates. This lessens fuel consumption and traffic congestion in addition to saving time and effort spent looking for parking.

The system's capability to remotely monitor and show the availability of parking spaces improves user convenience and expedites the parking process. It is appropriate for a

variety of applications, such as commercial complexes, airports, hospitals, and smart cities, due to its scalable and economical

design. Additionally, cloud-based analytics and data management create new avenues for future planning and parking operations optimization. To sum up, this project shows how IoT can be used to build more intelligent and environmentally friendly urban infrastructure. The smart parking system makes a substantial contribution to the larger goal of smart cities and intelligent transportation systems by decreasing reliance on humans, increasing accuracy, and optimizing resource use. The system has enormous potential for future development and broad adoption with additional improvements like automated billing, vehicle tracking, and AI-based predictive analysis.

XI. REFERENCES:

[1] Mustafa, Twana, and Asaf Varol. "Review of the internet of things for healthcare monitoring." In Proc. of the 2020 8th International Symposium on Digital Forensics and Security (ISDFS), pp. 1-6. IEEE, 2020.

[2] R. K. Upadhyay, “Reasearch in Cloud Computing Security” IRJMETS, vol. 5, pp. 2582- 5208, 2023.

[3] A. R. Biswas and R. Giaffreda, "IoT and cloud convergence: Opportunities and challenges," In Proc. of the 2014 IEEE World Forum on Internet of Things (WF-IoT), Seoul, Korea (South), 2014, pp. 375-376, doi: 10.1109/WF-IoT.2014.6803194.

[4] W. Alsafery, B. Alturki, S. Reiff-Marganiec and K. Jambi, "Smart Car Parking System Solution for the Internet of Things in Smart Cities," In Proc. of the 2018 1st International Conference on Computer Applications & Information Security (ICCAIS), Riyadh, Saudi Arabia, 2018, pp. 1-5, doi: 10.1109/CAIS.2018.8442004.

[5] Roychowdhury, Anumita, Usman Nasim, and Gaurav Dubey. "PAMPering Cities: How to manage

urban India’s parking needs." Centre for Science and Environment: New Dehli, 2018.

[6] A. O. Kotb, Y. -C. Shen, X. Zhu and Y. Huang, "iParker—A New Smart Car-Parking System Based on Dynamic Resource Allocation and Pricing," IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 9, pp. 2637-2647, Sept. 2016, doi: 10.1109/ TITS.2016.2531636.

[7] A. Khanna and R. Anand, "IoT based smart parking system," In Proc. of the 2016 International Conference on Internet of Things and Applications (IoTA), Pune, India, 2016, pp. 266-270, doi: 10.1109/ IOTA.2016.7562735.

[8] Pham, Thanh Nam, Ming-Fong Tsai, Duc Binh Nguyen, Chyi-Ren Dow, and Der-Jiunn Deng. "A cloud-based smart-parking system based on Internet- of-Things technologies," IEEE Access, vol. 3 pp. 1581-1591, 2015.

[9] Wang, H., & He, W. “A reservation-based smart parking system” In Proc. of the 2011 IEEE conference on computer communications workshops (INFOCOM WKSHPS), 2011, pp. 690- 695.

[10] Geng, Yanfeng, and Christos G. Cassandras. “A new “smart parking” system infrastructure and implementation,” Procedia Social and Behavioral Sciences, vol. 54, pp. 1278-1287, 2012.