SUMMER FELLOWSHIP REPORT

ON

ESIM ON CLOUD
Improving UI/UX by making the Simulation screen resizable and Draggable

Submitted by

Swarneshwar S

Integrated MSc Software Systems
PSG College of Technology
Coimbatore

Under the guidance of

Prof. Kannan Moudgalya
Chemical Engineering Department
IIT Bombay

March 2024

Acknowledgements

I, S Swarneshwar, a summer intern at FOSSEE - eSim on Cloud project, express my sincere gratitude to all those who have contributed to this project and entrusted me with responsibilities, helping me shape my ideas into innovative solutions.

I extend my heartfelt thanks to Mr. Nagesh Karmali and Ms. Firuza Aibara for providing me with the opportunity to contribute to this project. My deepest appreciation goes to my mentors for their invaluable guidance and support. Their expert advice and mentorship steered me through challenges and ensured the successful completion of assigned tasks.

I am also grateful to my colleagues for their excellent teamwork, effective communication, and timely execution of tasks, which significantly contributed to our collective achievements during the fellowship period

Contents

Introduction:
I am writing to provide a detailed report on the implementation of a draggable and resizable simulation screen for the eSim-on-cloud project, an online circuit simulation platform. The purpose of this report is to summarize the work completed, outline the key features and functionalities implemented, and highlight the benefits of the draggable and resizable simulation screen.

Overview:
The eSim-on-cloud project aims to provide users with a seamless online circuit simulation experience. The addition of a draggable and resizable simulation screen enhances the user interface by allowing users to interact with the simulation screen more intuitively and efficiently.

Implementation:
In the project, there was a need to improve the user experience by allowing users to view both the circuit and the simulation output simultaneously. Initially, the simulation screen occupied the entire display, making it impossible to have both elements in view at the same time. To address this, the decision was made to convert the simulation screen into a smaller dialog box.

To make the dialog box draggable, changes were made to the code to implement this functionality. However, it was discovered that the performance of the draggability feature did not meet the desired standards. Users experienced lag and delays when dragging the dialog box, which negatively impacted the overall user experience.

To overcome this performance issue, an alternative approach was considered. The decision was made to leverage the draggable dialog box component provided by Material UI, a popular UI library. By utilizing this pre-built component, it was expected that the draggability feature would perform more smoothly and responsively, aligning with the user's dragging speed.

With the Material UI draggable dialog box component implemented, users could now easily position the simulation output dialog box wherever they desired on the screen, allowing them to view the circuit and simulation output side by side according to their requirements.

Additionally, a new requirement emerged to incorporate the ability to resize the dialog box. Unfortunately, there were no existing components readily available to fulfill this requirement. Therefore, it was necessary to write custom code to introduce the resizability functionality to the existing draggable dialog box.

However, when the resizing functionality was added, it caused conflicts with the draggable feature. The two features interfered with each other, resulting in unexpected behavior and a degraded user experience. To address this issue, further modifications were made to the code.

These modifications involved introducing new divisions and components, segregating the draggable and resizable functionalities to ensure they worked harmoniously. By isolating and organizing the code more effectively, the draggable and resizable features were able to coexist seamlessly, allowing users to not only position the dialog box but also resize it as needed.

During the development process, a new issue arose where the resizable icon would occasionally disappear while resizing the dialog box. This inconsistency in the visibility of the icon was a usability concern. To rectify this, additional adjustments were made to the code to ensure that the resizable icon remained consistently visible during the resizing operation.

After addressing the issue with the resizable icon, the implementation of all the required functionalities was complete. However, a new challenge emerged when testing the code in different browsers. The code worked flawlessly on Firefox, but compatibility issues were encountered when using the Chrome browser.

To ensure cross-browser compatibility and provide a consistent user experience across all browsers, further modifications were made to the code. Special attention was given to ensuring that the components used were fully supported by all major browsers, including Chrome, Firefox, and others. By making the necessary changes and testing the code thoroughly in different browser environments, the application achieved compatibility across various platforms.

In summary, the project involved transforming the simulation screen into a draggable dialog box and incorporating resizability functionality. Through a

combination of custom code and utilizing the draggable dialog box component from Material UI, the user experience was significantly enhanced. Issues related to performance, conflicts between draggable and resizable features, visibility of the resizable icon, and browser compatibility were all addressed and resolved to create a seamless and user-friendly interface.

Challenges:

a. The only available feature for the Material UI dialog box was draggable components. Adding the resizable feature to the MUI draggable dialog box posed a major difficulty. The code written to incorporate the resizable feature affected the functionality of the draggable feature. As a result, the code had to be rewritten from scratch to include both draggable and resizable features for the simulation screen.

b. Initially, the code only worked well in the Firefox browser. Therefore, modifications were made to ensure compatibility with the Chrome browser as well.

Future Considerations:

While the implementation of the draggable and resizable simulation screen marks a significant milestone, there are a few areas that could be considered for further improvement

a. User Feedback: Collecting user feedback on the new features will provide valuable insights for refining and enhancing the draggable and resizable functionality.

b. Performance Optimization: Continuously optimizing the performance of the draggable and resizable simulation screen will ensure smooth and seamless user interactions, particularly when handling complex circuits.

c. Cross-Browser Compatibility: Regular testing and updates should be conducted to ensure compatibility across different web browsers and devices as they evolve.

Conclusion:

The implementation of a draggable and resizable simulation screen for the eSim-on-cloud project enhances the user experience, improves accessibility, and increases productivity. The new functionality has been integrated seamlessly into the existing user interface, providing users with a flexible and interactive circuit simulation environment.