Inspection and Unit Testing

Design Inspection

Product: DateNight Client
Author: Development Team

Defect #	Description	Severity	How Corrected
1	Movie times were displayed in a cluttered manner	Important	Fixed the movie showtimes to be shown in a user friendly manner
2	Users were unable to sort results in the list view	Important	Added sorting options to the list view
3	The date plan detail page showed incorrect info when a rating was not present	Important	Added a check and a message that says it has not been rated

Code Inspection

Product: DateNight API
Author: Development Team

Defect #	Description	Severity	How Corrected
1	International Showtimes API stopped sending data because it was down and caused the program to crash.	Critical	Waited for the API to come back up and put undefined checks to prevent the program from crashing when the API was down.
2	Certain venues would not be returned by the google maps api because they did not have a price level and we sent a price level with the request.	Critical	Checked what type of venues had a price level and only sent a price level for certain venues.
3	Tests would timeout because of the increased amount of venues being returned due to the bugfix above.	Important	Increased the timeout on getVenues test .
4	International Showtimes API was returning too many movie times slowing down	Important	Change the time range to only movie times for next 12 hours.

	everything		
5	Tests were returning data as a string instead of JSON and was not allowing access to members in a typical way.	Important	Parsed the string into a JSON object to allow easy access to its members.
6	International Showtimes API would return movies and cinemas without ids or titles and would cause the program to crash	Critical	Checked movie and cinemas to make sure they had all the fields required for showing movie times.
7	Venues object did not contain price level or open hours so it was not showing up on the frontend	Important	Added price level and open hours to the venues object so it shows up on the frontend.
8	Delete or rate a plan did not check the user submitting the request so any user could delete any plan in the database	Critical	Added a where clause into the db to check that the user has access to the plan being changed.
9	Delete or rate a plan did not check if a plan was found. This causes success to be returned when no plan was changed	Important	Check the DB response to see if one plan was changed. Return 404 if no plan was changed.
10	Plan detail page did not load content, even when it was fetching correctly	Critical	Added a logic statement to check if the fetch was done, the code was rendering too early.
11	Token returned from auth0 was incorrect due to no audience in the config file	Critical	Added audience to config file on frontend
12	Crowdsourced venue SQL queries were not properly escaped and posed an SQL Injection risk	Critical	Used an SQL escaping API to inject parameters safely
13	Crowdsourced venue query "great circle" formula was implemented using angles in degrees when the formula expected radians causing inaccuracies	Important	Converted all latitude and longitude values to radians before querying
14	API call to save a new dateplan was failing due to no authorization token specified	Critical	Added an authorization token taken from auth0

Unit Testing

Module: Map
Product: DateNight Client, DateNight API
Date: 10/10/19
Author: Jon Huber, Emanuel Pituch, Trevor Hill

Defect #	Description	Input	Output	Severity	How Corrected
1	Test getVenues endpoint when there is theatres selected	Submit form with entertainment selected and a location with a movie theatre	Expected: Movie showtimes show up in JSON object returned by endpoint Actual: No movie showtimes show up in JSON object	Critical	Added function to look up movie times if a place is a movie theater
2	Test getVenues endpoint when there are no theatres selected or nearby	Submit form with entertainment not selected or a location that does not have a movie theatre	Expected: Movie theater object does not exist inside non-movie theater places Actual: Movie theater object showing up anyway	Important	Added checking to see if a place is a movie theater and only add movie times if it is
3	Movies are shown on the frontend when a movie theater is clicked	Submit a form, find a movie theater in the results on the map, click it	Expected: A popup appears with movies showing and their times Actual: A popup appears with invalid/blank data	Important	Added check to see if movies were returned from the backend algorithm
4	User can submit a crowdsourced event to the backend algorithm	Submit a form, click on the button to submit a place, click a location, confirm	Expected: On the next search, the user will see that the location they have added appears Actual: On the next search, the user would not see the location they have added appear	Critical	Added button functionality and connected to the endpoint on the backend for sending the placeId
5	Test createVenue endpoint when placeID is undefined	Post request to /getVenues with an undefined body	Expected: 400 Error with message 'Undefined Place ID' Actual: 200 Success with message 'OK'	Critical	Added return in undefined value check to prevent code from continuing

6	Test createVenue endpoint when placeID is invalid	Post request to /getVenues with an invalid placeid	Expected: 400 Error with message Invalid Place ID' Actual: 200 Success with message 'OK'	Critical	Added return in valid value check to prevent code from continuing
7	Test createVenue endpoint when placeID is valid	Post request to /getVenues with a valid place ID	Expected: 200 Success with message 'OK' Actual: 400 Error with message Invalid Place ID'	Critical	Wrapped await api call in parenthesis before inline referencing fields with .json()
8	Test getVenues endpoint when crowdsourced venues are in range	Create a crowdsourced venue, then search for venues in that same area	Expected: Pin of created venue will populate on map. Actual: Pin of created venue does not appear on map	Critical	Flattened array of rows returned from database so that the API call recieves valid strings and not js objects.

Module: List
Product: DateNight Client
Date: 10/10/19
Author: Collin Li, Ray Truong

Defect #	Description	Input	Output	Severity	How Corrected
1	Sorting cards with no rating causes application to crash	Choose to sort cards by rating after making a query that returns at least 1 entry without a given rating	Expected: Cards appear in sorted order with unrated cards at the bottom Actual: list view is empty, crashes	Critical	Add a check in the sorting comparator function for non existing ratings
2	List view crashes when attempting to view plans on Sunday	Make any query that returns at least one event on a Sunday	Expected: list view of events Actual: empty view, error logged	Critical	Adjust indexing of front end to map Sunday to 0 instead of -1
3	Sorted cards appear in wrong sorting option	Select a sorting option	Expected: cards appear in sorted order according to selected option	Critical	Inverted logic of selection

			Actual: cards appear in sorted order according to a non selected option		
4	Sorted cards with no rating is left unsorted	Select a sorting option by rating or price and make a query with at least one result that contains no rating or price	Expected: unrated and no priced cards are filtered to the bottom Actual: unrated and no priced cards are left in place	Low	Added logic to comparator function to always mark unsorted cards as less or equal value to compared cards
5	Price values were not accurate because of an off by one bug	Select a price value of $ on the form	Expected: the corresponding number of $	low	Shifted the reference array to the right to account for off by one bug

Module: Date Plan List
Product: DateNight Client, DateNight API
Date: 10/10/19
Author: Jon Huber, Ray Truong, Andy Plank

Defect #	Description	Input	Output	Severity	How Corrected
1	Null activity array caused app to crash when formatting	Call format() using an array of activities that is empty	Expected: Return null Actual: Error is thrown	Critical	Added an if statement to check for null
2	Phone number always fails inspection due to dashes and () in the number	Call checkAct() with phone: as a valid phone number	Expected: checkAct() returns true Actual: checkAct() returns false	Critical	Changed checkAct to properly match a phone number
3	Boundary test for string rating	Call checkPlan() with a string as the rating	Expected: checkPlan() returns false Actual: checkPlan() returns true	Important	Changed checkPlan to check for isNaN to verify that it is a number
5	Deleting multiple plans	Delete	Expected: selected card	Critical	Changed mutation of

	causes deleted plans to reappear	multiple cards	is deleted Actual: selected card is deleted and deleted card reappears		card array to mutate state instead

Module: Date Plan
Product: DateNight Client, DateNight API
Date: 10/10/19
Author: Collin Li, Andy Plank

Defect #	Description	Input	Output	Severity	How Corrected
1	Apostrophe causes failure to insert	Post to /save with an activity with an apostrophe in the name	Expected: Row is inserted into the db Actual: Query fails and is not inserted	Important	Delimited all queries with an apostrophe
2	View dateplan btn is enabled when user has not specified a dateplan name	Go to list, and then 'View Dateplan'	Expected: View dateplan btn is disabled	important	Added a field to disabled the button until a name for the dateplan was created
3	Creating a name for the dateplan can be blank	Go to list, and then start dateplan. Do not type in anything, and then click create	Expected: The modal should not disappear since a no value was entered	important	Added a check to see if the text field was not blank.
4	An activity is added with no confirmation	Create a dateplan name and then add a random activity	Expected: a dialog should pop up prompting the user to confirm their selection	important	Added a dialog for the user to click confirm if confident in their decision
5	Clicking save in view plans did not pass in an auth token	Click save in view plans	Expected: Save should only work if the user is authorized	critical	Retrieved the user's auth token and passed it in the post request's body