Sprint 3 Plan
Product Name: Smart-Irrigation Backend
Team Name: Wubalubadubdubs
Sprint Completion Date: November 30, 2015

Goal:

The goal of this sprint is to finalize our core API system for the frontend use. In our finalization, we will include authentication for the frontend users so that only then can access the API with their given username and password. We are also going to include a few more features in our API that the frontend would like such as tracking the gallons per minute used by the farmers. The frontend will use this information to create visual graphs to the end user so that they can better understand their watering patterns. We also would like to simulate the sensors aggregating their readings to the PI.

Task Listing Organized by User Story

1. 13 pts: As a developer, I want to integrate our API work with the frontend.

   Task 1 : 8pts: Coordinate with the front end and answer any questions they have about the API
   Task 2 : 2pts: Provide examples of API calls to the front end.
   Task 2 : 3pts: Determine with the front end that everything they need is included by the API. If not, then ask them what is missing and implement these missing features.

2. 8 pts: As a developer, I want to implement security into our API so that only the frontend has access to the information on the database.

   Task 1 : 3pts: Create a database table that holds the user names and passwords of those who can use the API.
   Task 2 : 2pts: Secure the passwords on the database by salting them.
   Task 3 : 2pts: Research on Flask HTTP authentication and add it into the API.
   Task 4 : 1pt: Test HTTP authentication.

3. 5 pts: As an administrator, I want to transfer the database from AWS to a SOE server.

   Task 1 : 3pts: Receive permission from SOE to set up a server on their end.
   Task 2 : 1pt: Change config file so that database is going to SOE instead of AWS.
   Task 3 : 1pt: Research on how to deploy API on SOE instead of elastic beanstalk.

4. 5 pts: As front end developer, I would like a watering threshold included into the database so that I can build a graph that will show the end user how much water is being used.

   Task 1 : 3 pts: Include a database table that will hold watering information.
   Task 2 : 2 pts: Write API call that will insert watering information into database and return back watering information for the front end team.

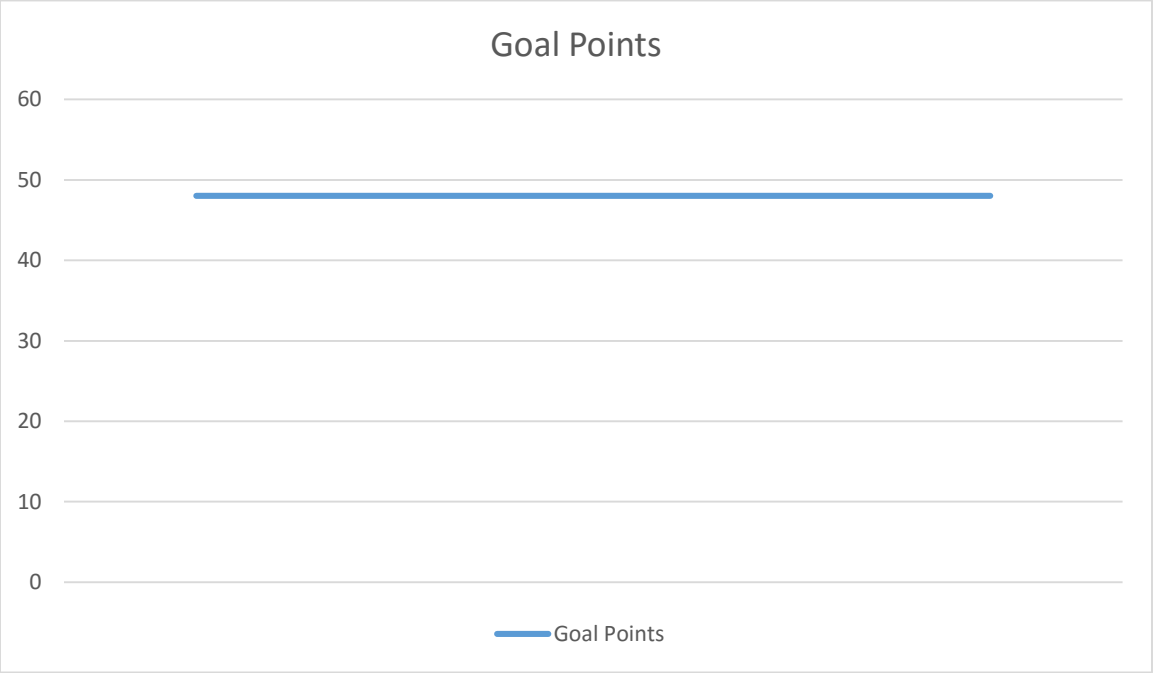
5. 3 pts: As a developer, I want to generate a simulated stream of data to be tested for rewritten pi code.

Task 1 : 1pt: Rewrite RandCSV.py so that every 15 minutes, a list is outputted with random generated values of the sensors.
Task 2 : 2pts: Create a python script which will listen to stdout, and upload to the database the values that are outputted.

6. 11 pts: As a developer, I want to store weather data in a database.
   Task 1 : 2 pts: Set up a table that will store weather data in the database with appropriate columns.
   Task 2 : 3 pts: Use wunderground API to retrieve relevant weather data.
   Task 3 : 8 pts: Create code that will push wunderground data into the weather data table.

7. 3 pts: As a developer, I want to use the API to specify weather statistics to use.
   Task 1 : 2 pts: Add in API call that will return all weather data in table.
   Task 2: 1 pt: Modify the call so that the frontend user can specify which fields it would like returned. (ex. Wind, humidity, 7-day forecast, etc.)

Initial Burnup Chart



Initial Scrum Board

Visible in Jack Baskin room 316.

Team Roles:

Theodore Handleman : Product Owner, Developer
Ryan Connors : Developer, QA
Coy Humphrey : Developer
Ernesto Hernandez : Scrum Master, Developer
Ryan Fulscher : Scrum Master, Developer

Scrum Times
All at Baskin 316
Monday @ 5:30pm
Thursday @ 5:30pm
Friday @ 12:30pm