

ID4-Profiling Front-End Functionalities – Daniel Benesh, Team 3 – Beap Engine

In this profiling exercise we will consider anything over 2s to be an unacceptable time frame for the component to render. I chose this time frame since the likelihood of a user leaving a page occurs at 2.5s. Source:

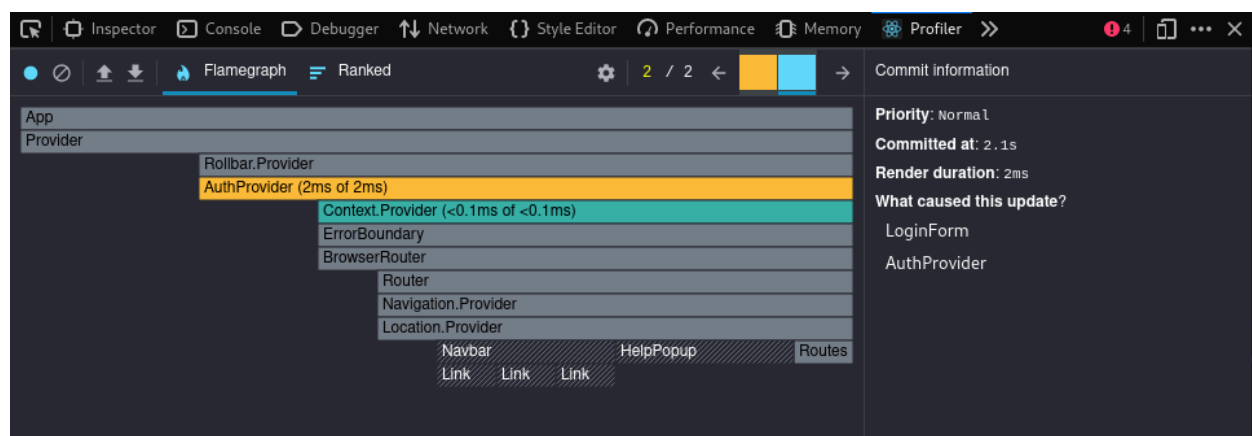
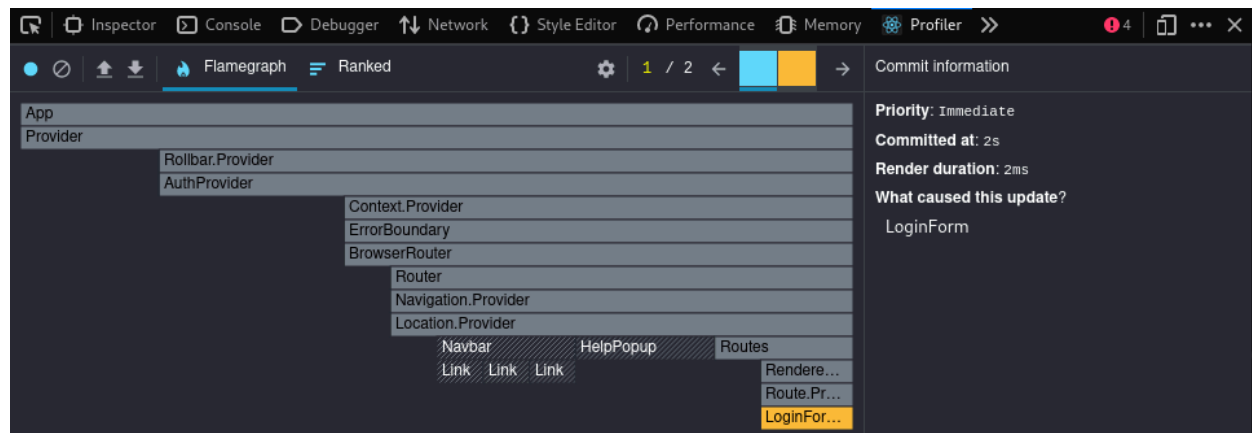
<https://www.browserstack.com/guide/how-fast-should-a-website-load>

1. Sign In Page

Actions taken:

Sign in -> Entered credentials

Results in Flamegraph Format:



Analysis:

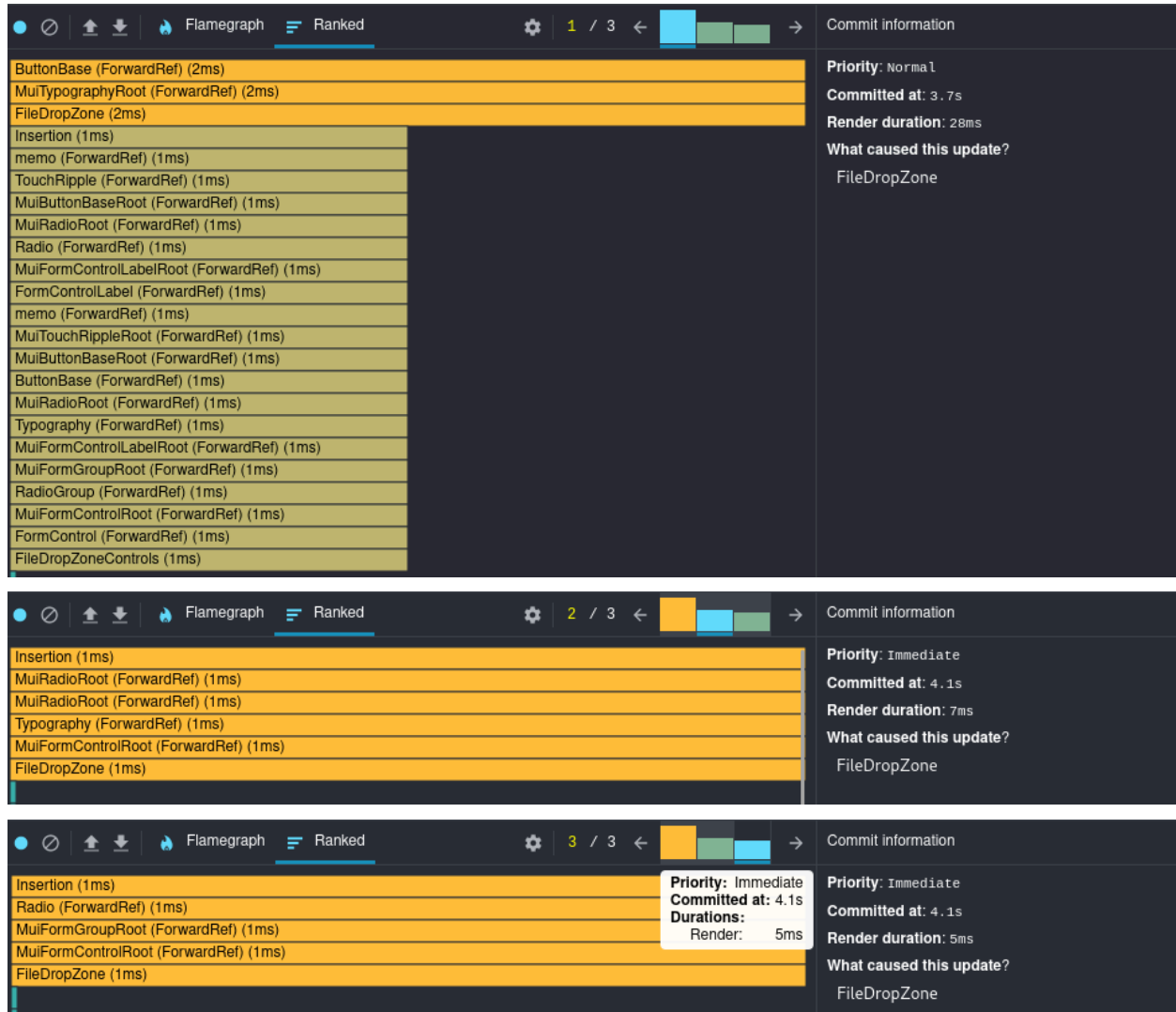
AuthProvider and LoginForm are the calls of interest. Both are rendering within an acceptable time frame of 2ms.

2. File Upload Page

Actions Taken:

i) Drag Single File from folder -> Drop in drop zone

Results in Ranked Format:



Analysis: FileDropZone is the call of interest. It is taking a total of 5ms to render. This is an acceptable time frame.

ii) Drag Folder Containing Multiple Files-> Drop in drop zone
Results in Ranked Format:

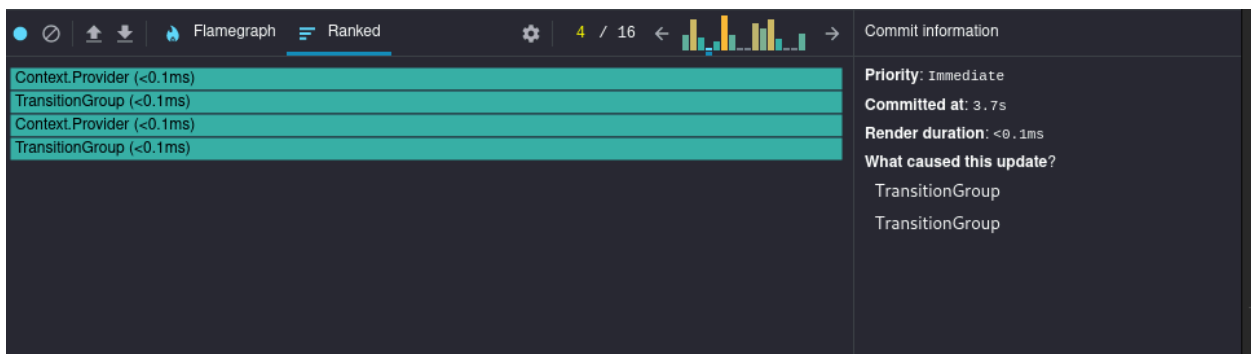
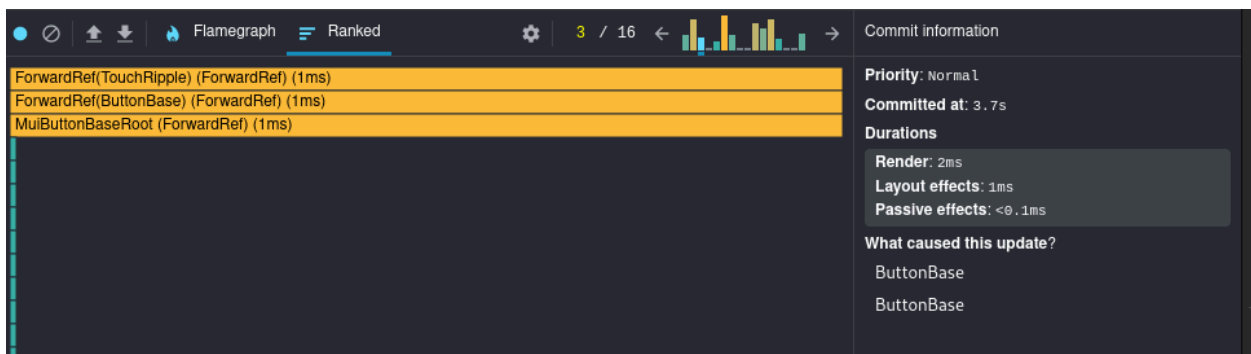
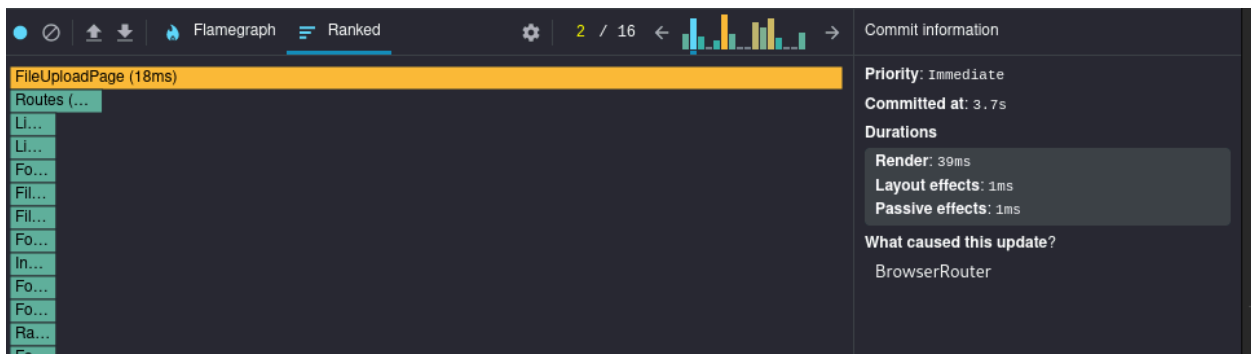
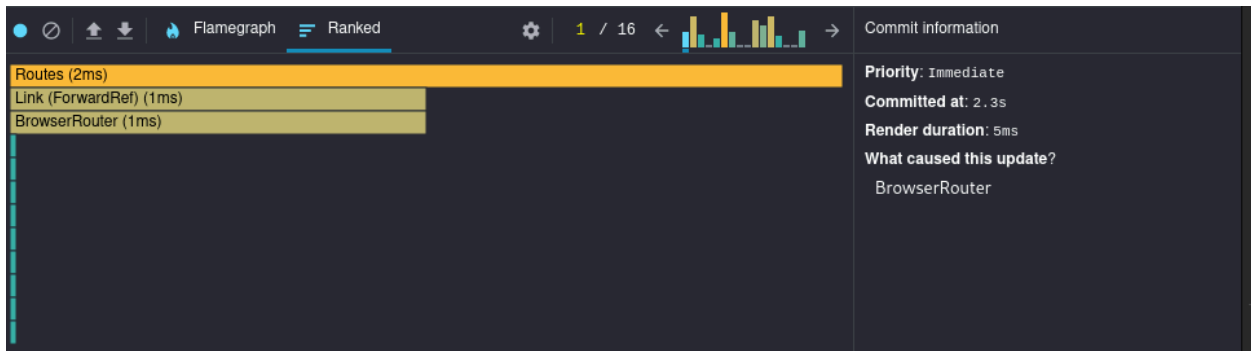


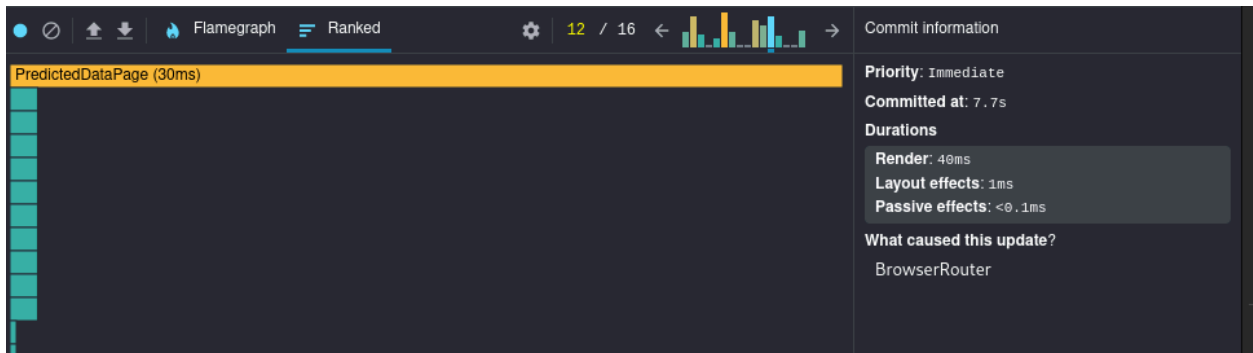
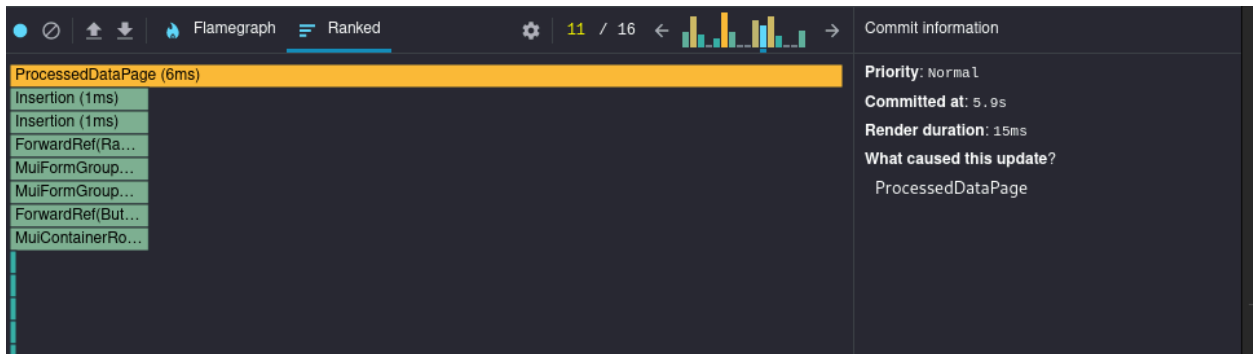
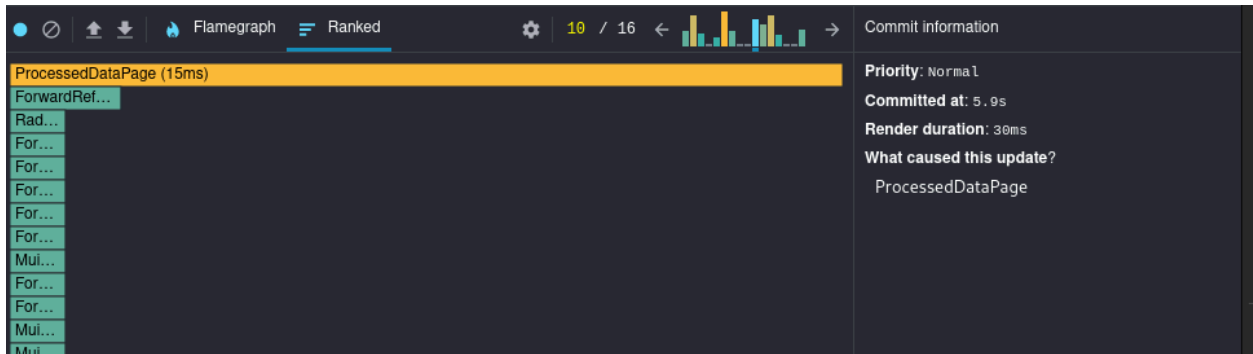
Analysis: FileDropZone renders in an acceptable time frame of 7ms.

3. Going Through all Links in Navbar and Logging Out

Actions taken: Home -> File Upload -> Processed Files -> Predicted Files -> Logout

Results in Ranked Format:







Analysis: The longest time to render in this succession of actions is 50ms caused by opening the link to `PredictedDataPage`, which is within the acceptable time frame. The `FileUploadPage` and `ProcessDataPage` are also quite high in comparison to the other files we have seen at 39ms and 30ms respectively. These pages both have much more elements to render. These pages also are not fully implemented.

Takeaways:

The pages that will require the most attention are those that will Predict, Process and Upload data. Once we have fully implemented these pages, they will require our attention in rendering them in a reasonable timeframe. We will need to make sure that a process is working by implementing a completion bar to let the user know that the system is working on the task the user called. This completion should be used on the uploading, processing and downloading of their activity data.

All the rendering of pages and use of the FileDropZone calls are working within a reasonable timeframe. This is encouraging for our team on the front end. Once we profile calls in the backend we will need to ensure that the user knows clearly the progress of the system while it is working various tasks. The intention is to make the app as user friendly as possible, which our team will count as a success.