Smart parking using machine learning

Presented by:

Rajiv Iyer

Sree Ganesha Chellappa

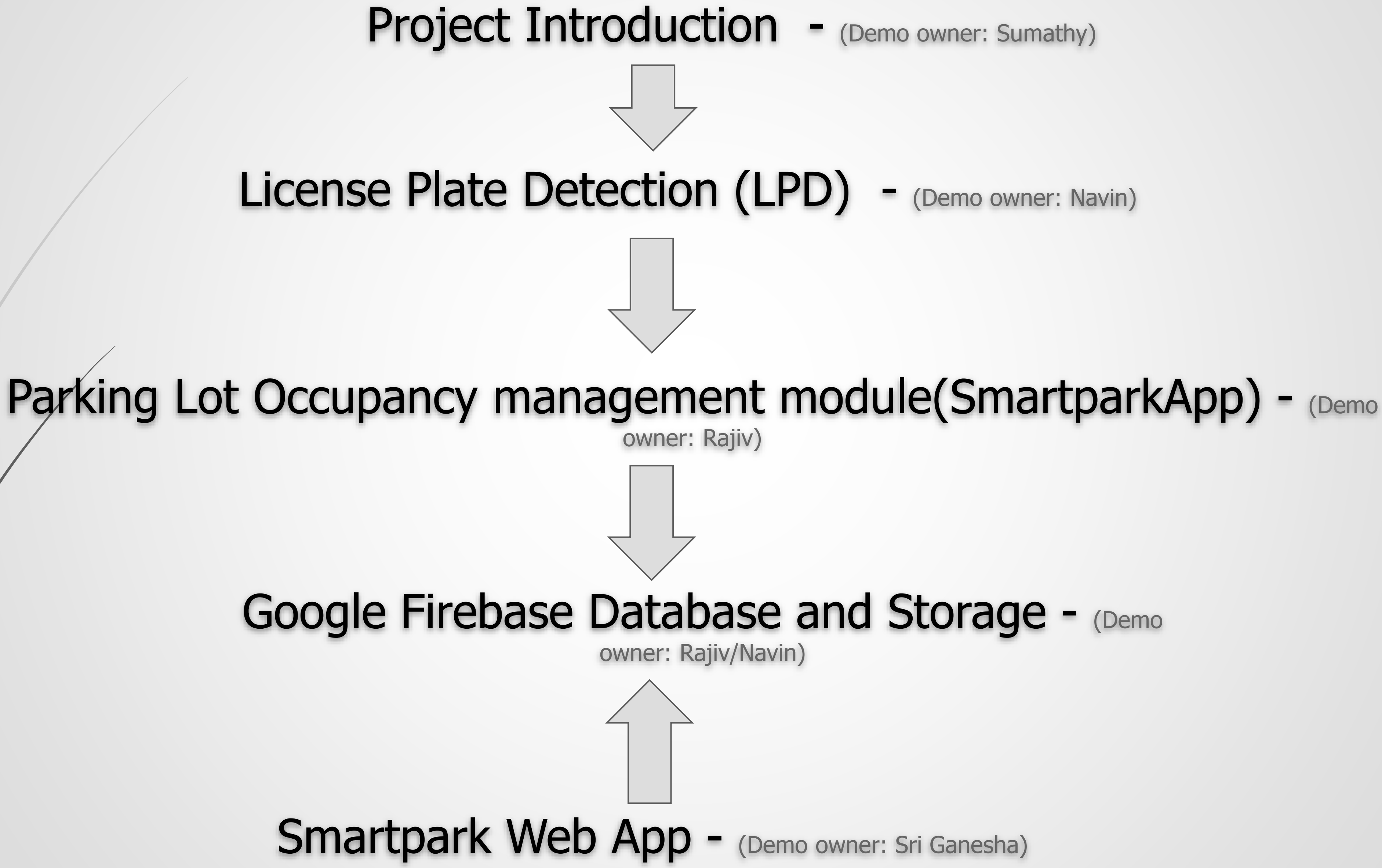
Navin Subbu

Muthu Sumathy Thevar

Under Guidance of

Prof. Hema Raut

Presentation structure :



Project Introduction - (Demo owner: Sumathy)

License Plate Detection (LPD) - (Demo owner: Navin)

Parking Lot Occupancy management module(SmartparkApp) - (Demo owner: Rajiv)

Google Firebase Database and Storage - (Demo owner: Rajiv/Navin)

Smartpark Web App - (Demo owner: Sri Ganesha)

Introduction

- In a country where it took 60 years to acquire 100 million vehicles but added another 100 million in just the next ten years, free and illegal parking has become both a serious urban planning and public health issue.
- The system makes use of live feed from CCTV cameras
- Detection of number plate of the car is done
- License Plate Detection (LPD) system plays an important role in many applications like electronic payment system like toll payment and parking fee payment.
- In this project recognition of the license plate is done, time stamping them and carry out the prediction of their entry and exit time and a database is created. On exiting, the same are validated.
- During setup of the system, the total available parking spots can be configured as Members and Outsiders/Visitors only spots.
- The parking spaces will consist of UDM sensors linked to NodeMcu/ESP32 Wi-Fi board that detects the presence of car parked. This will give a confirmation that the car has been parked successfully in that parking lot. When the car is moved back from the space, the slot is retained back in the system.

LITERATURE SURVEY

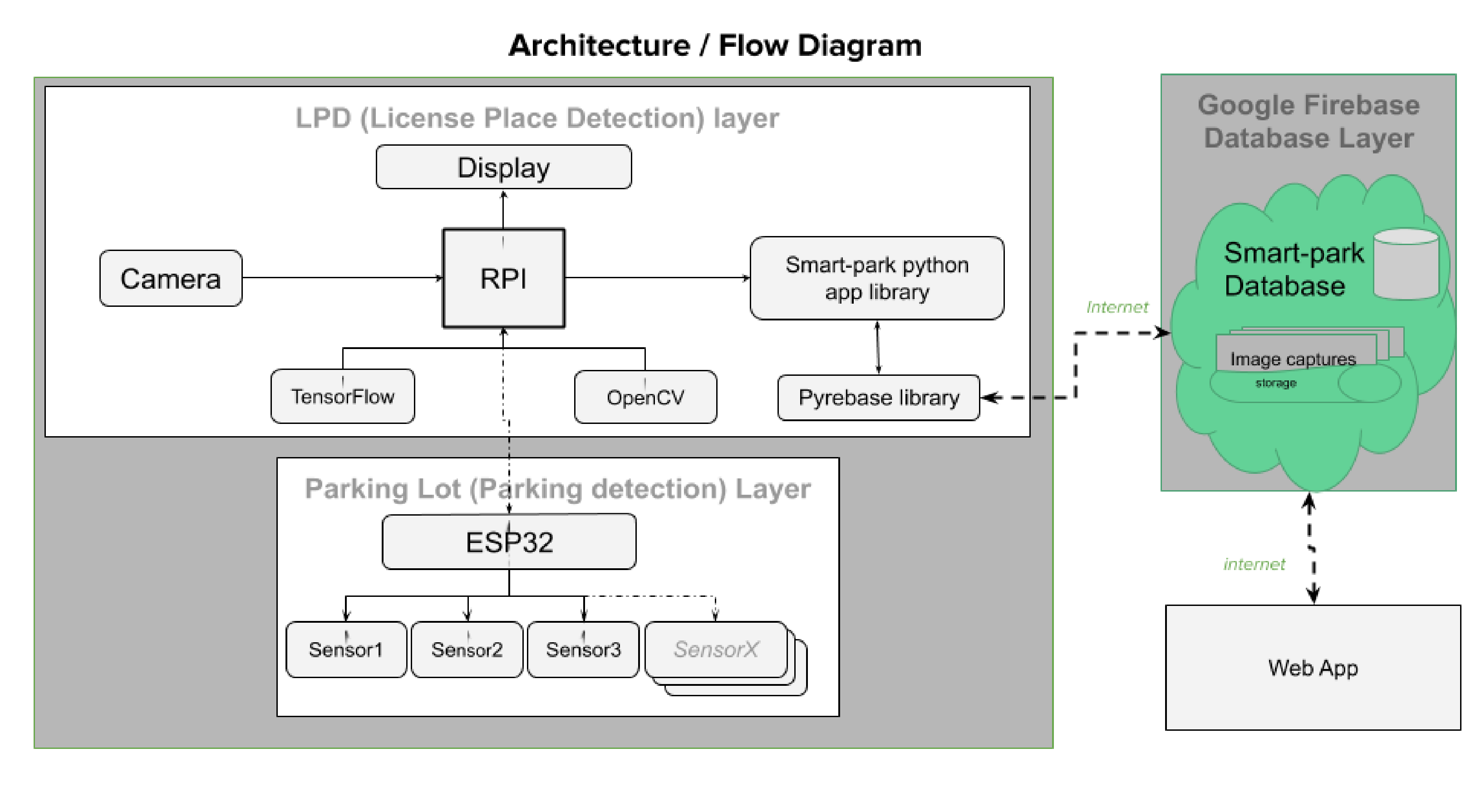
PROPOSED SYSTEM

- In this proposed system,main purpose is to build a parking system to save time and to increase security.
- Our parking system has an organized way of allotting residents/employees a permanent parking space.
- People waste time in finding parking spots to park their car so we can create a system efficient enough to overcome this time barrier.
- When the car enters the parking area the camera detects the number plate noting the time of entry. There will be a database consisting of details of the car owner if he is a resident/employee. If an outsider wants to park the car, the system will check for free outsider designated parking spaces.
- By using Time Stamping we will be able to assign a resident’s/employee’s spot to a visitor/outsider if he/she arrives on out of bounds time. We can increase security by using Time Stamping.

METHODOLOGY

- The project aims to take all the efficient aspects of the projects proposed in the literature survey and integrate it with an efficient feedback to be part of a solution for the parking problem.
- To understand the properties of NodeMCU ESP 8266 module
- To understand the basic python functions.
- To create a database using Google firebase.

FLOW DIAGRAM


Architecture / Flow Diagram
LPD (License Place Detection) layer
Display
Camera
RPI
Smart-park python app library
TensorFlow
OpenCV
Pyrebase library
Parking Lot (Parking detection) Layer
ESP32
Sensor1
Sensor2
Sensor3
SensorX
Internet
Google Firebase Database Layer
Smart-park Database
Image captures
storage
internet
Web App

HARDWARE AND SOFTWARE REQUIREMENTS

- Hardware Components:-
- Raspberry Pi Camera
- Raspberry Pi
- Ultrasonic distance measure
- NodeMCU ESP 8266

- Software Tools:-
- Anaconda Navigator
- Python and it’s modules
- Visual Studio Code
- Tensorflow
- Arduino Ide

System implementation

- System consists of following process :-
- License Plate Detection Using Tensorflow and OpenCV
- Image segmentation and Optical Character Recognition (OCR)
- Firebase Realtime Database and Storage
- Parking Lot Management System (Smart Parking)

License Plate Detection Using Tensorflow and OpenCV

- Using Tensorflow we trained a custom object detection model for detecting the object, in this case the license plate of a car and capture an image
- The hardware requirement for this is not much, most of the modern computers with a decent GPU will perform well.
- OpenCV is used to stream a live video from the video camera installed on the boom barrier at the entrance.
- The video captured is processed through the trained model, detection algorithms find the most probable locations of objects and draw bounding boxes.
- If the object is found, the bounding box that shows the objects passes the coordinates and the score which are then displayed in green on the live video.
- When the score of the detected object is greater than the desired value, an image is captured with the timestamp of when it was captured and stored in a local storage.

Image segmentation and Optical Character Recognition (OCR)

- The license plate was detected from the live feed and the frame of the video has been captured.
- The image is the processed for increasing the accuracy of OCR.
- The image is resized and converted to grey scale image.
- It is then processed through bilateral filtering.
- Edge detection is performed to detect the edge of the license plate.
- Contouring is to extract the contour with the license plate.
- Then masking is performed to isolate the contour and the cropped.
- The cropped frame is then processed for OCR.
- Optical character recognition in Python is done with the help of Google’s Tesseract Library.

Firebase Realtime Database and Storage

- Google firebase is an online real-time database management system.
- The feature is used to store the license plate number and the snapshot of the car from the front when it was detected.
- When the license plate is detected and processed through OCR, the text string is passed as a license parameter along with image of the car and current timestamp to smartparkapp.py module and is pushed to firebase.
- Simultaneously, the image captured is saved in the firebase storage for future references.

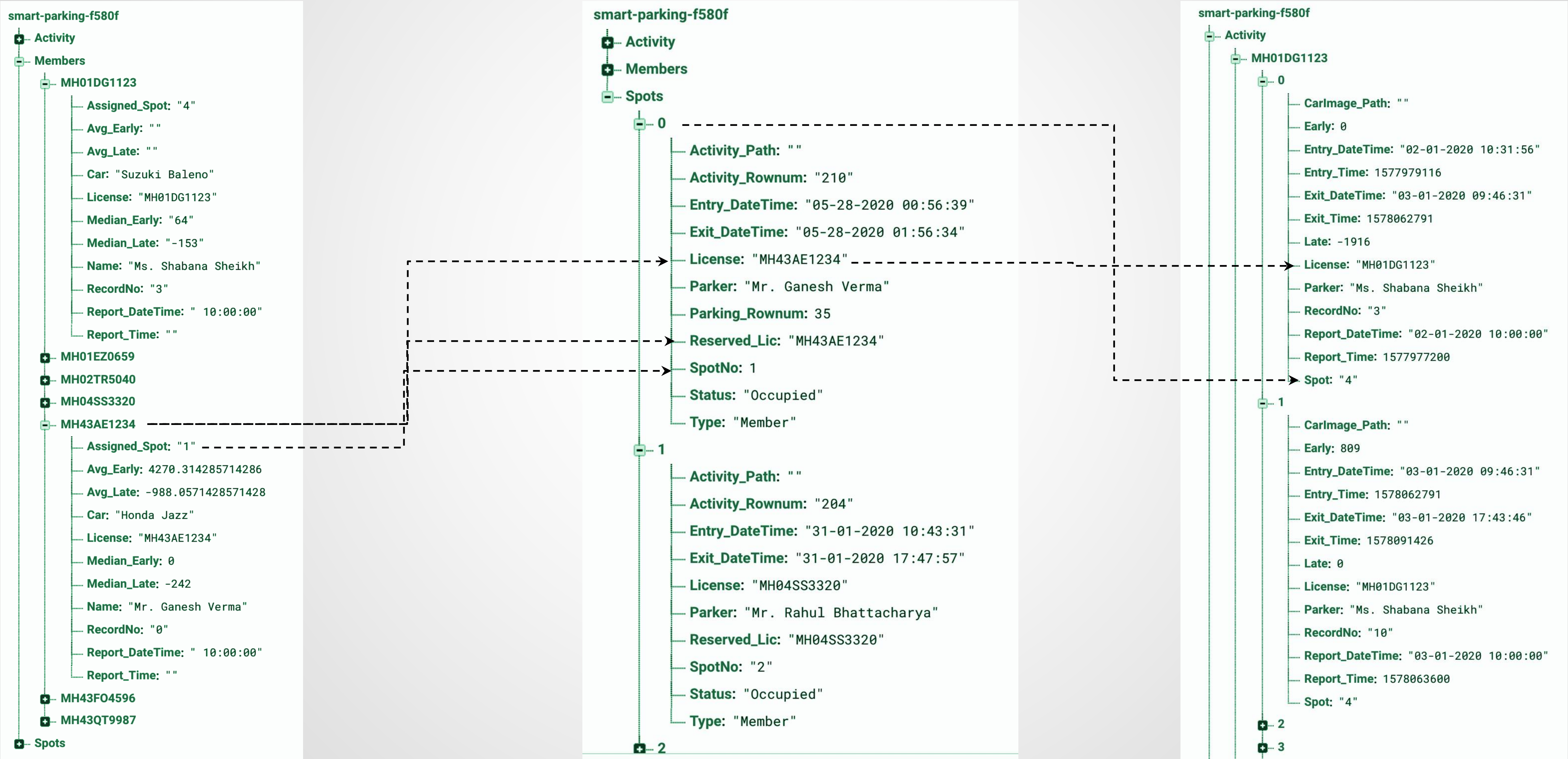
Google Firebase Database and Storage - (Demo owner: Rajiv/Navin)

Smartpark database schema
(https://smartpark-914f1.firebaseio.com/)

Members

Spots

Activity



```
smart-parking-f580f
  Activity
  Members
    MH01DG1123
      Assigned_Spot: "4"
      Avg_Early: ""
      Avg_Late: ""
      Car: "Suzuki Baleno"
      License: "MH01DG1123"
      Median_Early: "64"
      Median_Late: "-153"
      Name: "Ms. Shabana Sheikh"
      RecordNo: "3"
      Report_DateTime: " 10:00:00"
      Report_Time: ""
    MH01EZ0659
    MH02TR5040
    MH04SS3320
    MH43AE1234
      Assigned_Spot: "1"
      Avg_Early: 4270.314285714286
      Avg_Late: -988.0571428571428
      Car: "Honda Jazz"
      License: "MH43AE1234"
      Median_Early: 0
      Median_Late: -242
      Name: "Mr. Ganesh Verma"
      RecordNo: "0"
      Report_DateTime: " 10:00:00"
      Report_Time: ""
    MH43FO4596
    MH43QT9987
  Spots
```

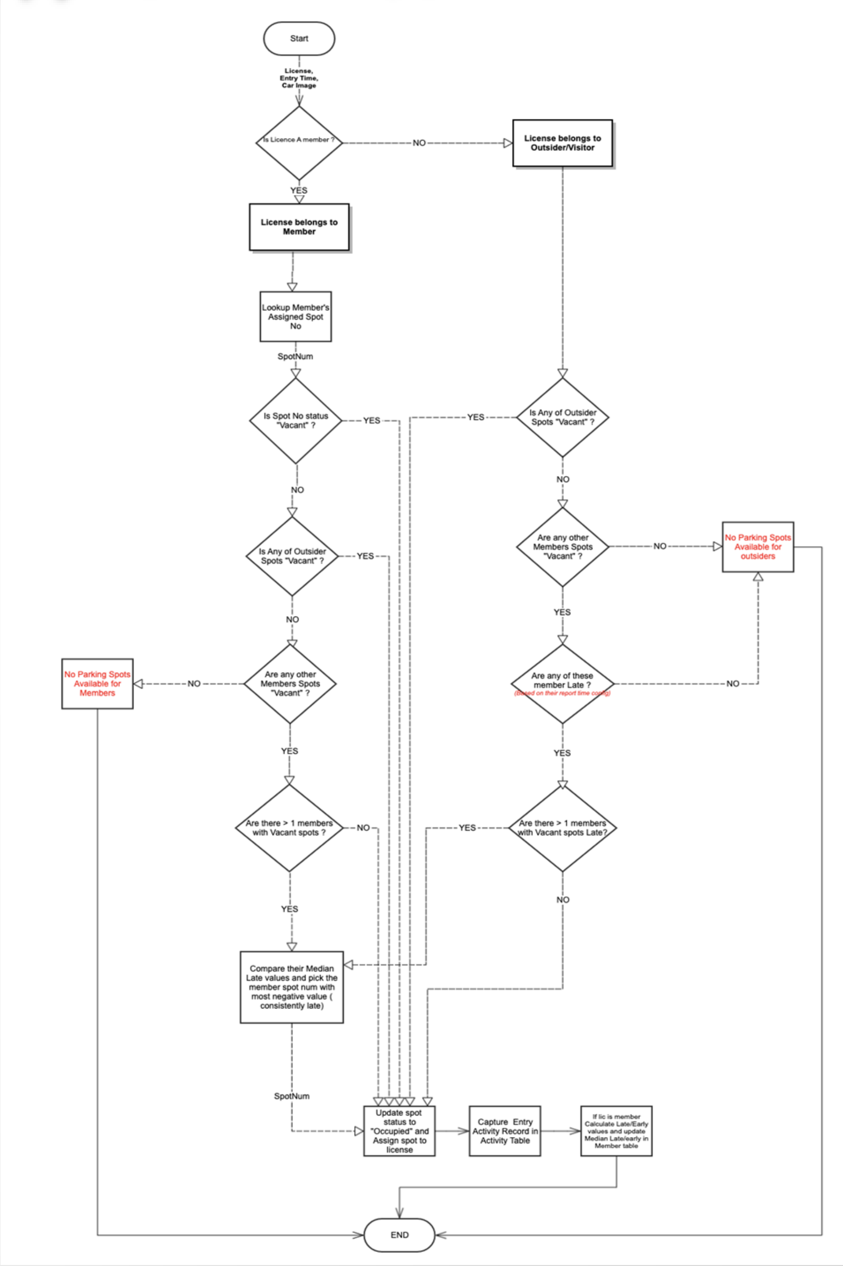
```
smart-parking-f580f
  Activity
  Members
  Spots
    0
      Activity_Path: ""
      Activity_Rownum: "210"
      Entry_DateTime: "05-28-2020 00:56:39"
      Exit_DateTime: "05-28-2020 01:56:34"
      License: "MH43AE1234"
      Parker: "Mr. Ganesh Verma"
      Parking_Rownum: 35
      Reserved_Lic: "MH43AE1234"
      SpotNo: 1
      Status: "Occupied"
      Type: "Member"
    1
      Activity_Path: ""
      Activity_Rownum: "204"
      Entry_DateTime: "31-01-2020 10:43:31"
      Exit_DateTime: "31-01-2020 17:47:57"
      License: "MH04SS3320"
      Parker: "Mr. Rahul Bhattacharya"
      Reserved_Lic: "MH04SS3320"
      SpotNo: "2"
      Status: "Occupied"
      Type: "Member"
    2
```

```
smart-parking-f580f
  Activity
    MH01DG1123
      0
        CarImage_Path: ""
        Early: 0
        Entry_DateTime: "02-01-2020 10:31:56"
        Entry_Time: 1577979116
        Exit_DateTime: "03-01-2020 09:46:31"
        Exit_Time: 1578062791
        Late: -1916
        License: "MH01DG1123"
        Parker: "Ms. Shabana Sheikh"
        RecordNo: "3"
        Report_DateTime: "02-01-2020 10:00:00"
        Report_Time: 1577977200
        Spot: "4"
      1
        CarImage_Path: ""
        Early: 809
        Entry_DateTime: "03-01-2020 09:46:31"
        Entry_Time: 1578062791
        Exit_DateTime: "03-01-2020 17:43:46"
        Exit_Time: 1578091426
        Late: 0
        License: "MH01DG1123"
        Parker: "Ms. Shabana Sheikh"
        RecordNo: "10"
        Report_DateTime: "03-01-2020 10:00:00"
        Report_Time: 1578063600
        Spot: "4"
      2
      3
```

Parking Lot Occupancy management module(SmartparkApp) - (Demo owner: Rajiv)

Some Key lot management requirements:

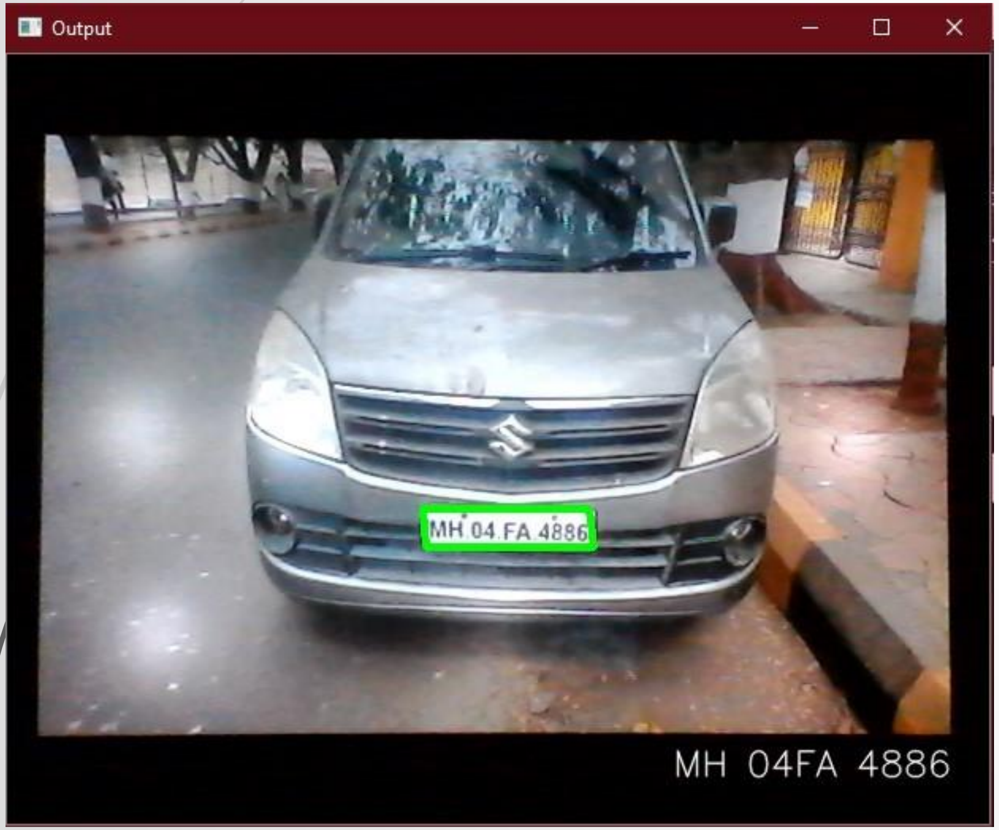
- Every member is pre-allotted a spot and gets the spot if its vacant on entry.
- Every member has a reporting time indicating expected arrival time.
- There are pre-designated spots for outsiders and outsiders are alloted that spot on entry
- If all outsider spots are occupied then outsiders get member spots provided that member is not close to his/her reporting time *(apply late arrival criteria)*
- If all member spots are occupied then members are alloted outsider spots provided they are vacant.
- If a member enters and his allotted spot and all outsiders are occupied then he is alloted another member’s spot. The choice of this other member is based on a ‘smart’ criteria *(apply member late median criteria)*


Start
License,
Entry Time,
Car Image
Is Licence A member ?
NO
License belongs to Outsider/Visitor
YES
License belongs to Member
Lookup Member's Assigned Spot No
SpotNum
Is Spot No status "Vacant" ?
YES
NO
Is Any of Outsider Spots "Vacant" ?
YES
NO
Is Any of Outsider Spots "Vacant" ?
YES
NO
Are any other Members Spots "Vacant" ?
NO
No Parking Spots Available for outsiders
YES
Are any other Members Spots "Vacant" ?
NO
No Parking Spots Available for Members
YES
Are any of these member Late ?
(Based on their report time config)
NO
YES
Are there > 1 members with Vacant spots ?
NO
YES
Are there > 1 members with Vacant spots Late?
YES
NO
Compare their Median Late values and pick the member spot num with most negative value (consistently late)
SpotNum
Update spot status to "Occupied" and Assign spot to license
Capture Entry Activity Record in Activity Table
If lic is member Calculate Late/Early values and update Median Late/early in Member table
END

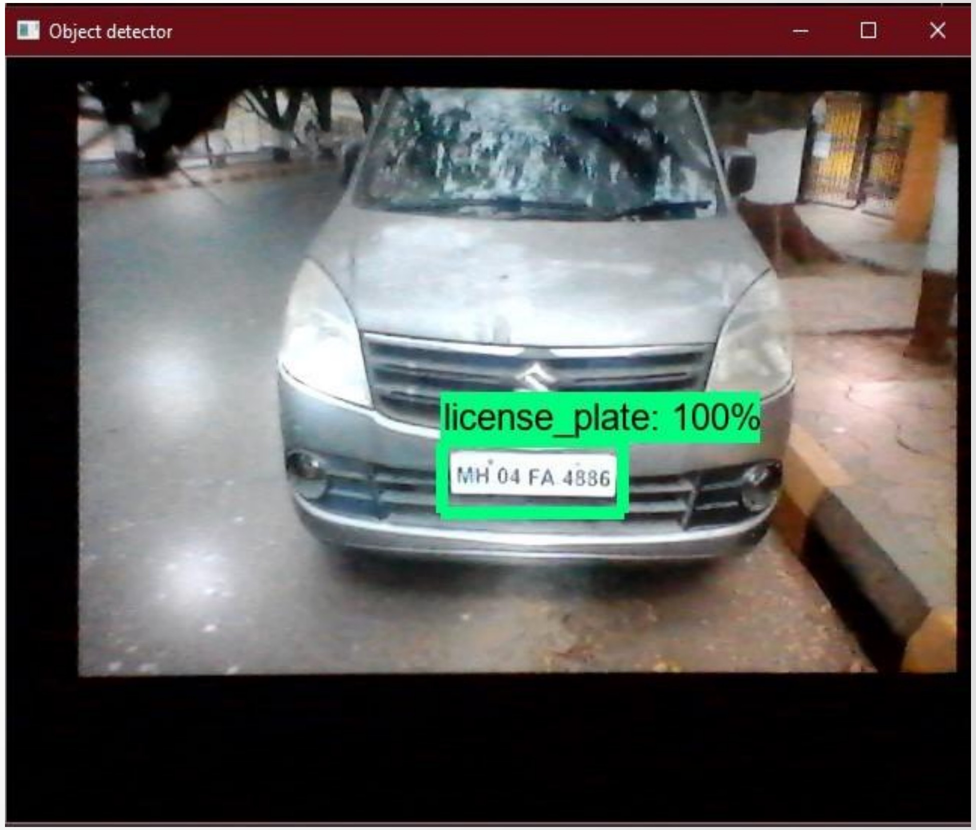
https://go.gliffy.com/go/publish/13366394

RESULTS

Object Detection Output



```
MH 04FA 4886
```



DEMO Usecase1: Navin

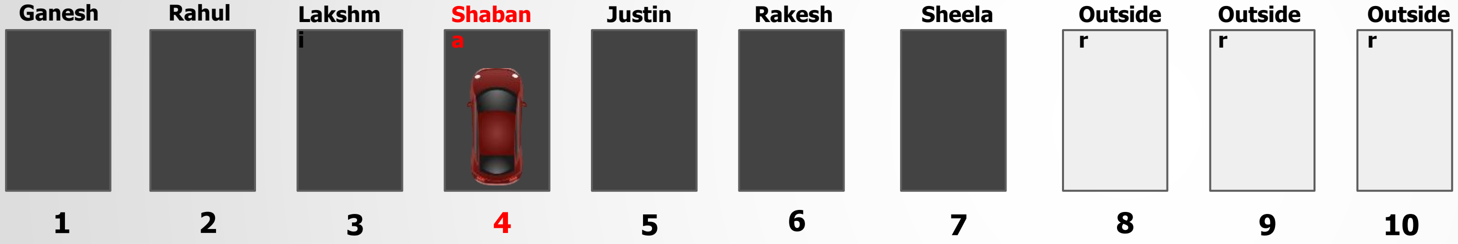
Demo use case 1 : LPD

▇ Member spot	□ Outsider spot
Member Car	Outsider Car

State 0: Empty parking lot. 7 are Member spots. 3 are Outsider spots.

Ganesh	Rahul	Lakshmi	Shabana	Justin	Rakesh	Sheela	Outsider	Outsider	Outsider
▇	▇	▇	▇	▇	▇	▇	□	□	□
1	2	3	4	5	6	7	8	9	10

State 1: LPD detects Shabana's car License **MH46AU1689** via camera, ML and OCR. Calls smartapp method RegisterCarEntry() with license Num. Smartapp recognizes Shabana as a member based on her license registered in the member tables and allots her Spot 4 which is pre-allocated to her. .



Ganesh	Rahul	Lakshmi	Shabana	Justin	Rakesh	Sheela	Outsider	Outsider	Outsider
▇	▇	▇	▇	▇	▇	▇	□	□	□
1	2	3	4	5	6	7	8	9	10

A Spot is allocated to Shabana. Shabana's car image should be captured in storage:

State 2: LPD detects Shabana's car License **MH46AU1689** via camera, ML and OCR. Calls IsCarParked() and then calls RegisterCarExit(). Shabana exits and vacates spot 4.

Ganesh	Rahul	Lakshmi	Shabana	Justin	Rakesh	Sheela	Outsider	Outsider	Outsider
▇	▇	▇	▇	▇	▇	▇	□	□	□
1	2	3	4	5	6	7	8	9	10

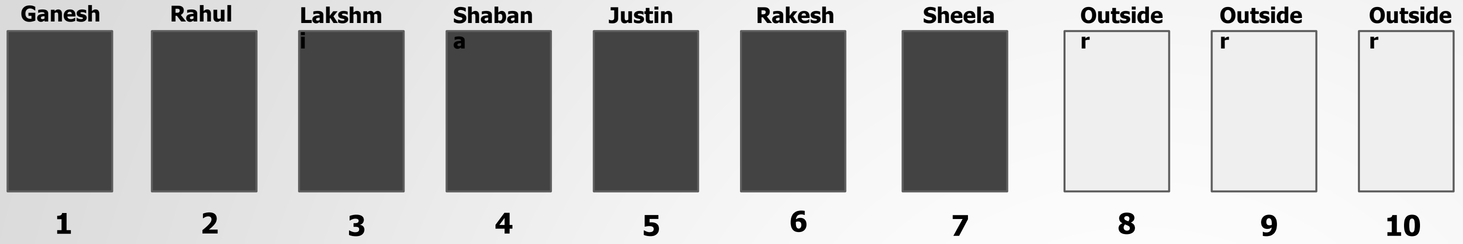
DEMO Usecase 2: Rajiv

Use case 2 : Complex parking scenario (1/3)

■ Member spot □ Outsider spot

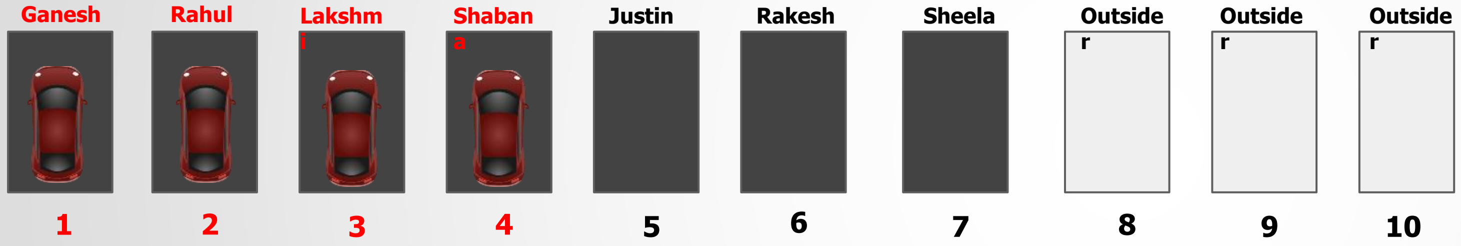
Member Car Outsider Car

State 0: Empty parking lot. 7 are Member spots. 3 are Outsider spots.



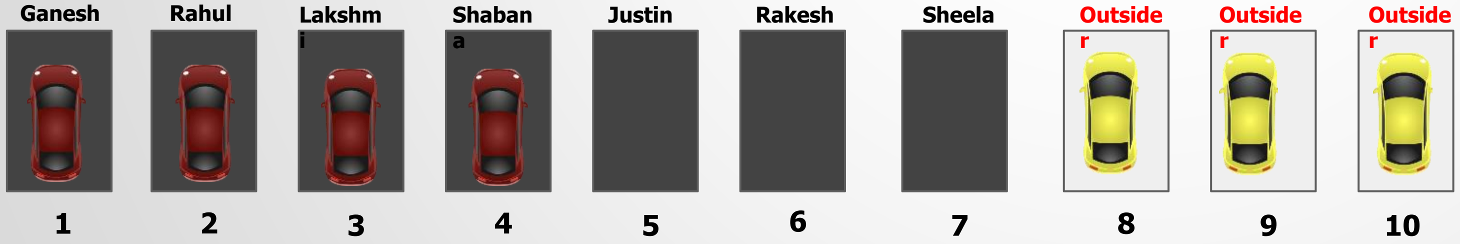
Ganesh 1
Rahul 2
Lakshmi 3
Shabana 4
Justin 5
Rakesh 6
Sheela 7
Outsider 8
Outsider 9
Outsider 10

State 1: 4 member cars enter.



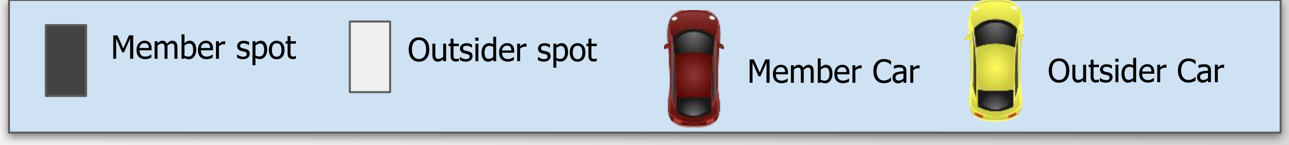
Ganesh 1
Rahul 2
Lakshmi 3
Shabana 4
Justin 5
Rakesh 6
Sheela 7
Outsider 8
Outsider 9
Outsider 10

State 2: 3 Outsider cars enter.

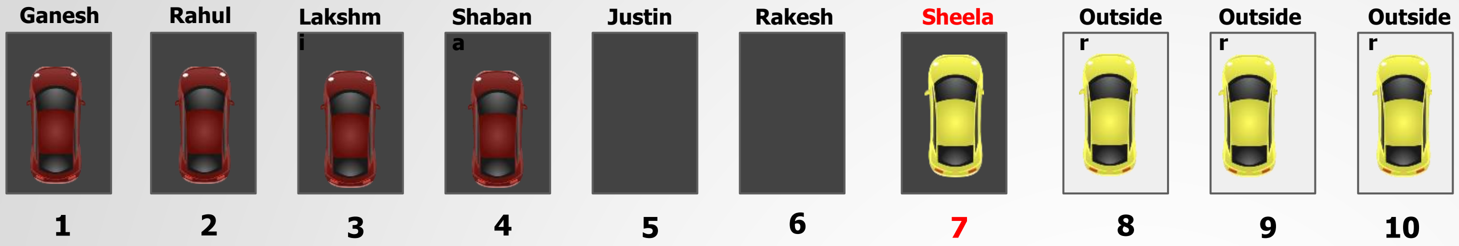


Ganesh 1
Rahul 2
Lakshmi 3
Shabana 4
Justin 5
Rakesh 6
Sheela 7
Outsider 8
Outsider 9
Outsider 10

Use case 2 : Complex parking scenario (2/3)

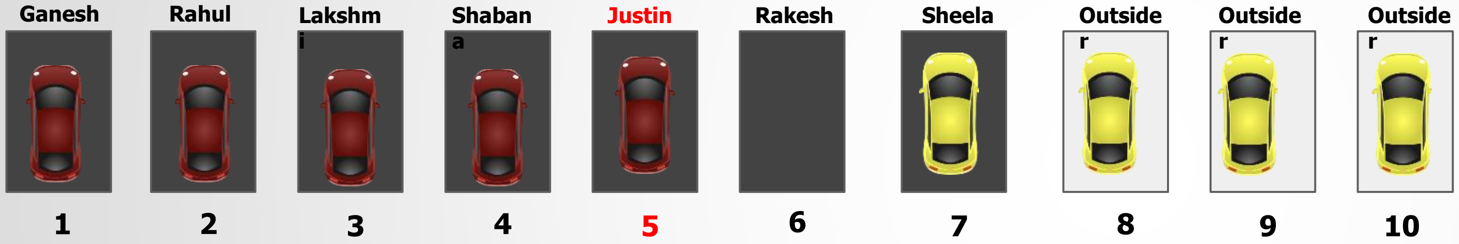


State 3: 4th outsider arrives at 10:30. Outsider is allocated Sheela's spot because of her median score.



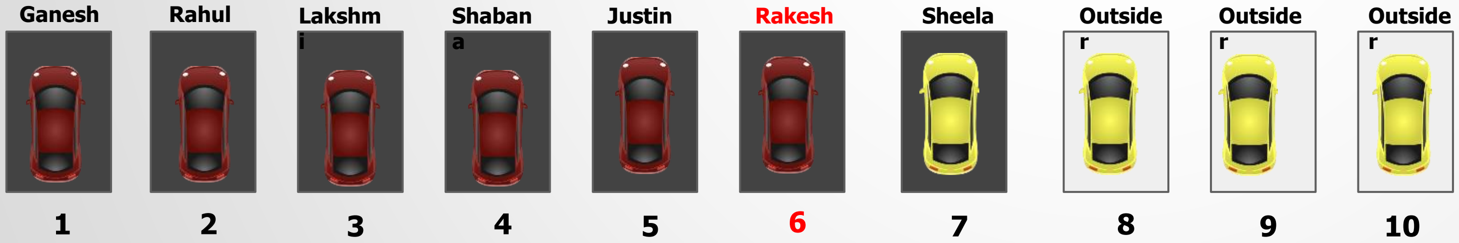
Late arrival Median values		
Justin	Rakesh	Sheela
-492	0	-769

State 4: 5th member Sheela whose spot has been occupied by outsider enters. She is allocated Justin spot.

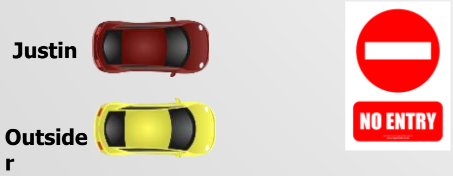


Late arrival Median values	
Justin	Rakesh
-492	0

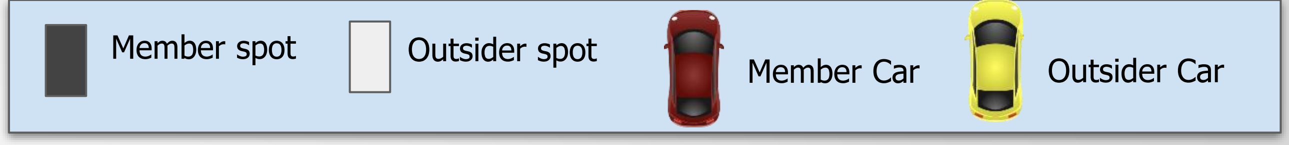
State 5: 3 Rakesh Arrives and occupies his own spot. All spots are now taken and parking lot is full



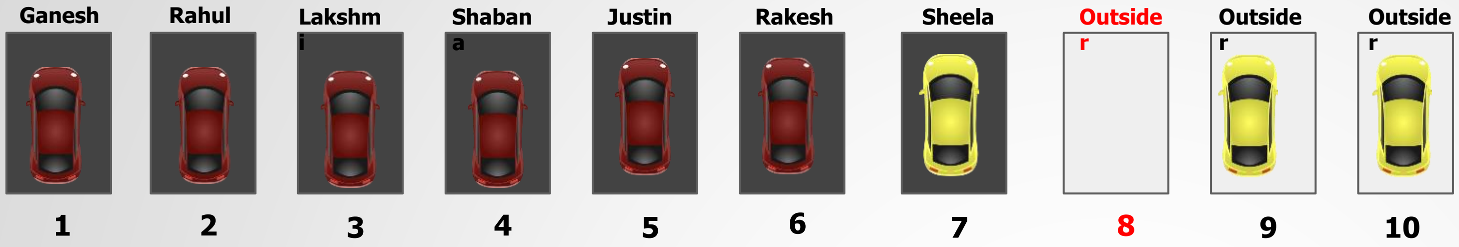
State 6: Justin, A member arrives is denied entry as lot is full. An outsider then arrives and he is denied entry as well.



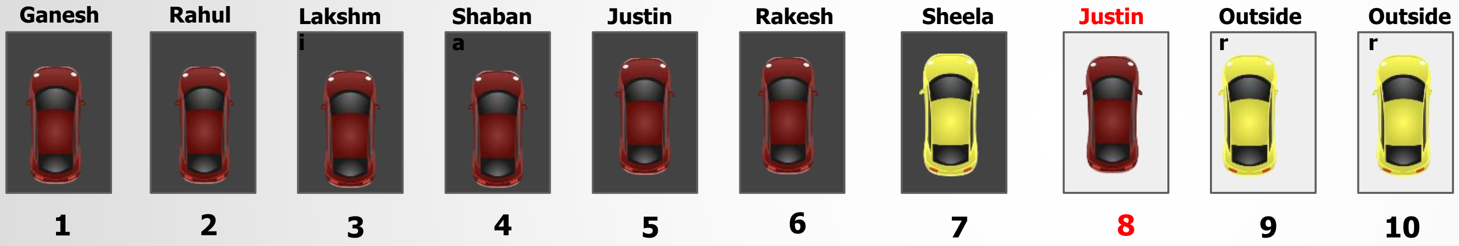
Use case 2 : Complex parking scenario (3/3)



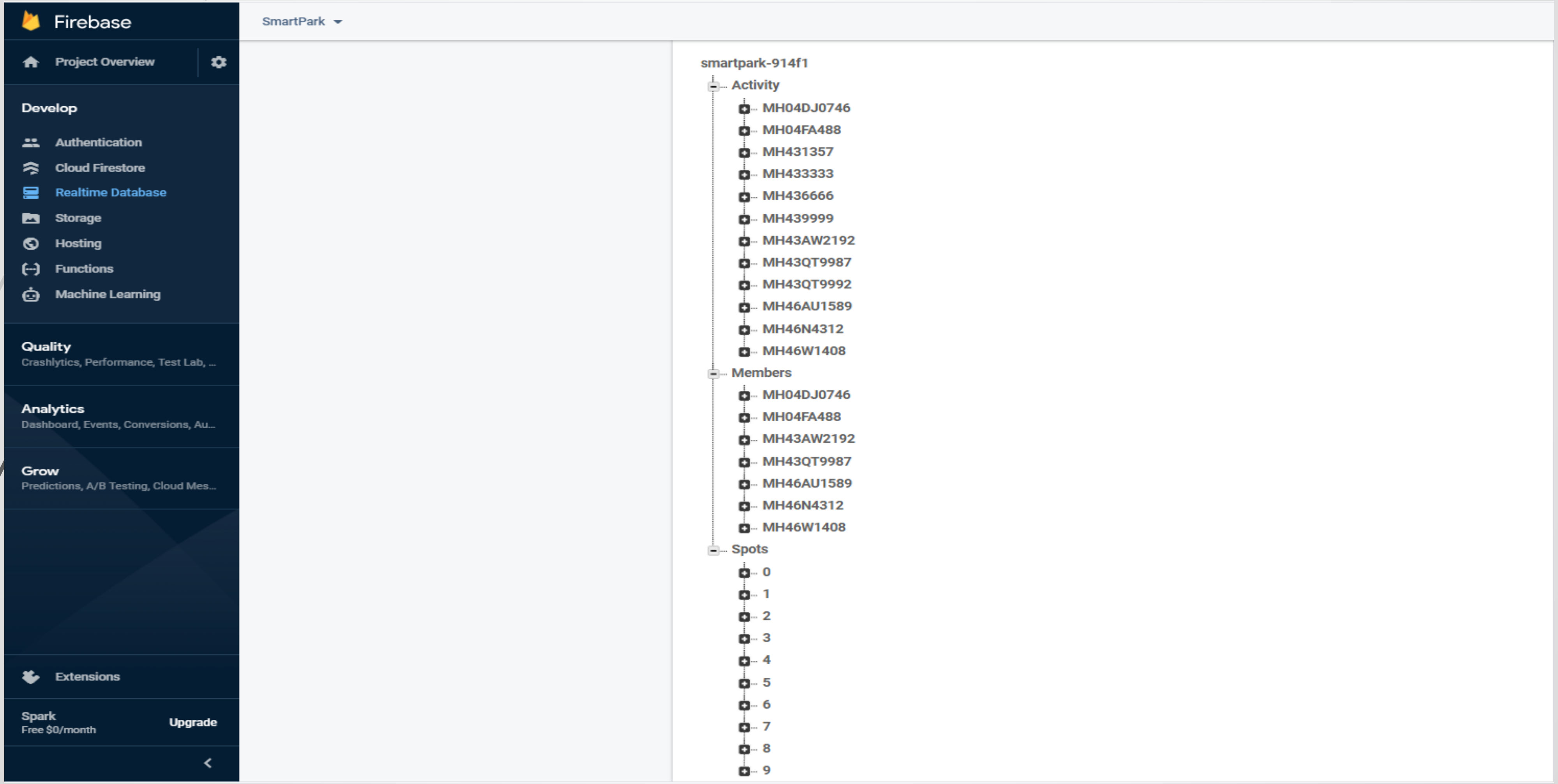
State 7: The 8th spot outsider exits. 8th spot becomes vacant.



State 8: Member Justin now attempts to enter again. He will be allotted the recently vacated outsider spot 8.

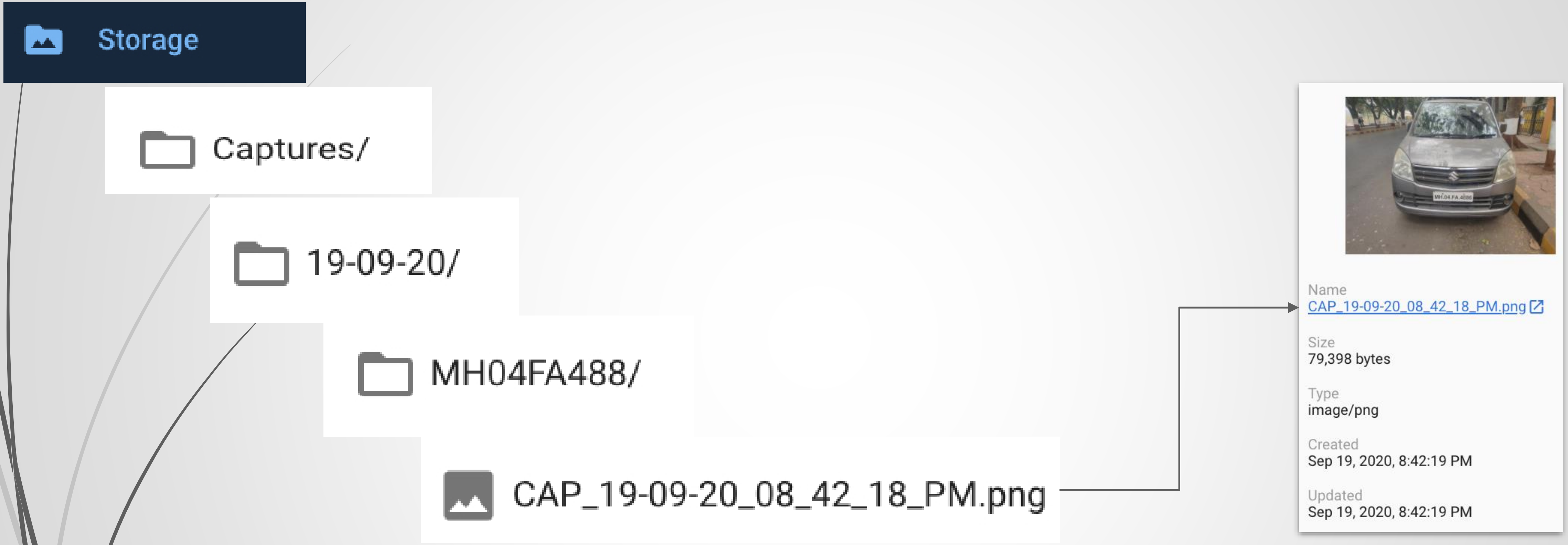


RealTime Database



```
smartpark-914f1
  Activity
    MH04DJ0746
    MH04FA488
    MH431357
    MH433333
    MH436666
    MH439999
    MH43AW2192
    MH43QT9987
    MH43QT9992
    MH46AU1589
    MH46N4312
    MH46W1408
  Members
    MH04DJ0746
    MH04FA488
    MH43AW2192
    MH43QT9987
    MH46AU1589
    MH46N4312
    MH46W1408
  Spots
    0
    1
    2
    3
    4
    5
    6
    7
    8
    9
```

Google Firebase Storage : Image capture hierarchy



Storage

Captures/

19-09-20/

MH04FA488/

CAP_19-09-20_08_42_18_PM.png

Name
CAP_19-09-20_08_42_18_PM.png

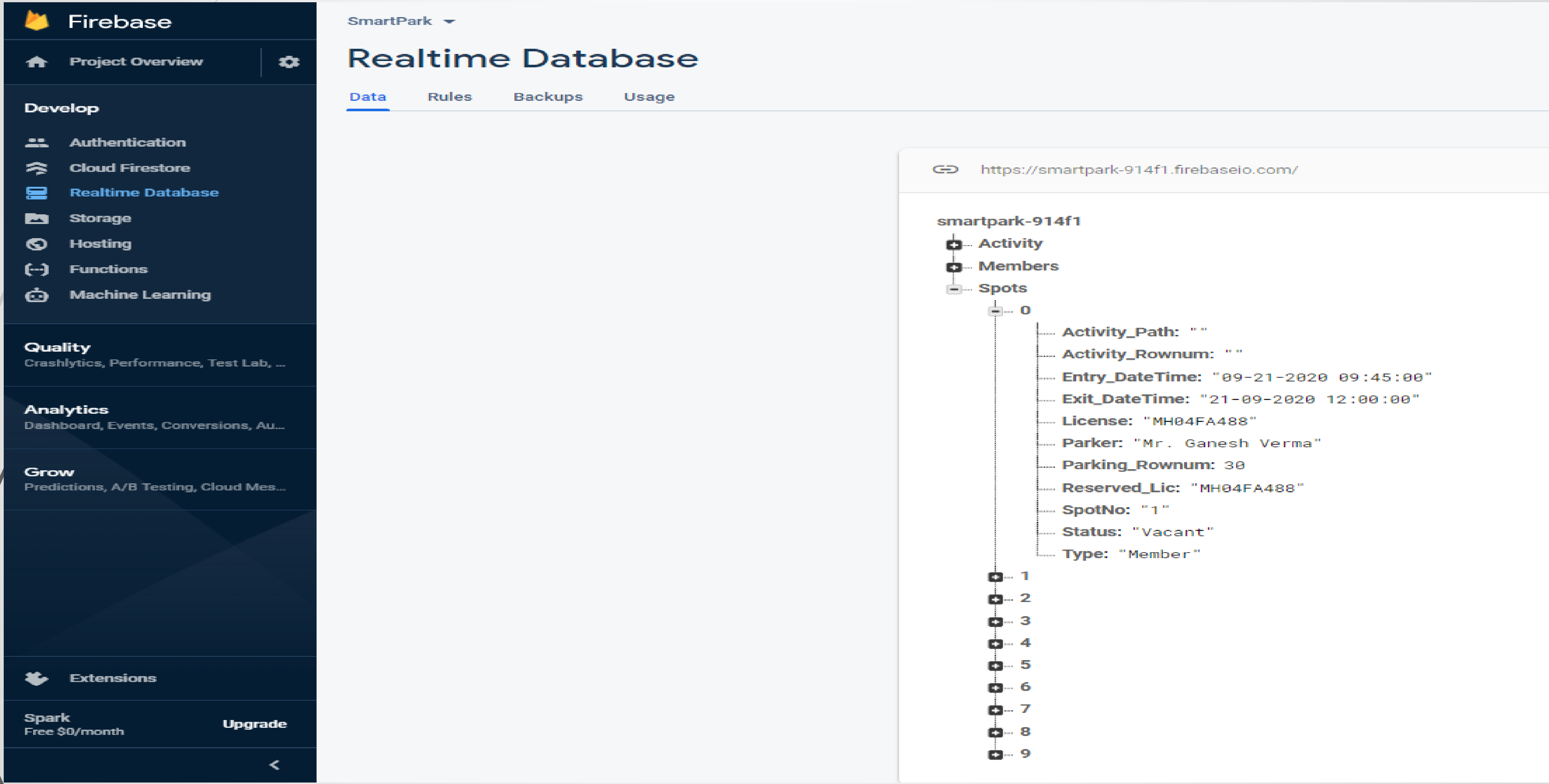
Size
79,398 bytes

Type
image/png

Created
Sep 19, 2020, 8:42:19 PM

Updated
Sep 19, 2020, 8:42:19 PM

Database Output



Realtime Database

https://smartpark-914f1.firebaseio.com/

```
smartpark-914f1
  Activity
  Members
  Spots
    0
      Activity_Path: ""
      Activity_Rownum: ""
      Entry_DateTime: "09-21-2020 09:45:00"
      Exit_DateTime: "21-09-2020 12:00:00"
      License: "MH04FA488"
      Parker: "Mr. Ganesh Verma"
      Parking_Rownum: 30
      Reserved_Lic: "MH04FA488"
      SpotNo: "1"
      Status: "Vacant"
      Type: "Member"
    1
    2
    3
    4
    5
    6
    7
    8
    9
```

FUTURE SCOPE

- In future the system can be extended which is not only specific to a private parking like malls, company parking, etc.
- It can also be implemented on public parking which has extending features such as reduction of traffic as search time for parking is reduced ,enhanced user experience as user can easily find the vacant parking lot, real-time data which is achieved by studying the arrival and leaving time of a particular car using ML,
- Decreased management costs as this reduces the labour cost and can be more efficient.
- Since data stored is real-time safety can be ensured which helps prevent parking violations and suspicious activity.

CONCLUSION

- The first approach of using existing surveillance cameras for detecting license plates is presented.
- An open source platform is used and improved to evaluate only objects that are moving.
- The algorithm run on a distributed platform that operate multiple cameras and sensors, to support comprehensive security management.
- The final output of the license plates was found to be very close to the actual value on the plates.

REFERENCES

Q&A and Feedback

Thank you