

Punk IPA API Filter

Plan:

- Create react file in bash
- add react file to repo
- clone repo
- import data (reference fetch lessons)
- create structure - similar to ticket tracket or pokedom
- Import structure from ticket tracker and pokedom



Red: App-container
Purple: cardContainer-container
yellow: car-component
Green: navBar-container
Blue: search-component
Mustard: filterButtonContainer-container
Pink: filterButton

Initial Filters:

- High Alcohol (ABV value greater than 6%)
- Classic Range (Was first brewed before 2010)
- High Acidity (pH lower than 4)

Primary Objective: Create a file template using react of webpage without functions

Secondary Objective: Format webpage with card using data file without functions.

New goals:

Large data file with pagination and searching whole array. Obtained from reading the api documentation

Overlay each card with all information (similar to the beatles)

Learn useeffect

Create container for buttons for page selection – need to work on edge cases
Create dropdown for results per page - complete

Create overlay for cards on hover – css practice

Create page overlay for cards on click – css practice

Create button for opening page with beer breakdown. – routing practice

Slider for filters rather than checkboxes

Inline deconstruct props for cleaner code

Done

- Card changes on hover
- Overlay appears on clicking card

To Do:

- Create a landing place for beer information
- Create a Nav-Bar/home Button to navigate back to original page
- Format all information on containers to be informative.
- Use routes to guide to relevant landing places

16:14 13/02/23

Done:

- All linking and basic files are complete

To Do:

- Format all information on containers to be informative.

To Do:

- Format Card- Container
- Format Hover Overlay
- Format Click Overlay
- Format Beer Page
- Format Nav-Bar
- Format WebPage
 - Add Proper Home Button

PageName

Name search

Filter boxes



Page Num / totalpages



card

card

card

card

card

Card container

card

card

card

card

card



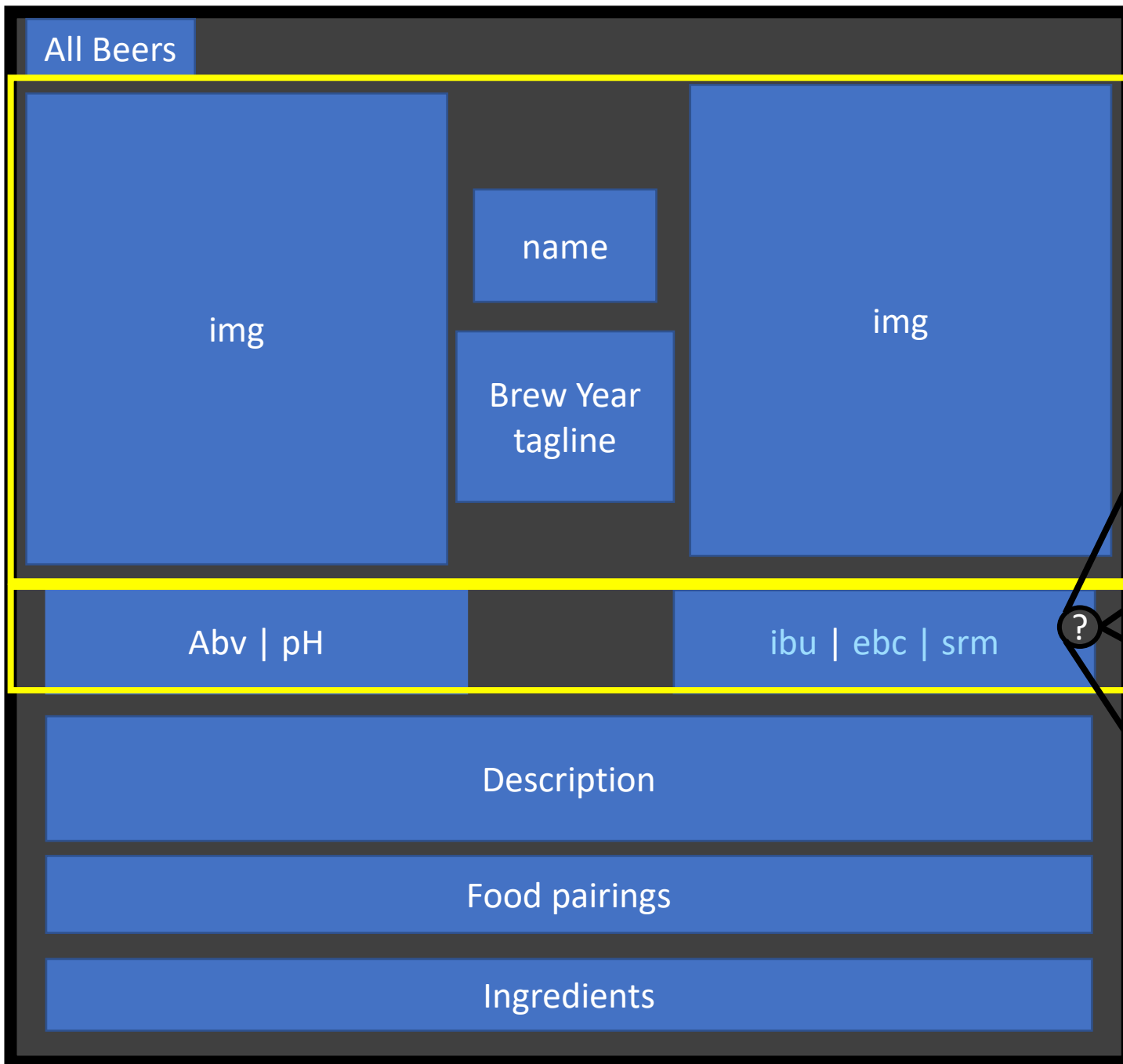
Page Num / totalpages



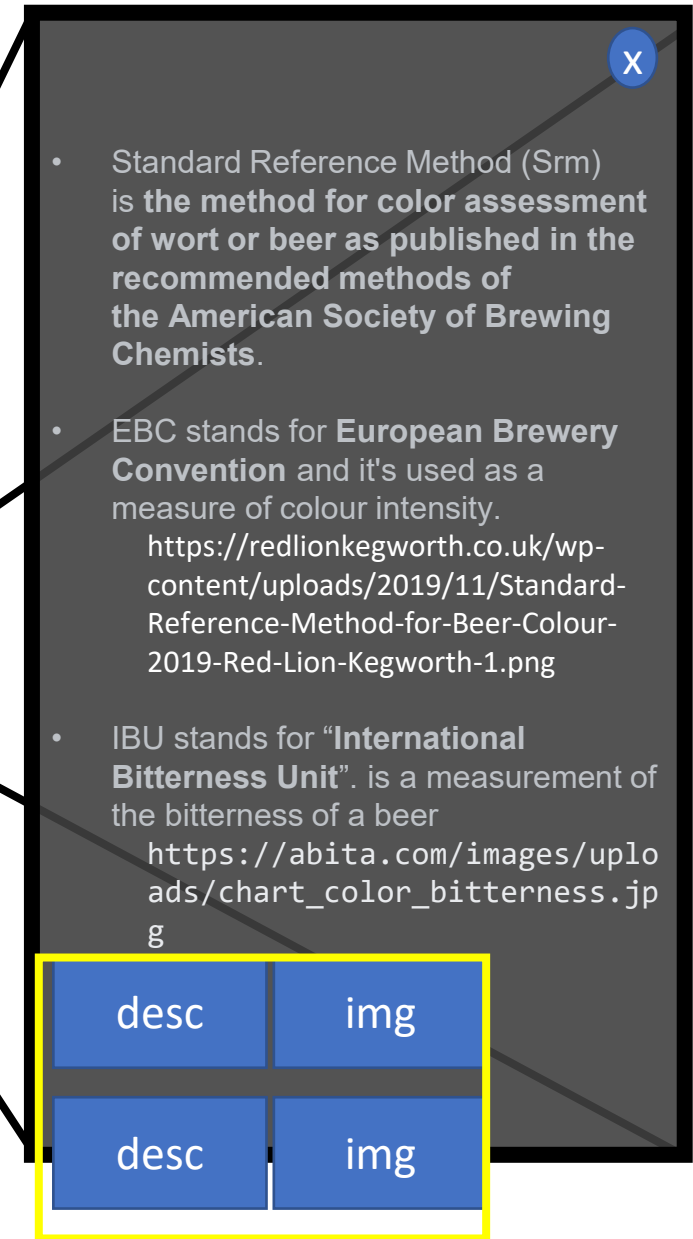
Beer dedicated
route

```
"name": "Jinx Pale Ale",
>tagline": "American Pale Ale.",
>first_brewed": "2018",
>description": "A new 4.7% ABV American pale
ale, Jinx Pale Ale is one of the core Fanzine beer
s, and will feature evolving editions as part of o
ur new subscription model.",
    "image_url": null,
>abv": 4.7,
>ph": 4.6,
Ingredients – to be formatted
>food_pairing": [
    "Monterey Jack stuffed grilled jalapeños",
    "Apple pie"
],
>ibu": 32,
>ebc": 11,
>srm": 10,
Overlay with explanations of acronyms.
```

```
"ingredients": {
  "malt": [
    {
      "name": "Pale Ale", {
        "amount": {
          "value": 3.12,
          "unit": "kilograms"
        }
      },
    {
      "name": "Munich",
      "amount": {
        "value": 0.48,
        "unit": "kilograms"
      }
    },
    {
      "name": "Dextrin Malt",
      "amount": {
        "value": 0.18,
        "unit": "kilograms"
      }
    }
  ],
  "hops": [
    {
      "name": "Simcoe",
      "amount": {
        "value": 6,
        "unit": "grams"
      },
      "add": "65", -when its added
      "attribute": "Bittering"
    },
    {
      "name": "Chinook",
      "amount": {
        "value": 10,
        "unit": "grams"
      },
      "add": "10",
      "attribute": "Flavour"
    },
    {
      "name": "Simcoe",
      "amount": {
        "value": 20,
        "unit": "grams"
      },
      "add": "0",
      "attribute": "Aroma"
    },
    {
      "name": "Chinook",
      "amount": {
        "value": 20,
        "unit": "grams"
      },
      "add": "0",
      "attribute": "Aroma"
    }
  ],
  "yeast": "Wyeast 1056 - American Ale™"
}
```

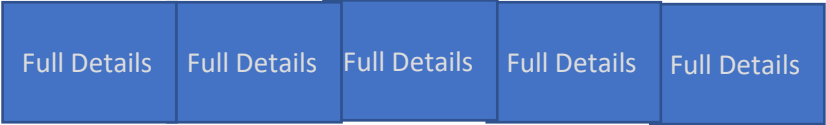



On mobile yellow boxes restructure to vertical



Beer container page overlay

```
"name": "Jinx Pale Ale",
"abv": 4.7,
"first_brewed": "2018",
"description": "A new 4.7% ABV American
pale ale, Jinx Pale Ale is one of the core Fan
zine beers, and will feature evolving edition
s as part of our new subscription model.",
"image_url": null,
"food_pairing": [
    "Monterey Jack stuffed grilled jalape
ños",
    "Apple pie"
],
"ph": 4.6,
```



beer card

```
"name": "Jinx Pale Ale",  
"tagline": "American Pale Ale."  
"image_url": null,  
large  
"abv": 4.7,
```

On mobile clicking
page links to full
website as hoverover
doesn't exist



Beer card hover overlay

```
"name": "Jinx Pale Ale",  
"tagline": "American Pale Ale."  
"description": "A new 4.7% ABV  
American pale ale, Jinx Pale Ale  
is one of the core Fanzine beer  
s, and will feature evolving editi  
ons as part of our new subscrip  
tion model.",  
"image_url": null,  
"abv": 4.7,  
Quick preview | Full Details
```



Webpage logic:

- getAllBeers uses the API to collect all the data from the API pages. The API fetch url is changed dependent on if ABV or Classic buttons are pressed filtering for those criteria. This is done at the App level
- allBeers is the array that is passed into Main and then CardList to be filtered there by the name-input as well as a ph-filter. It is then cut into pieces of length resultsPerPage and displayed.
- The filters to be applied are stored as a useState({object})
 - When a checkbox is click it has an attached function, that dependent on the value of the checkbox, changes the corresponding filter to the inverse of itself as all start false.
 - When a string is entered into the input box the value of the name key in filters is changed.
- Page buttons move through the allBeers array that is filtered then sliced inside of CardList
- arrayLengthFishing is a function that drills into the Main and then CardList to extract the number of elements in the array to be displayed. This means the value of finalPage can be updated dependent on what filters have been applied. Thus the page arrows cant increase passed this value.

Flaws in logic: Beers per page will mean if there's 20 on screen and it is changed to 40 after page 1 it will start at the 40th and skip the 20-40 that were on page 2 but are now on page 1.

All filters set user back to page one.