ID2: Tutorials given within the team and other investments in spreading knowledge

QA TEAM

- Cameron's showed a video on how to log in to Rollbar and see the reports. Additionally, he created a tutorial on how to use rollbar:

Rollbar Docs:

A spike branch has been created for the use of rollbar, our multilevel logging system

80-rollbar_spike

To see logs, log into

Loading Rollbar

https://app.rollbar.com/

and click on your profile, hover over "change account" and click beapengine

(after you've been invited to the team)

which can be found in the rollbar-username channel on our discord

To use the logging import rollbar in each page or component

```
import { useRollbar } from "@rollbar/react";
```

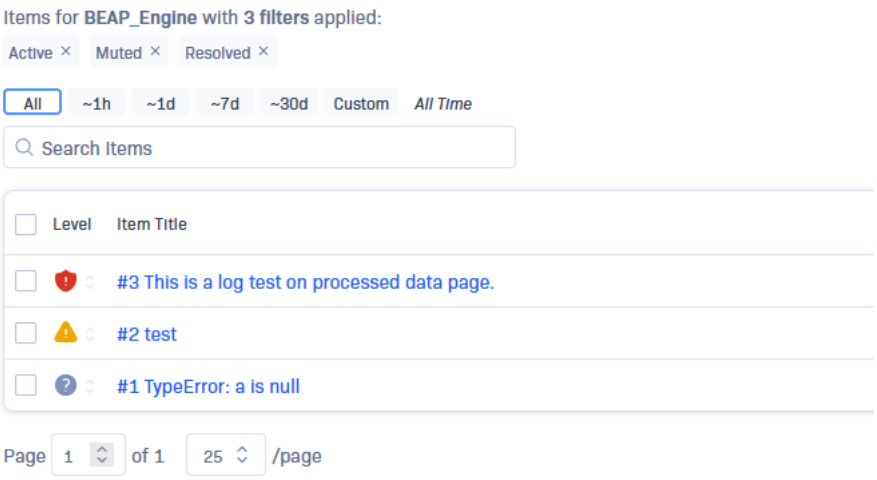
and create an instance of it

```
const rollbar = useRollbar();
rollbar.log("This is a log test on processed data page.");
```

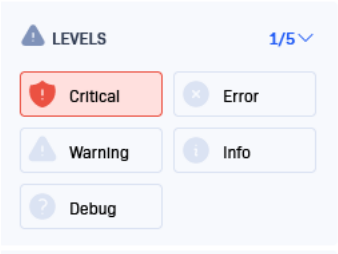
Now you can use many different logging features:

```
// log methods exist for each level
Rollbar.critical("Crash while processing payment");
Rollbar.warning("Facebook API unavailable");
Rollbar.info("User logged in");
Rollbar.debug("Cron job starting");

// can pass arbitrary params
Rollbar.info("User logged in", {loginType: "email+password"});

// rich metadata will be included automatically, but if you want to override:
Rollbar.scope({person: {id: "123"}}).info("User logged in");
```

These logs are stored on our rollbar project:



We can filter the logs



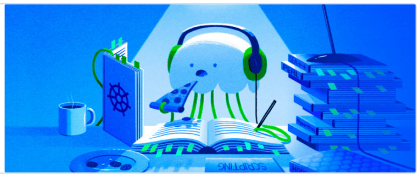
How to create and use Jest Snapshots(Dan):

The following link describes in detail the creation and use of Jest Snapshots

How To Write Snapshot Tests For React Components With Jest | DigitalOcean

In this tutorial, we will be looking at what snapshot tests are and how we can use snapshot testing to ensure our user interface does not change without the ...

https://www.digitalocean.com/community/tutorials/how-to-write-snapshot-tests-for-react-components-with-jest#step-1-creating-a-react-component-to-test



1. Creating a React Component to Test

- Use the following commands in the terminal:

  npx create-react-app@3.4.1 react-snapshot-tests

  cd react-snapshot-tests

  npm start

The react app should be running and can be viewed in your browser.

Next you will need to create a component you can test.

`mkdir src/components`

`nano src/components/Items.tss` or `vim src/components/Items.tsx`

You will need to copy and paste the code from the link above into the created Items.js file you have just created.

Update App.tsx by running the command:

`nano src/App.tsx` or `vim src/App.tsx`

You will need to copy and paste the code from the link above to edit App.tsx.

Revisiting the website created by `npm start`, there should be a screen with the values you established in App.tsx.

2. Writing Snapshot Tests

   Remove the generated file App.test.js.

   `rm src/App.test.js`

   Install react-test-renderer.

   `npm install react-test-renderer@16.13.1`

   Add a test file.

   `nano src/components/Items.test.js` or `vim src/components/Items.test.js`

   Write a test that renders the Items component with no items passed down as props:

   See code from the above link.

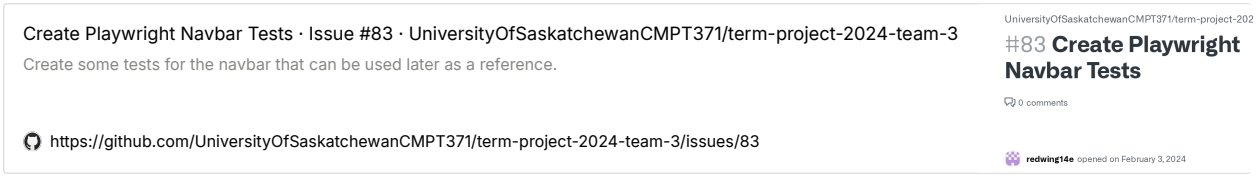
   Use command `npm test` to setup initialization tests.

3. Updating Snapshots test

   Similar to creating blank tests, setup your own values to test. See above link for more detailed information.

Playwright Doc:

A spike/example code has been made:



Make sure that you have playwright installed:

`cd frontend`

`yarn install`

`npx playwright install` or possibly able to do `yarn playwright install`

one way to run tests: `npx playwright test` or `yarn playwright test` or through ui

I highly recommend you install the playwright vscode extension:

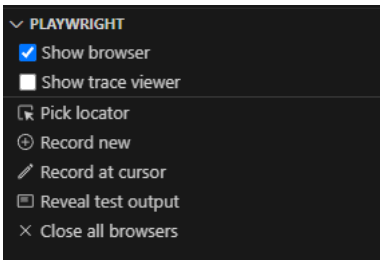
Name: Playwright Test for VSCode
Id: ms-playwright.playwright
Description: Run Playwright Test tests in Visual Studio Code.
Version: 1.0.21
Publisher: Microsoft
VS Marketplace Link:
https://marketplace.visualstudio.com/items?itemName=ms-playwright.playwright

As a main reference, I have used: https://playwright.dev/docs/intro

To start you should first get the system fully running on your system

(currently that is just `yarn start`),

With the vscode extension you should see (at the bottom):



The buttons (in my understanding) are:

Pick locator - generates a single line that you can copy and past where you want

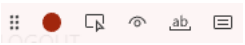
Record new (not recommended) - creates a new file and starts recording your actions

Record at cursor (HIGHLY recommended) - records the users actions and creates code to match at the current place you have selected in the file

Reveal test output - currently have not used

When you start one of the runners you will have to go to the site either http://localhost:3000 or http://127.0.0.1:3000

You should see this at the top



The right 3 allow you to assert different elements and when recording anything you click on is recorded

For any further info either refer to the examples or the docs or api