|  |
| --- |
| **[ Internet of things ]** |
|  |

Practical-10

**-:AIM:-**

**Raspberry pi basic Programming with Ultrasonic Sensor.**

Submitted By: Dharmay Sureja

Enrollment No:17012011056



**GANPAT UNIVERSITY**

**U. V. Patel College of Engineering**

**Computer Engineering Department**

**AIM:- Raspbian pi basic programming with Ultrasonic Sensors.**

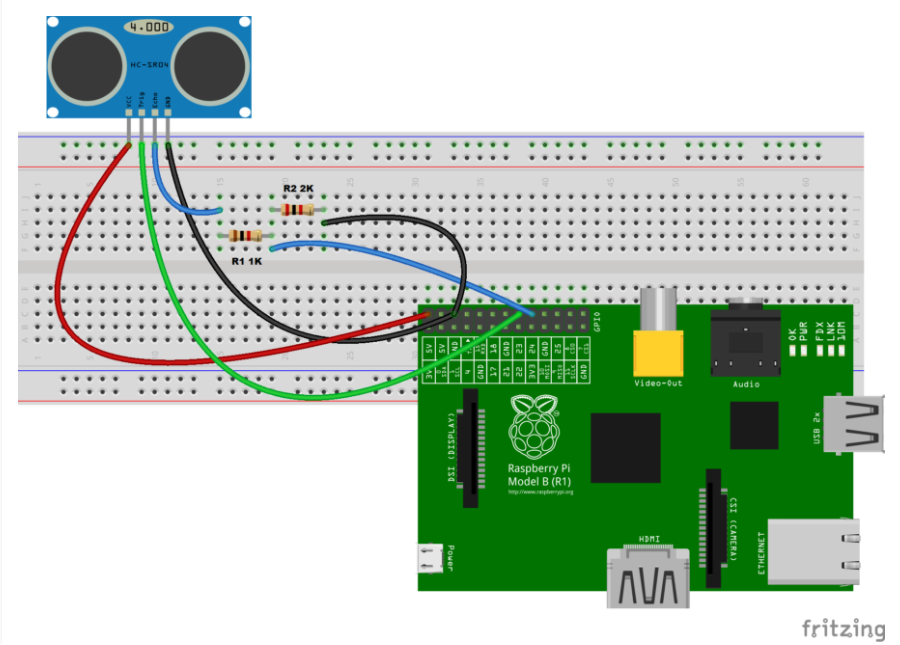
**Experiments**

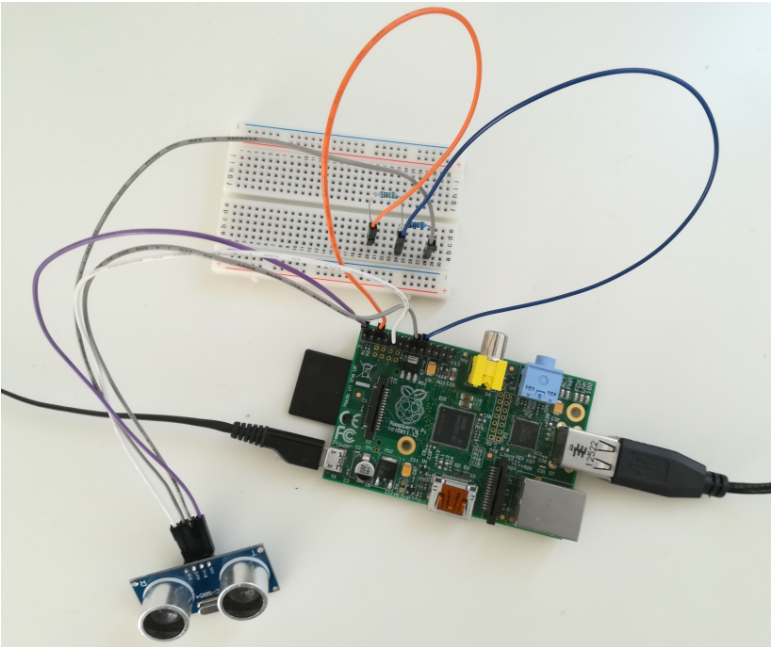
1. **Measure the distance of an obstacle using ultrasonic sensor.**

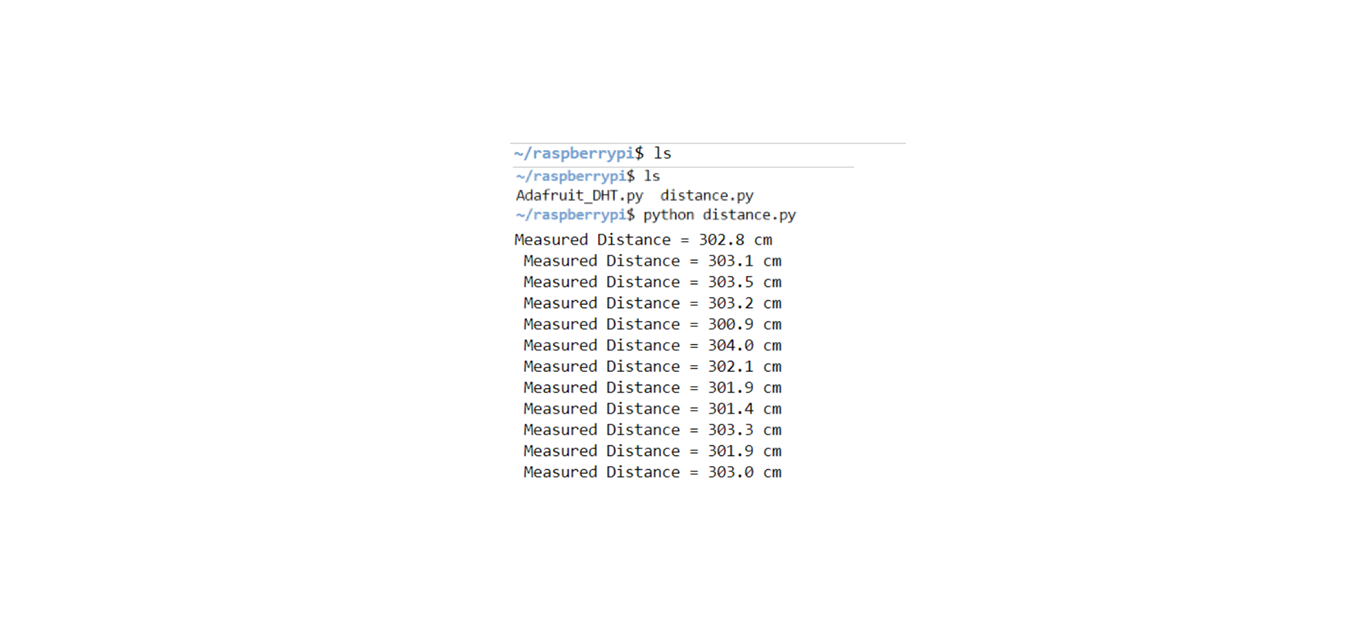
**Components used :** Breadboard , 2.2K Ohm Resistor, 1K Ohm Resistor , Raspberry pi ,

SD Card & Adapter , Jumper Wire , Ultrasonic Sensor , USB Cable

**Circuit**:





****

**Code** : distance.py

import RPi.GPIO as GPIO

import time

GPIO.setmode(GPIO.BCM)

GPIO\_TRIGGER = 23

GPIO\_ECHO = 24

GPIO.setup(GPIO\_TRIGGER, GPIO.OUT)

GPIO.setup(GPIO\_ECHO, GPIO.IN)

def distance():

GPIO.output(GPIO\_TRIGGER, True)

# set Trigger after 0.01ms to LOW

time.sleep(0.00001)

GPIO.output(GPIO\_TRIGGER, False)

StartTime = time.time()

StopTime = time.time()

while GPIO.input(GPIO\_ECHO) == 0:

StartTime = time.time()

# save time of arrival

while GPIO.input(GPIO\_ECHO) == 1:

StopTime = time.time()

# time difference between start and arrival

TimeElapsed = StopTime - StartTime

# multiply with the sonic speed (34300 cm/s)

# and divide by 2, because there and back

distance = (TimeElapsed \* 34300) / 2

return distance

if \_\_name\_\_ == '\_\_main\_\_':

try:

while True:

dist = distance()

print ("Measured Distance = %.1f cm" % dist)

time.sleep(1)

# Reset by pressing CTRL + C

except KeyboardInterrupt:

print("Measurement stopped by User")

GPIO.cleanup()

1. **Send SMS if Object distance is less then 10 cm ( using Twillo API )**

**Code : send\_sms.py**

**import RPi.GPIO as GPIO**

**import time**

**from twillo.rest import Client**

**account\_sid = "AC92b0aeee716d654f31561ac68bec0f12"**

**auth\_token = "95aa6dee3decb6a6c9fb224e4d7c4916"**

**GPIO.setmode(GPIO.BCM)**

**GPIO\_TRIGGER = 23**

**GPIO\_ECHO = 24**

**GPIO.setup(GPIO\_TRIGGER, GPIO.OUT)**

**GPIO.setup(GPIO\_ECHO, GPIO.IN)**

**def distance():**

**GPIO.output(GPIO\_TRIGGER, True)**

**# set Trigger after 0.01ms to LOW**

**time.sleep(0.00001)**

**GPIO.output(GPIO\_TRIGGER, False)**

**StartTime = time.time()**

**StopTime = time.time()**

**while GPIO.input(GPIO\_ECHO) == 0:**

**StartTime = time.time()**

**# save time of arrival**

**while GPIO.input(GPIO\_ECHO) == 1:**

**StopTime = time.time()**

**# time difference between start and arrival**

**TimeElapsed = StopTime - StartTime**

**# multiply with the sonic speed (34300 cm/s)**

**# and divide by 2, because there and back**

**distance = (TimeElapsed \* 34300) / 2**

**return distance**

**if \_\_name\_\_ == '\_\_main\_\_':**

**try:**

**while True:**

**dist = distance()**

**print ("Measured Distance = %.1f cm" % dist)**

**time.sleep(1)**

**if dist < 10:**

**client = Client(account\_sid,auth\_token)**

**message = client.message.create(**

**to = "+917845612390"**

**from = "575701"**

**body = "Hello from GNU’s Rasp\_PI")**

**print(message.sid)**

**quit(0)**

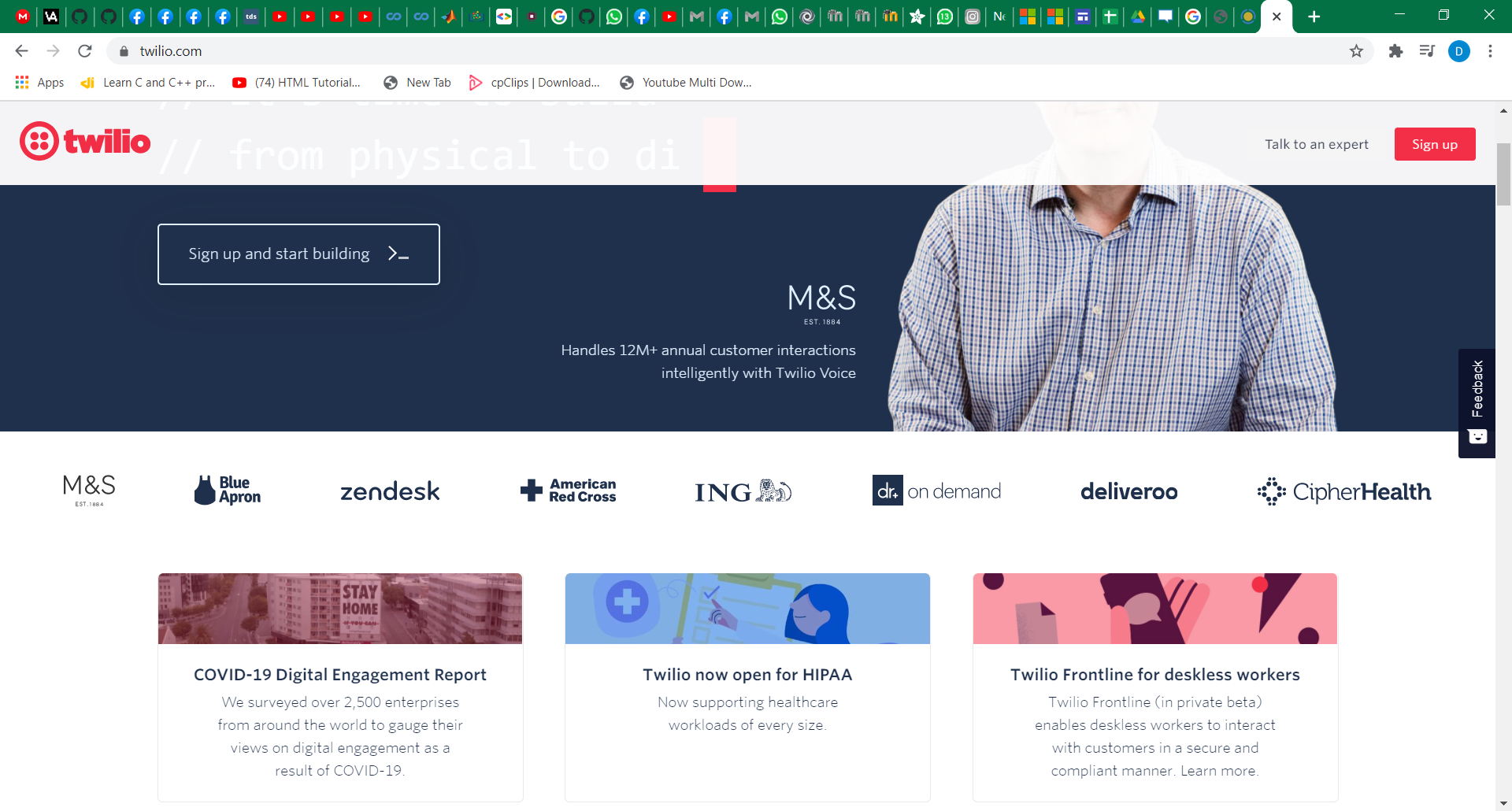
**# Reset by pressing CTRL + C**

**except KeyboardInterrupt:**

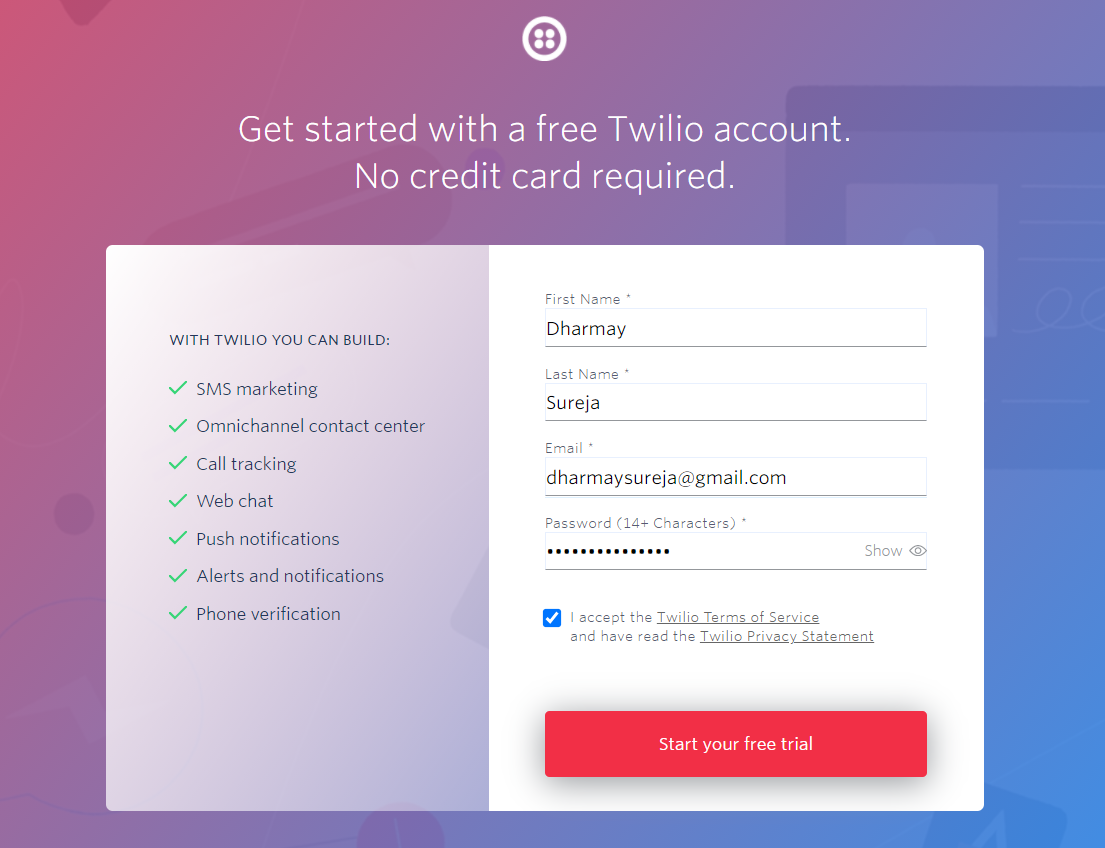
**print("Measurement stopped by User")**

**GPIO.cleanup()**

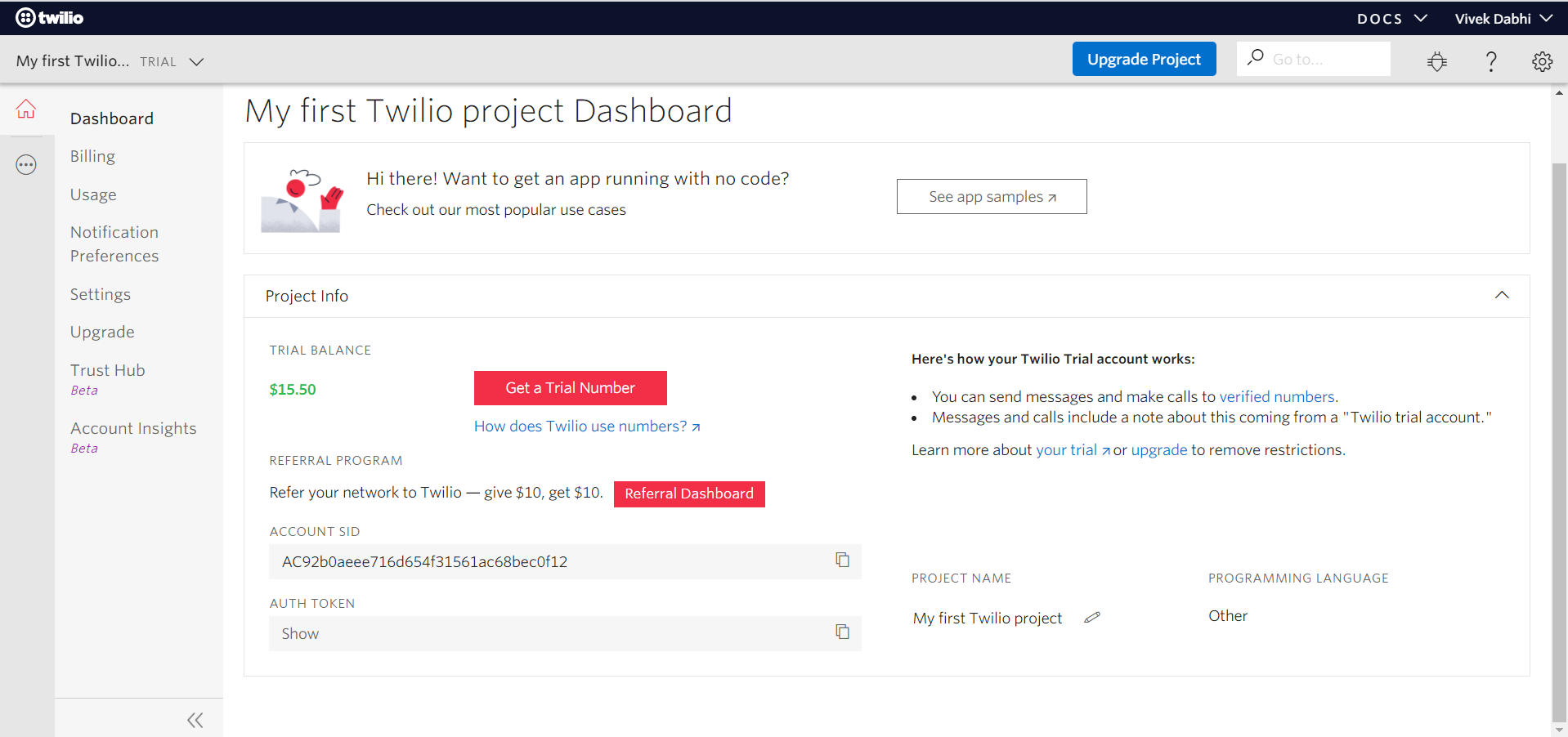
1. Open [www.twillo.com](file:///E:\Sem%20%237\IoT\Pr10\www.twillo.com)



1. Register with your mail and mobile Number .



1. Login [www.twillo.com](http://www.twillo.com) with your email address
2. Note the Authentication ID and Authentication Token.



1. Get twillo phone number (In order to make calls or send Messages through thw twillo

API , you need to get a Twillo Phone Number .

