

# Estación Meteorológica

Ernesto chung, Pedro Romero

*Universidad Latina de Panamá; Ingeniería en Mecatrónica*

[ernesto2080@hotmail.com](mailto:ernesto2080@hotmail.com); [pedroromero96@gmail.com](mailto:pedroromero96@gmail.com)

**Abstract : this meteorological station was created with the purpose of being able to experience by itself how the devices that capture the information work in order to reach the personnel**

**Key Words: SenseHAT, MIME**

**Resumen: esta estacion meteorologica se creo con el fin de poder uno experimentar por si mismo como funciona los dispositivos que captan la información para asi hacerle llegar al personal**

**Palabras claves: SenseHAT, MIME**

## I. Introducción

Una Raspberry Pi y un add on llamado SenseHat para determinar las principales variables meteorológicas de humedad, presión y temperatura. Luego el Raspberry Pi tomará dichos datos, los compilará a intervalos y enviará un correo electrónico con estos datos luego de un tiempo determinado. Además, se utilizará el SenseHat para mostrar la temperatura cada 5 segundos en su display de LEDs 8x8.

### A. Justificacion

Usaremos la raspberry Pi para tener nuestra propia estación meteorológica y asi uno poder estar atentos a cualquier cambio meteorologico en su hogar o lugar de trabajo

### B. Objetivo General

Crear un programa que utilice el Raspberry Pi y el SenseHAT para lograr mantener una observación constante sobre las principales variables ambientales en un lugar específico.

## II. ESTADO DEL ARTE

El proyecto consiste en la utilización de un Raspberry Pi para mantener una observación constante sobre las principales variables climáticas en un

ambiente específico. Estas variables son temperatura, presión y humedad.

El Raspberry Pi no solo se mantiene leyendo la temperatura de manera constante, sino que luego de un tiempo determinado (30 minutos como predeterminado) se envían registros de ese periodo (se toman 6 registros divididos en tiempos iguales) al correo electrónico determinado en el programa.

Aparte de esto, cada 5 segundos se muestra la temperatura actual en el display 8x8 del SenseHat.

A. razones para querer monitorizar las condiciones atmosféricas de un entorno. Para el cultivo de ciertos vegetales en invernaderos resulta imprescindible el estudio y análisis de temperatura, humedad, luminosidad etc. Los animales en terrarios o acuarios deben cumplir con ciertas condiciones atmosféricas para garantizar la supervivencia de estos.

## III. Hardware

Comprar el sense hat ya sea por medio de amazon o en las tiendas de electronicas con esta especificaciones técnicas

Gyroscope – angular rate sensor:  
±245/500/2000dps

• Accelerometer - Linear acceleration sensor:  
±2/4/8/16 g

• Magnetometer - Magnetic Sensor: ±4/8/12/16 gauss

• Barometer: 260 – 1260 hPa absolute range (accuracy depends on the temperature and pressure, ±0.1 hPa under normal conditions)

- Temperature sensor (Temperature accurate to  $\pm 2^{\circ}\text{C}$  in the  $0-65^{\circ}\text{C}$  range)
- Relative Humidity sensor (accurate to  $\pm 4.5\%$  in the  $20-80\%rH$  range, accurate to  $\pm 0.5^{\circ}\text{C}$  in  $15-40^{\circ}\text{C}$  range)
- 8x8 LED matrix display
- Small 5 button joystick



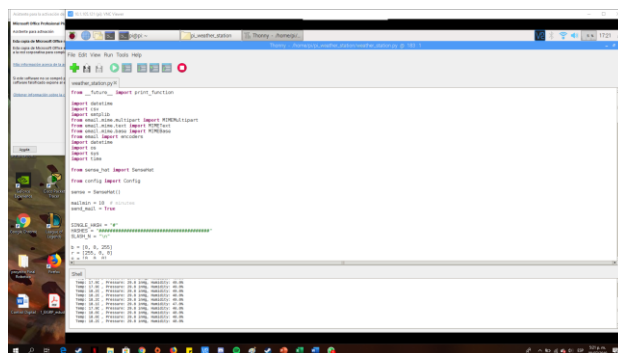
montaje es realmente fácil. Simplemente monte el Sense HAT en la parte superior de la Raspberry Pi, póngalo todo en su estuche y enciéndalo.



Conectar todos los componentes de la pantalla, mouse y teclado antes de conectar la fuente de poder



Al terminar de conectar todos los componentes ya con el sense hat encima de la raspberry pi corremos el código

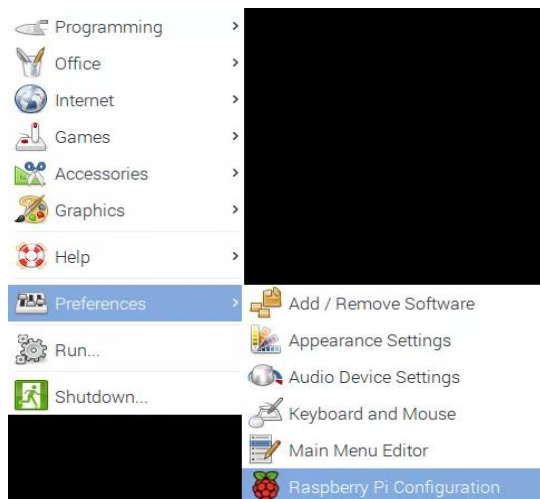


Ahí en donde empezaremos a sensor la temperatura del ambiente

Sense Hat: un periférico oficial para la Raspberry Pi y que gracias a sus múltiples sensores y a tener un pequeño joystick y un display led 8x8

#### IV. Software

El Raspberry PI necesita un sistema operativo para arrancar, así que tome su tarjeta microSD, descárguelo <https://www.raspberrypi.org/documentation/installation/installing-images/README.md>  
Abra el menú Raspberry en la esquina superior izquierda de la pantalla Pi, luego seleccione Preferencias → Configuración de Raspberry Pi



deberá actualizar el software principal de Pi. Abra una ventana de terminal y ejecute el comando:

```
sudo apt-get update
```

Este comando actualiza los índices de Pi de los paquetes de software disponibles. A continuación, ejecute:

```
sudo apt-get upgrade
```

Este comando recupera e instala las últimas y mejores versiones del sistema operativo Raspbian y otros paquetes de software instalados en el Pi.

#### Datetime:

[1]módulo proporciona clases para manipular fechas y horas de manera simple y compleja. Si bien se admite la aritmética de fecha y hora, el enfoque de la implementación está en la extracción eficiente de atributos para el formato y la manipulación de salida. Para la funcionalidad relacionada

Atributos: hour, minute, second, microsecond, y tzinfo.

#### [2] Import-Csvcmdlet

crea objetos personalizados con forma de tabla a partir de los elementos de los archivos CSV. Cada columna en el archivo CSV se convierte en una propiedad del objeto personalizado y los elementos en filas se convierten en los valores de propiedad. Import-Csv funciona en cualquier archivo CSV, incluidos los archivos generados por el Export-Csv cmdlet.

#### [3]smtplib

módulo incorporado para enviar correos electrónicos utilizando el Protocolo simple de transferencia de correo (SMTP). smtplib usa el protocolo RFC 821 para SMTP.

#### [4] Os

nos permite acceder a funcionalidades dependientes del Sistema Operativo. Sobre todo, aquellas que nos refieren información sobre el entorno del mismo y nos permiten manipular la estructura de directorios (para leer y escribir archivos)

Descripción	Método
Saber si se puede acceder a un archivo o directorio	os.access(path, modo_de_acceso)
Conocer el directorio actual	os.getcwd()
Cambiar de directorio de trabajo	os.chdir(nuevo_path)
Cambiar al directorio de trabajo raíz	os.chroot()
Cambiar los permisos de un archivo o directorio	os.chmod(path, permisos)
Cambiar el propietario de un archivo o directorio	os.chown(path, permisos)
Crear un directorio	os.mkdir(path[, modo])
Crear directorios recursivamente	os.makedirs(path[, modo])
Eliminar un archivo	os.remove(path)
Eliminar un directorio	os.rmdir(path)
Eliminar directorios recursivamente	os.removedirs(path)
Renombrar un archivo	os.rename(actual, nuevo)
Crear un enlace simbólico	os.symlink(path, nombre_destino)

#### [5]sys

se importa como

```
import sys
```

El módulo provee acceso a funciones y objetos mantenidos por el intérprete.

#### [6]Time

para manejar tareas relacionadas con el tiempo. Para usar las funciones definidas en el módulo, primero debemos importar el módulo.

Se importa como `import time`

encender uno o varios pixeles de diferentes colores, o incluso representar una figura en 8×8, todo ello podemos hacerlo utilizando la función `sense.set_pixel()`, a continuación

```
from sense_hat import SenseHat
sense = SenseHat()

sense.set_pixel(3, 4, [255, 0, 0])
```

Al ejecutar el script se encenderá un pixel de color rojo aproximadamente en el centro de la matriz led, concretamente en la columna 4 y en la fila 5. La diferencia entre los parámetros informados y el número de columna y fila se debe a que `sense.setpixel` numera los leds de 0 a 7, de izquierda a derecha y de arriba a abajo. Si lo que queremos es representar una figura, una forma más eficiente que nos ahorrará la introducción de muchísimo código es la siguiente:

```
n = [0, 0, 0, 0]
r = [255, 0, 0]
a = [0, 255, 0]
v = [0, 0, 255]
```

Asignar una matriz de valores a otra variable

```
matriz = [
n,n,n,n,n,n,n,n,
n,r,r,r,r,r,n,n,
r,r,v,r,v,r,r,n,
r,r,r,r,r,r,n,n,
n,r,r,r,r,r,n,n,
n,n,r,n,r,n,n,n,
n,r,n,n,n,r,n,n,
n,n,n,n,n,n,n,n,
]
```

`get_cpu_temp()`

puede que los valores del Sense HAT parecen un poco altas, eso es porque lo son. El culpable es el calor generado por la CPU del Pi que calienta el aire alrededor del Sense HAT cuando está sentado encima del Pi.

Para esto es dicho comando para que tome el valor de la temperatura del cpu y que me reemplace la temperaturas como se muestra a continuación

```
def get_cpu_temp():
    res = os.popen('vcgencmd measure_temp').readline()
    return float(res.replace("temp=", "").replace("C\n", ""))
```

## I. CODIGO

Importaciones de librerías necesarias para el funcionamiento del código

`from __future__ import print_function`

`import datetime`

`import csv`

`import smtplib`

`from email.mime.multipart import MIMEMultipart`

`from email.mime.text import MIMEText`

`from email.mime.base import MIMEBase`

`from email import encoders`

`import datetime`

`import os`

`import sys`

`import time`

del sensor se importan los valores que haya tomado el sense hat y lo envia al correo en 30 minutos

`from sense_hat import SenseHat`

`from config import Config`

`sense = SenseHat()`

`mailmin = 30 minutos`

`send_mail = True`

`SINGLE_HASH = "#"`

## HASHES

```
#####  
###"
```

```
SLASH_N = "\n"
```

Matriz para los colores de los leds

```
b = [0, 0, 255]
```

```
r = [255, 0, 0]
```

```
e = [0, 0, 0]
```

```
empty = [
```

```
e, e, e, e, e, e, e, e,
```

```
e, e, e, e, e, e, e, e,
```

```
e, e, e, e, e, e, e, e,
```

```
e, e, e, e, e, e, e, e,
```

```
e, e, e, e, e, e, e, e,
```

```
e, e, e, e, e, e, e, e,
```

```
e, e, e, e, e, e, e, e,
```

```
e, e, e, e, e, e, e, e
```

```
]
```

Toma el valor de la temperatura del cpu y la resta de lo que haya sentido el sense Hat para que muestre el valor verdadero ya que el cpu le afecta a la temperatura de sensor

```
def get_cpu_temp():
```

```
res = os.popen('vcgencmd  
measure_temp').readline()
```

```
return float(res.replace("temp=",  
"").replace("'C\n", ""))
```

```
def get_smooth(x):
```

```
if not hasattr(get_smooth, "t"):
```

```
get_smooth.t = [x, x, x]
```

```
get_smooth.t[2] = get_smooth.t[1]
```

```
get_smooth.t[1] = get_smooth.t[0]
```

```
get_smooth.t[0] = x
```

```
xs = (get_smooth.t[0] + get_smooth.t[1] +  
get_smooth.t[2]) / 3
```

```
return xs
```

```
def get_temp():
```

```
t1 = sense.get_temperature_from_humidity()
```

```
t2 = sense.get_temperature_from_pressure()
```

```
t = (t1 + t2) / 2
```

```
t_cpu = get_cpu_temp()
```

```
t_corr = t - ((t_cpu - t) / 1.5)
```

```
t_corr = get_smooth(t_corr)
```

```
return t_corr
```

```
def main():
```

```
global last_temp
```

```
mailcount = 0
```

```
while 1:
```

```
current_second =  
datetime.datetime.now().second
```

```
if (current_second == 0) or ((current_second %  
5) == 0):
```

```
calc_temp = get_temp()
```

```
temp_c = round(calc_temp, 1)
```

```
hum = round(sense.get_humidity(), 0)
```

```
pres = round(sense.get_pressure() * 0.0295300,  
1)
```

Imprime los valores y los muestra en la parte posterior del programa en los mensajes

```
print("Temp: %sC , Pressure: %s inHg,
Humidity: %s%%" % (temp_c, pres, hum))
```

```
sense.show_message("%sC" % (temp_c),
text_colour=[0, 0, 255], back_colour=[0, 0, 0])
```

```
sense.set_pixels(empty)
```

```
mailcount += 1
```

```
if mailcount == mailmin*12:
```

```
print ("%s mail" % (mailcount))
```

```
with open('att.csv','w') as csvfile:
```

```
filewriter = csv.writer (csvfile, delimiter=',',
```

```
quotechar='|',
quoting=csv.QUOTE_MINIMAL)
```

```
filewriter.writerow(['Time','Temperature','Press
ure','Humidity'])
```

```
filewriter.writerow(['5','%s'%(temp_5),'%s'%(pr
es_5),'%s'%(hum_5)])
```

```
filewriter.writerow(['10','%s'%(temp_10),'%s'%(
pres_10),'%s'%(hum_10)])
```

```
filewriter.writerow(['15','%s'%(temp_15),'%s'%(
pres_15),'%s'%(hum_15)])
```

```
filewriter.writerow(['20','%s'%(temp_20),'%s'%(
pres_20),'%s'%(hum_20)])
```

```
filewriter.writerow(['25','%s'%(temp_25),'%s'%(
pres_25),'%s'%(hum_25)])
```

```
filewriter.writerow(['30','%s'%(temp_30),'%s'%(
pres_30),'%s'%(hum_30)])
```

envia los datos obtenidos al correo

```
mailcount = 0
```

```
toaddr = "ernesto2080@hotmail.com"
```

```
msg = MIMEMultipart()
```

```
msg['To'] = toaddr
```

```
date=datetime.datetime.now()
```

```
msg['Subject'] = "Temperatura de %s"%(date)
```

```
body = "Incluido archivo CSV"
```

```
msg.attach(MIMEText(body, 'plain'))
```

```
filename = "att.csv"
```

```
attachment = open("att.csv", "rb")
```

```
p = MIMEBase('application', 'octet-stream')
```

```
p.set_payload((attachment).read())
```

```
encoders.encode_base64(p)
```

```
p.add_header('Content-Disposition',
"attachment; filename= %s" % filename)
```

```
msg.attach(p)
```

```
s = smtplib.SMTP('smtp.gmail.com', 587)
```

```
s.starttls()
```

```
s.login(fromaddr, "ciruspass")
```

# Converts the Multipart msg into a string

```
text = msg.as_string()
```

```
s.sendmail(fromaddr, toaddr, text)
```

```
if mailcount == mailmin*2:
```

```
temp_5=temp_c
```

```
pres_5=pres
```

```
hum_5=hum
```

```
if mailcount == mailmin*4:
```

```
temp_10=temp_c
```



```

pres_10=pres

hum_10=hum

if mailcount == mailmin*6:

temp_15=temp_c

pres_15=pres

hum_15=hum

if mailcount == mailmin*8:

temp_20=temp_c

pres_20=pres

hum_20=hum

if mailcount == mailmin*10:

temp_25=temp_c

pres_25=pres

hum_25=hum

if mailcount == mailmin*12:

temp_30=temp_c

pres_30=pres

hum_30=hum

time.sleep(1)

print("Leaving main()")

if __name__ == "__main__":

try:

main()

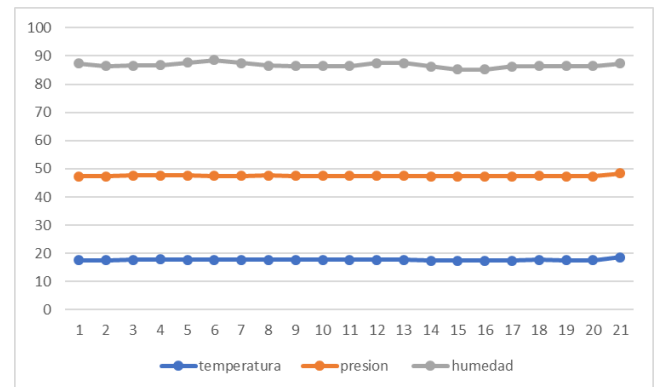
except KeyboardInterrupt:

print("\nExiting application\n")

```

## V. TABLA Y GRAFICA

Temp: 17.6C ,	Pressure: 29.8 inHg,	Humidity: 40.0%
Temp: 17.6C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.8C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.9C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.8C ,	Pressure: 29.8 inHg,	Humidity: 41.0%
Temp: 17.7C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.7C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.8C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.7C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.8C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.8C ,	Pressure: 29.8 inHg,	Humidity: 40.0%
Temp: 17.8C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.8C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.7C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.6C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.6C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.7C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.5C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.5C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.5C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.6C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.7C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.7C ,	Pressure: 29.8 inHg,	Humidity: 40.0%
Temp: 17.7C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.7C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.6C ,	Pressure: 29.8 inHg,	Humidity: 40.0%
Temp: 17.6C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.5C ,	Pressure: 29.8 inHg,	Humidity: 40.0%
Temp: 17.2C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.0C ,	Pressure: 29.8 inHg,	Humidity: 38.0%
Temp: 17.1C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.3C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.4C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.5C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.6C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.6C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.5C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.7C ,	Pressure: 29.8 inHg,	Humidity: 39.0%
Temp: 17.8C ,	Pressure: 29.8 inHg,	Humidity: 39.0%



## VI. ECUACIONES

[10] En una máquina térmica cualquiera, el trabajo que esta realiza corresponde a la diferencia entre el calor que se le suministra y el calor que sale de ella. Por lo tanto, la eficiencia es el trabajo que realiza la máquina dividido entre el calor que se le suministra:

$$\eta = \frac{W_{ci}}{Q_i} = \frac{Q_i - Q_f}{Q_i} = 1 - \frac{Q_f}{Q_i}$$

Donde  $W_{ci}$  es el trabajo hecho por la máquina en cada ciclo. Se ve que la eficiencia depende solo de  $Q_i$  y

de  $Q_f$ . Ya que  $Q_i$  y  $Q_f$  corresponden al calor transferido a las temperaturas  $T_i$  y  $T_f$ , es razonable asumir que ambas son funciones de la temperatura:

$$\frac{q_C}{q_H} = \frac{f(T_f)}{f(T_i)} = g(T_i, T_f)$$

Sin embargo, es posible utilizar a conveniencia, una escala de temperatura tal que

$$\frac{Q_f}{Q_i} = \frac{T_f}{T_i}$$

Sustituyendo la ecuación en la anterior relaciona la eficiencia de la sense Hat con la temperatura

#### Unidades de temperatura

Las escalas de medición de la temperatura se dividen fundamentalmente en dos tipos, las relativas y las absolutas. Los valores que puede adoptar la temperatura en cualquier escala de medición, no tienen un nivel máximo, sino un nivel mínimo: el cero absoluto.<sup>3</sup> Mientras que las escalas absolutas se basan en el cero absoluto, las relativas tienen otras formas de definirse.

#### Relativas

Artículo principal: Unidades derivadas del SI

Grado Celsius (°C). Para establecer una base de medida de la temperatura Anders Celsius utilizó (en 1742) los puntos de fusión y ebullición del agua. Se considera que una mezcla de hielo y agua que se encuentra en equilibrio con aire saturado a 1 atm está en el punto de fusión. Una mezcla de agua y vapor de agua (sin aire) en equilibrio a 1 atm de presión se considera que está en el punto de ebullición. Celsius dividió el intervalo de temperatura que existe entre estos dos puntos en 100 partes iguales a las que llamó grados centígrados °C. Sin embargo, en 1948 fueron renombrados grados Celsius en su honor; así mismo se comenzó a utilizar la letra mayúscula para denominarlos.

En 1954, la escala Celsius fue redefinida en la Décima Conferencia de Pesos y Medidas en términos de un solo punto fijo y de la temperatura absoluta del cero absoluto. El punto escogido fue el punto triple del agua que es el estado en el que las

tres fases del agua coexisten en equilibrio, al cual se le asignó un valor de 0,01 °C. La magnitud del nuevo grado Celsius se define a partir del cero absoluto como la fracción 1/273,16 del intervalo de temperatura entre el punto triple del agua y el cero absoluto. Como en la nueva escala los puntos de fusión y ebullición del agua son 0,00 °C y 100,00 °C respectivamente, resulta idéntica a la escala de la definición anterior, con la ventaja de tener una definición termodinámica.

Grado Fahrenheit (°F). Toma divisiones entre el punto de congelación de una disolución de cloruro amónico (a la que le asigna valor cero) y la temperatura normal corporal humana (a la que le asigna valor 100). Es una unidad típicamente usada en los Estados Unidos; erróneamente, se asocia también a otros países anglosajones como el Reino Unido o Irlanda, que usan la escala Celsius.

Grado Réaumur (°Ré, °Re, °R). Usado para procesos industriales específicos, como el del almíbar.

Grado Rømer o Roemer. Es una escala de temperatura en desuso que fue propuesta por el astrónomo danés Ole Christensen Rømer en 1701. En esta escala, el cero es inicialmente la temperatura de congelación de la salmuera.

Grado Newton (°N). Es probable que Anders Celsius conociera la escala termométrica de Newton cuando inventó la suya. Por consiguiente, la unidad de esta escala, el grado Newton, equivale a (aproximadamente 3,03) kelvines o grados Celsius y tiene el mismo cero de la escala de Celsius.

Grado Leiden. La escala Leiden se utilizaba a principios del siglo XX para calibrar indirectamente bajas temperaturas, proporcionando valores convencionales kelvin de la presión de vapor del helio.

Grado Delisle (°D) La escala Delisle es una forma de medir temperatura concebida en 1732 por el astrónomo francés Joseph-Nicolas Delisle (1688-1768).

#### **Presión atmosférica**

[8]presión atmosférica, a la presión que ejerce la atmósfera sobre la superficie de la Tierra. Esta presión puede ser medida por elementos relativamente sencillos, de hecho hay experimentos que pueden utilizarse en casa para comprender de



manera más sencilla y clara de que se trata esta presión.

La presión atmosférica se mide en Pascales (Pa), aunque para este tipo específico se suelen emplear otras como atmósferas (atm), bares (b), milibares (mb) o el milímetro de mercurio (mmHg) lo que quiere decir que:

Una atmósfera es la presión que se necesita para equilibrar una columna de 760 mm de mercurio. 1 atm = 101325 Pa, aunque en ocasiones se suele utilizar 1 atm = 101300 Pa, por lo que la presión atmosférica física formula es:

$$1 \text{ b} = 100000 \text{ Pa}$$

$$1 \text{ mb} = 100 \text{ Pa}$$

Un milímetro de mercurio es la presión necesaria para aumentar la altura del mercurio en el tubo de Torricelli un milímetro. 1 mmHg = 133,3 Pa.

Otra de la presión atmosférica formula física es la siguiente:

$$P_a = \rho gh$$

$\rho$  es la densidad del mercurio  $\rho = 13550 \text{ kg/m}^3$

$g$  es la aceleración de la gravedad  $g = 9.81 \text{ m/s}^2$

$h$  es la altura de la columna de mercurio  $h = 0.76 \text{ m}$  al nivel del mar

### Humedad

[7] La humedad relativa del aire, se define como la razón entre la presión de vapor de agua en un momento dado ( $P_v$ ) y la presión de vapor de agua cuando el aire está saturado de humedad ( $P_{vs}$ ), a la misma temperatura. La humedad relativa se puede expresar como decimal o como porcentaje.

Dijimos que el observador tenía entonces dos valores  $T$  y  $T_{bh}$  (Temperatura y Temperatura de

bulbo húmedo). Con estos valores podemos obtener la  $P_v$  (Presión de vapor)

### Fórmula psicrométrica

$$P_v = P_{vs,bh} - a_1 * P * (T - T_{bh})$$

$P_v$  = Presión o tensión de vapor

$P_{vs,bh}$  = Presión de vapor de saturación a la Temperatura de bulbo húmedo (Se obtiene a través del Cuadro N°2)

$a_1$  = Factor psicrométrico (varía con la ventilación) (se obtiene del Cuadro N°1)

$P$  = Presión atmosférica (Si no se conoce la presión, pero sí la altura, obtener  $P$  a través del Cuadro N°3)

$(T - T_{bh})$  = Diferencia o depresión psicrométrica (diferencia entre las temperaturas del termómetro de bulbo seco y el de bulbo húmedo)

### VII. Montaje de la estación meteorológica

El estuche Zebra no es resistente al agua, por lo que querrá instalarlo en un lugar protegido como un porche, o hacer un recinto que lo proteja mientras deja que los sensores respiren.

Para lugares expuestos, improvisé un simple protector contra la lluvia con un recipiente de plástico para alimentos y una tapa para llevar china. Está abierto en la parte inferior, con respiraderos en la parte superior, por lo que la temperatura debe igualarse fácilmente con el entorno (se aplica la ley de enfriamiento de Newton). No quería perforar agujeros en los costados, ya que eso dejaría entrar agua, así que monté el Pi en una pieza de madera contrachapada de  $\frac{1}{4}$  " y lo coloqué con velcro en el interior del contenedor.

- Contenedor de almacenamiento de plástico
- Tapa de un contenedor de comida rápida
- Pieza de madera contrachapada de 2 " × 6 "
- madera de contrachapado de 2 " × 3 "
- Velcro® de grado externo
- Montaje tornillos
- espaciadores

Lo fijamos al la placa de madera con los tornillos

Agrega espaciadores al tablero. Necesitará un tamaño específico para adaptarse a la separación entre el Pi y el Sense HAT. Los que están en la parte inferior del Pi pueden ser tan largos como quieras

Luego montamos el sense hat encima de la Rapberry con el sensor para que quede de esta forma

Y finalmente lo metemos en el envase de plástico y listo ya tenemos nuestra estación meteorológica

## VIII. CONCLUSION

Terminado este proyecto, he podido ver que en que una estacion meteorológica no solo se basa en agarrar los datos que se obtienen si no que va desde lo básico como desde el valor que toma el sensor, calcularlos usando dichas ecuaciones que se han mostrado a lo largo de este documento y poder tomar esos valores, graficarlos y gracias a esas graficas he comprender como se esta comportando nuestro ambiente gracias a las diferencias de los valores de un día con otro

Y estas diferencias nos ayudaran a prevenir catástrofes meteorológicas y poder anticiparnos a eventos fortuitos o no deseados

## IX. BIBLIOGRAFIA

[1]Python Software Foundation.( 04 de junio de 2019),Date Time ,<https://docs.python.org/2/library/datetime.html>

[2]Microsoft (2019,7,30) Importar-Csv,<https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.utility/import-csv?view=powershell-6>

[3] Real Python (2012–2019), Envío de correos electrónicos con Python,<https://realpython.com/python-send-email/>

[4]Uniwebsidad(2006-2019), Módulos de sistema,<https://uniwebsidad.com/libros/python/capitulo-10/modulos-de-sistema>

[5]Python Software Foundation,( 05 de junio de 2019) 28.1. sys- Parámetros y funciones específicos del sistema,<https://docs.python.org/2/library/sys.html>

[6]Derechos de autor © Parewa Labs Pvt. Ltd. Todos los derechos reservados.(30 de junio del 2016), Módulo de tiempo de Python<https://www.programiz.com/python-programming/time>

[7]El tiempo - Tutiempo Network, S.L.(03 del mayo del 2019), Parámetros de Humedad <https://www.tutiempo.net/meteorologia/ecuaciones.html>

[8]Hostmonster,(02 de enero del 2018), Presión Atmosférica<http://misistemasolar.com/presion-atmosferica/>

[9]Python Software Foundation,( 04 de junio de 2019) Tiempo de acceso y conversions ,<https://docs.python.org/2/library/time.html>

[10]Fundación Wikimedia, Inc.,( 2 ago 2019 ),

Temperatura,<https://es.wikipedia.org/wiki/Temperatura>