Kennedy Uzoho
6-1 Milestone: Dashboard Data Visualizations
CS-340 Client/Server Development

6-1 Milestone: Dashboard Data Visualizations66920i0

1. CRUD Python module that was created in Project One is ready to be used.

   Make sure the MongoDB database is up and running…

   Imports necessary libraries, then

   ```
   from animal_shelter import AnimalShelter
   ```

2. Added the functionality in the callback routine for the instantiation of the CRUD object. User authentication was applied in the CRUD object.

   ```
   #Set the username and password for mongoDB
   username = "user"
   password = "password"

   #instantiate the AnimalShelter object
   shelter = AnimalShelter(username, password)
   ```

3. Updated the code to create an interactive data table on the dashboard which shows an unfiltered view of the Austin Animal Center Outcomes data set.

   To populate the data onto the table, I utilized the CRUD Python module, from Project One, to run a “retrieve all/read/read_all” query and bring in the data from MongoDB.

   ```
   df = pd.DataFrame.from_records(shelter.read_all({}))
   ```

   Added code to filter interactive data table with MongoDB queries such as shelter.read(), update(), or delete()

4. Added a geolocation chart that displays data from the interactive data table in the existing dashboard using the below code snippet,

   ```
   def update_map(viewData):
      dff = pd.DataFrame.from_dict(viewData)
      # Austin TX is at [30.75,-97.48]
      return [
         dl.Map(style={'width': '1000px', 'height': '500px'}, center=[30.75,-97.48], zoom=10,
   children=[
            dl.TileLayer(id="base-layer-id"),
            # Marker with tool tip and popup
            dl.Marker(position=[30.75,-97.48], children=[
               dl.Tooltip(dff.iloc[0,4]),
   ```

```
                dl.Popup([
                    html.H1("Animal Name"),
                    html.P(dff.iloc[1,9])
                ])
            ])
        ])
    ]
```

IPYNB execution.