

Team Name: Git'r Done

Team Members: Brandon King, Jack Marty, Miles McCorkendale, Noah Berry, Sanjana Shenoy, Margad Batamgalan

Project Name: Weather Connect

User Acceptance Tests: (to be done in order due to interdependency)

Login:

Test-1: Create a user.

Test-2: Login using created user credentials

Test-3: Attempt logging in with proper username, and incorrect password.

Test-4: Attempt logging in with incorrect username, and proper password.

Test-5: Attempt to login with incorrect username and incorrect password.

Add Event:

Test-1: Add event -> Check that event has been created.

Test-2: Add event that is on the same day as previously created event -> Check that event has been added without error.

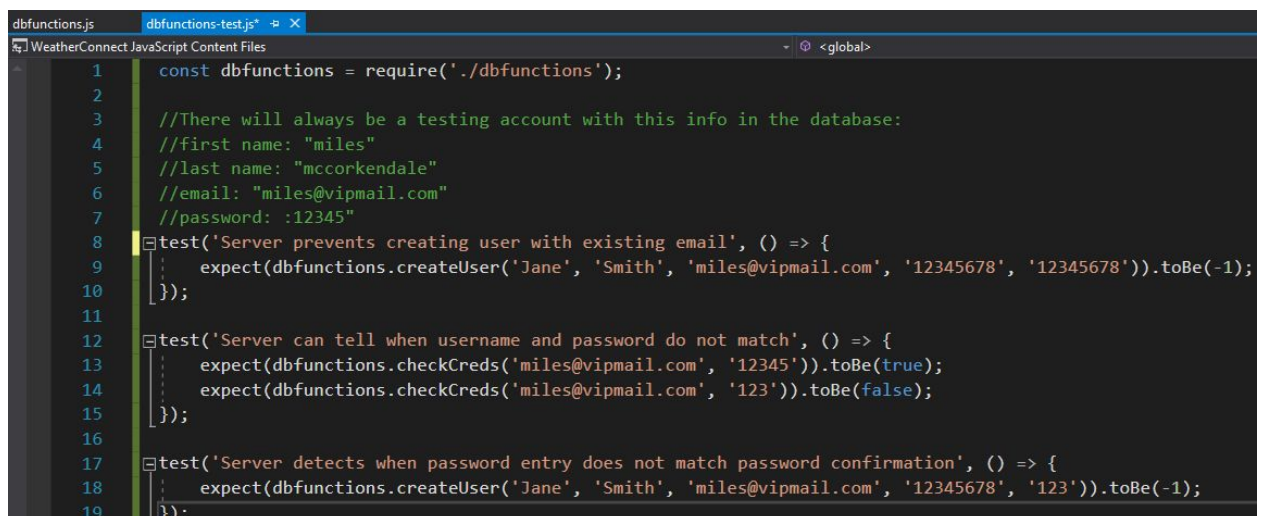
My Events:

Test-1: Check that all previously added events are all shown.

Automated Testing Tools:

[Jest JS](#)

Automated Test Code:



```
dbfunctions.js  dbfunctions-test.js  X
WeatherConnect JavaScript Content Files  <global>

1  const dbfunctions = require('./dbfunctions');
2
3  //There will always be a testing account with this info in the database:
4  //first name: "miles"
5  //last name: "mccorkendale"
6  //email: "miles@vipmail.com"
7  //password: "12345"
8  test('Server prevents creating user with existing email', () => {
9    expect(dbfunctions.createUser('Jane', 'Smith', 'miles@vipmail.com', '12345678', '12345678')).toBe(-1);
10 });
11
12 test('Server can tell when username and password do not match', () => {
13   expect(dbfunctions.checkCreds('miles@vipmail.com', '12345')).toBe(true);
14   expect(dbfunctions.checkCreds('miles@vipmail.com', '123')).toBe(false);
15 });
16
17 test('Server detects when password entry does not match password confirmation', () => {
18   expect(dbfunctions.createUser('Jane', 'Smith', 'miles@vipmail.com', '12345678', '123')).toBe(-1);
19 });
```