Smart Parking System

Mounika Manchala
Computer Science and Engineering
Santa Clara University
Santa Clara, USA
mmanchala@scu.edu

Puneet Bhushan
Computer Science and Engineering
Santa Clara University
Santa Clara, USA
pbhushan@scu.edu

Ruchiben Patel
Software Engineering
Santa Clara University
Santa Clara, USA
rapatel@scu.edu

Sai Grandhi
Computer Science and Engineering
Santa Clara University
Santa Clara, USA
sgrandhi@scu.edu

***Abstract*—As the number of vehicles on the road is increasing on a daily basis, parking, especially in urban areas is becoming a hassle for both the public and local authorities.With the emergence of the Internet of Things problems such as traffic congestion and parking issues can be avoided. In this paper, we present an IoT embedded system which indicates the availability of each parking space using ultrasonic sensors. This cloud integrated smart parking system enables the users to check the available parking slots and also have the facility to reserve the slot using an Android application.**

Keywords—Internet of things; Smart Parking; Raspberry pi; AWS Cloud Service, MQTT

I. Introduction

Internet of things(IoT)is the concept of connecting any device to the internet or to any other device. With the use of remote computers connected through the internet, these devices can be maintained and monitored. A growing trend in the use of IoT applications can be seen in the application of smart cities to reduce the issues of smart cities. One of the major issues in smart cities is parking. Increase in the number of vehicles and improper utilization of available parking spaces adds to traffic congestion. On an average, a driver spends 4%-5% time in cruising(searching for parking) in the cities like San Francisco. Problems related to parking can be reduced if the driver is informed in advance about the availability of the parking space. Proposed Smart Parking System tries to solve the cruising issue. Existing solutions for parking problems such as ParkMe mobile app works with the help of city parking departments, they added real-time data from smart meters and parking sensors found in LA, San Francisco, and other urban areas, that show, on a minute-by-minute basis, which spots are available[2].

Analysis

Finding free parking spaces is a big issue in most of the urban cities. Drivers spend an average of 17 hours a year searching for spots on streets, in lots, or in garages, according to recent studies. The hunt adds up to an estimated $345 per driver in wasted time, fuel, and emissions, according to the analysis by INRIX, a leading specialist in connected car services and transportation analytics.[3]

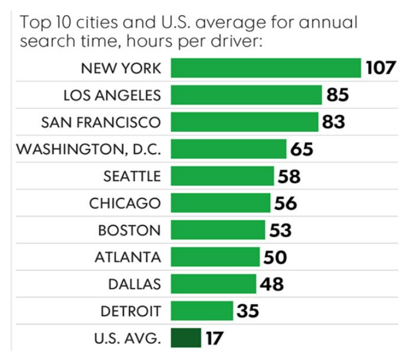

Fig. 1. Motorists spend an average of 17 hours and about $97 per year searching for places to park, according to a recent study. Cities with highest parking costs[3].

A motorist in large cities endure the toughest challenges. In New York City, the hardest-hit U.S. urban area, drivers on average spend 107 hours a year looking for parking spots, the report said. The searches add up to what the report estimates as $2,243 in wasted time, fuel, and emissions per driver, plus $4.3 billion in costs to the Big Apple. Los Angeles, San Francisco, Washington, and Seattle round out the top five cities with the toughest parking experiences, the report said[3].

The proposed smart parking system is real-time detection of available parking spaces. Components we used are raspberry pi, ultrasonic sensors, MQTT publish and subscribe, AWS IoT, AWS IoT Lambda, Dynamodb, Firebase, An Android mobile app. The schematics of paper is as follows: Section III presents the state-of-the-art in smart parking system. Section IV is about implementation and working of smart parking system and conclusions are present in section V.

TABLE 1: INRIX Parking Ranking – Hours Spent Searching for Parking[4]

Rank	U.S. City	Average 2-Hour Parking Cost (One mile of city center)	On-Street Search Time (mins/trip)	Off-Street Search Time (mins/trip)	Annual Search Time (hours/ driver/year)	Annual Search Cost Per Driver	Annual Search Cost Per City
1	New York	$33	15	13	107	$2,243	$4.3bn
2	Los Angeles	$14	12	11	85	$1,785	$3.7bn
3	San Francisco	$12	12	11	83	$1,735	$655m
4	Washington D.C.	$18	10	9	65	$1,367	$329m
5	Seattle	$10	9	8	58	$1,205	$490m
6	Chicago	$22	9	8	56	$1,174	$1.3bn
7	Boston	$26	8	8	53	$1,111	$262m
8	Atlanta	$6	8	8	50	$1,043	$251m
9	Dallas	$6	8	8	48	$995	$726m
10	Detroit	$9	6	6	35	$731	$209m
	US	*$4*	*2*	*2*	*17*	*$345*	*$72.7bn*

II. Need of cloud Service

A. Interoperability

Internet of Things involves the use of devices that are heterogeneous in nature. These devices may have different hardware or software configurations and it results in causing compatibility issues and it becomes very difficult in an IoT environment to ensure interoperability among these devices[5]. All these issues can be solved using Cloud service and it provides a common platform where various devices can connect and interact. Devices are allowed to share and exchange data in a format that is acceptable to them.[9]

B. Availability

The availability of resources becomes very easy when we integrate with a cloud database. With the use of cloud, the applications are always up and running and continuous services are being provided to the end users[9].

C. Scalability

Cloud service can provide us a scalable approach towards IoT by accommodating a wide range of devices. We can increase or decrease the number of resources to access a medium. The cloud allocates and deallocates the resources accordingly with the available resources[9].

III. System Architecture

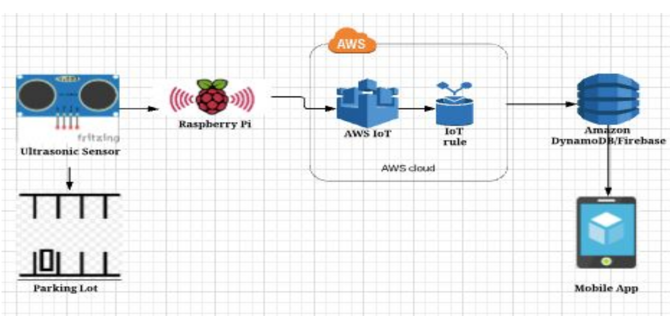

Fig. 2. Represents system architecture of proposed smart parking system

A. Parking Sensors

For our parking system, we have used Ultrasonic Sensors HC-SR04. The ultrasonic sensor uses SONAR to determine the distance to an object. The purpose of this sensor is to determine whether the individual parking slot is vacant or not. The sensors are connected to the 5V supply. It can measure a wide range from 2-400cm. It has 4 pin module.

- Vcc : +5V
- TRIG : Trigger (Input)
- ECHO : Output
- GND : Ground

Fig. 3. Ultrasonic sensor HC-SR04

B. Raspberry Pi

The Raspberry Pi is a credit-card-sized computer that plugs into your TV and a keyboard. It is a capable little computer which can be used in electronics projects, and for many of the things that your desktop PC does, like spreadsheets, word processing, browsing the internet, and playing games[6]. In our project, Raspberry Pi is a processing unit, which is a bridge between sensors and cloud. Raspberry Pi 3 is connected to the ultrasonic sensor. A C library PiGPIO is used to communicate with the ultrasonic sensor and collect data. Collected data is sent to the AWS IoT broker using a MQTT client on Raspberry Pi 3 written in NodeJS using AWS IoT Node SDK.

C. Amazon Web Service(AWS)

Amazon web service is a cloud service, which we have used to integrate the sensor information to store data in the database. We used AWS IoT as a virtual device to integrate with sensors data. We also used AWS Lambda to communicate to Firebase which is a real-time database.

AWS IoT core is a cloud platform and allows users to connect the devices easily and interact with cloud applications and other devices. I can support billions of devices and trillions of messages and can route the messages to AWS endpoints and to other devices reliably and securely. It can keep track of all messages all time even when they are not connected[8]. It provides the following benefits:

- Connect and manage your devices
- Secure device connections and data
- Process and act upon device data
- Read and set device state at any time

AWS lambda can run code virtually any type of application or backend service with zero administration[8]. It provides the following benefits:

- No Servers to manage
- Continuous Scaling
- Subsecond Metering

AWS Cognito allows adding user sign-up, sign-in, and access control to the mobile app easily. It can provide security to the app[8].

D. Database

We used Google Firebase for real-time communication to the Mobile Application. If there is any change in sensor data we can recognize the change in the app. We can reserve the parking lot in the app and we can see the changes in the database.

E. Mobile application

An android app is provided to end user to display parking slots data. The application is developed using Android SDK studio 3.1.0. The application is connected to the Firebase database. Data transfer is in JSON format from AWS IoT to mobile application.

IV. Implementation and Working

In this section, we will discuss the implementation and working of our smart parking system. The complete process of booking a parking slot, parking in a slot and leaving the car parking area is explained.

Hardware

- HC-SR04 ultrasonic sensor
- Raspberry Pi 3
- Android Phone

Software

- PiGPIO
- AWS IoT Core
- AWS Lambda
- AWS Cognito
- Google Firebase
- Communication: MQTT Protocol in AWS IoT

Implementation

The ultrasonic sensors are connected to the Raspberry Pi 3 which uses a C library PiGPIO interact with ultrasonic sensors and collect data. The data is sent to AWS IoT broker using a MQTT client on Raspberry Pi written in NodeJS through AWS IoT Node SDK. An AWS IoT Lambda pushes retrieved data to DynamoDB and Firebase. Firebase is connected to an Android application that the user can use to see if parking is available and also he can reserve a parking lot.

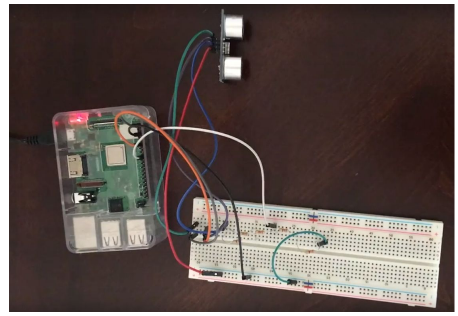

Fig. 4. Model Setup

Working of Smart Parking app:

Step 1: End-user installs the smart parking app on their mobile.

Step 2: User logs in with their credentials.

Following figures shows our app:

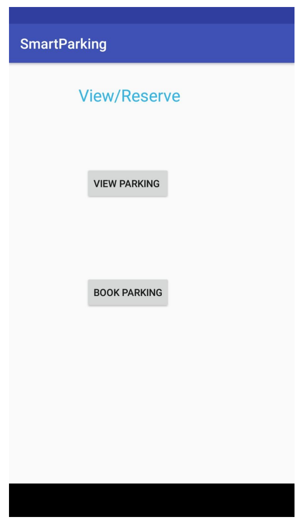

Fig. 5. Displays parking areas

Step 3: Checks for the available parking slots in particular parking area by clicking the “View Parking” tab.

Step 4: Enter the selected parking slot by clicking the “Book Parking” tab. Only available Parking lots are shown on the screen.

Step 5: Clicking on the “Reserve Lot” tab to complete the booking, If the reservation is successful, a toast message appears on the screen. Also, confirmation can be seen in the “View Parking” tab.

If the user is willing to use an empty parking spot without remotely booking it through the application, he/she can use the parking slot, which will be shown as in use (red color) in the application till the car is parked, so that no one else can book that slot remotely. The range set for the availability of a vehicle in the parking slot is determined by the ultrasonic sensor where the distance is 15 cms to 200 cms

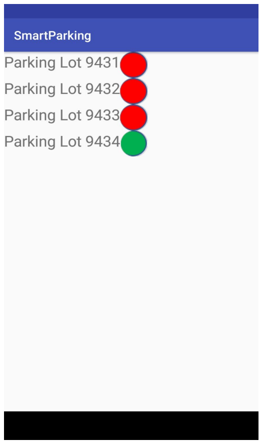

Fig. 6.View Parking App Interface

Fig.6. Displays parking spaces in particular area. Red color indicates the parking space is taken. Green color indicates parking lot is available for reservation. Every parking lot is given an ID (9431, 9432 etc.)

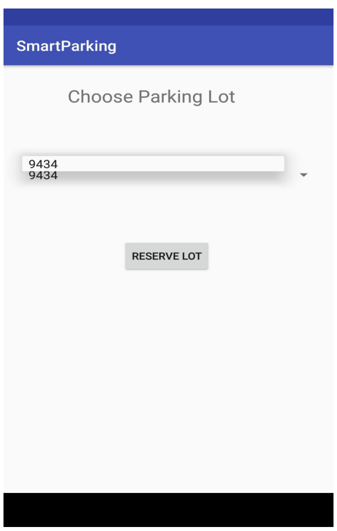

Fig. 7. Book Parking App Interface

.

V. Security Considerations

All the traffic from the ultrasonic sensor to the cloud is encrypted over Transport Layer Security(TLS). AWS Identity Authentication and Management is used for enabling the roles and policies to securely grant access to the specified services to communicate with them. Also, for the app security, AWS Login is provided by AWS Cognito is used which grants access to the resources only for the ones who are registered.

VI. Conclusions

The concept of smart cities, which seemed like a dream to humanity, can be seen changing into reality with the advancement of technology and various solutions using those technologies. This paper presents smart parking system which can be used to address the traffic congestion issues. The proposed system enables a user to browse for available parking slots in particular area. It also provides facility to remotely reserve the convenient slots to the end users by the use of a mobile application. The proposed system uses a real-time database, which can detect a real-time availability of the parking spaces. The efforts made in this paper are aimed at enhancing the quality of life of people.

VII. Future Work

The parking slots can be distributed into different categories such as that for handicapped, electric vehicles and general parking to give a better estimate of the available number of slots depending on personal preference.

Vehicle verification can be introduced using number plate verification by installing a camera or sharing an one-time passcode(OTP) and entering it in the app at parking location.

Acknowledgment

We thank Prof. Behnam Dezfouli for the lessons and guidance which helped in successful completion of the project.

References

[1] Internet of things. (2018, June 17). Retrieved from https://en.wikipedia.org/wiki/Internet_of_things

[2] Sarah Mitroff (11th feb 2013) SOLVING PARKING WOES, ONE APP AT A TIME "https://www.wired.com/2013/02/parking-parkme/"

[3] McCoy, K. (2017, July 13). Drivers spend an average of 17 hours a year searching for parking spots. Retrieved from https://www.usatoday.com/story/money/2017/07/12/parking-pain-causes-financial-and-personal-strain/467637001/

[4] INRIX. (n.d.). Searching for Parking Costs Americans $73 Billion a Year | INRIX. Retrieved from http://inrix.com/press-releases/parking-pain-us/

[5] Doukas, C., Capra, L., Antonelli, F., Jaupaj, E., Tamilin, A., & Carreras, I. (2015, January). Providing generic support for IoT and M2M for mobile devices. In Computing & Communication Technologies-Research, Innovation, and Vision for the Future (RIVF), 2015 IEEE

[6] Teach, Learn, and Make with Raspberry Pi https://www.raspberrypi.org/

[7] Select Firebase Realtime Database | Firebase Realtime Database | Firebase https://firebase.google.com/docs/database/

[8] https://aws.amazon.com

[9] Khanna, Abhirup & Anand, Rishi. (2016). IoT based Smart Parking System. 10.1109/IOTA.2016.7562735.

[10] Kafle, V. P., Fukushima, Y., & Harai, H. (2015, April). ID-based communication for realizing IoT and M2M in future heterogeneous mobile networks. In Recent Advances in Internet of Things (RIoT), 2015 International Conference on (pp. 1-6). IEEE.

[11] Ji, Z., Ganchev, I., O'droma, M., & Zhang, X. (2014, August). A cloud based intelligent car parking services for smart cities.(URSI GASS), 2014 XXXIth URSI (pp. 1-4). IEEE.