IOT based Smart Parking system

Amitha M.S[1], Arpitha A.M[1], Geetha S[1], Raviraj M[1], Chetana R[2]

Students, Department of Electronics and Communication, SJBIT, Bengaluru[1]
Associate Professor, Department of Electronics and Communication, SJBIT, Bengaluru[2]

---***---

Abstract - *Human population is increasing day by day. Increase in population leads to increase in number of vehicles that are being used for transportation. Apart from increased air pollution (which leads to global warming), the other main problem faced by people of metropolitan cities is difficulty in finding free parking space to park their vehicles. The inability to find free parking spaces not only leads to wastage of time, fuel and energy but also leads to traffic jam. In this study, we design and implement a prototype of IoT based Smart Parking System with reservation facility which helps drivers to find nearest vacant parking spaces and also provides an option to reserve them in advance. Thus it solves all above problems to certain extent.*

Key Words: **NodeMCU, Google App engine, Google compute engine, Google cloud platform, MIT App inventor, Python Script.**

1. INTRODUCTION

Transportation is one of the well-known problems in metropolitan areas around the world. Due to lack of efficient parking solutions drivers opt for blind searching to find free parking spaces. In blind searching, drivers purely depend on luck to find a vacant parking space. It often results in unnecessary roaming of vehicles to and fro. This type of search causes huge wastage of fuel every year leading to increase in carbon foot print.

This study aims at creating a reliable solution to above problems using recent technologies like IoT(Internet of Things), web based services such as Google cloud platform and open source IoT hardware platforms such as ESP8266 NodeMCU.

1.1 THE INTERNET OF THINGS (IoT)

The Internet of things (IoT) is the inter-networking of physical devices, vehicles (also referred to as "connected devices" and "smart devices"), buildings, and other items embedded with electronics, software, sensors, actuators, and network connectivity that enable these objects to collect and exchange data. The Global Standards Initiative on Internet of Things (IoT-GSI) defined the IoT as "the infrastructure of the information society". The IoT allows objects to be sensed or controlled remotely across existing network infrastructure, creating opportunities for more direct integration of the physical world into computer-based systems, and resulting in improved efficiency, accuracy and economic benefit in addition to reduced human intervention.

2. SYSTEM ARCHITECTURE

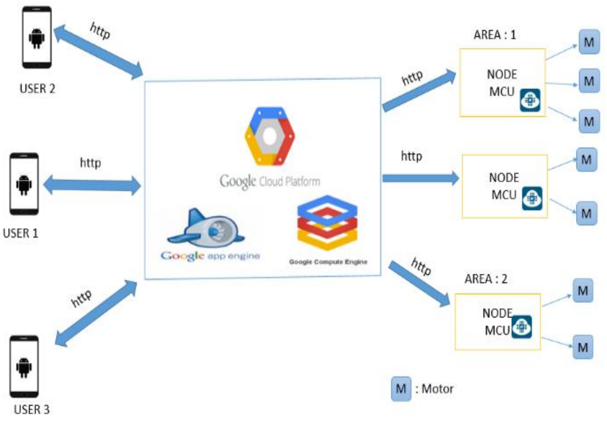

Fig -1: System Architecture

NodeMCU

Node MCU is an open source IoT platform. It includes firmware which runs on the ESP8266Wi-FiSoC from Espressif Systems, and hardware which is based on the ESP-12 module. The firmware uses the Lua scripting language.

Google cloud platform

Google Cloud Platform is a cloud computing service by Google that offers hosting on the same supporting infrastructure that Google uses internally for end-user products like Google Search and YouTube. Google Cloud Platform is a part of a suite of enterprise services from Cloud and provides a set of modular cloud-based services with a host of development tools. For example, hosting and computing, cloud storage, data storage, translations APIs and prediction APIs.

Google compute engine

Google Compute Engine is the Infrastructure as a Service (IaaS) component of Google Cloud Platform which is built on the global infrastructure that runs Google's search engine. Google Compute Engine enables users to launch virtual machines (VMs) on demand. VMs can be launched from the standard images or custom images created by users. GCE users need to get authenticated based on OAuth 2.0 before launching the VMs. Google Compute Engine can be accessed via the Developer Console, RESTful API or Command Line Interface.

Google app engine

Google App Engine is a cloud computing platform for developing and hosting web applications in Google-managed data centers. Applications are sandboxed and run across multiple servers. App Engine offers automatic scaling for web applications—as the number of requests increases for an application, App Engine automatically allocates more resources for the web application to handle the additional demand.

MIT App Inventor

An android application is developed using the MIT App Inventor.

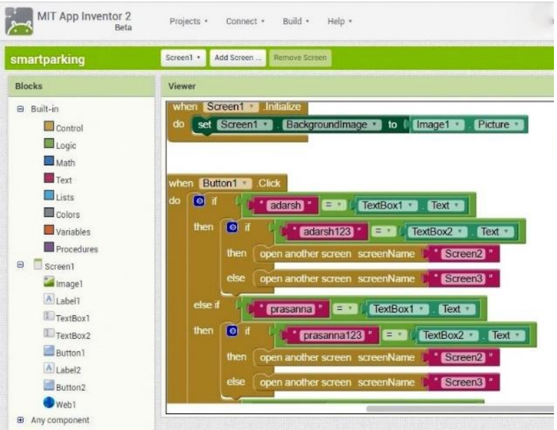
MIT App Inventor 2
Beta
Projects
Connect
Build
Help
smartparking
Screen1
Add Screen ...
Remove Screen
Blocks
Viewer
Built-in
Control
Logic
Math
Text
Lists
Colors
Variables
Procedures
Screen1
Image1
Label1
TextBox1
TextBox2
Button1
Label2
Button2
Web1
Any component
when Screen1 .Initialize
do set Screen1 . BackgroundImage to Image1 . Picture
when Button1 .Click
do if adarsh = TextBox1 . Text
then if adarsh123 = TextBox2 . Text
then open another screen screenName Screen2
else open another screen screenName Screen3
else if prasanna = TextBox1 . Text
then if prasanna123 = TextBox2 . Text
then open another screen screenName Screen2
else open another screen screenName Screen3

Fig -2: MIT App Inventor

App Inventor for android is an open-source web application originally provided by Google, and now maintained by the Massachusetts Institute of Technology (MIT).It allows newcomers to computer programming to create software applications for the Android operating system (OS). It uses a graphical interface which helps user to create application just by knowing execution logic.

3. WORKING PRINCIPLE

User end

At the user end, the user registers himself with the system by giving all relevant credentials. Once registered they are free to log into android app to view free parking slots in their area of interest. By convention all occupied slots are indicated in red, unoccupied slots in green and reserved slots in blue color.User can reserve an unoccupied parking slot. Cancellation of the reserved parking slots can also be done using android app.

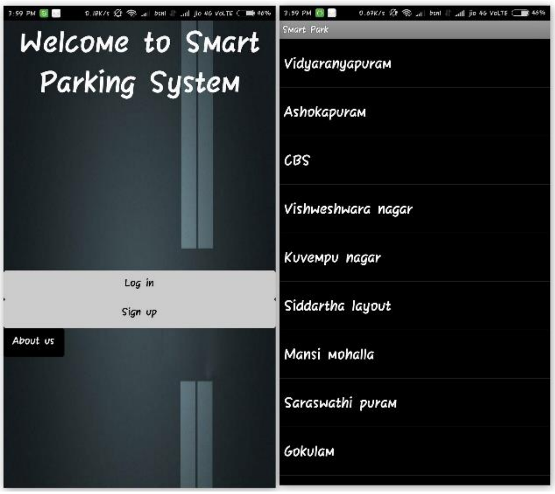

Fig -3: Android application

Server end

Google Cloud platform allow users to manage infrastructure, provisioning servers, and configuring networks.

A tinyWebDB database created in Google App Engine was used to store data sent by the android application. Two data structures were created for this application, out of which one stores user credentials like Username, password etc. which will be used during authentication process and the second data structure is used to maintain status of the parking slots i.e. booked, reserved or free.

A Virtual Machine(VM) was created on Google Compute Engine which comprised of 1vCPU, 3.75Gb memory and runs on Debian LINUX OS. Python – 3.6.1 was setup on the VM to run a python script.Python script will be continously monitoring the first database to serve the user requests. Any parking or reservation request from users for a particular parking slot of a particular parking area will be enrouted to that particular parking area proccessing unit i.e. NodeMCU. An efficient algorithm for reservation management is implemented using python script to solve speacial case problems.

Parking slot hardware end

Parking slot hardware consists of a group of parking units that are controlled by a single NodeMCU. Each parking unit has a parking bollard that is operated using a servo motor and group of leds that indicate the parking slot status. Whenever a parking request for a particular parking slot is received by NodeMCU, it commands servo motor of that slot to bring down the bollard and color of leds turn red and exactly opposite actions will be carried out when user exits the parking location. For all reservation requests received by NodeMCU, it commands group of leds to turn blue and position of parking bollard is unaltered.

4. CONCLUSION

In this paper an attempt to develop a proto type of IoT based Smart Parking System with Reservation Facility is made which helps in efficient parking management. Proposed system works well with good network infrastructure in all environmental conditions.

Based on the obtained results from testing we conclude that the proposed system can successfully reduce the problems of fuel wastage and traffic congestion caused by blind searching to a considerable amount. Enormous amount of time and effort put to find a free parking slot can be saved.

5. REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, “The Internet of things: a survey,” Computer Networks, vol. 54, no. 15, pp. 2787-2805, 2010.

[2] Kaivan Karimi and Gary Atkinson, ―What the Internet of Things (IoT) Needs to Become a Reality, White Paper, FreeScale and ARM, 2013.

[3] Choeychuen, K. Automatic parking lot mapping for available parking space detection. In Proceedings of the 5th International Conference on Knowledge and Smart Technology (KST), Chonburi, Thailand, 31 January–1 February 2013.