

---

# Internet of Things: Smart Parking - Project

---

Submitted By:

Name  
Aravind Maguluri  
Anurag Roy  
Bharat Kunj Gupta

Roll Number  
CS20MDS14011  
CS20MDS14010  
CS20MDS14017



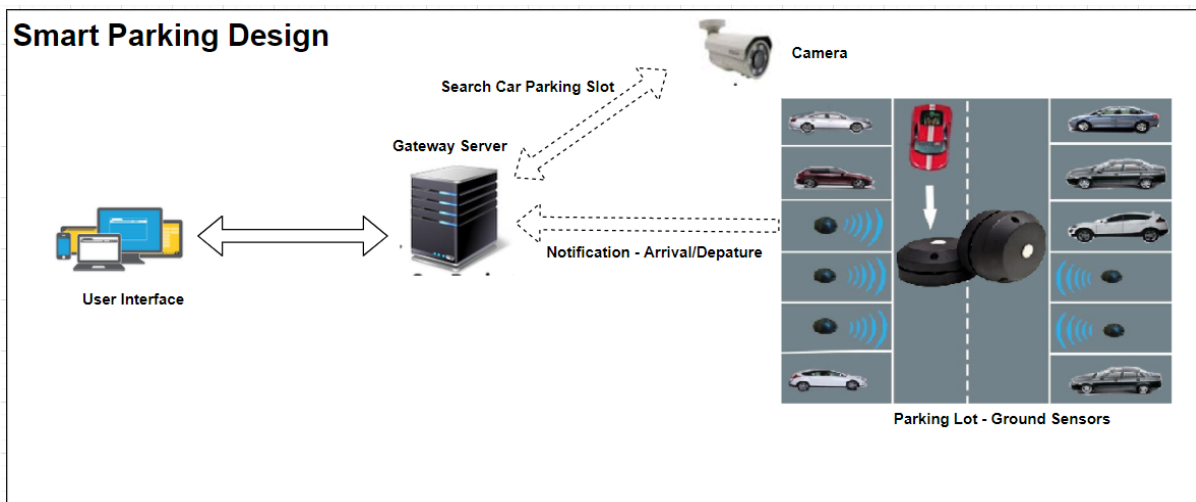
भारतीय प्रौद्योगिकी संस्थान हैदराबाद  
Indian Institute of Technology Hyderabad

M.TECH IN DATA SCIENCE(MDS) PROGRAMME  
DEPARTMENT OF COMPUTER SCIENCE  
INDIAN INSTITUTE OF TECHNOLOGY HYDERABAD

## Overview:

The increasing number of vehicles on the road, along with the mismanagement of parking space, leads to parking-related problems; thankfully, smart parking systems offer solutions. Smart Parking has emerged as a key internet of things (IoT) use case that falls under the umbrella of smart city concept. Smart parking solutions help to reduce congestion and pollution as they eliminate the need for vehicles to circle looking for spaces.

## High Level Design:



## System Components:

### Parking space sensor:

- This is a ground based sensor emulated using software
- When a car is parked in a parking slot or taken out from a slot, the sensor sends notification to Gateway Server

### Camera sensor:

- This is a camera that can take commands from the server and identify the location of car using OCR, this is emulated using software
- When a request is received to find a car with reg. plate details, camera finds the slot it is parked if present

### IOT Gateway Server:

- This is a server/hub with HiveMQ
- HiveMQ acts as a platform for communication between server and sensors

### User Interface:

- This is a simple command prompt that is available to the user to do the below queries
  - List free parking slots
  - Given car 4 digits, list all slots with matching car details

### ***Communication between components:***

- Parking space sensor only sends one-way updates to the Gateway server on the MQTT topic - smartparking/groundsensor/slot#
- Gateway server requests the camera sensor for the parking slot of a car with reg. details - smartparking/findcarbyplatedetails/request/
- Camera Sensor sends the details of the parking slots of the car given reg. details - smartparking/findcarbyplatedetails/response/
- User interface talks to the Gateway server to make requests

### ***Simulation of parking of cars:***

To simulate the movement of cars in and out of the parking lot, we are using data files to provide inputs to ground and camera sensors.

#### 1 - Ground sensor input file

- This file will not have car plate details
- This file will only have 1/0 depicting the occupied/vacant statuses of all ground sensors at any point of time
- Number of columns in file should be same as number of parking slots
- Each line in file is the simulated activity for all sensors and this is stepped through once every n seconds

#### 2 - Camera sensor input file

- This file will have car plate details for each parking slot
- Number of columns in file should be same as number of parking slots
- Each line in file is the simulated activity for all sensors and this is stepped through once every n seconds

\*Both files above should have same number of lines and columns to ensure same number of parking slots and steps through time

## **User Interface:**

Below is the prompt for the user interface.

### Smart Parking User Interface

---

Press 1 for listing free parking slots  
Press 2 for searching a car by its plates  
Press 0 to exit

## **Solution Approach & Observations:**

- **Ground Sensor:**
  - Each sensor loads its current occupied/vacant status from ground\_sensor.txt file
  - Each sensor is mapped to one parking slot during startup
  - Each sensor publishes its current status to the topic below
    - smartparking/groundsensor/slot#
  - Each sensor sleeps for 5 seconds before publishing its status to the topic
- **Camera Sensor:**
  - Camera sensor loads car details of each parking slot from ground\_sensor.txt file
  - Camera sensor subscribes to the topic below for listening to requests
    - smartparking/findcarbyplatedetails/request/
  - Once a request is received, matching car details are collected by sensor
  - Camera sensor publishes to the topic below
    - smartparking/findcarbyplatedetails/response/
  - Camera sensor sleeps for 5 seconds before loading most recent car details of slots
- **Gateway:**
  - Gateway is initialized by subscribing to all ground sensors topics
  - Gateway also subscribes to topic *smartparking/findcarbyplatedetails/response/*
  - When user requests for available slots, Gateway uses the most recent status from all ground sensors and prints available slots for parking
  - When user requests for a certain car with last 4 digits, Gateway publishes the car number on *smartparking/findcarbyplatedetails/request/* topic and waits for response on the *smartparking/findcarbyplatedetails/response/* topic
  - Once the response is received, it prints it on the terminal

### File Details:

src/  
Gateway.py  
camera\_sensor.py  
ground\_sensor.py  
run.sh

log/  
run\_log.log

data/  
camera\_sensor.txt  
ground\_sensor.txt

### Instructions:

Use the below command to run the utility

**sh run.sh**

### Screenshots / Message Log:

```
Smart Parking User Interface
-----

Press 1 for listing free parking slots
Press 2 for searching a car by its plates
Press 0 to exit

Key Input:1
Free slots available - ['slot-1', 'slot-2', 'slot-3', 'slot-5', 'slot-10']
```

```
Smart Parking User Interface
-----

Press 1 for listing free parking slots
Press 2 for searching a car by its plates
Press 0 to exit

Key Input:2
Enter last four digits of plate number 1632
{'Slot-2': 'Vehicle number - TS12AB1632'}
```

```
Smart Parking User Interface
-----

Press 1 for listing free parking slots
Press 2 for searching a car by its plates
Press 0 to exit

Key Input:2
Enter last four digits of plate number 9205
{'Slot-4': 'Vehicle number - HR12DQ9205', 'Slot-8': 'Vehicle number - TS12FE9205'}
```

For additional details, please refer to message logs in the file run\_log.log