

1.1 Problem Description

The aim of this project is to design a weather monitoring system that can take measurements of the temperature, humidity from DHT-11(Temperature and Humidity sensor) and save them on a server and this data will be pushed onto android application and the user can be aware of the readings recorded.

1.2 Procedure

Step 1: Setting up the pi for the Raspberry pi IoT Technology

- Head lessly Booting your Raspberry Pi IoT System

Step 2: Working with Raspbian os

- PuTTY Configuration

Step 3: Integrating DHT-11 Sensor with pi

- Python Library
- Software Setup

Step 4: Setting up MySQL Database

Step 5: Setting up Raspberry pi phpMyAdmin

- Configuring Apache for phpMyAdmin

Step 6: Developing an Android Application

1.2.1 Setting up the Pi for the Raspberry pi IoT project

In order to setup pi, you'll need two things:

- A downloaded image of the Operating System you want to put on the card.
- A Program called Win32 DiskImager.

Setup:

1. Download the Raspbian OS for your Pi. There are many other distributions you can use, but Raspbian remains the most common and convenient for beginners.
2. The latest version of Debian for the Raspberry Pi at the time of writing is Raspbian Wheezy.

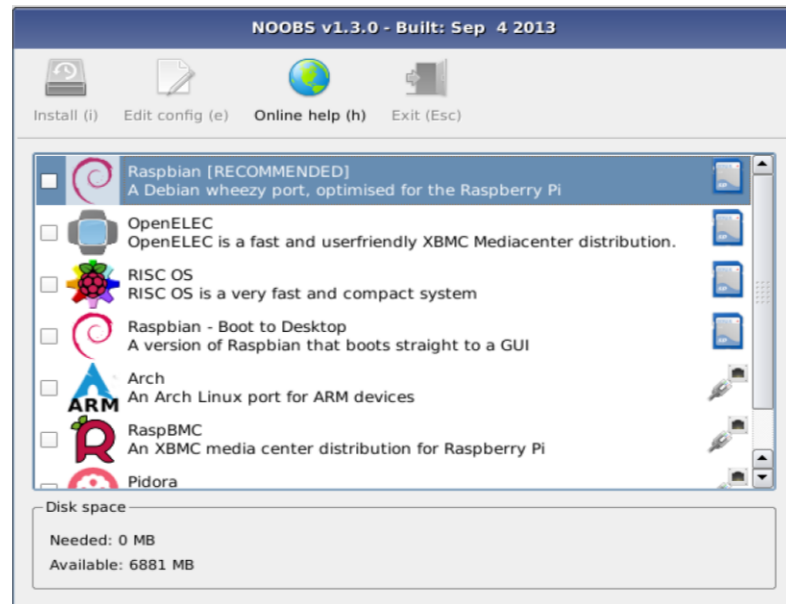


Figure 1: Downloading Raspbian OS

3. Your download should start by itself. You may need to confirm by clicking OK in your browser.
4. Download Win32 disk imager. Then unzip the file win32diskimager-binary.zip, then double-click Win32DiskImager.exe in the unzipped folder called win32diskimager-binary. Click Yes to confirm you want to run the program. Then it will start. Once you've established that it works, close it down and plug in your SD card.
5. Having plugged in your SD card, (re)start Win32Diskimager. Choose the drive you want to copy the image to.
6. Then click on the folder icon and choose the unzipped .img file from earlier that you want to put on the SD card. Then click Write, to write the Operating system on the card from the .img file.

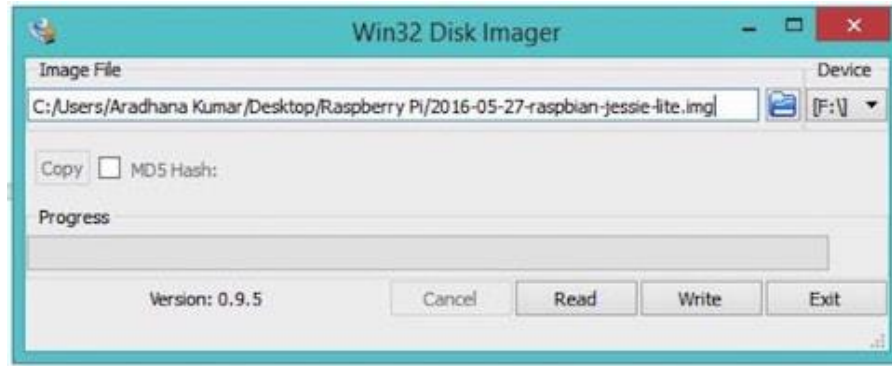


Figure 2: Extracting img file into SD Card

7. You will then be asked to confirm. Check carefully that you are writing to the correct device and if so, click Yes.

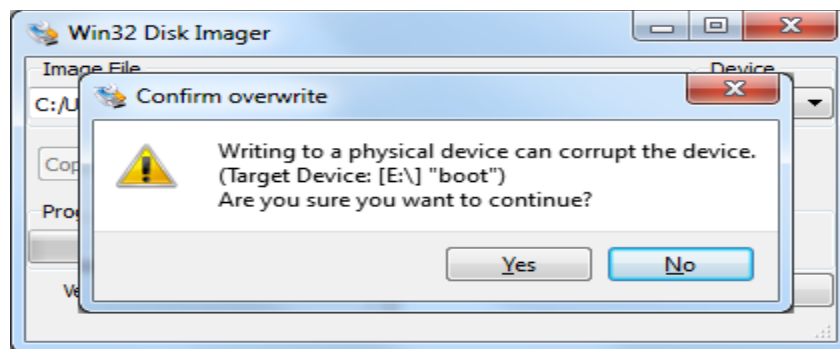


Figure 3: Writing into SD card

8. After its done, then you can eject the card reader and remove the SD card. Then you can try it out in your Raspberry Pi. If all went well, your Pi should boot up (start) into your new operating system. It may take a couple of minutes. If nothing's happened after that, look at the LEDs on the Raspberry Pi. If you only get the red power LED, either the card is not inserted properly or it can't boot this combination of card and image, or something went wrong with the process

Head lessly Booting your Raspberry Pi IoT System:

Getting a dedicated monitor, mouse, and keyboard to use your Raspberry Pi might become an unnecessary hassle. Access to the terminal is sufficient to get most things done. So we eliminate the need for extra hardware by logging into the Pi using your personal laptop through SSH.

- Latest versions of Raspbian come with SSH enabled by default, therefore, you can run Pi remotely even while setting it up for the first time. Get a router with DHCP

enabled. This is necessary because we want our Pi to have a unique IP address to be able to connect to it.

- Connect both your Laptop and your Pi to the router via ethernet cables. Your Laptop and Pi now share a Local Area Network and can identify each other using their unique IP address.
- To use SSH, we need the IP address of your Pi. Run the Advanced IP Scanner. This will list out the IP addresses of all devices on your network and their manufacturer

1.2.2 Working with Raspbian os

Get the IP address of pi either using advanced IP scanner or from logging on using particular router credentials into jiofi.local.html(if router is of jio network). Next you will be connecting your Raspberry Pi and adjusting some final settings. After that, you'll be ready to go!

- Run PuTTY
- Enter your Raspberry Pi's IP address

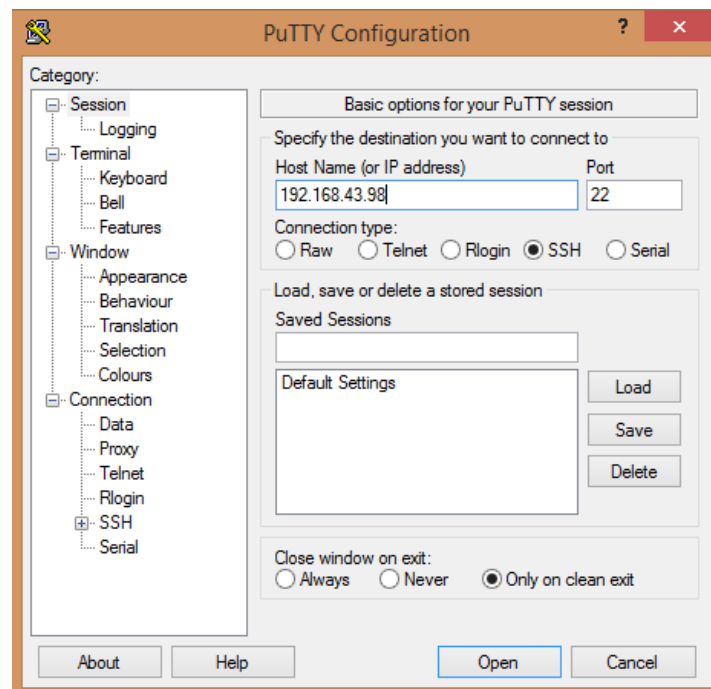
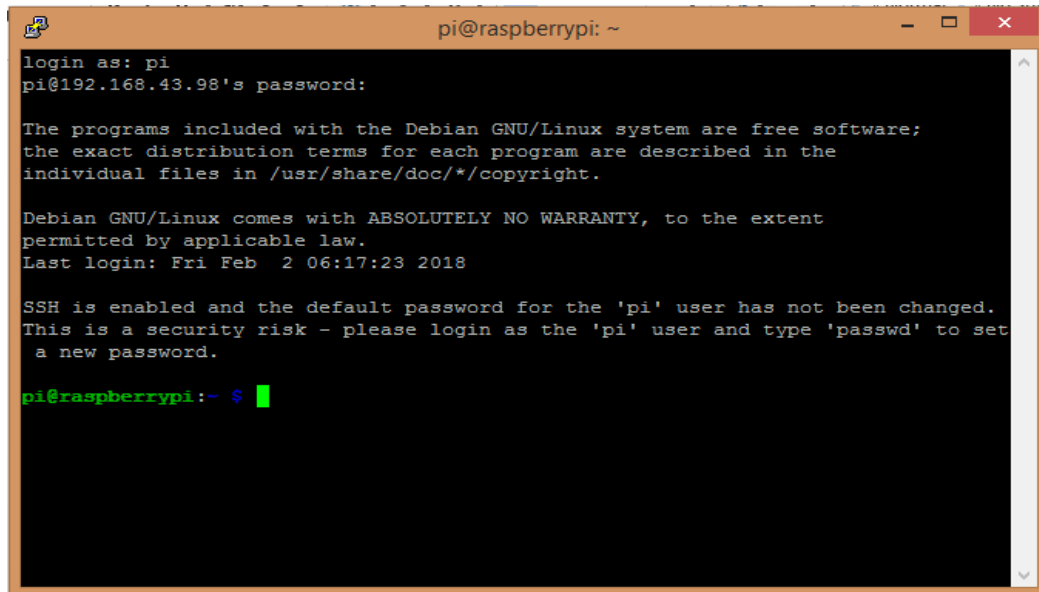


Figure 4: PuTTY Configuration

- You can now open the connection. A terminal window will pop up. To connect your Raspberry Pi, you will need to fill out these fields to login.

Login: pi Password: #####



```
pi@raspberrypi: ~
login as: pi
pi@192.168.43.98's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Feb  2 06:17:23 2018

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.

pi@raspberrypi:~ $
```

Figure 5: Raspbian OS

- Once you enter the login credentials, you will connect to your Raspberry Pi terminal window. Enter the RASPI-CONFIG window and hit enter. RASPI-CONFIG is the software configuration tool for Raspberry Pi.
- Go to “Expand Filesystem.” After a few blinks, you’ll see that your partitions are resized and the partition issue mentioned while installing the OS is gone.

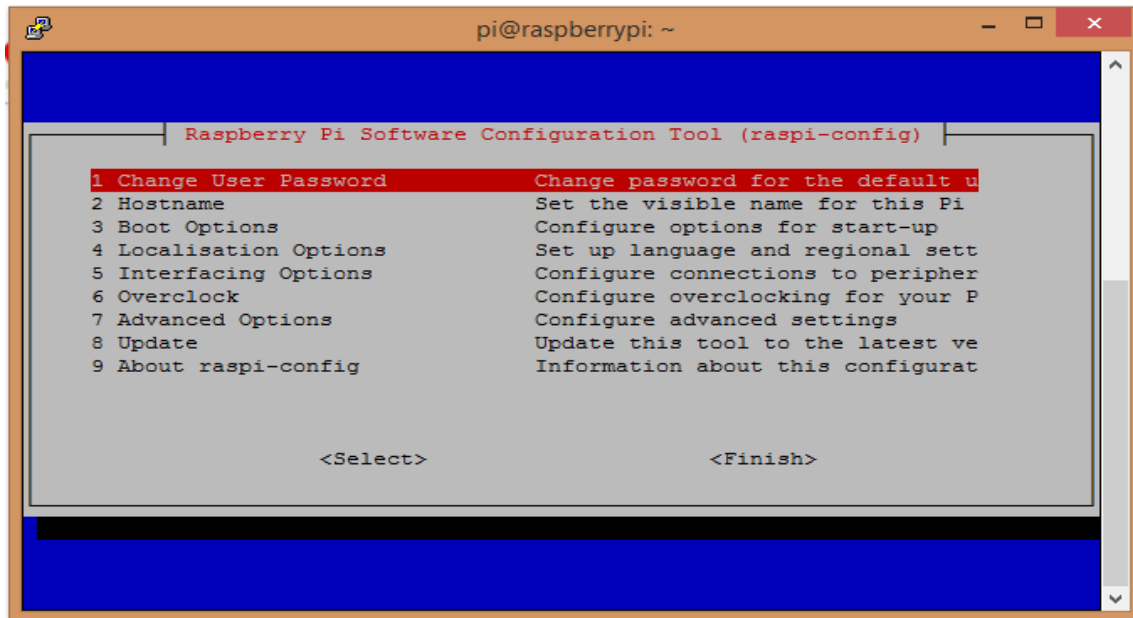


Figure 6: Raspberry Pi Software Configuration tool

- Reboot your Raspberry Pi (You may need to restart your PC as well) and re-connect like you just did.

1.2.3 Integrating DHT-11 with pi

The DHT11 is a low-cost temperature and humidity sensor. It isn't the fastest sensor around but its cheap price makes it useful for experimenting or projects where you don't require new readings multiple times a second. The device only requires three connections to the Pi. +3.3v, ground and one GPIO pin.

Besides the VCC and GND, you just need to use one GPIO pin on the Raspberry Pi to make use of the DHT11 module.

Raspberry Pi	DHT11 Module
3.3v P1	VCC (V)
GND P6	GND (G)
GPIO4 P7	DATA (S)

- Programming Code should be generated in order to configure sensor with pi and to get the temperature recordings.



Figure 7: Integrating DHT-11 sensor with pi

1.2.3.1 Python Library

The DHT11 requires a specific protocol to be applied to the data pin. In order to save time trying to implement this yourself it's far easier to use the Adafruit DHT library. The library deals with the data that needs to be exchanged with the sensor but it is sensitive to timing issues. The Pi's operating system may get in the way while performing other tasks so to compensate for this the library requests a number of readings from the device until it gets one that is valid.

1.2.3.2 Software Setup

1. To start with update your package lists and install a few Python libraries :

```
sudo apt-get update
```

```
sudo apt-get install build-essential python-dev
```

2. Then clone the Adafruit library from their repository :

```
git clone https://github.com/adafruit/Adafruit_Python_DHT.git
```

```
cd Adafruit_Python_DHT
```

3. Then install the library for Python 2:

sudo python setup.py install

1.2.4 Setting up MySQL Database:

1. To store our temperature values we will need a database, we will use MySQL for this.

Now install MySQL by entering the following, press y when prompted.

sudo apt-get install mysql-server mysql-client php5-mysql

2. A prompt will pop up asking you to enter a password for the MySQL database. Make sure you remember it/write it down!!!

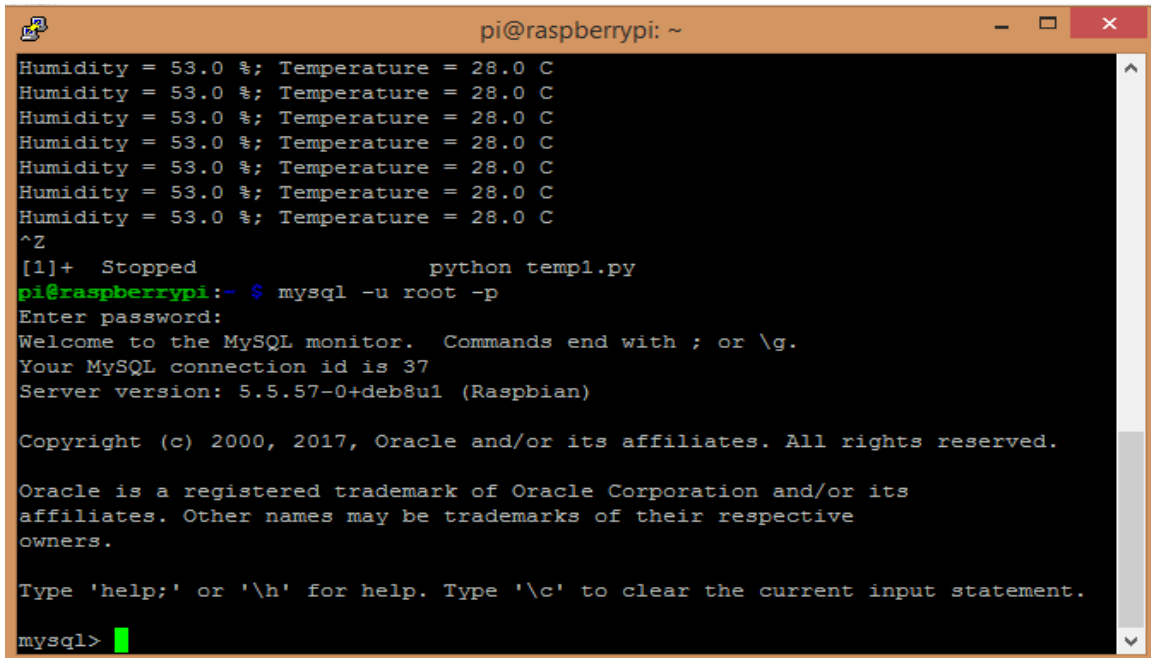
Now download python library for MySQL Database:

sudo apt-get install python-mysqldb

3. Now open MySQL:

sudo mysql -u root -p

4. This logs us in to MySQL as the root user (-u) and it will prompt for a password (-p) on entry.



```
pi@raspberrypi: ~
Humidity = 53.0 %; Temperature = 28.0 C
Humidity = 53.0 %; Temperature = 28.0 C
Humidity = 53.0 %; Temperature = 28.0 C
Humidity = 53.0 %; Temperature = 28.0 C
Humidity = 53.0 %; Temperature = 28.0 C
Humidity = 53.0 %; Temperature = 28.0 C
Humidity = 53.0 %; Temperature = 28.0 C
^Z
[1]+  Stopped                  python temp1.py
pi@raspberrypi:~ $ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 37
Server version: 5.5.57-0+deb8u1 (Raspbian)

Copyright (c) 2000, 2017, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

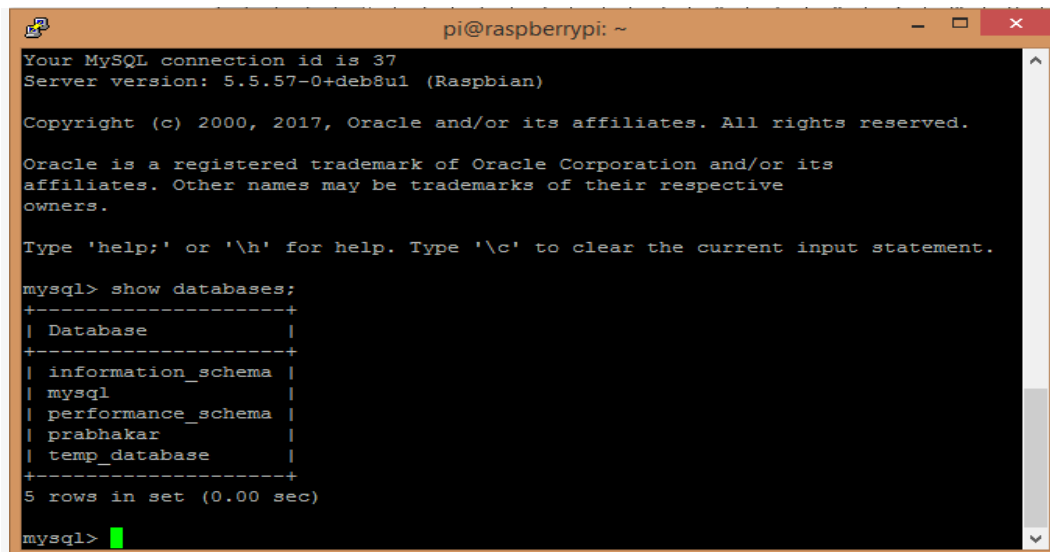
Figure 8: MySQL Database in Raspbian os

5. We are now going to create a database, I called mine temp_database which is a bit unimaginative.

CREATE DATABASE temp_database;

6. We can check this has worked by entering the below. A list of the databases currently held by MySQL will be displayed. Don't be alarmed if there is more than the one you just created.

SHOW DATABASES;

A screenshot of a terminal window titled 'pi@raspberrypi: ~'. The terminal shows the MySQL command prompt 'mysql>' followed by the command 'show databases;'. The output is a table with one column 'Database' and five rows: 'information_schema', 'mysql', 'performance_schema', 'prabhakar', and 'temp_database'. The terminal also displays copyright information for MySQL and the server version '5.5.57-0+deb8u1 (Raspbian)'.

```
pi@raspberrypi: ~
Your MySQL connection id is 37
Server version: 5.5.57-0+deb8u1 (Raspbian)

Copyright (c) 2000, 2017, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

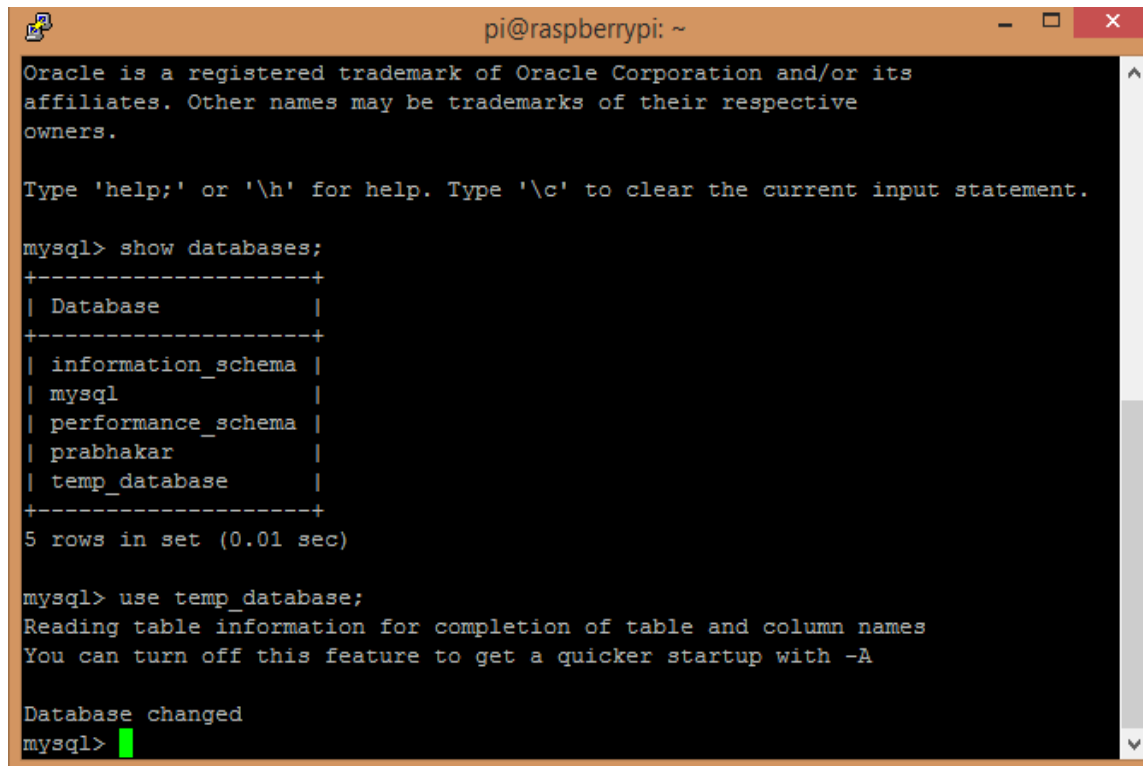
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| prabhakar |
| temp_database |
+-----+
5 rows in set (0.00 sec)

mysql>
```

Figure 9: List of Databases in MySQL

7. Now we want to make a new table in the temp_database. To this we firstly have to tell MySQL that we wish to use the temp_database:

USE temp_database;

A terminal window titled 'pi@raspberrypi: ~' with standard window controls. It displays the MySQL command-line interface. The user has entered 'show databases;', which returns a list of five databases: information_schema, mysql, performance_schema, prabhakar, and temp_database. The user then enters 'use temp_database;', which changes the current database to temp_database. The prompt 'mysql>' is followed by a green cursor.

```
pi@raspberrypi: ~
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database                |
+-----+
| information_schema      |
| mysql                   |
| performance_schema      |
| prabhakar               |
| temp_database           |
+-----+
5 rows in set (0.01 sec)

mysql> use temp_database;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql>
```

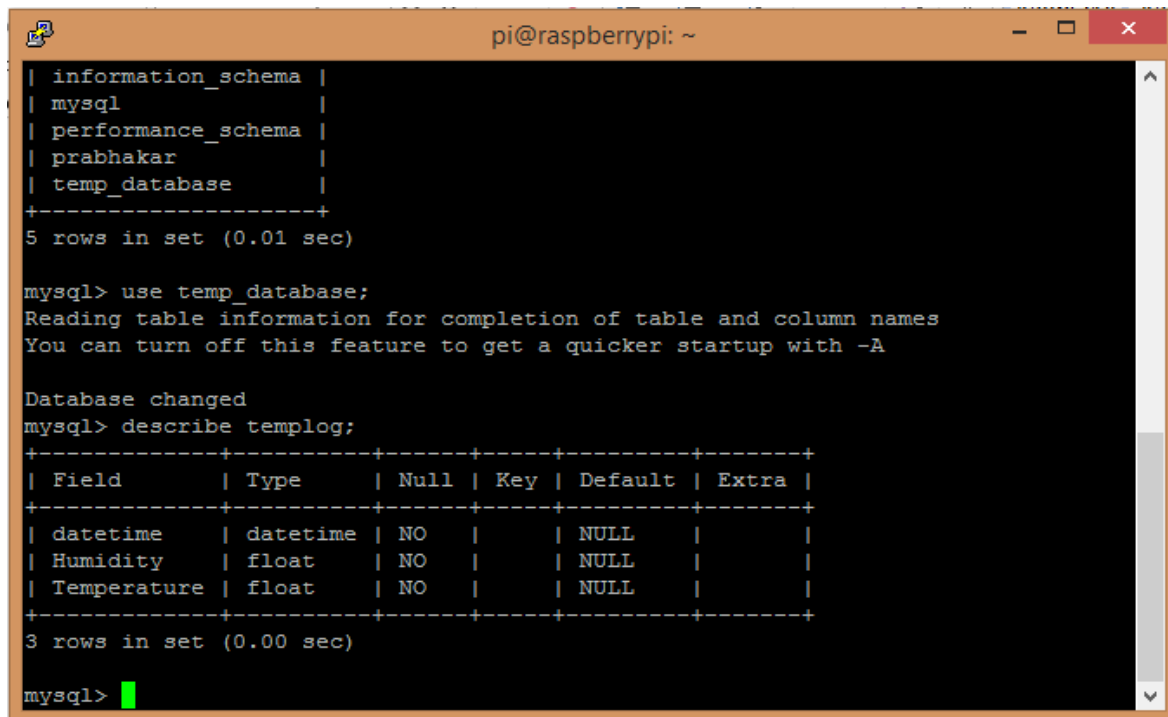
Figure 10: Accessing temp_database

8. We now create a table in MySQL using the following commands. We are making a table called templog that has three fields; datetime (of type DATETIME), temperature(of type FLOAT) and Humidity(of type FLOAT).

CREATE TABLE templog(datetime DATETIME NOT NULL,temperature FLOAT(5) NOT NULL, Humidity FLOAT(5) NOT NULL);

9.To check that our table is correct we can check by entering the following:

DESCRIBE templog;



```
pi@raspberrypi: ~  
| information_schema |  
| mysql |  
| performance_schema |  
| prabhakar |  
| temp_database |  
+-----+  
5 rows in set (0.01 sec)  
  
mysql> use temp_database;  
Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A  
  
Database changed  
mysql> describe templog;  
+-----+-----+-----+-----+-----+-----+  
| Field      | Type      | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| datetime   | datetime  | NO   |     | NULL    |       |  
| Humidity    | float     | NO   |     | NULL    |       |  
| Temperature | float     | NO   |     | NULL    |       |  
+-----+-----+-----+-----+-----+-----+  
3 rows in set (0.00 sec)  
  
mysql>
```

Figure 11: Structure of templog table in MySQL

You will get the following output, describing the table's fields.

10. Exit MySQL by pressing **ctrl+z**.

1.2.5 Raspberry Pi phpMyAdmin

1. I have already done with setting up our own web server and need to setup PHP for it.

Now, install the PHPMyAdmin package by entering the following command on your Raspberry Pi.

sudo apt-get install phpmyadmin

2. It will now begin to install. You will be presented with a screen asking the type of web server you want it to run off. Select apache2 in it.

3. Next we will need to configure PHPMyAdmin to connect to our MySQL database server. (The one we set up previously in installing the Raspberry Pi MYSQL step). To do this select **yes** at the next prompt.

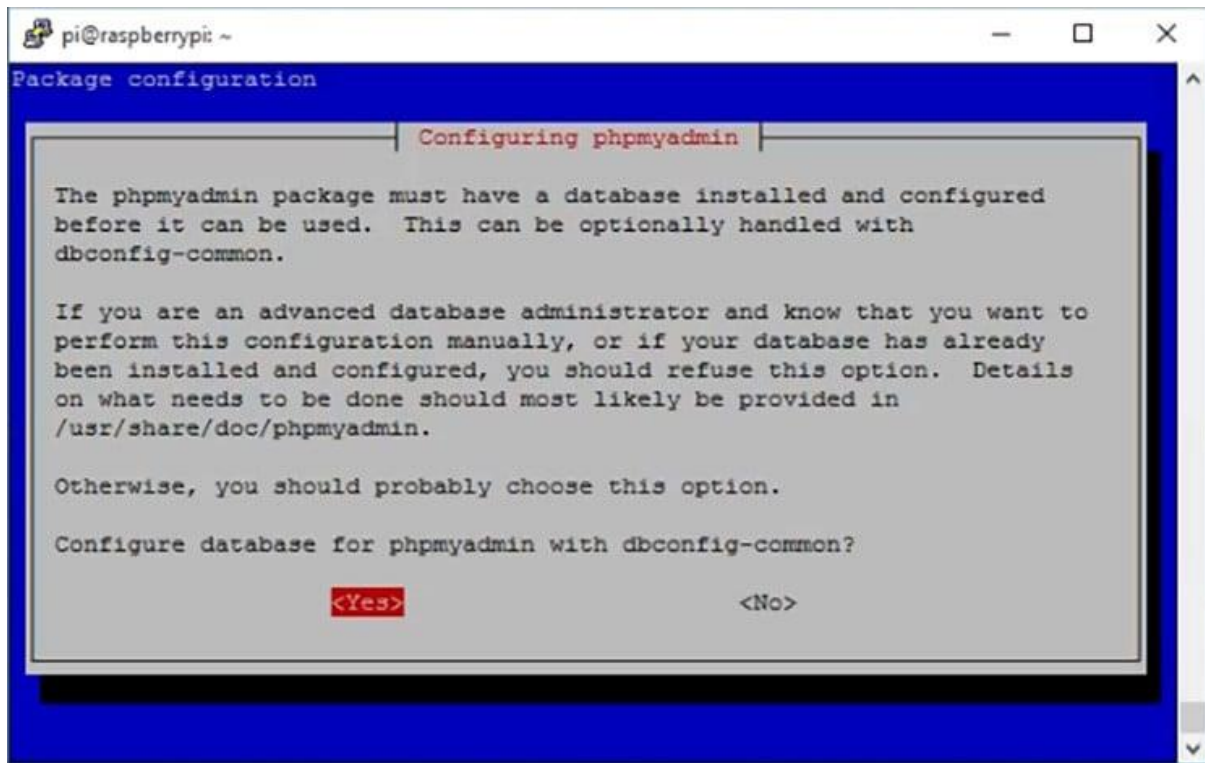


Figure 12: Configuring database for phpMyAdmin

4. The setup tool will now ask for a password, enter the one you set for **root** when setting up your SQL Server, this is needed for phpMyAdmin to talk with the SQL Server and manage your databases.
5. Next it will ask you to set a password for phpMyAdmin itself. It is best to set this password to something different to your root SQL password. Make sure you remember it as this is the password you will need to access the interface.
6. With that done we can now proceed to configure our web servers for use with phpMyAdmin.

Configuring Apache for phpMyAdmin:

- To begin setting up Apache for use with phpMyAdmin enter the following command into terminal:
sudo nano /etc/apache2/apache2.conf
- Now at the bottom of this file enter the following line:

Include /etc/phpmyadmin/apache.conf

Once done save & exit by pressing *CTRL +X* and then *y*.

- Now simply restart the Apache service by entering the following command:

sudo /etc/init.d/apache2 restart

Now you should be able to access the phpMyAdmin from a browser. To test go to the following address in your browser.

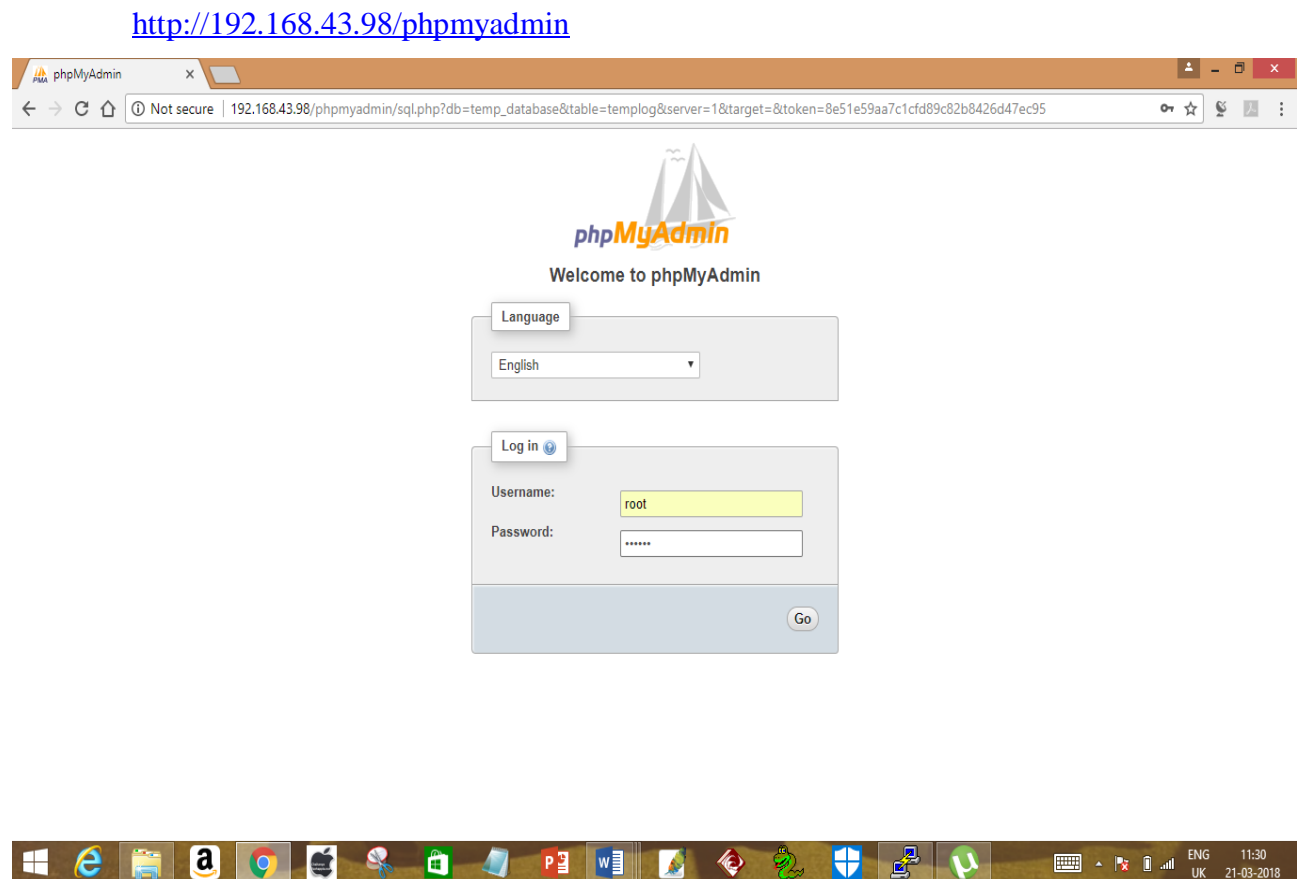


Figure 13: Accessing phpMyAdmin in browser

1.2.6 Developing an android application

Android:

Android is a software stack for mobile devices that includes an operating system,

middleware and key applications. The android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language.

The Android SDK includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator (based on QEMU), documentation, sample code, and tutorials. Currently supported development platforms include x86-architecture computers running Linux (any modern desktop Linux distribution), Mac OS X 10.4.8 or later, Windows XP or Vista.

1.2.6.1 Creating an Android Project

The first step is to create a simple Android Application using Android studio. When you click on Android studio icon, it will show screen as shown below

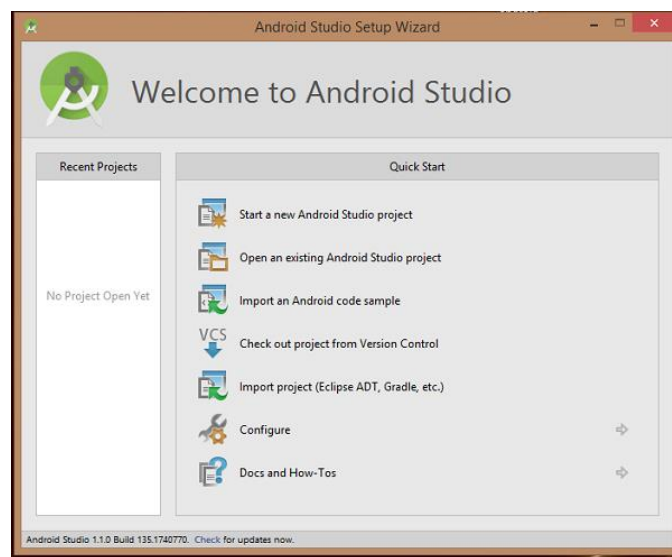


Figure 14: Home page of android studio

You can start your application development by calling start a new android studio project. In a new installation frame should ask Application name, package information and location of the project.—

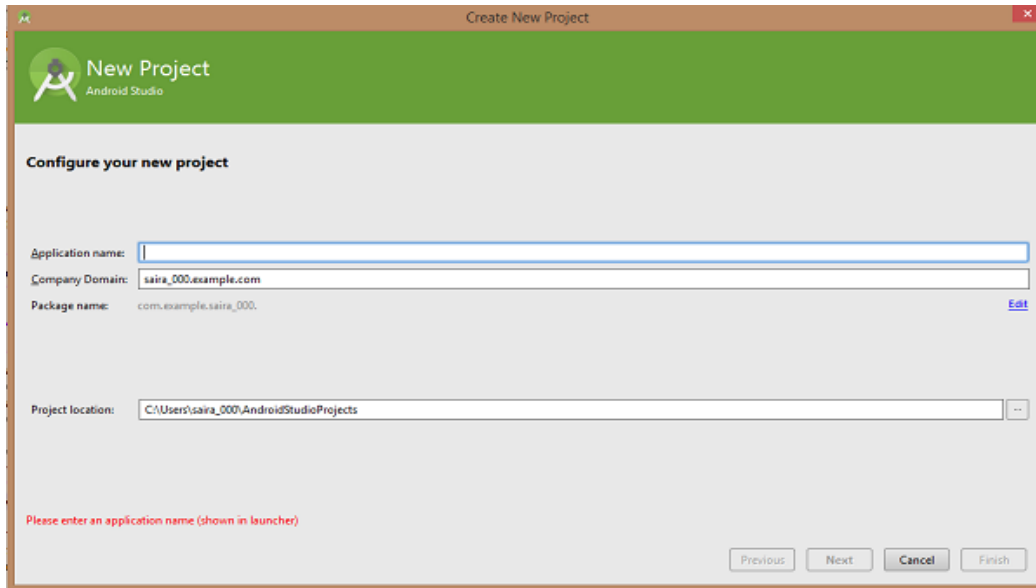


Figure 15: Creation of new project

The next level of installation should contain selecting the activity to mobile, it specifies the default layout for Applications. At the final stage it going to be open development tool to write the application code.