

# Evolving One Touch Parking Reservation System

Chinmay Ashtikar<sup>1</sup>, Abhishek Naik<sup>2</sup>, Alim Manjiyani<sup>3</sup>, Sankalp Awasthi<sup>4</sup>

<sup>1,2,3,4</sup>B. Tech Student (CSE), Vellore Institute of Technology Vellore, Tamil Nadu, India

<sup>1</sup>chinmay.ashtikar[at]gmail.com

<sup>2</sup>abhishek281199[at]gmail.com

<sup>3</sup>manjiyani.alim[at]gmail.com

<sup>4</sup>awasthi.sankalp1000[at]gmail.com

**Abstract:** Car Parking is the most challenging task nowadays in metropolitan and crowded cities as there is no systematic and well maintained parking everywhere. We are trying to build an automated car parking reservation system as an android application which will book for slots at nearby location for parking and also come up with the new solutions to some of the early techniques and algorithm used in earlier applications of car parking

**Keywords:** Third Generation Partnership project (3GPP), Second Generation (2G), Third Generation (3G), Fourth Generation, Additive white gaussian noise (AWGN), Near Field Communication (NFC).

## 1. Introduction

We find many difficulties in our everyday life and one of them can be the car searching car parking in traffic. As our need expands our venturing out increments yet because of different increase in the utilization of vehicles and increment in the population can be the major reason for difficulties in car parking and in the busiest hours [4]. In the main and the busiest hours most of the parking spaces gets full and this leaves the client to search the other parking region which lead to congestion and increasing unnecessary traffic among the city in searching for car parking. To conquer this issue there is unquestionably a requirement for structured parking in business environment. We tried to structure it with reserving car slot in advance for parking the client's vehicle with time and cost [2]. Cost function ought to likewise join with the parking cost and closeness to the destination. Here we have tried to implement a economical and an easy to manage. It is quite difficult for the drivers to search for parking in the peak times at metropolitan areas. In disciplinary damage can be caused by parking cars. Thus, it is a necessity to provide the parking slot as much as needed and where it is needed that will only help the user park his car safely [1]. Basically, parking system being the most growing system for smart cities. Currently, most of the existing car parks do not have a systematic system. There are mostly parks that are manual and very less effective. Every user's demands should be the system should be user friendly with robust security features and efficient and there we come up with online parking booking.

## 2. Motivation and Aim

### a) Motivation

Due to increase in population of the world and increasing number of vehicles, parking has become a major issue people face and due to which also traffic is affected on road causing disturbances on the road and can also lead to road accidents. We will try to make a user interface for people as an android application where they can reserve and pay or book for a parking space by time [5]. Also, some safety functionalities for person's vehicle.

### b) Aim For The Proposed Work

- Create an application for unofficial parking area where local parking schemes and system are running without any official paper work [7].
- Create an application for both the admin handlers and users to add/book or view their respective parking slots with ease regarding the UI usage [2].

## 3. Tools and Methodology

For building android application, Networking and hardware sources we have used different software and tools discussed below [6].

### Android Studio

Android Studio is the official integrated development environment (IDE), built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is an open source software available for download on any majorly used operating system like macOS, Windows and linux. It is used after ADT tool which was the main IDE for native Android application development. The main IDE used for our application to exist was android studio where we have spent time to build the front end and integration to java backend API and databases.

### a) JAVA

Java is also one official language on android studio we have written are api's in traditional android studio java platform. Every functionality on the application is maintained in JAVA.

### b) NFC

Near-field communication (NFC) is a lot of communication conventions by which two electronic gadgets convey when they are brought inside 4 cm (1 1/2 in) of one another [7]. NFC gadgets are used in contactless frameworks, like those utilized in Master cards and shrewd cards, and permit versatile to supplant these frameworks. NFC gadgets can go about as remarkable character check [7]. It offers a low-speed association with straightforward arrangement that can be utilized to bootstrap increasingly fit remote associations.

c) AWS Load balancer

Elastic Load Balancing naturally disperses approaching application traffic over different targets, for example, Amazon EC2 instances, containers, IP addresses, and Lambda functions. It can deal with the differing load of your application traffic in a solitary Availability Zone or over different Availability Zones [4]. The Adding of the load balancer plays a vital role in handling requests from a larger number of users. Here the balancer works in a simple way that is instead of taking the request by time it generates a thread and assign it to a particular request while computing the given thread, if any smaller request is hit then the current computation will transfer the execution to the smaller thread completes its execution and handle the request response simultaneously [4]. In a deadlock scenario the server will handle computation through double server setup with the sense of adjusting the load through the actual load on the thread.

4. Design Approaches and Details

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use the scroll down window on the left of the MS Word Formatting tool bar [6].

a) Proposed Work Concept

Automated car parking system coordinated by android application" is a little than anticipated model of an automated car parking structure that can control and manage number of cars that can be left in given space at some irregular time reliant on the availability of parking opening. Automated parking is a strategy for parking and existing cars using distinguishing device [6]. Canny Parking structure arranged with an image planning office. The car would be left with the usage of lift at various levels. Here picture dealing with is used to get the number plate and store in database for connection with avoid illegal car segment. Number Plate Recognition strategy for making free car parking structure uses picture getting ready reason to process the number plates of the vehicles. In this structure, the image of the license number plate of the vehicle is obtained. It is also segmented to get particular characters in the number plate. In these also ultrasonic sensors are used to perceive free-parking openings.

b) Proposed Framework of Application



Figure 1: Framework of Car parking System

5. Task Solutions

1) Android Application Prototype

The application deals with the 2 part that is admin part and user end, the former is used to add the slots of parking to the database and the later one is used to add the slots in the booking allotment [2].

The slots can be viewed in the section and thus the booking can be visually be classified as weather the slots are booked or they are still left to be occupied by some another occupant. The flow of rate depends on the user accessibility on the slot area at given time for a given moment.[3]

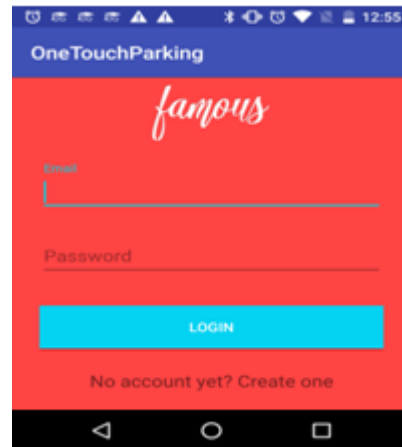


Figure 2: Login Page



Figure 3: Nearby parking spaces list



Figure 4: Adding Parking lot



Figure 5: Parking slots details

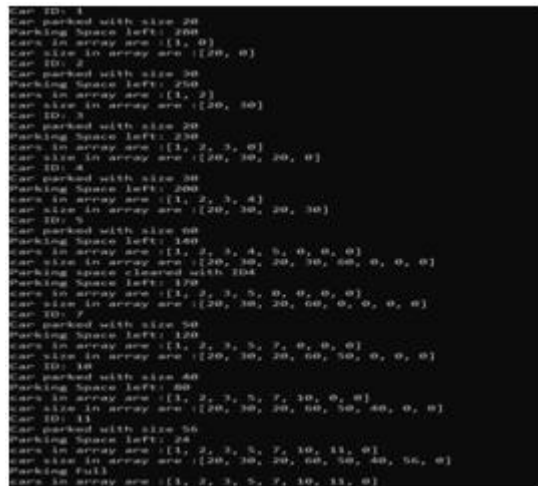


Figure 7: Dynamic slot allocation implementation output of the above test case

2) Dynamic Slot allocation

In this approach, we are not allocation any kind of slot for the vehicle. We are just generating a token number which is generated on the vehicle id and time [7]. It will be a runtime approach to decide row for the vehicle as it will depend on the size of the vehicle. The sonars placed on the entrance of the parking space will detect the size of the vehicle and slot is allocated as per the area availability in top down manner [4].

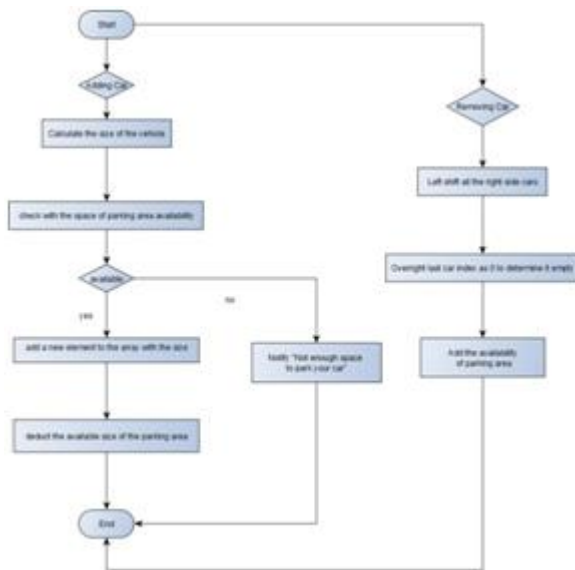


Figure 6: Flow Chart for dynamic memory allocation

Factors Affected:

- Searching time will be affected.
- Space utility
- Easier to control

**Test Cases**  
 array.addElement(1,20);  
 array.addElement(2,30); array.addElement(3,20);  
 array.addElement(4,30); array.addElement(5,6);  
 array.remove(3); array.addElement(7,50);  
 array.addElement(10,40) array.addElement(11,56);  
 array.addElement(12,30);

3) NFC Tag generation

For establishing NFC communication, we need two devices. Two part of the device, former is known as the initiator which is active device and later device is known as the target device. The target device might be active or passive [7]. The communication begins when the active device comes in the range of target device and generates a 13.56 MHz magnetic field and powers the target device. The target device can be a type of RFID card, tagora NFC device which gives the answer to initiator's requests.

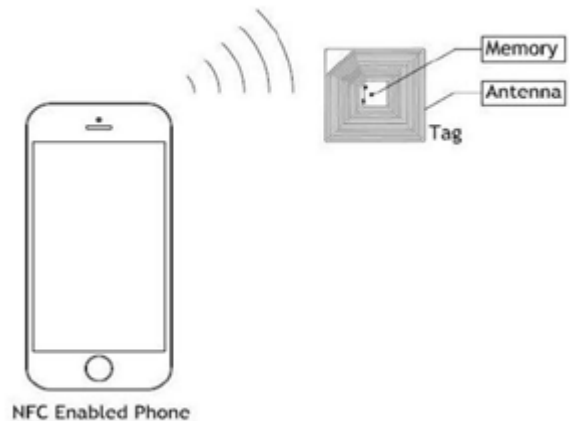


Figure 8: NFC Tag

Firstly, the client is required to book a space by signing in the OneTouch Parking Application. After login, he is asked to provide his/her NFC ID which acts as a unique ID related with each NFC supporting device. Then the client should book a space for parking according to the availability. As the client books a space, his/her NFC ID is added to the database. At the entry gate, as the client enters the parking space then he/she has to flash his/her NFC supporting device to the NFC reader [7]. Now the system will check whether this NFCID is present in the database, if it does the gates will open for that client and the time stamp will be noted and the client can take the vehicle to the client's convenient parking space. If this NFC ID do not exist in the database, client will not be permitted to enter the parking space.

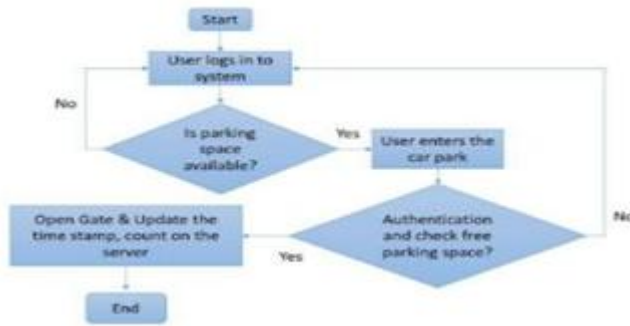


Figure 9: Flow chart of NFC Working (1)

When the client is finished parking, he/she will be incited to continue to the exit point. The client has to flash NFC ID again at the exit point. Now the system will search for the NFC ID in the database. If it exists in the database, the exit gates will open and the database entry of that NFCID will be erased. This will guarantee that the client books the space again at what ever point required. If the NFCID does not exist in the database, the client will be black listed from the booking slot in the application and appropriate moves will be made. Timestamp at the exit gate will be noted. Now both the time stamps will be used to calculate the fee and the client will be asked to pay the fee before exit.



Figure 10: Flow chart of NFC Working (2)

**Cloud Load Balancer**

As we can see that the current server is handling the 3 request from the user, as the old convention way of handling is based on the time of request is reorder through which the server handles the respective request, so as above stated scenarios the user1 got the response first the user 2 and then 3 , the current situation will be working fine for a low load type, considering the heavy load on this type of server setup the real problem arises, then the user n will take n time and user n+m will take more than (n+m) time to get the response, resulting in server crash[3].

The Server should be built and setup in a manner where extreme amount of the load won't affect the working of the instance. So, by use of Load balancer these things will not only get the benefit of more reliable servers in fact a faster one.

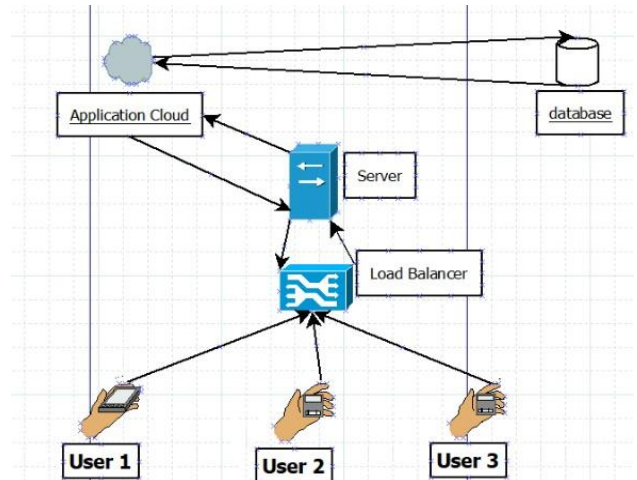


Figure 11: Load balancer servers

The Adding of the load balancer plays a vital role in handling requests from a larger number of users. Here the balancer works in a simple way that is instead of taking the request by time it generates a thread and assign it to a particular request while computing the given thread, if any smaller request is hit then the current computation will transfer the execution to the smaller thread completes its execution and handle the request response simultaneously [3]. In a deadlock scenario the server will handle computation through double server setup with the sense of adjusting the load through the actual load on the thread.

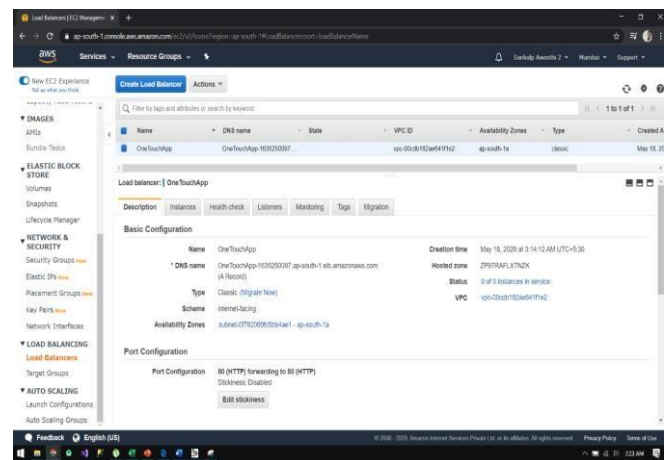


Figure 12: Load balancer status

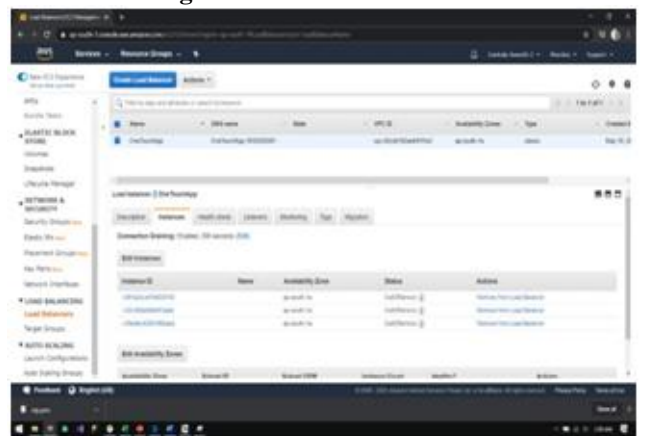


Figure 13: Deployed LB Running instances

## 6. Future Work

- We have tried implement the dynamic allocation property which can also be inherited with priority queue algorithm for better search time [7].
- More robustness in the properties are needed which are discussed above as in dynamic allocation of array we are shifting every vehicle at left a robust mechanism to do so or an alternative to a solution where we can eliminate that left shift[4].
- We tried to implement Aws load balancer to balance load on server to make application robust. There is uncertainty in the bulk of users using the application the solution should further be evolved[1].

## References

- [1] Trista Lin, Herve Rivano, Frederic Le Mouel. A Survey of Smart Parking Solutions. IEEE Transactions on Intelligent Transportation Systems, IEEE,2017.
- [2] A Survey on “Smart Parking” System Faiz Shaikh, NikhilkumarB.S.,
- [3] Omkar Kulkarni, Pratik Jadhav
- [4] Sabnam, Masiha, Mousumi Das, and Parismita A. Kashyap. "Automatic Car Parking System." ADBU Journal of Engineering Technology 4(2016).
- [5] Renuka R., S. Dhanalakshmi “Android Based Smart Parking System Using Slot Allocation & Reservations”(2015).
- [6] Jhonattan J. Barriga “Smart Parking: A Literature Review from the Technological Perspective”(2017).
- [7] SitiFatimahAbdulRazak,ChoonLinLiew,ChinPoolLee,Ki anMing Lim,” Interactive Android-Based Indoor Parking Lot Vehicle Locator Using QR-code”(2015).
- [8] Isak Shabani, Besmir Sejdiu, Fatushe Jasharaj,” Consuming Web Services on Android Mobile Platform for Finding Parking Lots” (2015).