**Who:**

Bader Albader

Jacob Tran

Tyler Valentine

Garrett Senor

Chuck Mezhir

**Title:**

Kong

**Automated Tests:**

**By Bader and Jacob**

**Hardware Test:**

***Description:***This test is to ensure that the sensors being used are working effectively. The sensors used are a temperature sensor and a humidity sensor.

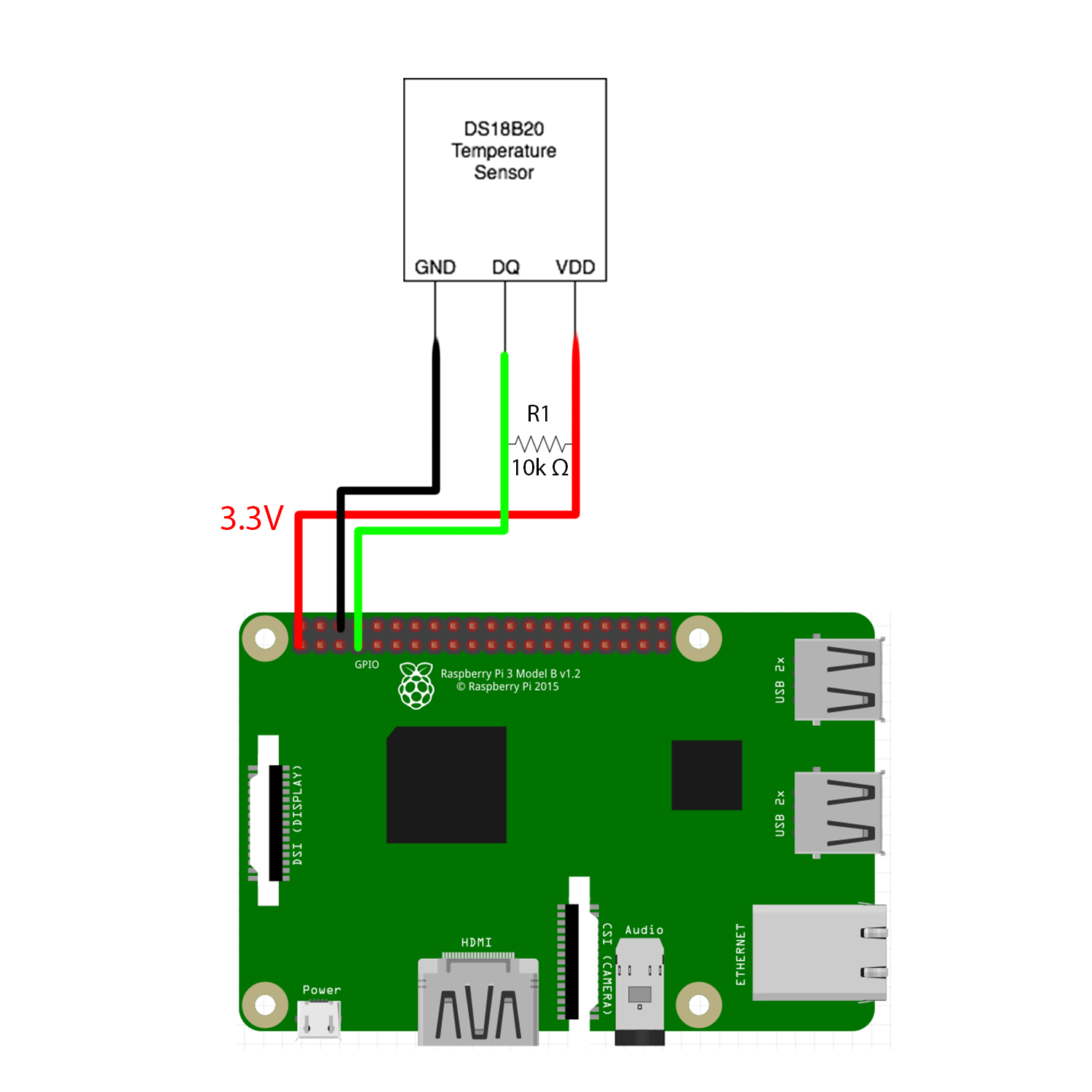
***Type:*** White box test.

***Setup:***

1. A Raspberry Pi 3 was used as the micro controller. The temperature sensor was connected to a 10K Ohm pull down resistor in order to read data. The sensor is also connected to 3.3V on the Pi and ground on the Pi.
2. A Python script was written that reads the temperature and humidity from GPIO pins on the Pi. Once the temperature is read is it fed to the backend database.

***Results:***

1. Temperature, humidity, data and time is printed out when the script is run.
2. The same data points are then updated on the database.

****

**Sensor Test:**

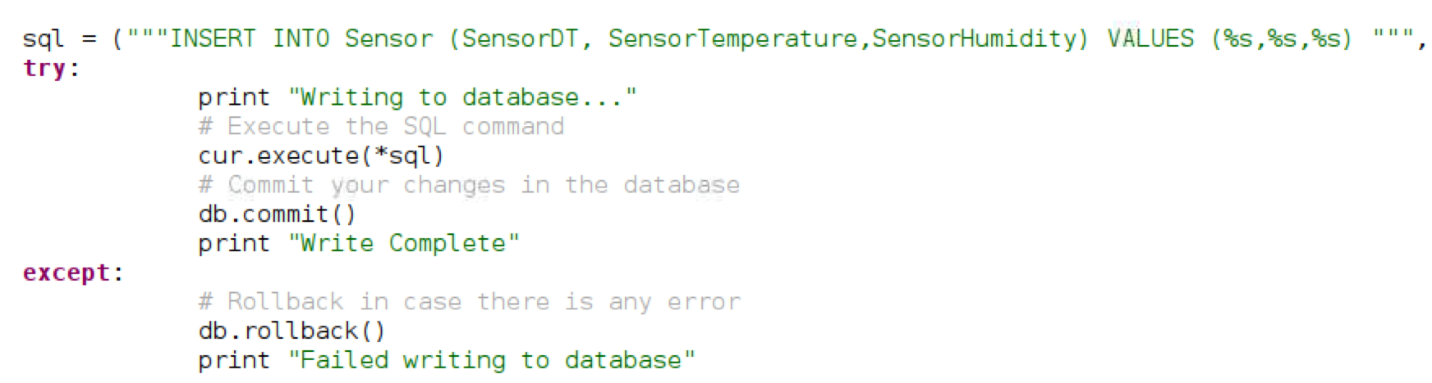
****

Figure 1: PyUnit test

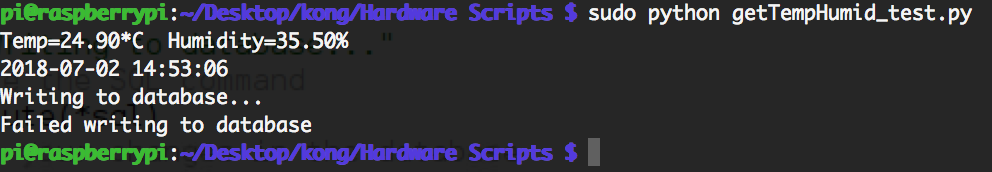
****

Figure 2: failed to connect database

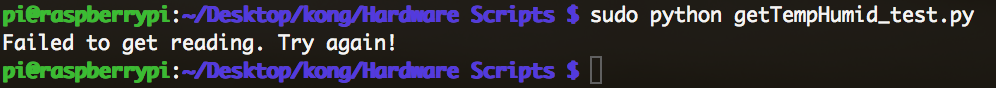
****

Figure 3: failed to read sensor

**Database Test:**

***Description:***This test is to ensure that the database is running and that data is being updated

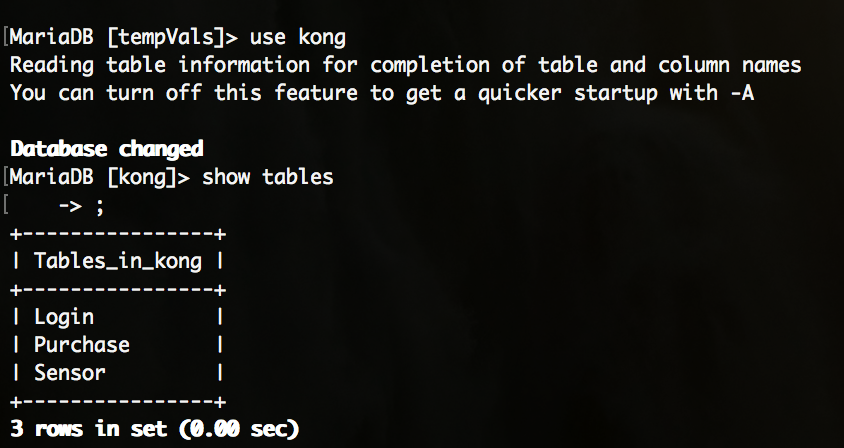
***Type:*** White box test.

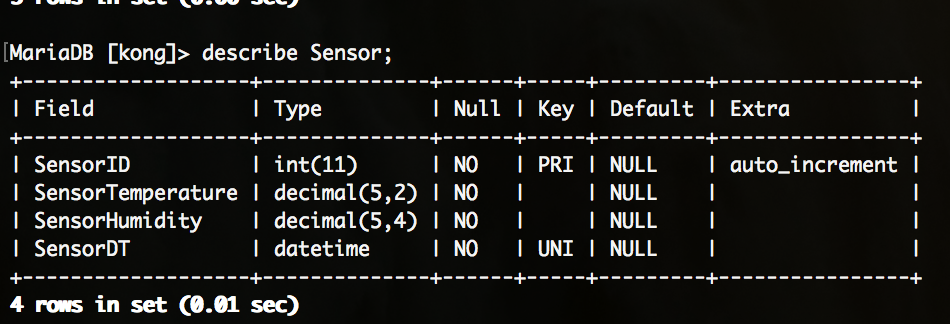
***Setup:***

1. We used python scripts to test the communication between the database and the Raspberry Pi 3. We temporarily used dummy data for the humidity data since our sensor is not set up
2. We then verify if the new input to the database is as expected.

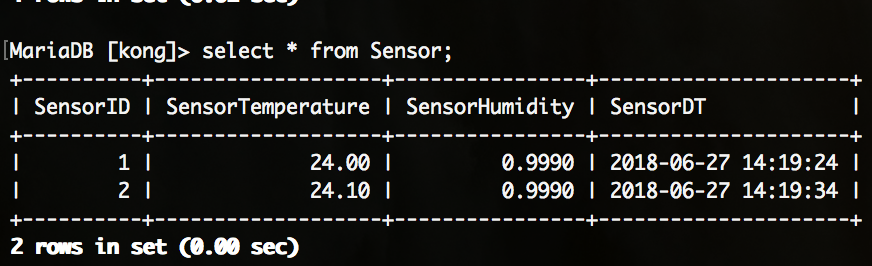
***Results:***

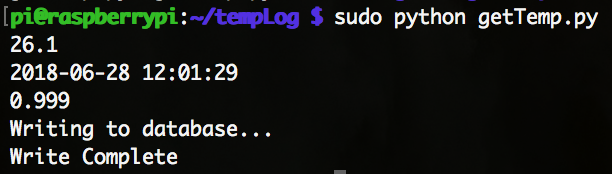
1. Database updated after the script is run.
2. Results are displayed in the mysql database.

****

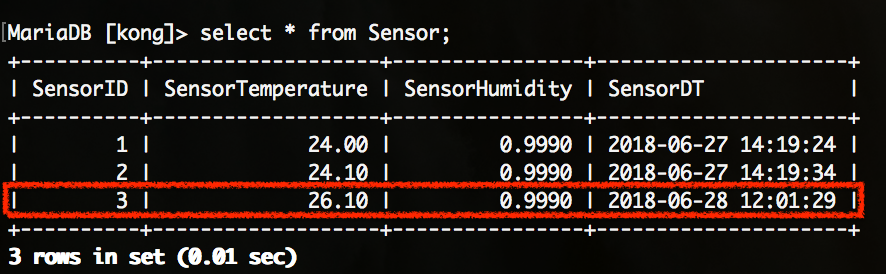
*Figure 1. ‘Kong’ Database Table.*

*Figure 2. ‘Kong’ Database, ‘Sensor’ Table data types*

*Figure 3. ‘Kong’ Database, ‘Sensor’ Table imported data.*



*Figure 4. Running script:*

*Figure 5. ‘Kong’ Database, ‘Sensor’ table, updated data after script run.*

**Manual Test:**

**Front End: Tyler and Charles**

* **Test 1**

**Feature set added**: Login and Password integration into the menu bar:

**What we did for testing:** After login and password fields were added into html code dummy user identities were entered to test the functionality of the system. Results of the testing confirmed (prior to database integration) that our dummy identities would, in fact, route the user to the proper landing page and leaving the fields blank would prompt the user to enter the required information. These results matched our expectations for this aspect of the landing page.

**User Story**: User Jenny wants to log into her account and check the status of her greenhouse. She enters the appropriate log in credentials (her email address and chosen password) into the login field on the main page. Upon hitting the log in button she is taken to the data.html page which successfully displays the current temperature and humidity in her greenhouse as well as historical data encompassing the last 24 hour period, week and fortnight. **Test Successful**



* **Test 2**

**Feature set added**: Creating a new user account:

**What we did for testing:** When a customer decides they want to purchase the product they are prompted to enter new username and password information. We needed to test to make sure that this information was successfully recorded in preparation for handling by the database and back-end. Testing confirmed that information was collected and was ready for storage. Additionally, when any of the required fields were not properly entered the user was prompted to provide said information before a new account would be created. Finally, we confirmed that was an account was successfully created the user was passed to the proper confirmation page.

**User Story:** User Aidan has visited the main page and is interested in buying our product. He decides to purchase and clicks the buy now button. He is routed to the signup page and asked to create a username (his email address) and password as well as shipping and payment information. He enters a username that is already in use and is prompted to create a new username. Upon creation of a valid username he is routed to the success page letting him know that his account has been successfully created. He can now expect to receive his product in the mail and begin accessing data online once hardware setup has been completed. **Test Successful.**



* **Test 3**

**Feature set added**: Displaying real-time and historical data on the webpage:

**What we did for testing:** Test that graphing data plugin is displaying information properly. Also, test that real-time data has been updating as expected. Testing confirmed that data from the raspberry was being collected and sent to the database to be displayed on the website.

**User Story:** User Miguel is untrusting of technology by nature. He wants to make sure that the hardware is has ordered from our website is working properly and that the results are being displayed properly online. He gets a digital thermometer at the local hardware store and takes measurements in his greenhouse at regular intervals to be compared with the data that is being displayed on our website. After careful review he finds the real-time data displayed on the sight matches his expectations based on his own measurements. Peace of mind is achieved! **Test Successful.**

