**Operations Manual for Features Being Worked On**

**Introduction:**

The following document is intended to give the reader an overview of the existing functionality of features that we are working on extending. It will mirror the O&M for the app in terms of demonstrating the capabilities of the features. At the same time, since these features are not incorporated into the live version of the web application, it will also include the necessary steps required to complete these features and incorporate them. It is intended for future team members more than for Zorzal, and as a result is slightly more technical and assumes a basic familiarity with the codebase.

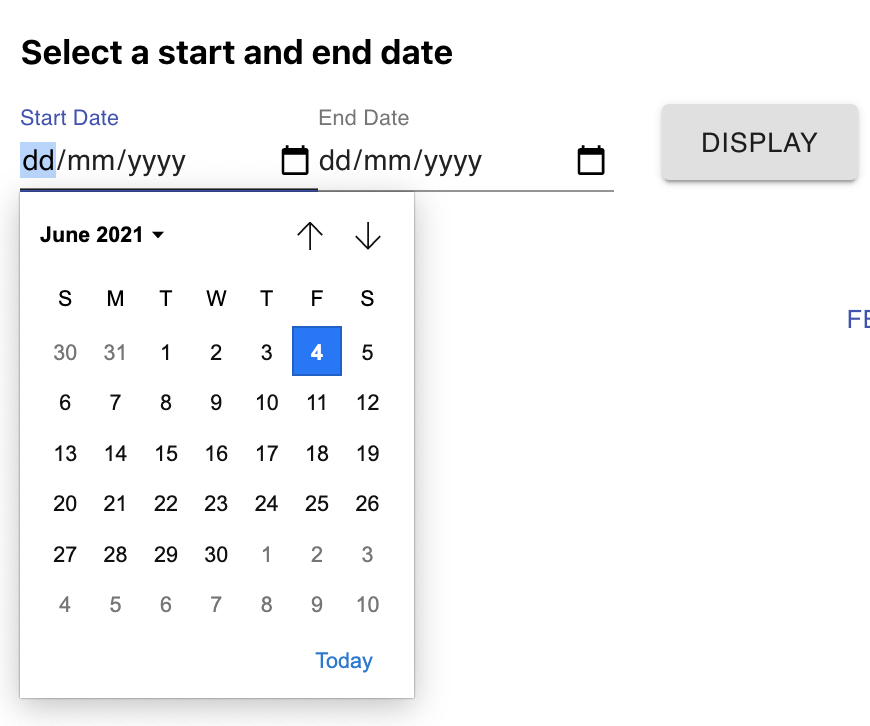
**‘View’**

The ‘View’ component will allow Zorzal employees to analyze the number of shipments scanned through every action point and checkpoint in a given time period. Our job is to collect all the information from the database; filter out all the records that did not occur between the inputted time frame; organize all the action records by type (ex: ‘Create Box’, ‘Box 1’, ‘Box 2’, and so on); and count the frequencies of each category. The frequency tells us how many units (box or lot depending on the action) were scanned at an action point, and these will be the y-values of the bar charts.

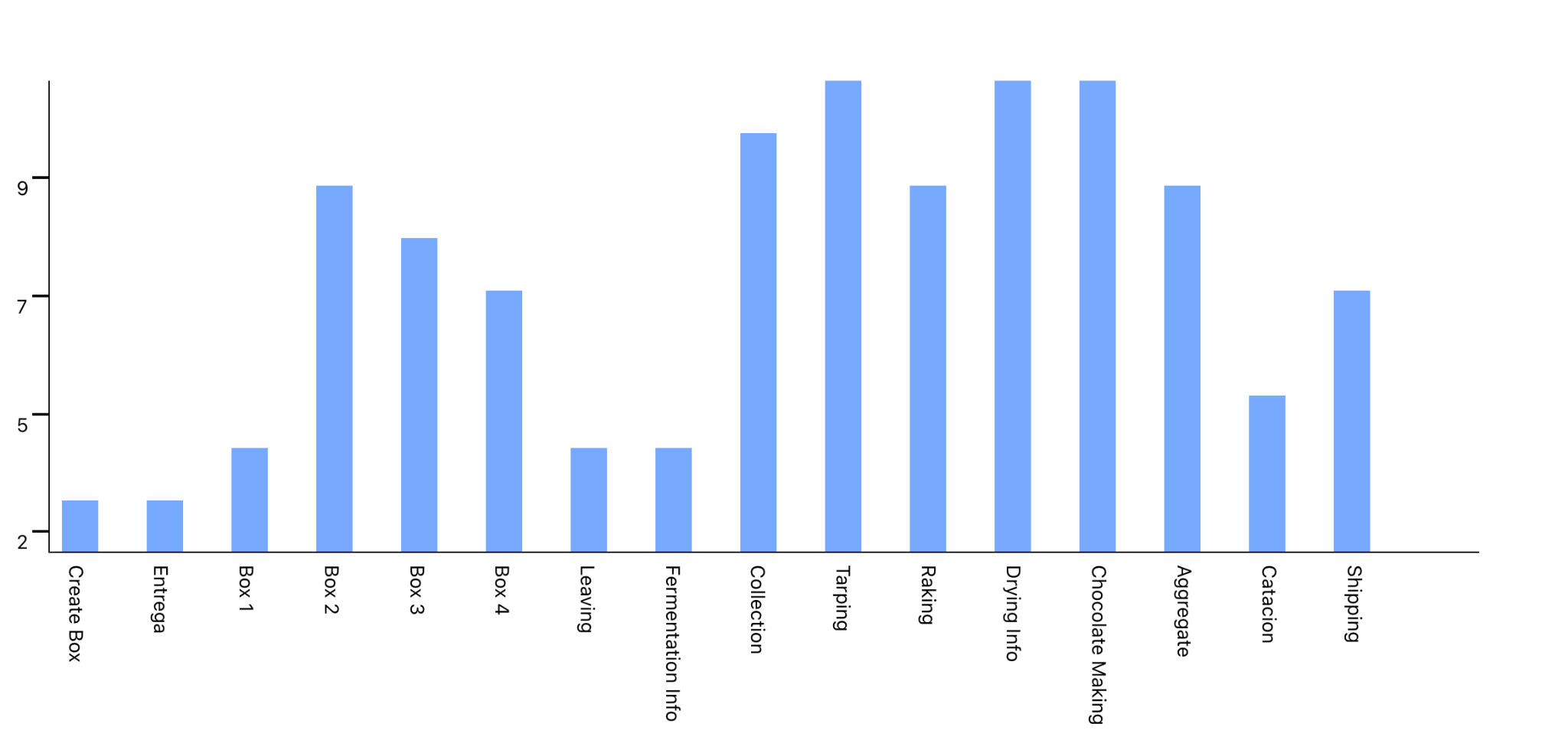
**Step 1**: Select a start and end date

The user will need to choose a start date by either typing the numbers in the date field or by clicking on the calendar icon and selecting the date.

The same goes for the end date, except the user will now pick a later date to serve as the end date.



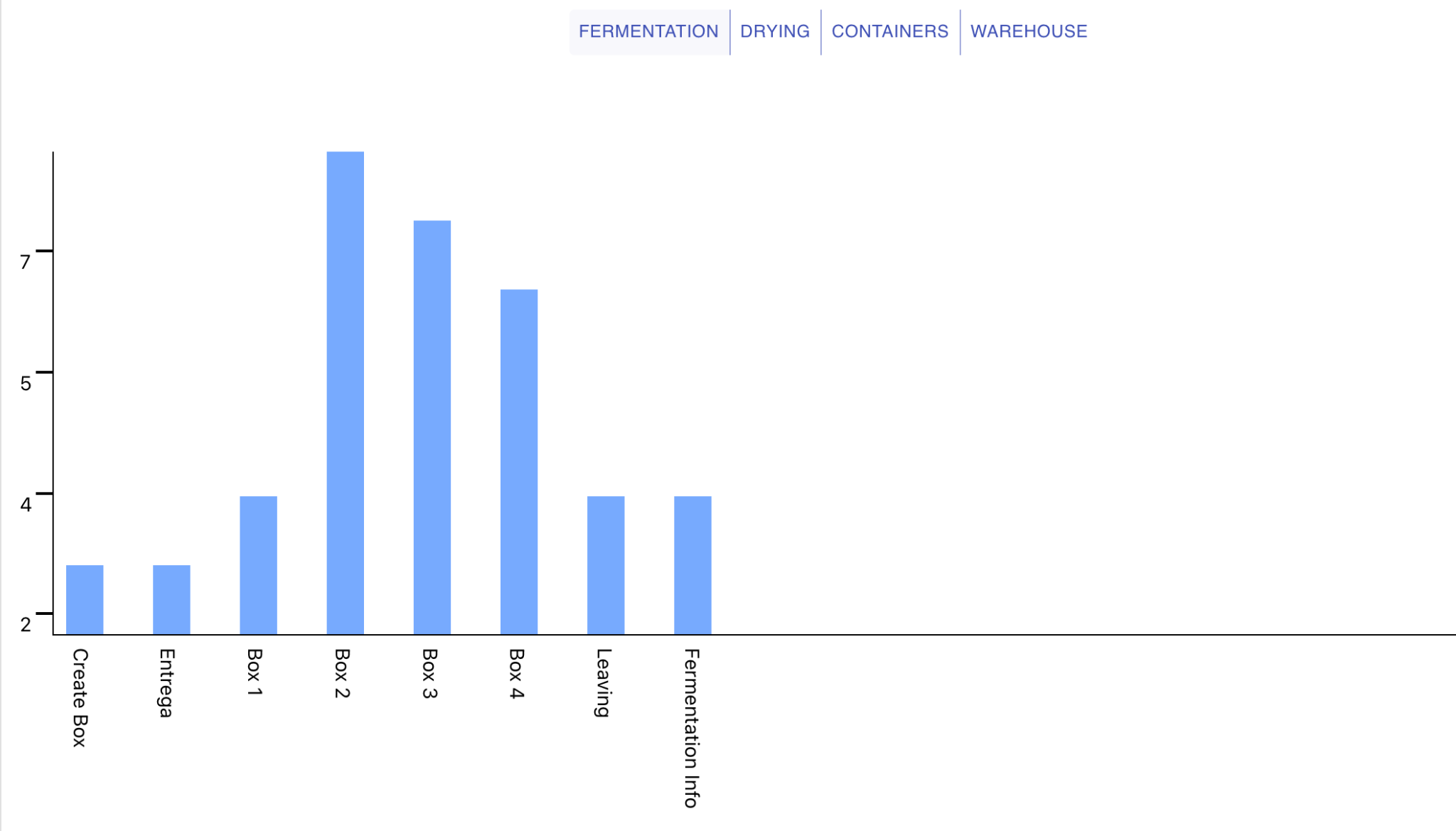
Now, the user should be able to click ‘Display’ and a bar chart showing all the action points throughout the supply chain will appear.



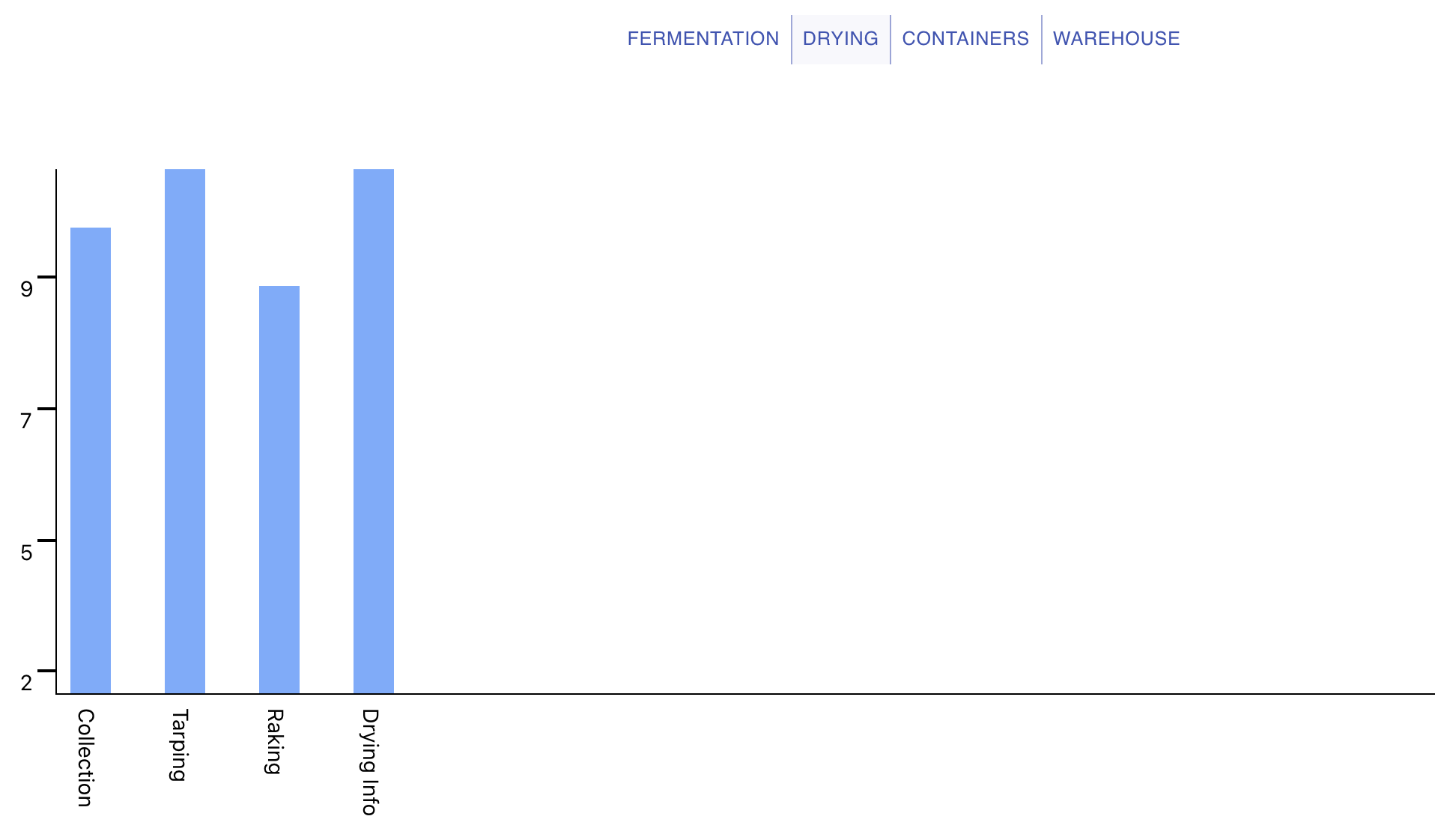
**Step 2**: Click on the graph you want to view

Apart from a bar chart that displays all the action points, we made bar charts that display the action points according to what checkpoint it belongs to. If the user clicks on the ‘Fermentation’ button, a new bar chart will appear, which only shows the action points belonging to Fermentation.

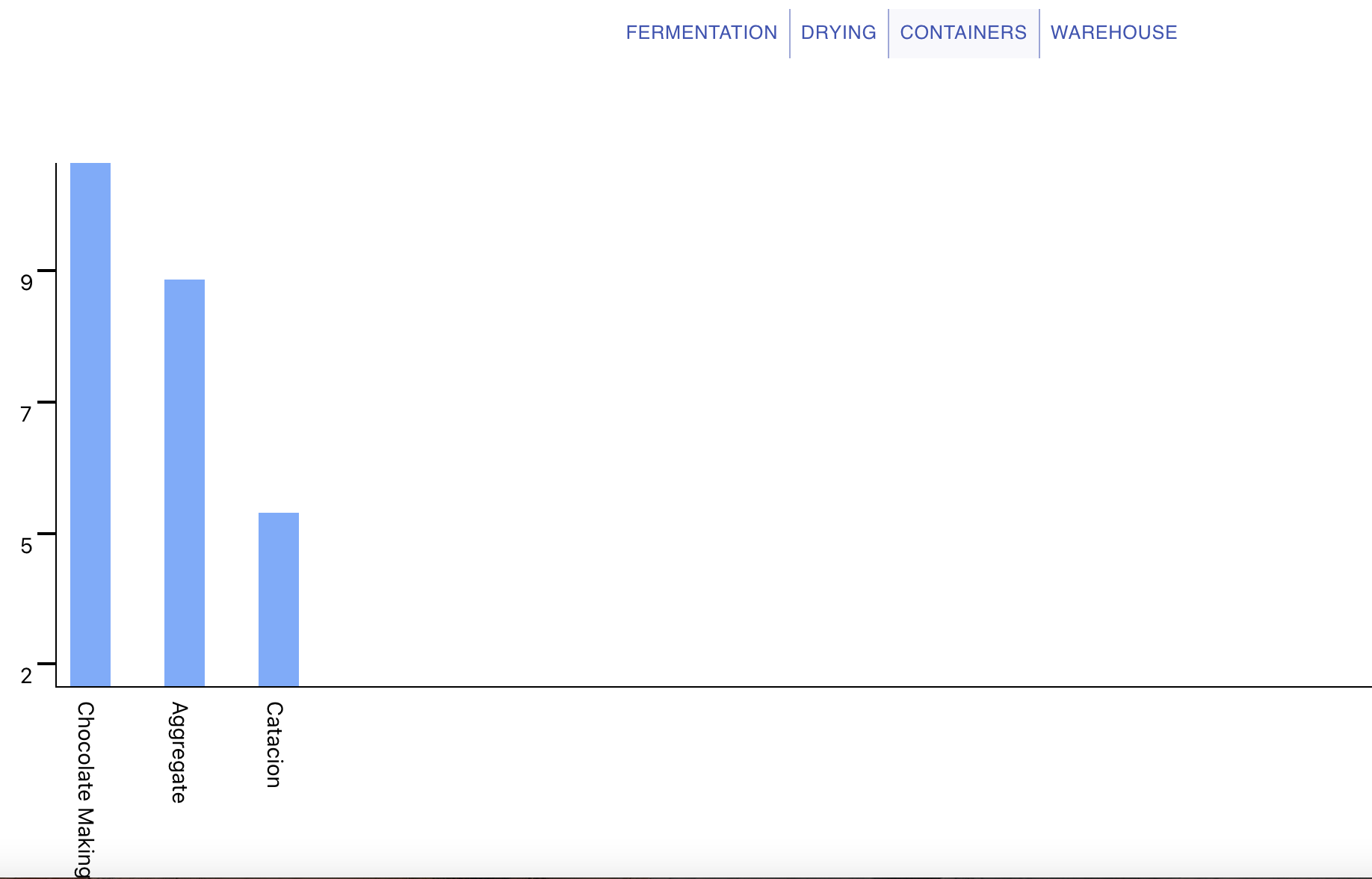
Fermentation:



Drying:



Containers:



Warehouse:



**What’s left to be done:**

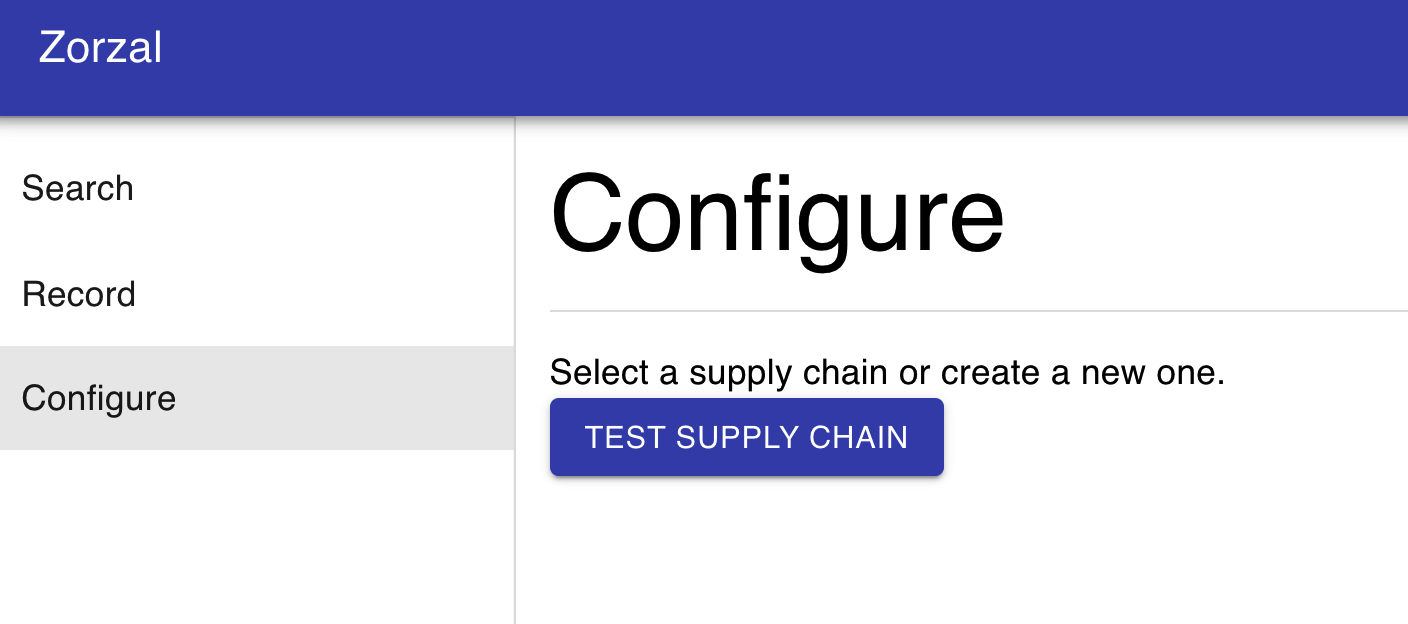
1. Right now if you click on Display without selecting the time frame, the graph will still render. It’s because the data used to create the graphs is hard coded. It’s not coming from the database that contains the action records. We need to finish the functions that will fetch the data and organize it according to the action points and render it as bars. So far, we wrote the functions that display a graph, fetch the data from the firebase in between the dates selected, and process the data into a format that could be understood by the function that displays the graph. Next, we need to connect the Display button with these functions written so that once the user clicks on “Display” or specific checkpoints, the functions we have already written fires rather than the hardcoded graph.
2. When you click on one of the buttons from the checkpoint menu (Fermentation, Drying, Containers, Warehouse), the selected checkpoint will remain shaded until you remove the mouse. But, it should remain shaded while its graph is still displayed because it indicates to the user which graph is being rendered. That is something we have to work on.
3. We want to incorporate a hover feature that will tell you the exact frequency of a bar when you hover over it.
4. An error should pop up if you try to input an end date that comes before the start date, but right now, the graph still displays.

**Configure**

The Configure portion of the web application was designed as a way for Zorzal employees, particularly Charles, to be able to reconfigure their supply chain, editing the structure of the application without ever opening about the code base or even the database. A large portion of the UI functionality has been incorporated, and what follows is a discussion of how to use the UI for configuration as well as a set of steps to make the component fully operational.

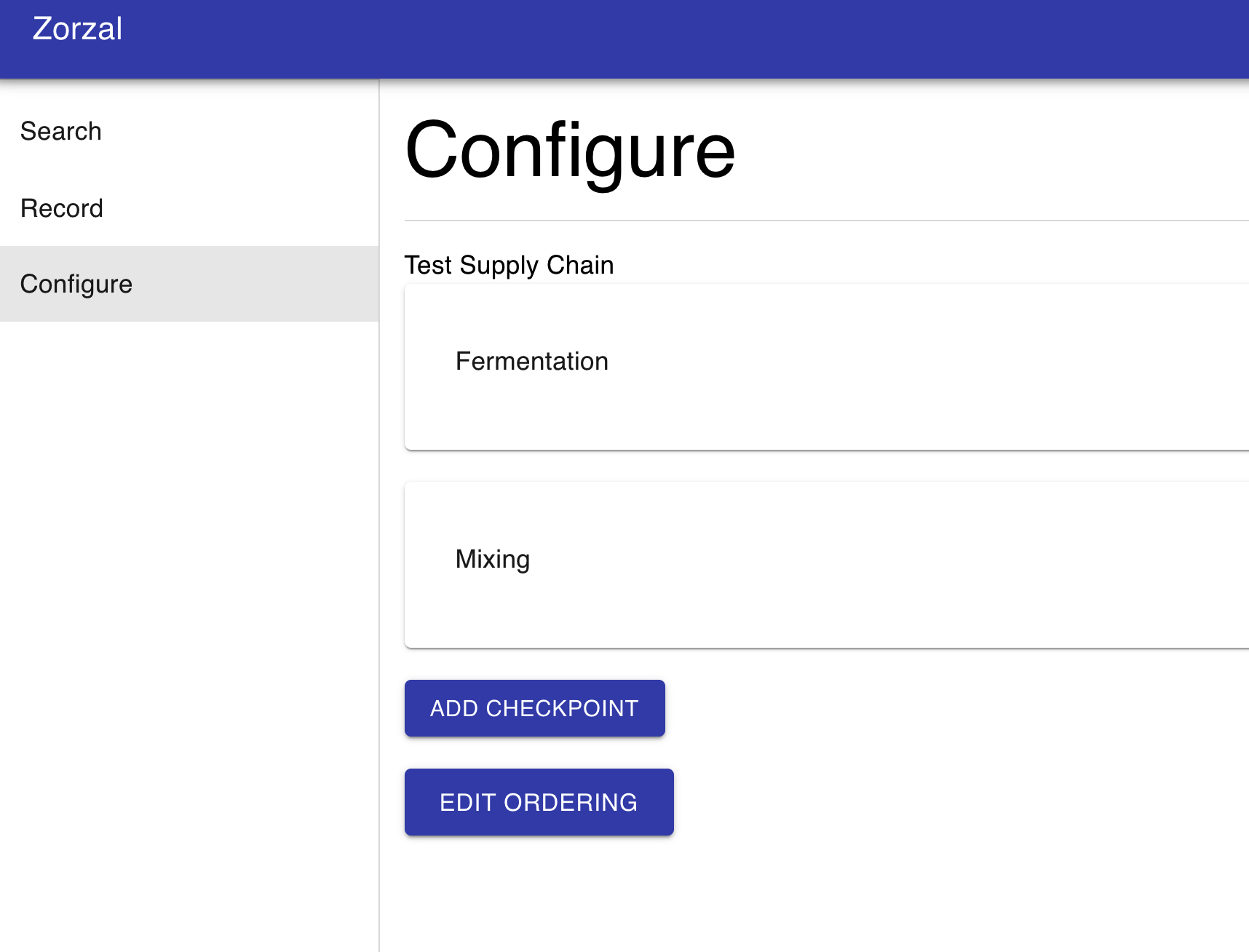
**Step 1:** Click on Configure

Navigate to the configure page by clicking on the configure button on the left menu on the screen.

****

**Step 2:** Choose the supply chain to edit.

Select the supply chain that you would like to edit by clicking on one of the blue buttons.

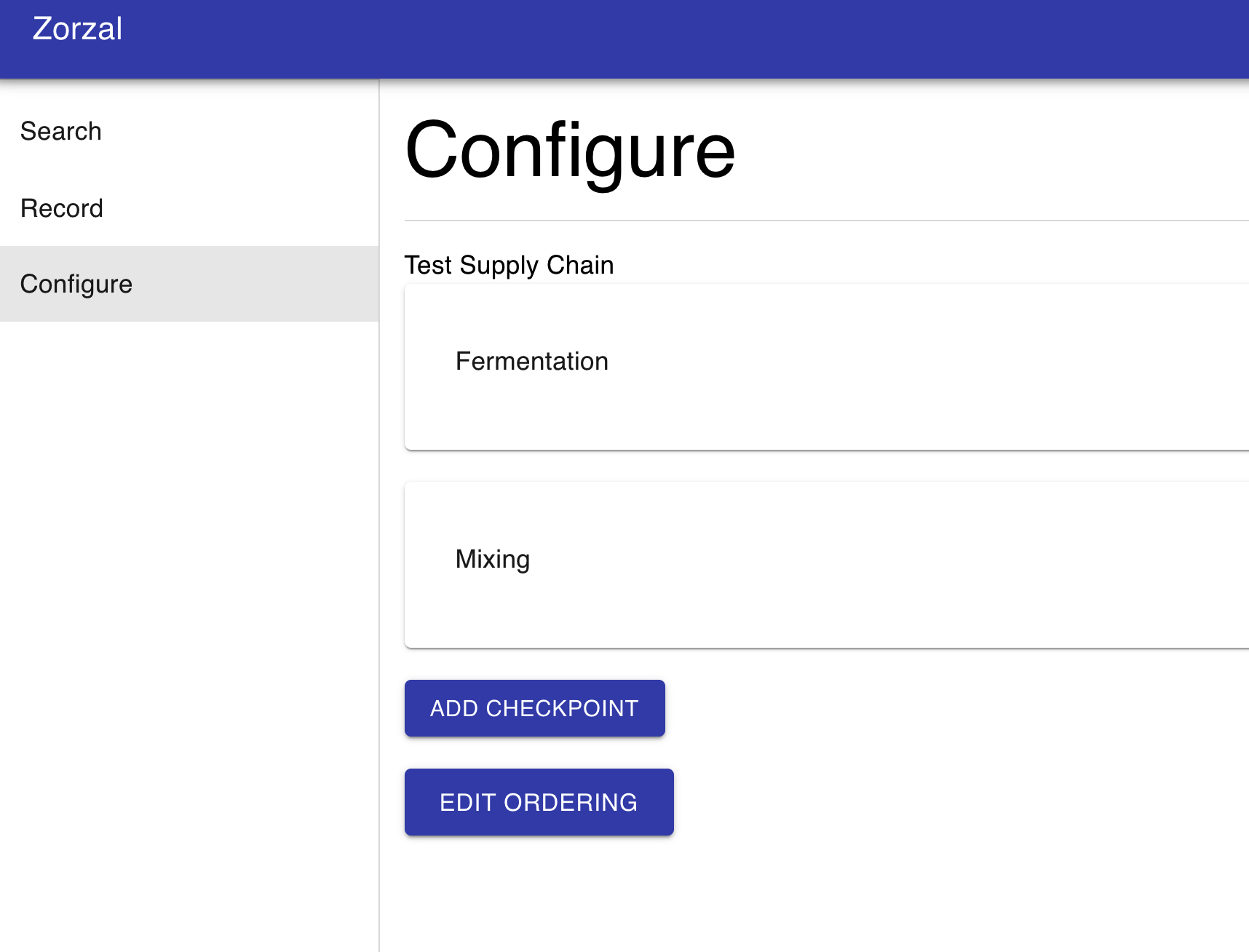


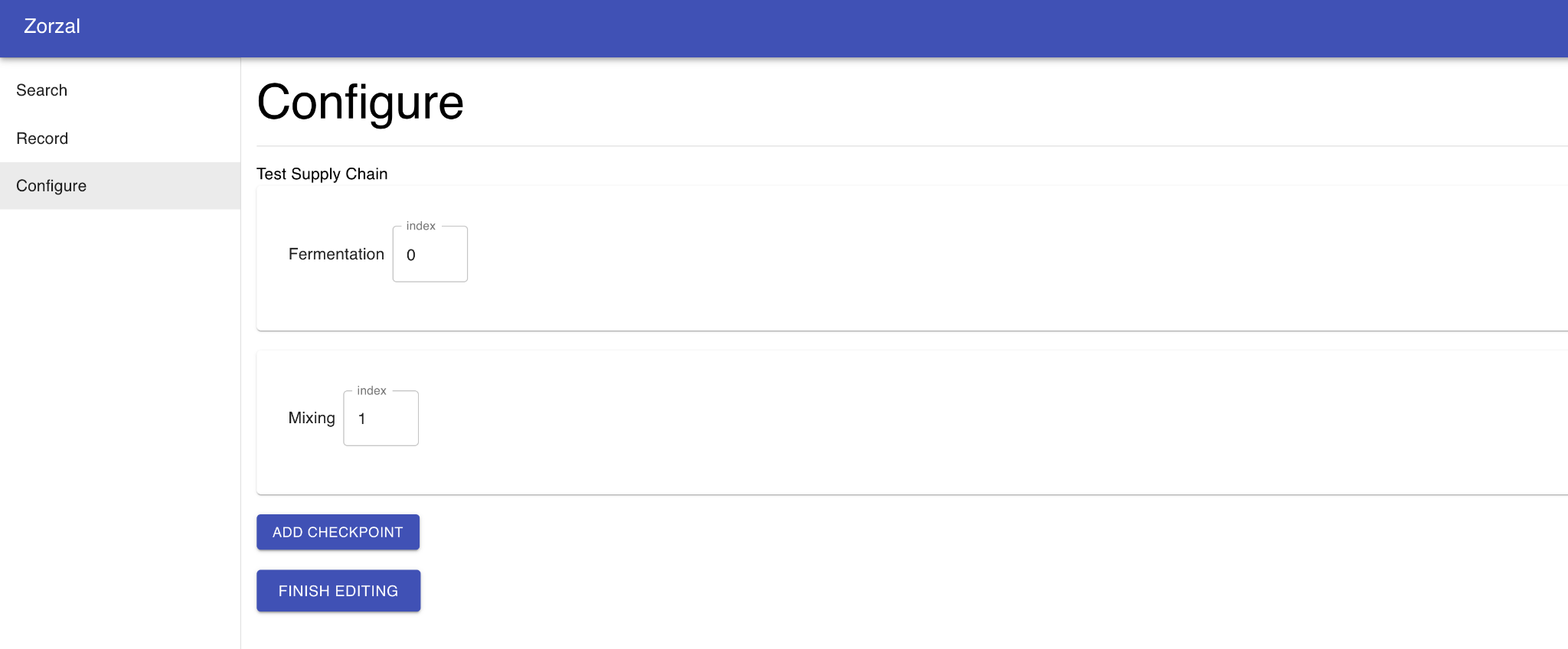
**Step 3:**

Now the user can edit the supply chain. There are various ways that the supply chain can be edited and each will be described separately in what follows:

**Edit Checkpoint Ordering**

To edit ordering click on the edit ordering button at the bottom of the screen.

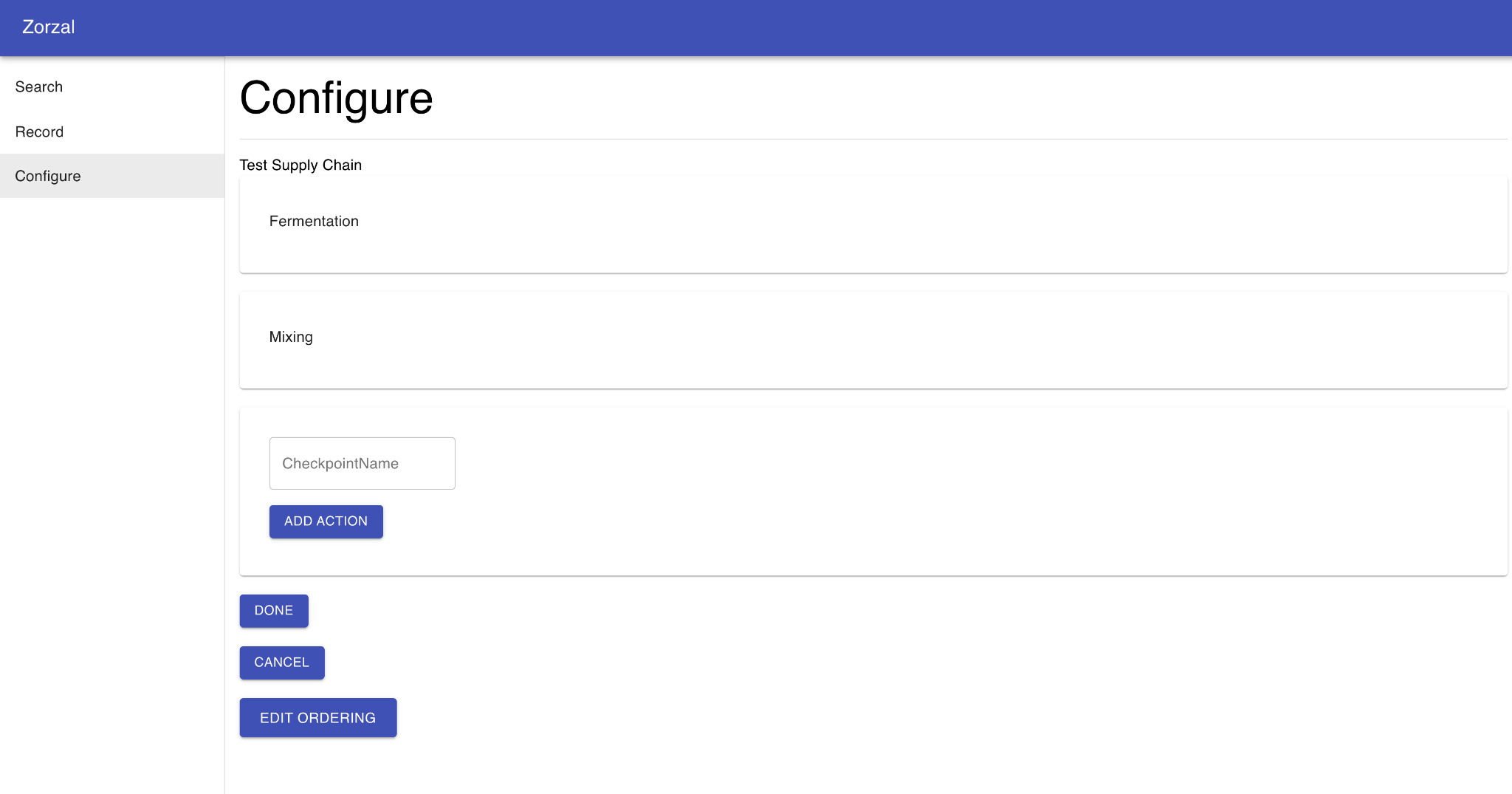


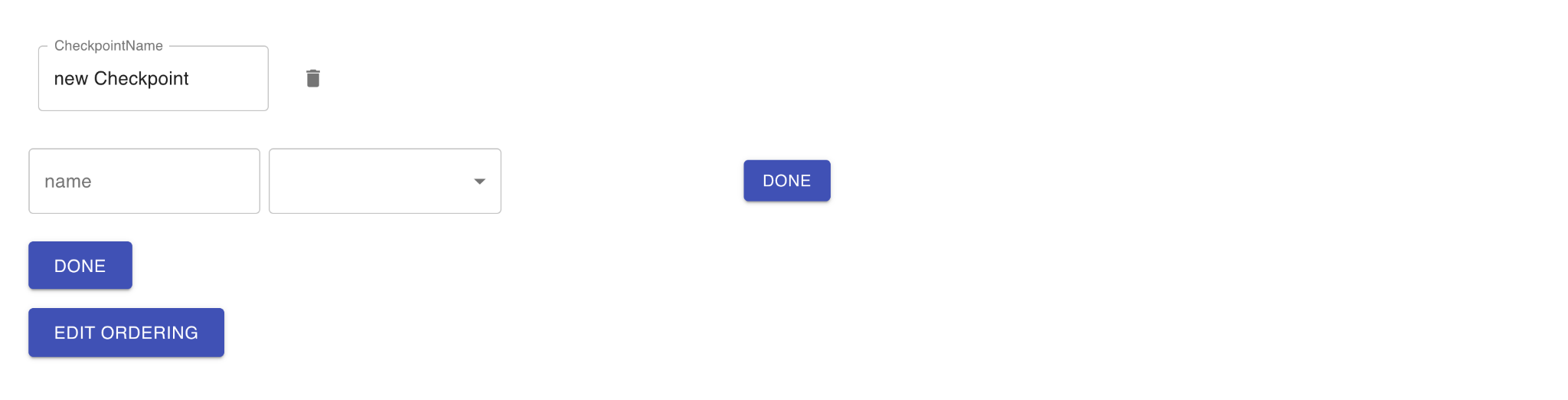
Upon clicking the Edit Ordering, the indices for the checkpoints will be rendered on the screen for editing. 

Now, the user is free to edit the ordering by changing the indices. The application will not allow invalid indices, sending an error notification, ensuring that arrays do not break due to invalid indexing. Note: (editing the action ordering can be performed in exactly the same way.)

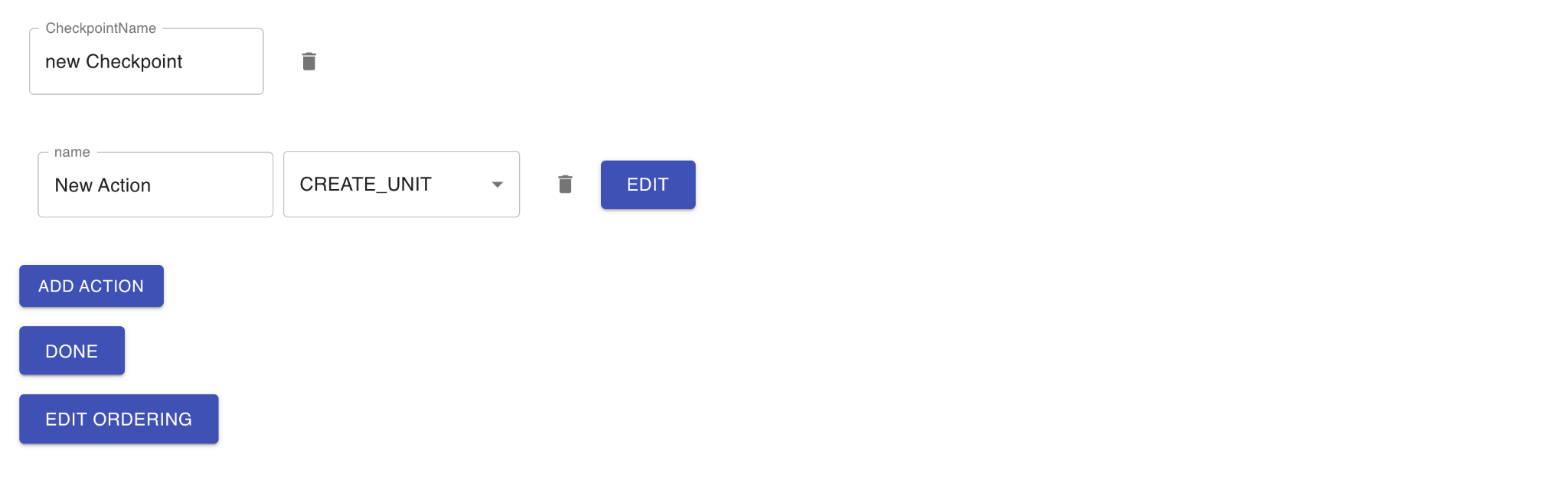
**Adding and Removing Checkpoints**

The next editing functionality is the ability to add and remove checkpoints. To add a checkpoint simply click on the add checkpoint button at the bottom of the screen.



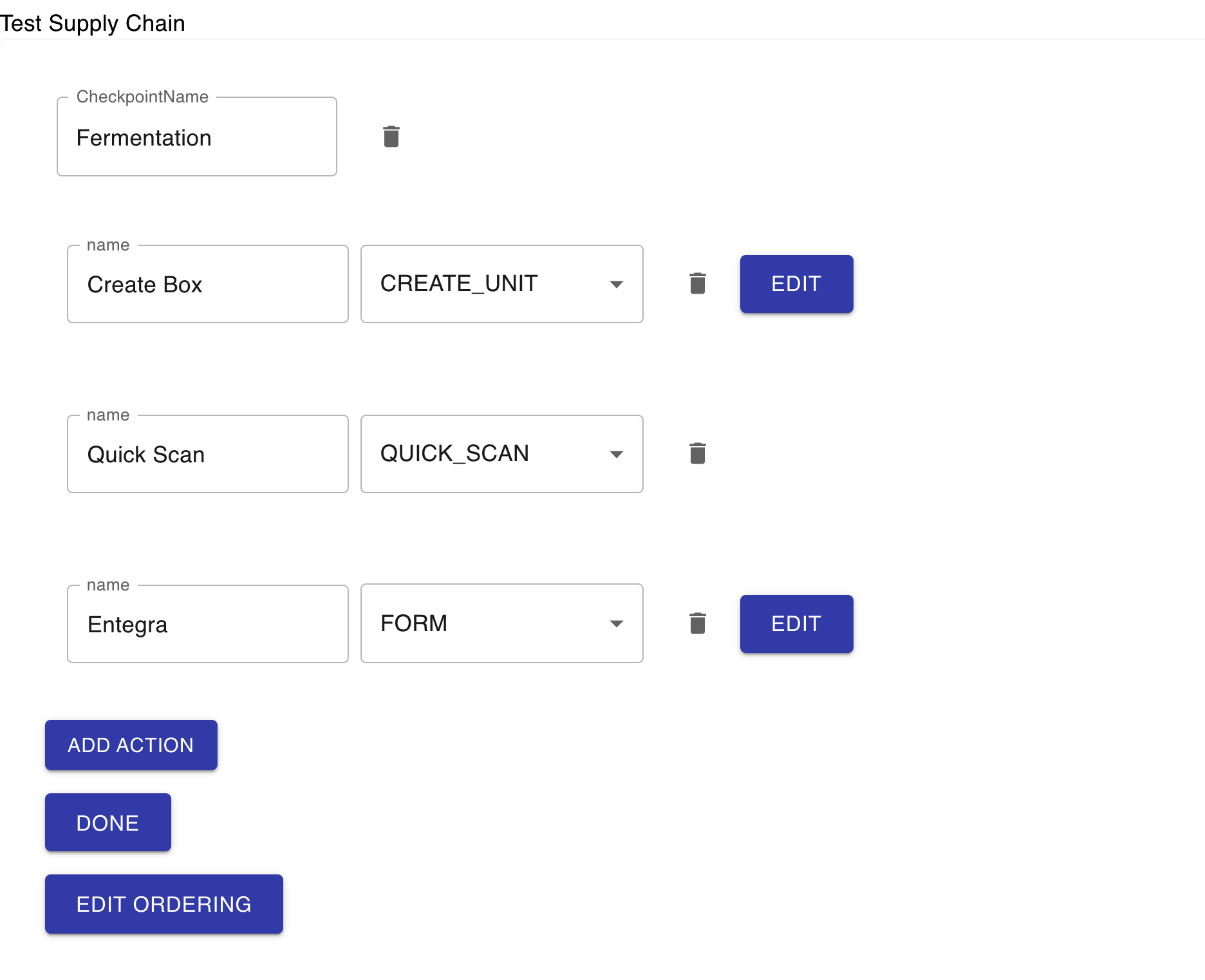
A new text field will be rendered on the screen for entering the name of the new checkpoint. To add this checkpoint, the user types in the name and clicks “done”. The user can also add actions in the same way for this new checkpoint by selecting the add action button within the checkpoint. 

After clicking on the add action button the user specifies the name of the action as well as the type of action before clicking the done button on the right of the screen. Clicking “done” will create the actual action within the checkpoint.

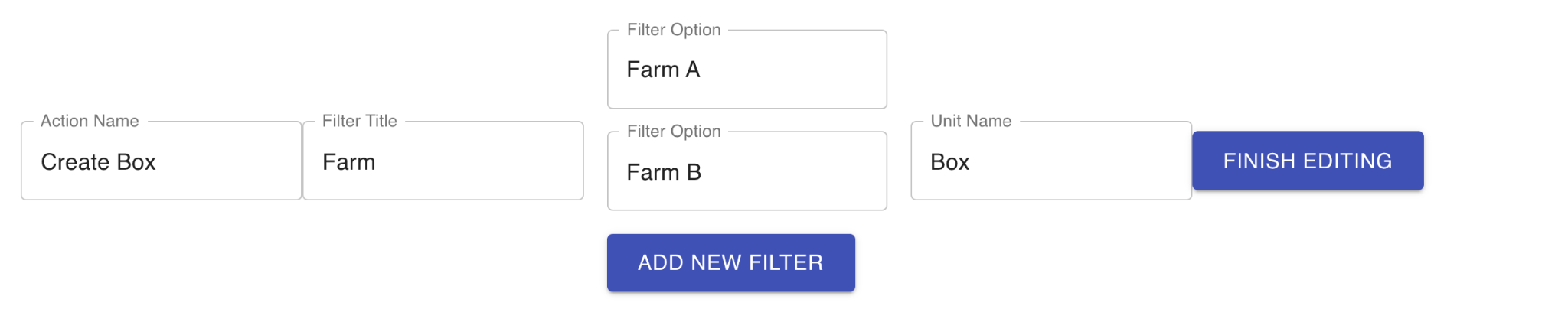


Now we have a newly created action called “New Action” associated with the checkpoint, “new Checkpoint.” To delete a checkpoint or action simply, click on the trash icon to the right of the checkpoint or action.

**Editing Existing Checkpoints and Actions**

To edit an existing checkpoint simply click on one of the checkpoints on the app. This will expand the checkpoint and make it editable. 

Now the user can change the name of the checkpoint by typing a new name inside of the text field at the top. Similarly the user can change the name of any action as well as their types. Certain actions like CREATE\_UNIT and FORM have parameters beyond the name and type that can be edited. (Note: editability has not been extended for the form action yet. The button is included as a placeholder for the future.) To edit the CREATE\_UNIT action, click on the edit button to the right of the action.



Now, all of the parameters for the action are editable. The user can still rename the action, but the user can also rename the dropdown title on the create unit page of the application (Filter Title). At the same time the user can edit existing filter options like what kinds of farms are available. Finally, the user can edit the name of the unit being created. When the user is done editing the action, simply click Finish Editing and the action will be updated.

**Step 4:** Finish Editing

The final step in this process would be to click a submit button that would update the database to reflect the changes made by the user. This aspect of the component has not been implemented.

**Remaining work on Configure:**

1. Since users are allowed to drastically change the framework for the application it is imperative to incorporate numerous checks to ensure that the user input is valid and makes sense given that we are creating a supply chain. For example the first action of the first checkpoint logically should be a CREATE\_UNIT page.
2. In addition, certain actions have requirements to be usable. For example, CREATE\_UNIT needs filter information and a unit name parameter. Checks need to be included that ensure that when adding or editing an action these requirements are always met.
3. There should be checks to prevent naming collisions between actions and checkpoints.
4. There should be an option to delete filter options when editing the CREATE\_UNIT action.
5. Forms need to be editable as a basic list of field values as well as including categories and subcategories in layout.
6. Connection to the backend needs to be established so that the changes persist beyond the editing session.
7. This component needs to be thoroughly tested with the backend to ensure that changes to the supply chain are accurately updated to the database and then accurately reflected in the new version of the app.
8. The code needs to be thoroughly documented as these components are relatively complex and require a large amount of state management.