



Final Deliverable - Screen Dumps
Course Group 4
Date: 12/06/2021
Dr. Ramiro Liscano

Group Members:
Shanjay Kailayanathan - 100624670
Jana Kanagalingam -100603975
Ireni Ruthirakuhan - 100657302
Jerusha Macwan - 100723319

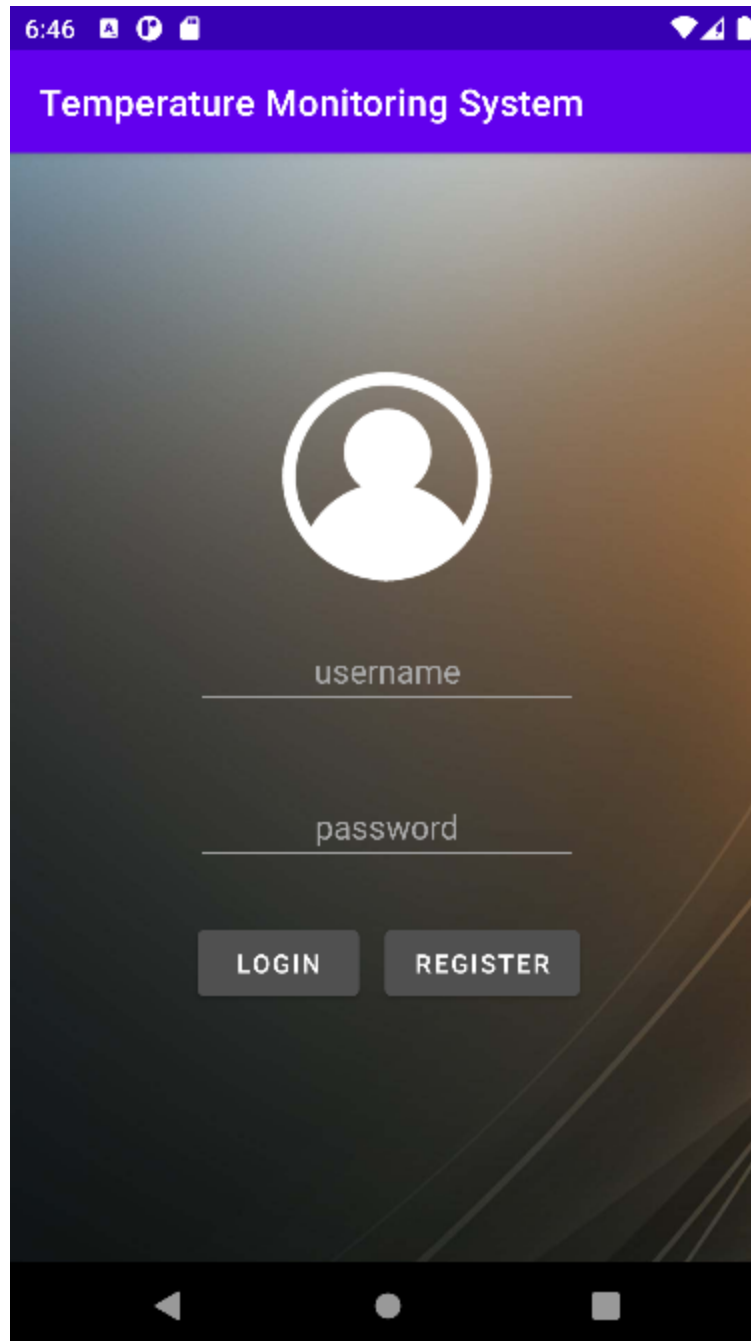


Figure 1: App login page where the user can register or login if they're already registered.

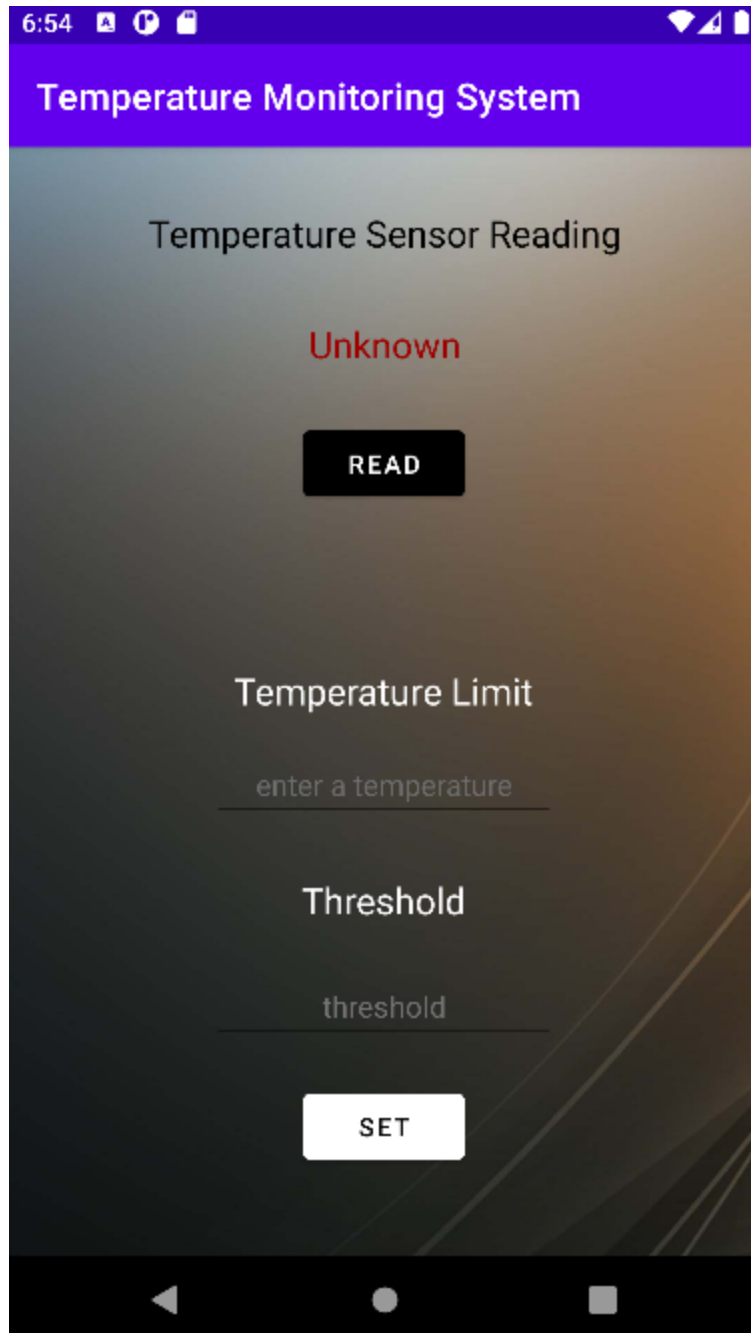


Figure 2: The main page after logging in which'll allow the user to view the temperature reading as well as set the temperature limit & threshold.

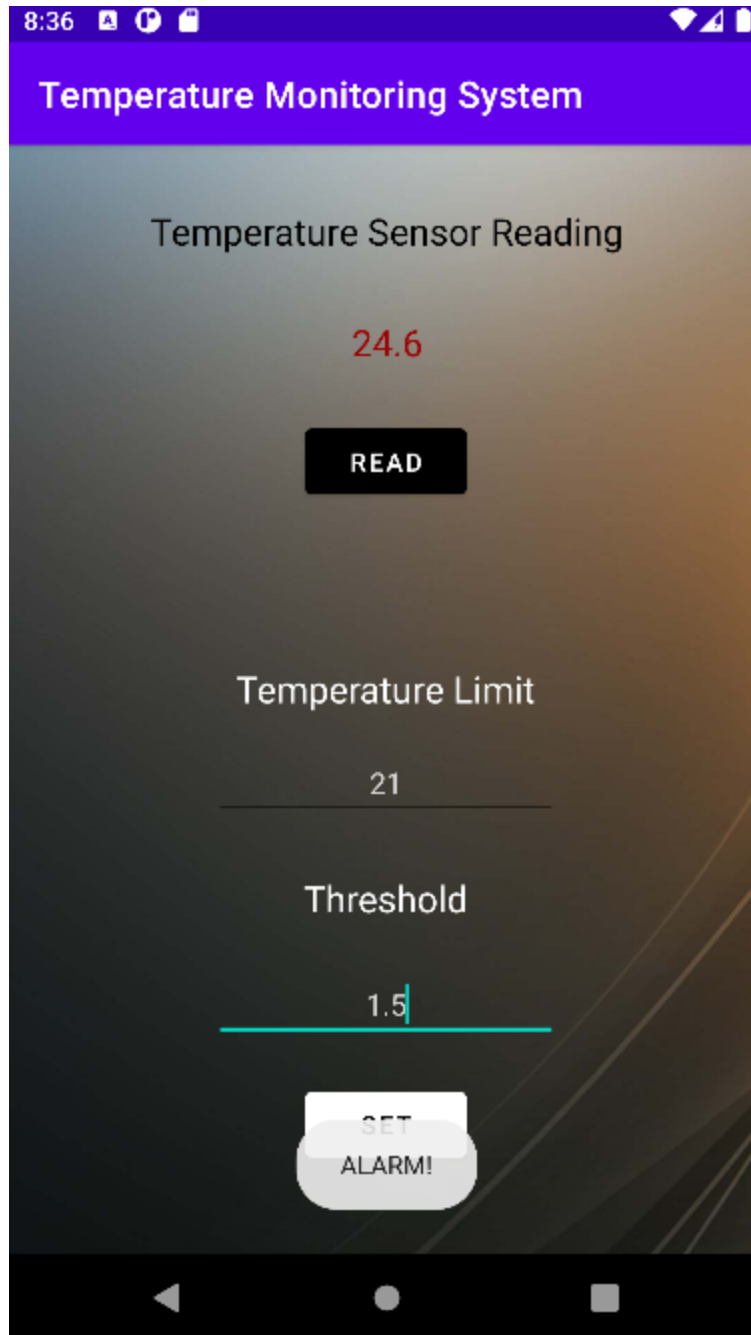


Figure 3: When the temperature exceeds the limit & threshold, the user will be notified in the app.

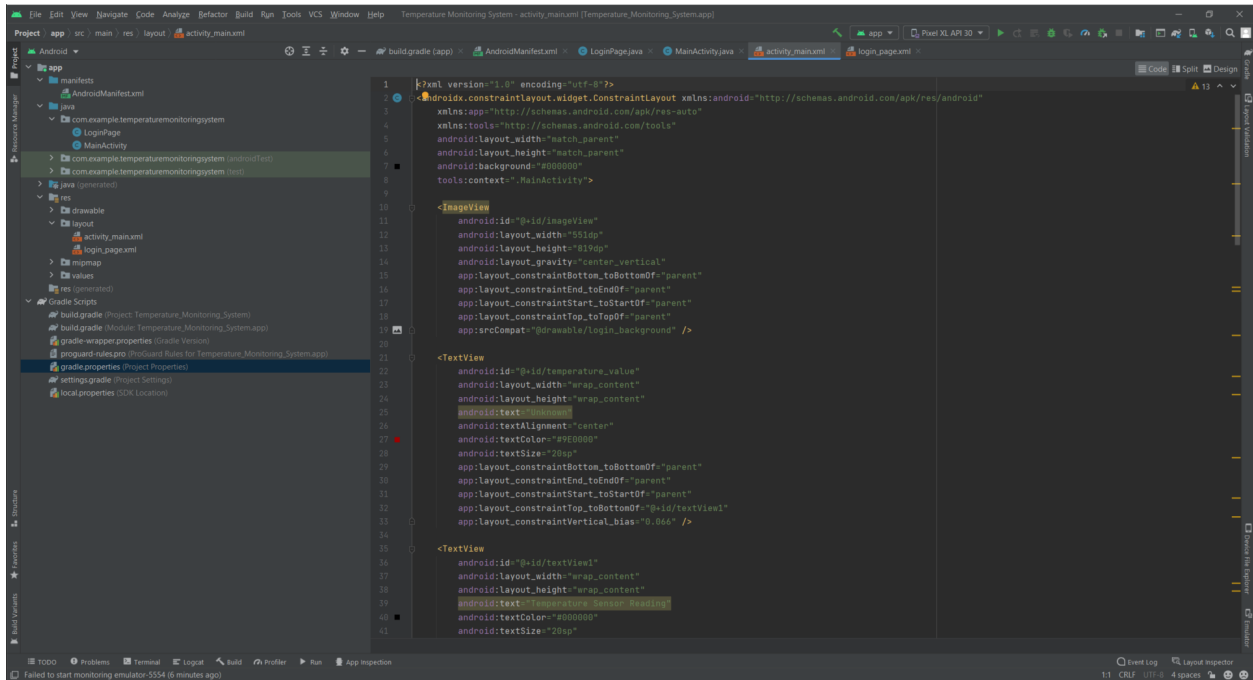


Figure 4: A glimpse of our android code for our mobile application.

```

root@ubuntu-Group4: ~
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-91-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue Dec  7 00:32:43 UTC 2021

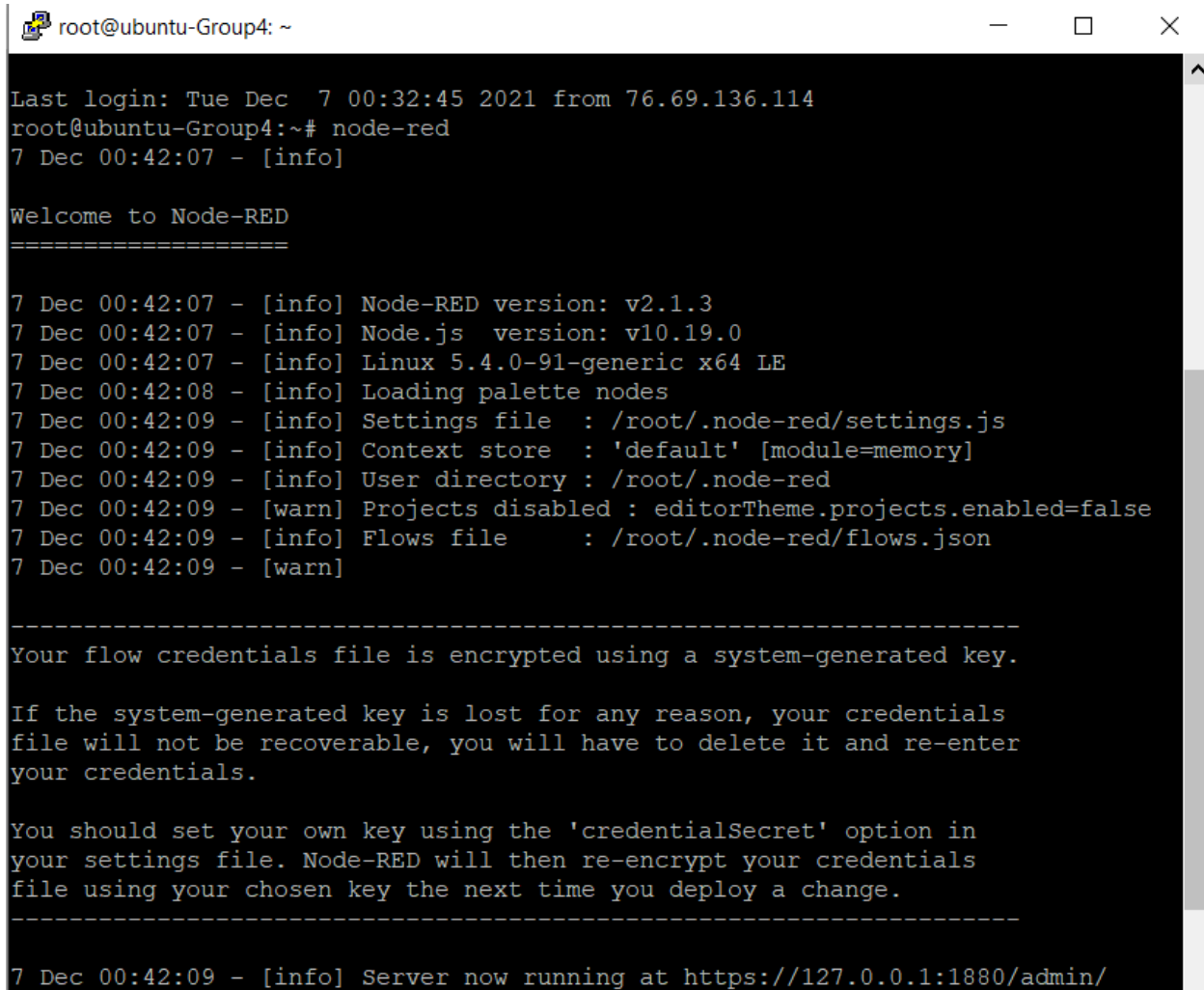
System load:  0.0           Users logged in:      0
Usage of /:   8.1% of 48.29GB IPv4 address for eth0: 137.184.169.237
Memory usage: 32%          IPv4 address for eth0: 10.20.0.5
Swap usage:   0%           IPv4 address for eth1: 10.118.0.2
Processes:   116

0 updates can be applied immediately.

Last login: Thu Dec  2 17:42:45 2021 from 174.119.112.32
root@ubuntu-Group4:~# mosquito
1638837181: mosquito version 2.0.14 starting
1638837181: Using default config.
1638837181: Starting in local only mode. Connections will only be possible from
clients running on this machine.
1638837181: Create a configuration file which defines a listener to allow remote
access.
1638837181: For more details see https://mosquitto.org/documentation/authenticat
ion-methods/
1638837181: Opening ipv4 listen socket on port 1883.
1638837181: Error: Address already in use
1638837181: Opening ipv6 listen socket on port 1883.
1638837181: mosquito version 2.0.14 running

```

Figure 5: A screenshot of our DigitalOcean MQTT server/broker running.

A terminal window titled 'root@ubuntu-Group4: ~' with standard window controls. The terminal output shows the command 'node-red' being executed, followed by a series of informational and warning messages. The messages include the last login time, Node-RED version (v2.1.3), Node.js version (v10.19.0), Linux version (5.4.0-91-generic x64 LE), and details about the settings file, context store, user directory, and flows file. A warning message indicates that projects are disabled. The terminal also displays a notice about encrypted flow credentials and instructions on how to set a custom key. Finally, it shows the server is now running at 'https://127.0.0.1:1880/admin/'.

```
root@ubuntu-Group4: ~
Last login: Tue Dec  7 00:32:45 2021 from 76.69.136.114
root@ubuntu-Group4:~# node-red
7 Dec 00:42:07 - [info]

Welcome to Node-RED
=====

7 Dec 00:42:07 - [info] Node-RED version: v2.1.3
7 Dec 00:42:07 - [info] Node.js  version: v10.19.0
7 Dec 00:42:07 - [info] Linux 5.4.0-91-generic x64 LE
7 Dec 00:42:08 - [info] Loading palette nodes
7 Dec 00:42:09 - [info] Settings file   : /root/.node-red/settings.js
7 Dec 00:42:09 - [info] Context store  : 'default' [module=memory]
7 Dec 00:42:09 - [info] User directory : /root/.node-red
7 Dec 00:42:09 - [warn] Projects disabled : editorTheme.projects.enabled=false
7 Dec 00:42:09 - [info] Flows file    : /root/.node-red/flows.json
7 Dec 00:42:09 - [warn]

-----
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
-----

7 Dec 00:42:09 - [info] Server now running at https://127.0.0.1:1880/admin/
```

Figure 6: Our node-red service running using DigitalOcean.

```
ArduinoCode | Arduino 1.8.13
File Edit Sketch Tools Help

ArduinoCode

/*
 * SOFE 4610U - Final Project
 * Dr. Ramiro Liscano
 * Group 4
 * Shanjay Kailayanathan - 100624670
 * Jana Kanagalingam -100603975
 * Ireni Ruthirakuhan - 100657302
 * Jerusha Macwan - 100723319
 */

#include <ESP8266WiFi.h>
#include <PubSubClient.h>
#include <DHT.h>

#define DHTPIN 5      // Digital pin connected to the DHT sensor
#define DHTTYPE DHT11 // DHT 11

DHT dht(DHTPIN, DHTTYPE);
float t = 0.0;
float tempLimit = 0.0;
float tempThresh = 0.0;
float upperLim = tempLimit + tempThresh;
float lowerLim = tempLimit - tempThresh;
int thresh = 0;
unsigned long previousMillis = 0; // will store last time DHT was updated
```

Figure 7: Some of our Arduino code that was used for the NodeMCU.

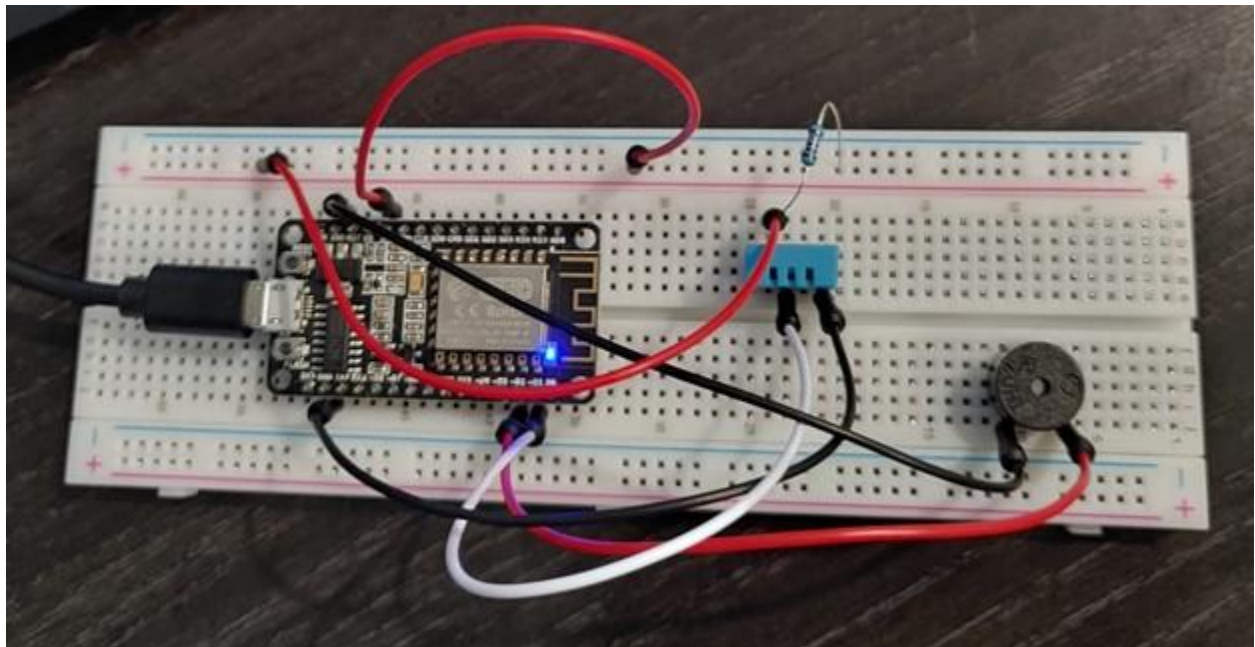


Figure 8: Our NodeMCU, temperature sensor & buzzer.