Project Proposal Report CSE596, Fall 2020
Project Title: Computer Heat Monitor
Teammates:
217CS2029, Berke Biten
217CS2032, Ali Atahan Topal

1 Problem Statement

Computers can overheat sometimes if we use heavy software/programs/games, and we might not always notice this ourselves. We can leave our computers to process a long job like video rendering and we would like to see what our PC's heat status.

2 Current Systems

There are sensors in computer specs that can be monitored by some softwares. But these are only observable by ourselves manually. These don't send notifications when heats are very high.

3 System Overview

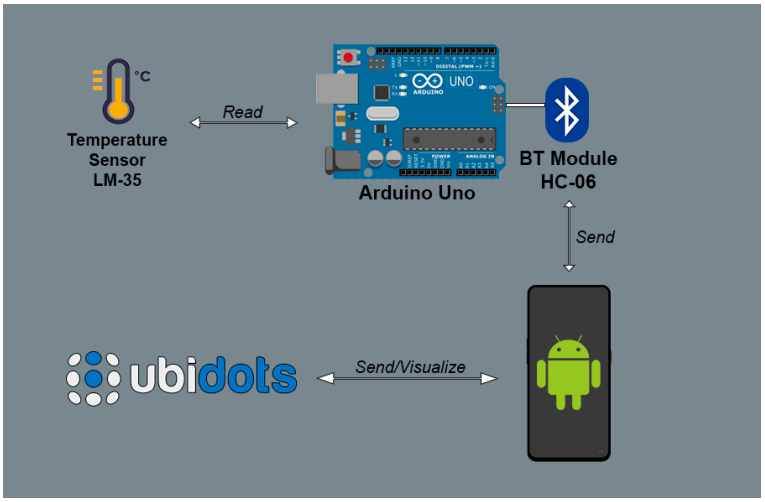

Figure 1: The project overview

4 Hardware Requirements

- Heat Sensor - LM35
- Arduino Bluetooth Module - HC06
- Android Phone

5 Software Requirements

5.1 Use Cases

Use case name **Check Temperature**

Participating Actors Initiated by **User**

Flow of events
1. **User** opens the android application from the phone and clicks to the measure temperature button.

2. **SYSTEM** measures the temperature via Arduino UNO and the temperature sensor and sends the read temperature to the phone via the Bluetooth module.

3. **User** can see the measure temperature from the application.

4. **SYSTEM** saves the measured temperature to the database.

Entry condition **User** has already installed the application to his/her phone.

Exit conditions **User** sees the measured temperature and temperature saved to the database.

Use case name **Send Notification**

Participating Actors Initiated by **SYSTEM**

Flow of events
1. **SYSTEM** measures the heat of the computer in every 2 minutes and saves the measurements to the database.

2a. If the last measurement is above the specified temperature **SYSTEM** sends the temperature to the phone as a notification.

2b. If the last 5 measurement is above the specified temperature **SYSTEM** sends the temperature to the phone as a notification.

2c. If the last measurement is below the specified temperature and the measurement from 2 minutes ago is above the specified temperature, **SYSTEM** sends a notification to the phone that says "Your device has cooled down.".

3. **User** receives the notification and views it.

Entry condition **User** has already installed the application to his/her phone.

Exit conditions **User** views the notification sent by the **SYSTEM**.

5.2 Functional Requirements

The main function of our system is measuring the temperature within a computer. This measurement must be done with a frequency. This measurements data must be kept in a database. Using this data, user must be able to see the current heat of the computer by his/her phone. If temperature stays high for a specified time, system must send a notification to the user's phone. If temperature stays high for a specified time and cools down after, system must send a notification to the user's phone. If temperature stays high for a long time, system must send a notification to the user's phone.

5.3 Non-functional Requirements

- Usability: Any user who has an android phone and the device should be able to use our system.
- Reliability: Our system should be able to start running **whenever needed**.
- Performance: Our system should be able to measure the temperature **every 2 minutes by itself**. System should be able to measure the temperature whenever asked with a **small delay**. Notifications should be sent with a **small delay**

6 Testbed Details

The testbed of our project is essentially a computer. We would like to measure the temperature of the computer to maintain a healthy life for the machine. As you know every machine has a life time and overheating can be a serious problem that decreases a machine's life time considerably. We will test our project with a computer and make the sensor measure the temperature that come out of the fan or directly inside of the computer. We will use a thermometer to test if the sensor is working as intended. We will measure the temperature with both the thermometer and sensor at the same time and compare the results. If both temperatures are the same the test will be successful.

7 Performance Tests

Performance goal for our project is having the least possible delay while sending the measured temperature show up on the phone screen, saving the measured temperature to the database. We will test how many seconds does it take to insert the measurement to the database, how many seconds does it take to see the temperature from the phone screen and if the distance between the phone and the arduino does it changes the delay or not, if the delay is consistent in each measurement and if the temperature sensor measures the temperature correct or not.

8 Project Timeline