CSC 317 Final Documentation
Fall 2019

Bradley Justice
https://github.com/pambalos/csc317-final

Write-Up

Project Overview

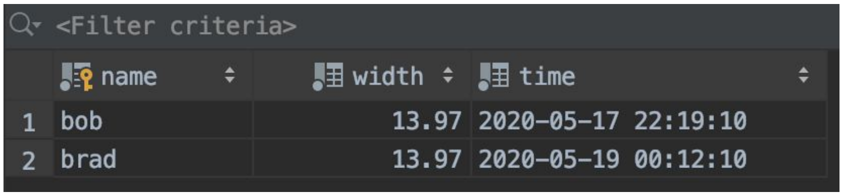
This project involved creating a fully functional backend and front end webapp, for the purpose of data collection, display and review. In this case, it is the case of receiving car vehicle data segregated by session.

Technical Overview

I chose to go simply with the MAMP, (MacOS, Apache, MySQL, PHP) stack as it is something that I have, at this point, become intimately familiar with, and as such, knew that I could build this product with.

To start, I had to come up with a database design that would allow me to track all of the sessions and vehicles in a way that made sense and worked. I ended up coming up with 6 different objects; vehicles, sessions, wheelrecords, linerecords, echorecords, and otherrecords. This design allowed me to keep track of all of the data that I needed to, at least according to the design specification, as I understood it. Below are examples of my database:

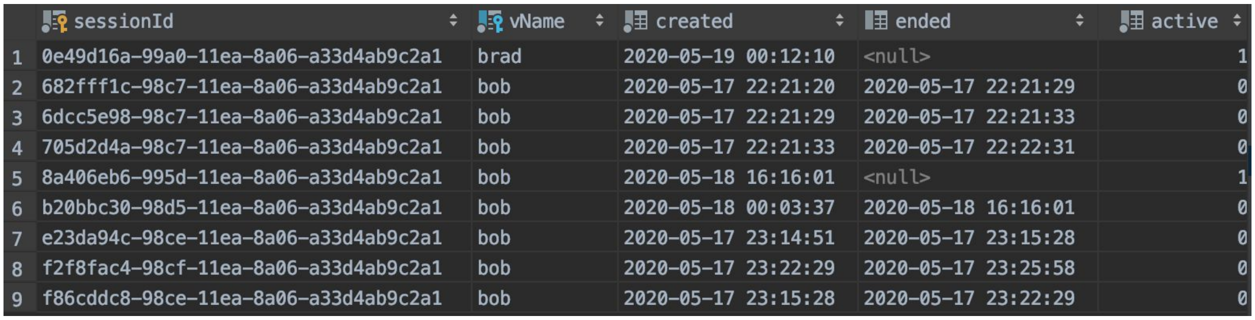
Vehicles:



<Filter criteria>

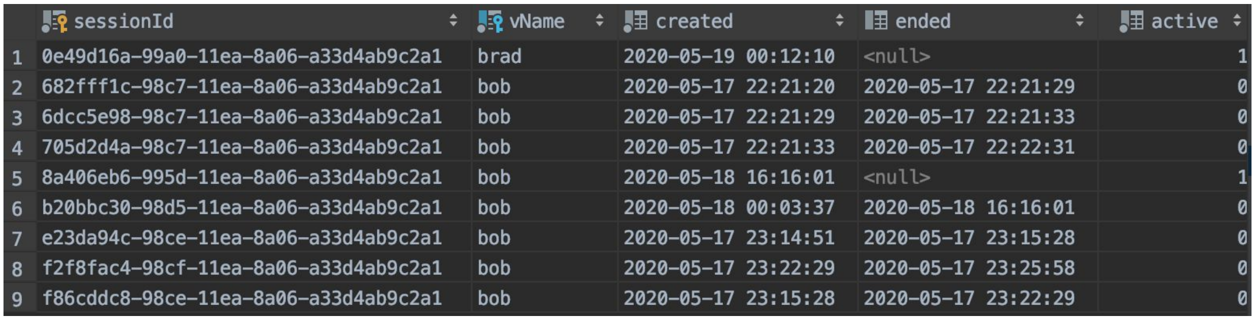
	name	width	time
1	bob	13.97	2020-05-17 22:19:10
2	brad	13.97	2020-05-19 00:12:10

Sessions:



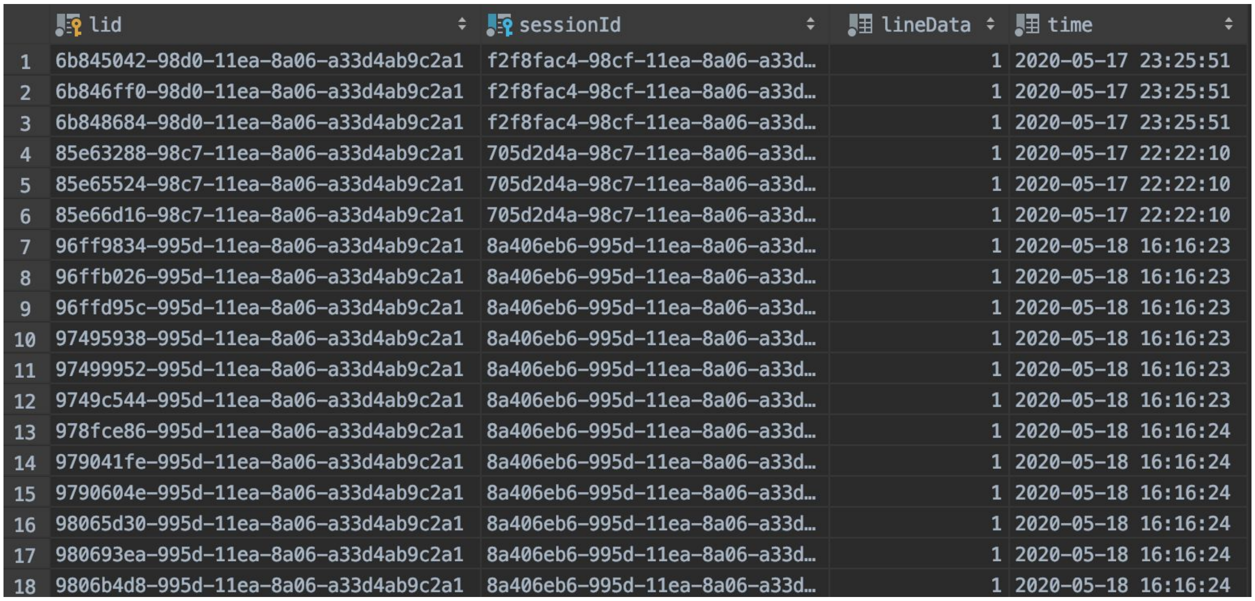
	sessionId	vName	created	ended	active
1	0e49d16a-99a0-11ea-8a06-a33d4ab9c2a1	brad	2020-05-19 00:12:10	<null>	1
2	682fff1c-98c7-11ea-8a06-a33d4ab9c2a1	bob	2020-05-17 22:21:20	2020-05-17 22:21:29	0
3	6dcc5e98-98c7-11ea-8a06-a33d4ab9c2a1	bob	2020-05-17 22:21:29	2020-05-17 22:21:33	0
4	705d2d4a-98c7-11ea-8a06-a33d4ab9c2a1	bob	2020-05-17 22:21:33	2020-05-17 22:22:31	0
5	8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	bob	2020-05-18 16:16:01	<null>	1
6	b20bbc30-98d5-11ea-8a06-a33d4ab9c2a1	bob	2020-05-18 00:03:37	2020-05-18 16:16:01	0
7	e23da94c-98ce-11ea-8a06-a33d4ab9c2a1	bob	2020-05-17 23:14:51	2020-05-17 23:15:28	0
8	f2f8fac4-98cf-11ea-8a06-a33d4ab9c2a1	bob	2020-05-17 23:22:29	2020-05-17 23:25:58	0
9	f86cddc8-98ce-11ea-8a06-a33d4ab9c2a1	bob	2020-05-17 23:15:28	2020-05-17 23:22:29	0

Wheel Records:



	sessionId	vName	created	ended	active
1	0e49d16a-99a0-11ea-8a06-a33d4ab9c2a1	brad	2020-05-19 00:12:10	<null>	1
2	682fff1c-98c7-11ea-8a06-a33d4ab9c2a1	bob	2020-05-17 22:21:20	2020-05-17 22:21:29	0
3	6dcc5e98-98c7-11ea-8a06-a33d4ab9c2a1	bob	2020-05-17 22:21:29	2020-05-17 22:21:33	0
4	705d2d4a-98c7-11ea-8a06-a33d4ab9c2a1	bob	2020-05-17 22:21:33	2020-05-17 22:22:31	0
5	8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	bob	2020-05-18 16:16:01	<null>	1
6	b20bbc30-98d5-11ea-8a06-a33d4ab9c2a1	bob	2020-05-18 00:03:37	2020-05-18 16:16:01	0
7	e23da94c-98ce-11ea-8a06-a33d4ab9c2a1	bob	2020-05-17 23:14:51	2020-05-17 23:15:28	0
8	f2f8fac4-98cf-11ea-8a06-a33d4ab9c2a1	bob	2020-05-17 23:22:29	2020-05-17 23:25:58	0
9	f86cddc8-98ce-11ea-8a06-a33d4ab9c2a1	bob	2020-05-17 23:15:28	2020-05-17 23:22:29	0

Line records:



	lid	sessionId	lineData	time
1	6b845042-98d0-11ea-8a06-a33d4ab9c2a1	f2f8fac4-98cf-11ea-8a06-a33d…	1	2020-05-17 23:25:51
2	6b846ff0-98d0-11ea-8a06-a33d4ab9c2a1	f2f8fac4-98cf-11ea-8a06-a33d…	1	2020-05-17 23:25:51
3	6b848684-98d0-11ea-8a06-a33d4ab9c2a1	f2f8fac4-98cf-11ea-8a06-a33d…	1	2020-05-17 23:25:51
4	85e63288-98c7-11ea-8a06-a33d4ab9c2a1	705d2d4a-98c7-11ea-8a06-a33d…	1	2020-05-17 22:22:10
5	85e65524-98c7-11ea-8a06-a33d4ab9c2a1	705d2d4a-98c7-11ea-8a06-a33d…	1	2020-05-17 22:22:10
6	85e66d16-98c7-11ea-8a06-a33d4ab9c2a1	705d2d4a-98c7-11ea-8a06-a33d…	1	2020-05-17 22:22:10
7	96ff9834-995d-11ea-8a06-a33d4ab9c2a1	8a406eb6-995d-11ea-8a06-a33d…	1	2020-05-18 16:16:23
8	96ffb026-995d-11ea-8a06-a33d4ab9c2a1	8a406eb6-995d-11ea-8a06-a33d…	1	2020-05-18 16:16:23
9	96ffd95c-995d-11ea-8a06-a33d4ab9c2a1	8a406eb6-995d-11ea-8a06-a33d…	1	2020-05-18 16:16:23
10	97495938-995d-11ea-8a06-a33d4ab9c2a1	8a406eb6-995d-11ea-8a06-a33d…	1	2020-05-18 16:16:23
11	97499952-995d-11ea-8a06-a33d4ab9c2a1	8a406eb6-995d-11ea-8a06-a33d…	1	2020-05-18 16:16:23
12	9749c544-995d-11ea-8a06-a33d4ab9c2a1	8a406eb6-995d-11ea-8a06-a33d…	1	2020-05-18 16:16:23
13	978fce86-995d-11ea-8a06-a33d4ab9c2a1	8a406eb6-995d-11ea-8a06-a33d…	1	2020-05-18 16:16:24
14	979041fe-995d-11ea-8a06-a33d4ab9c2a1	8a406eb6-995d-11ea-8a06-a33d…	1	2020-05-18 16:16:24
15	9790604e-995d-11ea-8a06-a33d4ab9c2a1	8a406eb6-995d-11ea-8a06-a33d…	1	2020-05-18 16:16:24
16	98065d30-995d-11ea-8a06-a33d4ab9c2a1	8a406eb6-995d-11ea-8a06-a33d…	1	2020-05-18 16:16:24
17	980693ea-995d-11ea-8a06-a33d4ab9c2a1	8a406eb6-995d-11ea-8a06-a33d…	1	2020-05-18 16:16:24
18	9806b4d8-995d-11ea-8a06-a33d4ab9c2a1	8a406eb6-995d-11ea-8a06-a33d…	1	2020-05-18 16:16:24

Data Collection:

There were a number of database design decisions that I had to make, including what linked to what other objects, in order to get the relationship between all the data that I wanted. I came up with session records, which were linked to by many different session data points. In this way, I was able to very easily get the data that I wanted.

I built a number of different webpages/endpoints to handle the requests. These acted essentially as Get endpoints. These endpoints included:

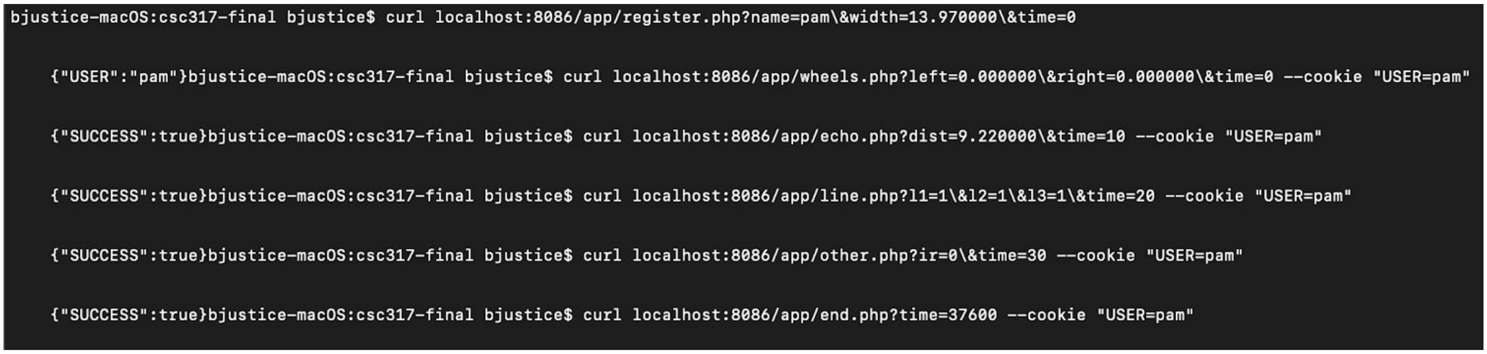
/register?name=XXXX&width=###.###
Added a new vehicle run to the system, returns a cookie called USER=[name] that would be included for the other commands. Width is the width of the vehicle in cm. It overwrites any other "active" session for that named vehicle

/wheels?left=###.###&right=###.###
Records the speed of the left and right wheel in cm/sec for that vehicle in the current session

/echo?dist=###.###
Records the result of the echo sensor in cm for the vehicle in the current session

/line?l1=##&l2=##&l3=##
Records the result of ONE or MORE l1, l2, l3, line sensors

Here is a screenshot showcasing these endpoints:



```
bjustice-macOS:csc317-final bjustice$ curl localhost:8086/app/register.php?name=pam\&width=13.970000\&time=0

    {"USER":"pam"}bjustice-macOS:csc317-final bjustice$ curl localhost:8086/app/wheels.php?left=0.000000\&right=0.000000\&time=0 --cookie "USER=pam"

    {"SUCCESS":true}bjustice-macOS:csc317-final bjustice$ curl localhost:8086/app/echo.php?dist=9.220000\&time=10 --cookie "USER=pam"

    {"SUCCESS":true}bjustice-macOS:csc317-final bjustice$ curl localhost:8086/app/line.php?l1=1\&l2=1\&l3=1\&time=20 --cookie "USER=pam"

    {"SUCCESS":true}bjustice-macOS:csc317-final bjustice$ curl localhost:8086/app/other.php?ir=0\&time=30 --cookie "USER=pam"

    {"SUCCESS":true}bjustice-macOS:csc317-final bjustice$ curl localhost:8086/app/end.php?time=37600 --cookie "USER=pam"
```

Backend (PHP):

I used PHP for this entire project as it was something that I have become quite familiar and comfortable using alongside html and javascript.

One thing I was experimenting with but dont think I will complete in time for the submission deadline was a routing engine. As can be seen from the source code, there is a '.htcaccess' file that sits in the root, that, when turned on, will redirect all traffic to the index.php file, which I was working on implementing as a router, but did not finish.

Instead, all of the data collection php endpoints just sit in the root folder.

Frontend (HTML, CSS & JS):

I ended up just using good old html, css and javascript to make my front end pages responsive, dynamic, and filterable. In each case, I simply had a php script that echo the json_encode of the desired data. In this way, I made my front and backend communicate.

In addition to this method of data communication, I also decided to do all of my filtering in the ***Javascript***, and not the php, that way I wouldn't have to reload the data. (granted, I could technically write endpoints that return filtered data, and have a javascript update, but I decided it made more sense and would be easier to populate the data ahead of time, then apply whatever filters I wanted via front end style manipulation (display = none).

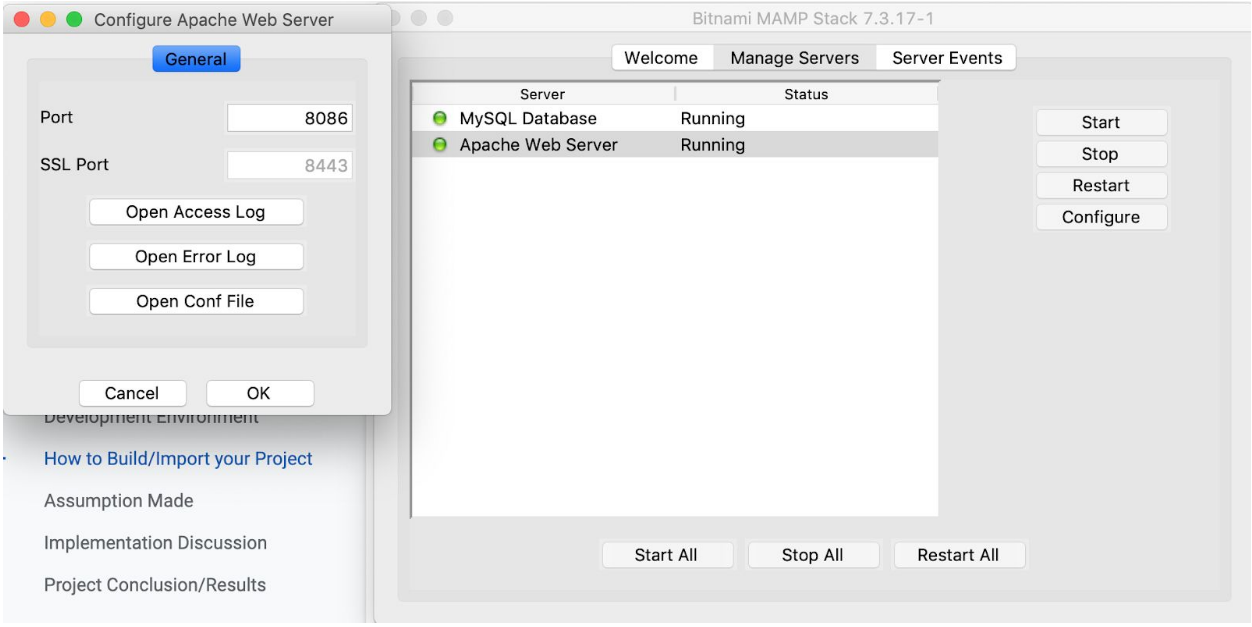
Development Environment

I used the MAMP stack from Bitnami at https://bitnami.com/stack/mamp
This allowed me to install and run a full Apache, MySQL and PHP stack, which I was familiar with using from the storefront group project.

How to Build/Import

As mentioned before, this project is built to run PHP on an Apache Web server. So long as you have php 5.4 or later, it should compile and run fine. You can simply clone the Github folder from https://github.com/pambalos/csc317-final into the php server, and be able to access the website.

Here is a screenshot of my stack manager:



Port 8086

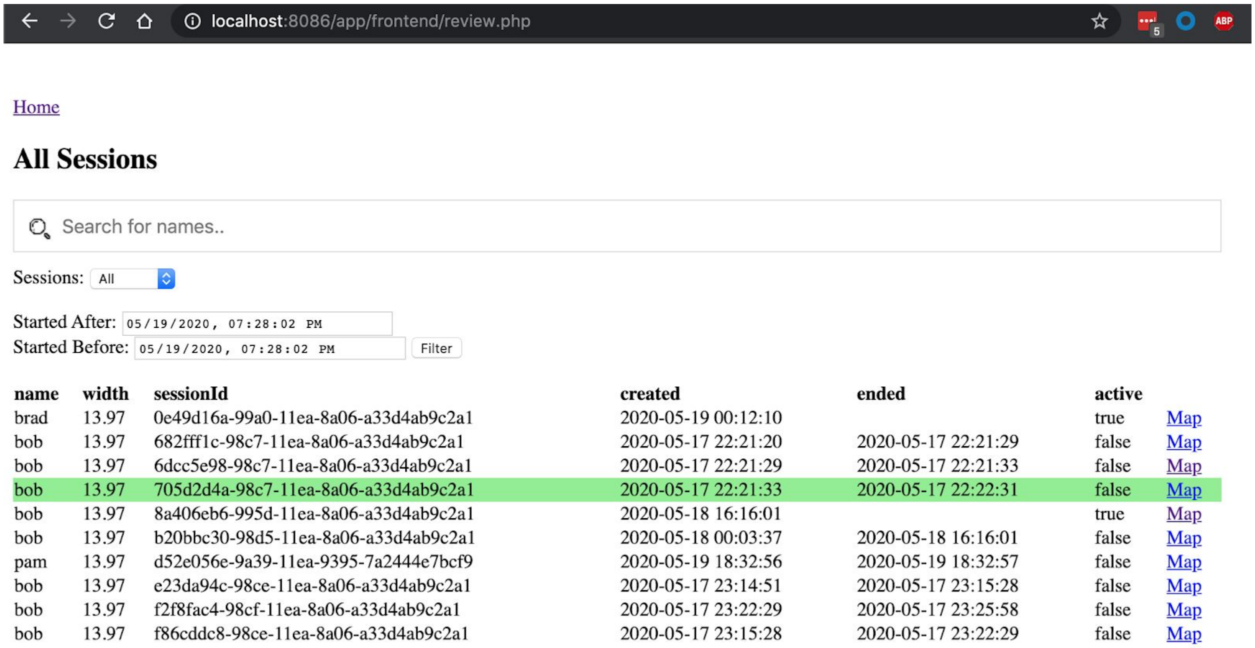
SSL Port 8443

Server	Status
MySQL Database	Running
Apache Web Server	Running

As can be seen, I simply had a MySQL database, and an Apache Web Server.

Demo

/review



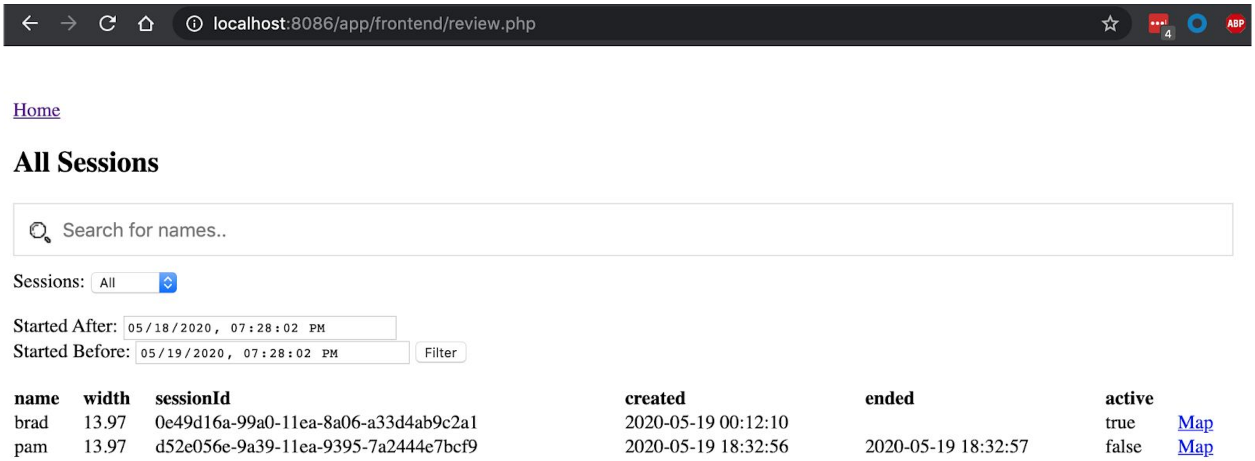
All Sessions

Sessions: All

Started After: 05/19/2020, 07:28:02 PM
Started Before: 05/19/2020, 07:28:02 PM

name	width	sessionId	created	ended	active	
brad	13.97	0e49d16a-99a0-11ea-8a06-a33d4ab9c2a1	2020-05-19 00:12:10		true	Map
bob	13.97	682fff1c-98c7-11ea-8a06-a33d4ab9c2a1	2020-05-17 22:21:20	2020-05-17 22:21:29	false	Map
bob	13.97	6dcc5e98-98c7-11ea-8a06-a33d4ab9c2a1	2020-05-17 22:21:29	2020-05-17 22:21:33	false	Map
bob	13.97	705d2d4a-98c7-11ea-8a06-a33d4ab9c2a1	2020-05-17 22:21:33	2020-05-17 22:22:31	false	Map
bob	13.97	8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	2020-05-18 16:16:01		true	Map
bob	13.97	b20bbc30-98d5-11ea-8a06-a33d4ab9c2a1	2020-05-18 00:03:37	2020-05-18 16:16:01	false	Map
pam	13.97	d52e056e-9a39-11ea-9395-7a2444e7bcf9	2020-05-19 18:32:56	2020-05-19 18:32:57	false	Map
bob	13.97	e23da94c-98ce-11ea-8a06-a33d4ab9c2a1	2020-05-17 23:14:51	2020-05-17 23:15:28	false	Map
bob	13.97	f2f8fac4-98cf-11ea-8a06-a33d4ab9c2a1	2020-05-17 23:22:29	2020-05-17 23:25:58	false	Map
bob	13.97	f86cddc8-98ce-11ea-8a06-a33d4ab9c2a1	2020-05-17 23:15:28	2020-05-17 23:22:29	false	Map

This page lists all of the sessions that are saved in the database. You can click on any one of these rows to navigate to /datareview, or /map. As can be seen, there are filter options of vehicle name, session active status, and the date filter. All of these have been tested and work. Ie.
(date filtered)



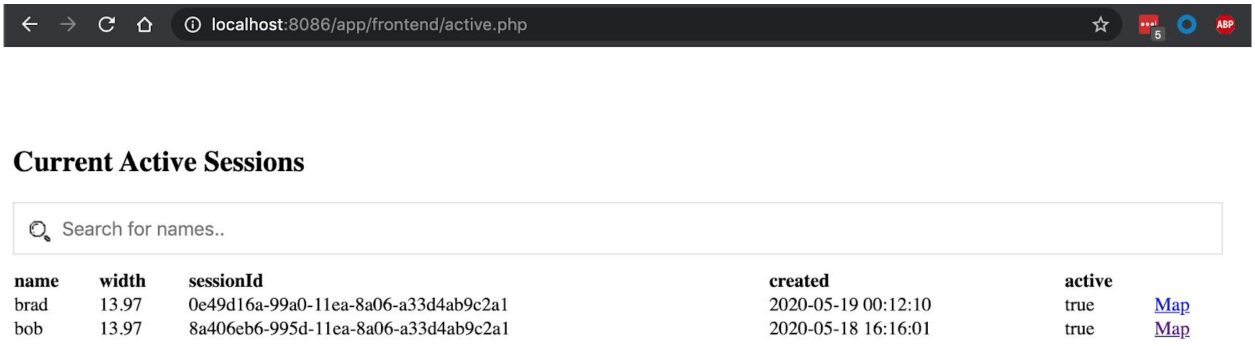
All Sessions

Sessions: All

Started After: 05/18/2020, 07:28:02 PM
Started Before: 05/19/2020, 07:28:02 PM

name	width	sessionId	created	ended	active	
brad	13.97	0e49d16a-99a0-11ea-8a06-a33d4ab9c2a1	2020-05-19 00:12:10		true	Map
pam	13.97	d52e056e-9a39-11ea-9395-7a2444e7bcf9	2020-05-19 18:32:56	2020-05-19 18:32:57	false	Map

/Active



localhost:8086/app/frontend/active.php

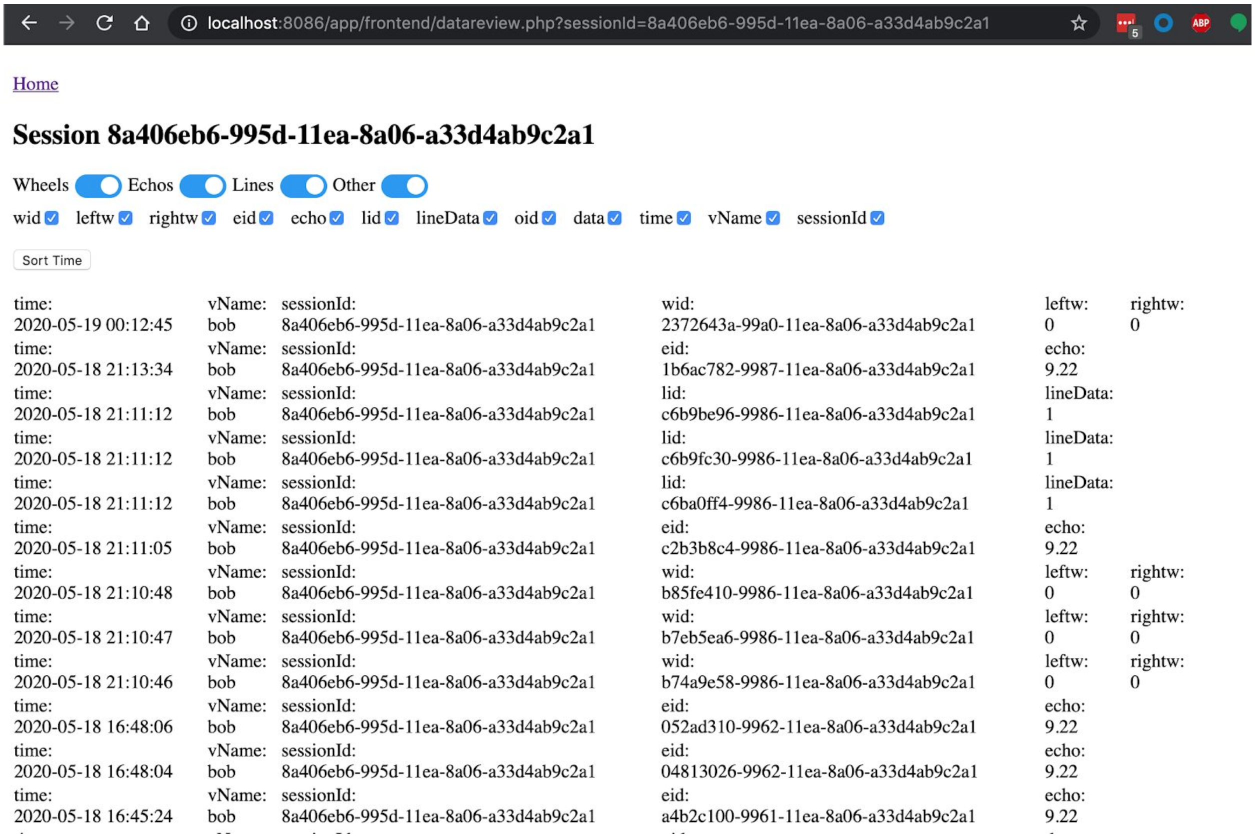
Current Active Sessions

Search for names..

name	width	sessionId	created	active	
brad	13.97	0e49d16a-99a0-11ea-8a06-a33d4ab9c2a1	2020-05-19 00:12:10	true	Map
bob	13.97	8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	2020-05-18 16:16:01	true	Map

This page simply shows all of the active sessions.

/datareview?sessionId=####



localhost:8086/app/frontend/datareview.php?sessionId=8a406eb6-995d-11ea-8a06-a33d4ab9c2a1

Home

Session 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1

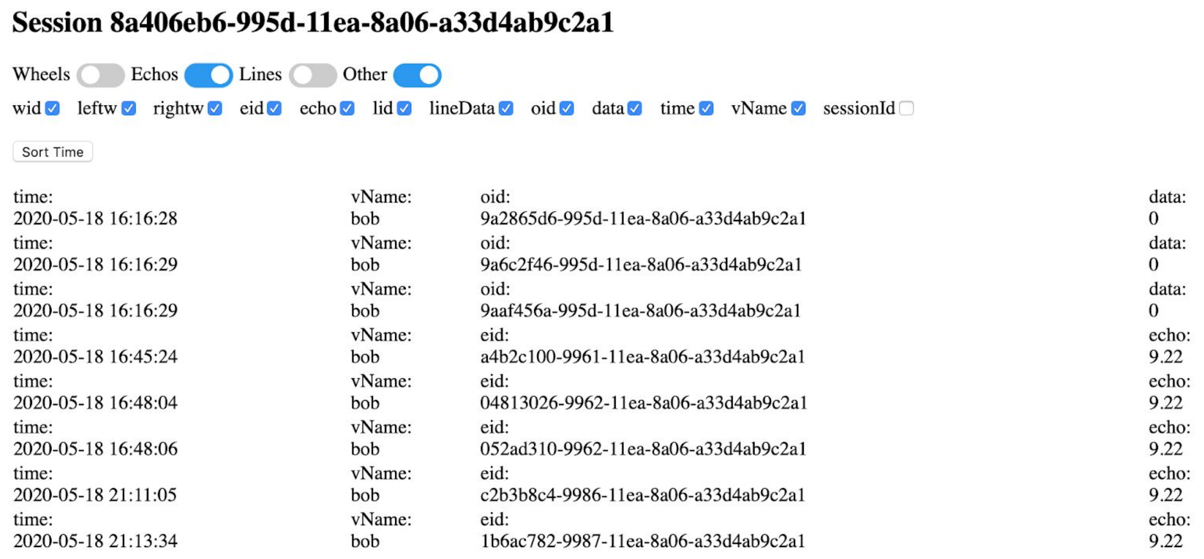
Wheels ● Echos ● Lines ● Other ●

wid ☑ leftw ☑ rightw ☑ eid ☑ echo ☑ lid ☑ lineData ☑ oid ☑ data ☑ time ☑ vName ☑ sessionId ☑

Sort Time

time: 2020-05-19 00:12:45	vName: bob	sessionId: 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	wid: 2372643a-99a0-11ea-8a06-a33d4ab9c2a1	leftw: 0	rightw: 0
time: 2020-05-18 21:13:34	vName: bob	sessionId: 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	eid: 1b6ac782-9987-11ea-8a06-a33d4ab9c2a1	echo: 9.22	
time: 2020-05-18 21:11:12	vName: bob	sessionId: 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	lid: c6b9be96-9986-11ea-8a06-a33d4ab9c2a1	lineData: 1	
time: 2020-05-18 21:11:12	vName: bob	sessionId: 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	lid: c6b9fc30-9986-11ea-8a06-a33d4ab9c2a1	lineData: 1	
time: 2020-05-18 21:11:12	vName: bob	sessionId: 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	lid: c6ba0ff4-9986-11ea-8a06-a33d4ab9c2a1	lineData: 1	
time: 2020-05-18 21:11:05	vName: bob	sessionId: 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	eid: c2b3b8c4-9986-11ea-8a06-a33d4ab9c2a1	echo: 9.22	
time: 2020-05-18 21:10:48	vName: bob	sessionId: 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	wid: b85fe410-9986-11ea-8a06-a33d4ab9c2a1	leftw: 0	rightw: 0
time: 2020-05-18 21:10:47	vName: bob	sessionId: 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	wid: b7eb5ea6-9986-11ea-8a06-a33d4ab9c2a1	leftw: 0	rightw: 0
time: 2020-05-18 21:10:46	vName: bob	sessionId: 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	wid: b74a9e58-9986-11ea-8a06-a33d4ab9c2a1	leftw: 0	rightw: 0
time: 2020-05-18 16:48:06	vName: bob	sessionId: 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	eid: 052ad310-9962-11ea-8a06-a33d4ab9c2a1	echo: 9.22	
time: 2020-05-18 16:48:04	vName: bob	sessionId: 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	eid: 04813026-9962-11ea-8a06-a33d4ab9c2a1	echo: 9.22	
time: 2020-05-18 16:45:24	vName: bob	sessionId: 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1	eid: a4b2c100-9961-11ea-8a06-a33d4ab9c2a1	echo: 9.22	

This page takes one parameter, the sessionId, and retrieves every single record (via php) that is associated with that sessionId (wheels, echos, lines, others, dist) and compiles them into a single array that gets iterated and printed as a table via javascript. The filtering is also all done via javascript and took quite a bit of fiddling around with until I got it right. As can be seen, filtering works both on column, record type, and time:



Session 8a406eb6-995d-11ea-8a06-a33d4ab9c2a1

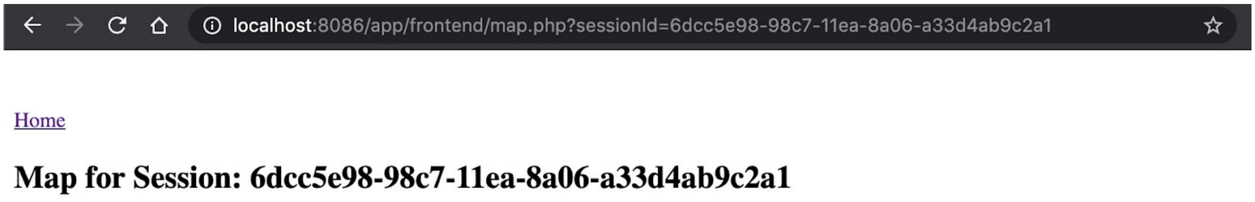
wid ☑ leftw ☑ rightw ☑ eid ☑ echo ☑ lid ☑ lineData ☑ oid ☑ data ☑ time ☑ vName ☑ sessionId □

time: 2020-05-18 16:16:28	vName: bob	oid: 9a2865d6-995d-11ea-8a06-a33d4ab9c2a1	data: 0
time: 2020-05-18 16:16:29	vName: bob	oid: 9a6c2f46-995d-11ea-8a06-a33d4ab9c2a1	data: 0
time: 2020-05-18 16:16:29	vName: bob	oid: 9aaf456a-995d-11ea-8a06-a33d4ab9c2a1	data: 0
time: 2020-05-18 16:45:24	vName: bob	eid: a4b2c100-9961-11ea-8a06-a33d4ab9c2a1	echo: 9.22
time: 2020-05-18 16:48:04	vName: bob	eid: 04813026-9962-11ea-8a06-a33d4ab9c2a1	echo: 9.22
time: 2020-05-18 16:48:06	vName: bob	eid: 052ad310-9962-11ea-8a06-a33d4ab9c2a1	echo: 9.22
time: 2020-05-18 21:11:05	vName: bob	eid: c2b3b8c4-9986-11ea-8a06-a33d4ab9c2a1	echo: 9.22
time: 2020-05-18 21:13:34	vName: bob	eid: 1b6ac782-9987-11ea-8a06-a33d4ab9c2a1	echo: 9.22

In this screenshot, the sort time has been clicked (switches time ordering), as well as filtering out wheels and line records, and hiding sessionId.

/map?sessionId=###

Also takes on parameter, but doesnt actually do anything - I couldnt figure out the distance mapping equations given.



localhost:8086/app/frontend/map.php?sessionId=6dcc5e98-98c7-11ea-8a06-a33d4ab9c2a1

Map for Session: 6dcc5e98-98c7-11ea-8a06-a33d4ab9c2a1

Project Conclusion/Results

This project was actually a very fun exercise in terms of back end development. I very much enjoyed building the ‘API’ to receive and upload data, as well as figuring out clean and efficient ways to communicate between my front end and back end. Throughout this project I feel like I really got a lot of useful practice in that regard.