Sprint-3 Artifact: Team Pear

Members: Heidi Dye, Noah De Mers, Ian Oh, Israel Sanchez Lara, Bao Lam Le

Features:

- Users can change the time of day for the various pickup times, this will delete an entire row of data that does not pertain to the desired search.(UPDATE and DELETE)
- Users can choose which riding service they want to view search queries for. (UPDATE and DELETE)
- Users can insert a ride by identifying the date, time of pick-up, the location of the pick-up, and service that provided the ride. (INSERT)
    - Date will be in the format “MM/DD/YYYY”
    - Time of pick-up will be in the format “Hour:Minute:00”
    - Location will be in the format “BuildingNumber StreetName“
        - State is assumed to be New York
    - Name of service will be in the format “Name”

GUI:

- Check boxes for the day of week, time of day, and ride sharing service (UPDATE and DELETION)
    - Selecting checkboxes delete rows in search_with_query.csv that do not match the selected checkboxes
        - Updates the client with search_with_query.csv
    - Deselecting checkboxes restores search_with_query.csv to be the same as search.csv and it checks all the checkboxes that are selected and performs the same operation as selecting the checkboxes.
        - Updates the client with search_with_query.csv
- Drop down menu for searching of features (INSERT)
    - Makes a new csv file (search.csv) for the main search operation and (search_with_query.csv) for search results with checkboxes selected
- Inserting new ride information:
    - 3 drop downs will include:
        - A calendar drop down to record the date of the pickup
        - A time drop down the hour and minute of the pickup
        - A drop down comprised of names of acceptable for-hire ride services
    - 1 text field will used for entering an address
- Four text fields labeled for each element of a ride will be shown for the user to enter the date, time of pickup, location, and name of the service offering the ride

    - Upon submission, the new data will be added to the appropriate csv depending on the date and ride service used.

To-Do:

- Finish implementing search queries (or fields of interest)
- Make update/insert/delete operations for search queries.
- Have test cases that will check the update/insert/delete operations.

Completed:

- Parse the csv files into javascript array
- Send data from the dataset to the client

Test Cases

- Feature 1: as a user, I want to see rows that pertain to a certain time of day.
- Test case 1: as a user, in the Data page, I input the time of day for the records and selecting “confirm”.
- Test case 2: as a user, in the Data page, I input an incorrect time of day for the records and selecting “confirm”.

Correct Output: The website should erase all the rows that do not pertain to the desired search.

- Feature 2: as a user, I want to choose which riding service to query for.
- Test case 1: as a user, in the Data page, I input the ride service and selecting “confirm”.
- Test case 1: as a user, in the Data page, I input a non-existing ride service and selecting “confirm”.

Correct Output: The website should update and show all the rows pertaining to the ride service.

- Feature 3: as a user, I can update a ride by entering the date, time, location and name of service.
- Test case 1: as a user, in the Data page, I input the expected data in the correct format and should expect the website to update.

- Test case 2: as a user, in the Data page, I input the data in the wrong format and an error should be thrown asking me to input the data again.

Correct Output: The website should accept the input and update accordingly.

Completed Tasks:

- Heidi: Server-client communication uber dataset, returned fields from dataset
- Heidi: built version 1 of universal uber parsing
- Noah: Implemented parse function for Uber and other FHV data sets
- Noah: Implemented a Comparison function to compare Uber rides to other FHV ride services
- Bao: Parse in data from cab_rides.csv
- Bao: Implemented analytics “Most popular pickup destination in Boston.”
- Bao: Implemented analytics “Most expensive or cheapest service available between Uber and Lyft in Boston.”