**Memo**

To: Professor Alan Pisano, Professor Osama Alshaykh, Fulya Ekiz Kanik

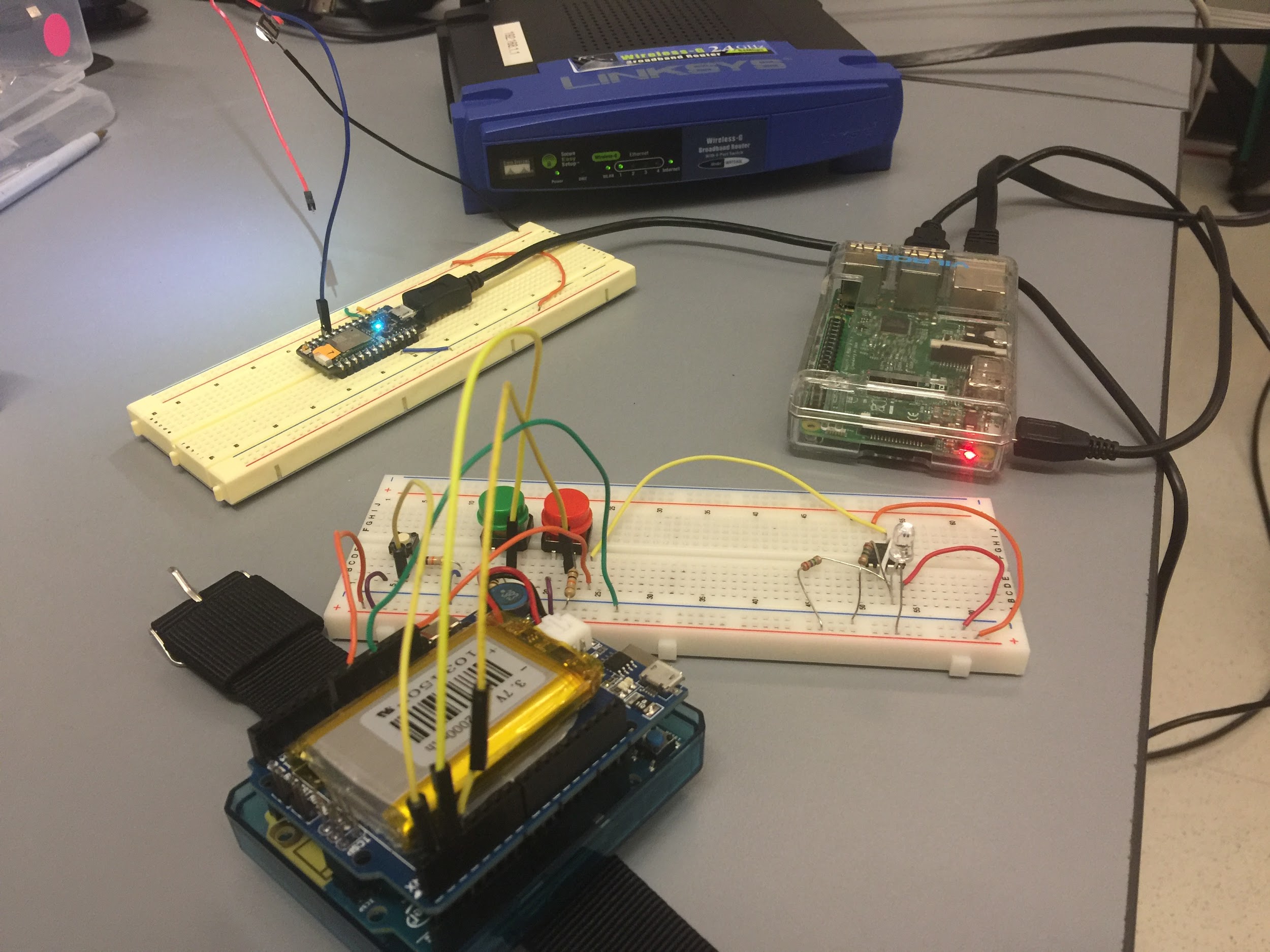
From: Caroline Jones, Michael Haley, Sneha Pradhan, Tanatsigwa Hungwe, Zehua Zhao

Team: 1

Date: March 31, 2017

Subject: **Luminesense System Functional Deliverable Test Plan**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Figure 1: View of all system components including raspberry pi, router, luminaire, and wearable*

1. **Customer Requirements**
   1. Overall

The customer has requested a gesture-based system with which a user can exert control over “luminaires,” which are IP-enabled LEDs used in the customer’s research. The system must have two modes: an adaptive mode and a gesture mode which switch control seamlessly. The system must be compatible with the customer’s existing systems, including their luminaires and the way that they have set these up in their lab. The wearable itself used to detect gestures should be responsive and be able to be scaled down to a more compact form. The user must be able to effortlessly interact with the system not just when issuing commands but also when viewing their energy usage data and customizing their system including gestures. This interaction is accomplished with a web application.

* 1. Adaptive Mode

The adaptive mode should take control of the system when a user first enters the room or when all users exit the room. The adaptive mode should use the customer’s positional tracking systems in order to detect the presence of a user in the room. When a person enters the empty room, the adaptive mode should register this and turn the lights on in the room. When all people exit the room, the adaptive mode should recognize that nobody is still present and turn the lights off.

* 1. Gesture Mode

The gesture mode should give users the ability to use gestures to change the state of the luminaires in the room. The wearable itself must be worn on the wrist of a user and should respond to their gesture commands. Individual and groups of luminaires should be able to be selected with the wearable which respond promptly to a gesture command issued by the user. Additionally, the gesture mode must be accompanied by a gesture library - a tome which details all available gestures and commands that said gestures execute. The user must have the ability to customize their gestures by redefining the gesture gesture library to their choosing.

* 1. Web Application

The web application gives the user more control over their system. Using the web application, the user must be able to get interact with the system - this includes viewing energy usage reports, gesture commands customization, and sending directly to the luminaires.

1. **Significance Of Deliverable In Meeting The Totality Of Customer Requirements**
   1. Overall

This functional test demonstrates the final product that will be integrated with the

customer’s existing systems. This test will be an end-to-end demo wherein we will showcase the methods and tools needed to fulfill the requirements set by the customer. The test will be conducted by focusing on a typical use case. This approach will show the improvements made in the current over previous design existing solutions and prove the technical validity of our design.

* 1. Adaptive Mode

The adaptive mode will detect whether there are people in the area and alert the system accordingly. Users will be able to be detected by the single pixel cameras that are installed in the room. Now the users can walk in the room and they will be perceived by the single pixel cameras then the luminaires will turn on. When users leave the room, luminaires will turn off since cameras don’t detect any people in the room.

* 1. Gesture Mode

The gesture mode gives the user a direct way to issue commands to the luminaires in the room. The user does this through the use of a wearable, which was a key customer requirement. The wearable itself is primarily made up of an Arduino 101 board which has the Intel Curie onboard. One of the customer’s requirements regarding the wearable is that it should be able to be scaled down to a more manageable size in the future. The Intel Curie chip can be integrated into a more compact board for a future wearable design, which satisfies this requirement. The gesture mode itself is responsive to gestures that the user completes with no latency. Additionally, the user is able to select one or multiple lights through the use of a button. This satisfies the customer’s requirement to give the user intuitive gesture-based control over individual and groups of luminaires.

* 1. Web Application

The website application that accompanies the Luminesense system is live at

luminesense.herokuapp.com. This fulfills the customer’s request to give the user more control of their system beyond using gestures as well as to track their energy usage. The main functions of this application are the energy usage information and the ability to issue commands to the system and choose gesture preferences. The energy usage graph displays the energy usage of the system over time, with kilowatt-hours on the vertical axis and time on the horizontal axis. Using this, users can track their energy usage over time, which was a key customer requirement. Users can also issue commands to luminaires on the system through the web application, and change their gesture preferences for commands. This allows the user to interact with the luminaires if their wearable is not operable for some reason. This fulfills the customer’s request to give the user more fine grained control over their system and to give users the ability to customize their gesture preferences. Moreover, the users can directly control the lights with our UI.

1. **Equipment and Setup**
   1. Overall

This functional deliverable test showcases an end-to-end demonstration of the system. The test will be performed in the customer’s lab which contains the positioning system and the luminaires. It is necessary to perform the test in the lab to show that we have satisfied the key customer requirement of integration with the customer’s existing technology and systems. A typical use case will be shown wherein a user enters the room and the system detects their presence using the adaptive mode. The user then selects lights and issues commands to them. Finally, the user interacts with the web application to view their energy usage, change their gesture preferences, and issue new commands to the system.

* 1. Adaptive Mode

The basis for adaptive mode are the 11 single pixel sensors (indexed 0-11 excluding 4) mounted in SLURP. A Raspberry Pi B is also mounted in the lab and contains a driver which polls each sensor’s red, green, blue, and clear segments approximately 10 times per second. An algorithm using python’s parallel processing support runs on the Pi’s quad-core ARM CPU and detects occupancy status based on the moving average of each sensor’s poll values. An MQTT Broker protocol is used to run the drivers/programs necessary from a host machine.

* 1. Gesture Mode

The gesture mode depends on the communication loop between the wearable, the

raspberry pi, and the particle photons on the luminaires. The wearable consists of an Arduino 101, a battery shield, and a photodiode. The person performing the test will use the wearable to select the luminaires that they wish to interact with through a button press, and then perform a gesture that will change the state of the luminaires.

* 1. Web Application

The web application, hosted on Heroku, will be accessed through the user’s computer. The user will first view their energy usage trends. These are calculated on the Raspberry Pi and are periodically pushed to a postgreSQL database. When the web application page is loaded, it loads, parses, and performs calculations on this data in order to display accurate energy usage information. The user then navigates to the gesture page, where they can issue a command to the luminaires. This is achieved by sending the command over the particle cloud api. The user can then customize what commands their gestures correspond to. The user’s gesture preferences are stored in a database on the cloud, which is then read by the raspberry pi. The Raspberry Pi then pulls these preferences and makes decision making on the gestures based on this information.

1. **Data Collection**
   1. Overall

The overall integrity of the system can be seen through the successful operation of the system as a whole, from initial user interaction to the end result seen either on the web app or on the luminaires. Beyond this, most of the data collected regarding gesture sensing and occupancy sensing will be printed to the terminals of the respective hardware.

* 1. Adaptive Mode

When the Raspberry Pi sensor driver is run with provided parameters, is begins streaming the current RGB values for each sensor into a .txt file, along with additional information in regard to timing. If the driver experiences problems while sampling, these issues will be reported in the console (via print statements) The algorithm will output appropriate print statements (“e.g. OCCUPANT DETECTED AT SENSOR X”) once the sensors detect active occupants in the room, or will output a different message if the room is empty.

* 1. Gesture Mode

When the wearable is used to select a luminaire or is used to perform a gesture, it

outputs a detailed log of information to the arduino’s serial port which can be viewed in the console in the arduino IDE. This output specifies the luminaire selected and the gesture performed. This information is also logged to the console by the raspberry pi, so checking these two statements against each other can show that the correct data was sent over bluetooth. By checking this against the actual command that is executed on the luminaires, one can verify the integrity of the gesture mode.

* 1. Web Application

The web application prints relevant information to its console, which can be viewed through the heroku logs if printed through the server code or through the browser’s console tab if printed in the html code. Connecting to the database can be confirmed through printing the database connection information and by printing the data that was pulled from the database regarding energy usage. When issuing a command to the luminaires through the web app, it is apparent that the luminaires respond to the command. Additionally, this information, including the command and the request, is printed to the console. Finally, the output of the gesture preferences is also printed to the console and the database adjustment that results from this is shown on the console as well.

1. **Measurable Criteria For Success**
   1. Overall

The overall criteria for success for this project is both adaptive mode and gesture mode can work together, meanwhile the two modes can be used at the same time without conflicts. Every command should be responded accurately by each component in this project. The user should also be able to intuitively interact with the system and the system should respond promptly with an appropriate reaction to the user’s actions.

* 1. Adaptive Mode

The system should detect and respond to the presence of active room occupants or new entrants within without a discernable delay Additionally, to contrast this, the system should find no occupants when the sensor area is empty, and turn off the luminaires promptly in response to the lack of a presence of users.

* 1. Gesture Mode

All the luminaires should be turned on when the session button is pressed, which means sessions begins, in three seconds. All the luminaires selected by pressing the selection button should all obey the gesture performed by the wearable device. When the end-session button is pressed, all the luminaires that have not been selected will return to a default state.

* 1. Web Application

The web application will measure the time that the lights have been off when the adaptive mode is not activated. The time period counting will be pulsed whenever someone enters the room or the gesture mode is activated. All the actions that have been performed by the Raspberry Pi can be printed as log which will proof the accuracy of the web application.