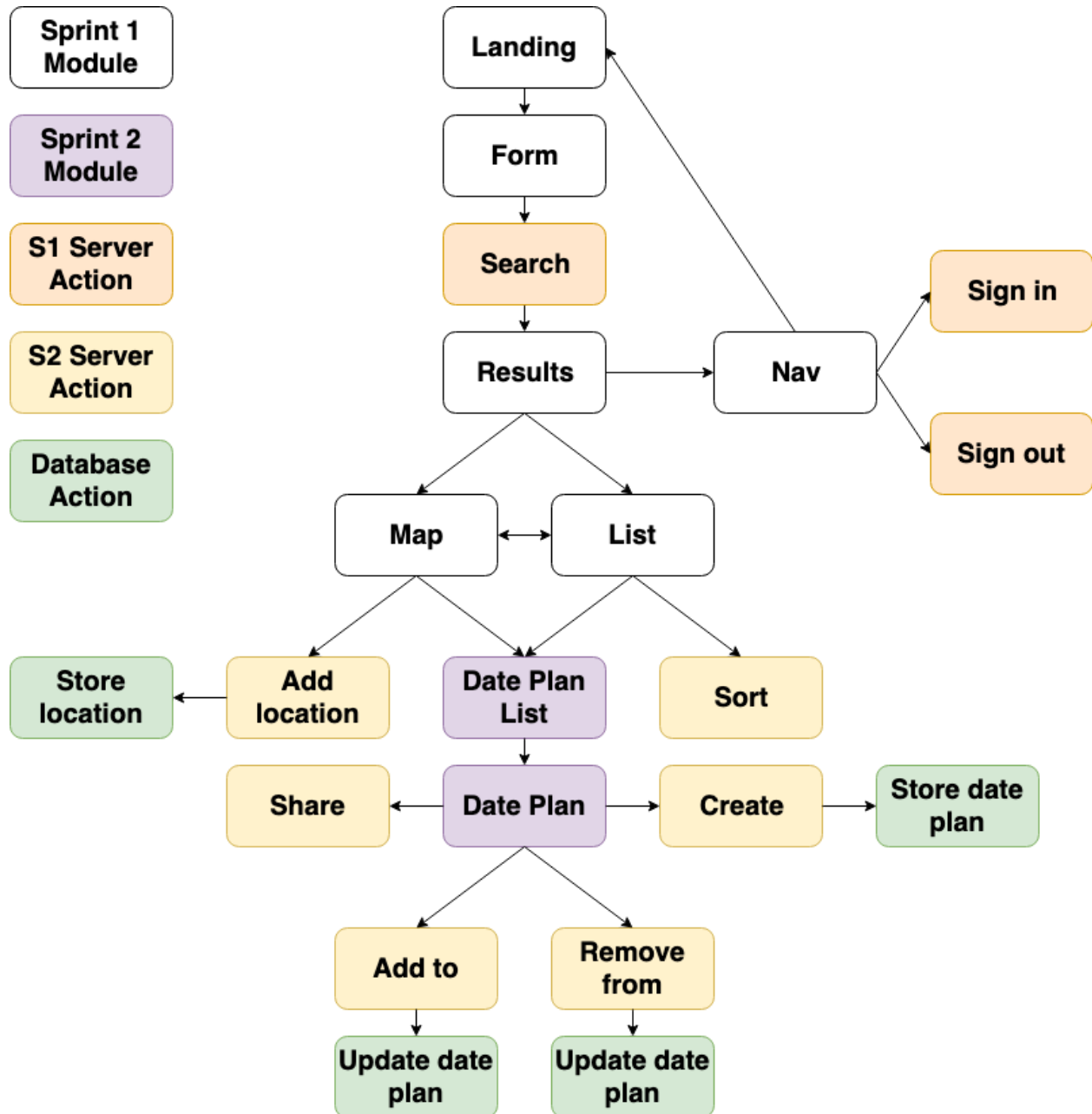# Incremental and Regression Testing - Team 2

Module Map

Modules

Landing Module

The Landing module is just a simple landing page for the application.
Inputs: None
Outputs: Page with button to navigate to Form module
Dependencies: Form, Nav

Form Module

The Form module is a form where the user fills out the information that they want to use for their search query and submits it to the search algorithm.
Inputs: Location, Price Range, Date Type, Min Time, Max Time
Outputs: Page with a form, Object that is sent to Search module
Dependencies: Form, Search

Search Module

The Search module is our main backend algorithm that takes input from the Form module and performs a custom search using the information provided.
Inputs: Object from the Form module
Outputs: Array that contains locations that match the search parameters
Dependencies: Form, Results

Results Module

The Results module is a simple container for the Nav module and either the Map module or the List module.
Inputs: Array from the Search module
Outputs: Page with Nav module and either Map or List module
Dependencies: Search, Nav, Map, List

Nav Module

The Nav module is a navbar that contains a link to the Landing module, tabs to switch between the Map and List modules, and account login/logout functionality.
Inputs: None
Outputs: Navbar with link to Landing, tabs for Map and List, button for logging in and out

Dependencies: Results, Landing

Map Module

The Map module is a Google maps component that shows users the locations and information of the results from the Search module.
Inputs: None
Outputs: Map with pins, Info popups when pin is clicked
Dependencies: Results, List, Date Plan List

List Module

The List module is an alternate view to the Map module that just shows information in a simple list instead.
Inputs: None
Outputs: List with cards for each result, Information on each card about the result
Dependencies: Results, Map, Date Plan List

Date Plan List Module

The Date Plan List module is a container that holds a list of all date plans that a user has created.
Inputs: Date Plans from account
Outputs: Page with list of Date Plans
Dependencies: Map, List, Date Plan

Date Plan

The Date Plan module is an individual date plan that a user has created and can update or delete.
Inputs: Date Plan from account
Outputs: Card with information of the date plan
Dependencies: Date Plan List

Form of Incremental Testing

We followed top-down testing for this project because it made the most sense for our development cycle. We wanted to test everything as we made it, and it made the most sense for us to start developing at the top of the application.

Incremental Testing and Regression Testing

Module	Landing
Incremental Testing - NONE BECAUSE TRIVIAL MODULE	
Regression Testing - NONE BECAUSE TRIVIAL MODULE	

Module	Form		
Incremental Testing			
Defect #	Description	Severity	How to correct
1	The form sent the information formatted wrong and the server could not read the sent information	1	Reformatted the object to match the API standards
Regression Testing			
Defect #	Description	Severity	How to correct
1	Currently there is no validation for entering something that is NaN in the time inputs	2	Create logic check to make sure that users can't enter a letter for the time input
2	There was no check to make sure that the user submitted a valid location	1	Created logic that ensured user used Google Autocomplete to select a location
3	There was no option for the user to select a 4th price point	2	Created a 4th option for the user to select

4	The user was not allowed to select multiple date types	1	Changed the select to allow for multiple

Module	Search		
Incremental Testing - NONE			
Regression Testing			
Defect #	Description	Severity	How to correct
1	Currently if the search coordinates are undefined then the program crashes	1	Fixed function to check if coordinates are
2	Currently if the search location coordinates provided to function are too long or too short then the program crashes	1	Fixed function to check length of coordinates array
3	Currently if the Search location coordinates provided to function are too big or small then the program accepts it and return empty data	1	Fixed function to check bounds of the coordinates
4	Currently if the price point is out of bounds then the program accepts it and return empty data	1	Fixed function to check bounds of the coordinates
5	Currently if the price point provided to function is not defined, the program crashes	1	Fixed function to check if price point is defined

6	Currently if the date types provided to function are not defined then the program crashes	1	Fixed function to check if date types are defined
7	Currently if invalid date types are provided to the function then the program crashes	1	Fixed function to check if all date types are valid
8	Currently, if an invalid array length for the date types is provided to the function, the program crashes	1	Fixed function to check amount of arguments

Module	Results
Incremental Testing - NONE BECAUSE MODULE NOT STARTED	
Regression Testing- NONE BECAUSE MODULE NOT STARTED	

Module	Nav		
Incremental Testing			
Defect #	Description	Severity	How to correct
1	When navigating to the List page, the results were displayed because you can't pass data from sibling to sibling	1	Rewrote the component structure to have Map and List share a parent component that holds the results
2	When navigating to the Map page, the results were displayed because you can't pass data from sibling to sibling	1	Rewrote the component structure to have Map and List share a parent component that holds the results

Regression Testing			
Defect #	Description	Severity	How to correct
1	When you click on the login url, it shows and "invalid callback url".	1	Correctly configured the allowed callback urls on the Auth0 platform.
2	When you logged in, your profile information would not be fetched	1	Properly saved the information returned from Auth0 to the service worker

Module	Map		
Incremental Testing - NONE			
Regression Testing			
Defect #	Description	Severity	How to correct
1	Map was not centered around the location that the user searched	2	Added logic that centered the map on the location that the user searched for

Module	List		
Incremental Testing - NONE			
Regression Testing			
Defect #	Description	Severity	How to correct
1	Currently there is no indication that there are no results in the list	2	Add a simple message that says there were no results
2	Currently there is no error checking to see if the reviews exist	2	Add logic to check if the reviews are undefined

3	Currently there is no error checking to see if the contact information exist	2	Add logic to check if the contact information is undefined

Module	Date Plan List
Incremental Testing - NONE BECAUSE MODULE NOT STARTED	
Regression Testing - NONE BECAUSE MODULE NOT STARTED	

Module	Date Plan
Incremental Testing - NONE BECAUSE TRIVIAL MODULE	
Regression Testing - NONE BECAUSE TRIVIAL MODULE	