

Project 2: Full-Stack & Visualization



Pamela



Swetha



Mark

Despite the Internet, Kids Still Involved in Extracurricular Activities



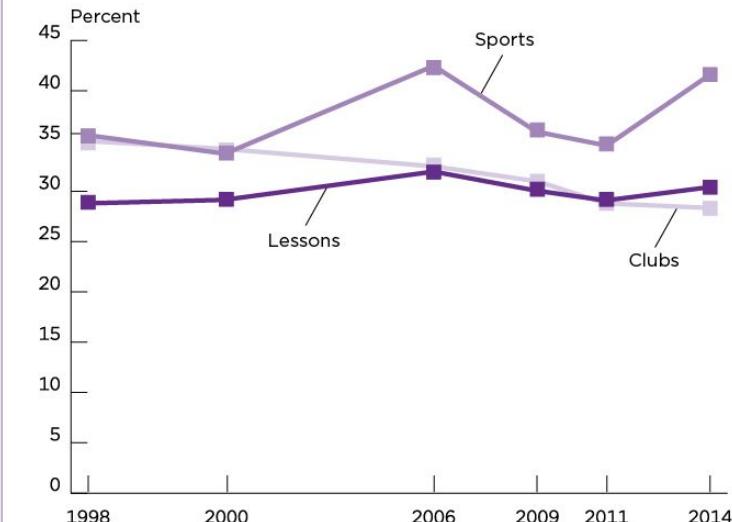
Children and Teenagers More Engaged in Sports Than 15 Years Ago

Rising participation in after-school activities

“83% parents agree that afterschool programs help working parents keep their jobs.”

“Cost and lack of a safe way for their children to get to and come home from afterschool programs are among the barriers that low-income households report keep them from enrolling their children in an afterschool program”

Children 6 to 17 Years Old Involved in After-School Activities: 1998–2014



Source: U.S. Census Bureau, Survey of Income and Program Participation, 2014 Panel, Wave 1; 2008 Panel, Wave 10; 2008 Panel, Wave 4; 2004 Panel, Wave 8; 1996 Panel, Wave 12; 1996 Panel, Wave 6.

Parents & After-School Kids Activities

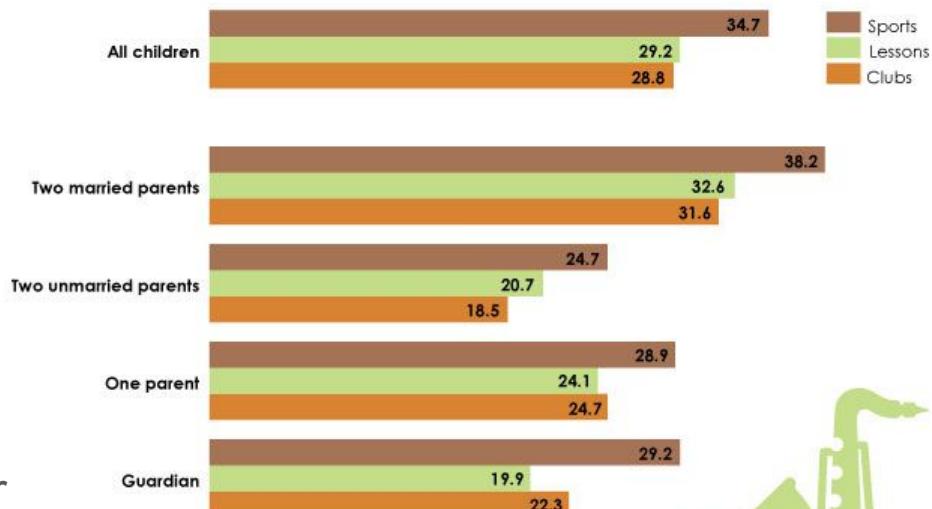
Kids with **married parents** generally participate in **more activities**

Kids with a **single parent** or guardian generally participate in **fewer activities**

Availability of **safe rides** for kids to their activities would make it **easier** for parents

Beyond the Classroom

Percent of Children Participating in Extracurricular Activities



Note: Children may participate in more than one activity.

United States
Census
Bureau

U.S. Department of Commerce
Economics and Statistics Administration
U.S. CENSUS BUREAU
census.gov

Source: Survey of Income and Program Participation (SIPP),
2008 Panel Wave 10

Problem: Kids need rides to activities

Parents need to get their kids to a number of after-school activities, often conflicting with their work schedules

UBER & Lyft **doesn't allow** for minors to be picked up by their drivers

Uber and Lyft prohibit unaccompanied minors

Both of the major ride-hail companies prohibit travel with anyone under 18 years of age who is not accompanied by an adult. However, enforcement of this rule obviously has to be left up to the drivers. So, one driver may allow your **kids** to ride and another may not. Jul 6, 2019



Source: Google

Solution: Kids Ride



Safe & Reliable, Worry-free, Quality Transportation.
Professional transport services for your kids

Project Requirements

Your visualization must include a **Python Flask–powered RESTful API, HTML/CSS, JavaScript**, and at least **one database** (SQL, **MongoDB**, SQLite, etc.).

Your project should fall into one of the below four tracks:

- 1) A custom "creative" D3.js project (i.e., a nonstandard graph or chart)
- 2) A combination of web scraping and Leaflet or Plotly**
- 3) A dashboard page with multiple charts that update from the same data
- 4) A "thick" server that performs multiple manipulations on data in a database prior to visualization
(must be approved)

Your project should include at least **one JS library that we did not cover**.

Your project must be powered by **a data set with at least 100 records**.

Your project must include some level of **user-driven interaction** (e.g., menus, dropdowns, textboxes).

Your final visualization should ideally include **at least three views**.

Webscraping

Scraped Google Website for:

- 1) Kids Dance Classes
- 2) Kids Sports Classes
- 3) Kids Music Classes

Google kids dance classes new jersey

Rating Hours

Sort by Best match

Ellis Dance Studio
4.4 ★★★★★ (12) · Dance sch...
7.4 mi · Nutley, NJ
(973) 320-2371
Open now
It is a great place to bring your children. ... and...

Gotta Dance
4.2 ★★★★★ (10) · Dance sch...
3.0 mi · Elmwood Park, NJ
(201) 794-0001
in the area absolutely send your kids here! ... for a friend...

DANcetc. Dance/fitness school for kids & adults
Davis Academy of Irish Dancing
Edgewater Performing Arts | Dance Studio...
Ellis Dance Studio
Jersey City Ballet
Across the Floor
Edison School of Music and Dance
Piscataway
Edison
Princeton
Flemington
Hempstead
Queens
Stamford
White Plains
Hicks
Map data ©2019 Google 10 km Terms of Use

Originally wanted to scrape from Kids Guide NJ, Groupon, & Yelp
Ended up scraping from Google because they are well-organized

Required: Chromedriver.exe to run in the background

Webscraping Google For Kids Classes

Find “div” tag, class =”dbg0pd”

Webscraped for:

1. Name:
2. Type: (ex. Dance)
3. URL:
4. PhoneNo:
5. Address: (Raw → Org)
6. City:
7. State:
8. ZipCode:
9. Location: [Lat, Lon]

The screenshot shows a Google search results page for "kids music classes new jersey". The search bar at the top has the query "kids music classes new jersey". Below the search bar, the page title "kids music classes new jersey" is visible. The first result is a blue-highlighted link to "Kts Music School". To the right of the result, there is a map showing locations like "Musical Island of NJ", "Edison", "Rockers", and "Trenton". The DevTools Elements tab is open, showing the DOM structure. A tooltip over the "Kts Music School" link highlights the "div.dbg0pd" element, which is described as having a width of 209.33 x 25 pixels, a margin of 0px 0px -1px, and padding of 2px 0px 1px. The DOM tree on the right side of the DevTools shows the full HTML structure, including the heading for the Kts Music School result.

Geopy.Geolocator Library: Nominatim

From **Street Address**,
Geopy gets **Latitude** and **Longitude**
In Array format: **[latitude, longitude]**

First Install: !pip install geopy

Import: From geopy.geocoders import Nominatim

```
geolocator = Nominatim(user_agent="my-application")
```



Documentation: <https://geopy.readthedocs.io/en/stable/>

```
try:
    locForLatLong = dataframe["address"] + dataframe["city"]
    print(locForLatLong)
    location = geolocator.geocode(locForLatLong, timeout=10)
    print(location.address)
    print((location.latitude,location.longitude))
    dataframe["location"] = [location.latitude,location.longitude]

except AttributeError:
    print('Cant Find the address')
```

Dance Classes stored in Dataframe

```
In [5]: #df1.to_excel("outputDance.xlsx")  
df1
```

Out[5]:

	address	city	location	name	phoneNo	state	type	zipcode
0	234 Franklin Ave	Nutley	[40.815737, -74.161944]	Ellis Dance Studio		NJ	Dance Classes	07110
1	285 Market St	Elmwood Park	[40.9001665555556, -74.118701888889]	Gotta Dance		NJ	Dance Classes	07407
2	357 Broad St %231	Bloomfield	[40.805467, -74.1925796228294]	Broadway Performing Arts	7482787	NJ	Dance Classes	07003
3	7-15 Fair Lawn Ave	Fair Lawn	[40.9242474, -74.1398333]	DANCetc.		NJ	Dance Classes	07410
4	355 Eisenhower Pkwy %23100	Livingston	[40.7967800408163, -74.3412397346939]	Grooves Unlimited Dance Studio		NJ	Dance Classes	07039
5	29 Edward Ct	Clifton	[40.88596, -74.162009]	Davis Academy of Irish Dancing	9733409026	NJ	Dance Classes	07011
6	599 Chestnut St %23 A	Union	NaN	Cherney Dance Studio	9	NJ	Dance Classes	07083

Then, do same for Sports and Music Lessons

Dance, Sports, Music Lessons merged into one df

Merged all Dataframes into a single dataframe

In [12]: ►

```
df = df1.append(df2, ignore_index=True)
df = df.append(df3, ignore_index=True)
df
```

Setup MongoDB connection through pymongo: pymongo object

In [20]: ►

```
# Set up the MongoDB connection through pymongo
myclient = pymongo.MongoClient("mongodb://localhost:27017/")
# Create DB
db = myclient["KidsRidedb3"]
# Create collection and insert all the data into the MongoDB
mycol = db["activityData6"]
mycol.insert_many(split_data)
```

Out[20]: <pymongo.results.InsertManyResult at 0x245d7db6088>

Data in MongoDB: Data rows with NA dropped

activityData1: Documents 328

MongoDB Compass Community - localhost:27017/KidsRidedb3.activityData1

Connect View Collection Help

My Cluster

7 DBS 7 COLLECTIONS

HOST localhost:27017

CLUSTER Standalone

EDITION MongoDB 4.2.0 Community

Filter your data

KidsRidedb3

activityData1

activityData6

admin

config

local

store_inventory

team_db

test

KidsRidedb3.activityData1

DOCUMENTS 328

Documents Aggregations Explain Plan Indexes

INSERT DOCUMENT VIEW LIST TABLE

Filter

Documents

328

`_id: 1
address: " 780 Salem Ave"
city: " Elizabeth"
> location: Array
name: "The Institute of Music for Children"
phoneNo: ""
state: "NJ"
type: "Dance Classes"
zipcode: "07208"`

`_id: 2
address: " 187 Bloomfield Ave"
city: " Bloomfield"
> location: Array
name: "Just Dancin'"
phoneNo: ""
state: "NJ"
type: "Dance Classes"
zipcode: "07003"`

activityData6: Documents: 303

MongoDB Compass Community - localhost:27017/KidsRidedb3.activityData6

Connect View Collection Help

My Cluster

7 DBS 7 COLLECTIONS

HOST localhost:27017

CLUSTER Standalone

EDITION MongoDB 4.2.0 Community

Filter your data

KidsRidedb3

activityData1

activityData6

admin

config

local

store_inventory

team_db

test

KidsRidedb3.activityData6

DOCUMENTS 303

Documents Aggregations Explain Plan Indexes

INSERT DOCUMENT VIEW LIST TABLE

Filter

Documents

303

`_id: 1
address: " 285 Market St"
city: " Elmwood Park"
> location: Array
name: "Gotta Dance"
phoneNo: ""
state: "NJ"
type: "Dance Classes"
zipcode: "07407"`

`_id: 2
address: " 7-15 Fair Lawn Ave"
city: " Fair Lawn"
> location: Array
name: "DANCetc."
phoneNo: ""
state: "NJ"
type: "Dance Classes"
zipcode: "07410"`

Cleaned “Location” data:

In [16]: cleandf = df.dropna(axis=0, subset=['location'])
cleandf

Flask Application (Python)

Four @app.routes():

@app.route("/")

render_template("index.html")

@app.route("/map")

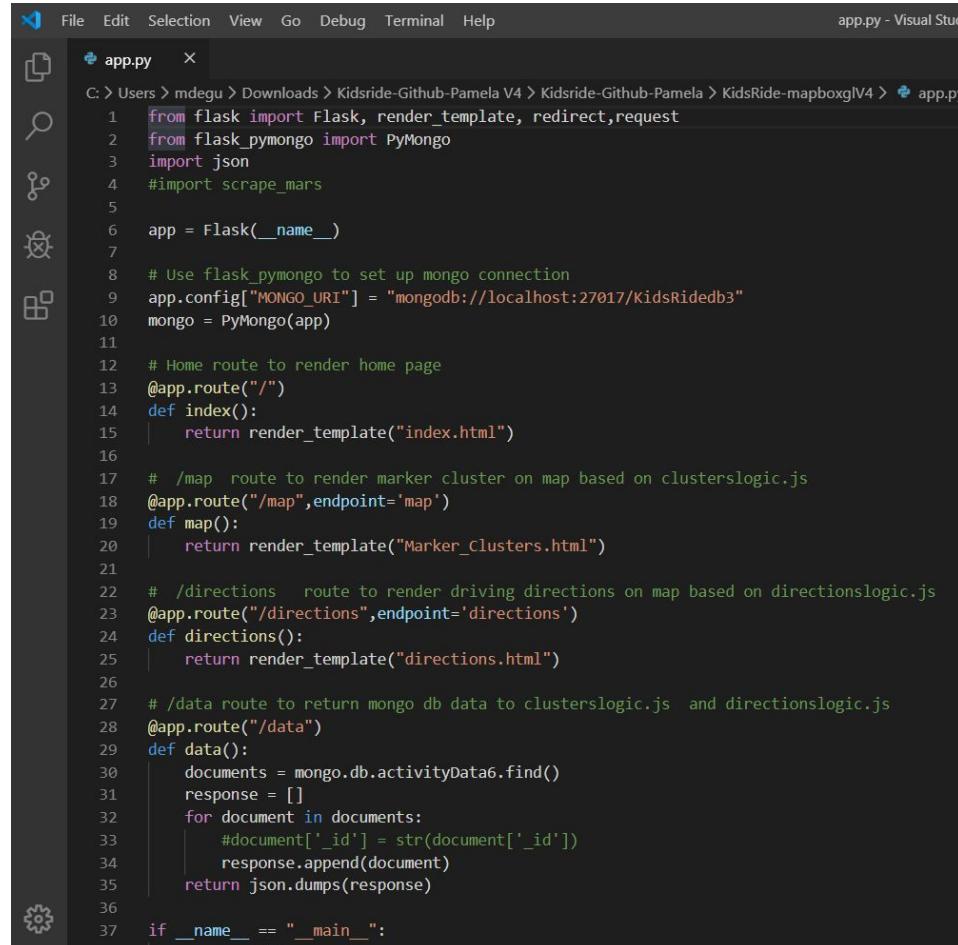
render_template("map.html")

@app.route("/directions")

render_template("directions.html")

@app.route("/data")

Shows the webscraped data



The screenshot shows a Visual Studio Code window with the file 'app.py' open. The code implements a Flask application with four routes:

```
from flask import Flask, render_template, redirect, request
from flask_pymongo import PyMongo
import json
# import scrape_mars

app = Flask(__name__)

# Use flask_pymongo to set up mongo connection
app.config["MONGO_URI"] = "mongodb://localhost:27017/KidsRidedb3"
mongo = PyMongo(app)

# Home route to render home page
@app.route("/")
def index():
    return render_template("index.html")

# /map route to render marker cluster on map based on clusterslogic.js
@app.route("/map", endpoint='map')
def map():
    return render_template("Marker_Clusters.html")

# /directions route to render driving directions on map based on directionslogic.js
@app.route("/directions", endpoint='directions')
def directions():
    return render_template("directions.html")

# /data route to return mongo db data to clusterslogic.js and directionslogic.js
@app.route("/data")
def data():
    documents = mongo.db.activityData6.find()
    response = []
    for document in documents:
        #document['_id'] = str(document['_id'])
        response.append(document)
    return json.dumps(response)

if __name__ == "__main__":
    app.run()
```

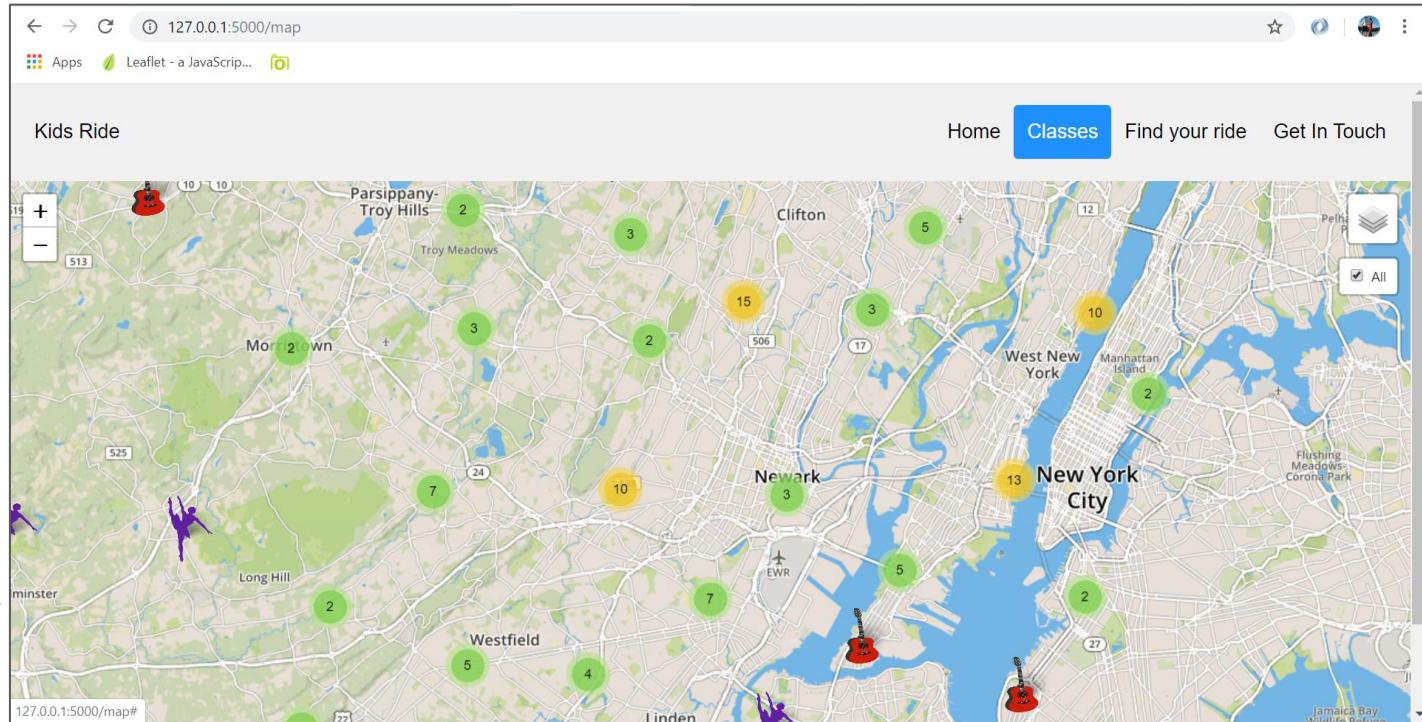
Visualizations: All - MarkerClusters

Shows quantity
of kids activities
clustered in areas

Helps determine
where KidsRide
opportunity exists

More activities:
More opportunity

\$\$\$\$\$\$



Visualizations by type:

Music:



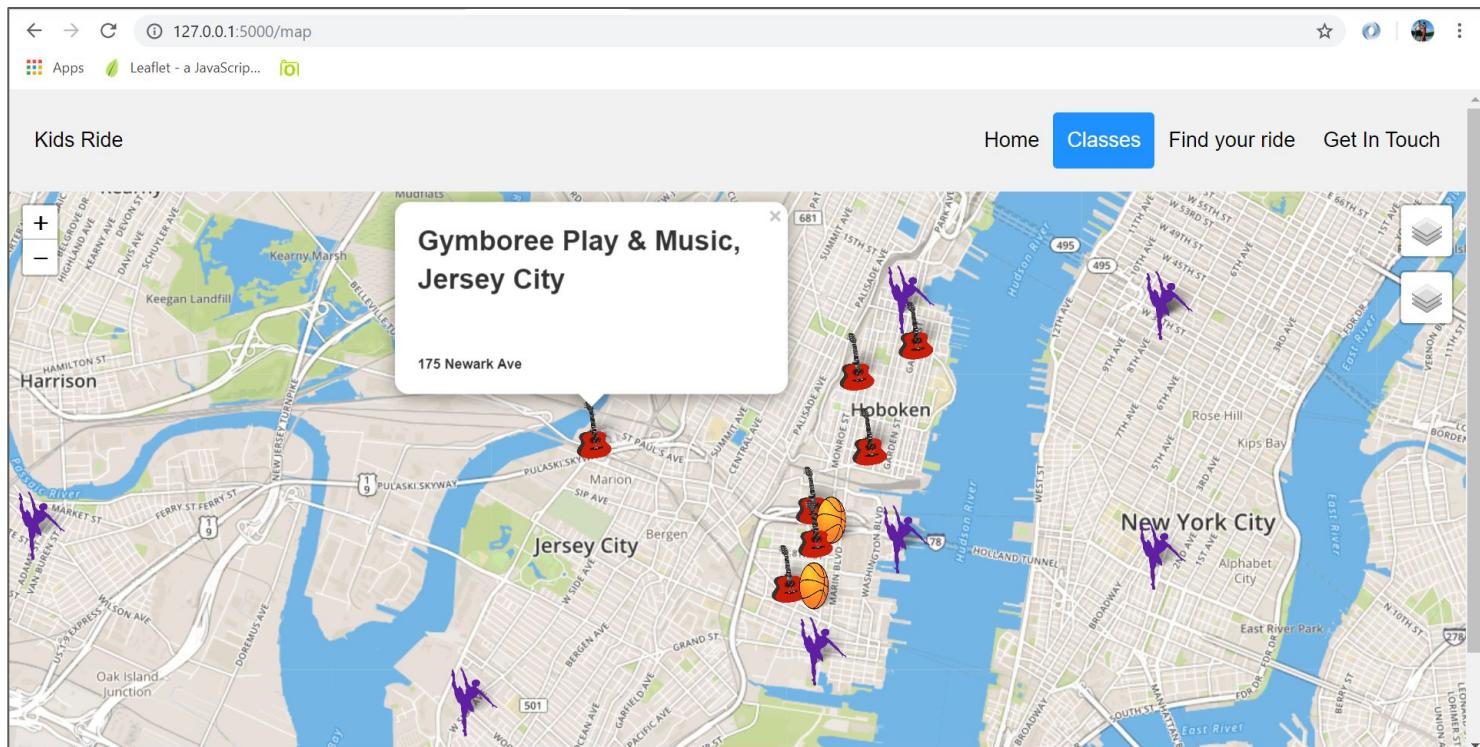
Sports:



Dance:



Popup shows:
Name & Address



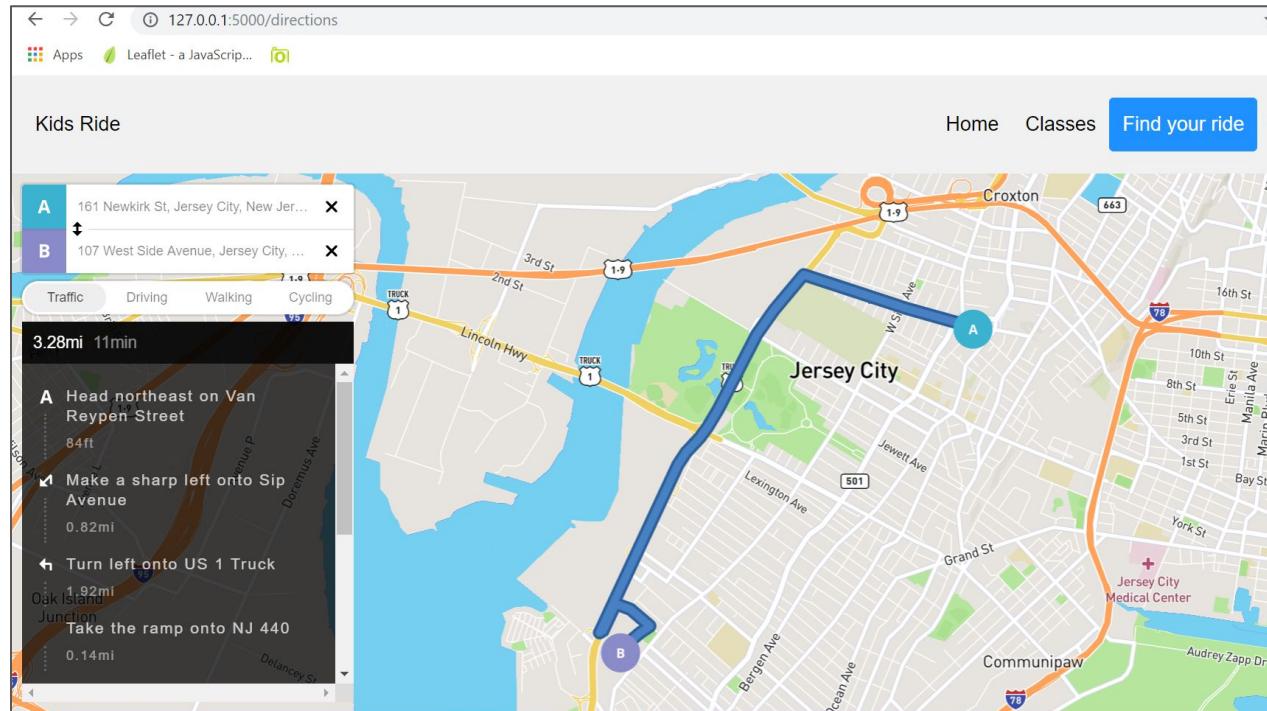
Mapbox GL JS

Geolocator JavaScript that we haven't used in class before

Gives Driving Directions
from Point A to Point B:

1. Traffic
2. Driving
3. Walking
4. Cycling

Gives Distance and ETA

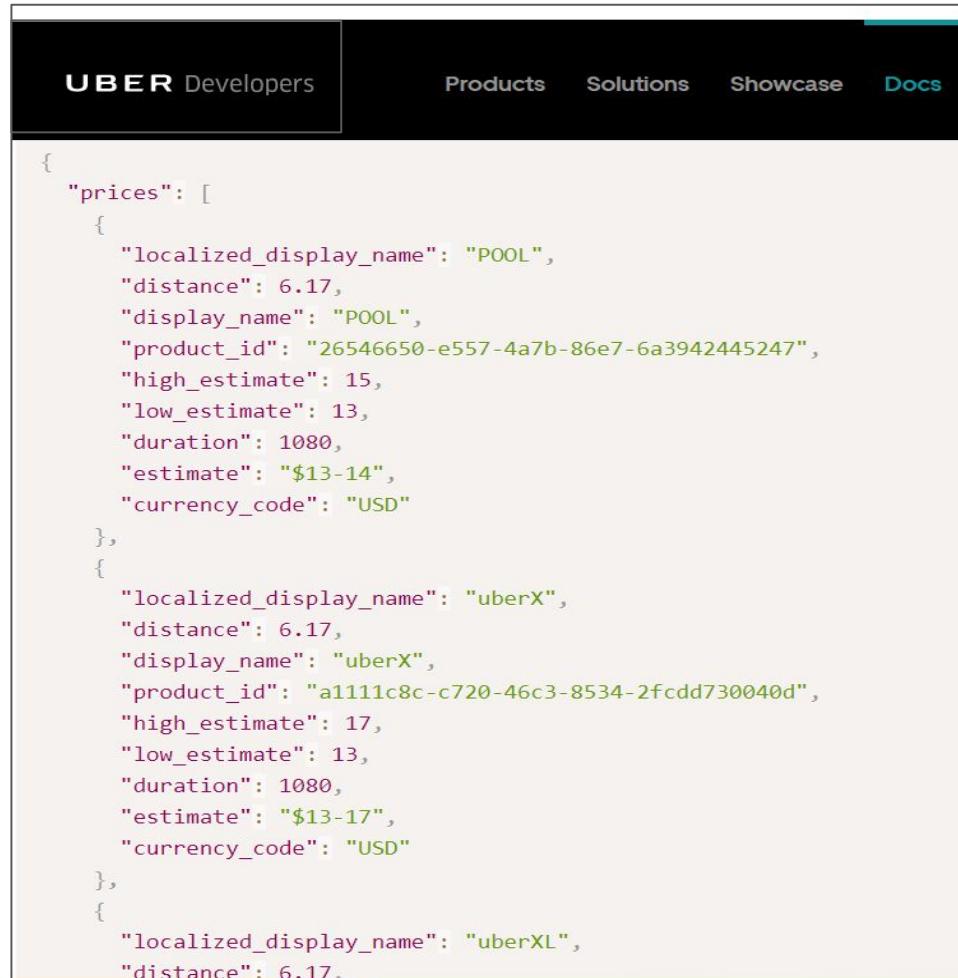


UBER API

Designed to calculate price estimates
from any Location A to any Location B

Query Parameters

Name	Type	Description
start_latitude	float	Latitude component of start location.
start_longitude	float	Longitude component of start location.
end_latitude	float	Latitude component of end location.
end_longitude	float	Longitude component of end location.
seat_count(optional)	int	The number of seats required for uberPOOL. Default and maximum value is 2.



The image shows a screenshot of the Uber Developers website. At the top, there's a navigation bar with the "UBER Developers" logo, followed by links for "Products", "Solutions", "Showcase", and "Docs". Below the navigation, a large code block displays a JSON object representing price estimates for three different Uber products: POOL, uberX, and uberXL. The JSON structure includes fields like localized display names, distances, product IDs, high and low estimates, durations, and currency codes.

```
{
  "prices": [
    {
      "localized_display_name": "POOL",
      "distance": 6.17,
      "display_name": "POOL",
      "product_id": "26546650-e557-4a7b-86e7-6a3942445247",
      "high_estimate": 15,
      "low_estimate": 13,
      "duration": 1080,
      "estimate": "$13-14",
      "currency_code": "USD"
    },
    {
      "localized_display_name": "uberX",
      "distance": 6.17,
      "display_name": "uberX",
      "product_id": "a1111c8c-c720-46c3-8534-2fcdd730040d",
      "high_estimate": 17,
      "low_estimate": 13,
      "duration": 1080,
      "estimate": "$13-17",
      "currency_code": "USD"
    },
    {
      "localized_display_name": "uberXL",
      "distance": 6.17,
      "display_name": "uberXL",
      "product_id": "a1111c8c-c720-46c3-8534-2fcdd730040d",
      "high_estimate": 17,
      "low_estimate": 13,
      "duration": 1080,
      "estimate": "$13-17",
      "currency_code": "USD"
    }
  ]
}
```

Conclusions / Challenges / Next Steps

Demonstrated Full Stack application incorporating all the Project Requirements

Limitations: Webscraping must be completed beforehand in Jupyter Notebook

Webscraping Google is hard-coded:

1. Each category is hard-coded: Dance, Sports, Music Classes
 - a. Need flexibility to search for user-input activities, and add a layer for each “type”
2. The quantity of pages webscraped for each category is hard-coded
 - a. Need flexibility to search & scrape until Google runs out of hits in the area

Next Steps

Integrate UBER API Price Estimate:

- Shows UBER pricing for [LatLngA, LatLngB] to compare pricing for KidsRide
 - Problems with Authentication OAUTH Access Token

Improved User Visualization and Data Analysis:

- Add a User Input asking where the Kid's starting point is
- MapBox GL JS should be used in a way that when you click on a specific Activity location, the website stores that location as Point B
- Add additional layers (each) for Schools, Libraries, Malls and other categories

Live Demo



Go to: Kids Ride Website