IOT : MINI PROJECT

Made some basic research about how to make a webpage
And researched about Firebase ,Html, java script

HTML : Hyper Text Markup Language *: It is used to create Web Pages .They include writing ,adding links to other web pages , adding pictures to webpage,we can add tables,forms,audio ,videos……*

HTML is a language used in front end of web Development, defines the structure and layout of a Web document by using a variety of tags and attributes.

ref:https://www.w3schools.com/html/

JavaScript :

Js is a high level language,which is mainly used for scripting the client side i.e; along with HTml. Sometimes it is also used in the backend web development.

It adds dynamic and responsive nature to the websites
reference:https://www.w3schools.com/js/
Youtube: Telusko Learnings Javascript

CSS :Cascading Style Sheets.

CSS *describes how HTML elements are to be displayed on screen, paper, or in other media.**CSS** saves a lot of work. It can control the layout of multiple web pages all at once.*

Ref: https://www.w3schools.com/css/default.asp

FireBase :

Firebase provides a realtime database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebases cloud.Fire Base is by Google

For Reference use : firebase documentation link : https://firebase.google.com/docs

NodeMCU: Esp8266 bootloader

Esp8266 is a WiFi module capable of Amazing features

It basically provides a server type space to share information or data via it .The chip esp8266 is heart for WiFi projects and NodeMCU is a bootloader used for coding on esp826.NodeMCU can run on Arduino IDE once proper Libraries are imported .NodeMcu also has a microcontroller in-built in it(There's a 32-bit RISC processor clocked at 80MHz, along with a generous RAM complement and support for up to 16mb of external flash storage for Esp8266 model)

https://www.ahirlabs.com/2017/10/21/what-is-nodemcu-esp8266/

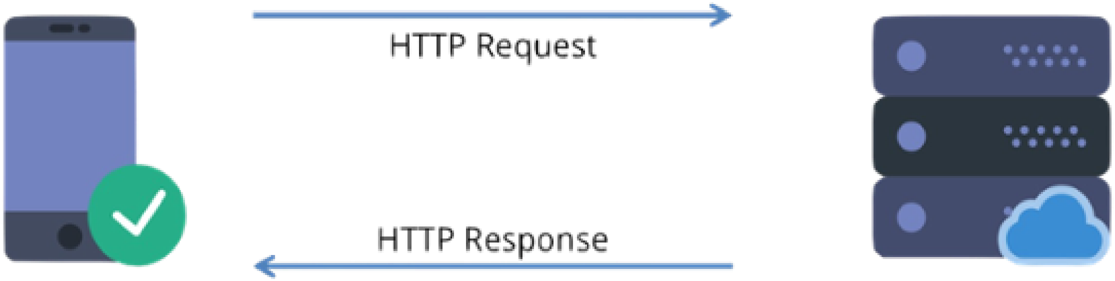
HTTP requests :

It is ***Hyper Text Transfer Protocol.***

When we look into any random webpage or website ,there will many links and buttons .As when we click them we get some work or some event done and all this is process doesn't happen just by a click,but in the background a type of communication process will be going between client and server.

Client : *Most of the it will be a bowser or an app.It makes HTTP requests to the server.*

Server :*It is basically a computer specially coded to play with the http requests and manage the requests between clients and database.*



The main requests of Http protocol are : 1)GET request
2)POST request
3)PUT request
4)DELETE request

GET request : *The client by using this type of request can call the server to read or get the data from the respective data database(request),then the server studies the request and send* ***response (****it's sent by the server to the client with the data what he requested for****).***

POST request : *By using this request method ,the client can update data into the database(ex:like we post some images or other stuff on social media).*

PUT request : *By using this request the client is able to make new directories into the database.*

DELETE request :*The name itself define its nature.This request is used to delete some data or directory.*

HTTP protocol :

Protocol defines how messages are formatted and transmitted .Its defines the rules and regulations of the way of communication we are using.
So, HTTP protocol defines how data is formatted and transmitted using this way of communication.

//to be continued//

FireBase :

When we consider this IOT project ,our main aim is to collect data from sensors and send them to cloud or database and making some actions with the data collected.
For storing this data we need a database.

Real-time data: *Real-time data (RTD) is information that is delivered immediately after collection. There is no delay in the timeliness of the information provided.*

Real-time Database :*It is the database which collect the real time data .It overrides the new data over the old data.*

This kind of database can be mainly used for Navigation,IOT projects …..

*NOTE: The data in the database is stored in the ''****JSON(****JavaScript Object Notation****)''*** *format. I.e; {''key":"value"}*

So,here the ***firebase*** *which provides the* ***real-time database.***

One of the best point regarding this firebase is that it manages its own server for our requirement(project).i.e; All the http requests are managed by the firebase itself.

Alternative can be used by using nodejs as our personal server and mongoDB as our database.

Steps involved to proceed in to project:
Step1*: A web page is made using html or javascript as per convenience.*

Step2*: A database (Firebase in our case) is selected for data storage .*
*Here , we have to make a Real-Time database in firebase, which provides some details about the database and we are going to include them in webgui and nodemcu in order to access the data from database. Detailed process how to include firebase into your webGui (To access this in arduino ,**FirebaseArduino** library to be installed.) @https://firebase.google.com/docs/web/setup.*

Step3*: A javascript code is wrote for behind page processing when API is used for data analysis.*

Step4*:NodeMCU is set up according to need and coded using arduino IDE.*
(Using nodeMcu play an important role. The coding for nodeMcu is done using Arduino IDE .For this to happen ,some libraries to be installed in the arduino ide.Ex: ArduinoJson,ESP8266).
Step5*:Activate the HOTSPOT and POWER NodeMCU and see the serial monitor.*

TEST 1:

Simple led on/off:

1.A web page was made showing LED ON and LED OFF and it was interfaced or linked with FIREBASE software .
2.When LED ON is clicked the FIREBASE have led_status as 1(led_status is a json object which we used in database.)
3.The nodemcu is coded in such a way that when it reads 1 in led_status if provides logic one to D7 pin and led is turned on .

Problems faced:

1.If the arduino code shows error for unable to execute javascript ,then library <ArduinoJson.h> library must be included from https://github.com/bblanchon/ArduinoJson. (Always better to use an updated version of libraries)
Add this to c:/program(x86)/Arduino/library for Json string error i.e:add the library to libraries folder of arduino.(This library can also be downloaded directly from arduino ide)

2.The led might lag and won't respond to on and off if done repeatedly due to overriding of data before it gets executed. Stable internet connection can solve the problems.

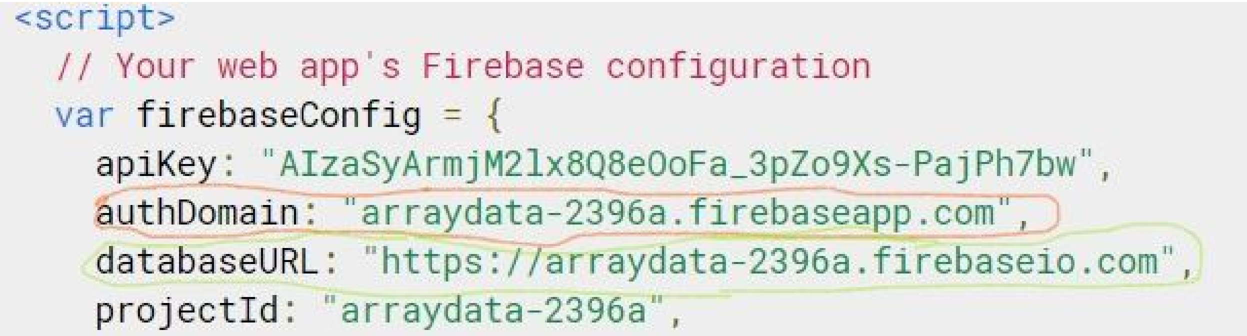
TEST 2

Multiple led controlling :

Only change in Code can help for controlling multiple led, by making an array of led_status and changing the circuitry multiple led can be controlling.

<u>Problems faced:</u>

1. *Data overriding .It can be solved by reconstructing javascript with suitable options.(It will be cleared by just updating the value of object we need ,instead of all the objects of database).*
2. *FIREBASE not connecting to NodeMCU.*
   Actually the problem is not with the code.The problem is with the link of host we used .(we used a wrong link of the database).



```
<script>
  // Your web app's Firebase configuration
  var firebaseConfig = {
    apiKey: "AIzaSyArmjM2lx8Q8eOoFa_3pZo9Xs-PajPh7bw",
    authDomain: "arraydata-2396a.firebaseapp.com",
    databaseURL: "https://arraydata-2396a.firebaseio.com",
    projectId: "arraydata-2396a",
```

Here instead of databaseURL ,we used authDomain (as they were looked similar)

TEST 3

Parking slot system

Once the leds were working fine ,the next aim was to build a parking system which could show empty and non empty slots available
The changes to led codes were

1. *IR sensor were added to NodeMCU*
2. *Firebase.SetInt command was used to send data to firebase which was automatically updated in web or application*
3. *An app was created to manage instead of web gui.*

Once the Nodemcu was connecting finely with internet and Firebase next we went on designing a circuit which could turn on and off a 230V AC bulb and take input from IR sensors simultaneously.

HARDWARE:

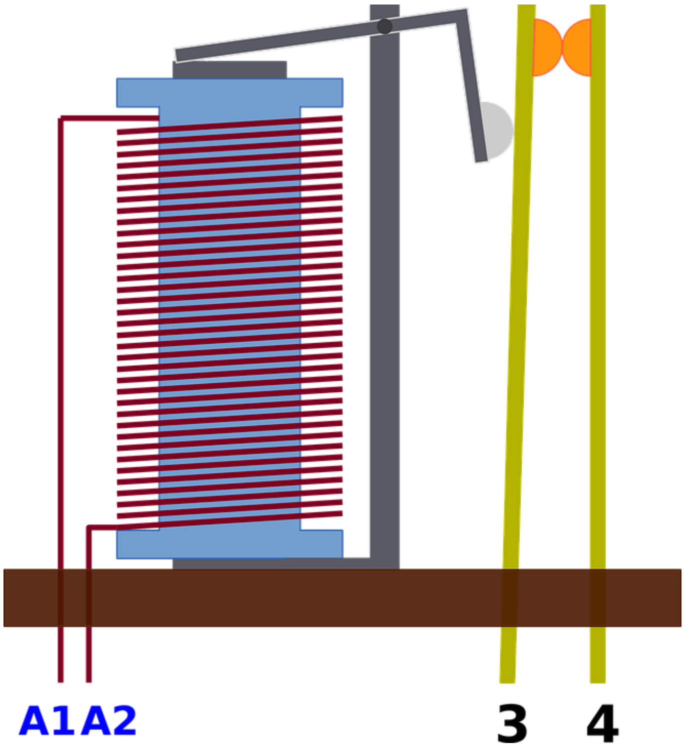
The circuit consisted of NodeMCU connect via headers and powered from Vin pin with a 9V [Two batteries in parallel (Reason discussed later)]

FOR IR SENSORS(3 used) :

To power IR sensors the 9V was directly connected to LM7805 and three pins (pairs of gnd and vcc) were given 5.2V by connecting them in parallel.

FOR CONTROLLING AC APPLIANCES (230V bulb in our case):

Relay was used for satisfying the need to control the AC devices. Relay is small box containing an electromagnetic coil which could attract the 3 terminals provided in other Side of relay.



As shown here when A1 and A2 are energised the contact 3 and 4 gets connect due to magnetic fields and AC current can flow through it .
A normal relay requires 3.3/5/6/12V DC INPUT and current in range of 40mA to 100mA.

Problem: NodeMCU can support only 3.3V and max 12 mA current .
So to drive this relay Transistor in CE configuration was used for power amplification.

We used

Transistor	*: BC337 (Any NPN would probably work with gain around 200-250)*
Base resistor	*: 11k ohms*
Collector resistor	*: 33 ohms*
Relay	*: 5V 45-60mA*

Use this formula to calculate for other formulas :

Collector resistor : (Relay voltage - 0.7)/Relay current [sometimes if the transistor still doesn’t work,take 1 instead of 0.7]
Relay current : Resistance of relay coil / Relay voltage
Base resistor : Pin voltage (nodemcu or arduino depends)-0.7/Base current
Base current : Collector current / Beta

Thus after driving relay from nodeMCU and glowing BULB we are able to control any appliances working on 230 V AC.

Now it was time to control the project with app.
We used MIT app inventor to develop the apps such as for the ***parking system.***

What is an app inventor?
App Inventor *is a cloud-based tool, which means you can build apps right in your web browser. It uses a graphical user interface (GUI) very similar to the programming languages Scratch and the StarLogo TNG user interface, which allows users to drag and drop visual objects to create an application that can run on mobile devices. It allows you to develop apps without getting into the complexity of back-end development.*

We interfaced this app with firebase server to display the availability of the parking slots in our parking slot iot system.

.