Partial Report

Project Title: Prescient: Context-Aware Home Monitoring and Control Central
ID JEMS: 155008

Students: Rodrigo Schmitt Meurer
Rafael Eriberto Mariot Scarduelli
Daniel Manzoni Seerig

Professor: Antônio Augusto Medeiros Fröhlich

University: Federal University of Santa Catarina

Declaration of Responsibility

We, students from project ID 155008, hereby declare that the provided information in this report is true, under supervision of our professor who signed below.

Professor's name: Antônio Augusto Medeiros Fröhlich

Professor's signature: _____

Date: 04/06/2016

Proposed Scheduled

Date	Task
04/04	Send proposal
20/05	Receive board
20/05	Start documentation
20/05 - 10/06	Development of Galileo's case with board sensors
20/05 - 25/05	Board Sensors Implementation and Tests
25/05 - 30/05	Integration of Galileo with LISHA's Smart Room
01/06 - 10/06	Integration Validation
10/06 - 15/07	Application Development for smart devices control
15/07 - 20/07	Implementation and Evaluation of user on network discovery
20/07 - 30/07	Interface Galileo Gateway with Web and Database
01/08	Start development of context-aware algorithm
21/08	Start development of data visualization in hybrid application
20/09	Finish data visualization in hybrid application
20/09	Send partial report
05/10	Conclude context-aware algorithm
06/10 - 16/10	Final Tests, bug corrections and implementation validation
20/10	Prepare for presentation
20/10	Finish documentation
30/10	Submit final report
01/11 - 04/11	Presentation at SBESC

Update Scheduled

Date	Task	Percentage
04/04	Send proposal	100 %
20/05	Receive board	100 %
20/05	Start documentation	25 %
22/05 - 30/05	Development of hybrid app base graphical interfaces based on HTML, JavaScript and Jquery Mobile	100 %
01/06 - 05/06	Integration and validation of the graphical interface with charting libraries for data visualization	100 %
05/06 - 07/06	Study of board capabilities and embedded operating system choice	100 %
07/06 - 10/06	Development of smartphone detection daemon on PC	100 %
08/06	Technical Webinar	100 %
10/06 - 20/06	Ubilinux installation and setup on Galileo	100 %
20/06	Setup subnet and wireless router to test the daemon on Galileo	100 %
21/06	Primary tests for the smartphone detection in Galileo (Proof of Concept)	100 %
22/06 - 30/06	First tests with Galileo and EPOSMoteIII (attempting to control LISHA's Smart Room lights)	100 %
01/07 - 10/07	Evaluation of desired sensors to attach to Galileo and EPOSMoteIII	100 %
10/07 - 12/07	Instalation and tests with wiringx86 for Galileo Gen2	100 %
13/07 - 25/07	Sensor Tests on Galileo Board with wiringx86	80 %
30/07	Start development of context-aware algorithm	30 %
01/08 - 08/08	Communicate hybrid application with python in Galileo through Flask API	80%
08/08 - 12/08	Performance evaluation for the user detection daemon	100 %
24/08	Technical Webinar	100 %
01/09 - 10/09	Construction of Galileo and EPOSMoteIII Case	90 %
11/09	Galileo and EPOSMoteIII instalation on Case	100 %
12/09	Design printed circuit to handle the sensors and other peripherals	20 %
15/09	Start construction of Smart Room Model for the Demo at SBESC	15 %
20/09	Partial report submission	100 %

20/09	Reinstall AC remote control	-
08/10 - 10/10	Finish training machine learning algorithm	-
11/10	Validate results and overall implementation	-
11/10	Record Video for presentation	-
12/10 - 19/10	Finish demo model of the smart room	-
20/10	Prepare for presentation	-
20/10 - 28/10	Finish documentation	-
30/10	Submit final report	-

Difficulties and Workarounds

Situation 1

In our project we proposed to identify the user in the smart room by the MAC address of it's cellphone. To do that we chose to use the NMap tool that is able to show all the MAC addresses of the devices connected to the network. At first we tried using the system call from the arduino IDE to use linux's functions.

Solution 1

Instead of using the linux image we chose to use an alternative embedded linux distribution based on debian, called ubilinux. ubilinux is an embedded Linux distribution from Emutex, based on Debian Jessie. It is targeted at embedded devices that have limited memory and storage capabilities. This Linux distribution uses the same package manager as Ubuntu, Debian and other distributions, the apt-get. Therefore this distribution has access to a much wider range of packages available for linux. This way we were able to sucessefully use the Nmap tool, thus enabling us to detect the smartphone in the network.

Situation 2

After installing Ubilinux in the galileo board we were unable to program the GPIO pins of the board or program anything via the arduino interface.

Solution 2

There is a python module that allows python to access the GPIO pins of the Galileo board. This library is developed by Emutex and provides a simple and unified API to talk to the GPIO pins in the Intel Arduino Capable boards. Up to now this module supports: Writing to a GPIO pin configured as output, Reading from a GPIO pin configured as high impedance input, Reading from a GPIO pin configured as pullup input, Reading from a GPIO pin configured as pulldown input, Reading from a GPIO pin configured as analog input (ADC), Writing to a GPIO pin configured as analog output (PWM). Since our project does not use any further capabilities

Situation 3

On our first try to detect the smartphone with a certain MAC address connected to the network we implemented a bash script on a PC that uses the Nmap tool to verify if this certain address is connected or not to the network. In the PC we did not encounter any performance related problems. However a PC processor is many times faster than a embedded such as the one used on the Galileo Board. Since user detection is a process that must be running with high frequency to detect if the user is present of left the network, when running it on Galileo it

makes the processor busy for much time, reducing the overall performance of the system.

Solution 3

Situation 4

machine learning framework

Solution 4

Keras and tensorFLow