We need to figure out how to feed it the zip code

Once they select a park, there will be an ID associated for it

Another component listening for select with

Get park and find what city it is in

Use that string to build the URL for the weather API

Need to know when they select the park, so they can know the API, so they can get the weather forecast

will take info from API and turn into HTML

WeatherComponent

- WeatherHTMLConverter

LISTEN: CHANGE EVENT from ParkSelect

{ parkID: <number>
location: <number / zipcode>}

LOGAN BIG BRAIN BOX

DISPATCH EVENT
What user does

IMPORT/EXPORT
For us

Reference associatesDialog, criminalList in Glassdale

associatesDialog is a child of criminalList

criminalList would be our Preview component

associatesDialog would be our park/eateries/attractions list

Similar to criminal.js

ParkComponent

- ParkHTMLConverter

EateryComponent

- EateryHTMLConverter

AttractionComponent

- AttractionHTMLConverter

LISTEN: CLICK EVENT from DetailsButtonComponent

.find
parkSelected
{ parkID: <number> }

Converts HTML for Park Chosen Name

eaterySelected
attractionSelected

- ParkDataProvider
- EateryDataProvider
- AttractionDataProvider
- WeatherDataProvider

DISPATCH: CLICK EVENT
saveItinerarySelected
{ saveItineraryID: <number> }

SaveItineraryButtonComponent

look to show notes in glassdale

Event Hub
<main class="container">

DISPATCH: CHANGE EVENTS
parkSelected
{ parkID: <number> }

eaterySelected
{ eateryID: <number> }

attractionSelected
{ attractionID: <number> }

- ParkSelectComponent
- EaterySelectComponent
- AttractionSelectComponent
- SaveditineraryComponent

DISPATCH: CLICK EVENT
detailButton
{ ID: <number> }

DetailsButtonComponent

IMPORT names and info from Park, Eatery, Attraction

parkSelected
{ parkID: <number> }

eaterySelected
{ eateryID: <number> }

attractionSelected
{ attractionID: <number> }

LISTEN: CHANGE EVENT from ParkSelect, EaterySelect, AttractionSelect

IMPORT names from Park, Eatery, Attraction

parkSelected
{ parkID: <number> }

eaterySelected
{ eateryID: <number> }

attractionSelected
{ attractionID: <number> }

ItineraryPreviewComponent

Similar to criminalList.js

ID comes from the API

Home Page

<main class="container">

<header class="navigation">

- Logo
- Parks
- Attractions
- Eateries
- Headline

??? After MVP

Filter saved intineraries by author name

<section class="filters">

-
  - Parks Dropdown
  - Acadia
  - American Samoa
  - Arches
-
  - Attractions Dropdown
-
  - Eateries Dropdown

<aside class="saveditineraryBox">

Saved Itineraries

- Itinerary 1
- Itinerary 2
- Itinerary 3

<section class="itineraryPreview">

- Park Chosen Name — Details Button
- Attraction Chosen Name — Details Button
- Eatery Chosen Name — Details Button

<section class="">

<h2>Park Forecast Title<h2>

- weather icons images box
- weather icons images box
- weather icons images box

Save Itinerary Button

<Footer> Copyright <Footer>

Itinerary Preview Box

call park/eatery/attraction list component / render

Event Listener in the Parks dropdown

<h2>Park Forecast<h2>
weather icons images

Details Button When Clicked

Dialog Box

Description
Address
Etc.

(For Park, Eatery, and Attraction)

Similar to alibis in Classdale

Parks Page

Parks

- Logo
- Parks
- Attractions Sites
- Eateries

main.js

ParkSelect
EaterySelect
AttractionSelect
ItineraryPreviewComponent
SaveditineraryComponent

ItineraryPreviewComponent

call the list components
ParkComponent
EateryComponent
AttractionComponent
WeatherComponent
DetailsButtonComponent
SaveItineraryButtonComponent

ParkSelectComponent
Will listen to see what park got picked
Will have the dispatch event of click
innerHTML dropdown - Parks Select lives here, we will also .map the parks to get ID

- EaterySelectComponent
- AttractionSelectComponent
- SaveditineraryComponent

Stephanie

ParkComponent
Tells it where we want it to be in HTML (class/ID)

- ParkHTMLConverter
  Converting the data we have from ParksDataProvider into HTML

Logan

EateryComponent

- EateryHTMLConverter

Jess

AttractionComponent

- AttractionHTMLConverter

WeatherComponent

- WeatherHTMLConverter

Rickie

- SaveItineraryButtonComponent

Sosi

- DetailsButtonComponent

Group Program

ParkDataProvider
Gets the data from the API (fetch)
Stores all the parks in an empty array (slice)

- EateryDataProvider
- AttractionDataProvider
- WeatherDataProvider

Database

parks: {
fullName: string
description: string
addresses: string
operatingHours: string
contacts.phoneNumbers.phoneNumber: string
id: number
}

eateries: {
businessName: string,
description: string,
ameneties.playground:
ameneties.diaperFacility:
id: number,
}

attractions: {
name:
description:
state:
id:
}

weather: {
}

The Hmmm Box ???

???
What does weather API need from Parks API
need city name pulled out of address for parks for weather API

???
Parks API will potentially load slow
so we may need to copy parks data into our local database

API/Parks.json

???
need if/else statements to match state to a specific eatery

???
address.stateCode