

Report on Front-end Implementation Challenge

Challenges Faced:

1. Hover Effect Synchronization:

Making sure that the hover effect worked smoothly on both the text and checkmark, in the menu items posed a challenge. At first the hover effect was uneven as the background color didn't cover both elements uniformly. This meant tweaking the CSS to guarantee that the entire dropdown item behaved as one unit when it was hovered over.

2. Toggle Effect Implementation:

Creating a seamless toggle effect for the skills sections posed another obstacle. The goal was to ensure that the sections transitioned smoothly in and out of view when clicked or unclicked. Problems emerged when the toggle effect did not function as intended attributed to misalignment, between CSS and state management resulting in transitions or sections failing to appear or disappear as planned.

Questions Raised:

1. How to improve the UI transitions responsiveness?

How can we improve the performance of UI effects to ensure they work seamlessly on all devices and screen sizes especially the transitions?

2. How Can We Integrate Testing for Visual and Interactive Elements?

How can we incorporate testing, such as snapshot tests and visual regression testing to ensure that hover and toggle effects function properly?

Suggestions for improvement

1. Enhance Testing Practices:

Implement visual regression testing to automatically identify any changes, in user interface like hover and toggle states to guarantee that modifications do not disrupt functionality.

2. Use the React Profiler

Utilize the React Profiler tool to pinpoint and resolve any bottlenecks in rendering UI components guaranteeing that transitions and animations maintain their fluidity.

3. Continuous Integration for Style Checks:

Implement a continuous integration (CI) pipeline for style linting and testing to automatically enforce style rules.