SMART PARKING SYSTEM PROJECT

Objective : To create a smart parking system that sends and alert when a parking lot gets occupied and also shows peak time during which parking lots are occupied

Components Required:

1.Node Mcu

2.Ultrasonic Sensor

3. Jumper Wires

Concept :

1. Ultrasonic sensor is used ti detect the presence of car in front of it using the node mcu.
2. Data from the ultrasonic sensor is sent to the node mcu.
3. Node mcu then sends this data to thingspeak , where a graph of the data sent gets plotted. This helps us know peak time at which cars are parked .
4. Whenever a car is detected in front of the sensor , a condition is triggered that sends a message saying 'Parking lots are occupied !'.

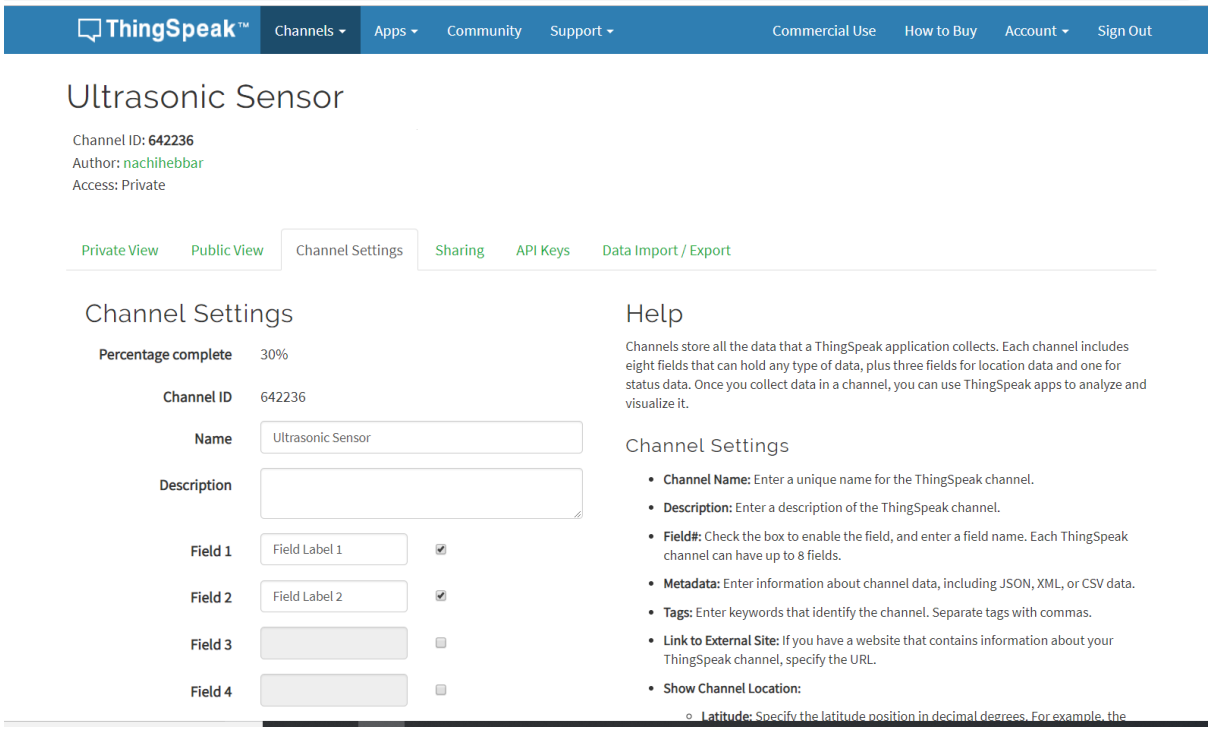
Here instead of message , we'll send a email to the user just to display different functionalities of iot.

Procedure :

1. Connect the ultrasonic sensor the node mcu in the following configuration :

Ultrasonic		Node Mcu
Vcc	-	Vin
Gnd	-	Gnd
Echo	-	D0
Trig	-	D1

2. Now to upload data onto thingspeak from your sensor , create an account on thingspeak and create and channel where you will upload data as shown.



You can create as many fields as you want , but we require only 1 as only one graph will be plotted.

3. Save the channel and note down the channel id and writeapikey given.

5.After completing the above steps , upload the code onto the node mcu using Arduino IDE.

Code :

thingspeak | Arduino 1.8.7

File Edit Sketch Tools Help

thingspeak §

```
#include "ThingSpeak.h"
#include <ESP8266WiFi.h>

//----------------  Fill in your credentails   ---------------------
char ssid[] = "wifi_name";              // your network SSID (name)
char pass[] = "1234567890";          // your network password
unsigned long myChannelNumber = 642236;  // Replace the 0 with your channel number
const char * myWriteAPIKey = "PBW1CN17C2C25QM4";    // Paste your ThingSpeak Write API Key between the quotes
//-----------------------------------------------------------------

WiFiClient  client;

int echo=D0;
int trigger=D1;

void setup() {
  //Initialize serial and wait for port to open:

  Serial.begin(115200);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for native USB port only
  }

  WiFi.mode(WIFI_STA);
  ThingSpeak.begin(client);
  pinMode(trig,OUTPUT);
  pinMode(echo,INPUT);
}

void loop() {

  // Connect or reconnect to WiFi
  if(WiFi.status() != WL_CONNECTED){
    Serial.print("Attempting to connect to SSID: ");
```

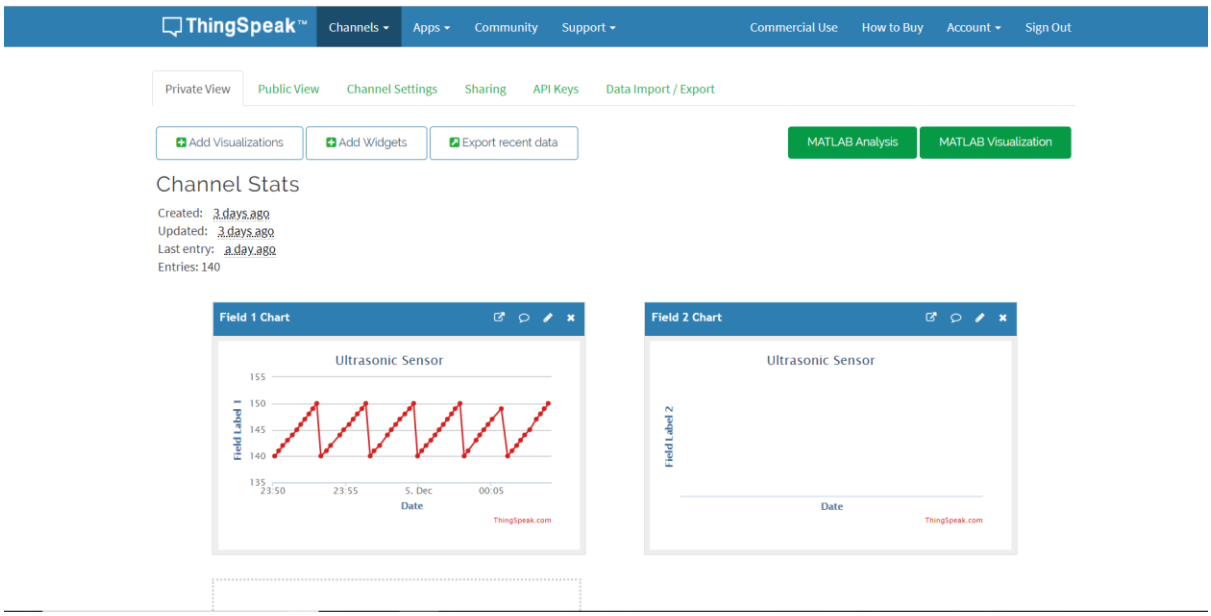
thingspeak | Arduino 1.8.7

File Edit Sketch Tools Help

thingspeak §

```
void loop() {

  // Connect or reconnect to WiFi
  if(WiFi.status() != WL_CONNECTED){
    Serial.print("Attempting to connect to SSID: ");
    Serial.println(ssid);
    while(WiFi.status() != WL_CONNECTED){
      WiFi.begin(ssid, pass);
      Serial.print(".");
      delay(5000);
    }
    Serial.println("\nConnected.");
  }

  digitalWrite(trigger,HIGH);
  delay(2000);
  digitalWrite(trig,LOW);
  int duration=pulseIn(echo,HIGH);
  int meters=(duration)/85.84

  // Write to ThingSpeak. There are up to 8 fields in a channel, allowing you to store up to 8 different
  // pieces of information in a channel.  Here, we write to field 1.
  int x = ThingSpeak.writeField(myChannelNumber, 1,meters, myWriteAPIKey);

  // Check the return code
  if(x == 200){
    Serial.println("Channel update successful.");
  }
  else{
    Serial.println("Problem updating channel. HTTP error code " + String(x));
  }


  delay(10000); // Wait 20 seconds before sending a new value
}
```

6.After uploading the code onto your node mcu you can see check your channel feed to get a graph plotted of the ultrasonic sensor's data.



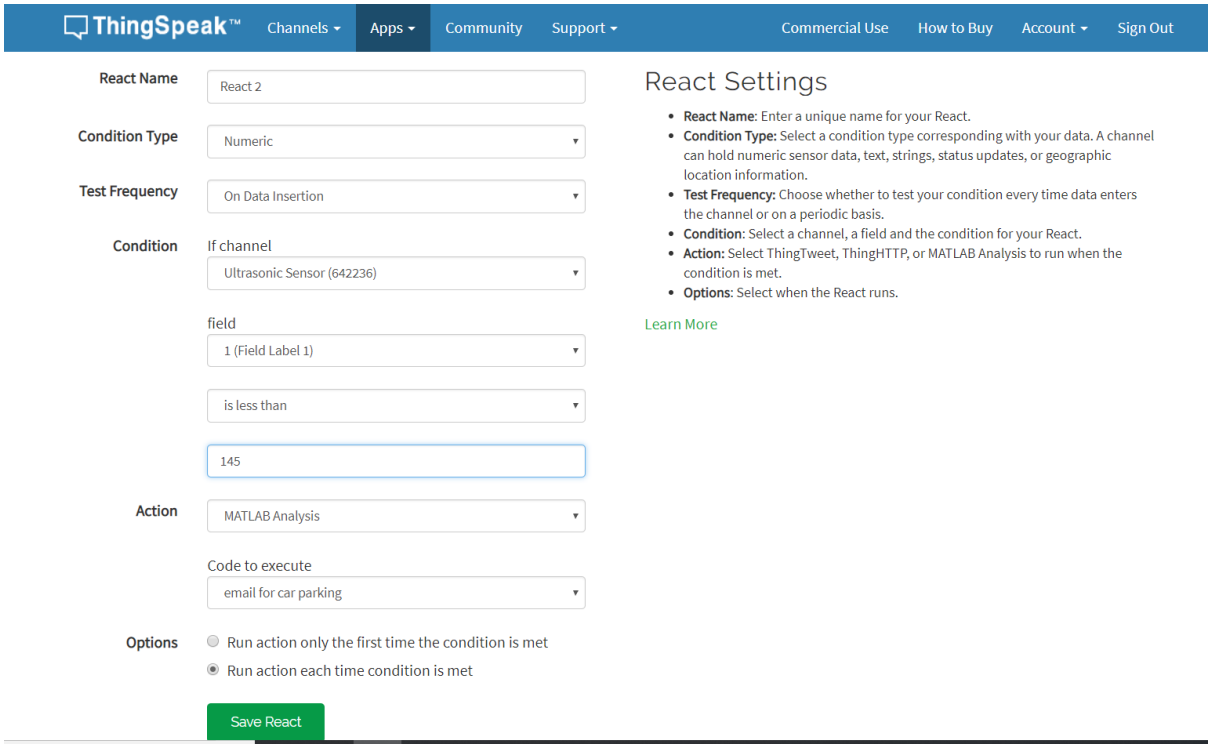
From the graph you can observe peak times at which a car is parked

6.Now to set a trigger when a particular condition is met, do the following steps on thingspeak.

i. Go to Apps->React

ii. We have to create a react such that every time our sensor's data falls below a certain level (say 145), a matalb code gets executed that will send us an email.

iii. Create the react as shown



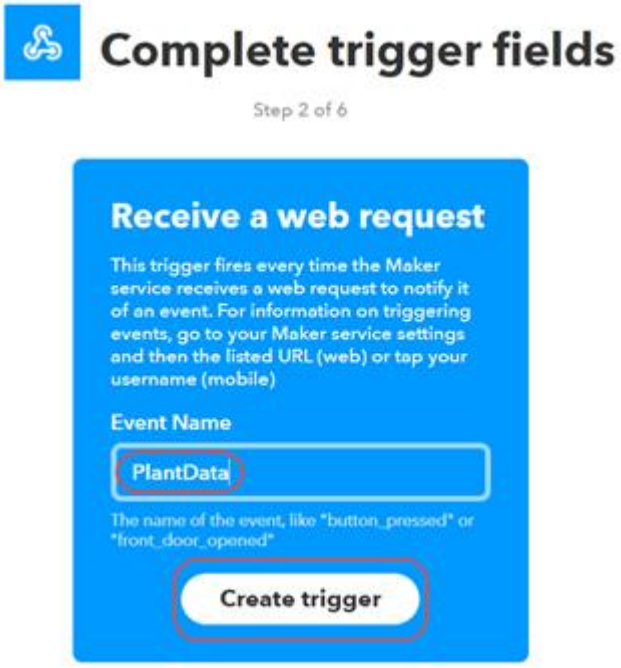
7. Now all that is left is to create an account on IFTTT, and create service that helps us send us email, whenever our matlab code calls it .

Creating an IFTTT service :

1. Create an IFTTT account, or log into your existing account.
2. Create an Applet. Select **My Applets**, and then click the **New Applet** button.
3. Select input action. Click the word **this**.

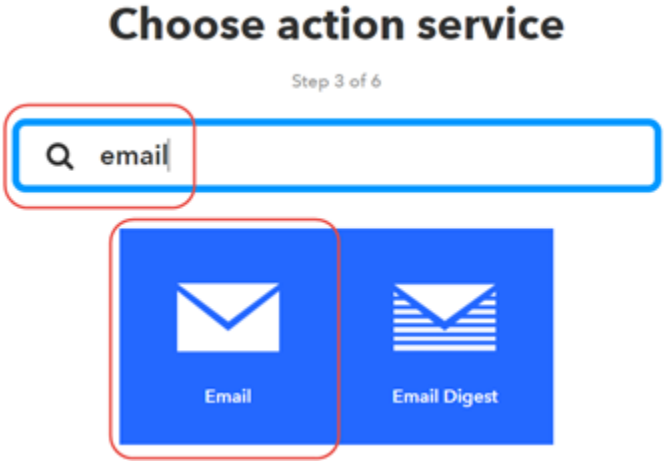
4. Select the Webhooks service. Enter Webhooks in the search field. Select the **Webhooks** card.
5. Complete the trigger fields. After you select Webhooks as the trigger, click the **Receive a web request** box to continue. Enter

an event name. This example uses PlantData as the event name. Click **Create Trigger**.

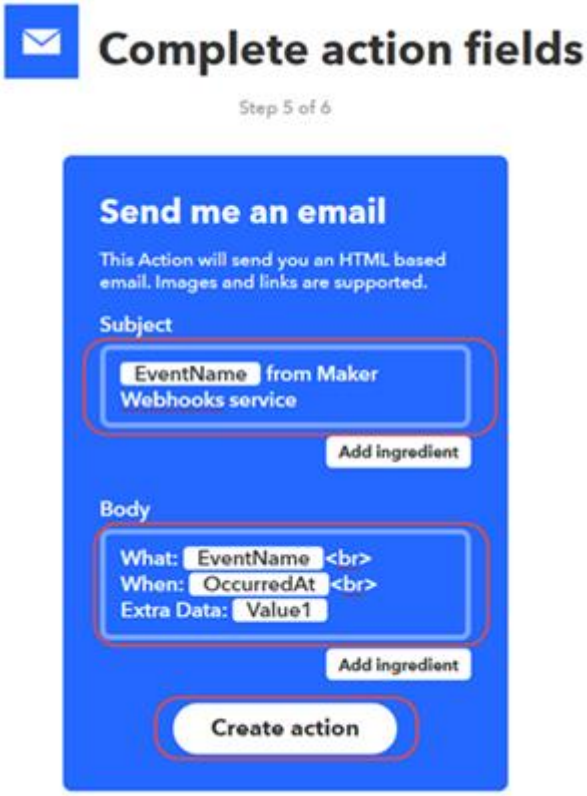


Now the trigger word **this** is the Webhooks icon.

6. Select the resulting action. Click the word **that**. Enter email in the search bar, and select the **Email** box.



7. Now enter the information that you want to be sent to you whenever parking lot gets occupied in the given fields.



8. Retrieve your Webhooks trigger information. Select **My Applets** > **Services**, and search for Webhooks.
   Select **Webhooks** and then the **Documentation** button. You see your key and the format for sending a request. Enter the event name. The event name for this example is Car_Parking
9. https://maker.ifttt.com/trigger/**{event}**/with/key/XXXXXXXXXXXXXXXXXXXXXX

Now your email service is setup and all you have to do is write the matlab code that connects to IFTTT and calls this service.

10.Go to Apps->Matlab Analysis and write the following code with your credentials.

ThingSpeak™ Channels Apps Community Support Con

Apps / MATLAB Analysis / email for car parking / Edit

Name

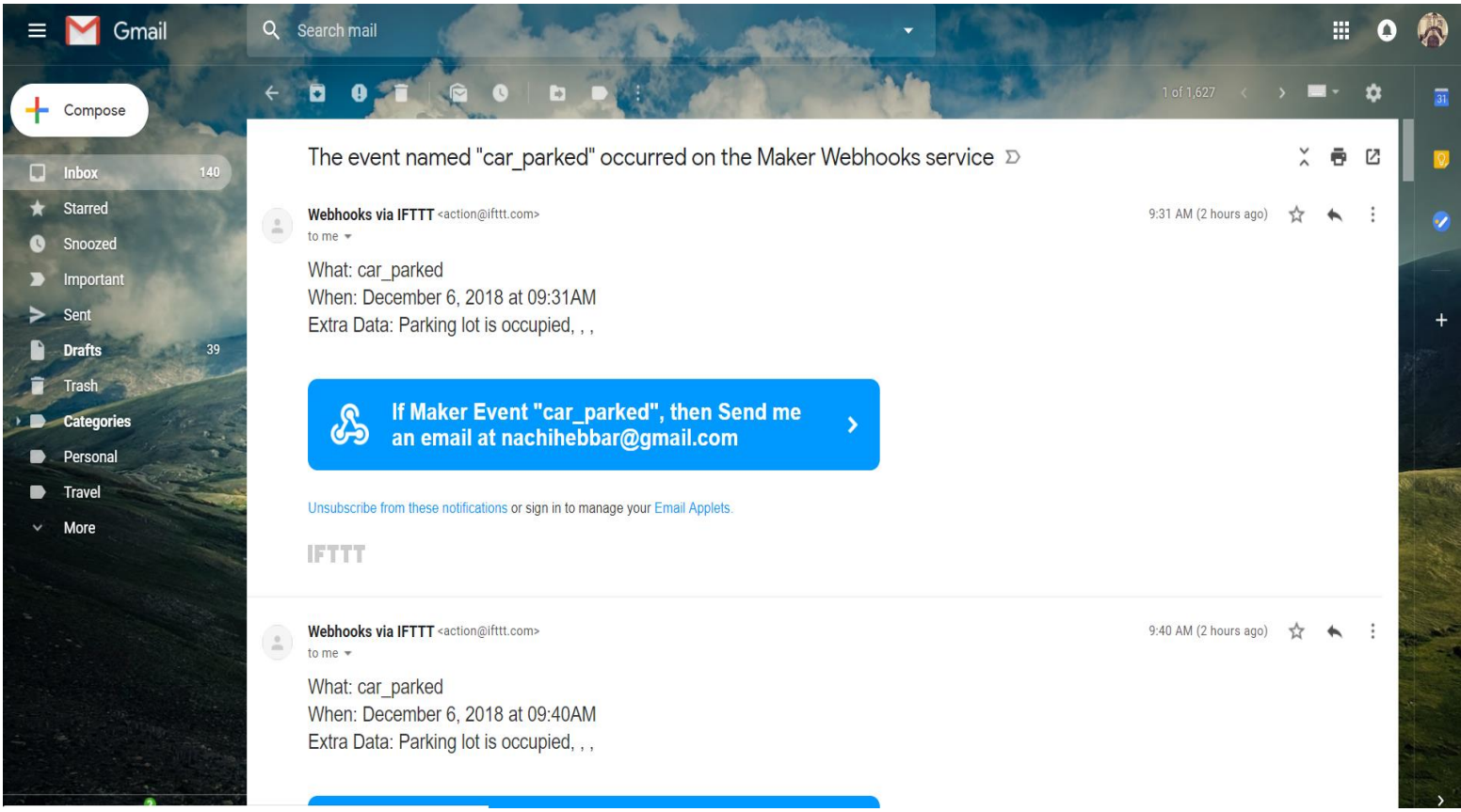
email for car parking

MATLAB Code

```
channelID=642236;
ifttURL="https://maker.ifttt.com/trigger/car_parked/with/key/c5JgAadbc9YM13xvOnXv57";

message_sent="Parking lot is occupied";

webwrite(ifttURL,'value1',message_sent);
```

Save and Run Save

11.Save and Run ! Everytime the parking slots get occupied you get a notification , using which you can perform various actionns like incrementing a number on a led display,etc . Also from the graph you can understand what is the peak time for car parking and park accordingly .


Gmail
Search mail
Compose
Inbox 140
Starred
Snoozed
Important
Sent
Drafts 39
Trash
Categories
Personal
Travel
More
1 of 1,627
The event named "car_parked" occurred on the Maker Webhooks service
Webhooks via IFTTT <action@ifttt.com>
to me
9:31 AM (2 hours ago)
What: car_parked
When: December 6, 2018 at 09:31AM
Extra Data: Parking lot is occupied, , ,
If Maker Event "car_parked", then Send me an email at nachihebbar@gmail.com
Unsubscribe from these notifications or sign in to manage your Email Applets.
IFTTT
Webhooks via IFTTT <action@ifttt.com>
to me
9:40 AM (2 hours ago)
What: car_parked
When: December 6, 2018 at 09:40AM
Extra Data: Parking lot is occupied, , ,