

Localendar

Unit Test Descriptions

11/25/2022

How to run tests:

- Go to frontend > Home.jsx and uncomment the line at the bottom of the file
- Go to frontend > Events.jsx and uncomment the line at the bottom of the file
- In the main folder, run "npm test" to run all test files, or "npm test <test file>" to run a specific test file
- If you receive the error message "Cannot find module <module name> from <test file>", enter "npm install --save <module name>" to install the missing module

User Story 1: As a user, I want the ability to sign up for and log into a Localendar account so I can have my own profile.

Modules being tested:

- backend/src/auth.js

Post

- Equivalence classes

Equivalence Classes	Description
EC _{post}	User was successfully written to the database
EC _{error}	User failed to be written to the database

- Tests for Equivalence classes

Equivalence Class	Test Case	Expected Output
EC _{post}	Input a valid JSON	Valid user input
EC _{error}	Invalid JSON	User

Get

- Equivalence classes

Equivalence Classes	Description
EC _{get}	User was successfully retrieved from the database
EC _{error}	User failed to be retrieved from the database

- Tests for Equivalence classes

Equivalence Class	Test Case	Expected Output
EC _{post}	Input a valid JSON	First entry stored in the user database is test@test.com
EC _{error}	Invalid JSON	Error

User Story 3: As a user, I want the ability to view local events.

Modules being tested:

- Home.jsx > generateEvents(events)

generateEvents(events):

- Equivalence classes

Equivalence Classes	Description
EC _{allEvents}	Show all events, including past ones
EC _{futureEvents}	Only show upcoming events

- Tests for equivalence classes

Equivalence Class	Test Case	Expected Output
EC _{allEvents}	Event Date: 2021-10-12 Event Time: 11:45:10	event.view = true
EC _{futureEvents}	Event Date: 2022-11-19 Event Time: 04:30:12	event.view = false

User Story 4: As an event host, I want the ability to post events so that people can see my event.

Modules being tested:

- backend/src/event.js > POST

POST:

- Equivalence classes

Equivalence Classes	Description
EC _{post}	Event was successfully written to the database
EC _{error}	Event failed to be written to the database

- Tests for Equivalence classes

Equivalence Class	Test Case	Expected Output
EC _{post}	Input a valid JSON	Valid event input
EC _{error}	Invalid JSON	Error

User Story 5: As an event host, I want users to see an accurate location of my event so they know where to go, and check if other inputs are valid before posting

Modules being tested:

- EventForm.jsx > Submit Button(check if button is disabled or not)

Submit Button

- Equivalence classes:

Equivalence Classes	Description
EC _{normal}	Normal, all inputs are valid
EC _{missing one input}	One of the inputs is not valid or empty
EC _{invalid time format}	The input entered for time is invalid or not in the correct format
EC _{invalid date format}	The input entered for date is invalid or not in the correct format
EC _{invalid location}	Location does not exist or is not in the right format to get a valid latitude and longitude
EC _{missing multiple inputs}	Multiple inputs are either invalid or empty

- Tests for equivalence classes:

Equivalence Classes	Test Case	Expected Output
EC _{normal}	Input a valid JSON	Submit button enabled
EC _{missing one input}	name: "", date: '2023-01-01', time: '12:00:00', location: '6313 Evergreen Ave, Las Vegas, NV, 89107', lat: 36.162119, long: -115.2312065, eventdescription: 'testing new event',	Submit button disabled

	<i>*missing name input</i>	
EC _{invalid time format}	name: 'Event Name', date: '2023-01-01', time: '12am', location: '6313 Evergreen Ave, Las Vegas, NV, 89107', lat: 36.162119, long: -115.2312065, eventdescription: 'testing new event', <i>*invalid time format 12am</i>	Submit button disabled
EC _{invalid date format}	name: 'Event Name', date: 'Dec 23rd', time: '12:00:00', location: '6313 Evergreen Ave, Las Vegas, NV, 89107', lat: 36.162119, long: -115.2312065, eventdescription: 'testing new event', <i>*invalid date format 'Dec 23rd'</i>	Submit button disabled
EC _{invalid location}	name: 'Event Name', date: '2023-01-01', time: '12:00:00', location: 'house', lat:, long: , eventdescription: 'testing new event', <i>* invalid location 'house'</i>	Submit button disabled
EC _{missing multiple inputs}	name: ", date: ", time: ", location: '6313 Evergreen Ave, Las Vegas, NV, 89107', lat: 36.162119, long: -115.2312065, eventdescription: 'testing	Submit button disabled

	new event', <i>*missing name, date, and time</i>	
--	---	--

User Story 6: As a user, I want to see the events sorted by distance so that I can see which ones are closest to me.

Modules being tested:

- Home.jsx > handleSelect()
- Home.jsx > calculateDistanceInMiles(userLatitude, userLongitude, eventLatitude, eventLongitude)

handleSelect():

- Equivalence classes:

Equivalence Classes	Description
EC _{normal}	Normal, valid address
EC _{at-event}	Same address as an event
EC _{no-data}	No data in the event database to sort

- Tests for equivalence classes:

Equivalence Class	Test Case	Expected Output
EC _{normal}	161 Centereach Mall, Centereach, NY 11720	Valid sort
EC _{normal}	10655 Scripps Poway Pkwy, San Diego, CA 92131	Valid sort
EC _{at-event}	9150 W Church St, Des Plaines, IL 60016	Valid sort
EC _{no-data}	161 Centereach Mall, Centereach, NY 11720 <no data in event table>	Nothing sorted

calculateDistanceInMiles(userLatitude, userLongitude, eventLatitude, eventLongitude):

- Equivalence Classes:

Equivalence Classes	Description
---------------------	-------------

EC _{+, -, 0}	Positive, Negative, and 0 valued coordinates
EC _{edges}	Edge values of latitude and longitude
EC _{decimal}	Coordinates with decimals
EC _{close}	Coordinates that are closely distanced

- Test cases:

Equivalence Class	Test Case	Expected Output (miles)
EC _{+, -, 0}	(0, 0, 0, 0)	0
EC _{+, -, 0}	(0, 0, 75, -10)	5190.3
EC _{+, -, 0}	(-80, 170, 0, 0)	6898.2
EC _{+, -, 0}	(-45, -135, -30, -100)	2160.1
EC _{+, -, 0}	(60, 0, 29, -75)	3999.6
EC _{edges}	(-90, 180, 90, -180)	12429.7
EC _{edges}	(-90, -180, 90, -180)	12429.7
EC _{decimal}	(-30.550399934, 80.3399599, 89.000388552, 18.0038845)	8283.3
EC _{decimal}	(55.320870823570, 14.999399300, -18.2993955, 170.99399999)	9584.1
EC _{close}	(32.9299175559866, -117.09412823562863, 32.939298140655715, -117.03337138806742)	3.59

User Story 7: As a user, I want to see events in order of the date they occur so I can see events occurring soon.

Module(s) being tested:

- Home.jsx > handleSortByDate()

handleSortByDate(events)

- Equivalence Classes:

Equivalence Classes	Description
EC _{same-date-same-time}	Events have same dates and times
EC _{same-date-different-time}	Events have same dates but different times
EC _{same-time-different-date}	Events have same times but different dates
EC _{different-time-different-date}	Events have different dates and times

- Test Cases:

Equivalence Classes	Test Case	Expected Output
EC _{same-date-same-time}	<pre>[{ eventname: '0', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '1', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '2', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '3', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '4', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', },]</pre>	<pre>[{ eventname: '0', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '1', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '2', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '3', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '4', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', },]</pre>
EC _{same-date-different-time}	<pre>[{ eventname: '0', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '1', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '05:30:00', }, { eventname: '2', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '03:30:00', }, { eventname: '3', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '04:30:00', }, { eventname: '4', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '01:30:00', },]</pre>	<pre>[{ eventname: '4', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '01:30:00', }, { eventname: '2', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '03:30:00', }, { eventname: '3', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '04:30:00', }, { eventname: '1', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '05:30:00', }, { eventname: '0', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', },]</pre>
EC _{same-time-different-date}	<pre>[{ eventname: '0',</pre>	<pre>[{ eventname: '2',</pre>

	<pre>eventdate: '2027-04-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '1', eventdate: '2027-03-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '2', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '3', eventdate: '2027-12-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '4', eventdate: '2027-11-01T08:00:00.000Z', eventtime: '06:30:00', } }];</pre>	<pre>eventdate: '2027-02-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '1', eventdate: '2027-03-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '0', eventdate: '2027-04-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '4', eventdate: '2027-11-01T08:00:00.000Z', eventtime: '06:30:00', }, { eventname: '3', eventdate: '2027-12-01T08:00:00.000Z', eventtime: '06:30:00', } }];</pre>
EC _{different-time-different-date}	<pre>[{ eventname: '0', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '05:30:00', }, { eventname: '1', eventdate: '2026-02-01T08:00:00.000Z', eventtime: '04:30:00', }, { eventname: '2', eventdate: '2025-02-01T08:00:00.000Z', eventtime: '03:30:00', }, { eventname: '3', eventdate: '2024-02-01T08:00:00.000Z', eventtime: '02:30:00', }, { eventname: '4', eventdate: '2023-02-01T08:00:00.000Z', eventtime: '01:30:00', }]</pre>	<pre>[{ eventname: '4', eventdate: '2023-02-01T08:00:00.000Z', eventtime: '01:30:00', }, { eventname: '3', eventdate: '2024-02-01T08:00:00.000Z', eventtime: '02:30:00', }, { eventname: '2', eventdate: '2025-02-01T08:00:00.000Z', eventtime: '03:30:00', }, { eventname: '1', eventdate: '2026-02-01T08:00:00.000Z', eventtime: '04:30:00', }, { eventname: '0', eventdate: '2027-02-01T08:00:00.000Z', eventtime: '05:30:00', }]</pre>