**Pulse and temperature reader**

* **Electronic Measurements and Systems project -**

**Coordinating teacher: Holonec Rodica**

**Top Alexandru – Group 30421**

**Stancu Gabriel - Group 30422**

**Goal:**

* Record BMP and temperature;
* Send data to a device connected wirelessly to be processed;
* Based on data make predictions or signal alerts in case there are dangers to health.

**Microcontroller:**

* Arduino Leonardo 32u4

**Sensors and Actuators:**

* Temperature sensor TMP36;
* Pulse sensor from pulsesensor.com;
* HC-06 Bluetooth Module.

**Code description:**

The code initializes the Bluetooth communication and, during a loop, it reads data from the analogic pins 0 and 1, where the temperature sensor and the pulse sensor are connected. For the temperature sensor it uses a formula which transforms the analog signal to a temperature reading in Celsius.

float voltage = reading \* 5.0;

voltage /= 1024.0;

float temperatureC = (voltage - 0.5) \* 100 ;

The pulse sensor reads heartbeats constantly and returns the BPM which is measured as average of the last 10 measurements. If there are enough measurements taken and the pulse sensor detects a heartbeat when it passes through the loop then it sends the data from it and the temperature sensor over Bluetooth to the application.

The pulse sensor may return incorrect values if the pressure applied is not constant or there is movement between the skin and the sensor. The sensor is recommended to be mounted on the ear lobe or using velcro on the tip of a finger.

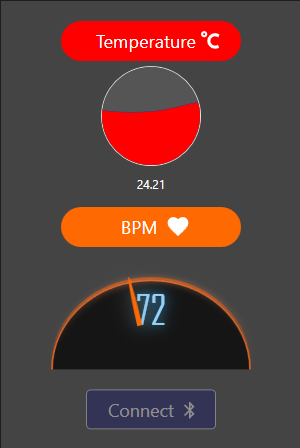
These are just a few sensors, Arduino is a modular platform so depending on the chosen board you can add multiple sensors such as accelerometer to detect falls, EKG sensors and many more.

The Arduino code which takes the reading and creates custom packets to be sent to the application over Bluetooth.



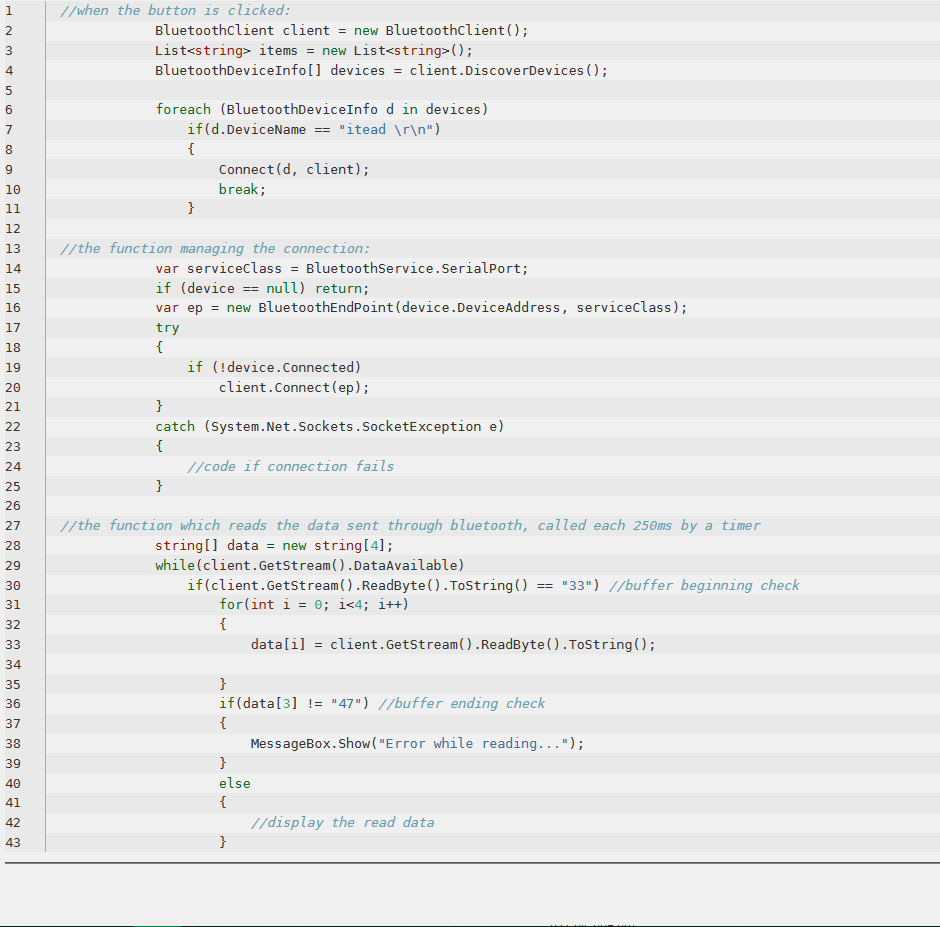
The application is built using WPF (Windows Presentation Foundation) in Visual C#. It reads the bytes every 250 ms. This application is just an example but it can be adapted into a health bracelet application or an application for medical purposes where it gives alerts in case there are dangers to a person’s health.

The user interface is displayed below:



The first graphical display element shows the body temperature with a wave-like progress bar, just like the common thermometers! The second graphical display element shows the BPM value, under the design of an accelerometer. The connect button establishes the Bluetooth connection between the application and the Arduino board (if not already established), and once established, the button becomes inactive, until the connection is interrupted.

The code that ensures the communication between the Arduino board and the user interface is listed below:



**Conclusion**

The state of the application presents an early version, with the basic, but the most important features implemented. In later versions we are going to expand the interaction possibility of the user, by adding more measurement functionalities, connect it to a local database in order to allow both the user to track his data over time, but also for our data collection phase, so that we will be able to calibrate the sensors better and better as the application is used.