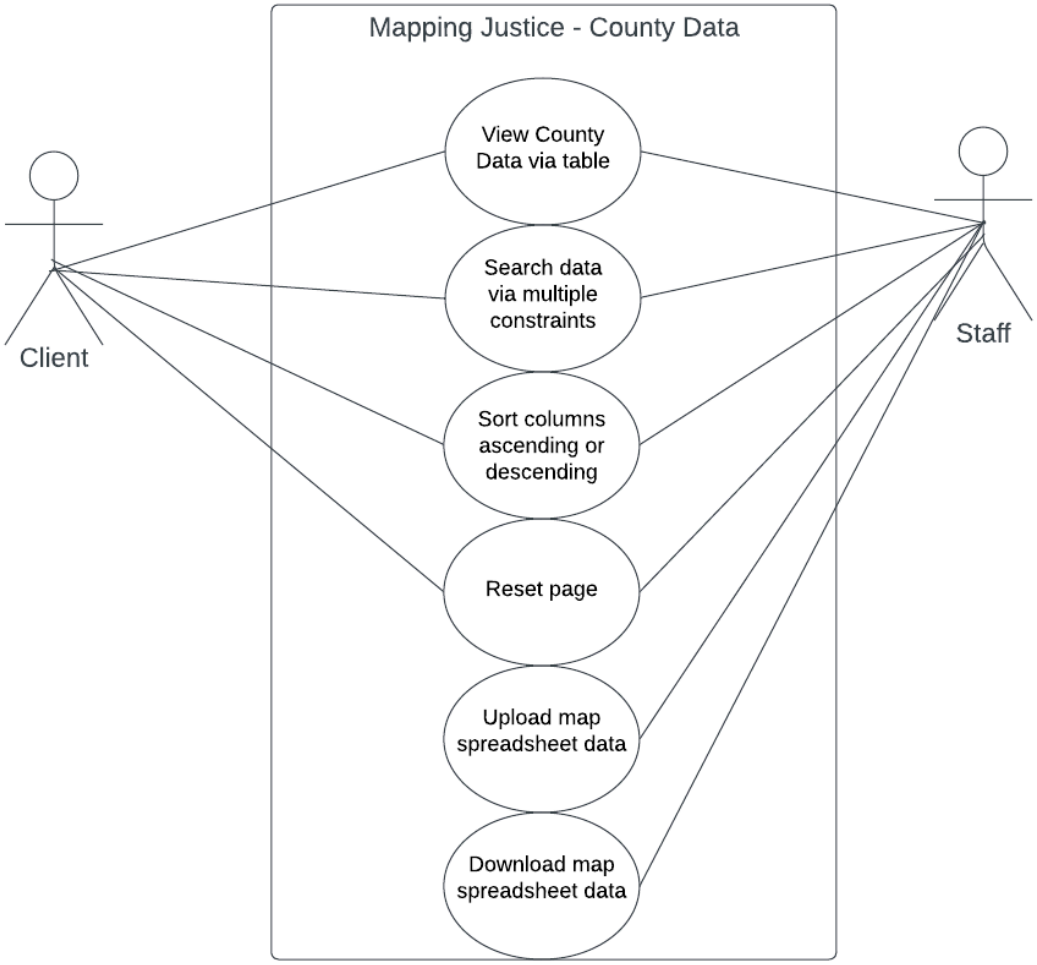
Hannah Champury
CSC 415-02
Assignment 3

Analysis and Design

- **Use Case Diagram**



- **Previous Implementation:**
  - View county data via a table
  - Search data using county name
  - Upload map spreadsheet data
  - Download map spreadsheet data
- **My implementation:**
  - Searching for data using multiple search constraints using textboxes & dropdowns if able to - add feedback for no results found
  - Reset page button that refreshes the page to see all data
  - Sorting each column of data using arrows next to column title in ascending/descending order alphabetically or numerically
- **Additional Functionality:**
  - Limited number of results per page
  - Navigate multiple pages

    - Additional options for exporting (Excel, PDF, CSV)
- **Future Implementation:**
    - Search suggestions
    - Autocomplete functionality
    - Combine data table with the Map County Maps to see side by side
    - Customize what columns they want to see
    - Working design for several types of devices
- **Use Case Description**
    - **Use case:** *Search Data Using Multiple Constraints*
      Primary actor: Any user
      Goal in context: To search for data using one or more constraints
      Preconditions: Login for system as user
      Trigger: Enter constraints & click "Search Button"
      Scenario: (everything goes correctly)
        1. User opens application/web browser
        2. User enters login credentials
        3. User selects "Mapping Justice" then "County Data"
        4. User enters any of the following information: county name, population size, total number of pending cases, number of pending cases for black/white/other, total number of death sentence cases, number of death sentences cases for black/white/other
        5. User clicks "Search"
        6. User receives list of all counties its additional information found represented in a table

      Exceptions: (things that can go wrong)
        1. User can enter invalid information
        2. User can enter information that does not exist: display message
        3. User enters no information & clicks "Search"
        4. File provided is not accessible or unavailable
        5. Server downtime
        6. Database connection issues

      Priority: High Priority for County Map Data

      When Available: At all times

      Frequency of use: Many times per day

      Channel to actor: Via web browser

      Secondary actors: Staff, Support technicians

      Channels to secondary actors: Via web browser

      Open issues: (how business wants to handle situations)

- Performance when handling large amounts of data
- Data integrity
- Usability
- Security for sensitive data & unauthorized access

- **Use case:** *Reset Page*
  Primary actor: Any user
  Goal in context: To reset page to initial page
  Preconditions: Login for system as user
  Trigger: Click “Reset Page” button
  Scenario: (everything goes correctly)
    1. User opens application/web browser
    2. User enters login credentials
    3. User selects “Mapping Justice” then “County Data”
    4. User can search for any specific data at any time
    5. User clicks “Reset Page”
    6. Page will reset showing all data

  Exceptions: (things that can go wrong)
    1. Unsaved changes made by admin
    2. Confirmation of reset

  Priority: Medium Priority for County Map Data

  When Available: At all times

  Frequency of use: Many times per day

  Channel to actor: Via web browser

  Secondary actors: Staff, Support technicians

  Channels to secondary actors: Via web browser

  Open issues: (how business wants to handle situations)

  - Accessibility & Usability: placement & appearance
  - Performance: if larger amount of data imported
  - Making sure sorting/filtering is not affected

- **Use case:** *Sorting by Column*
  Primary actor: Any user
  Goal in context: To sort data based on specific column in ascending/descending order
  Preconditions: Login for system as user
  Trigger: Click arrows next to any column to sort data in table by that column
  Scenario: (everything goes correctly)
    7. User opens application/web browser

8. User enters login credentials
9. User selects “Mapping Justice” then “County Data”
10. User can search for any specific data at any time
11. User clicks on arrow next to column name
12. User will be sorted based on which column arrow pressed

Exceptions: (things that can go wrong)

3. Can only sort by 1 column
4. Change may not be noticeable by users
5. Performance may be affected based on size of dataset & sorting algorithm

Priority: Medium Priority for County Map Data

When Available: At all times

Frequency of use: Many times per day

Channel to actor: Via web browser

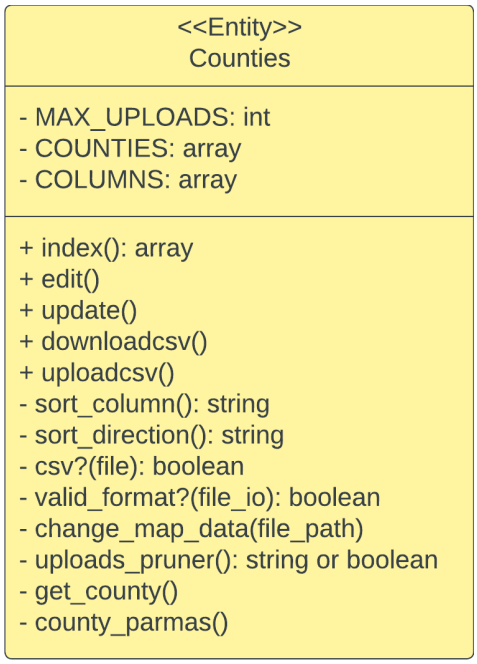
Secondary actors: Staff, Support technicians

Channels to secondary actors: Via web browser

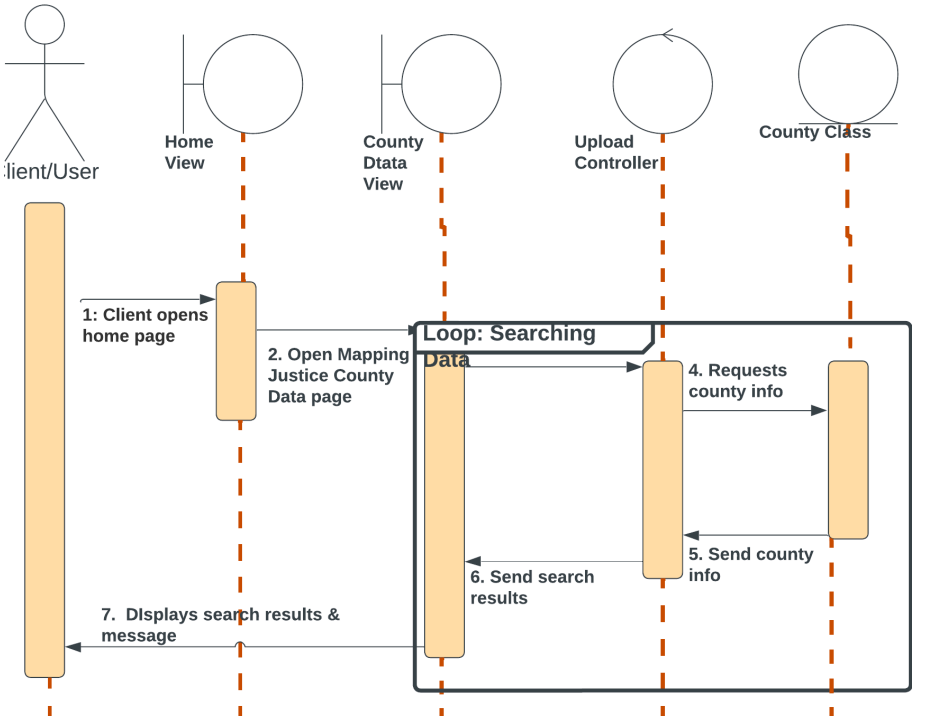
Open issues: (how business wants to handle situations)

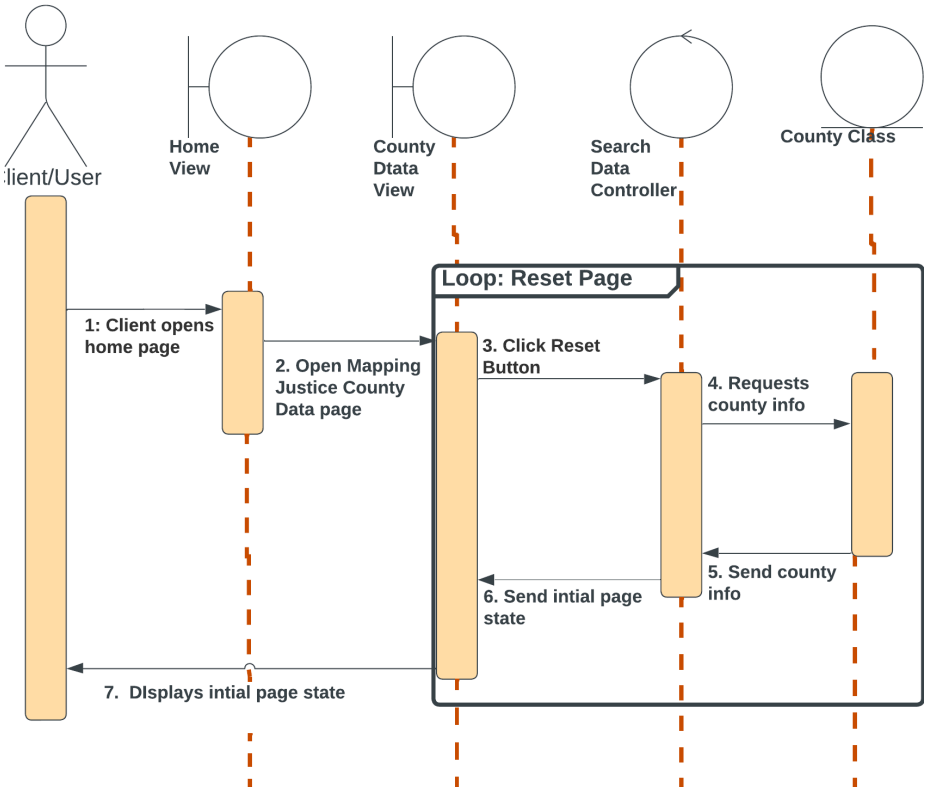
- Accessibility & Usability: placement & appearance
- Performance: if larger amount of data imported
- Retaining sorted information in case of page refresh

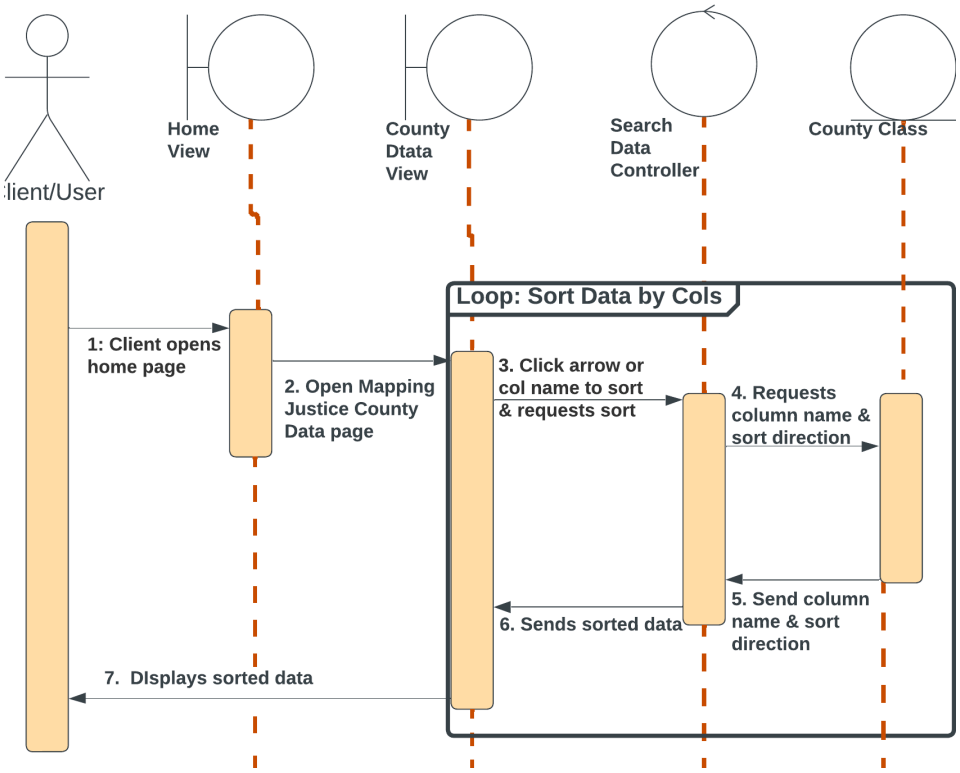
- **Design Class Diagram**



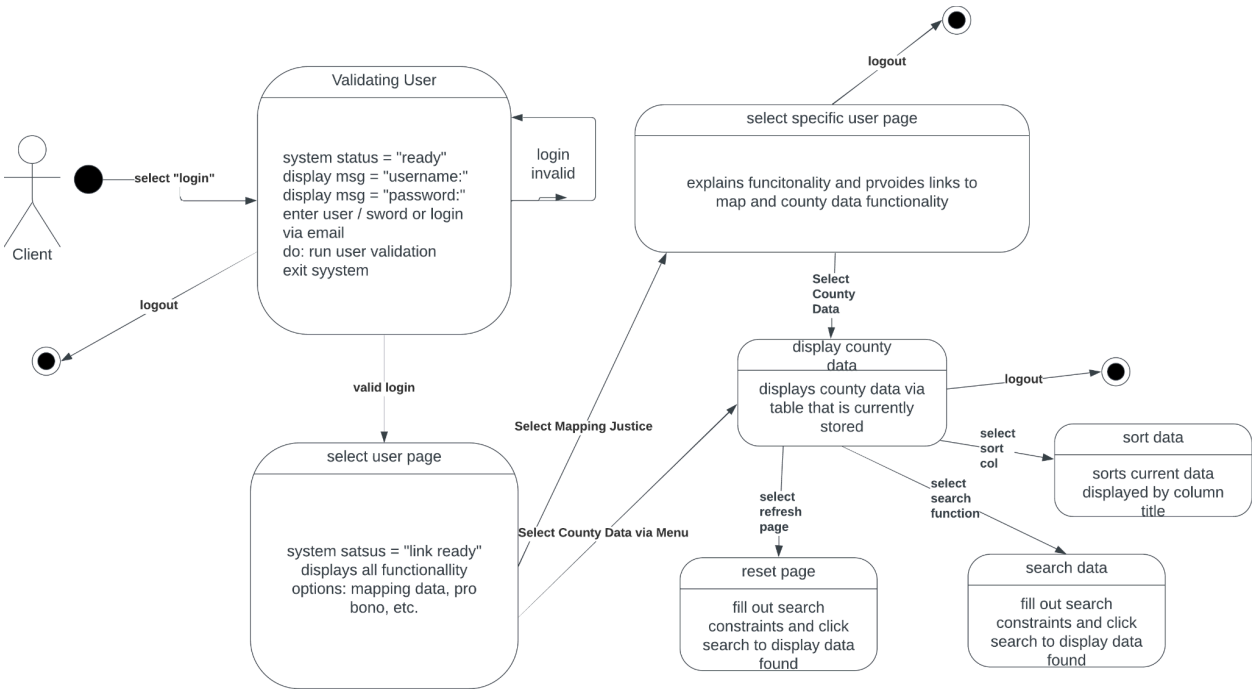
- **System Sequence Diagram**


lient/User
Home View
County Dtata View
Upload Controller
County Class
1: Client opens home page
2. Open Mapping Justice County Data page
Loop: Searching Data
4. Requests county info
5. Send county info
6. Send search results
7. DIsplays search results & message

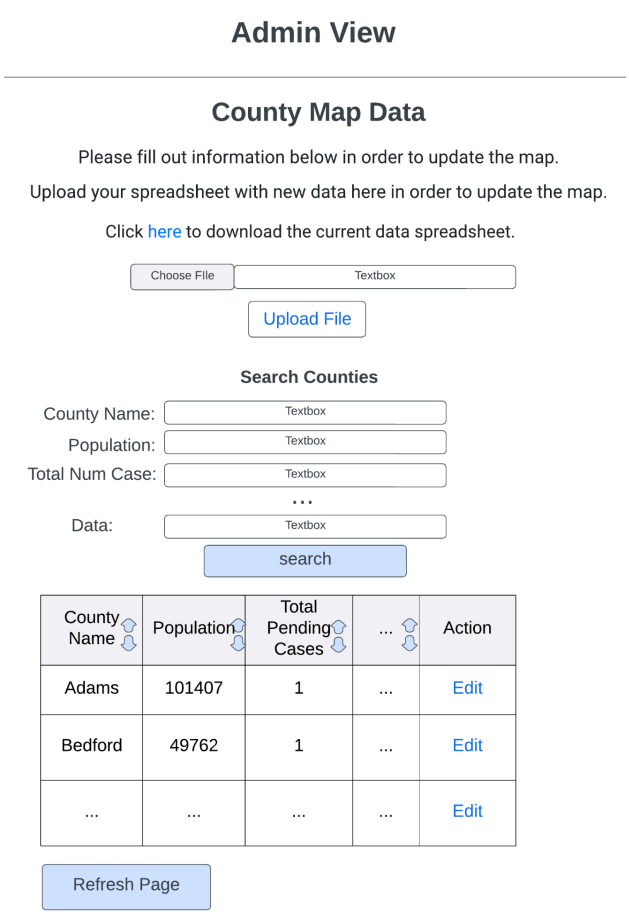

lient/User
Home View
County Dtata View
Search Data Controller
County Class
Loop: Reset Page
1: Client opens home page
2. Open Mapping Justice County Data page
3. Click Reset Button
4. Requests county info
5. Send county info
6. Send intial page state
7. DIsplays intial page state

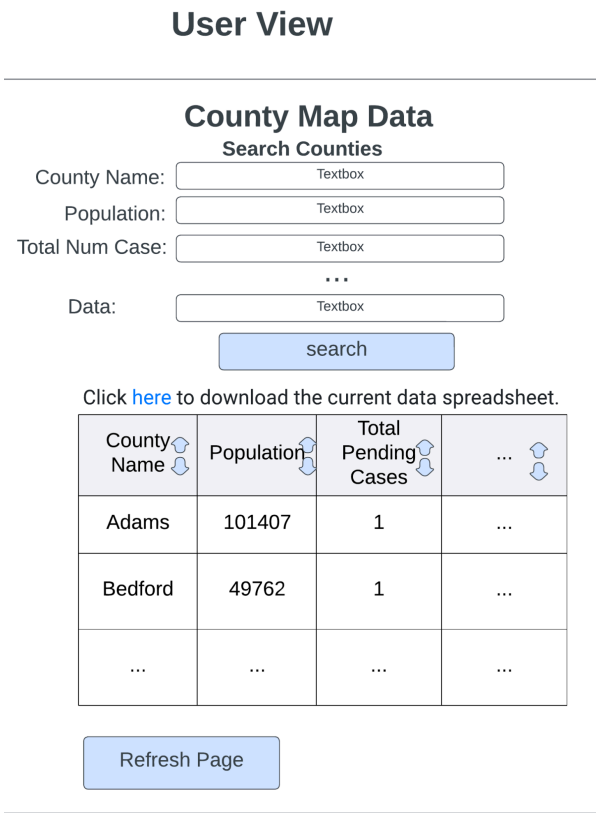


- **System State Diagram**



- **UI Design**





- **Discussion**
  - **Modularity & Encapsulation for Information Hiding & Reuse**
    - Uses classes and methods where each method does one task & be as independent as possible
    - Separate each task/feature into different aspects following the MVC architecture using Views, Controllers, and Models
    - Uses access modifiers: private & public to protect visibility & accessibility of methods and data structures
  - **Elegance & Efficiency of algorithms**
    - Uses appropriate data structures that are well suited for the program
    - Provides modularity and encapsulation
    - Provides information hiding
    - Uses efficient techniques for searching, sorting, & data manipulation
    - Is well documented and commented
- **Test Case Design**

Functionality Tested	Inputs	Expected Output	Actual Output
Search with multiple constraints	Total Pending Cases: 2 Total Current Cases: 1	Lists any counties that have both data inputs	
Search with multiple constraints	Total Pending Cases: 12 Total Current Cases: 10	No counties listed	
Search with multiple constraints	County Name: Adams Population: 101407	Only lists the county Adams	
Search with multiple constraints	County Name: Adams Population: 101408	No counties listed	
Search with multiple constraints	Pending Black: 1 Pending White: 1 Pending Other: 0	Only lists counties with all three	
Search with multiple constraints	Pending Black: 10 Pending White: 7 Pending Other: 0	No counties listed	
Search with multiple constraints	Current Black: 1 Current White: 1 Current Other: 0	Only lists counties with all three	
Search with multiple constraints	Current Black: 5 Current White: 7 Current Other: 6	No counties listed	
Reset Page	Click reset	Page unchanged	
Reset Page	Search data, then click reset	Page resets to initial page	
Sort Cols	Click “County” col title	Sorts table by County in ascending order	
Sort Cols	Click “County” col title again from previous test	Sorts table by County in descending order	
Sort Cols	Click “Population” col title	Sorts table by Population in	

		ascending order	
Sort Cols	Click “Population” col title again from previous test	Sorts table by Populationin descending order	
Sort Cols	Click “Total Pending …” col title	Sorts table by Total Pending in ascending order	
Sort Cols	Click “Pending Black” col title again from previous test	Sorts table by Pending Black in descending order	
Sort Cols	Click “Pending Black” col title	Sorts table by Pending Black ascending order	
Sort Cols	Click “Pending White” col title again from previous test	Sorts table by Pending White in descending order	
Sort Cols	Click “Pending White” col title	Sorts table by Pending White ascending order	
Sort Cols	Click “Pending Other” col title again from previous test	Sorts table by Pending Other in descending order	
Sort Cols	Click “Pending Other” col title	Sorts table by Pending Other ascending order	
Sort Cols	Click “Total Current Cases …” col title	Sorts table by Total Current Cases in ascending order	
Sort Cols	Click “Current Black” col title again from previous test	Sorts table by Current Black in descending order	
Sort Cols	Click “Current Black” col title	Sorts table by Current Black ascending order	
Sort Cols	Click “Current White”	Sorts table by	

	col title again from previous test	Current White in descending order	
Sort Cols	Click “Current White” col title	Sorts table by Current White ascending order	
Sort Cols	Click “Current Other” col title again from previous test	Sorts table by Current Other in descending order	
Sort Cols	Click “Current Other” col title	Sorts table by Current Other ascending order	
Sort Cols	Click “County” col title again from previous test	Sorts table by County in descending order	
Download file	Search data using 1 constraint, click here to download	Downloads file with correct data	
Download file	Search data using multiple constraints, click here to download	Downloads file with correct data	
Download file	Search data using different constraints than prev, click here to download	Downloads file with correct data	