Sprint 2 Defect Log

Guna Kondapaneni, Lisa Campbell, Matthew Pace,
Brian Kaetzel, Joseph Khawly

Severity: 1-5, 1 is most severe, 5 is least

Design Inspections

Fetch Calendar List Flow Design Inspection			
Defect	Description	Severity	How Corrected
1	Routes would send REST call back in String Format.	3	Using the ResponseEntity<T> class, and .toPrettyString() function of the GenericJSON class, we were able to fix the error
	Code Before		Code After

	`public String ...{`
	`...//code in here`
	`return events.toPrettyString();`
	`}`
			`public ResponseEntity<String> ...{`
			`...//code in here`
			`return new ResponseEntity<String>(events.toPrettyString(), httpHeaders, HttpStatus.OK);`
			`}`

2	Frontend would post the calendar to import via query parameters, while the backend expected it as JSON in the request body.	1	Specify the contentType of the request and build the JSON for the request.
	Code Before		Code After

	`const data = {`
	`  calID,`
	`  userName: this.state.user.name`
	`}`
	
	`  ajax({`
	`    url: '/calendar/add',`
	`    type: 'post',`
	`    data,`
	`    ...`
	`  })`
			`const data = {`
			`  calID,`
			`  userName: this.state.user.name`
			`}`
			
			`  ajax({`
			`    url: '/calendar/add',`
			`    type: 'post',`
			`    contentType: 'application/json',`
			`    data: JSON.stringify(data),`
			`    ...`
			`  })`

Fetch Calendar List Flow describes the interaction of the Endpoint and ImportPage Modules.

Module Inspections

UserPage Module Inspection

Defect	Description	Severity	How Corrected
1	The calendar shows too many different views. Our application only needs the week view.	4	Specify the defaultView and views props for the calendar component.
	Code Before		Code After
	`<BigCalendar` `  ...` `/>`		`<BigCalendar` `  defaultView='week'` `  views={['week']}` `  ...` `/>`

Endpoint Module Inspection

Defect	Description	Severity	How Corrected
1	/calendar/event only processes one event at a time, when the frontend often has many to process. This results in the frontend making many calls to this endpoint, when it could be accomplished with only one call.	4	TODO: /calendar/event now looks for multiple events in the JSON it gets.
	Code Before		Code After
	`addCalendarEvent() {` `  // Processes a single event` `  return "OK"` `}`		`addCalendarEvent() {` `  for (event : RequestBody) {` `    // Process event` `  }` `  return "OK"` `}`

StressFormPage Module

Defect	Description	Severity	How Corrected
1	Stress form validation was	2	Created custom integer

	allowing non-integer values **Input:** List of events **Expected Output:** List of events with integer valued stress levels or error when non-integer values entered **Actual Output:** No error was displayed when non-integer values were entered		verification instead of using built in function.
	Code Before		Code After

	`…`
	`const num = parseInt(val)`
	`…`
			`…`
			`const num = this.filterInt(val)`
			`…`
			
			`filterInt(value) {`
			
			`if`
			`(/^(\-|\+)?([0-9]+|Infinity)$/.te`
			`st(value))`
			`{ return Number(value); }`
			
			`return NaN;`
			`}`

Unit Tests

DynamoDB Unit Tests			
Defect	Description	Severity	How Corrected
1	DynamoDB not getting Tables **Input:** Nothing **Expected Output:** “OK” **Actual Output:** Error message from AWS about credentials	4	Make sure that the credentials are setup correctly

	`// no code`		`public static void`
			`main(String...args) {`
			`DBSetup.remoteDB();`
			`….`

			`}`

ImportPage Unit Tests			
Defect	Description	Severity	How Corrected
1	User was unable to submit their chosen calendar to import. **Input:** User selected calendar from dropdown. **Expected Output:** Render a button that can send this data to the server. **Actual Output:** Nothing	3	Add the button and make an ajax request for it to fire.
	Code Before		Code After

	`// no code`		
			```
			postCalendarAdd(calID) {
			  const data = {
			    calID,
			    userName:
			this.state.user.name
			   }

			    ajax({...})
			}

			<Button
			  onClick={() =>
			postCalendarAdd(calID)}
			>
			  Submit
			</Button>
			```

UserPage Unit Tests			
Defect	Description	Severity	How Corrected
1	While writing a unit test to see if the calendar pops up, we found that the calendar does not correctly display all day events. **Input:** List of events, some of	3	Write functions for the startAccessor and endAccessor props of the calendar component that detects the presence of fields indicating that an event is all

	which are all day events. **Expected Output:** Timed events render on the calendar, and all day events render above their expected days. **Actual Output:** As above, but the all day events in the input are missing.		day.
	Code Before		Code After
	`<BigCalendar` `...` `startAccessor='start.dateTime'` `endAccessor='end.dateTime'` `...` `/>`		`accessor(time, event) {` `// Returns correct string for` `// normal events, all day` `// events` `}` `<BigCalendar` `...` `startAccessor={event =>` `accessor('start', event)}` `endAccessor={event =>` `accessor('end', event)}` `...` `/>`

Unit Testing Automation Strategy

1. We are using Jest.js and Enzyme on the front end, for React testing. To run all the front-end tests and front-end test suites, simply execute *npm test* via command line. We are using JUnit and TestNG for the backend tests. *mvn package* will run all of the backend tests.