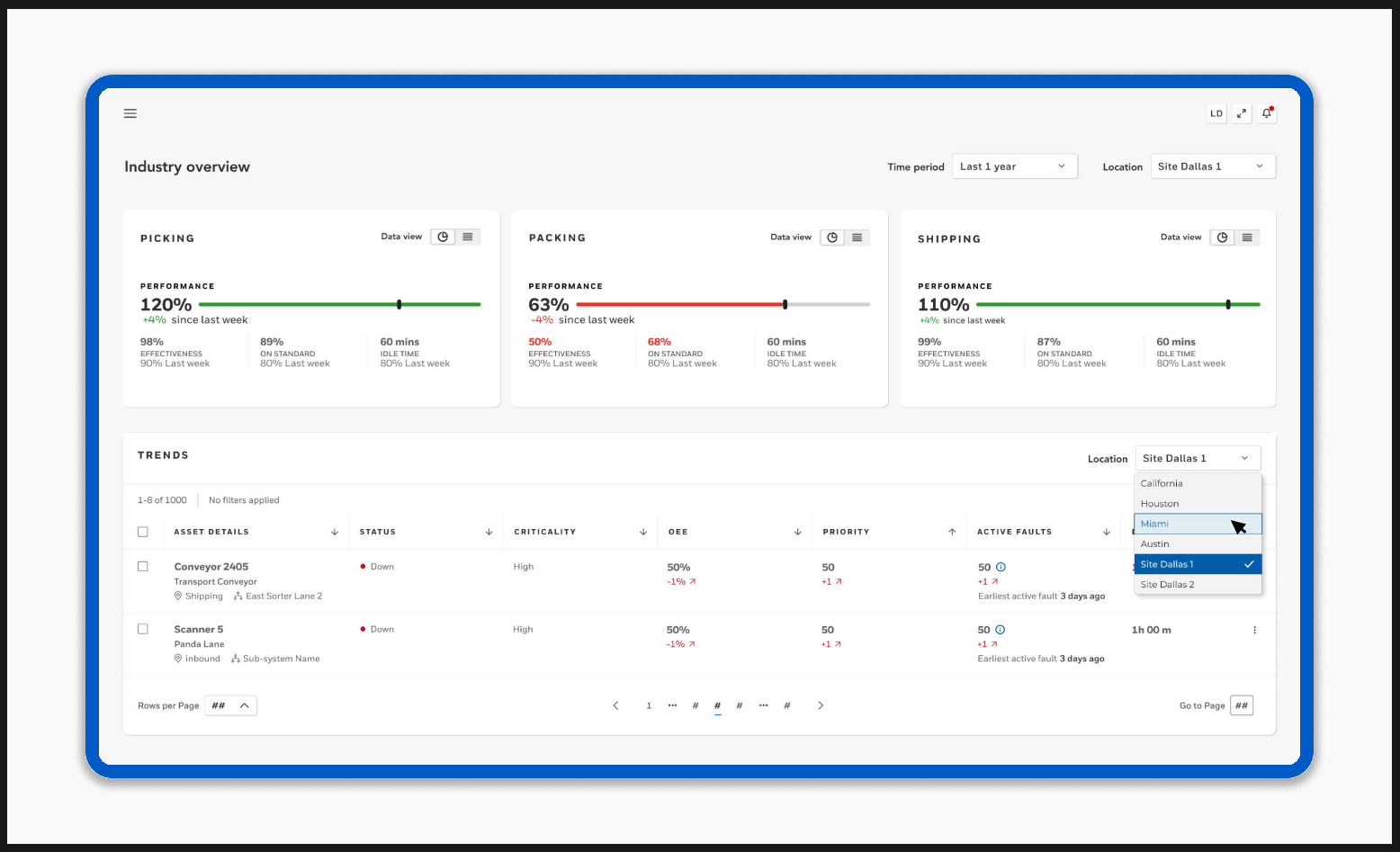
Enhancing Dropdown component

Design Systems Case study

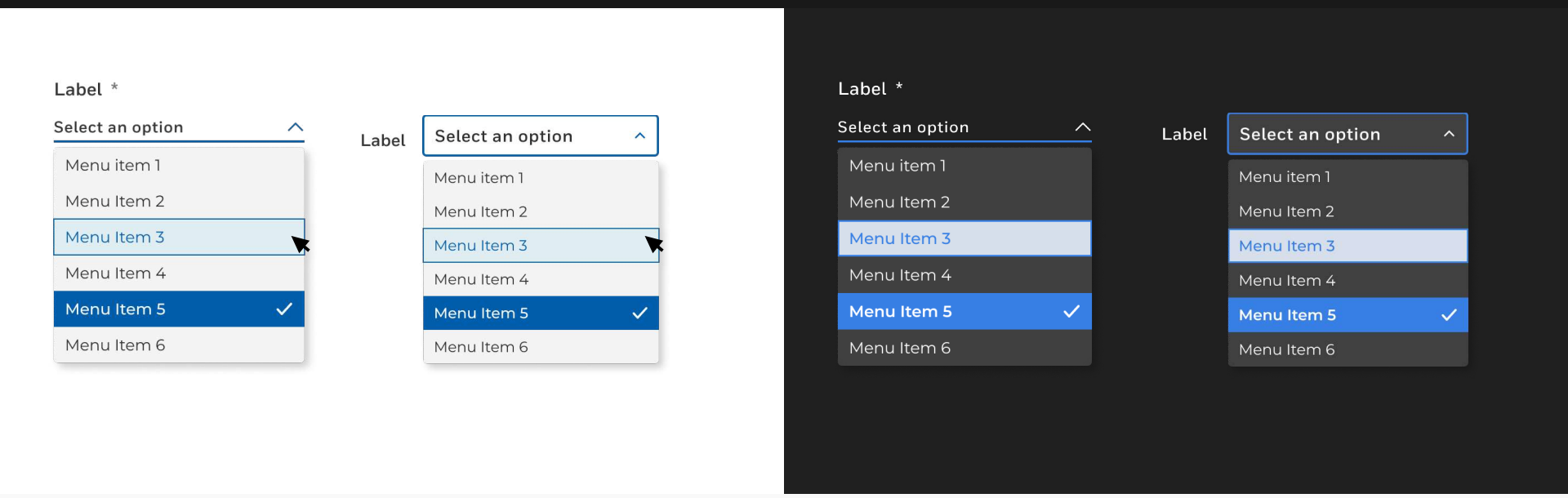


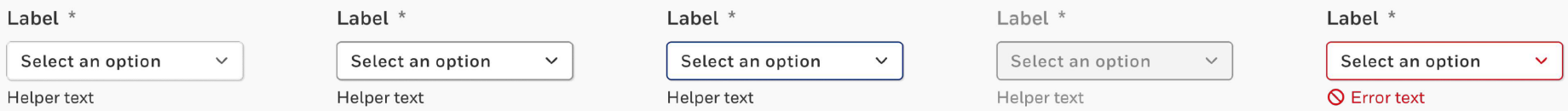
1. Overview

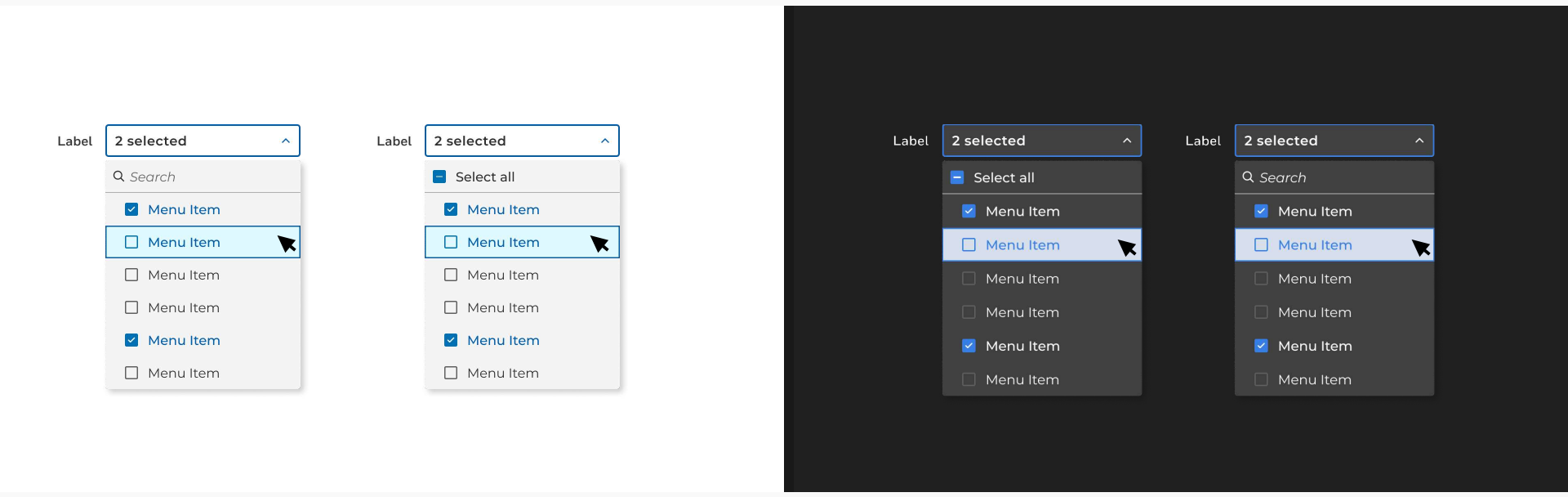
The dropdown component, also known as select, is a form element that upon interaction opens a list of pre-determined value options from which the user can choose.

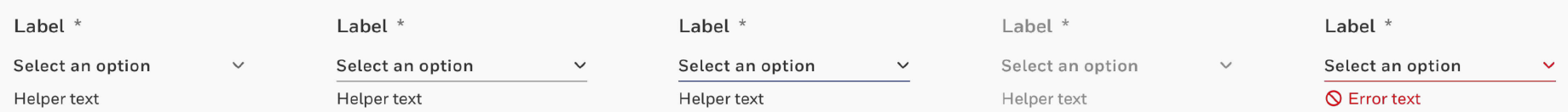
It is a versatile and valuable UI element that addresses the unique needs of SaaS products and dashboards. Its ability to efficiently manage space, organize information, reduce cognitive load, and support customization makes it an integral part of creating intuitive and effective user interfaces.

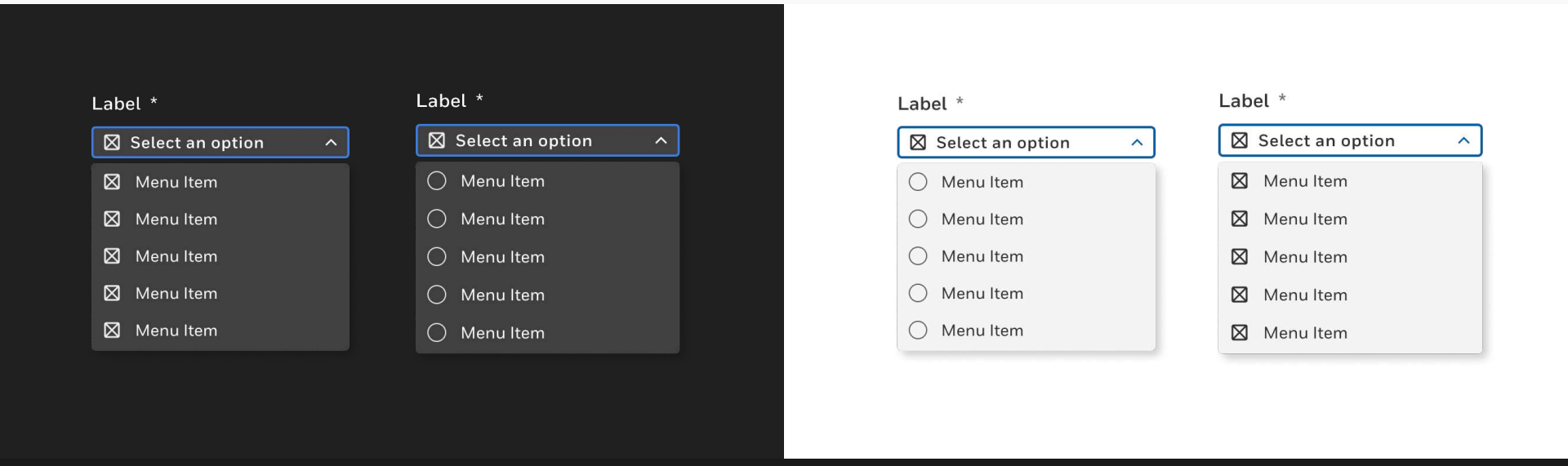
A dropdown component in a design system library ensures consistency and provides a standardized, modular solution for presenting and organizing options across diverse interfaces, enhancing usability and design coherence.











2. Challenges

As a Design System UX designer, dealing with scalability and system-level thinking of components poses unique challenges.

Navigating these challenges requires a combination of strategic planning, effective communication, and a continuous feedback loop with involved stakeholders.

Opportunity

In order to create any new component or element which has to be added to the Design system library needs to analyzed on a holistic level. And thus this brings the need to adhere to following aspects while adding component and guidelines:

1. Address development edge-cases.
2. Ensure usability of components.
3. Cater to use-cases of different products.
4. Maintain scalability across platforms.
5. Design consistent UI styles & patterns.
6. Define usage guidelines using documentation.

4. Navigating through complexities

Throughout the process, efficient and constant cross-functional collaboration with all the involved stakeholders was the key to developing robust designs and solutions for the dropdown component.

Involved stakeholders:

3 Product managers	**3** UX managers	**2** UX Researchers
4 Developers	**6** DS team designer	**7** UX Designers

Keeping stakeholders in loop

1. Understood product requirements and use-cases with product owners.
2. Collaborated with product designers to align and ensure consistent designs.
3. Validating and approved component variants with Design system team.
4. Iterated designs with developers to satisfy development constraints.
5. Align with UX managers and Product managers to meet business requirements.
6. Considered Accessibility modifications for users with disabilities.

---

5. Solution / Design

Case 1:
Grouped menu variant

Defining use-case:

In our SaaS products, users were facing challenges when interacting with large sets of data or options within dropdowns. The current design lacks a structured approach, making it difficult for users to quickly locate and select specific items. As the dataset continues to grow, the usability of the dropdowns diminishes, leading to a suboptimal user experience.

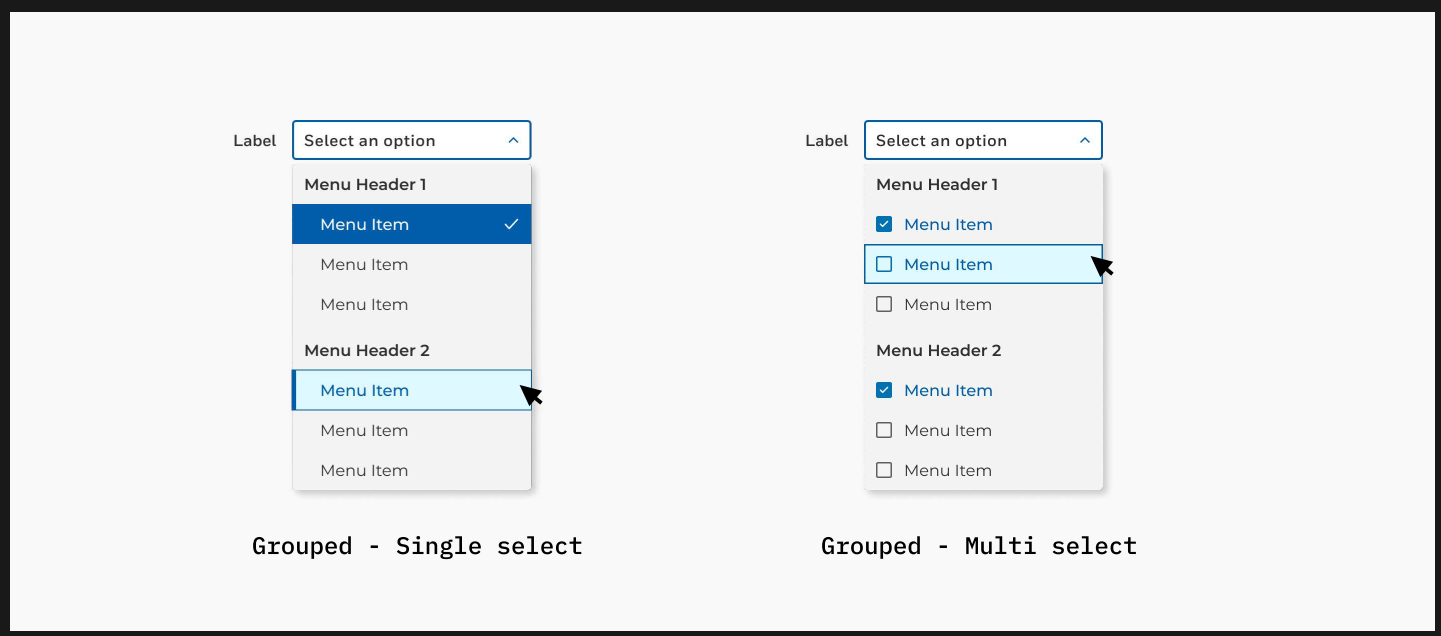
User pain points:

Reduced visual hierarchy due to lack of categorizations and organization

Cluttered list leads to user frustration while finding options

Complexity due to inefficient long scrolling list.

How did we solve?



The requirement was proposed by one of the SaaS based product team catering to multiple site locations. The application allows users to select and multi-select assets from a huge list of options belonging to grouped categories.

Case 2:

Action menu variant

Defining use-case:

One another use-case that was needed by a product team was to give the users the flexibility to take further action when needed through the dropdown list. In SaaS products, sometimes there are so many options to select from which complicates user experience.

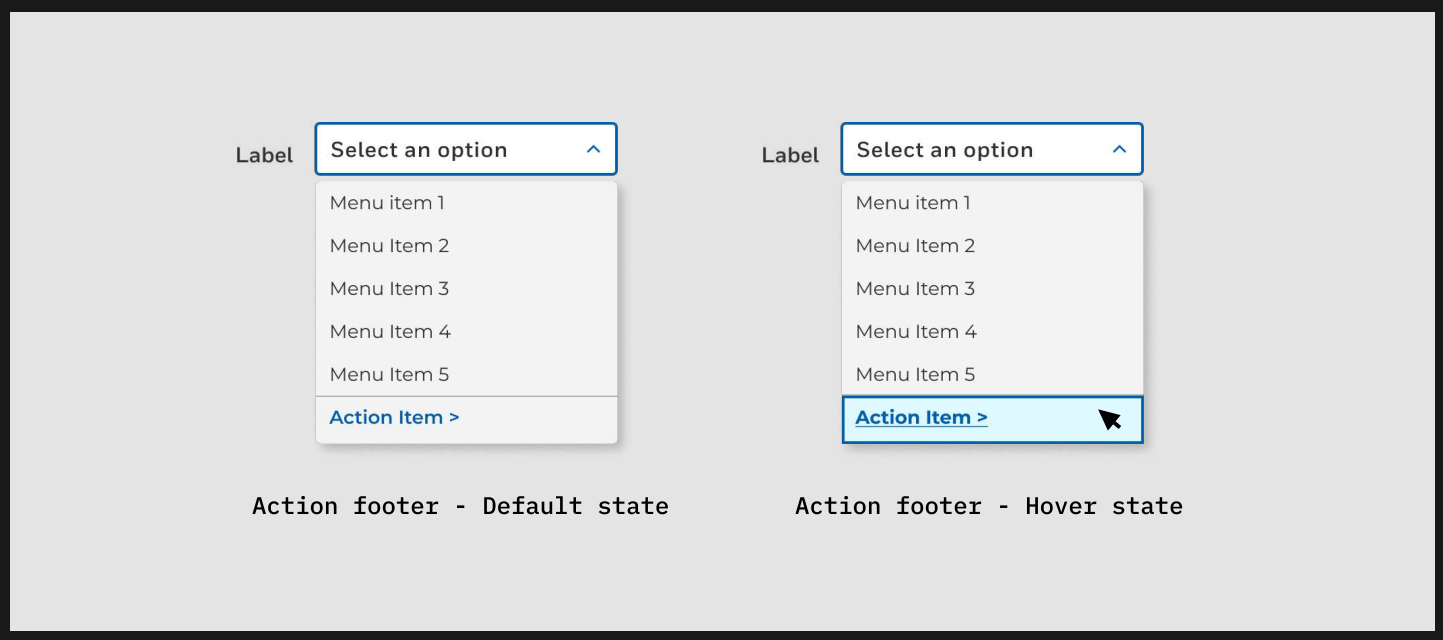
User pain points:

Scrolling through a long list of items to complete task

Need of simplified navigation while finding options

Reduce clutter within already dense Saas dashboards.

How did we solve?



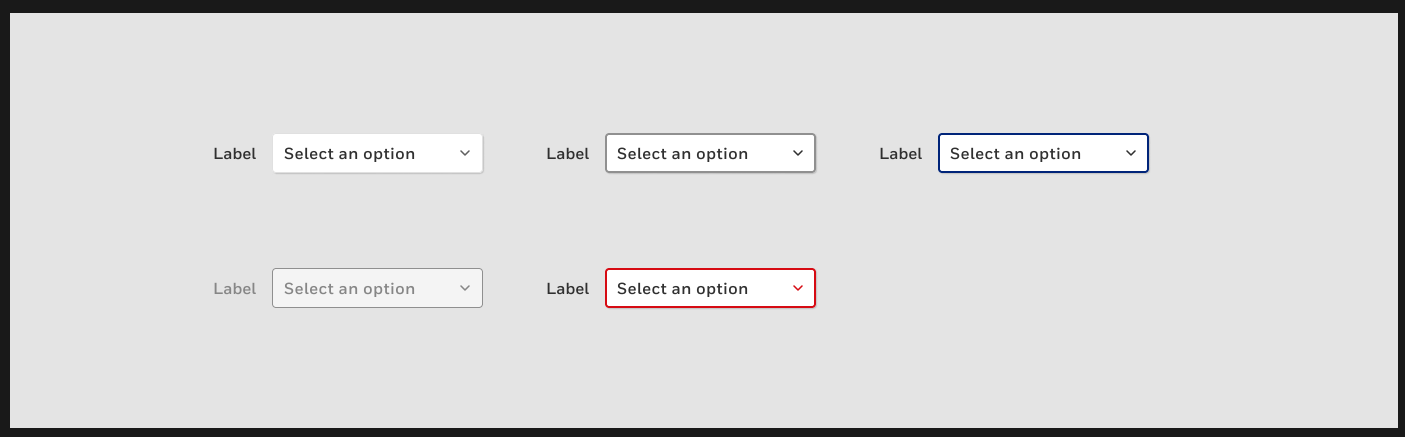
Action footer - Default state

Action footer - Hover state

Case 3:

Meeting accessibility standards

3.1 : Creating “Side label/ Horizontal label” variants



Complying to WCAG guidelines

According to WCAG, labels play a huge role in making sure the content is accessible for users. In dashboards due to less space, top labels for dropdowns does not always align with the design.
WCAG 2.2 - 2.4.6 and 3.3.2

Scalable for Design system library

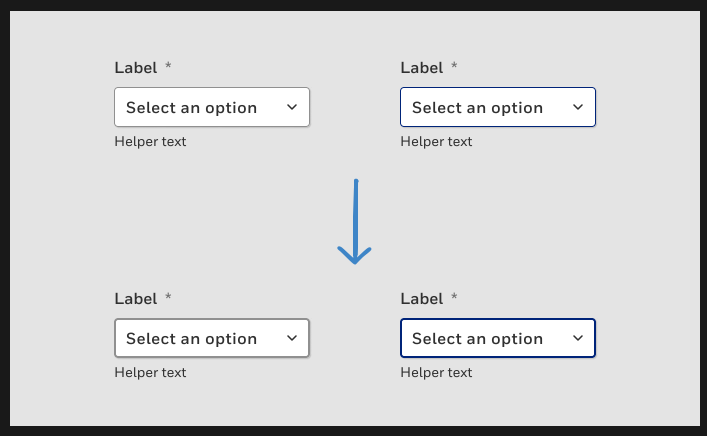
The horizontal or side label component created for the dropdown was developed keeping in mind the scalability with other components.
The variant was applied for Data table page indicator and for text entry forms for the the DS library.

3.2: Enhancing different states of dropdown

Updating components to new WCAG 2.2 guidelines

The new WCAG 2.2 guideline states that in order for the focus state to be visible for users, the border weight should be 2px. Keeping that in mind, I modified not only the hover and focus border weight, but also implemented within the design elements in default state.

WCAG 2.2 - 2.4.13

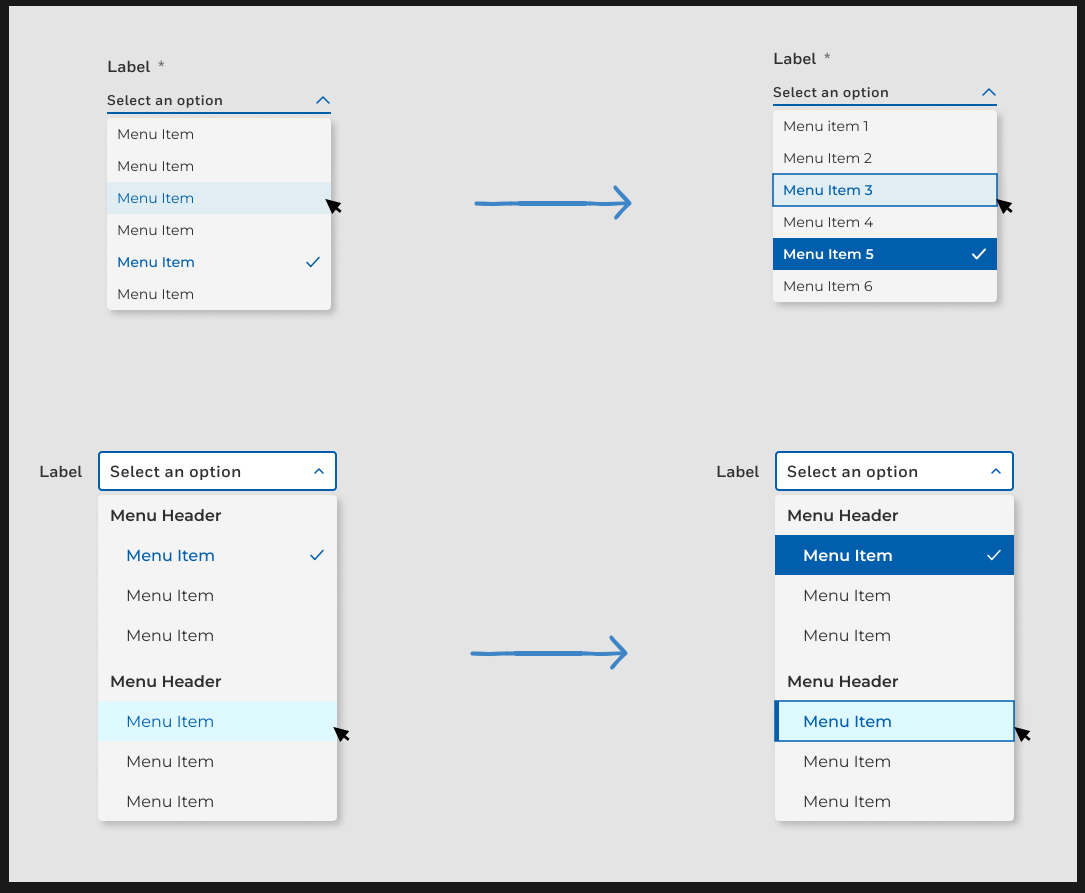


Increased the border wieght of the trigger to 2 px.

Improving interactions for better visual accessibility

Apart from updating the hover and focus states according to WCAG criteria 2.4.13, I worked on updating the visual contrast of different states and interactions of the dropdown component. More clear indications of change in states f elements helps the user to distinguish easily.

Hover state background blue did not meet ideal contrast ratio



Highlighted hover state with 2 px of high contrast border.

Selected state was just a change of text color. Limited visibility.

Filled section with High contrast blue to indicate selected state.

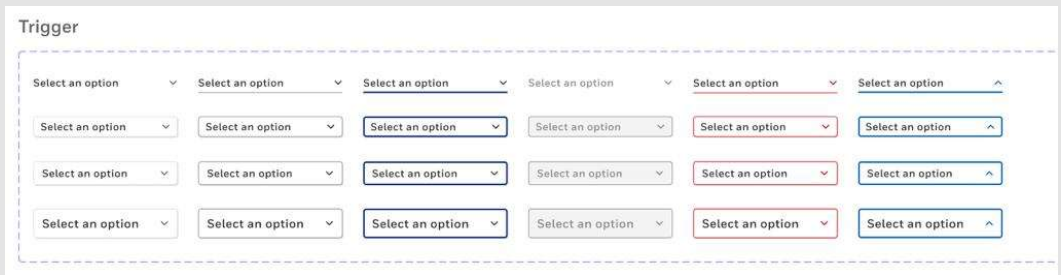
---

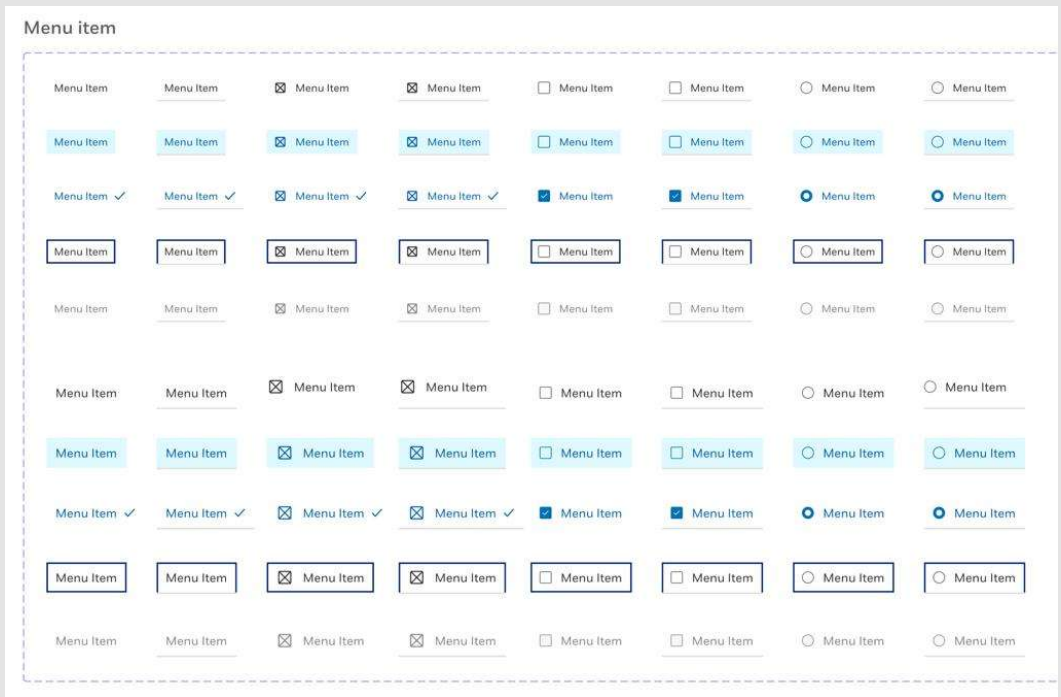
6. Outcomes

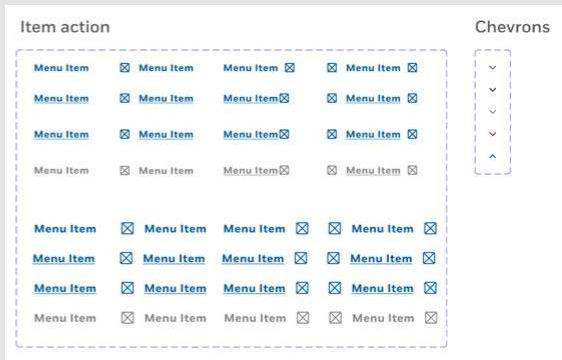
Through this project on Dropdown, the component got updated in terms of accessibility.

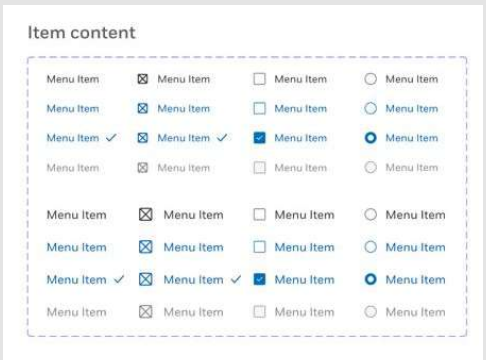
New variants were crated catering to new use-cases from different product teams.

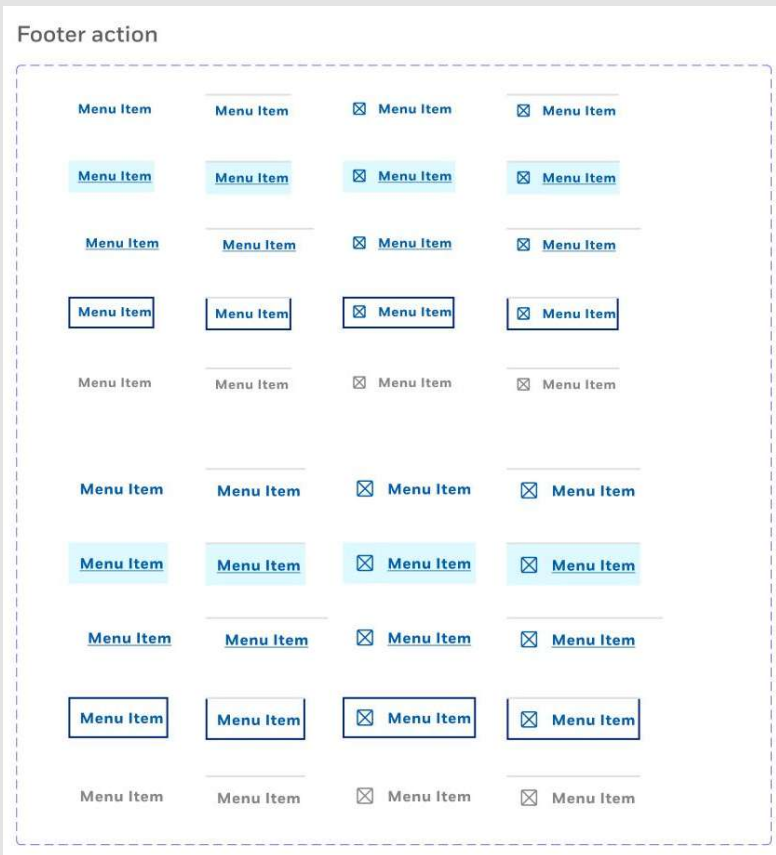
Refined and cleaned repetitive variants, solved responsive issues and gave a new look and interactions for the component.

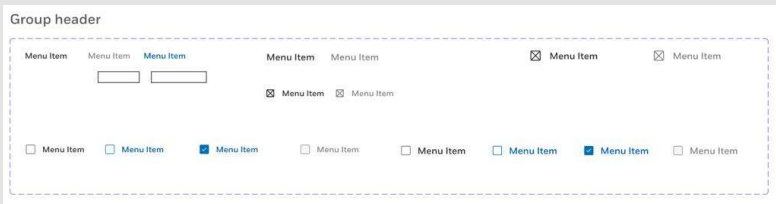


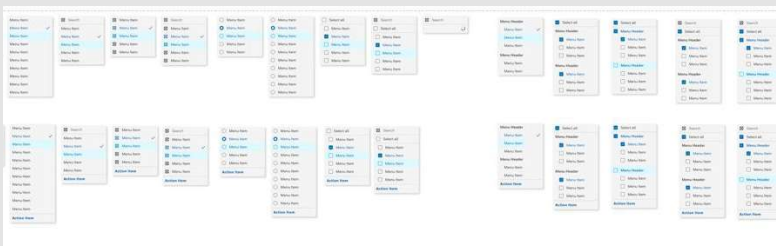












Established a set of Nested components and main components. These components were added to the Design system library for the designers to use catering to varied use-cases.

6. Learnings

- Building scalable components across products
- Catering to different use-cases
- Using decision-tree to differentiate components
- Documenting guidelines for designers to ensure consistency