

UP Trail Networks

Joel Whalen
Connor Laitinen

The Problem

- 28 Bluetooth enabled motion sensors
- Known latitudes and longitudes
- Each sensor creates a list of timestamps for trail activity

1 12:18:09 11/26/17/4

2 12:18:23 11/26/17/4

...

- Manually collected, updated, and analyzed
- Wrong date time format (should be YYYY-MM-DDTHH:MM:SS)

The Solution - Server

- Nodejs server that runs on euclid.nmu.edu:4009
- Users upload the manually collected text files, and nodejs hands them off to the Python Converter module for processing
 - Easier than doing asynchronous database calls in Javascript
 - Server waits for Python process to finish successfully
- POST and GET used to upload files
- Socket.IO for all other client-server communication
- Master CSV file containing a dictionary of all latitude/longitude values and trail node name

The Solution - Database

- MongoDB is easier to use than MySQL
- Don't have to change mongo shell query syntax at all because Nodejs is fully compatible
- Aggregate queries do the heavy lifting with date time calculations
- Always trust date time objects! Don't reinvent the wheel

The Solution - Client

- Heatmap.js for Google Maps API
 - Best API experience we've ever had
 - Essentially one line of code to update the heat map from the database query
 - MVCArray automatically updates heat map data
- Graph.js for Chart API
 - Uses the same queries as Heatmap.js
 - Simply pipes the data into a different API to be viewed in a chart instead of a map

The Solution - Collaboration

- Github made group work simple
- Easy to track changes and merge conflicts
- We both have a good handle on source control and project management
- Kept separate branches, making bug fixes as easy as “git checkout <branch>” to see what was wrong