Test Cases for Event Discovery and Public Calendar:

- Vitest testing code is in the github

```
✓ src/test/EventList.test.tsx (2 tests) 17ms
✓ src/test/SearchMessages.test.tsx (2 tests) 56ms
✓ src/test/Calendar.test.tsx (2 tests) 15ms

Test Files  3 passed (3)
     Tests  6 passed (6)
  Start at  13:11:59
  Duration  543ms (transform 116ms, setup 44ms, collect 335ms, tests 88ms, environment 875ms, prepare 12ms)
```

Test Case - Search Functionality

1) Short Query Error

Goal: Ensure that very short search queries show an error instead of running a useless search.

Steps:

1. In the search bar, type a single character
2. Leave category, location, organization, and date filters empty
3. Wait a moment for the search logic to run

Expected Result:

- Under the search bar, SearchMessages displays an error such as:
  "Type at least 2 characters to search."
- The list of events does not update based on the 1-character query.

2) Valid Search with Matching Events

Goal: Verify that a valid search query correctly filters events and shows a result count.

Steps:

1. Identify a word that appears in at least one event's title or description (e.g., "workshop" or "concert").
2. Type that word into the search bar (e.g., workshop).
3. Make sure category, location, organization, and date filters are empty.

Expected Result:

- Only events whose title or description contain that word (case-insensitive) are shown in the list.

- SearchMessages displays a status text similar to:
  "3 results found for "workshop"." (number depends on data)
- No short-query error is shown.
- The app remains responsive and does not crash.

Test Case – Browse and Filter Events

1) Filter by Category

Goal: Confirm that the category dropdown correctly filters the event list.

Steps:

1. Ensure the search bar is empty and other filters (location, organization, date) are blank.
2. In the Category dropdown, select Academic.
3. Observe the events displayed in the list.

Expected Result:

- Only events with category = "academic" are displayed.
- No error messages are displayed.
- The layout and scrolling behavior remain normal.

2) No Events Match Filters (Empty State)

Goal: Ensure that when filters exclude all events, the UI shows a clear empty-state message instead of just a blank area.

Steps:

1. Go to "Browse & Filter Events".
2. Set filters to a combination that should match no events
3. Leave search and date empty.
4. Observe the event list area.

Expected Result:

- No event items are displayed in EventList.
- A friendly empty-state message appears, e.g.:
  "No events match your filters. Try changing the date, category, or search terms."
- The page does not crash or show any red error screens.

Test Case – Integrated Calendar View

1) Calendar Loads Events

Goal: Check that the calendar view displays events from the shared events data.

Steps:

1. Go to the Calendar section of the app.
2. Look for a date where you know an event exists in the events array
3. Use the Month / Week / Day toggle if needed to navigate to that date.

Expected Result:

- The calendar renders without any errors.
- At least one event appears on the correct date, showing its title.
- Changing between Month, Week, and Day views does not remove events or break the UI.

2) Click Event Shows Details (Alert)

Goal:
 Verify that clicking a calendar event triggers the eventClick handler and shows detailed information in an alert.

Steps:

1. Locate any visible event in the calendar (e.g., click on the event block with the title text).
2. Click on the event once.

Expected Result:

- A browser alert popup appears.
- The alert text includes at least a Title line
- Other lines such as Description, Location, Organization, etc. are also shown, with "N/A" where data is missing.
- The app does not freeze or crash after closing the alert.