# GUI Documentation

After starting the application, one should navigate to <http://127.0.0.1:5000/> wherein one will encounter the main page for the GUI for the query application. On this home page, one can upload any and all files needed for the application as well as query the backend for the system. The steps to do so are presented below.

## File Upload

At the top of the page is a file input to populate the backend system.

Graphical user interface, application

Description automatically generated

Clicking on the File button will open a file upload dialogue. From here the user has the option to either upload files one at a time or multiple at the same time. The important criteria here is the order in which the files are uploaded. They must be uploaded Routes, Drivers, then Assignments. Likewise, the name of the routes file must be: “Routes.csv”, the driver file must be named: “Driver.csv”, and the Assignment file must be named “Assignment.csv”. Once the appropriate file(s) have been selected from the file upload dialogue, select import to begin populating the backend with the provided data. If the upload is successful, the user will return to the GUI home page. If an error occurs, one will be redirected to the error page.

## General Querying Information

The user must provide a well formatted string with no excessive whitespace to get proper results, however no field is case sensitive with the exception of the route ID for Query 3. (Since route IDs can contain both numbers and letters). All relevant query information for the selected query with the user provided data is retrieved upon the user hitting the orange submit button.

## Query 1: Given a Driver Name Retrieve all Information and Routes

Clicking on the tab labeled Query 1, the user will be given the option to submit the first and last name of the driver they would like to look up in the system. When the query is successful, the system will respond with page containing all of the driver’s relevant information, including their id, first and last name, age, home city and state. Likewise, a table will populate containing the route number for each route the driver is assigned and the day of the week on which that route departs.

## Query 2: Given a City, Retrieve All Routes Through It

Clicking on the tab entitled Query 2, the user is met with a prompt to provide a city name. Should the user click on the city field and be unable to type, they may simply click on the tab marked Query 2 again and press the tab key 4 times to enter into the City field. From here the user is free to submit a city name to the system. Upon doing so, the system will respond with the route information for all routes entering and exiting that city.

## Query 3: Given a Route ID Retrieve All Route and Driver Information

Clicking on the tab entitled Query 3, the user is given a prompt to provide a route ID. Upon providing this route ID, the system responds by providing the user all route details in large orange text. Under the route details, the user will find a schedule for all of the days of the week the route operates, who the driver of the route for that day of the week is, the driver’s age, and what home city from which that driver hails.

## Query 4: Given Two Cities Retrieve All Direct Routes Between Them

Upon clicking on the tab labeled Query 4, the user will be prompted to enter a Departure and Destination city. Upon entering and submitting this data, the user will be met with a result screen first containing a table with all routes that go directly from the departure city to the destination city with all of their associated route details such as their ID, type, departure time and travel time. Underneath this will be a table containing the direct routes via their ID, the day of the week the leave on, and the details of the driver such as their name, age, and home city.

## Query 5: Given Two Cities and a Day of the Week Retrieve All Direct and Indirect Routes Between Them

Upon clicking the tab entitled Query 5, the user will be prompted for a departure city, destination city, and must select a day of the week. Upon doing so and submitting the data, the user will be brought to a screen containing two tables. The first will be direct routes that leave on the given day of the week to take the user from the departure city to the destination city. Below it will be another table containing indirect routes to take the user from the departure city to the destination city. It is possible these routes will not be fully organized such that the user will have to find the present indirect path between the two cities themselves. However, such a path should be present in the table.

## Error Page

## Graphical user interface, text, application Description automatically generated

This page serves as a general landing page in the event of user issues. For instance, if a user makes a mistake during file upload, they could find themselves redirected to this page. Likewise, if they submit a query which returns no results, they could find themselves at this page. If at any time a user finds themselves at this page, they can simply click the blue “Please Try Again” text to be redirected back to the home page of the application

## Back to Querying Button



On every result page is a “Back to Querying” button that when pressed returns the user to the main page of the application such that they can go perform other queries.