**ECE 411  
Project Manager:** Andrew Greenberg  
**Team Members:** Lionel Diaz, Noah Harvey, Binh Phan, Tyler Seitz and Matthew Walters  
November 17, 2016

**UML Physical View**

**Refrigerator**

**Server**

**User**

Database

Pc/Workstation

Computer / Workstation

TCP/IP

TCP/IP

**Internet**

Temp. Sensor

Monitor

Control module

Web Server

**Assume**: User can control system by using PC or via the internet

**UML Class Diagram**

Assuming system has a lot of sensor, 1 monitor and 1 control module.

Temp. Sensor

**UML Use Case**

- Current Temp.

- Current Temp.

- Sensor Id

- Sensor Id

- Current Temp.

- Sensor Id

- Current Temp.

Receives

Sends to

1

1

1

\*

\*

1

Control Module

Monitor

Database

|  |  |
| --- | --- |
| **Use-Case** | Adding temperature sensor |
| **Actors** | User, Server, Database and Router |
| **Description** | * In the beginning, power is supplied for temperature sensor module. User will press WPS button on the router and the sensor module. This will allow the sensor to automatically connect to the users WiFi internet. Once connected the will push the setup button causing the status LED to blink. This button tells the unit to tell the server that a user is setting up a new temperature sensor. * In this time, the user will read the unit id number off of the sensor module and input it into the server along with their desired temperature setting. In the server module, it will send a new status value to tell the unit that the “setup” was successful. Upon receiving the new status, the sensor module sets the LED to a solid state to indicate to the user that they successfully added the unit. After this process, temperature sensor will send data packages including current temperature, unit ID, current status and current temperature setting to the server module. In the end, the server module will store the data containing the unit ID and its respective current temperature in the database when valid data is seen by the server. |
| **Stimulus** | Connecting the sensor into the refrigerator for regulating temperature to maintain acceptable temperature for vaccine storage. |
| **Response** | For the stimulus information, with the current temperature confirmation, a corrective action is maintained for acceptable temperature or changed to acceptable temperature range. |

**HTTP GET STRING**

* **Assumption:**
* No real-time clock on sensors and monitor.
* Resolution temperature: 0.1˚ C.
* Min: -20 ˚C and Max: 50 ˚C, Δ = 70 ˚C.
* 700 points needed, therefore a minimum of a10-bits are needed if unsigned.
* Current temperature will be sent as a12-bit 2’s complement signed number.
* Temperature = 12-bit 2’s complement signed number for representing the measured temperature.
* Status = 4-bit binary number, allowing for 16 unique statuses. May be used for error conditions or other settings in the future.
* Unit ID: 1 million units => 20-bit number. Setting this to a higher than likely needed number allows for some of the bits to represent a revision or model number of the sensor module if needed in the future.
* Temperature Setting = 12-bit 2’s complement signed binary number for the desired temperature.
* **Assign values:**
* **Value 1 = 12-bit signed 2’s complement binary number //**Measured temperature
* **Value 2 = 20-bit unsigned binary number //**Unit ID
* **Value 3 = 4-bit unsigned binary number //**Status
* **Value 4 = 12-bit signed 2’s complement binary number //**Temperature setting
* **Write HTTP URL:**
* Get data from temperature sensor to database:

http://tempmonitoring.com/tempcheck?temp=value1&ID=value2&status=value3&tempset=value4

* Get data from the server to the temperature sensor:

**Assume**: Controller ignores the “temp” value sent from server to sensor module. This allows the transmitting and receiving http strings to maintain the same data structure on the sensor module.

http://tempmonitoring.com/tempset?temp=value1&ID=value2&status=value3&tempset=value4

**UML Interaction View:**

* Assuming the sensor will periodically wake up every one hour to sense temperature and send to the database. In the other case, the sensor also wake up to sense temperature when the refrigerator is open**,** then send the data to database so that server can control monitor to adjust the temperature inside.

