Product Backlog
Team: HealthCat (vsyed & ajfische)

Web App Interface
1. Login
   1.1. When a logged out user tries to access any page on site, direct to Log In page
   1.2. Input email and password
   1.3. Login to the site
   1.4. Go to Profile page
2. Register
   2.1. Click *Register* on Log In page
   2.2. Fill out form (email, password, confirm password, name)
   2.3. Send email confirmation
   2.4. Upon clicking email link, display confirmation and link to Log In page
3. Reset Password
   3.1. A logged out user can enter email to send a new password link.
4. Edit Profile information about a himself/ herself
   4.1. Edit Location, Profile Picture, and Email
5. View Profile
   5.1. Display name, profile picture, location on left sidebar
   5.2. Display list of bowls with a list of pets under each bowl
6. Change Password
   6.1. Enter old password
   6.2. Enter new password
   6.3. Confirm new password
7. Add KibbleControl food bowls
   7.1. Click *Add Bowl* on Profile page
   7.2. Input id of bowl and name for bowl (such as 'Kitchen','Cat Bowl', or 'Dog Bowl')
   7.3. Click *Connect*
   7.4. Send signal to connect to bowl, have user push button on bowl, bowl sends confirmation, and link bowl id to user
   7.5. If successful, go to Bowl Settings form
8. Edit KibbleControl bowl settings
   8.1. Click the 'Edit' icon next to bowl name
   8.2. Check/uncheck to receive Bowl Empty Alert
   8.3. Set open and close times
9. Add Pet Profiles
   9.1. Click *Add Pet* on Profile page underneath the bowl name that the pet should eat from
   9.2. Fill out form of pet information (age, weight, indoor/outdoor, etc)
   9.3. Fill out form of diet restrictions (how much food per day, time intervals, etc)
   9.4. Click *Submit* at bottom of form
   9.5. Display pet name and picture on Profile page

10. Update Pet Profiles
    10.1. Click the 'Edit' icon next to pet name on Profile page
    10.2. Edit form inputs as needed and hit *Update*
11. Display Statistics for Pet
    11.1. Click *Statistics* nav button in navbar
    11.2. Display interactive charts and graphs about pet's diet from data collected from KibbleContol bowl
    11.3. Allow user to show/hide each pets' data on each graph
    11.4. Allow user to change timescale for each chart
    11.5. Chart for Food Consumption over Time
    11.6. Chart for Time of Day that Food is Eaten
    11.7. Chart for Duration of Sitting
12. Display Statistics for Bowl
    12.1. Click *Statistics* nav button in navbar
    12.2. Click *Bowl Statistics* on Statistics page
    12.3. Display charts and graphs of which pets are eating at which bowl
    12.4. Display how much food is eaten from each bowl per day
    12.5. Display how much food is left
13. Make Recommendations based on Pet Profile
    13.1. Recommend food brand/type
    13.2. Recommend ideal weight
    13.3. Recommend feeding schedule
    13.4. Recommend amount to feed per day
14. Analyze Statistics
    14.1. Interpret data into text to display to user.
15. Make Recommendations based on Analysis of Statistics
    15.1. Recommend feeding schedule
    15.2. Recommend amount to feed

Bowl Communication

The communication between the bowl running Raspberry PI and the web app server will be handled by vsyed. Web app sends REST API calls to bowl to program the bowl. Bowl sends http requests to server to update database.

Raspberry Pi

SqlLite database shall hold minimum data for the box to operate without internet connection. Sprint1 shall include what all data is held in the database in Raspberry Pi.

Web server can compute when a cat can eat or not based on the time and other cat requirements and reply True or False.

The logic required to do that shall be programmed by us.

SQLite database

Control program makes queries like

localhost:8000/openOrNot/2245hrs/CAT4C

Django Web server

Mech Control program

When ever a change in cat timings to eat happens the change is pushed as a POST request to the port where the app is running.

Updates log is sent back after each cat eats food as a POST request.
CATID , Bowl ID ,
bowlPasscode , time eaten, amount eaten

Empty bowl alert is sent.
Bowl ID, bowlPasscode

Heroku

This web application shall be deployed on Heroku or other...
This holds all the data that require this set up to be working. It has the ability to serve all the statistics updated by then.

Postgresql

Django Application

UI

User