
**Find the event tickets that work for you!**

**Team `JSONmohabir:`** Noah Fichter, Julius Freyra, Anya Keller, and Jason Mohabir

## About

Canoe, our project, is a website that utilizes 5 APIs to help the user find the event tickets that best suit their requirements. The user inputs the type of event they are looking for, the price range, and the numbers of seats, and the site displays a list of the different ticket options. Other parameters can be also selected in dropdown menu to refine the search. The default sorting for the list is by relevance but the user can change that to highest price, lowest price, and date of event. The ticket options that fulfill the input criterion will be displayed in on a paginated output that include the ticket seller, event name, date / time, price and seat number. The results per page are are capped at 10 so if the number of results goes over that limit, there is an option to see the next page of results.

## Roles

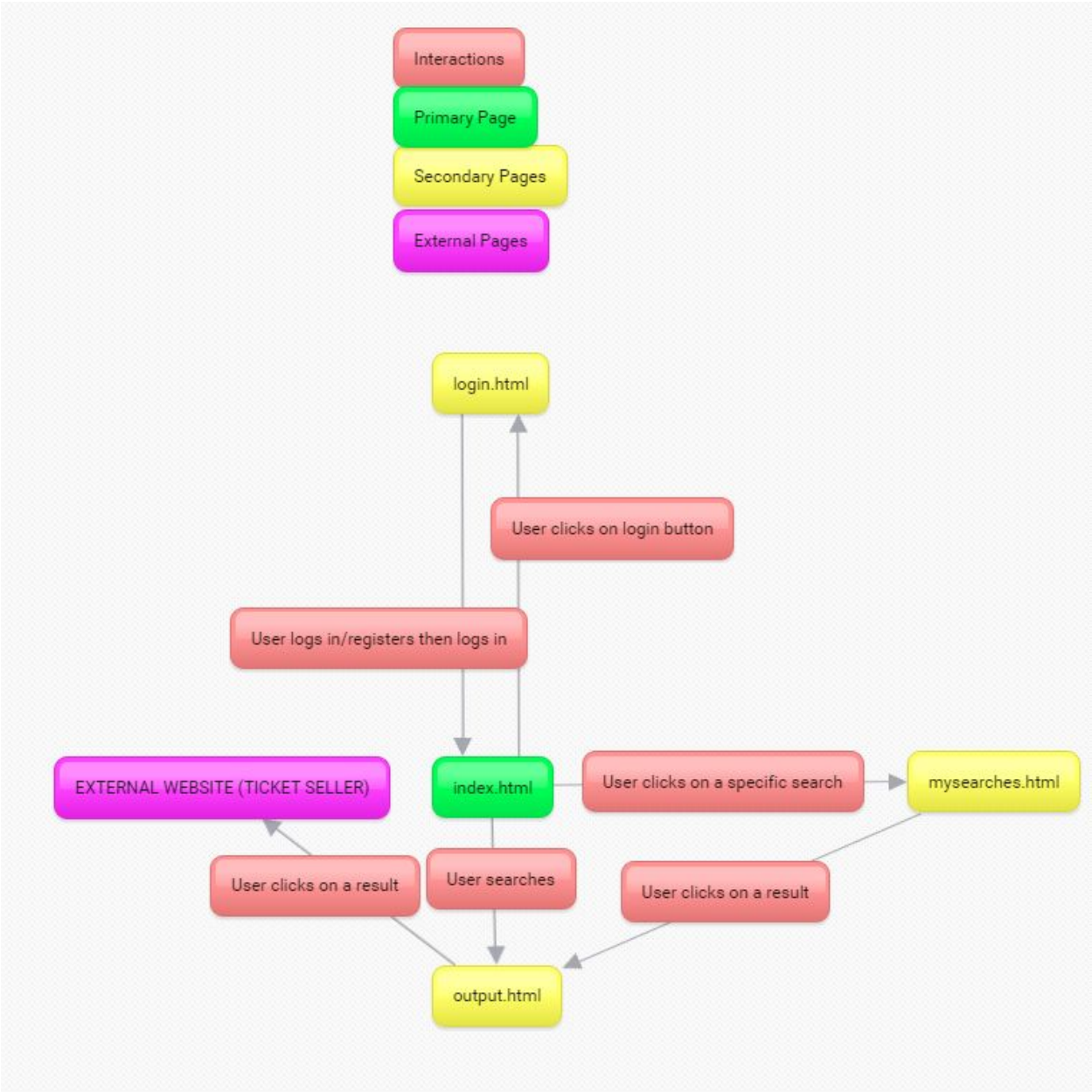
- *Prime Minister* Anya:
- *Foreign Minister* Noah: [Frontend]
  - Foundation
  - Sitemap and Page Mockups
- *Interior Minister* Julius: [Backend]
  - Backend flask routes and util file
  - Component map and list on design doc
- *Communication Minister* Jason: [Backend]
  - Backend flask routes and util files
  - About section of design doc

## Components

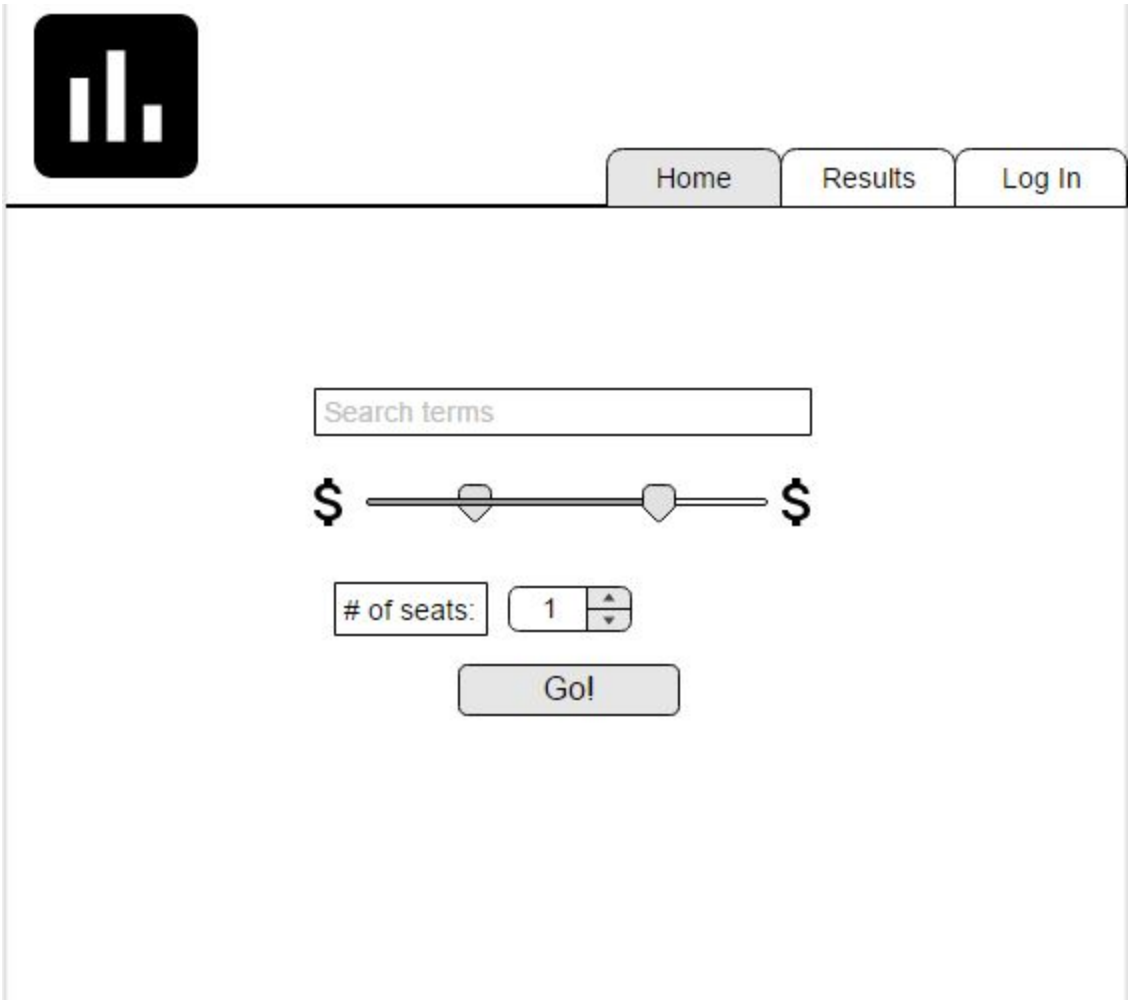
- App.py
    - Backend and Frontend Connector
    - Parses search queries and retrieves ticket information from API Python files
    - Sends ticket information to output.html and mysearches.html to display
    - Parses and authenticates user information when logging in/registering
- Mainpage: index.html
    - Asks for query parameters
        - Type of event
        - Price range
        - Seating range
    - Ranks and sorts results based on query parameters
        - Default sorts by price
        - Option to sort by seating
- Result Display: output.html
    - Displays tickets from each API based on user's search query
- Login Expansion: Possible login creation to save desired settings
    - Pinned events/types of events
    - Login/Register Page: login.html
        - Takes in all user registration info/login info
        - Info parsed/authenticated by app.py in conjunction with usermanager.py
    - Usermanager: usermanager.py; users.db
        - Database and Python file work in conjunction to retrieve/update user info including saved searches
    - Saved Searches Page: mysearches.html
        - Displays a user's saved searches and links to the searches' respective ticket sellers
- APIs: Each has a specific Python file to format sent queries and parse returned information
    - Seatgeek http://platform.seatgeek.com/
    - Eventjoy https://www.eventjoy.com/developer
    - Ticketcity https://www.ticketcity.com/web-service-api.html
    - Ticketleap http://dev.ticketleap.com/
    - Ticketmaster http://developer.ticketmaster.com/

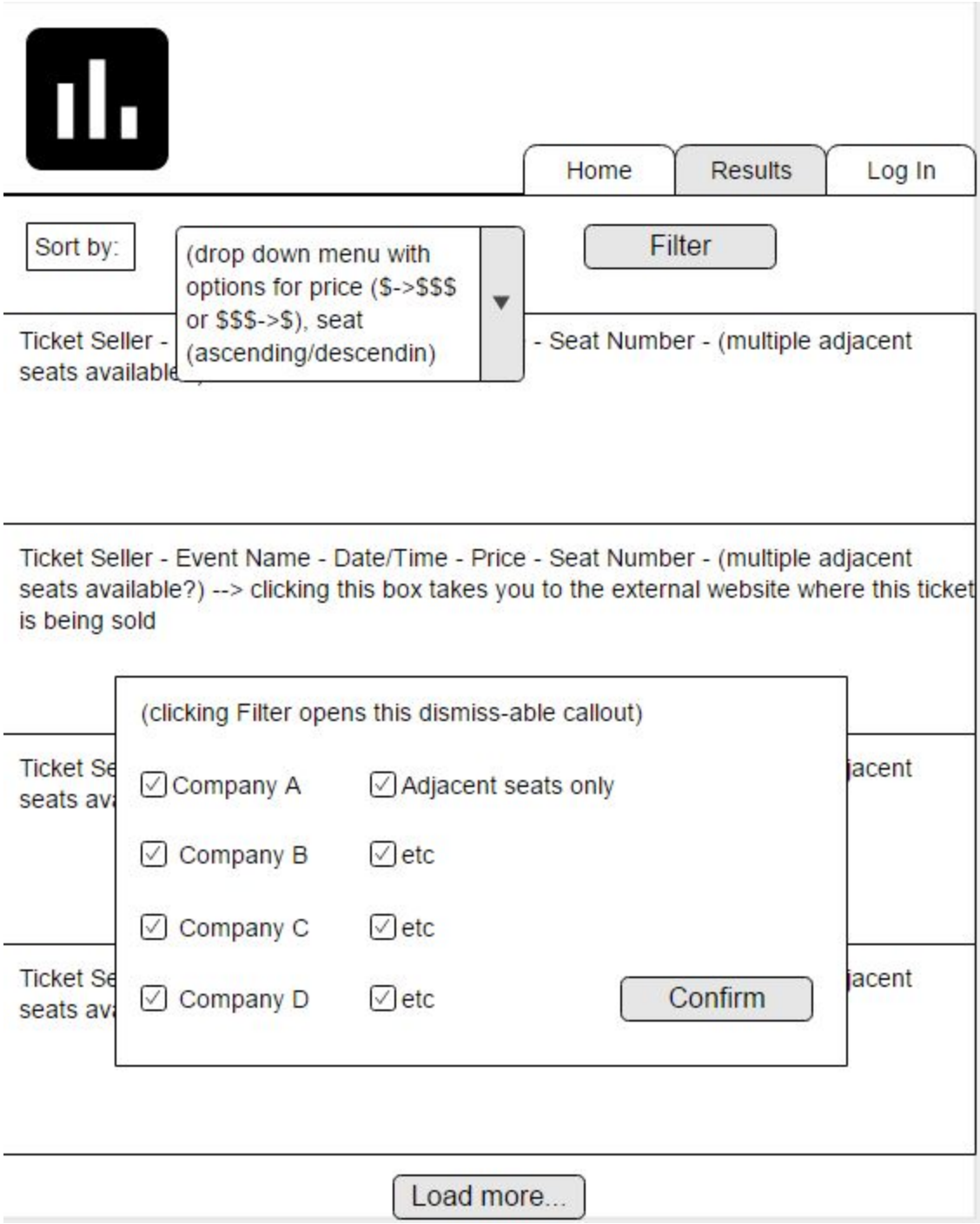
# Component Map

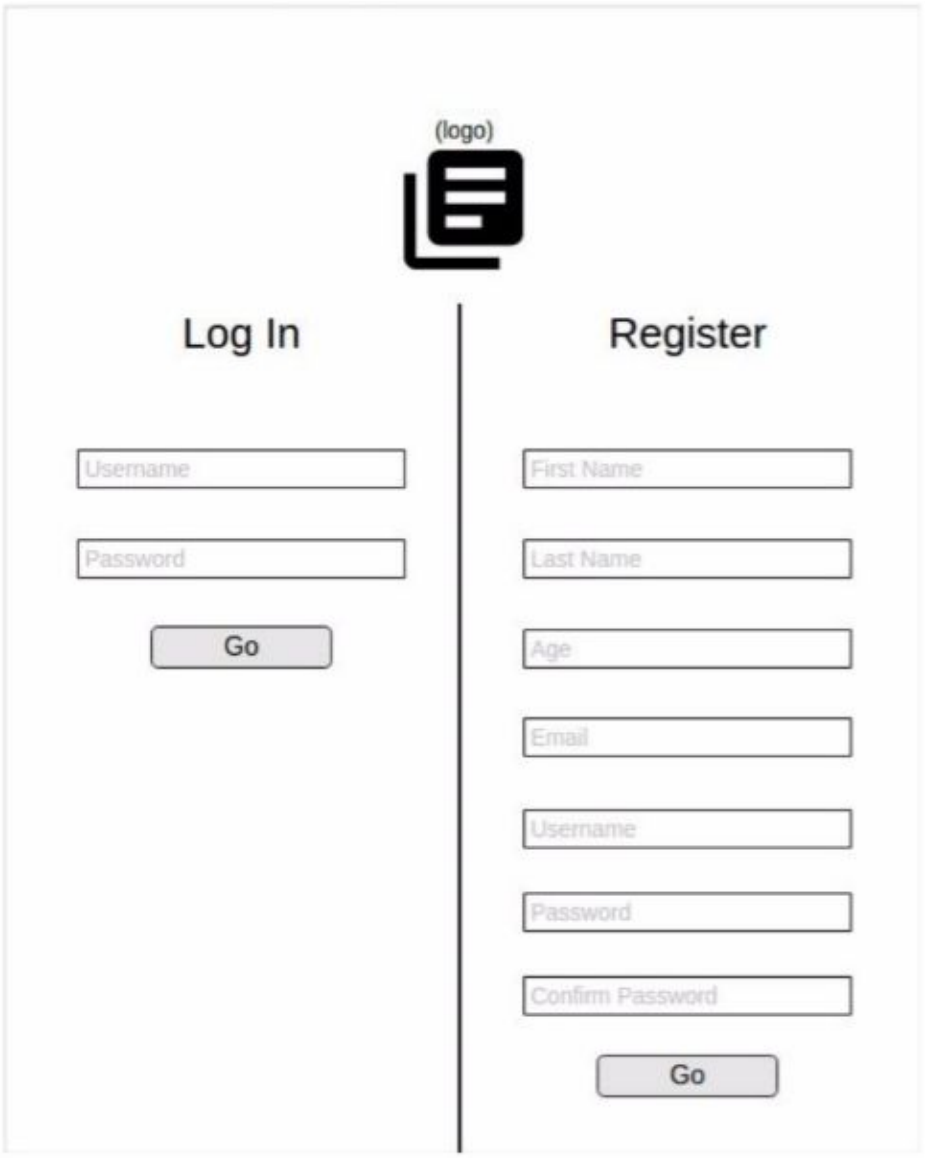
## Site Map (*all user-related stuff is if we have time)



## Page Mockups (*all user-related stuff is if we have time)




Home
Results
Log In
Sort by:
(drop down menu with options for price ($->$$$ or $$$->$), seat (ascending/descendin)
Filter
Ticket Seller -
seats available
- Seat Number - (multiple adjacent
Ticket Seller - Event Name - Date/Time - Price - Seat Number - (multiple adjacent seats available?) --> clicking this box takes you to the external website where this ticket is being sold
(clicking Filter opens this dismiss-able callout)
Company A
Adjacent seats only
Company B
etc
Company C
etc
Company D
etc
Confirm
Load more...


(logo)
Log In
Username
Password
Go
Register
First Name
Last Name
Age
Email
Username
Password
Confirm Password
Go

Welcome back, <user>.

Home | Results | My Searches

Sort by: ◯ Most recent ◉ Oldest

Search date - Search terms - Price range - Seats needed --> Clicking this box links to the results page with these specific search terms

Search date - Search terms - Price range - Seats needed

Search date - Search terms - Price range - Seats needed

Search date - Search terms - Price range - Seats needed

Load more...

**Future Explorations**

- User Functionality
    - Users will be able to create accounts
    - Save preferences for events and keep track of previous searches
    - Send updates to prefered form of communication about new deals
- More APIs
    - Use other APIs that require OAuth in order to provide exhaustive list of deals
    - Develop algorithm for sorting through multiple APIs to extract needed data
- Updates
    - Users will be able to track teams, venues, sports of their choice and get updates on upcoming events