EE/CPRE/SE 491 - sdmay29
Automating Inventory Management & Routing through Sensor Networks
Week 8 Report
10/24/18 - 11/4/18
Client: Jimmy Paul
Faculty Advisor: Goce Trajcevski

Team Members:
David Bis - *Meeting Facilitator*
Hanna Moser - *Meeting Scribe*
Adam Hauge - *Report Manager*
Sam Guenette - *Public Relations*
Ben Gruman - *Resource Acquisition*
Noah Bix - *Documentation Manager*

Past Week Accomplishments

This reporting period was spent improving our Project Plan and building our first proof-of-concept prototype. All team members contributed to working on the project plan which was posted to the team website on October 26th after receiving feedback from the senior design TAs as well as from our client and faculty advisor. Major progress has been made on our prototype as preparations have been made to present it to our client as well as the rest of the senior design class.

Project Plan V2 - All

- Revised Project Plan based on feedback from TA, Adviser, and Client
- Added new information to Project Plan based on maturation of team and project
- Report posted to team webpage

Improved API for Data Visualization Front-end - David, Sam

- Added an endpoint to the API that gets data from the database to view pertinent data regarding stockroom inventory
- Wrote SQL script that joins product data and monitoring device data
- Endpoint performs processing on data to determine if a product is below a specified threshold
- Added datatable and registration component that monitors and and groups devices by product being monitored
- Began setting up the ordering component architecture that interacts with corresponding data-sets and saves an order to the database.

Barcode Scanner and Weight Sensor Update Database - Adam, Noah, Sam, David

- Wrote trigger for Raspberry Pi to detect when barcode scanner detects barcode
- Update database from Raspberry Pi based on scanned barcode
- Sample weight sensor to update database from Raspberry Pi

Weight Sensor - Ben

- Wrote driver to gather data from weight sensor
- Configured and soldered physical connections between load cell, amplifier/ADC, and Raspberry Pi pins

Sensor Network Configuration - Adam

- Configured socket system to allow for communication between networks
    - Master raspberry pi runs server.py script
        - Script continuously waits for requests to connect with clients
        - Clients make connection with server, send data, and disconnect
    - Sensors run client.py script
        - Script makes connection with server, packs data, sends packet, then disconnects
        - Unlike server, client script ends after data has been sent
        - Script only works for barcode scanner at the moment, script will be adapted for weight sensor later
- Implemented System to prevent timeout and mismatched data
    - Clients are designed to disconnect as soon as data is received by server
    - Clients are designed to wait until server is ready for a connection
- Created packet system to transmit data efficiently
    - Packets designed to display their sensor type, as well as their allocated data

Frontend Design and Development - Hanna

- Implemented data display page
    - Pulls data from database for each of the products at a specific location and displays following fields for each individual product:
        - Product ID
        - Product name
        - Threshold
        - Current level
        - Reorder status
- Received feedback on latest screen sketch versions for data display and sensor registration, and updated screen sketches accordingly

Pending Issues

- **Barcode Sensor** - All
    - We are unsure of what purpose the barcode sensor is going to provide at this point
        - Using the barcode sensor for monitoring inventory may be inefficient
        - The scanner may be used for Crafty’s employees to register a new product
        - The barcode scanner idea might be scrapped altogether

- **Weight Sensor** - Adam
    - Weight sensor needs to send constant updates to server
    - Script has not been created to handle weight sensor yet
- **Network Setup** - Adam
    - Setting up the network hardware requires that each unit knows the IP address of the raspberry pi
        - Need to implement a system that allows for a static IP to be used or one that allows the IP address to be discovered automatically.

Plans for Upcoming Reporting Period

- **Implement API for Registering Sensor Devices** - David
    - Write endpoints for adding new sensor devices to the sensor network
    - Include endpoints for editing existing sensor devices, such as changing what product they are monitoring
    - Implement Ordering component
        - Pull information corresponding products in a company's pantry and produce an order based on current levels.
- **Sonar Sensor Research** - Noah
    - Research and select sonar sensor
- **Frontend Design and Development** - Hanna
    - Continue to update screen sketches as per suggestions from teammates and client
    - Prettify the data display page by displaying the data for each of the individual products in separate boxes that are color coded
        - Color coding is indicative of current level status
            - Red: has reached threshold and needs reordering
            - Yellow: close to reaching threshold
            - Green: far away from reaching threshold
    - Get the skeleton code for the device registration page implemented

Individual Contributions

Team Member	Contribution	Weekly Hours	Total Hours
David Bis	Project Plan V2 Improved API for Front-End	15	61
Hanna Moser	Project Plan V2 Created Simple Data Display Page Continued Altering Screen Sketches	15	51
Adam Hauge	Project Plan V2 Barcode Scanner and Weight Sensor to Database	15	60

	Sensor Network Configuration		
Sam Guenette	Sensor communication programming Added Database Refactoring & Optimization Company Registration Component Automated Ordering backend	12	60
Ben Gruman	Weight Sensor Prototype Project Plan Editing	10	36
Noah Bix	Barcode Sensor Prototype Project Plan V2 Python Programming	10	51

Gitlab Activity Summary

```
---------------------------------------------------------
Action: closed, Fri Nov 2 2018
Author: guenette
Title: Add Dummy Data
---------------------------------------------------------
Action: pushed to branch master, Wed Oct 31 2018
Author: Ben Gruman
---------------------------------------------------------
Action: pushed to branch master, Wed Oct 31 2018
Author: Ben Gruman
---------------------------------------------------------
Action: pushed to branch master, Wed Oct 31 2018
Author: Ben Gruman
---------------------------------------------------------
Action: pushed to branch master, Wed Oct 31 2018
Author: Ben Gruman
---------------------------------------------------------
Action: pushed to branch sockets, Tue Oct 30 2018
Author: ahauge
---------------------------------------------------------
Action: pushed to branch sockets, Tue Oct 30 2018
Author: ahauge
---------------------------------------------------------
Action: opened, Mon Oct 29 2018
Author: ahauge
Title: Assign static IP address
---------------------------------------------------------
Action: pushed to branch master, Sun Oct 28 2018
Author: hjmoser
---------------------------------------------------------
```

```
Action: pushed to branch sockets, Sun Oct 28 2018
Author: ahauge
------------------------------------------------------
Action: pushed to branch sockets, Sun Oct 28 2018
Author: ahauge
------------------------------------------------------
Action: pushed to branch master, Sun Oct 28 2018
Author: dsbis
------------------------------------------------------
Action: pushed to branch sockets, Sun Oct 28 2018
Author: ahauge
------------------------------------------------------
Action: pushed new branch sockets, Sun Oct 28 2018
Author: ahauge
------------------------------------------------------
Action: pushed to branch master, Sun Oct 28 2018
Author: dsbis
------------------------------------------------------
Action: pushed to branch master, Sun Oct 28 2018
Author: dsbis
------------------------------------------------------
```