University of Pittsburgh

Department of Computer Science

CS 2310:
Multimedia Software Engineering

SIS Healthcare System For Senior Squash Players

Final Report

Done by:
Rakan Alseghayer (raa88)

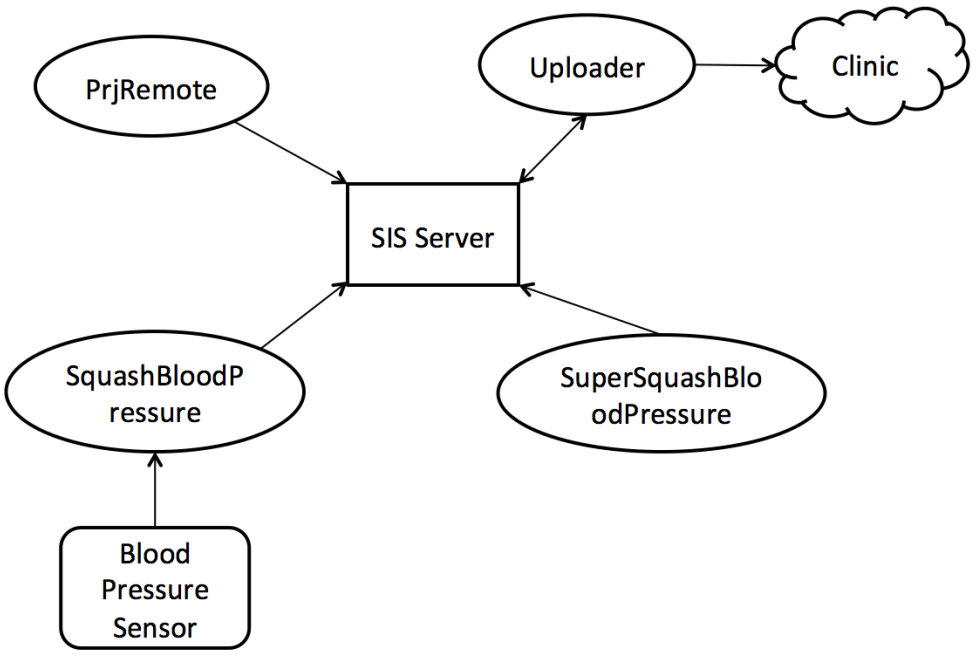
- **Introduction:**

In this project, I attempt to learn more about the SIS test bed by extending Exercise 4. The idea that I have is related to my passion sport Squash. Squash is a very intense sport, and also, very popular amongst senior citizens. The doctors always recommend that a patient do a blood pressure reading 4 times a week to be in the safe side. In this project, SIS will be an excellent solution.

- **Overview:**

The process of taking the blood pressure measurements can be automated easily through the SIS healthcare system. Instead of going to the clinic 4 times a week, a player can have a sensor to measure that at home. However, the player has to notice and take care of the abnormal readings and the false positives. Also, they have to report manually to their doctors about that. Hence, the SIS healthcare system can be a perfect solution through setting and connecting the sensor and do the reading automatically 4 times a week. If the reading was abnormal, for 4 consequent times for a single reading, then the system should display a message to prohibit the player from playing, and (preferably contact the doctor's office for an appointment).

- **System Components:**



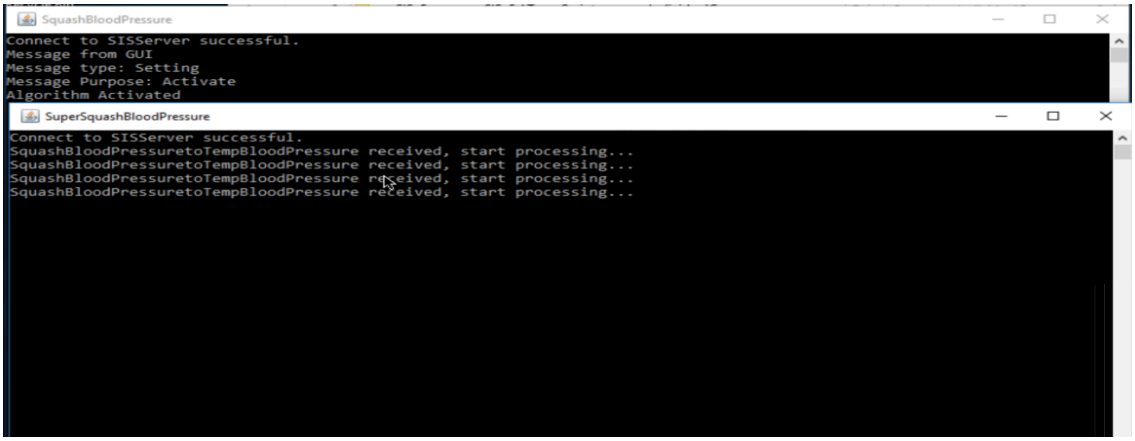
These components are as follows (two of which are implemented and demonstrated):

1. SIS Server:
   It is the heart of the system. A centralized server that gathers all the components of the system, all the messages are passed through it, and all the communication happens through it.
2. SquashBloodPressure: (implemented + demonstrated)
   Is the component that is supposed to get the readings from the sensor, and then pushes them as alert messages to the super component "SuperSquasBloodPressure"
3. SuperSquashBloodPressure: (Implemented + demonstrated)
   The super component that should process the readings received from the basic components "SquashBloodPressure". It checks if the readings are abnormal for 4 consecutive readings, it displays a warning that stops the player from playing squash.
4. Uploader: (NOT implemented neither demonstrated)
   A very helpful component in this system. I t would receive a message from the super component and send an email to the doctors clinic to schedule an appointment, or to flag a reading for the medical history of the player.
5. PrjRemote: (demonstrated)
   Used to debug the super component by sending an abnormal alert as if it was sent from the basic component "SquashBloodPressure."

- **A scenario in screen shots:**

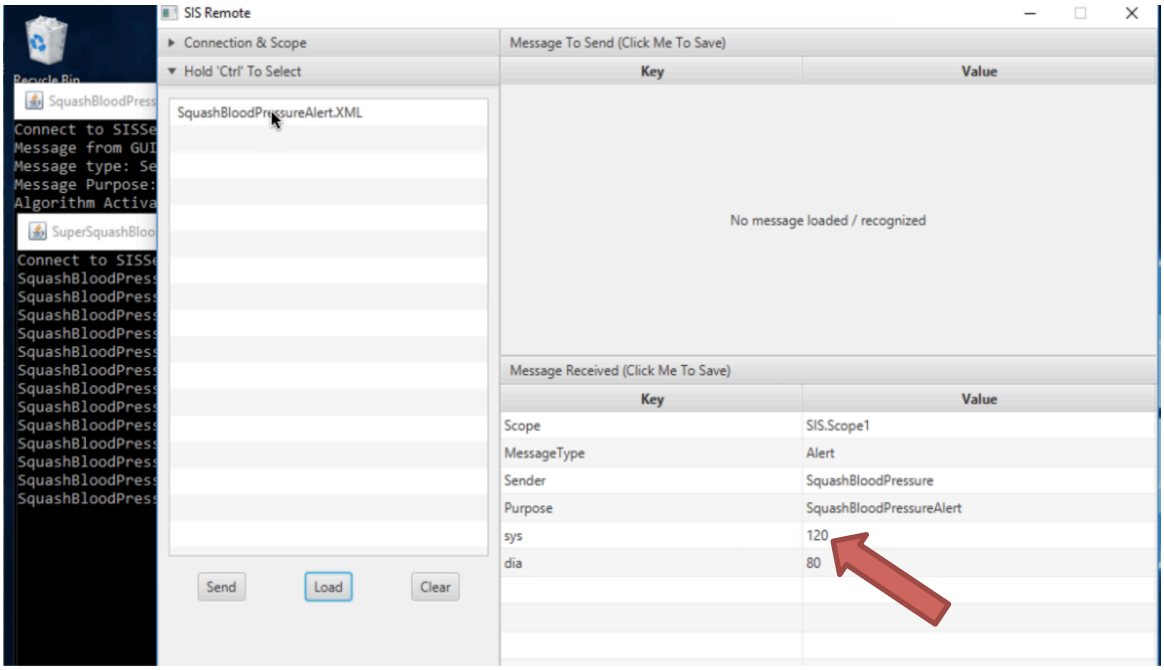
The following is a sequence of screen shots that demonstrates a scenario of the Squash context usage under the SIS healthcare system.

Initially, running the server is the first thing to do, so that step is ignored as it is shown in the video. Starting from the components firing up, we fire up the Super component "SuperSquashBloodPressure", and then we run the basic component "SquashBloodPressure":

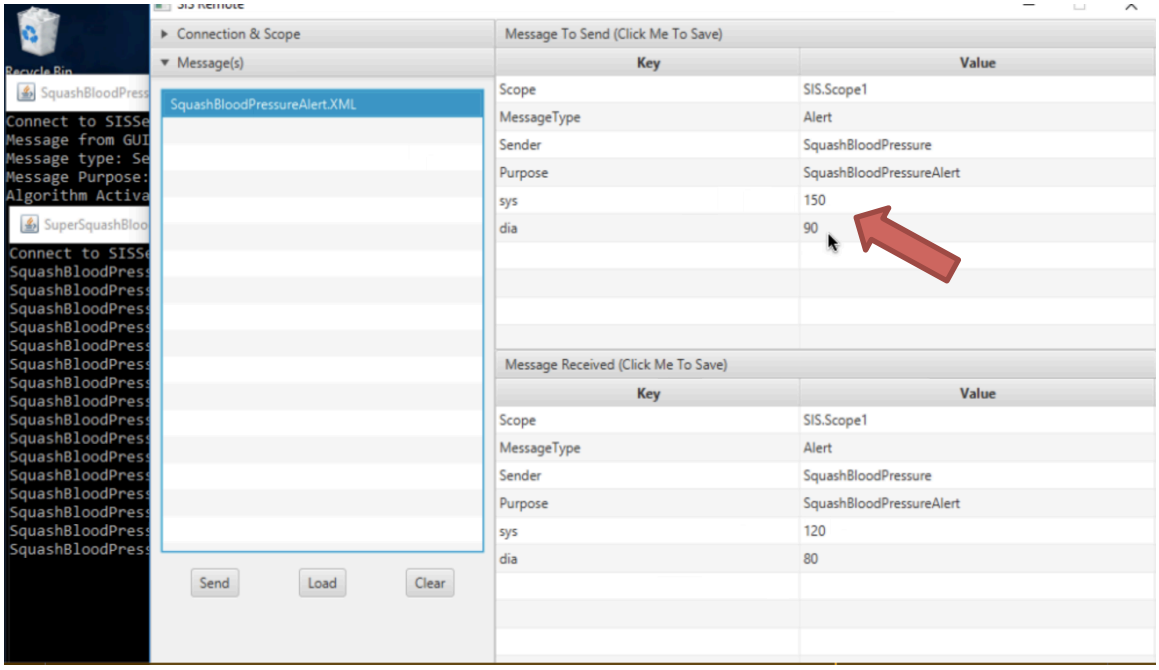


We notice that the super component is running and receiving the readings from the basic component.

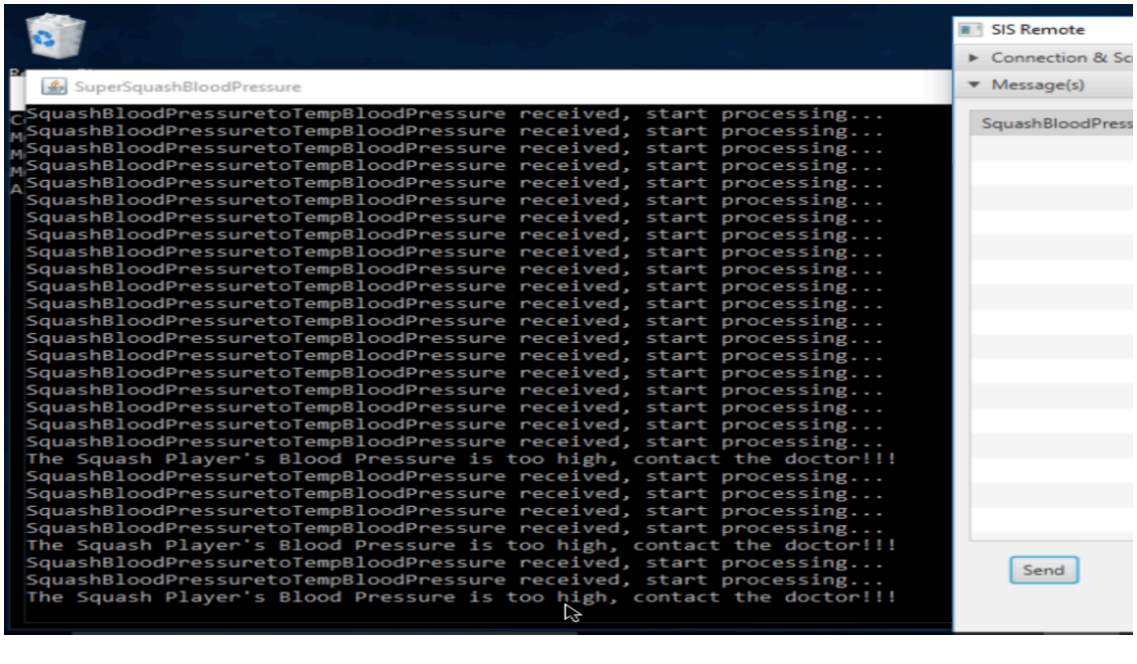
Now, in order to simulate the abnormal error, we use the PrjRemote, running the PrjRemote will show the normal "simulated reading from the basic component as shown next:



Now, we load the message that contains the alert. We notice the values in the message:



We send it to the super component. This will result in doing nothing for 4 consecutive times. After that, the alert will be generated each time the reading is present. Shown below:



- **The Video Demonstration:**

In order to watch a video demo of the system running, please check the following link: http://screencast.com/t/p537GOwm6dr4

- **Conclusion:**

After doing this project, I feel now that I have gained both the theoretical knowledge from the Social model, as well as the practical familiarity with the SIS healthcare system structure and implementation. About the blood pressure sensor, I tried to connect the sensor to my machine through the virtual machine. However, I have received different kinds of errors. Although I have included all the Java files and installed the driver for the sensor, the process resulted always in failure. Hence, I was forced to simulate it through the PrjRemote.

- **References:**

[1] Professor SK Chang webpage lecture on the SIS Test-bed.