Software User Manual

MyUI

2IPE0, Software Engineering Project - Q4 (2022)

Group 8

Full Name	**Student ID**
Aydoslu, Eren	1414593
Boelhouwers, Daan	1457152
Farla, Richard	1420380
Guha, Nishad	1408674
Hellenbrand, Jeroen	1414844
Kustermans, Quinten	1439561
Lamme, Jeroen	1443062
Nijsten, Samuel	1422324
Radu, Adriana	1418548
Thoumoung, Yemoe	1412051

Supervisor: de Beer, Huub
Project managers: Andrade, Ricardo & Ilin, Cristian
Customer: Budziak, Guido

Eindhoven, July 4, 2022

Abstract

This document is the Software User Manual (SUM) for MyUI, which describes in detail how to use the features of the application. The user can consult this document when they do not understand how to use a feature of the application or experience an error. MyUI is a web-based application that lets users generate user-specific UIs based on the underlying dataset and user permissions. This product has been developed by a group of students at the Eindhoven University of Technology as part of their Software Engineering Project.

Contents

Document status sheet

General

Document title:	Software User Manual	
Document identifier:	SUM/0.2	
Authors:	Aydoslu, Eren	1414593
	Boelhouwers, Daan	1457152
	Farla, Richard	1420380
	Guha, Nishad	1408674
	Hellenbrand, Jeroen	1414844
	Kustermans, Quinten	1439561
	Lamme, Jeroen	1443062
	Nijsten, Samuel	1422324
	Radu, Adriana	1418548
	Thoumoung, Yemoe	1412051
Document status:	Final Version	

Document history

Date	Authors	Changes
24-05-2022	Adriana Radu	Document structure, initial abstract
25-05-2022	Adriana Radu	Initial introduction
15-06-2022	Adriana Radu	Initial tutorials
20-06-2022	Adriana Radu	Appendix A
22-06-2022	Yemoe Thoumoung	References
23-06-2022	Adriana Radu, Yemoe Thoumoung, Richard Farla, Jeroen Lamme, Quinten Kustermans	Final changes
29-06-2022	Adriana Radu, Richard Farla	Implement supervisor feedback

Document change records

Version	Date	Section	Authors	Reason
0.1	23-06-2022	All	Adriana Radu, Yemoe Thoumoung, Richard Farla, Jeroen Lamme, Quinten Kustermans	Initial version
0.2	30-06-2022	All	Adriana Radu, Richard Farla	Implement supervisor feedback

1 | Introduction

1.1 | Intended readership

MyUI provides functionality for two distinct users: end-users and administrators. Note that these users are not mutually exclusive. We characterize these two types of users as follows:

End-users use MyUI to construct a personalized UI called a dashboard, based on their user profiles as defined in Hasura [1]. The end-user can modify and personalize the data representation of the data stored on the Postgres database via a combination of interaction elements present in MyUI and by inputting GraphQL queries. Hence, they are required to be proficient in GraphQL. In Chapter 2, these technologies are described further. Chapter 3 covers the most relevant use cases for the end-user.

Admins have the highest permissions in MyUI. Besides the normal functionality present for end-users, admins can change the system configuration within MyUI. Tutorial 3.34 discusses the changing of global settings. They are also able to edit, add and/or delete permission profiles in Hasura. This requires them to be proficient in handling Hasura.

1.2 | Applicability

This document applies to version 1.0 of MyUI, which was released on 24/6/2022.

1.3 | Purpose

MyUI is a web application that allows organizations to give different levels of access to their data while allowing users to personalize how to view this data. This personalization can be done mostly through UI interactions. Users should be able to perform data modifications on said data, without having to perform repetitive queries.

The purpose of the Software User Manual (SUM) is to guide the users of MyUI through their usage of the web application via several descriptive tutorials. Moreover, to increase understanding, these tutorials are accompanied by a set of screenshots of the associated web pages. When a user encounters errors while using MyUI, they can consult Appendix A to identify and recover from the error.

1.4 | How to use this document

This document contains three main sections: Tutorials, Reference, and Error messages with recovery procedures. The tutorials (Section 3) and the reference (Section 4) both discuss the functionalities of the web application. The difference between these two sections is that the tutorials are step-wise instructions on the exact procedure needed to use MyUI's features, while the reference section contains a description of every button and field and its functionality. Furthermore, the reference section contains tables for possible errors on each page and sub-page. These errors cover the error messages discussed in Appendix A thus, when an error is encountered, the reader can refer to the mentioned appendix to find further information on the errors and possible recovery procedures.

For any unclear terms, the reader can refer to Appendix B for the respective term's definition. In Appendix C often used terms are linked to pages they are referenced on. This can be used to quickly find the right page in the manual.

1.5 | Related documents

The User Requirements Document (URD) of MyUI [2] is related to the Software User Manual. The URD defines the requirements for MyUI, these requirements form the basis of the functionalities that are present in MyUI. How to use these functionalities will be explained in this document. All the output presented by MyUI is in accordance with the requirements presented in the URD.

1.6 | Conventions

Throughout this document, the *italic font* is used to indicate buttons, field names, and menu items. Red boxes and arrows are used to highlight the relevant components for each tutorial. Placeholder text is written between angle brackets (e.g. Username:<username>).

1.7 | Problem reporting

The team will no longer be working on MyUI after its completion. Since the software for MyUI will be open-sourced, if the user encounters any issues using the application, they should open an issue in the Github of MyUI. This Github is created and maintained by our customer, Connected.Football B.V., represented by Guido Budziak.

1.8 | List of references

[1] "Hasura." [Online]. Available: https://hasura.io/

[2] Eindhoven University of Technology, "MyUI: User Requirements Document," 2022.

[3] "Queries and Mutations." [Online]. Available: https://graphql.org/learn/queries/

2 | Overview

In this section, we will describe the problem statement that MyUI solves, how MyUI solves this problem, and the requirements to use MyUI. This will be done for both end-users and administrators.

2.1 | End-user

2.1.1 | Problem statement

End-users want to view data from their organization in a convenient way. They want to avoid performing repetitive tasks like writing queries to the database for standard actions on the data, such as deleting, editing, and inserting data.

2.1.2 | MyUI's solution

MyUI provides end-users the option to create custom dashboards. In these dashboards, end-users can view data from their organization by adding grid views based on queries to the database. After writing these queries, the grid views will be stored on the dashboard. An example of how a dashboard can look can be seen in Figure 2.1. The dashboards are saved for each individual end-user, such that the end-user does not have to rewrite the same query to view the data multiple times. These grid views will show a table containing the data retrieved by the query. The end-user can change the size of the grid view and the pagination settings of the table, allowing the end-user to view the data in a way they find comfortable. The end-user can also use the grid views to delete, edit and insert data, assuming they have the rights to do so, with the click of a button removing the strain of writing a query for these actions.

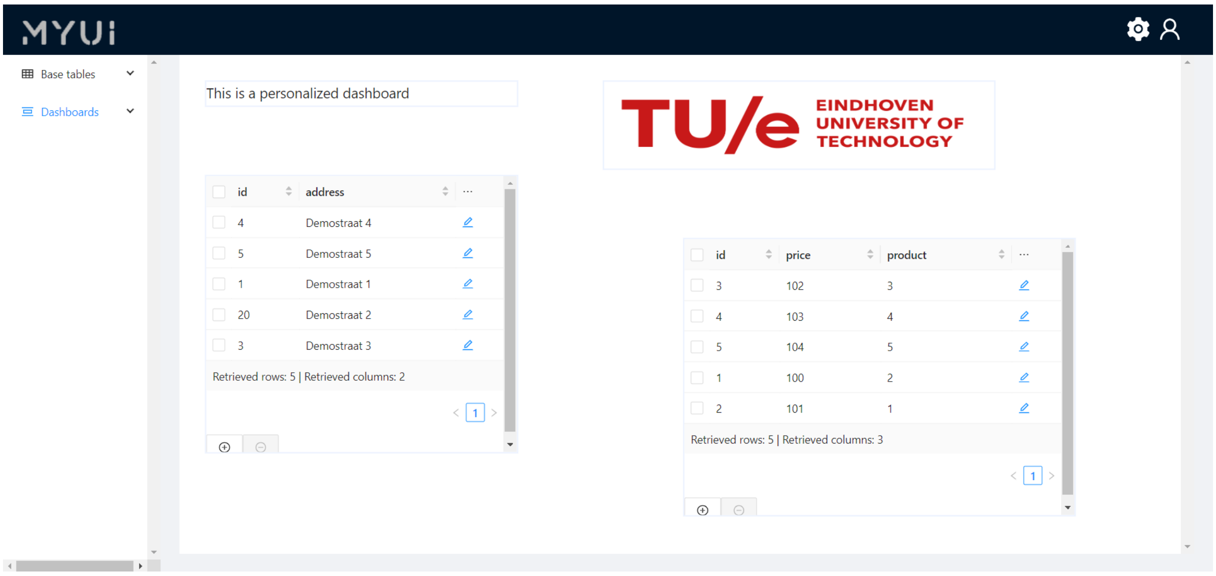

Figure 2.1: An example of a personalized dashboard

2.1.3 | Requirements for using MyUI

For end-users to use MyUI, the end-user should have login credentials provided by an organization that uses MyUI and the end-user should have internet access to reach the website that serves MyUI. Additionally, using MyUI requires having a basic understanding of GraphQL because it depends on end-users to create GraphQL queries in order to create grid views. For end-users that feel like their grasp on GraphQL is limited, it is recommended to follow online tutorials on how to write GraphQL queries [3].

2.2 | Administrators

2.2.1 | Problem statement

Administrators want to provide members of their organization access to data of the organization, however, they want to give different users different access to the data in the same database. Furthermore, they want users to be able to view the data without having to code a UI representation of the data for each user.

2.2.2 | MyUI's solution

Hasura is a web application server that connects to a Postgres database and implements a data access scheme. MyUI makes use of Hasura to be able to provide different end-users with different access levels for the data of the organization. Therefore all queries to the database go through Hasura. Access levels are set by dividing the end-users into different roles which the admin can create. For each role the admin can specify the different types of access the end-user has for each table and each column. Thereby solving part of the problem statement. The other part of the problem statement is solved by allowing users to create their own dashboards, which do not require any coding from the administrators.

2.2.3 | Requirements for using MyUI

For administrators to use MyUI it is required for the data of the organization to be stored in a Postgres database. Knowledge of Hasura and how to configure user roles are also required such that the permission system can be made.

3 | Tutorials

This section contains step-by-step tutorials for each of the functionalities the end-user can perform using MyUI. For easy understanding, the tutorials are supported by screenshots of the application. Each tutorial contains a small description of the interaction, followed by the preconditions, procedure, expected outcomes, and lastly, likely errors the user may encounter. It is assumed that the preconditions are not violated for any of the tutorials. However, for certain precondition violations, the manual nevertheless provides error messages to guide the user to proper usage of the functionality. For more details about each individual error message, the user can consult Appendix A.

3.1 | Log in

Functional description:

This tutorial describes how a user can log in to MyUI.

Cautions and warnings:

None.

Preconditions:

- The user has valid credentials on MyUI.
- The user is not currently logged into MyUI.

Procedure:

1. Navigate to the MyUI web page.
2. The user will be presented with the login page of MyUI (Figure 3.1).
3. Enter the <username> in the *Username* field.
4. Enter the <password> in the *Password* field.
5. Click on the *Submit* button (Figure 3.2).

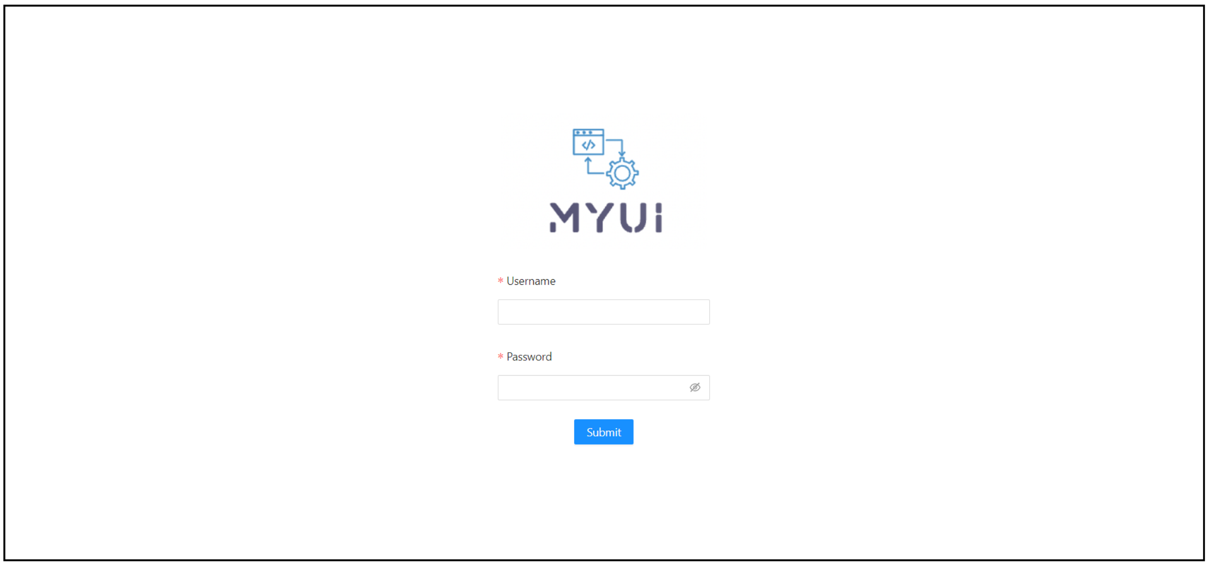

Figure 3.1: The login page of MyUI

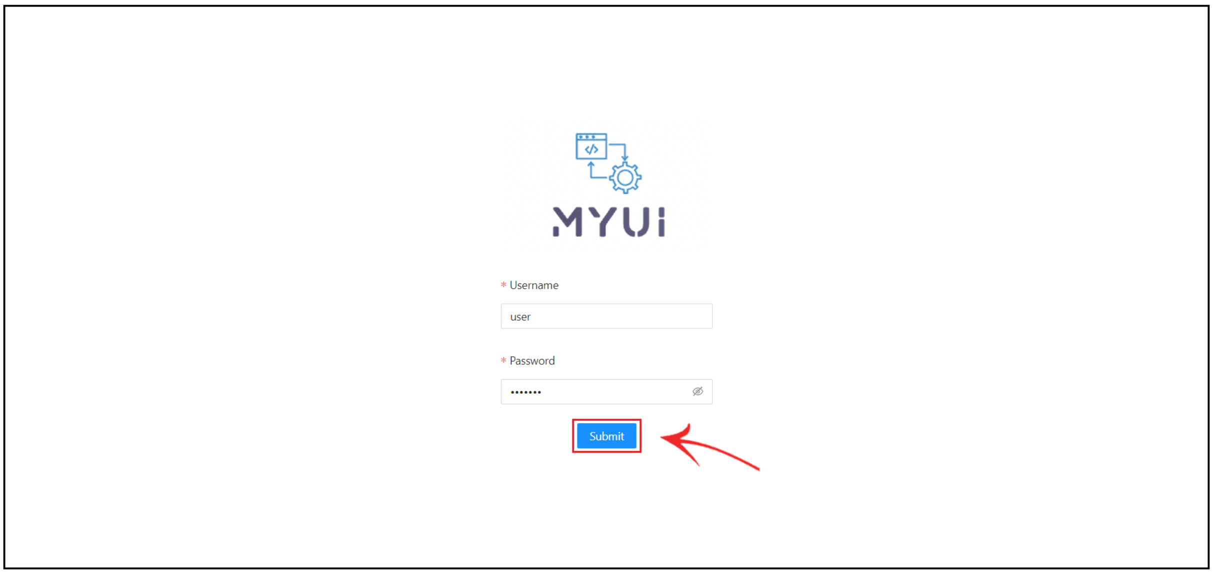

Figure 3.2: The login page of MyUI with the required credentials filled in

Expected outcome: After clicking on the *Submit* button, the page refreshes and shows the MyUI homepage (Figure 3.3).

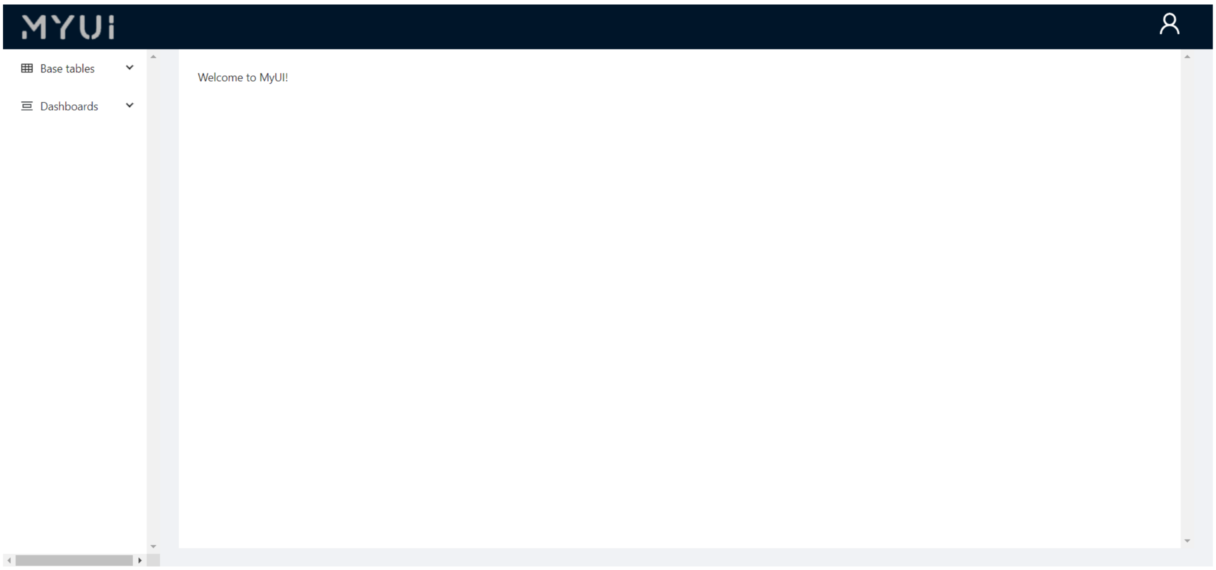

Figure 3.3: The homepage of MyUI

Likely errors:

Entering an incorrect username will result in an error page, displaying "Error localhost:3000". On the other hand, entering an incorrect password will cause the page to refresh. Clicking the *Submit* button without entering either one of the fields will cause error messages to show (A.1).

3.2 | Log out

Functional description:

This tutorial describes how a user can log out of MyUI.

Cautions and warnings

The contents of the menu shown in this description may differ depending on the user's permission profile (regular users do not have access to the *Global Settings* button) (Figure 3.4).

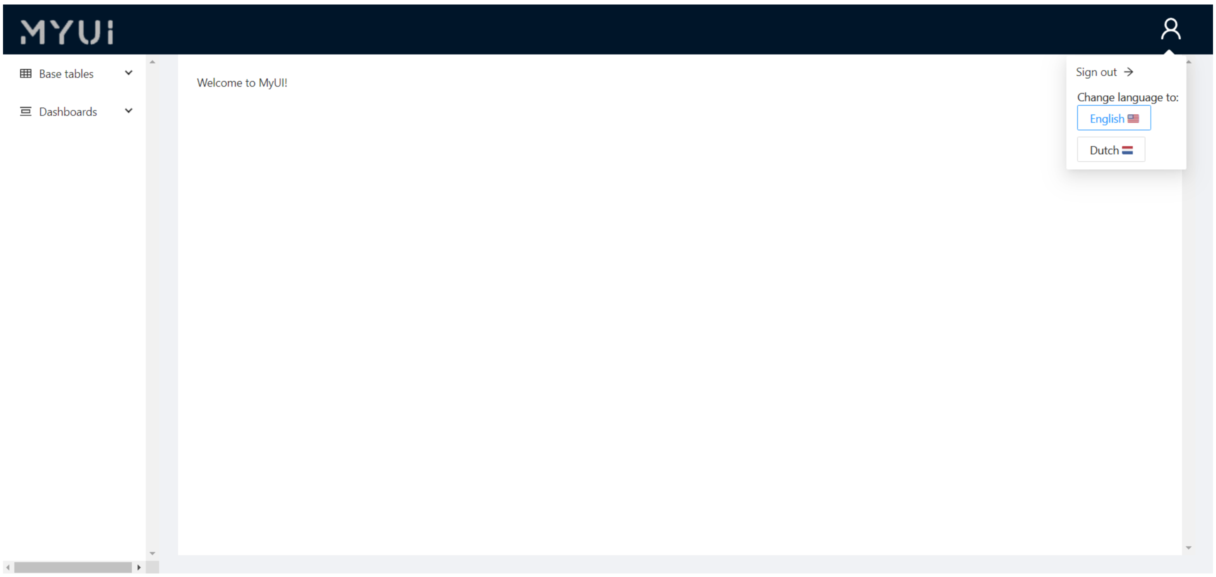

Figure 3.4: The dropdown menu contents seen from a regular user's account

Preconditions:

- The user is currently logged into MyUI.

Procedure:

1. Click on the user icon in the header of MyUI (Figure 3.5).
2. The user will be presented with a dropdown menu.
3. Click on the *Sign out* button (Figure 3.6).

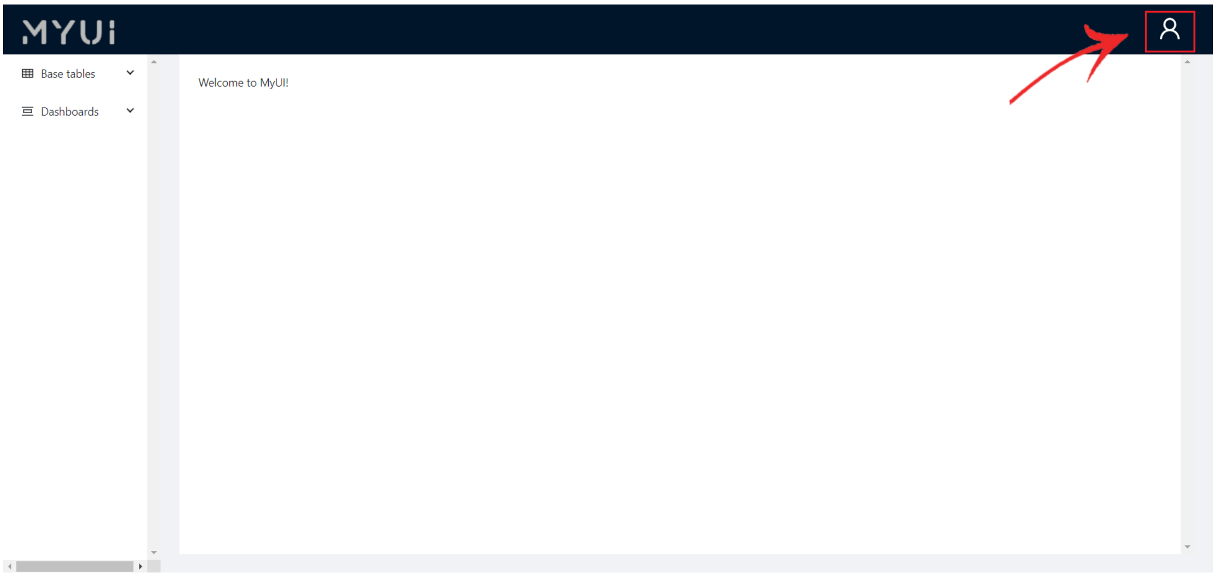

Figure 3.5: The user icon in the header of MyUI

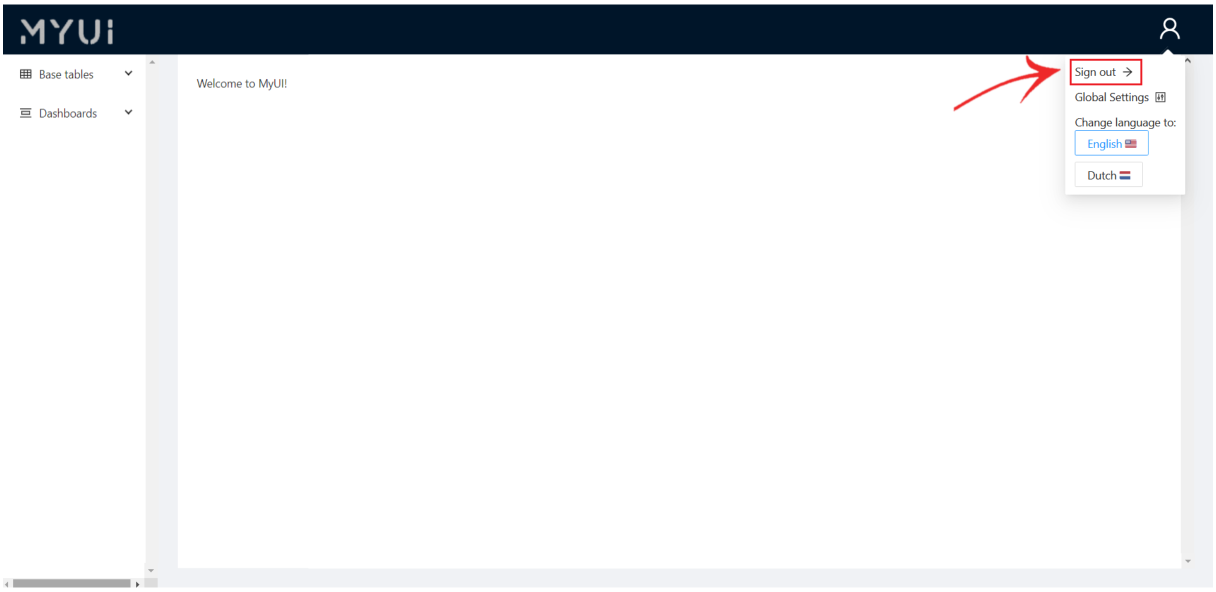

Figure 3.6: The sign out button

Expected outcome: After clicking on the *Sign out* button, the user is forwarded to the login page (Figure 3.1 on page 10).

Likely errors:

None.

3.3 | Change Language

Functional description:

This tutorial describes how a user can change the language of their MyUI application.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The language of the application is different from the language the user wishes to switch to.

Procedure:

1. Click on the user icon in the header of MyUI (Figure 3.5 on page 13).
2. The user will be presented with a dropdown menu.
3. Click on the desired language (Figure 3.7).

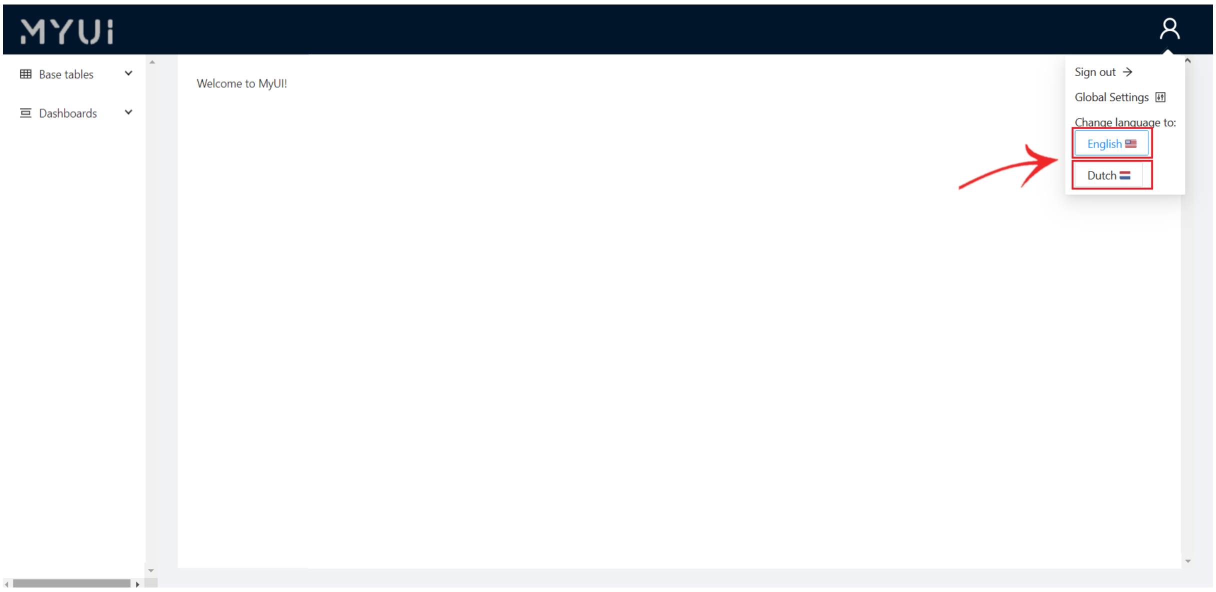

Figure 3.7: The language buttons

Expected outcome: After clicking on the desired language, the application will be displayed in that language and the user will be redirected to the home page (Figure 3.8).

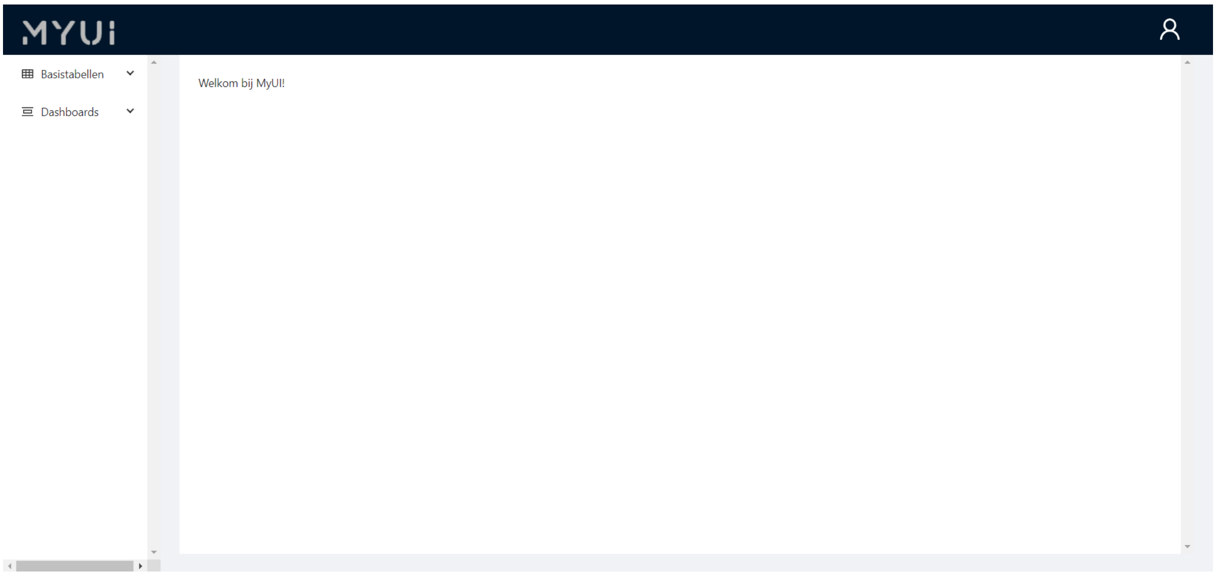

Figure 3.8: MyUI displayed in Dutch after changing the language

Likely errors:

None.

3.4 | Unfold dropdown menu

Functional description:

This tutorial describes how a user can unfold a closed dropdown menu in MyUI. A closed dropdown menu can be identified by an open arrow pointing downwards present next to the name of the menu.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The dropdown menu is folded.

Procedure:

1. Click on the dropdown menu (Figure 3.9).

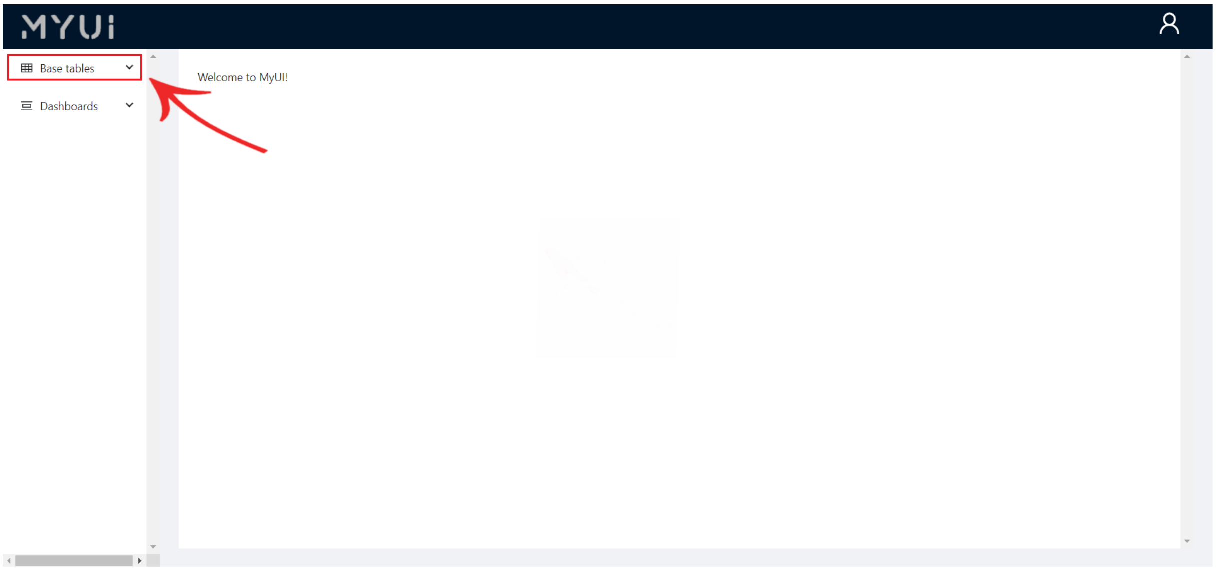

Figure 3.9: The *Base tables* dropdown menu

Expected outcome: After clicking on the dropdown menu, the menu unfolds and displays its respective options as clickable elements (Figure 3.10).

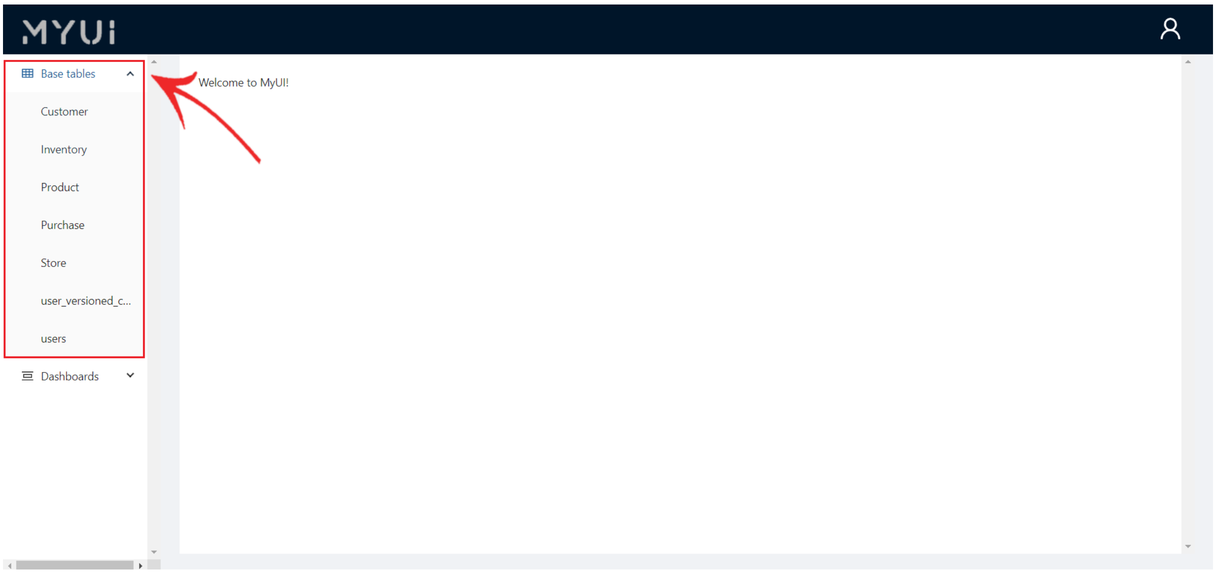

Figure 3.10: The list of *Base tables*

Likely errors:

None.

3.5 | Fold dropdown menu

Functional description:

This tutorial describes how a user can fold an open dropdown menu in MyUI. An open dropdown menu can be identified by an arrow pointing upwards present next to the name of the menu.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The dropdown menu is unfolded.

Procedure:

1. Click on the dropdown menu (Figure 3.10 on page 17).

Expected outcome: After clicking on the dropdown menu, the menu folds, and its elements are hidden (Figure 3.9 on page 16).

Likely errors:

None.

3.6 | Loading a base table

Functional description:

This tutorial describes how a user can view the contents of a base table in MyUI.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- There is at least one base table the user can access.

Procedure:

1. If the dropdown menu labeled as *Base tables* is not unfolded, unfold the menu (Tutorial 3.4 on page 16).
2. The user will be presented with a list of names corresponding to the tables present in the database that they can view according to their permissions (Figure 3.10 on page 17).
3. The user should click the option in the dropdown menu that is labeled with the name of the table they wish to view.

Expected outcome: After clicking on the name of the desired table, this name is highlighted in the *Base tables* dropdown menu and the respective table will appear in the user's working area with the data queried from the database (Figure 3.11).

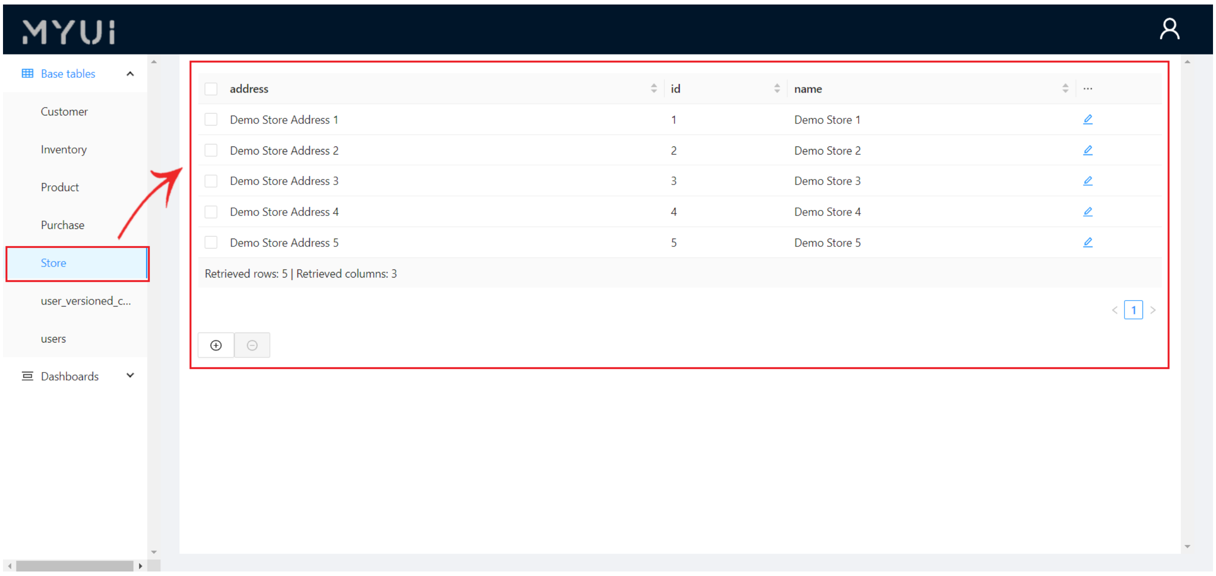

Figure 3.11: The table "Store" loaded in the user's working area

Likely errors:

It may occur that the selected base table does not render in the user's working area and a loading animation is shown indefinitely.

3.7 | Select a row

Functional description:

This tutorial describes how a user can select a row in a grid view.

Cautions and warnings:

If the user clicks outside the MyUI application, the selection is reset.

Preconditions:

- The user is currently logged into MyUI.
- The user has a grid view loaded in their working area.
- The user is not in edit mode.
- The grid view contains data.

Procedure:

1. Click on the checkbox next to the desired row (Figure 3.12).

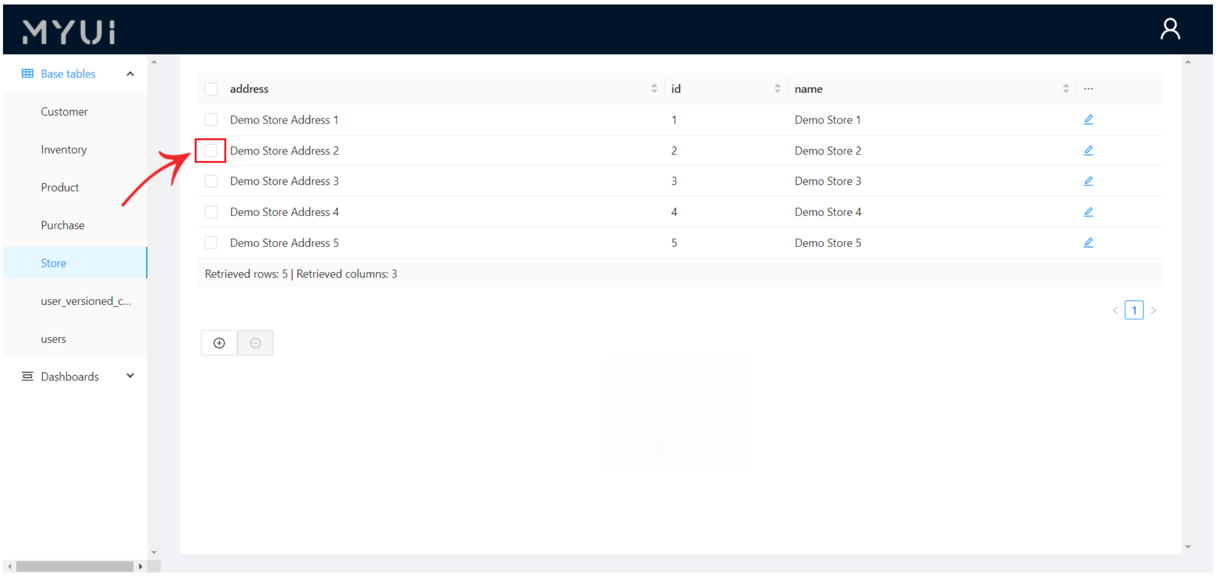

Figure 3.12: The checkbox for selecting a row

Expected outcome: After clicking on the checkbox, this box is checked and the respective row is highlighted (Figure 3.13). If no rows were selected previously, the checkbox in the grid view's header will now indicate that a row is selected. If all rows are now selected, that checkbox will indicate that all rows are selected.

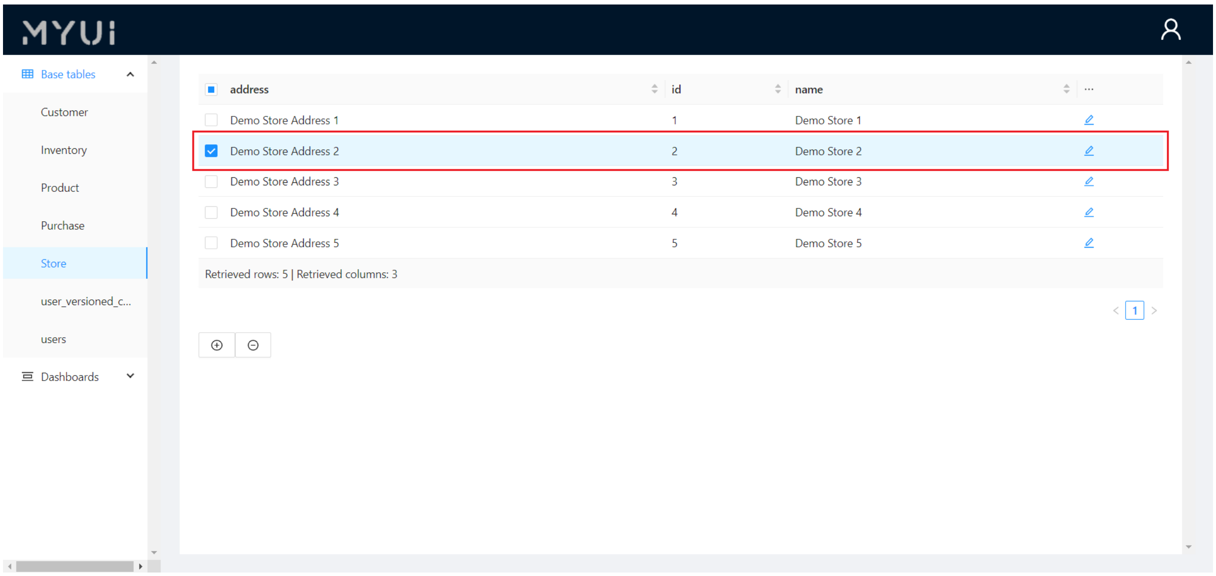

Figure 3.13: The selected row is highlighted

Likely errors:

None.

3.8 | Deselect a row

Functional description:

This tutorial describes how a user can deselect a selected row in a grid view.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The user has a grid view loaded in their working area.
- The user is not in edit mode.
- The grid view contains data.
- There is a selected row.

Procedure:

1. Click on the checkbox of the selected row (Figure 3.14).

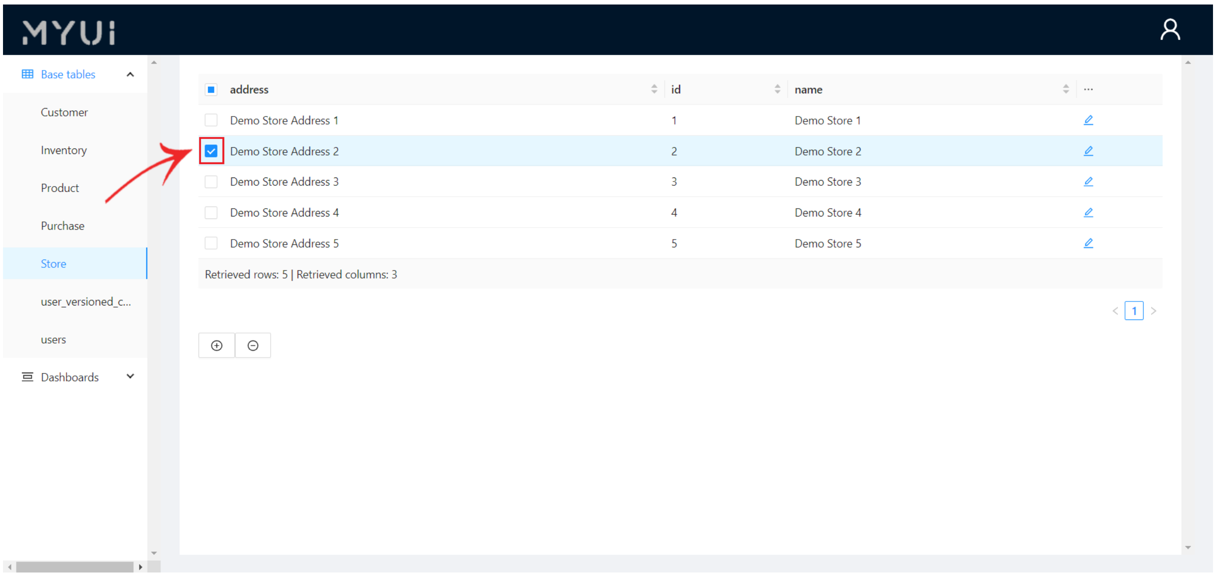

Figure 3.14: Click to deselect the previously selected row

Expected outcome: After clicking on the checkbox, the respective row is deselected. If all rows were selected previously, the checkbox in the grid view's header will now indicate that multiple, but not all rows are selected. If no rows are now selected, that checkbox will indicate it is unchecked.

Likely errors:

None.

3.9 | Select all rows

Functional description:

This tutorial describes how a user can select all rows in a grid view at once.

Cautions and warnings:

If the user clicks outside the MyUI application, the selection is reset.

Preconditions:

- The user is currently logged into MyUI.
- The user has a grid view loaded in their working area.
- The grid view contains data.
- Not all rows are selected.

Procedure:

1. Click on the checkbox in the grid view header (Figure 3.15).

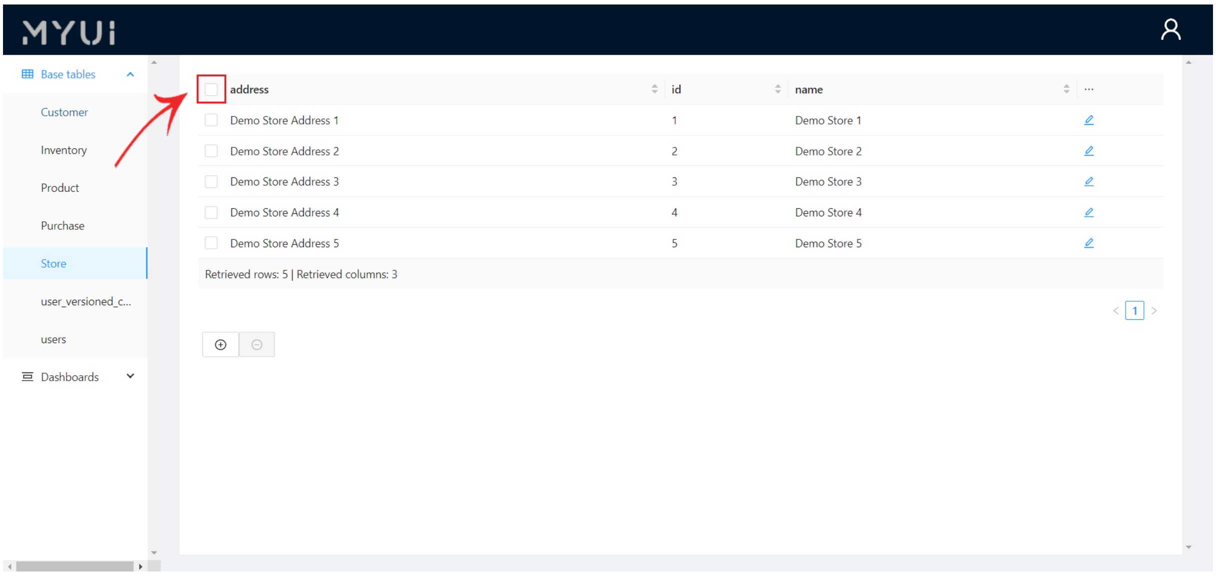

Figure 3.15: The checkbox to select all rows at once

Expected outcome: After clicking on the checkbox, all rows are highlighted (Figure 3.16).

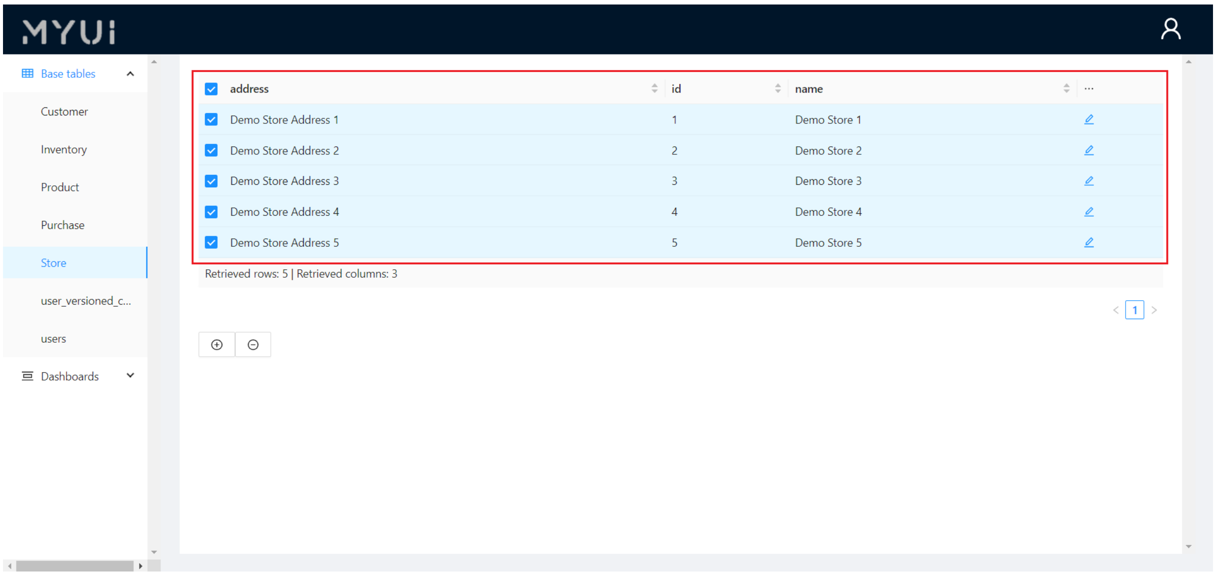

Figure 3.16: All the rows of the grid view are selected

Likely errors:

None.

3.10 | Deselect all rows

Functional description:

This tutorial describes how a user can deselect all selected rows in a base table at once.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The user has a grid view loaded in their working area.
- The grid view contains data.
- All rows are selected.

Procedure:

1. Click on the checkbox in the grid view header (Figure 3.17).

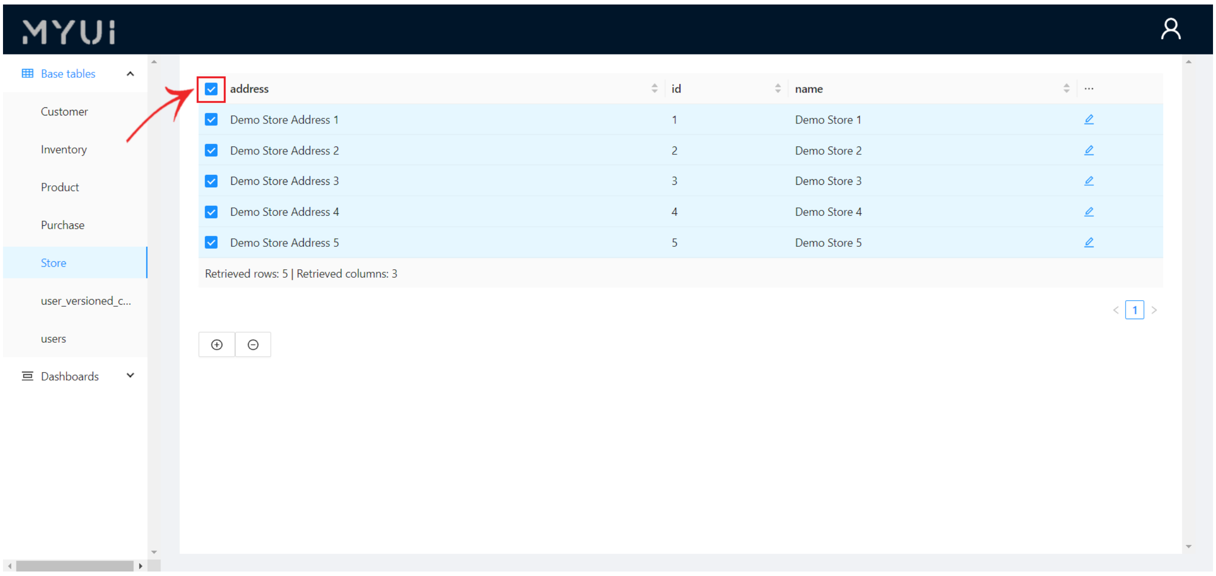

Figure 3.17: Click to deselect all rows at once

Expected outcome: After clicking on the checkbox, all rows that were previously selected are deselected, and thus no longer highlighted either.

Likely errors:

None.

3.11 | Insert a row

Functional description:

This tutorial describes how a user can insert a new row in any grid view they have access to.

Cautions and warnings:

If the user does not have the right insert permissions for the respective grid view, the + button will not be shown.

Preconditions:

- The user is currently logged into MyUI.
- The user has a grid view loaded in their working area.
- The user is not in edit mode.
- The grid view contains a table.
- The user has insert permission for the respective grid view.

Procedure:

1. Click on the + button at the bottom of the grid view (Figure 3.18).
2. The user will be presented with the *Add a row* prompt (Figure 3.19).
3. Enter the data of the new row in each text field.
4. Click on the *Submit* button (Figure 3.20).

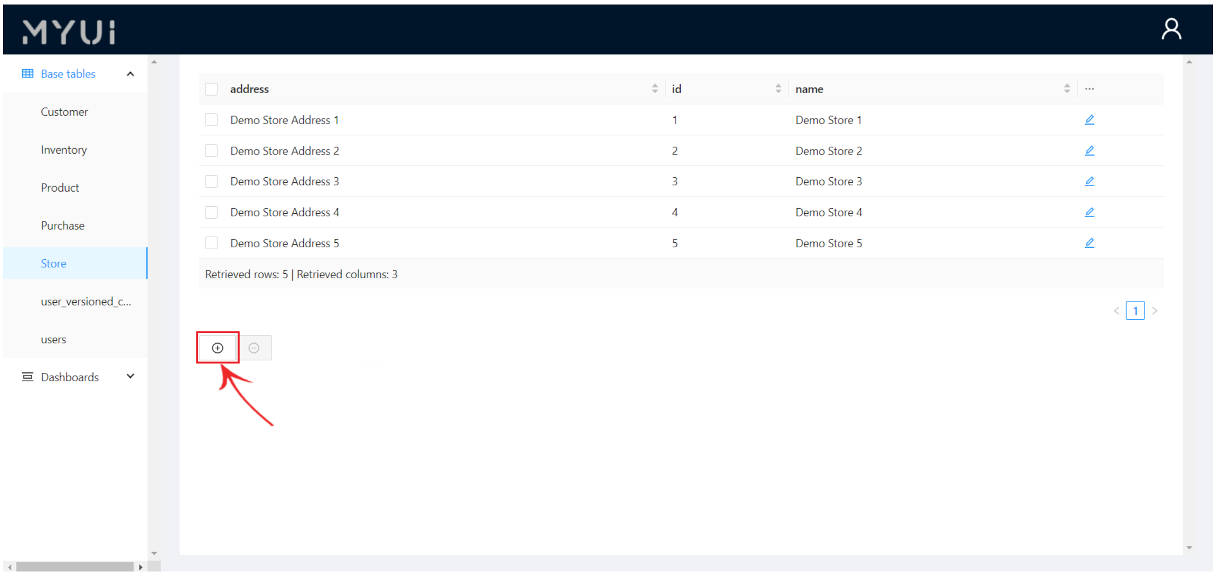

Figure 3.18: The + button for inserting new rows

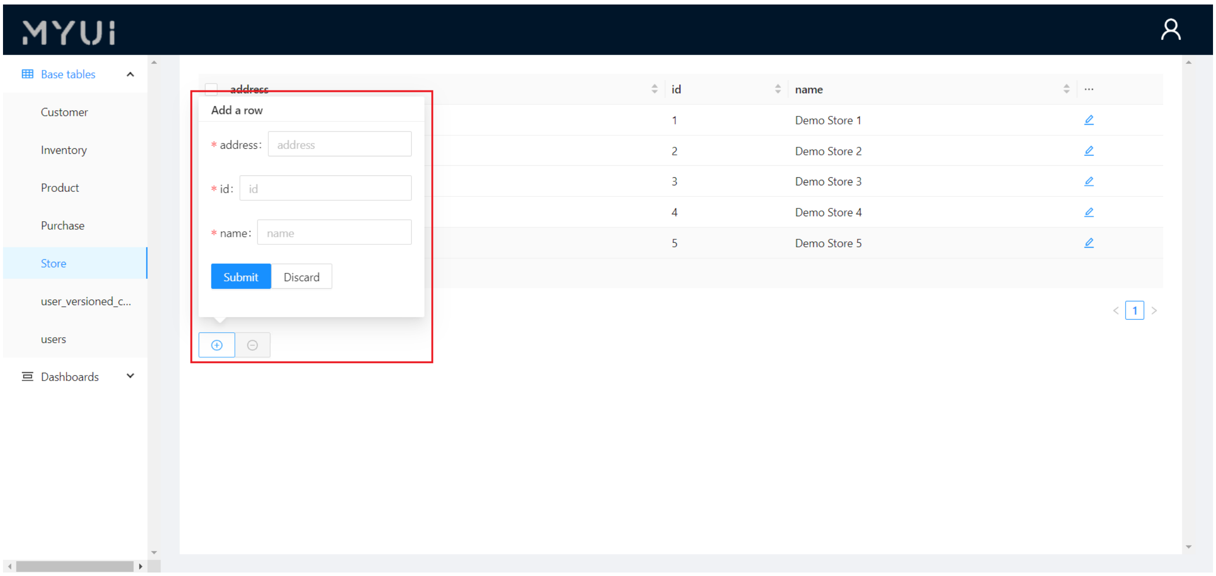

Figure 3.19: The *Add a row* prompt

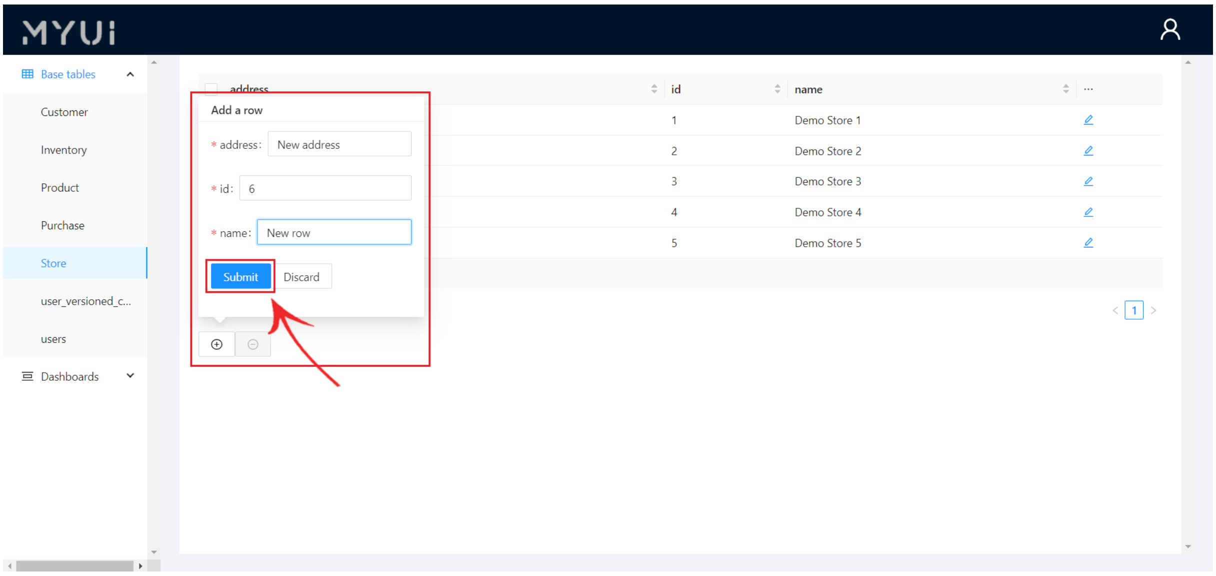

Figure 3.20: The *Add a row* prompt with data filled in

Expected outcome: After clicking on the *Submit* button, the *Add a row* prompt disappears and a notification is shown stating that the row has been added. The new row is shown in the grid view with the data put in by the user (Figure 3.21).

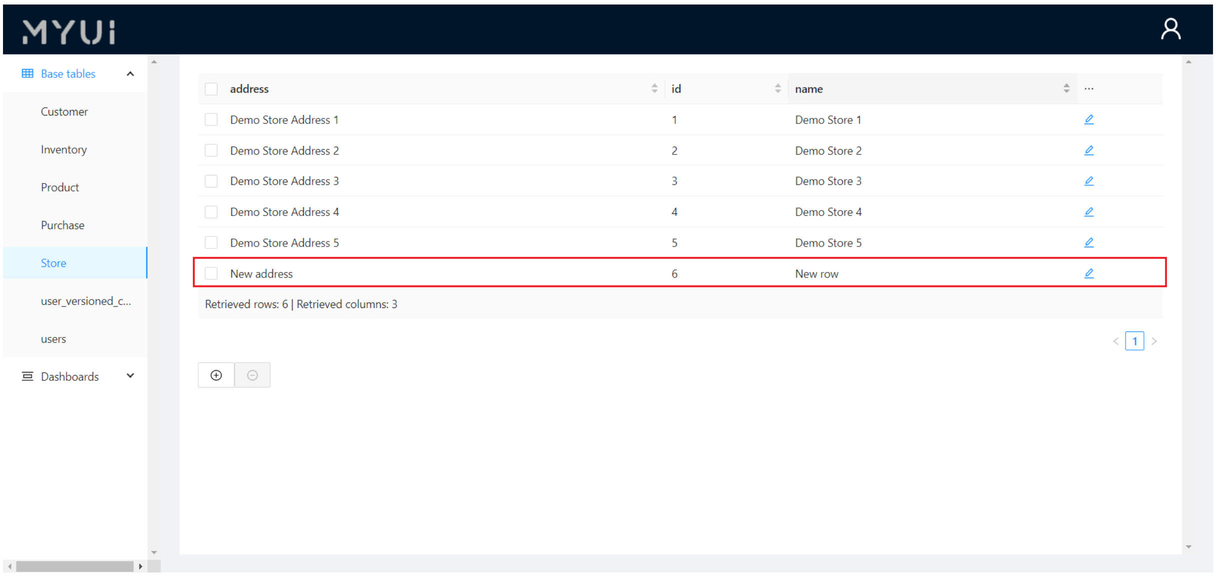

Figure 3.21: The newly inserted row

To cancel the addition click on the *Discard* button (Figure 3.22). The *Add a row* prompt will disappear.

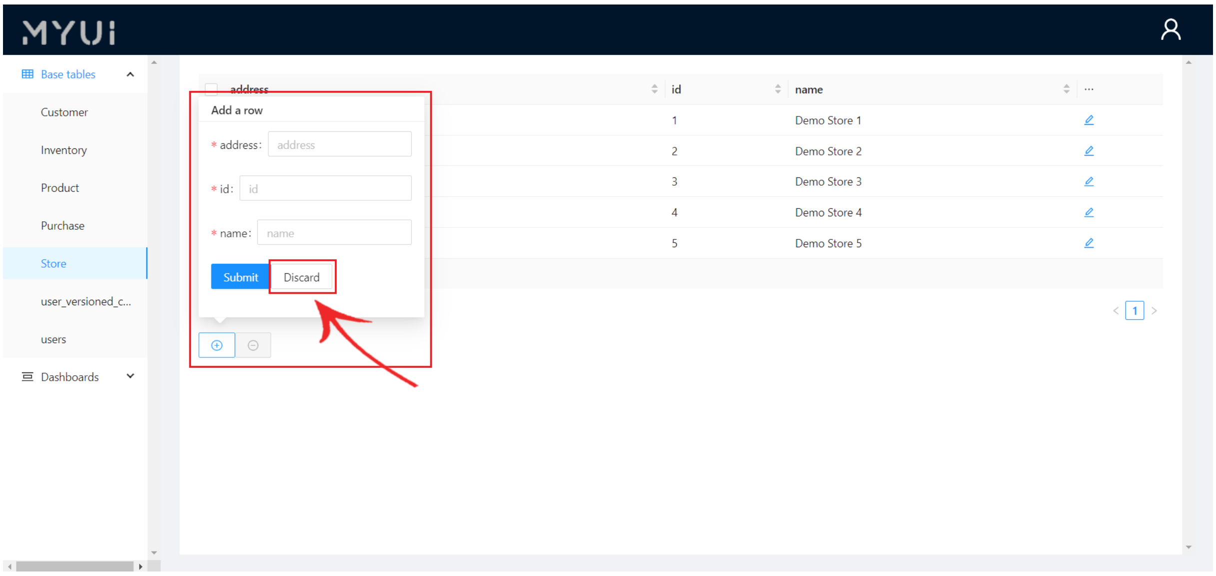

Figure 3.22: The *Discard* button for canceling the row addition

Likely errors:

All the text fields are required to be filled in for the row to be added to the grid view. The user will get error messages for each text field that is not filled in. These error messages can be found in Appendix A.2. Column entries might violate unique constraints or foreign key violations, resulting in a Hasura error message, which can be found in Appendix A.2.

Attempting to insert a row whilst a non-nullable column is not present in the table will result in a Hasura error message, which can be found in Appendix A.5. Filling in the text fields with the wrong type of input will cause a Hasura error message to be displayed. These error messages can be found in Appendix A.5.

3.12 | Delete selected rows

Functional description:

This tutorial describes how a user can delete the selected rows from any grid view they have access to.

Cautions and warnings:

If no row is selected, the - button is disabled. Once the rows are deleted, the action cannot be undone.

Preconditions:

- The user is currently logged into MyUI.
- The user has a grid view loaded in their working area.
- The user has delete permission for the respective grid view.
- The user is not in edit mode.
- The grid view contains a table.
- There are selected rows in the grid view.

Procedure:

1. Click on the - button at the bottom of the grid view (Figure 3.23).
2. The user will be presented with the *Delete a row* prompt.
3. Click on the *Yes* button (Figure 3.24).

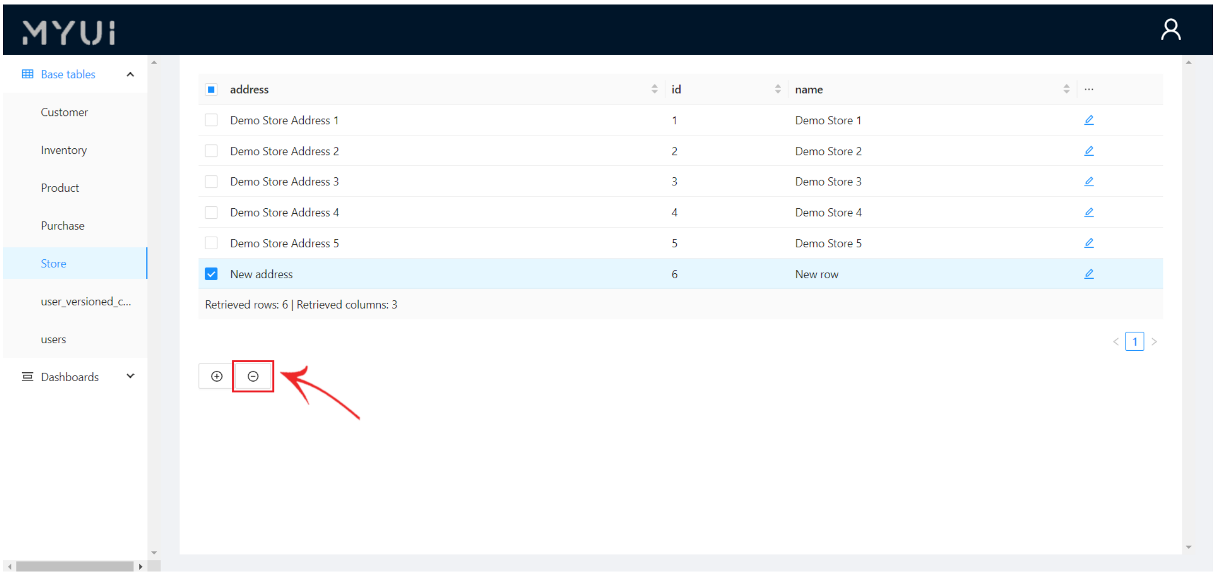

Figure 3.23: The - button for deleting a row

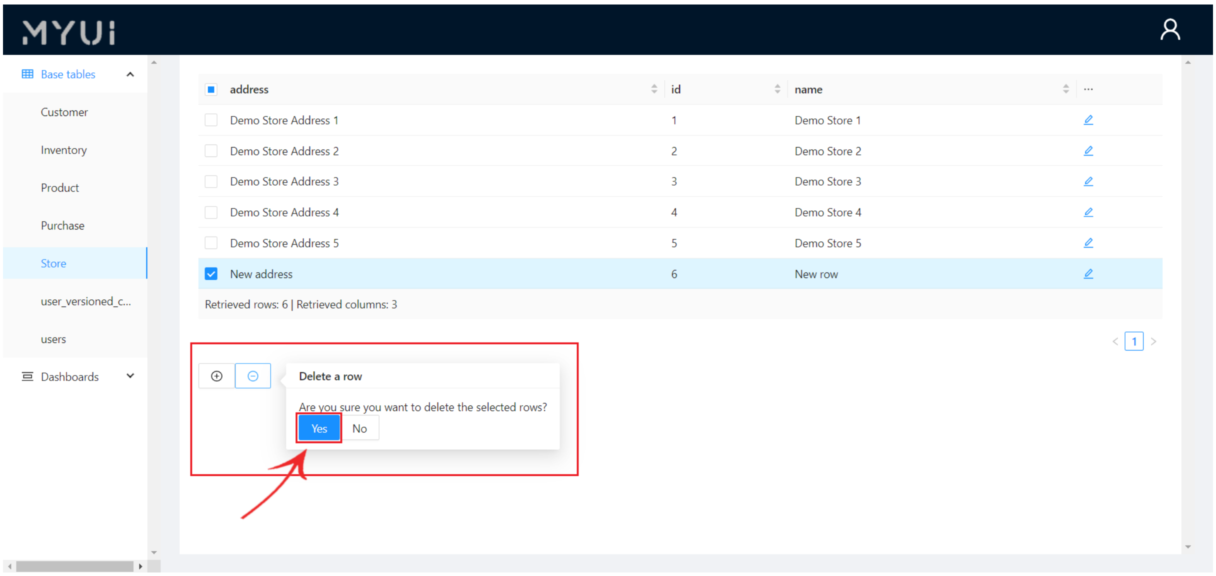

Figure 3.24: The *Delete a row* prompt with the *Yes* button

Expected outcome: After clicking on the *Yes* button, the *Delete a row* prompt disappears and a notification is shown stating that the selected rows have been deleted. The selected rows are deleted from the grid view. This action cannot be undone.
To cancel the deletion click on the *No* button. The *Delete a row* prompt will disappear (Figure 3.25).

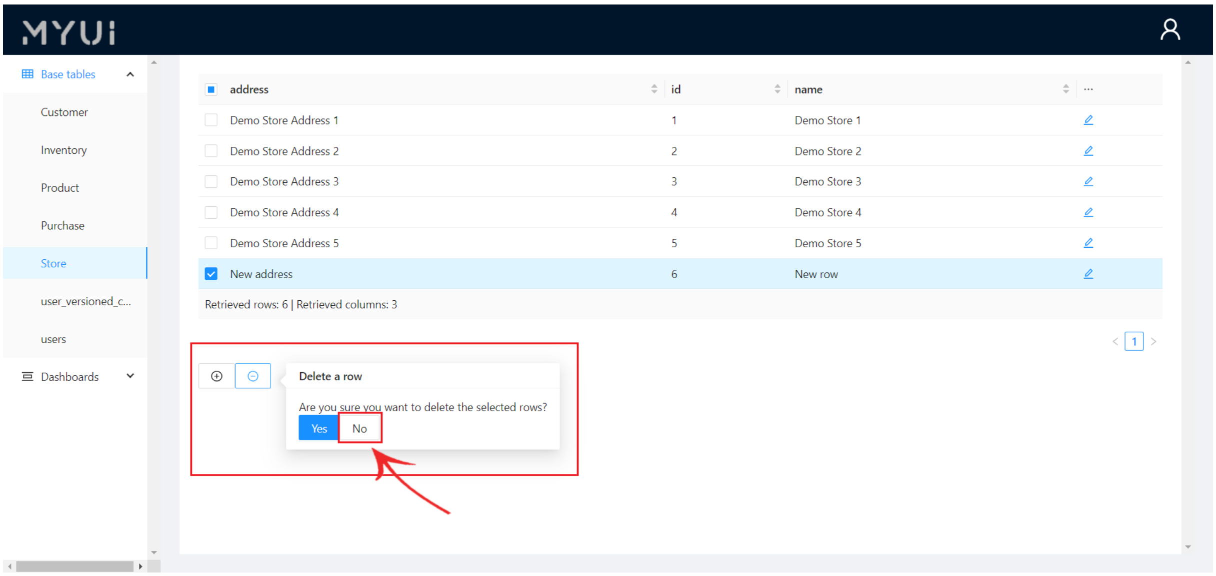

Figure 3.25: The *No* button for canceling the deletion

Likely errors:

Attempting to delete a row from the *user_versioned_config* table will cause a Hasura error message to show (A.5).

3.13 | Edit row

Functional description:

This tutorial describes how a user can edit the data inside any row from any grid view they have update permissions to.

Cautions and warnings:

If the user does not have the right update permissions for the respective grid view, the edit-row button will not be shown.

Preconditions:

- The user is currently logged into MyUI.
- The user has a grid view loaded in their working area.
- The user has update permission for the grid view.
- The user is not in edit mode.
- The grid view contains a table.

Procedure:

1. Click on the edit-row button on the desired row to be edited (Figure 3.26).
2. Text fields for each attribute will appear on the row.
3. Fill in the desired data in each text field (Figure 3.27).
4. Click on the save-row button (Figure 3.28).

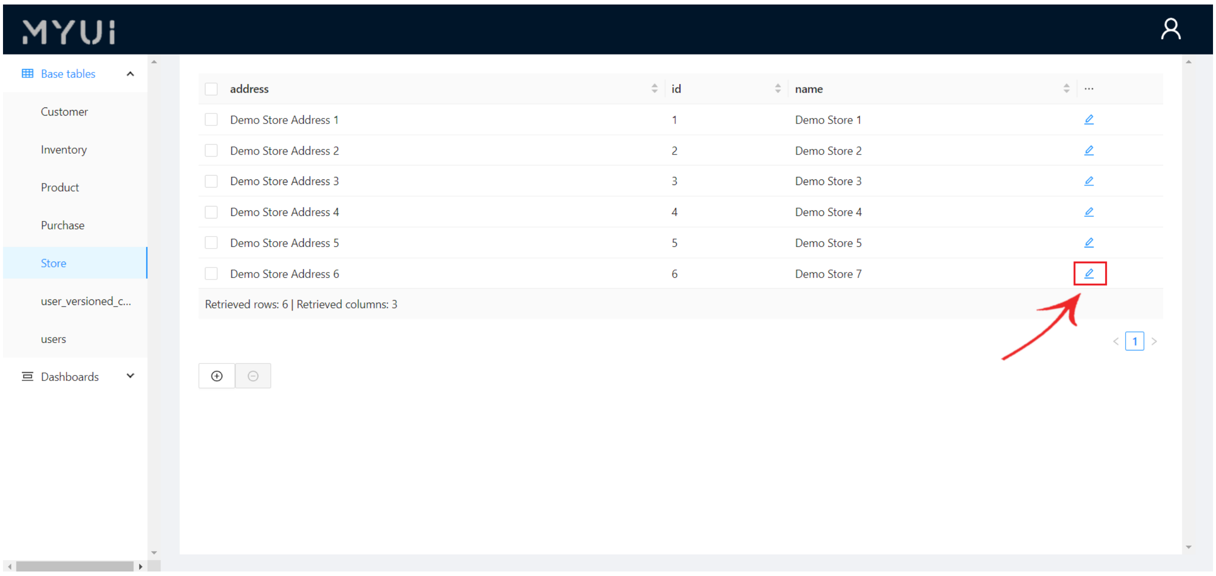

Figure 3.26: The edit-row button

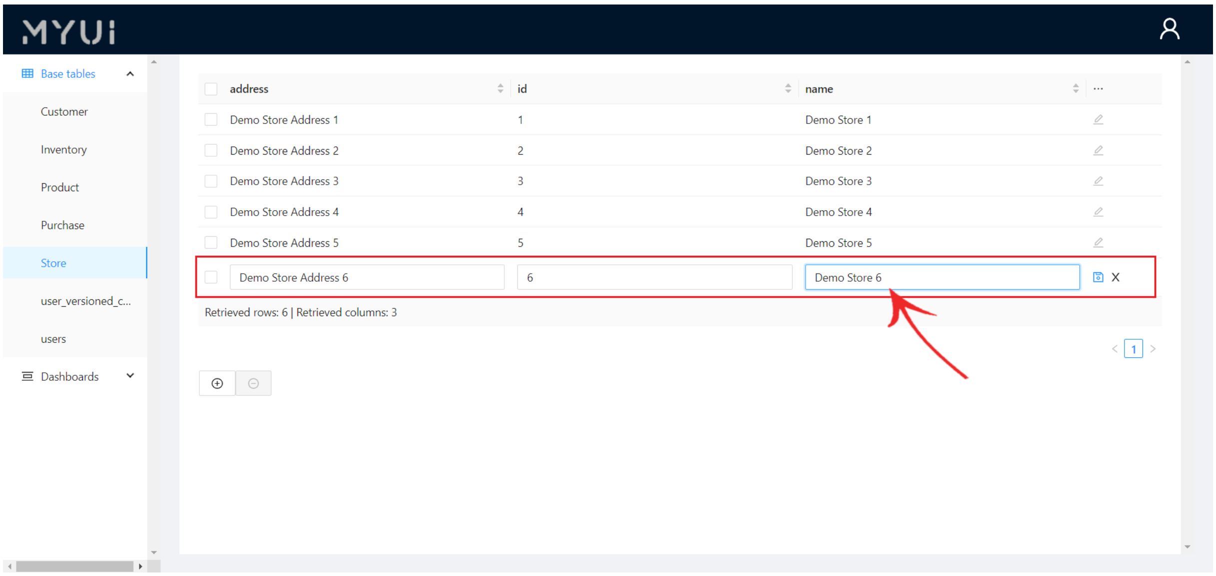

Figure 3.27: The value of the attribute is changed

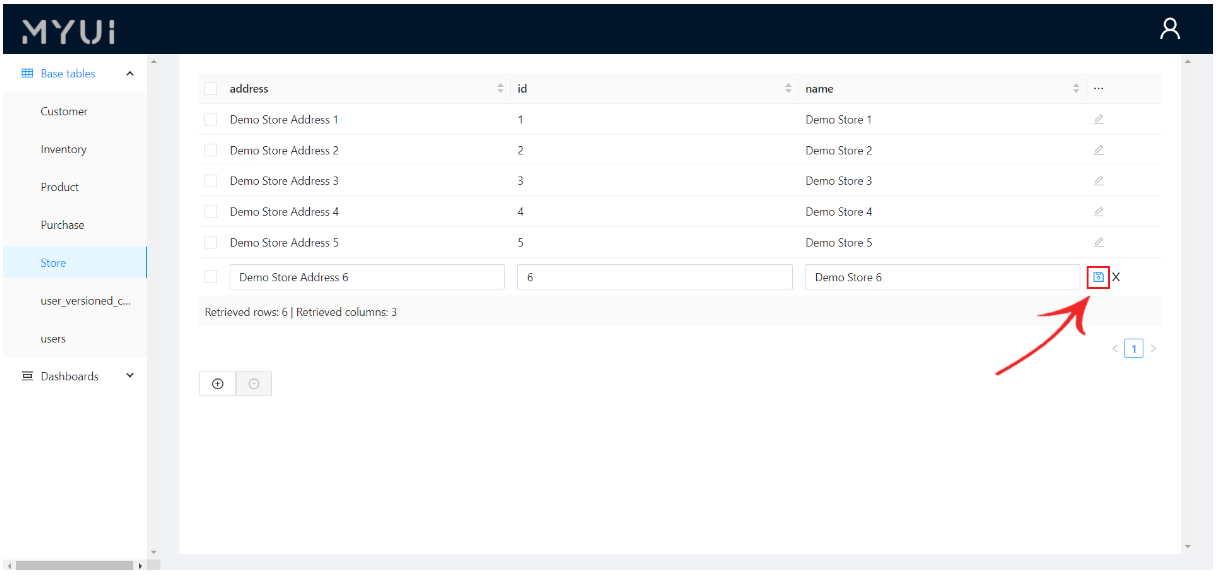

Figure 3.28: The save-row button

Expected outcome: After clicking on the save button indicated by the file icon, the changes made to the data inside the row are automatically shown in the grid view (Figure 3.29). The text fields for editing disappear.

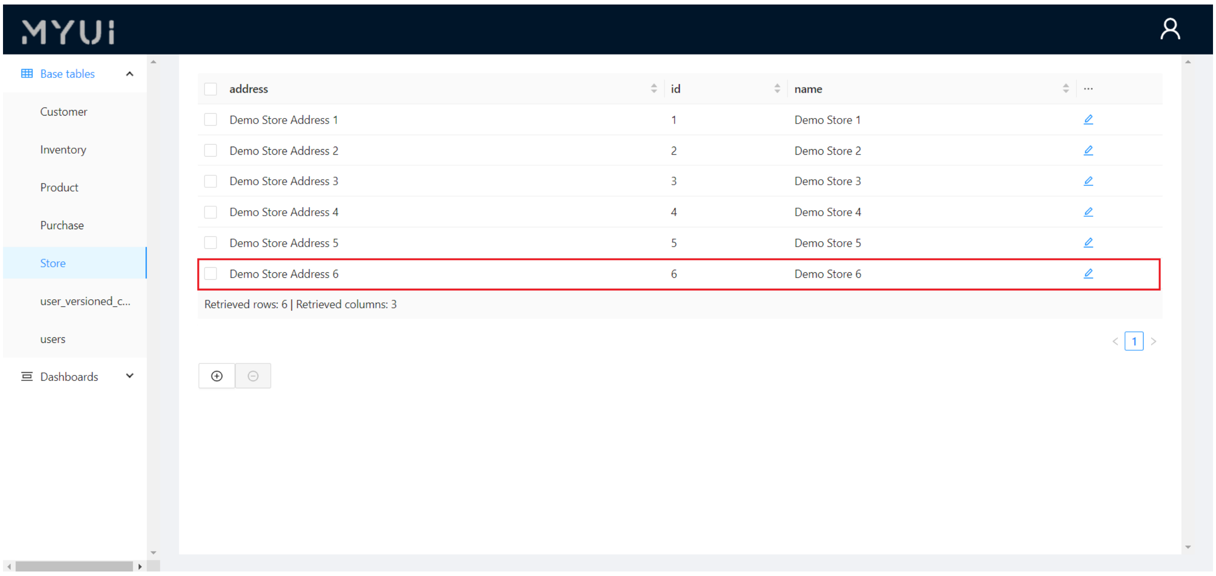

Figure 3.29: The row is saved and displayed in the table with the made changes

To cancel the editing click on the discard-changes button (Figure 3.30). The user will be presented with a warning prompt regarding the unsaved changes (Figure 3.31). Clicking on the *OK* button discards the unsaved changes. Clicking on the *Cancel* button aborts the canceling process. The text fields for each attribute are still displayed.

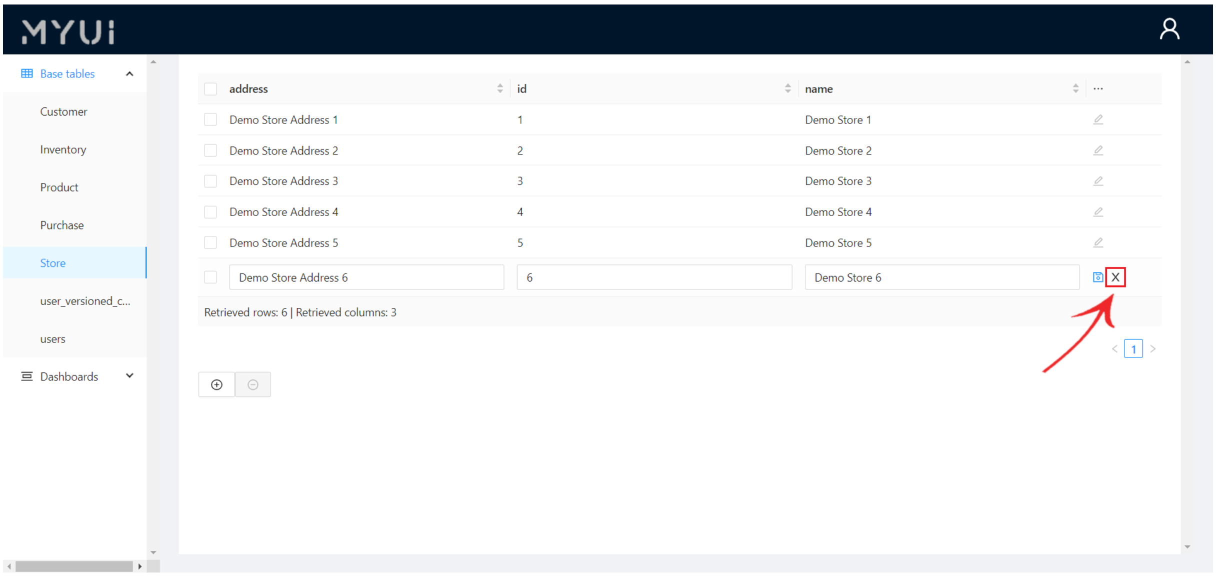

Figure 3.30: The discard-changes button to cancel the editing

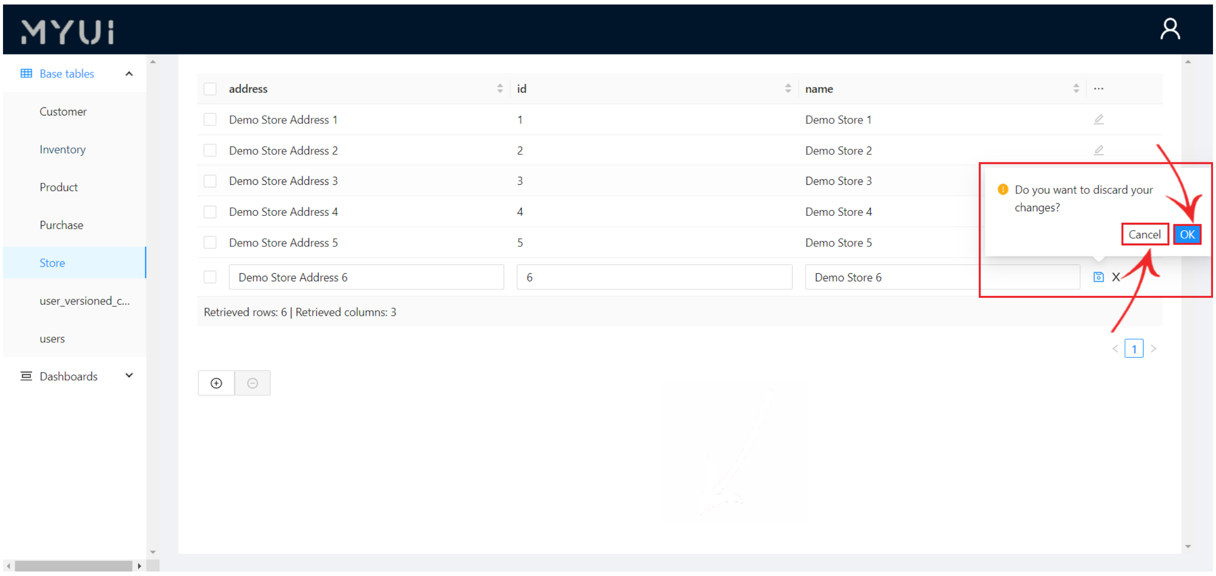

Figure 3.31: The warning prompt and the *Cancel* and *OK* buttons

Likely errors:

If the user is not allowed to edit a particular column, the user will be shown an error below that column (A.2).
The row's text fields are required fields. The user will be presented with error messages for each field that is not filled in (A.2).
Column entries might violate unique constraints or foreign key violations, resulting in a Hasura error message (A.5).

3.14 | Ordering rows of a column ascendingly

Functional description:

This tutorial describes how a user can sort the rows of a specific column ascendingly.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The user has a grid view loaded in their working area.
- The grid view contains data.
- There is no ordering applied to the rows of the desired column.

Procedure:

1. Click on the header of the desired column to be ordered on (Figure 3.32).

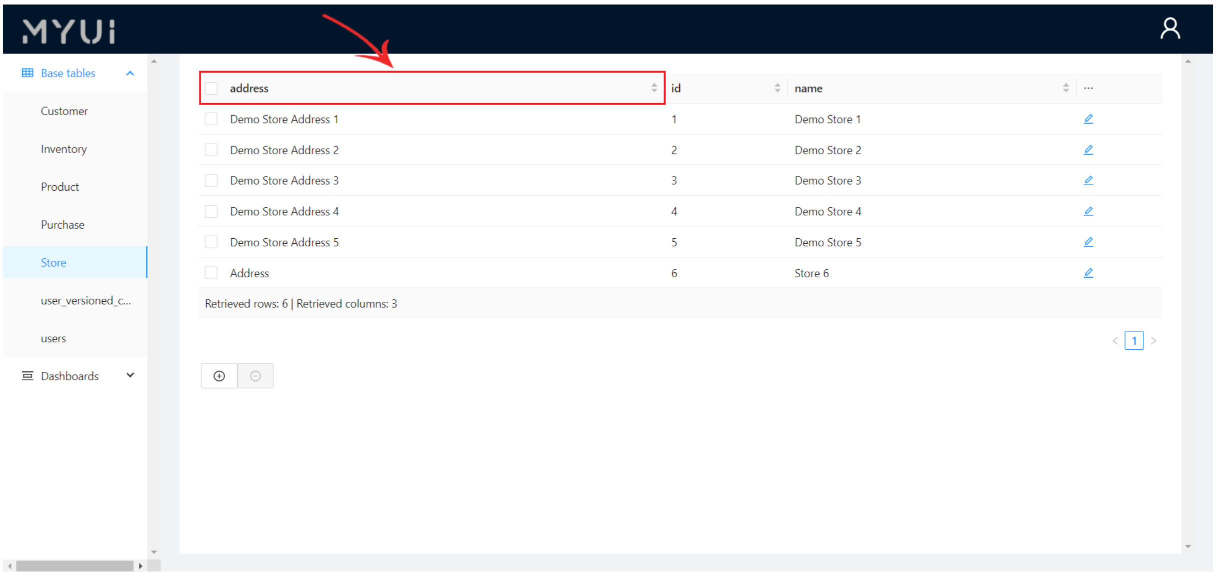

Figure 3.32: The header of the column

Expected outcome: After clicking on the header, the rows are sorted in ascending order (Figure 3.33).

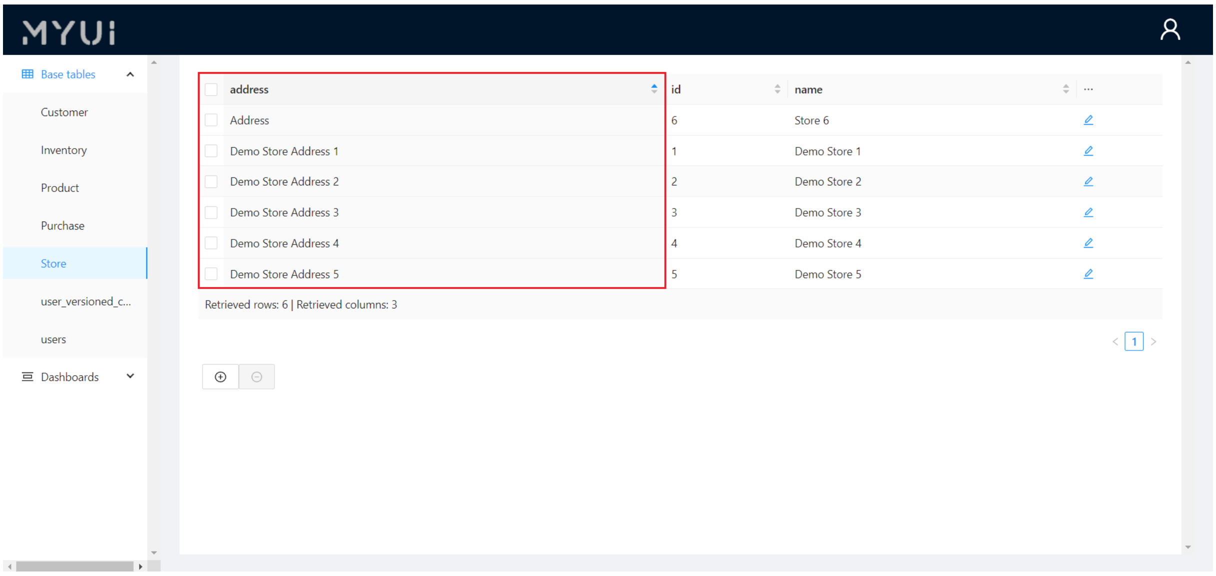

Figure 3.33: The rows of *address* column are ordered ascendingly

Likely errors:

None.

3.15 | Ordering rows of a column descendingly

Functional description:

This tutorial describes how a user can sort the rows of a specific column in descending order.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The user has a base table loaded in their working area.
- The rows of the desired column are sorted in ascending order (Tutorial 3.14 on page 35).

Procedure:

1. Click on the header of the desired column to be ordered on (Figure 3.32 on page 35).

Expected outcome: After clicking on the header, the rows are sorted in descending order (Figure 3.34).

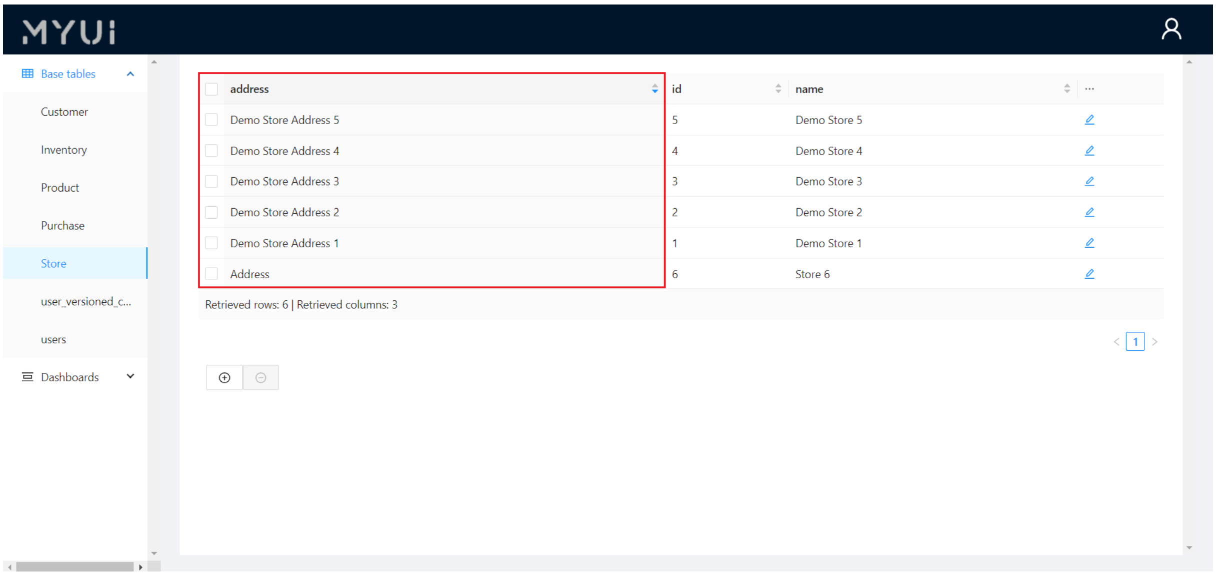

Figure 3.34: The rows of the *address* column are ordered descendingly

To reset the ordering, click on the column header.

Likely errors:

None.

3.16 | Changing row ordering to default

Functional description:

This tutorial describes how a user can change the ordering of rows back to default.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The user has a base table loaded in their working area.
- The rows of the desired column are sorted in descending order (Tutorial 3.15 on page 37).

Procedure:

1. Click on the header of the desired column to be ordered on (Figure 3.32 on page 35).

Expected outcome: After clicking on the header, the rows are sorted in default order.

Likely errors:

None.

3.17 | Change page of grid view

Functional description:

This tutorial describes how a user can move between the pages of a grid view.

Cautions and warnings:

If there is no previous or succeeding page, the respective button is disabled and the user cannot click on it.

Preconditions:

- The user is currently logged into MyUI.
- The user has a grid view loaded in their working area.
- The grid view contains more than one page.

Procedure A:

1. Click on the page number wished to view at the bottom right corner of the grid view (Figure 3.35).

Procedure B:

1. Click on the angle bracket (< and >) button at the bottom right corner of the grid view (Figure 3.36).

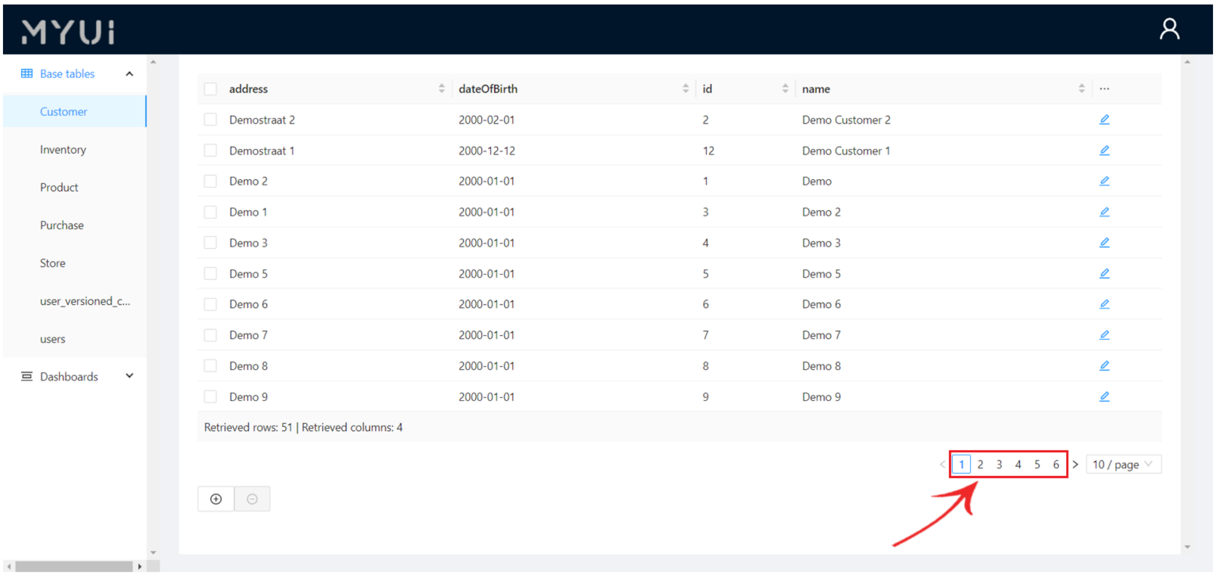

Figure 3.35: The page numbers of the grid view

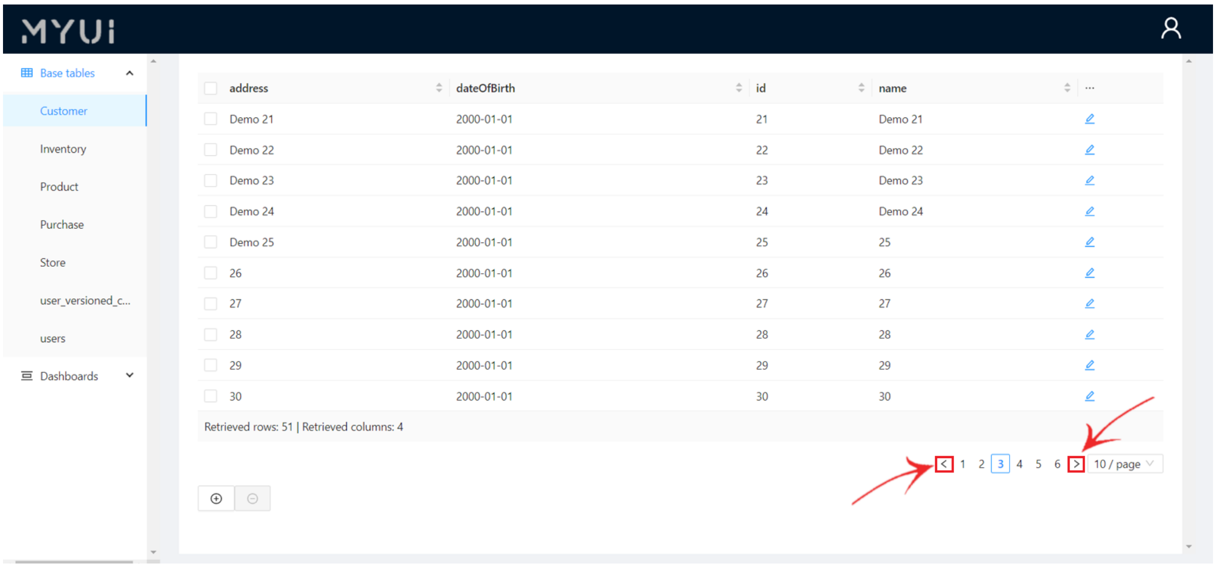

Figure 3.36: The previous and next buttons

Expected outcome: After clicking on the page number as Procedure A (on page 39) describes, the selected page of the grid view will be displayed. Using this method, the user can jump between any page. Using Procedure B (on page 39), the user can only change between the previous or next page of the current page displayed. After clicking on the left angle bracket, <, the previous page is displayed. After clicking on the right angle bracket, >, the succeeding page is displayed.

Likely errors:

None.

3.18 | Change the number of displayed rows per page

Functional description:

This tutorial describes how a user can change the number of rows of a grid view displayed per page.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The user has a grid view loaded in their working area.
- The grid view contains more than 10 rows.

Procedure:

1. Click on the displayed-rows dropdown menu at the bottom right corner of the grid view (Figure 3.37).
2. The user will be presented with the number of rows per page options.
3. Click on the desired number of rows to be displayed (Figure 3.38).

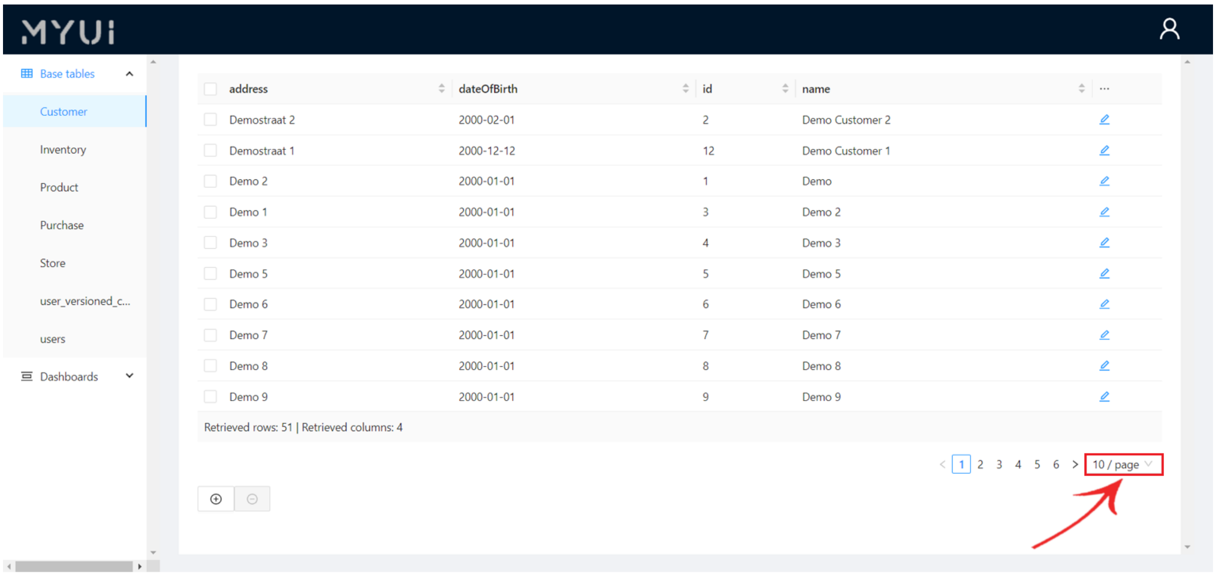

Figure 3.37: The displayed-row dropdown menu

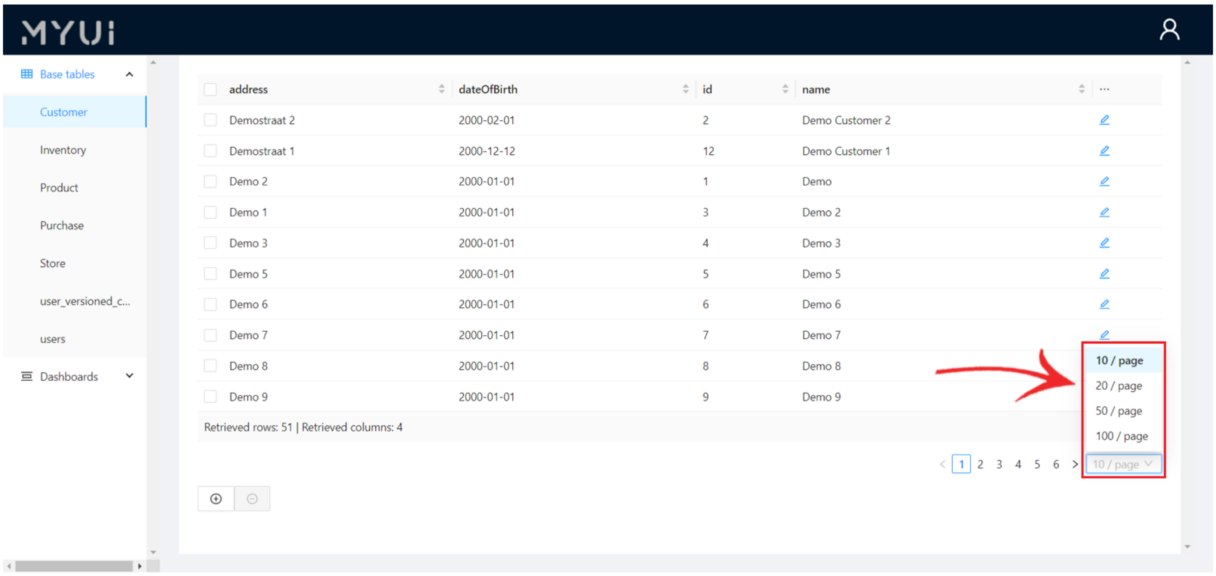

Figure 3.38: The number of rows per page options

Expected outcome: After clicking on the desired number of rows, the page refreshes and the grid view will be displayed with the selected number of rows per page (Figure 3.39).

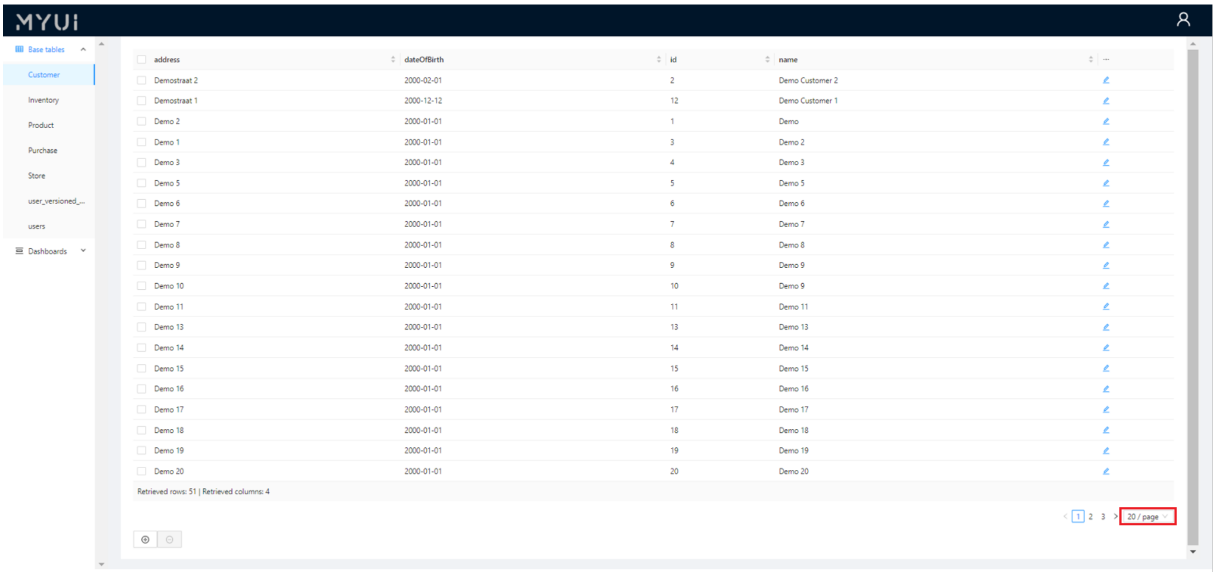

Figure 3.39: The grid view displayed with 20 rows per page

Likely errors:

None.

3.19 | Adding a dashboard

Functional description:

This tutorial describes how a user can add a dashboard to the *Dashboards* menu.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The *Dashboards* dropdown menu is folded.

Procedure:

1. Click on the dropdown menu labeled as *Dashboards* (Figure 3.40).
2. The dropdown menu unfolds.
3. Click on the *Add* button (Figure 3.41).
4. The user will be presented with the *Add a new dashboard* prompt (Figure 3.42).
5. Enter the <name> of the new dashboard in the *Name* field.
6. Click the *OK* button (Figure 3.43).

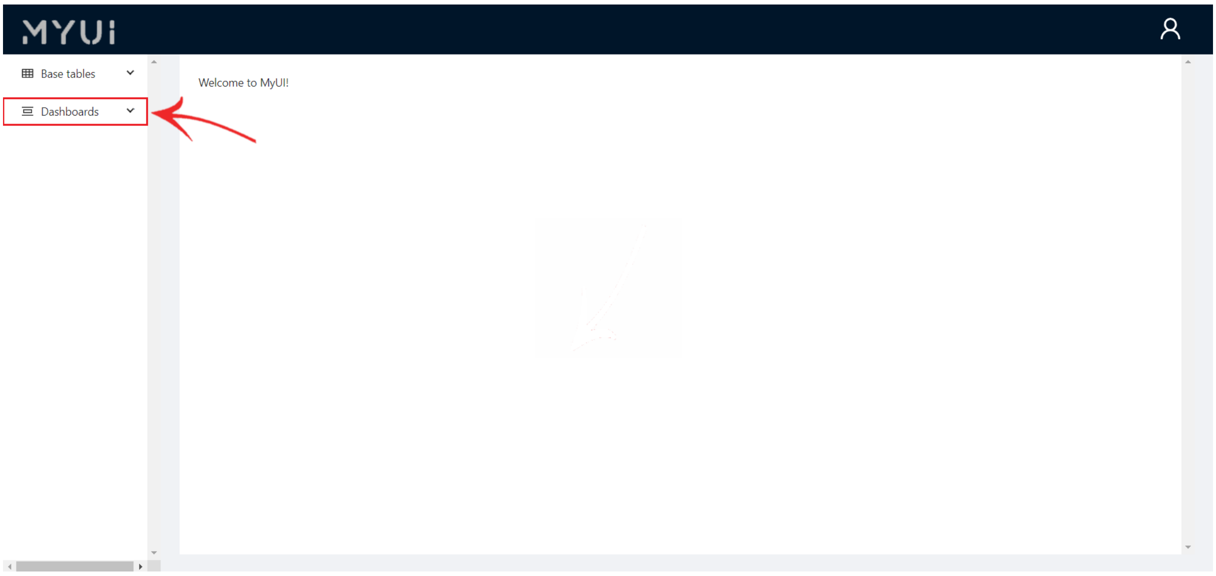

Figure 3.40: The *Dashboards* dropdown menu

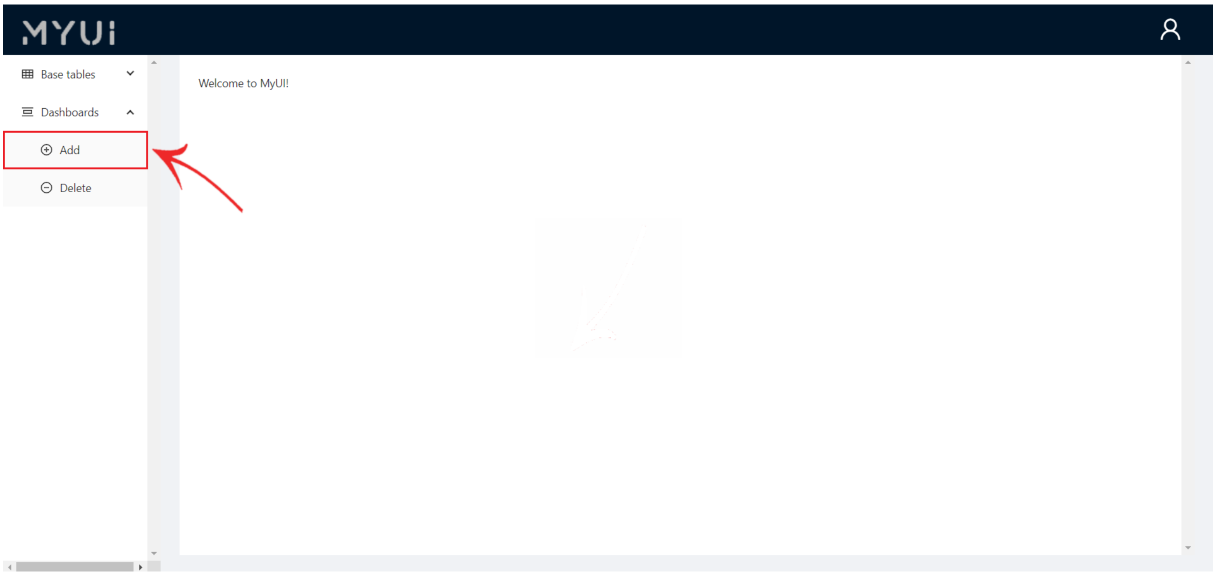

Figure 3.41: The *Add* button

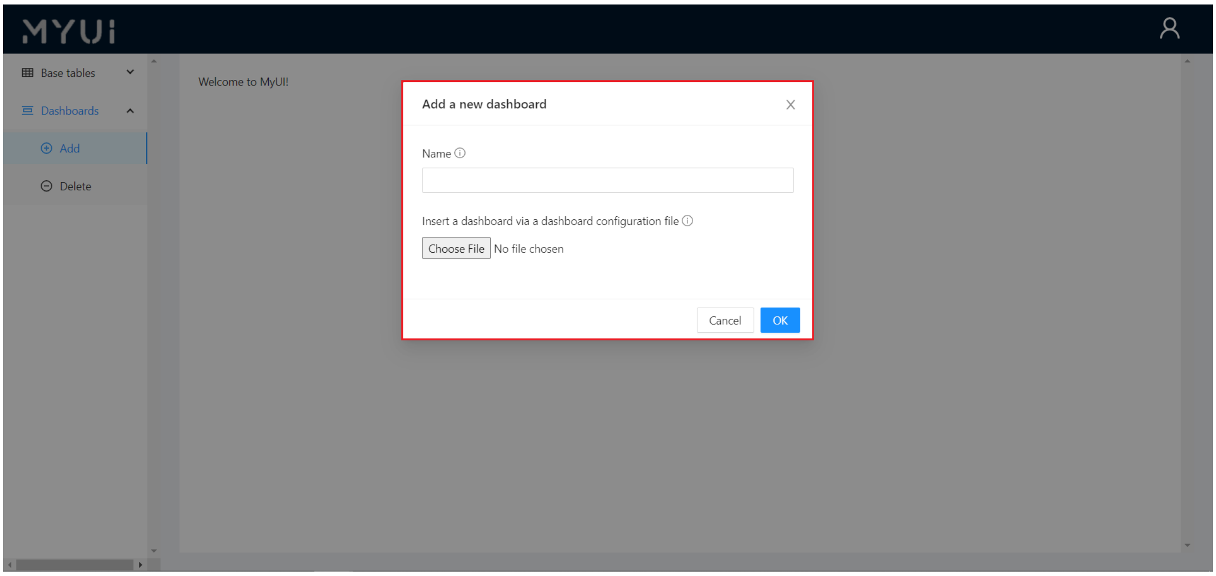

Figure 3.42: The *Add a new dashboard* prompt

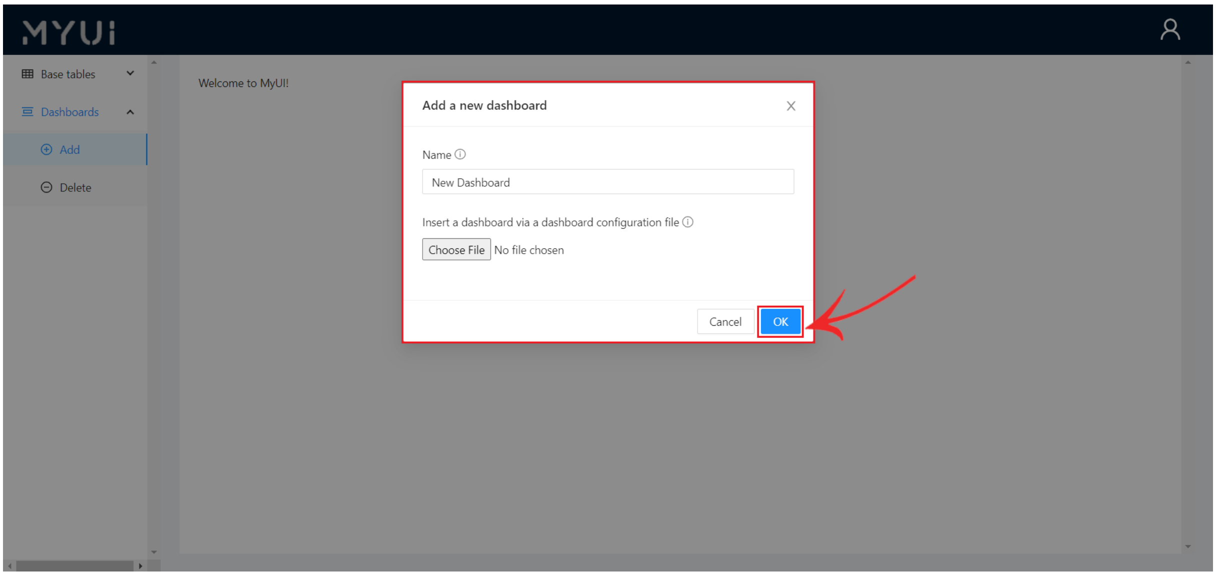

Figure 3.43: The <name> filled in and the *OK* button

Expected outcome: After clicking the *OK* button, the *Add a new dashboard* prompt disappears and a notification is shown stating that the action was successful. The newly added dashboard will appear in the *Dashboards* menu and it will be loaded in the user's working area (Figure 3.44).

Figure 3.44: The newly added dashboard and the notification

To cancel the addition, click on the *Cancel* button or click on the *X* button in the prompt header (Figure 3.45). The *Add a new dashboard* prompt will disappear.

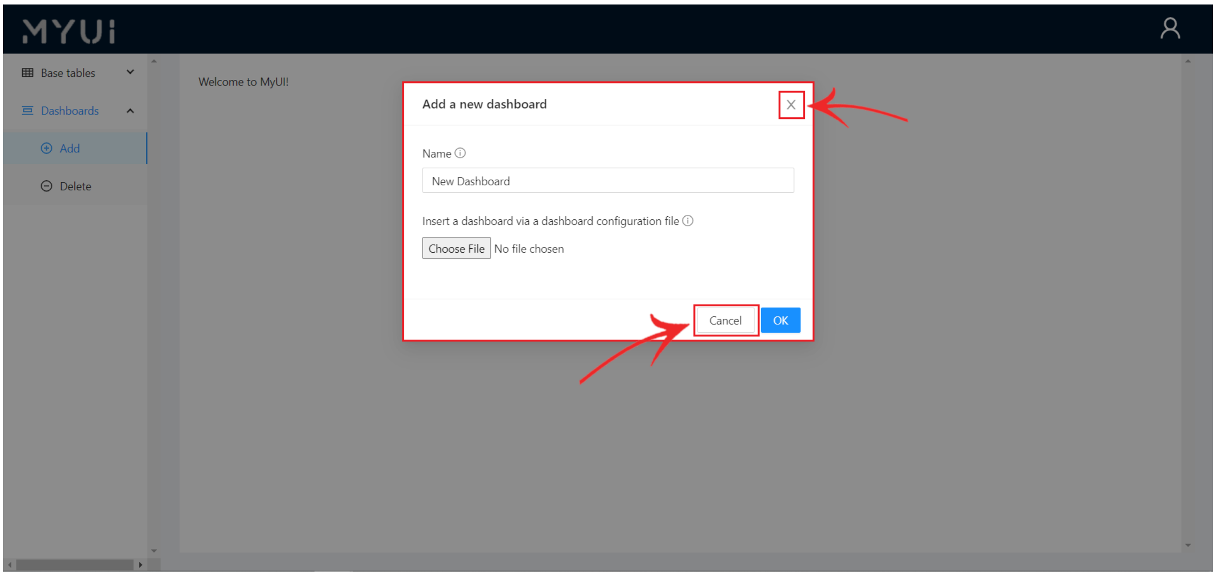

Figure 3.45: The *Cancel* and *X* buttons

Likely errors:

Filling in an invalid dashboard name in the *Name* field and clicking the *OK* button will cause error messages to show (A.3).

3.20 | Deleting a dashboard

Functional description:

This tutorial describes how a user can delete a dashboard from the *Dashboards* menu.

Cautions and warnings:

Once the dashboard is deleted, the action cannot be undone.

Preconditions:

- The user is currently logged into MyUI.
- There exists at least one dashboard in the *Dashboards* menu.
- The *Dashboards* dropdown menu is folded.

Procedure:

1. Click on the dropdown menu labeled as *Dashboards* (Figure 3.40 on page 43).
2. The dropdown menu unfolds.
3. Click on the *Delete* button (Figure 3.46).
4. The user will be presented with the *Remove a dashboard* prompt (Figure 3.47).
5. Enter the <name> of the dashboard wished to be deleted in the text field.
6. Click the *OK* button (Figur 3.48).
7. The user will be presented with a prompt asking for deletion confirmation.
8. Click on the *Yes* button (Figure 3.49).

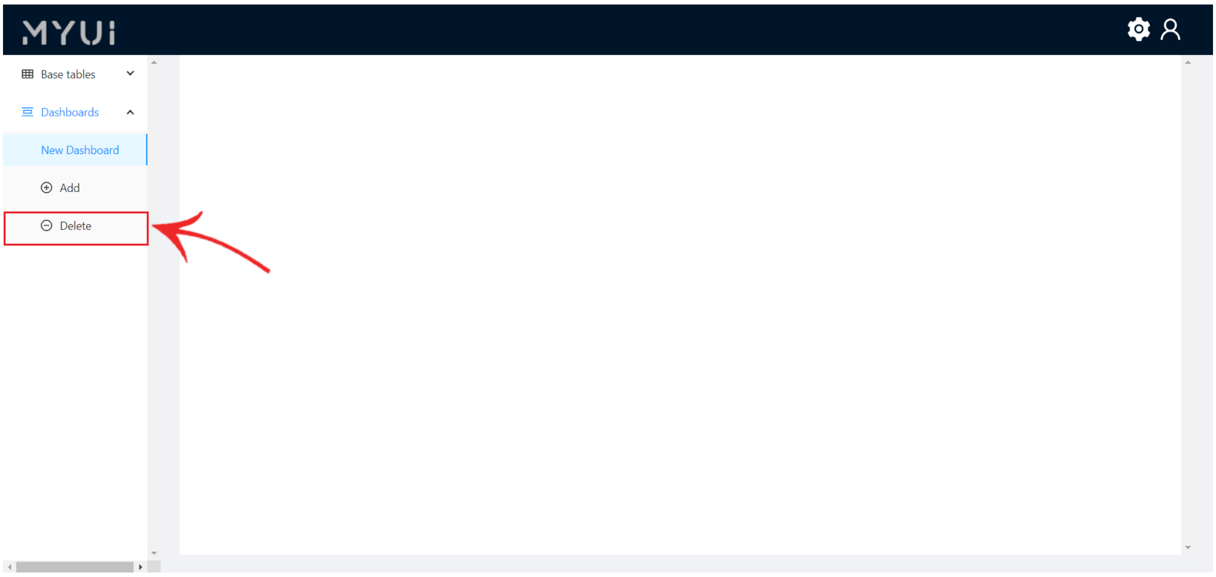

Figure 3.46: The *Delete* button

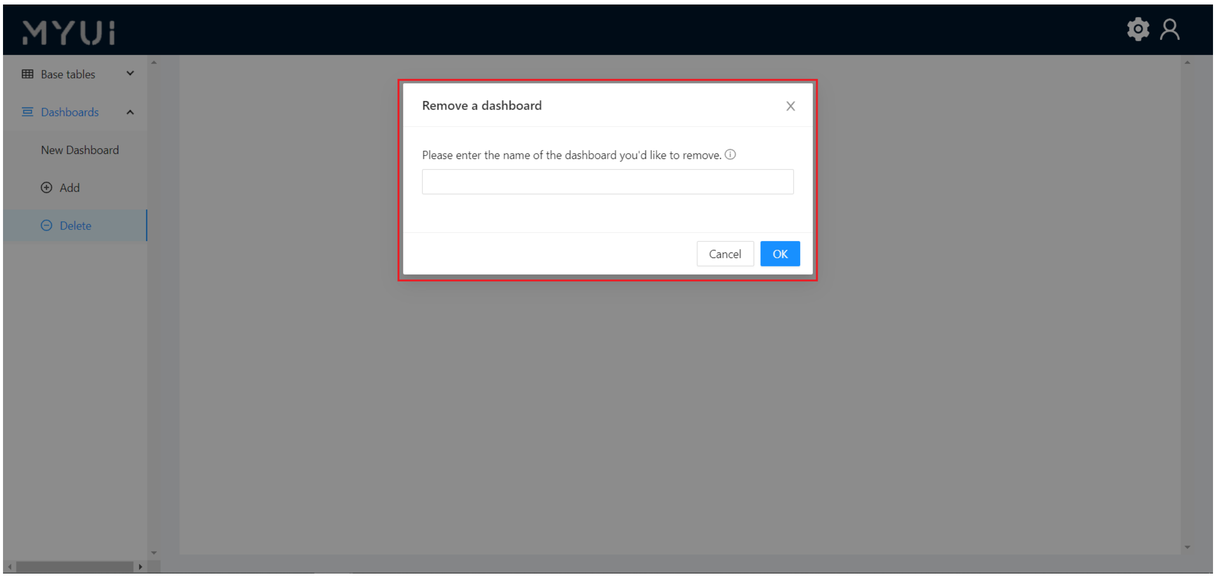

Figure 3.47: The *Remove a dashboard* prompt

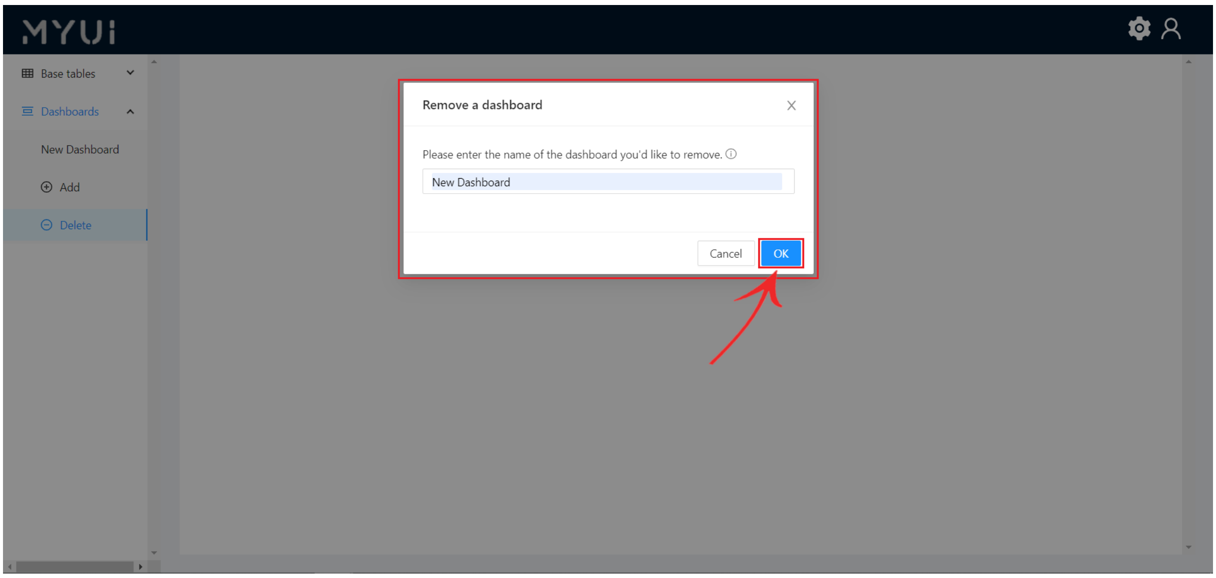

Figure 3.48: The *OK* button

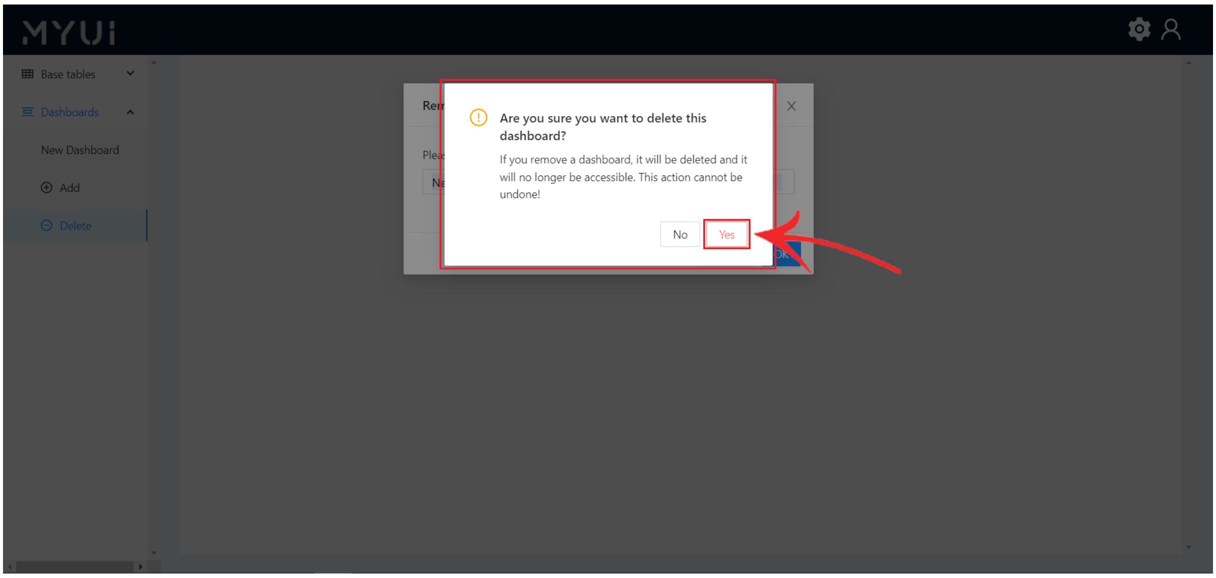

Figure 3.49: The *Yes* button

Expected outcome: After clicking on the *Yes* button, the *Remove a dashboard* prompt disappears and the dashboard with the inputted name is deleted from the *Dashboards* menu (Figure 3.50). This action cannot be undone.
If the user deletes a dashboard that is currently loaded in their working area, MyUI will show the homepage.

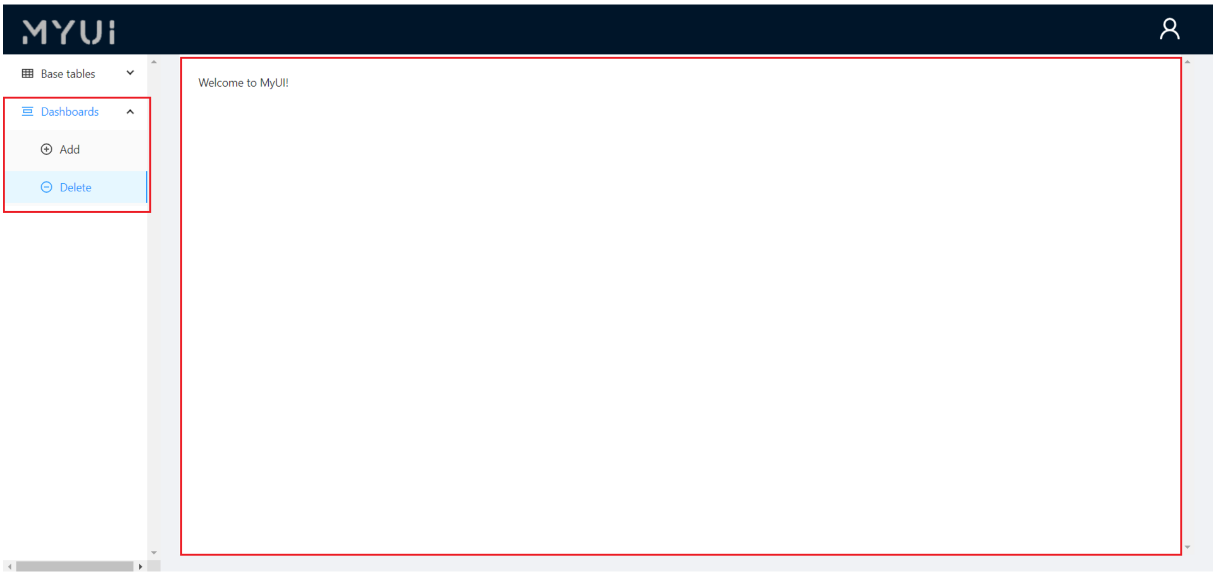

Figure 3.50: The dashboard *New Dashboard* is deleted from the *Dashboards* menu

Clicking *No* on the confirmation prompt takes the user back to the *Remove a dashboard* prompt.
To cancel the deletion, click on the *Cancel* button or click on the *X* button in the prompt header (Figure 3.51). The *Remove a dashboard* prompt will disappear.

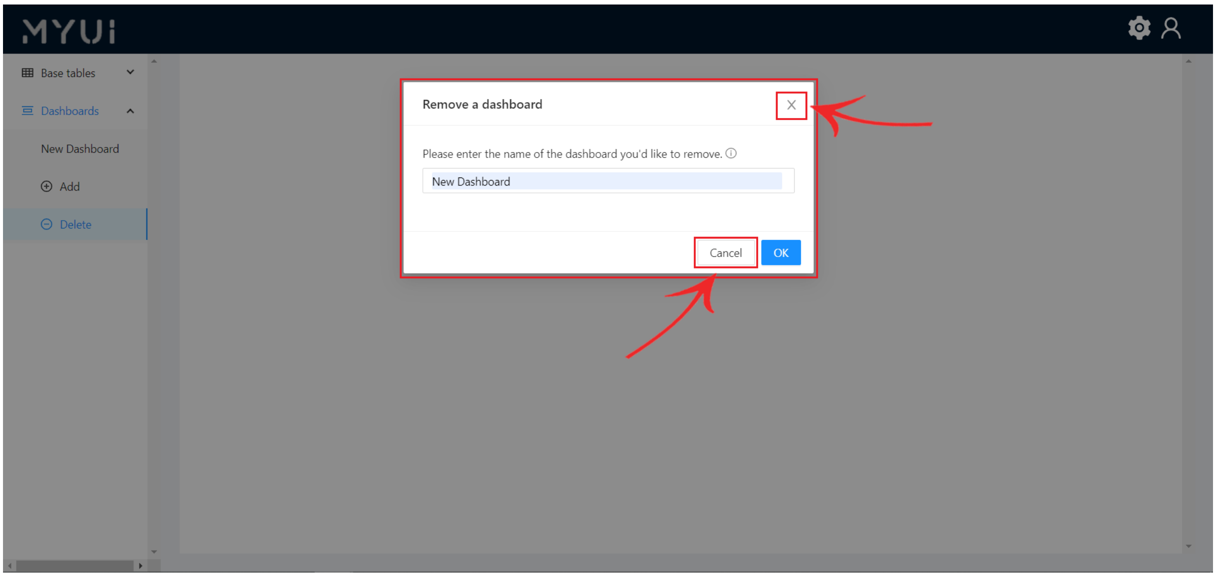

Figure 3.51: The *Cancel* and *X* buttons

Likely errors:

Filling in an incorrect name or clicking the *OK* button without filling anything in the text field will cause error messages to show (A.3).

3.21 | Loading a dashboard

Functional description:

This tutorial describes how a user can load a dashboard in MyUI.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- There is at least one dashboard in the *Dashboards* menu.
- The *Dashboards* dropdown menu is folded.

Procedure:

1. Click on the dropdown menu labeled as *Dashboards* (Figure 3.40).
2. The user will be presented with a list of the dashboards present in their respective user configuration file (Figure 3.52).
3. Click on the option in the dropdown menu that is labeled with the dashboard the user wishes to view.

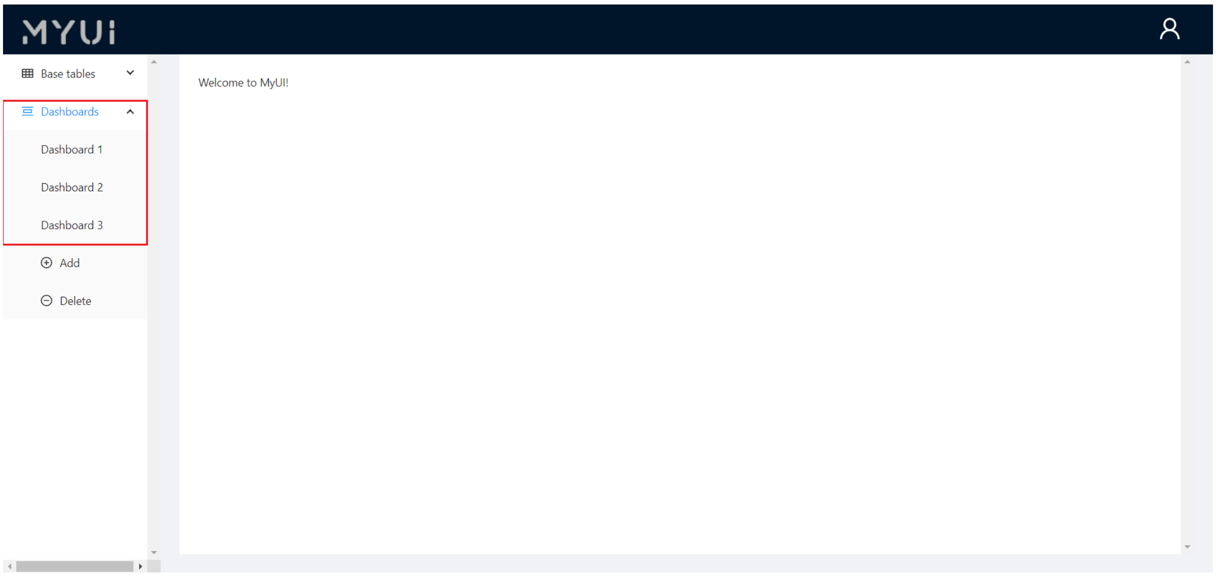

Figure 3.52: The list of *Dashboards*

Expected outcome: After clicking on the name of the wanted dashboard, the respective dashboard will appear in the user's working area with the saved preferences for this dashboard (Figure 3.53).

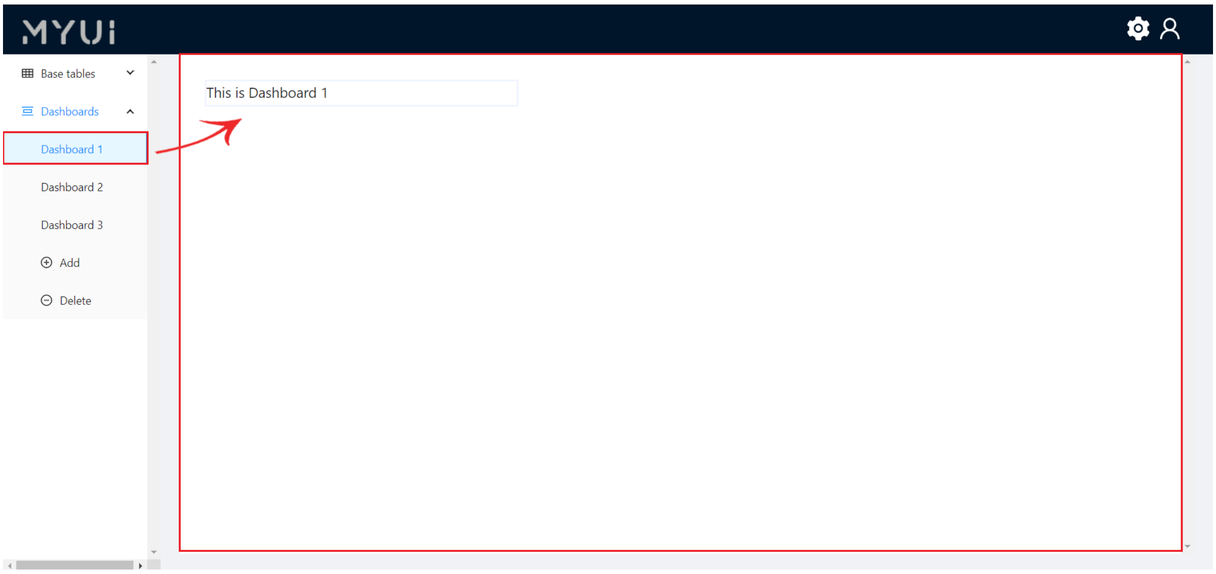

Figure 3.53: The dashboard *Dashboard 1* loaded in the user's working area

Likely errors:

None.

3.22 | Enter edit mode

Functional description:

This tutorial describes how a user can enter edit mode.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The user has a dashboard loaded in their working area.
- The user is currently not in edit mode.

Procedure:

1. Click on the gear icon in the header of the application. This is called the change-mode button (Figure 3.54).

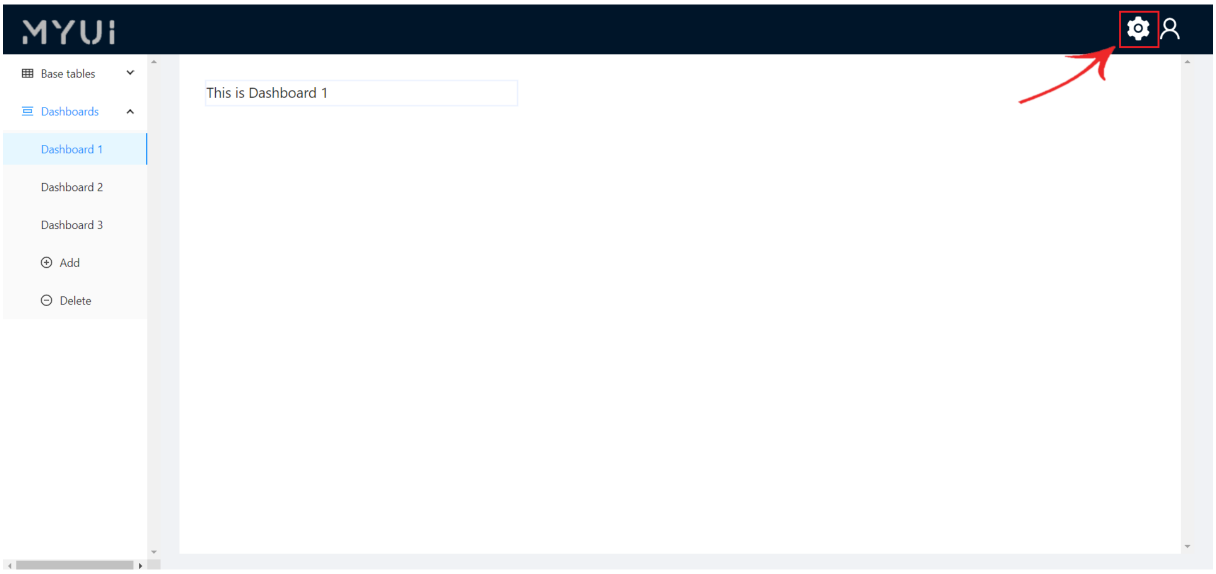

Figure 3.54: The change-mode button

Expected outcome: After clicking on the change-mode button, the user enters edit mode. This can be identified by the spinning gear icon and the edit mode sidebar (Figure 3.55).

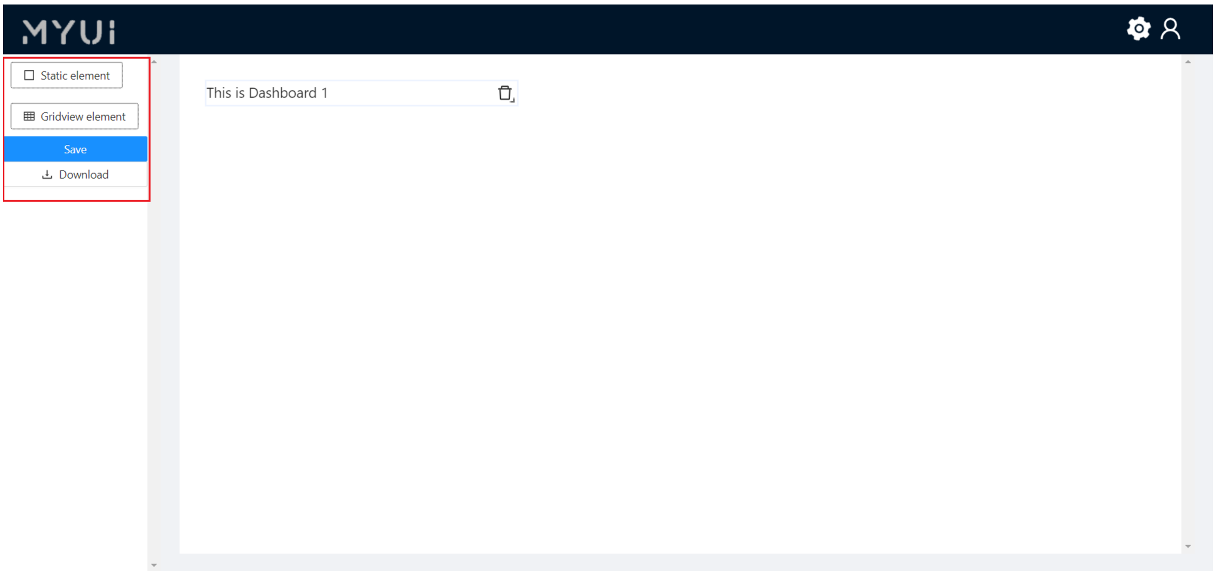

Figure 3.55: The edit mode sidebar

Likely errors:

None.

3.23 | Exit edit mode

Functional description:

This tutorial describes how a user can exit edit mode.

Cautions and warnings:

If there are unsaved changes made in the edit mode session, after clicking on the change mode button to exit edit mode, the user is presented with a warning prompt regarding the unsaved changes. Clicking on the *Cancel* button removes the prompt and the user remains in edit mode. Clicking on the *Discard changes* button deletes the unsaved changes. This action cannot be undone. Clicking on the *Save changes* button saves the changes (Figure 3.56). After saving, a notification is shown stating that the changes made have been saved.
After saving or discarding the changes, the warning prompt disappears and the user exits edit mode.

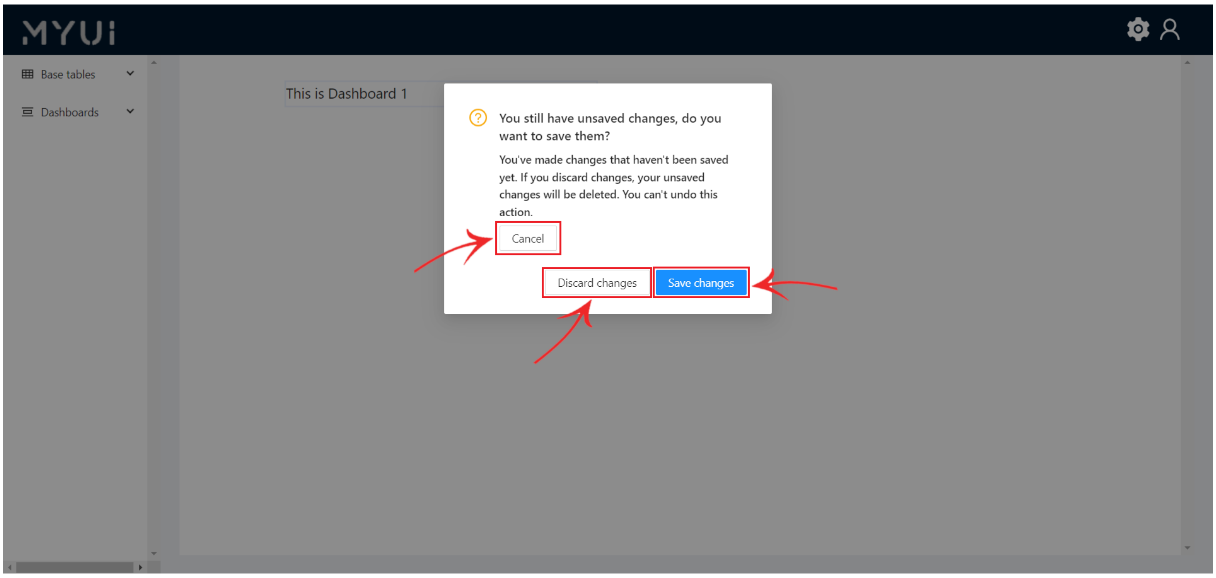

Figure 3.56: The warning prompt if there are unsaved changes

Preconditions:

- The user is currently logged into MyUI.
- The user has a dashboard loaded in their working area.
- The user is in edit mode.

Procedure:

1. Click on the change-mode button in the header of the application (Figure 3.57).

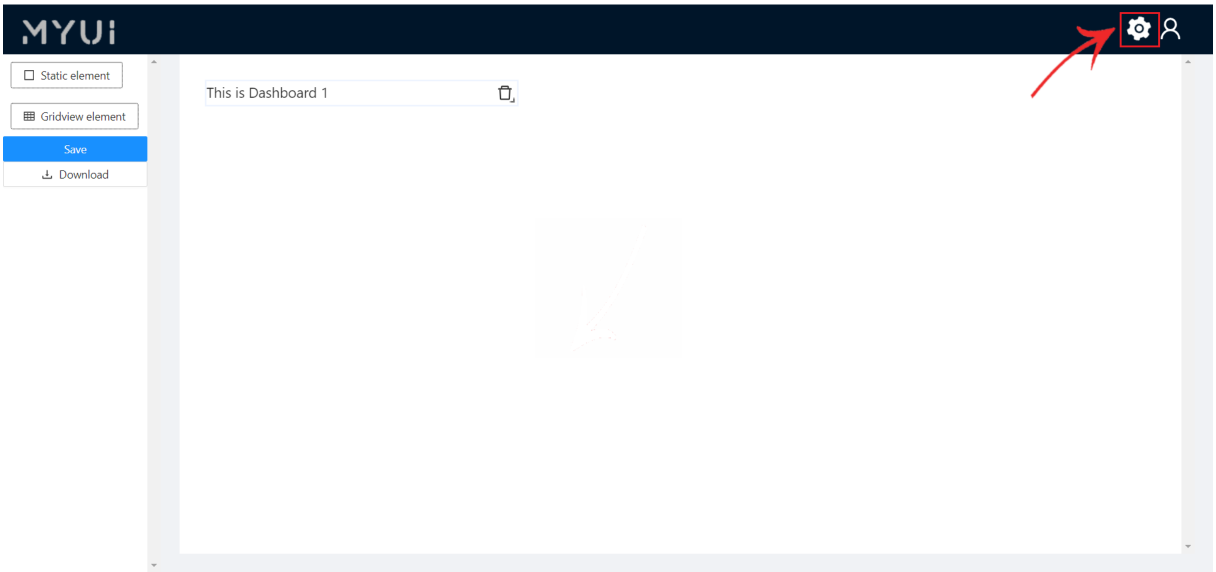

Figure 3.57: The change-mode button

Expected outcome: After clicking, the edit mode sidebar disappears and the gear icon stops spinning indicating that the user is not in edit mode anymore.

Likely errors:

None.

3.24 | Add static elements

Functional description:

This tutorial describes how a user can add static elements to their dashboard while in edit mode.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The user has a dashboard loaded in their working area.
- The user is in edit mode.

Procedure:

1. Drag and drop the *Static element* button to the working area (Figure 3.58).
2. The user will be presented with the edit menu (Figure 3.59).
3. Enter the desired text in the textbox of the element.
4. Click on the *OK* button (Figure 3.60).

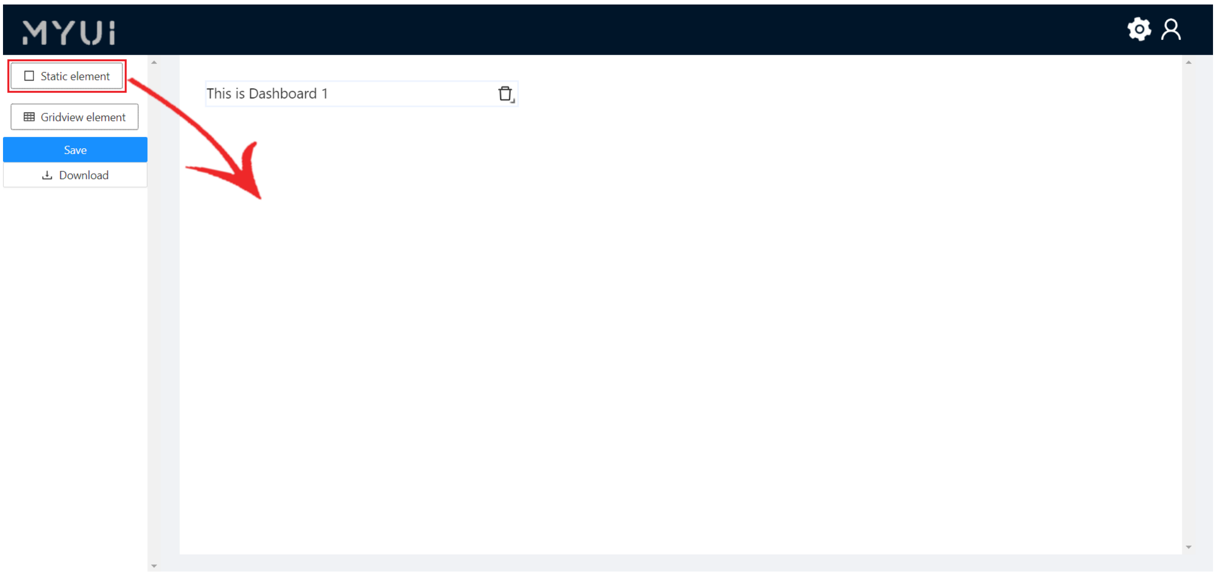

Figure 3.58: The *Static element* button

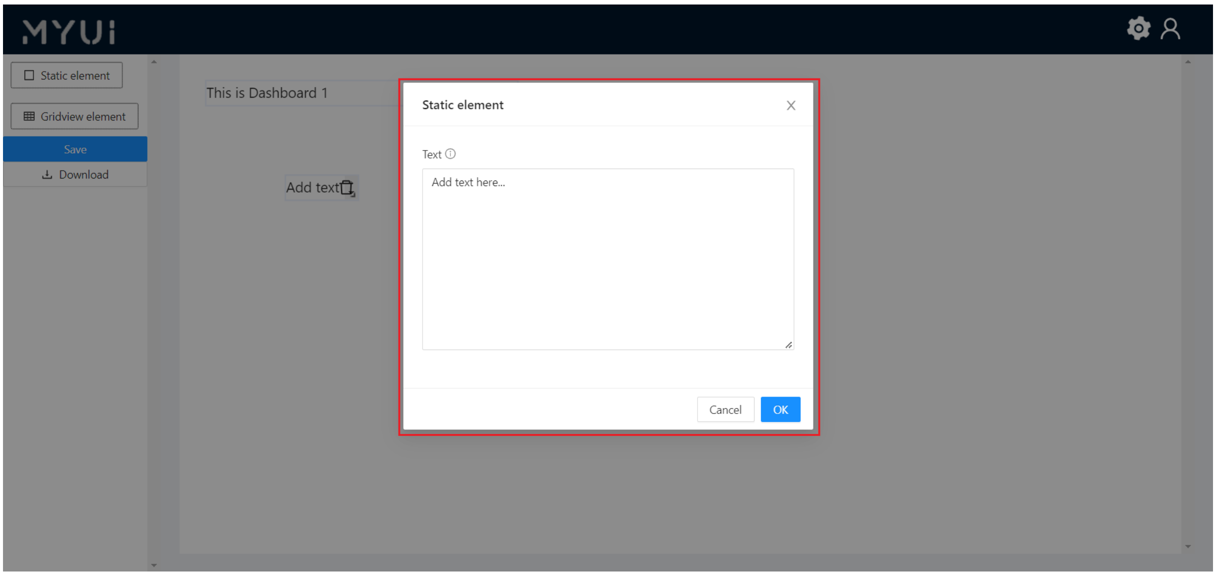

Figure 3.59: The edit menu for the static element

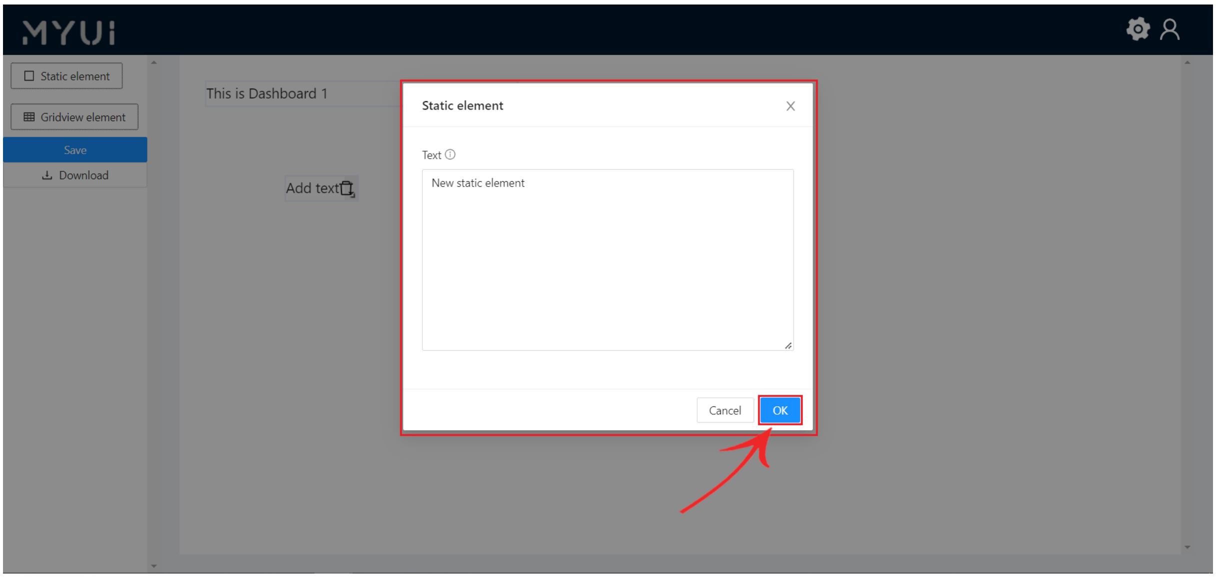

Figure 3.60: The *OK* button

Expected outcome: After clicking on the *OK* button, the static element gets added to the dashboard (Figure 3.61).
Clicking on the *Cancel* button in the edit menu makes the pop-up disappear. The static element is added to the dashboard but any changes made to the content of the textbox are discarded.

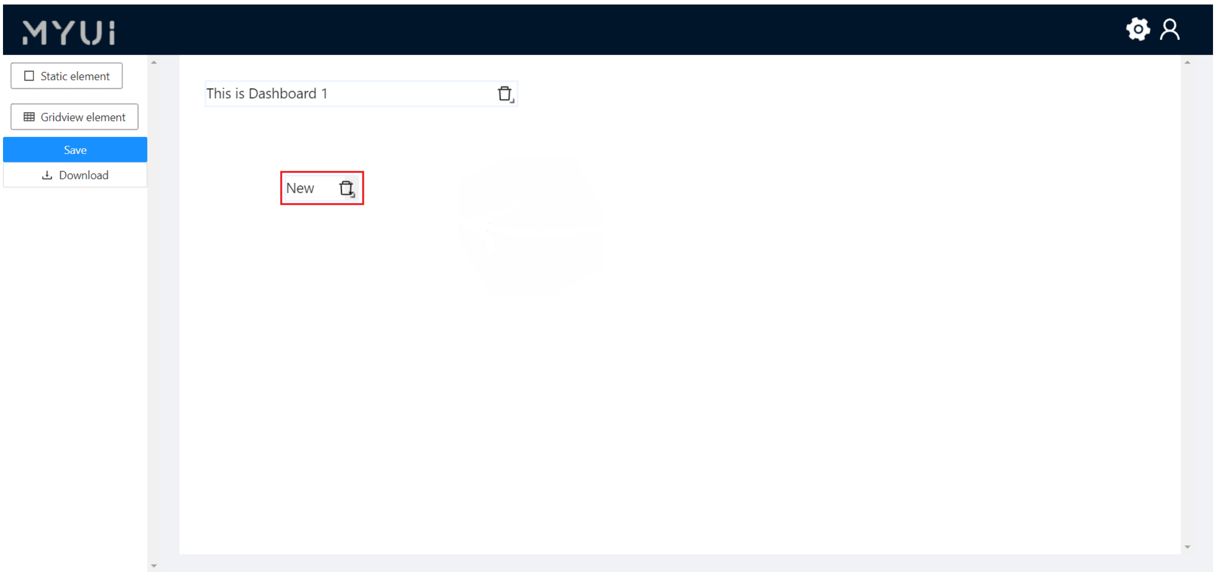

Figure 3.61: The newly added static element

Likely errors:

If the user adds (double) quotation marks (") or backslash (\) to the text field, MyUI will display an error message stating that this character is not allowed (A.4).

3.25 | Add media to a static element

Functional description:

This tutorial describes how a user can add a static element containing images, videos, audio clips, or GIFs in their dashboard while in edit mode.

Cautions and warnings:

The allowed extensions for media are .png, .jpeg, .jpg, .gif, .mp3, and .mp4. If the user adds media with a different extension than the mentioned ones, the media URL will be displayed as a plain URL.

Preconditions:

- The user is currently logged into MyUI.
- The user has a dashboard loaded in their working area.
- The user is in edit mode.

Procedure:

1. Drag and drop the *Static Element* button to the working area (Figure 3.58 on page 57).
2. The user will be presented with the edit menu (Figure 3.59 on page 58).
3. Add the URL of the media in the textbox of the element.
4. Click on the *OK* button (Figure 3.62).

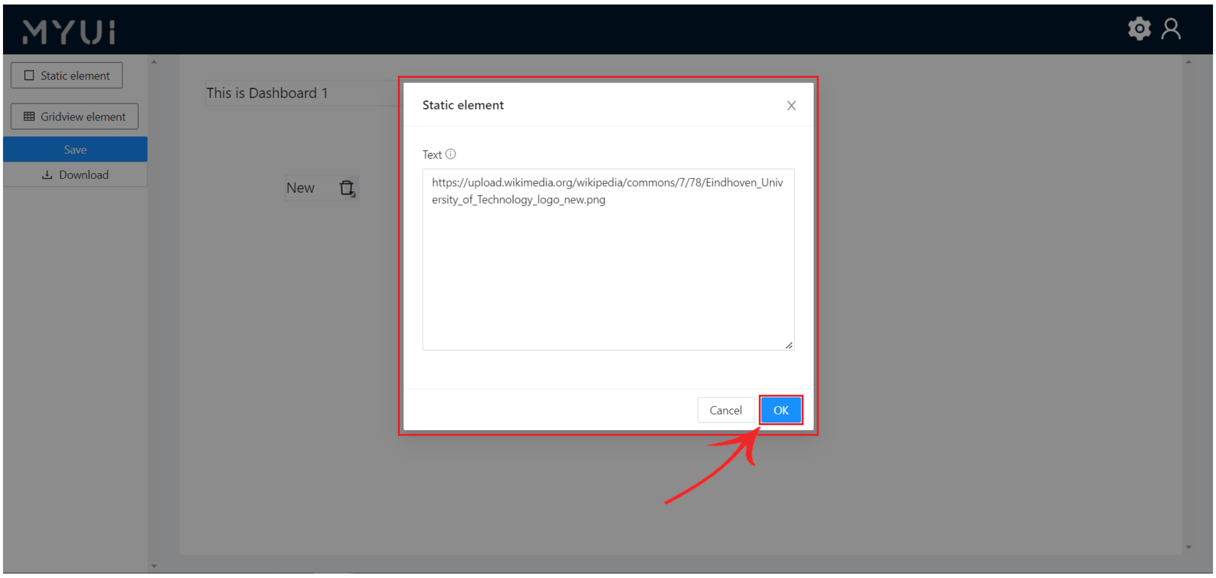

Figure 3.62: The *OK* button

Expected outcome: After clicking on the *OK* button, the static element gets added to the dashboard and the media is rendered according to the URL display setting (Figure 3.63).
Clicking on the *Cancel* button in the edit menu makes the pop-up disappear. The static element is added to the dashboard but any changes made to the content of the textbox are discarded.

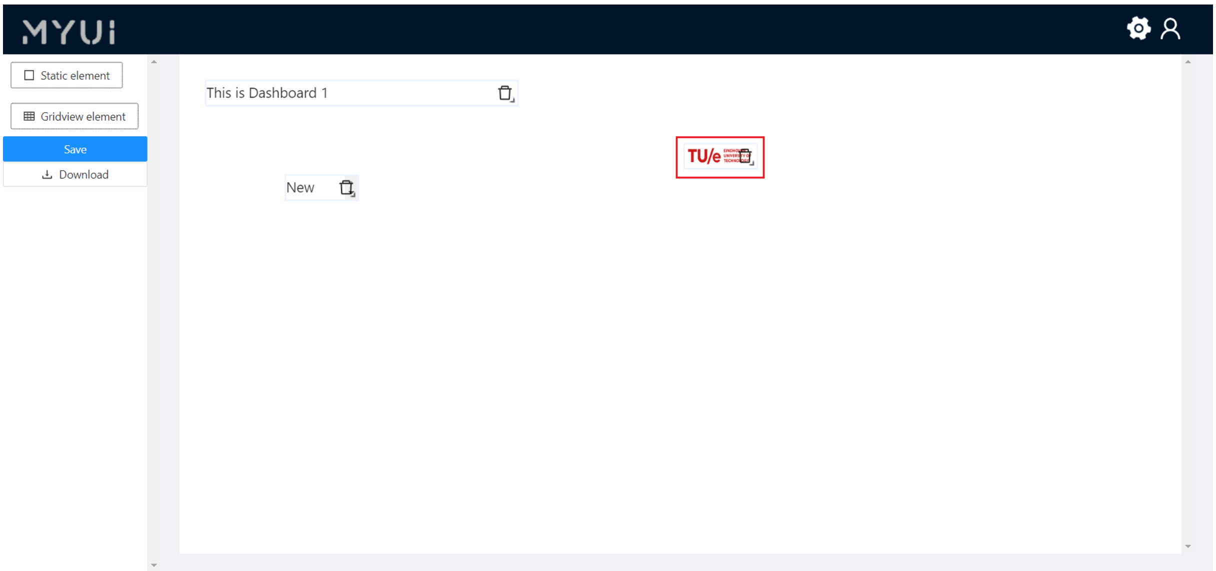

Figure 3.63: The newly added static element with media content

Likely errors:

If the user inserts a URL for a nonexistent media, the static element is displayed with a broken image icon (A.4).

3.26 | Resizing an application element

Functional description:

This tutorial describes how a user can resize a static or grid view element within their dashboard while in edit mode.

Cautions and warnings:

There cannot be two elements on top of each other.

Preconditions:

- The user is currently logged into MyUI.
- The user has a dashboard loaded in their working area.
- The user is in edit mode.
- The loaded dashboard contains at least one application element.

Procedure:

1. Click and hold the right bottom corner of the wanted application element (Figure 3.64).
2. Drag the corner of the element until it gets to the desired size which is indicated by the red outline (Figure 3.65).
3. Release the corner of the application element.

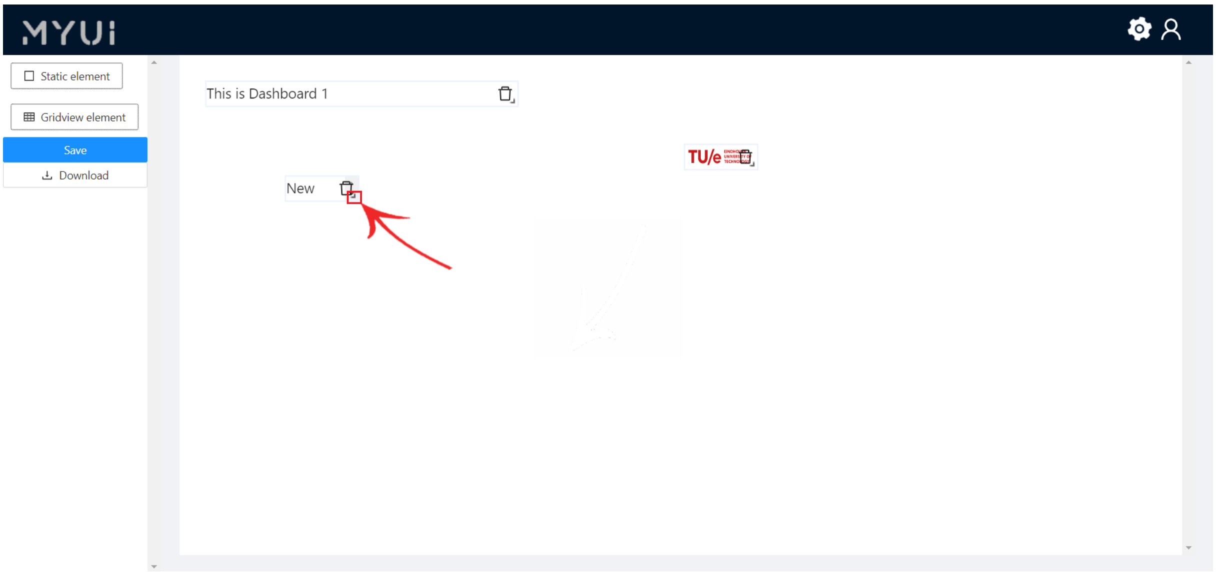

Figure 3.64: The bottom corner for resizing an element

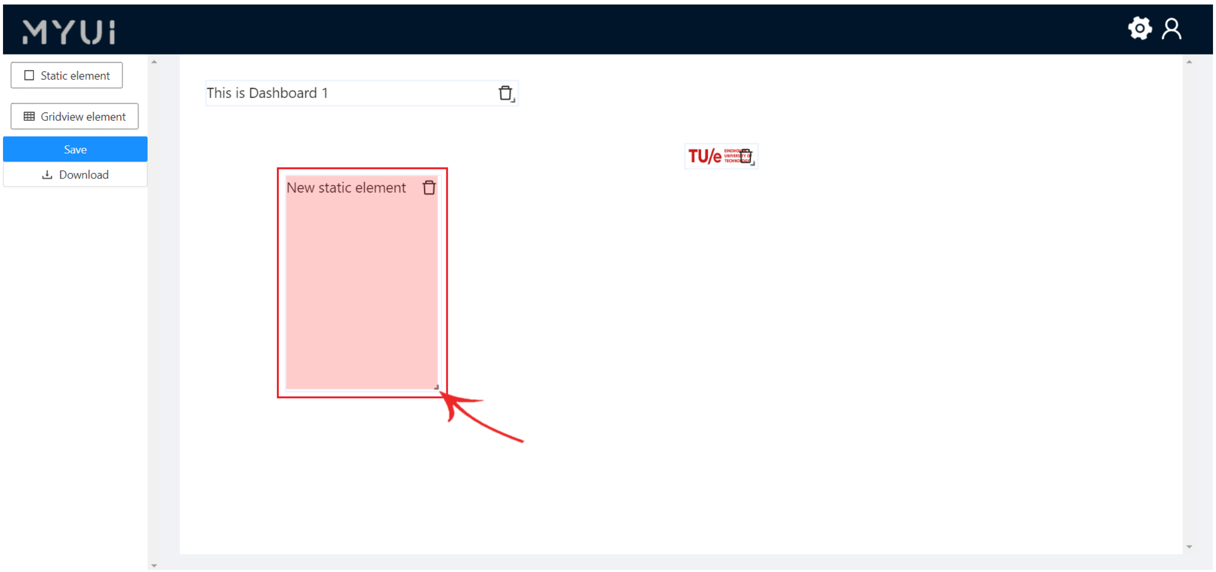

Figure 3.65: The outline of the size of the application element is shown in red

Expected outcome: After dropping the corner, the element will be displayed in the modified size (Figure 3.66).

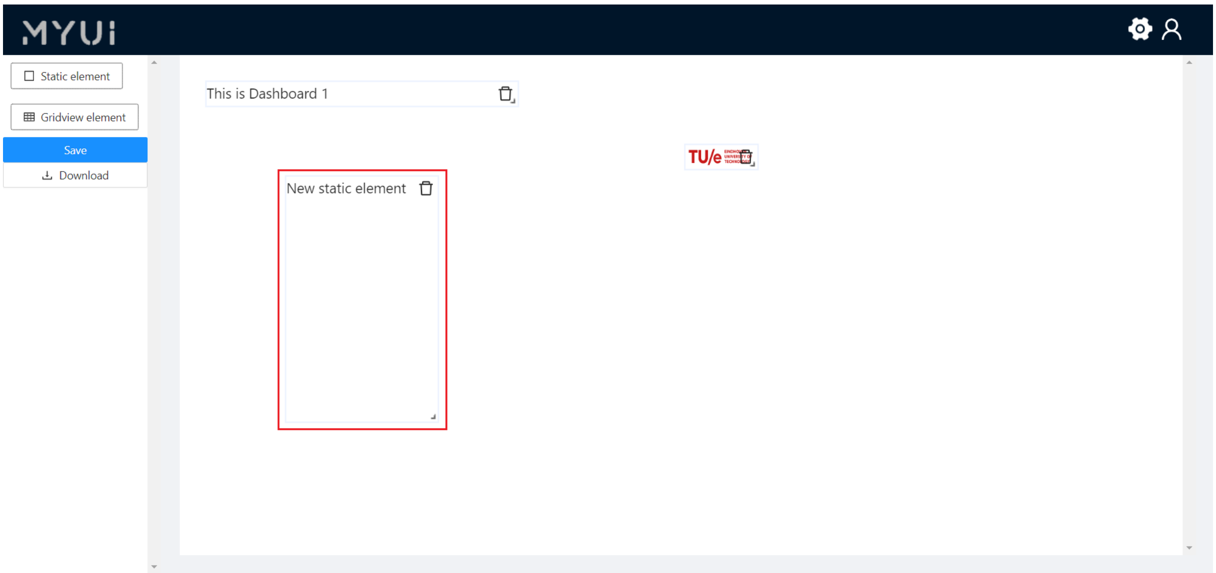

Figure 3.66: The resized application element

Likely errors:

None.

3.27 | Add a grid view element

Functional description:

This tutorial describes how a user can add a grid view element to their dashboard.

Cautions and warnings:

The query must have valid semantics before the user can submit it.
If the grid view element displays *"no data"*, the user has either insufficient permissions to view the data or there exists no data for the executed query.

Preconditions:

- The user is currently logged into MyUI.
- The user has a dashboard loaded in their working area.
- The user is in edit mode.

Procedure:

1. Drag and drop the button *Gridview element* to the working area (Figure 3.67).
2. The user will be presented with the edit menu (Figure 3.68).
3. Enter a syntactically correct GraphQL query in the text field.
4. Click on the *OK* button (Figure 3.69).

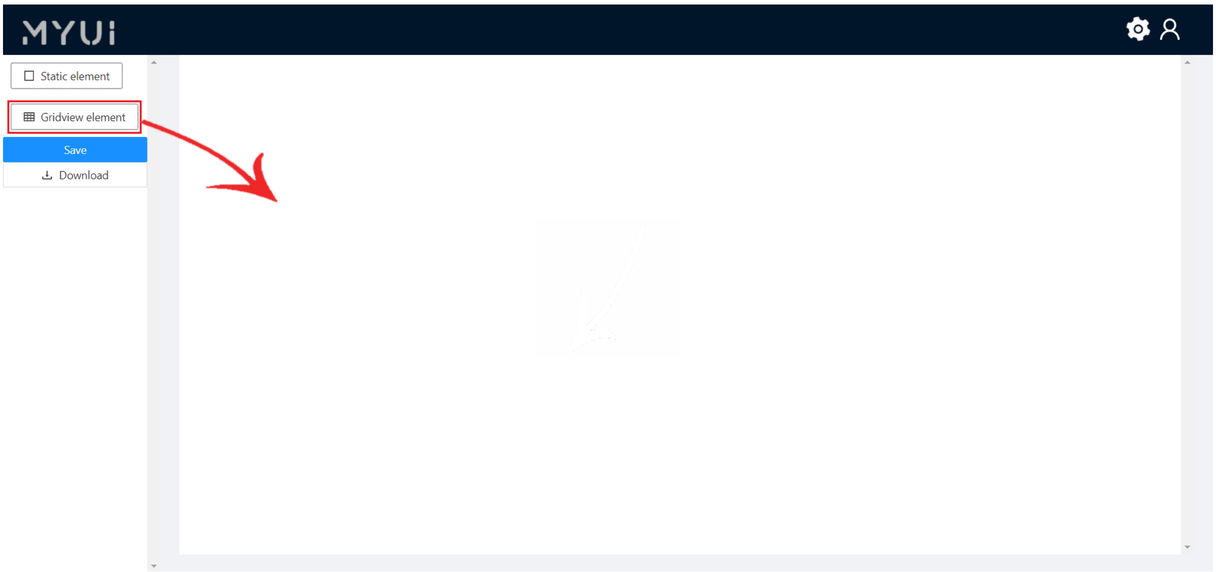

Figure 3.67: The *Gridview element* button

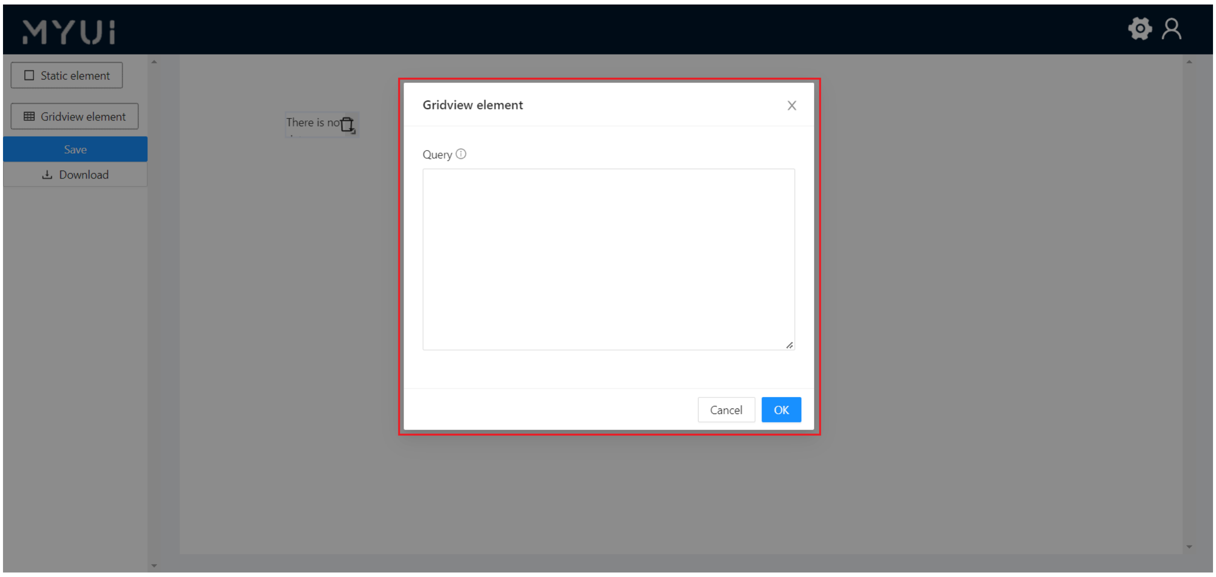

Figure 3.68: The edit menu for the grid view element

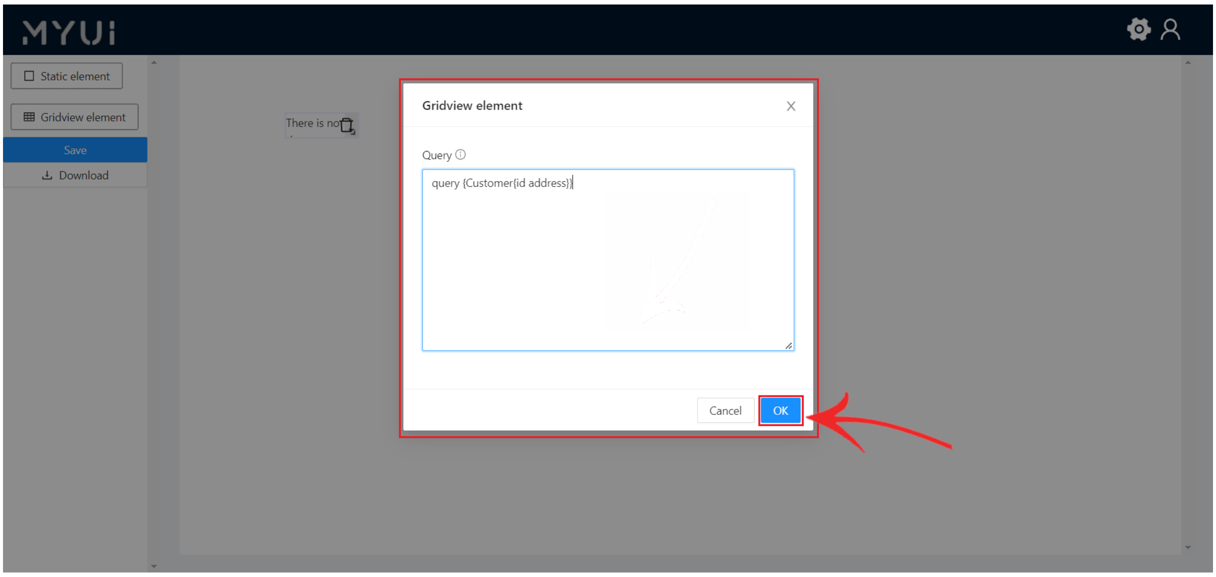

Figure 3.69: The GraphQL query is written in the text field and the *OK* button is clicked on

Expected outcome: After clicking on the *OK* button, the user will see the result of the query in their dashboard (Figure 3.70).
Clicking on the *Cancel* button in the edit menu makes the pop-up disappear. The grid view element is added to the dashboard but any changes made to the content of the textbox are discarded.

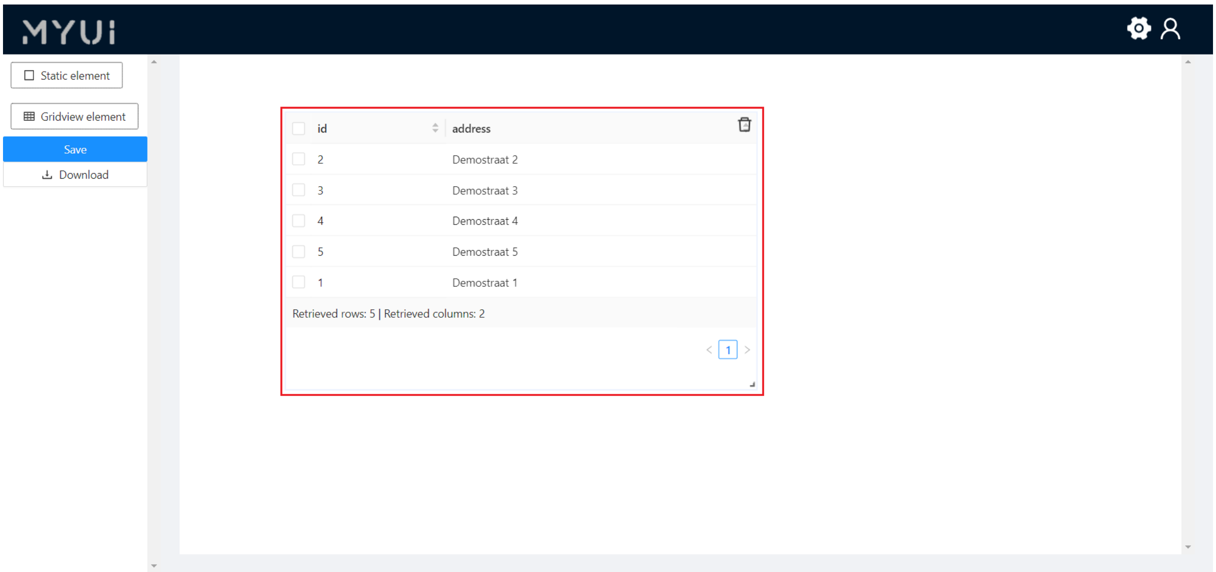

Figure 3.70: The GraphQL query result shown in the dashboard

Likely errors:

None.

3.28 | Edit content of an application element

Functional description:

This tutorial describes how a user can edit the content of an application element in their dashboard while in edit mode.

Cautions and warnings:

Depending on the changes made, the user can encounter the cautions and warnings described in Tutorial 3.25 (on page 60) and in Tutorial 3.27 (on page 64).

Preconditions:

- The user is currently logged into MyUI.
- The user has a dashboard loaded in their working area.
- The user is in edit mode.
- The loaded dashboard contains at least one application element.

Procedure:

1. Double-click on the desired application element.
2. The user will be presented with the edit menu (Figure 3.59 and Figure 3.68).
3. Change the content of the element.
4. Click on the *OK* button (Figure 3.62 and Figure 3.69).

Expected outcome: After clicking on the *OK* button, the application element gets added to the dashboard with the newly modified content.
Clicking on the *Cancel* button in the edit menu makes the pop-up disappear. The changes made to the content of the textbox are discarded.

Likely errors:

None.

3.29 | Moving an application element

Functional description:

This tutorial describes how a user can move an application element within their dashboard while in edit mode.

Cautions and warnings:

There cannot be two elements overlapping, or placed outside the user's working area.

Preconditions:

- The user is currently logged into MyUI.
- The user has a dashboard loaded in their working area.
- The user is in edit mode.
- The loaded dashboard contains at least one application element.

Procedure:

1. Click and hold the element.
2. Drag it to the desired position on the working area (Figure 3.71).

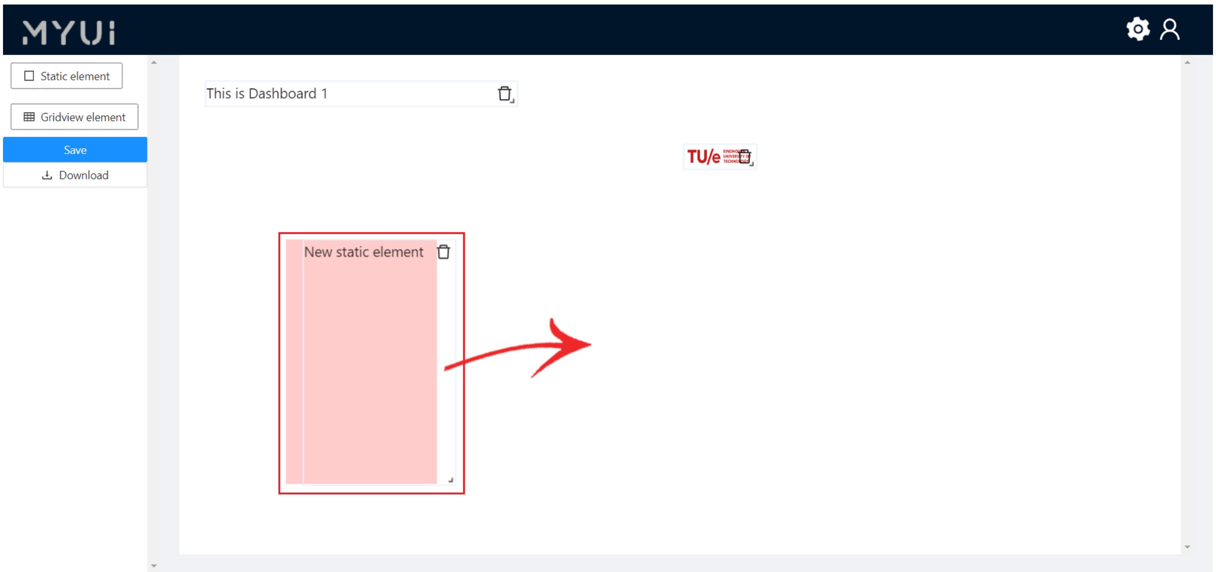

Figure 3.71: The outline of where the application will be placed is shown in red

Expected outcome: After clicking and holding on the desired element, an outline of where the application element would be placed is shown in red. Once dragged to the wanted position, the application element will be moved there.

Likely errors:

None.

3.30 | Deleting an application element

Functional description:

This tutorial describes how a user can delete an application element from their dashboard while in edit mode.

Cautions and warnings:

Once the application element is deleted, the action cannot be undone.

Preconditions:

- The user is currently logged into MyUI.
- The user has a dashboard loaded in their working area.
- The user is in edit mode.
- The user has added an application element to the dashboard.

Procedure:

1. Click on the delete element button in the top right corner of the element to be deleted (Figure 3.72).
2. The user will be presented with a warning prompt.
3. Click on the *Yes* button on the warning prompt (Figure 3.73).

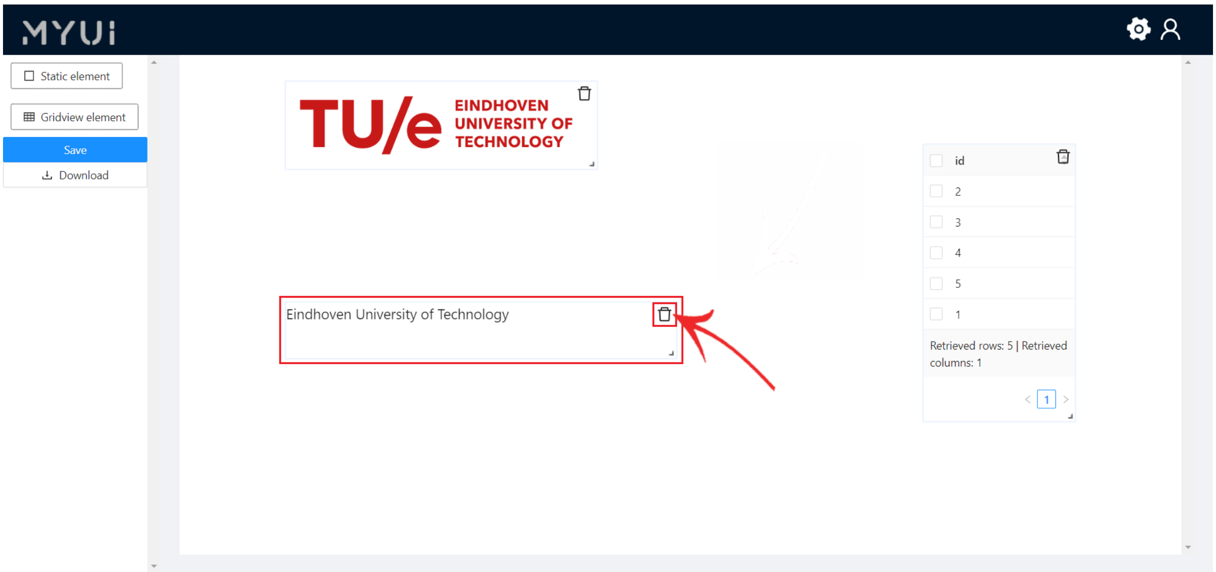

Figure 3.72: The delete element button

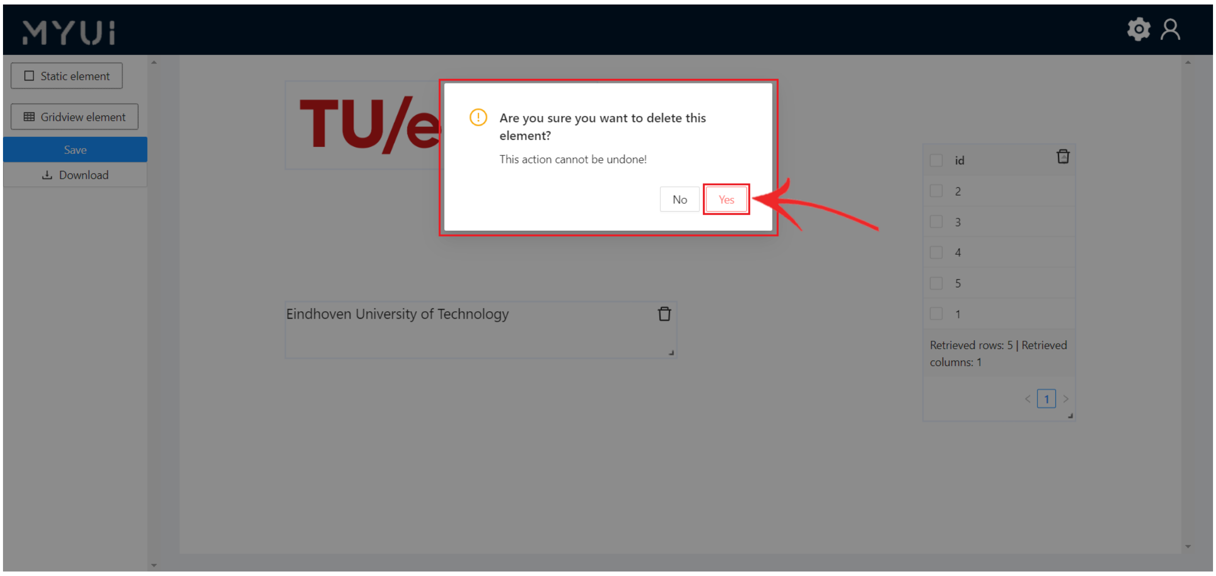

Figure 3.73: The *Yes* button on the deletion warning prompt

Expected outcome: After clicking on the *Yes* button, the element will be deleted from the dashboard (Figure 3.74). This action cannot be undone.

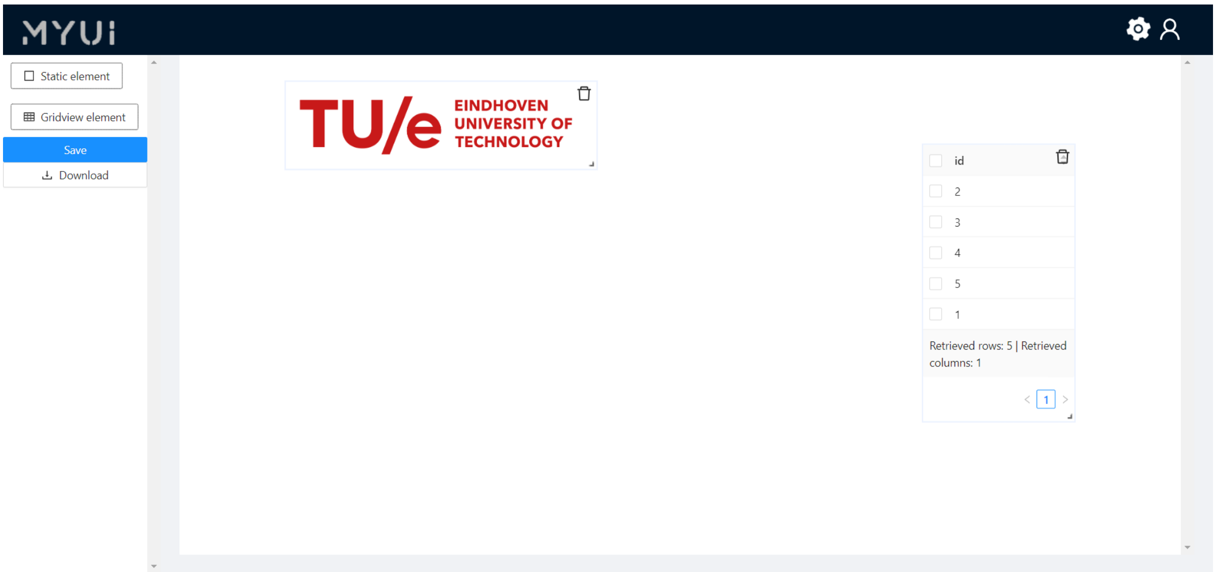

Figure 3.74: The element is deleted from the dashboard

To cancel the deletion, click on the *No* button (Figure 3.75). The warning message will disappear and the static element remains on the dashboard.

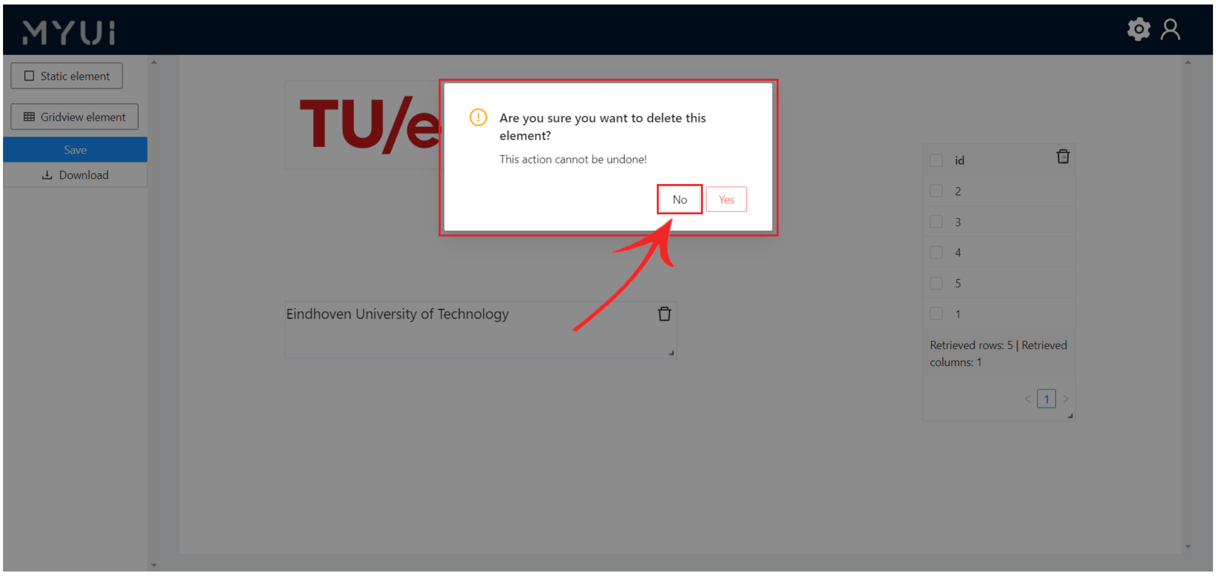

Figure 3.75: The *No* button to cancel the element deletion

Likely errors:

None.

3.31 | Saving changes in edit mode

Functional description:

This tutorial describes how a user can save their changes made to a dashboard while in edit mode.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The user has a dashboard loaded in their working area.
- The user is in edit mode.

Procedure:

1. Click on the *Save* button in the edit mode sidebar (Figure 3.76).

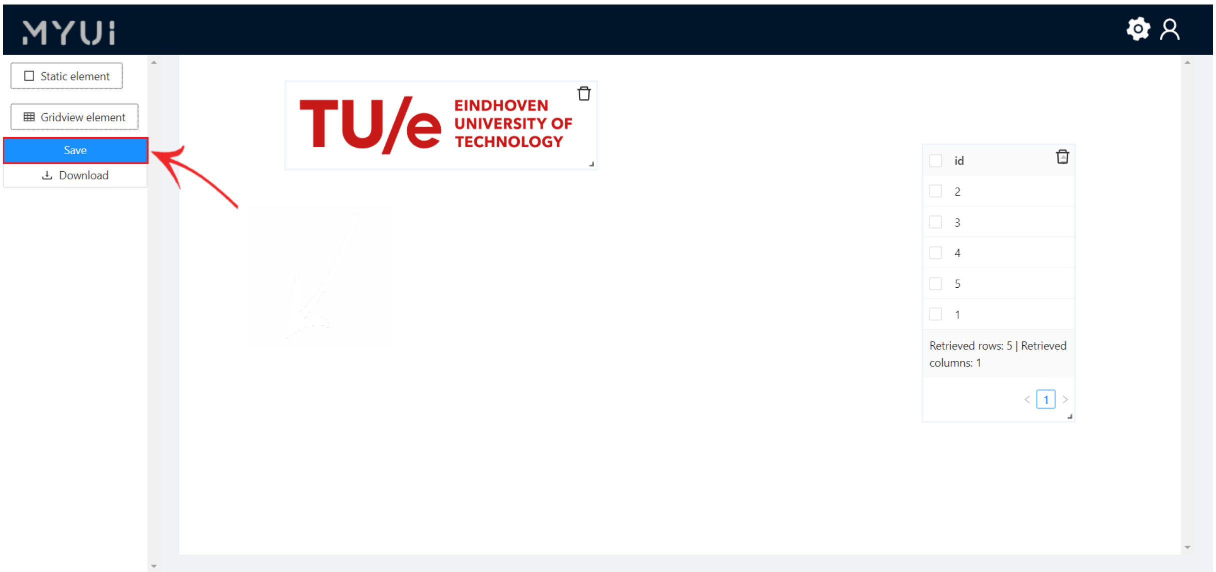

Figure 3.76: The *Save* button

Expected outcome: After clicking on the *Save* button, a notification is shown stating that the changes made have been saved. The user can then exit edit mode without getting the unsaved changes warning message.

Likely errors:

None.

3.32 | Download dashboard configuration

Functional description:

This tutorial describes how a user can download a dashboard configuration in JSON format.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The user has a dashboard loaded in their working area.
- The user is in edit mode.

Procedure:

1. Click on the *Download* button in the edit mode sidebar (Figure 3.77).

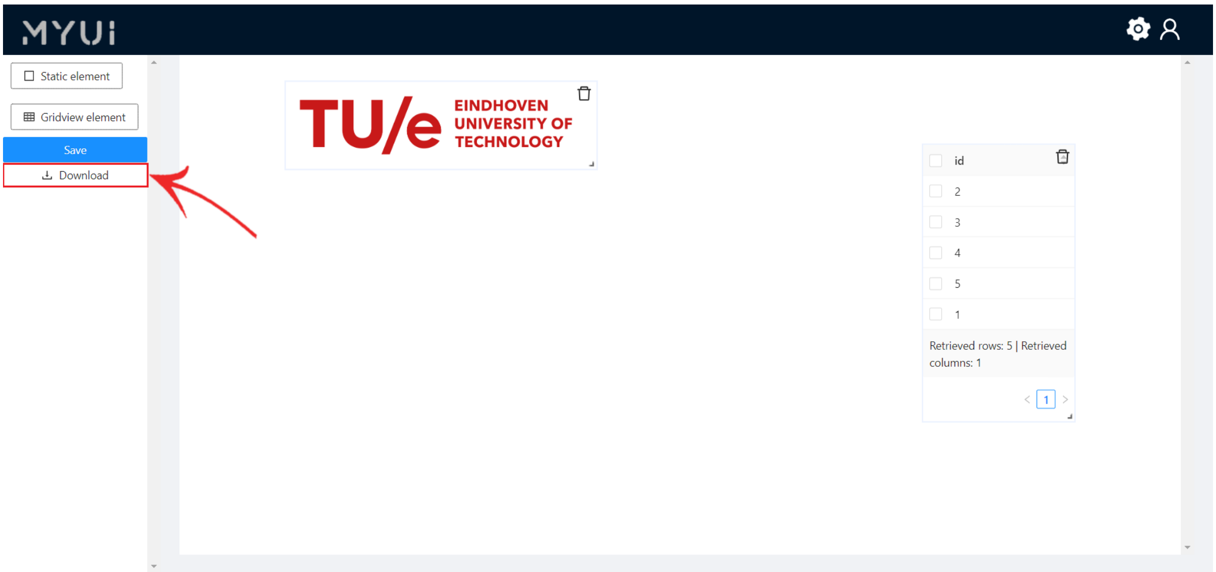

Figure 3.77: The *Download* button

Expected outcome: After clicking on the *Download* button, the configuration for the dashboard loaded in the user's working area is downloaded in JSON format (Figure 3.78).

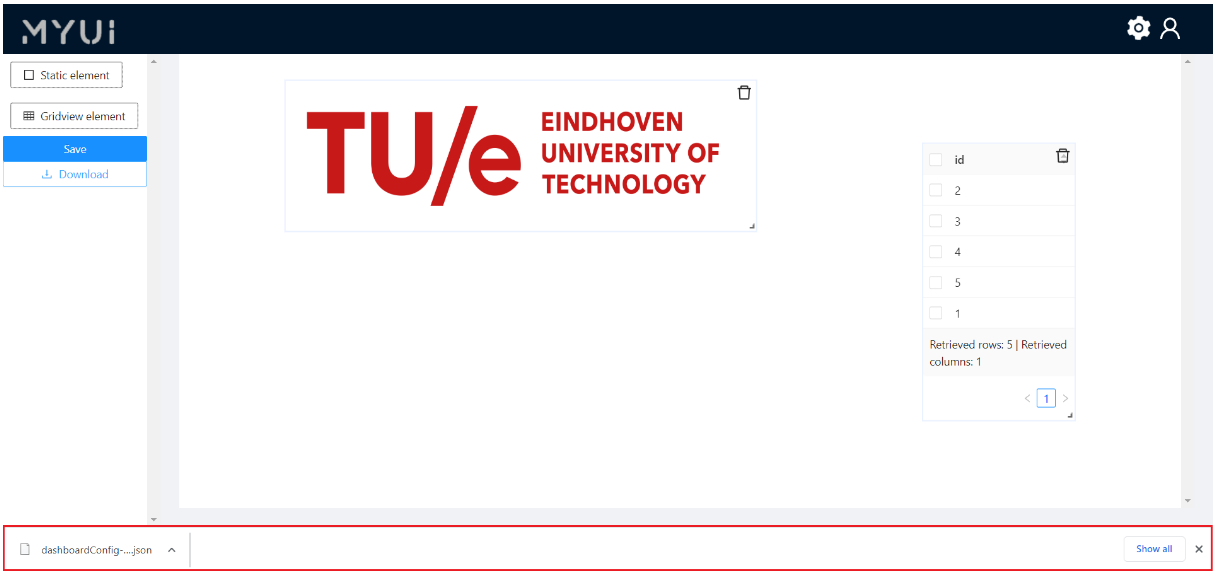

Figure 3.78: The downloaded dashboard configuration

Likely errors:

None.

3.33 | Upload dashboard configuration

Functional description:

This tutorial describes how a user can upload a dashboard configuration in JSON format to create a new dashboard.

Cautions and warnings:

Uploading an unsupported file will cause the application to break.

Preconditions:

- The user is currently logged into MyUI.
- The user has a valid JSON dashboard configuration file on their device.
- The *Dashboards* dropdown menu is unfolded.

Procedure:

1. Click on the *Add* button (Figure 3.41 on page 44).
2. The user will be presented with the *Add a new dashboard* prompt (Figure 3.42 on page 44).
3. Enter the <name> of the new dashboard in the *Name* field.
4. Click on the *Choose File* button (Figure 3.79).
5. Navigate to and choose the desired JSON file to be uploaded from the user's device.
6. Click on the *Open* button (Figure 3.80).
7. The name of the chosen file to be uploaded will appear next to the *Choose File* button.
8. Click the *OK* button (Figure 3.81).

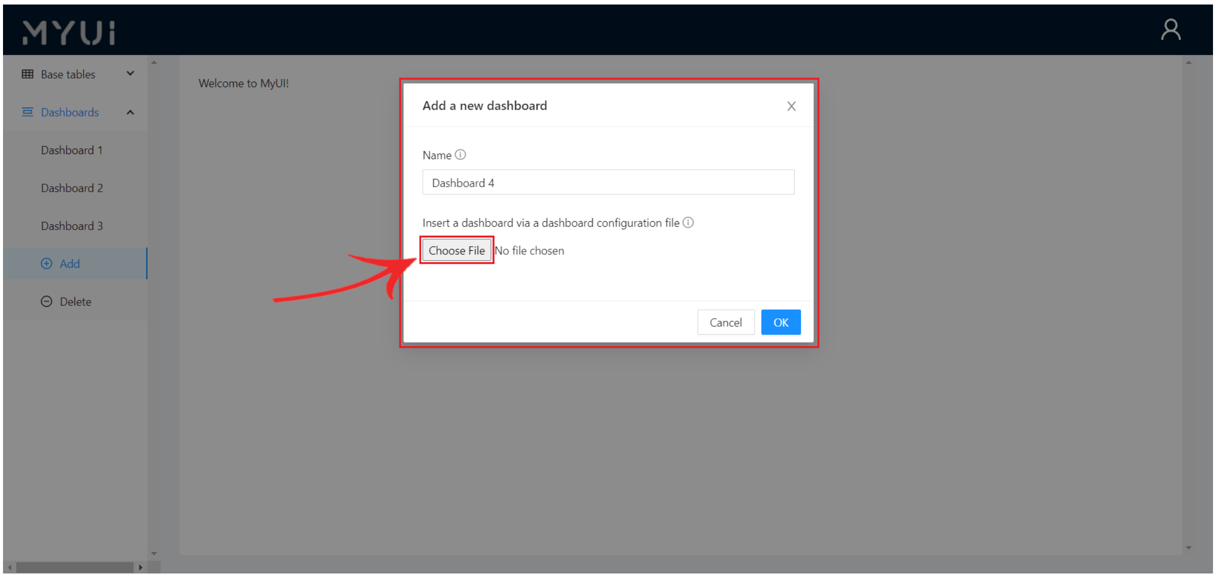

Figure 3.79: The *Choose File* button

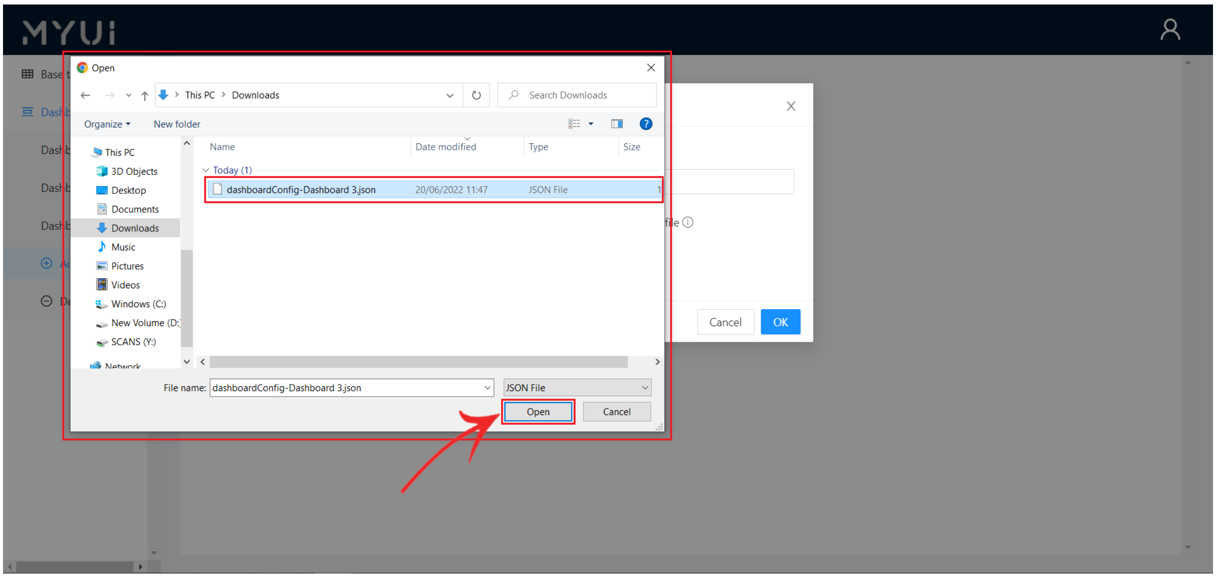

Figure 3.80: The *Open* button

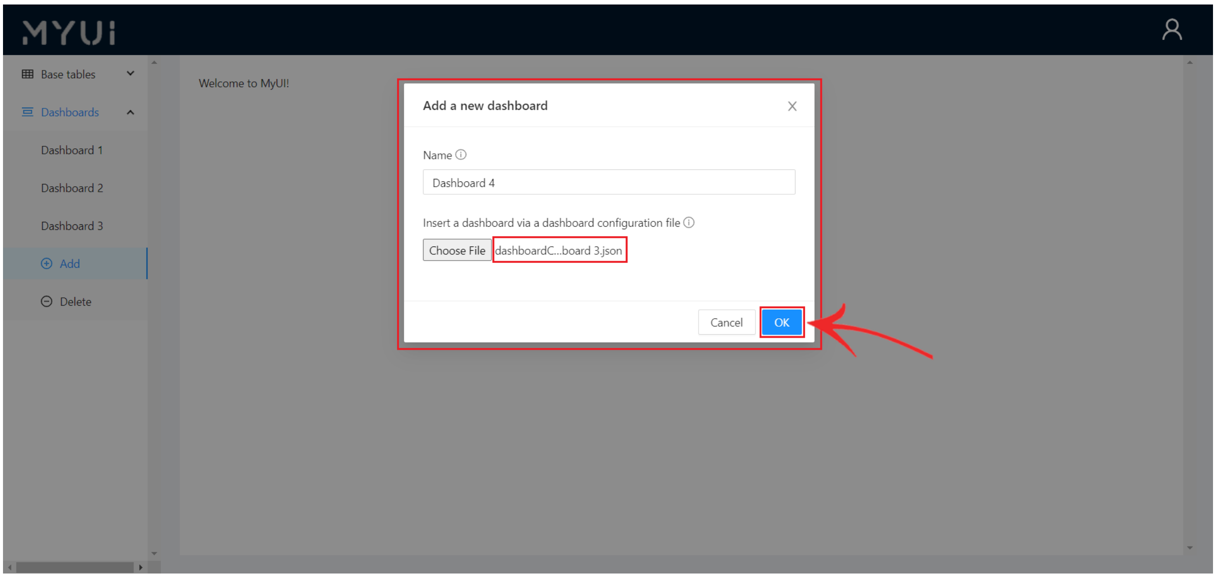

Figure 3.81: The *OK* button for adding the dashboard

Expected outcome: After clicking on the *OK* button, the *Add a new dashboard* prompt disappears and a notification is shown stating that the action was successful. The newly added dashboard will appear in the *Dashboard* menu with the uploaded configuration (Figure 3.44 on page 45).
To cancel the process, click on the *Cancel* button or click on the *X* button (Figure 3.45 on page 46). The *Add a new dashboard* prompt will disappear.

Likely errors:

Filling in an invalid dashboard name in the *Name* field and clicking the *OK* button will cause error messages to show (A.3).

3.34 | Change URL display setting

Functional description:

This tutorial describes how an administrator can change the display settings of the media added to static elements in their dashboards and media table entries in gridviews. The two display options are: URL and Media.

Cautions and warnings:

None.

Preconditions:

- The user is currently logged into MyUI.
- The user has the admin role.

Procedure:

1. Click on the user icon in the header of MyUI (Figure 3.5 on page 13).
2. The user will be presented with a dropdown menu.
3. Click on the *Global Settings* button (Figure 3.82).
4. The user will be presented with the *Global Settings* prompt.
5. Hover over the *URL Display Setting* dropdown menu (Figure 3.83).
6. Select the desired setting.
7. Click on the *OK* button (Figure 3.84).

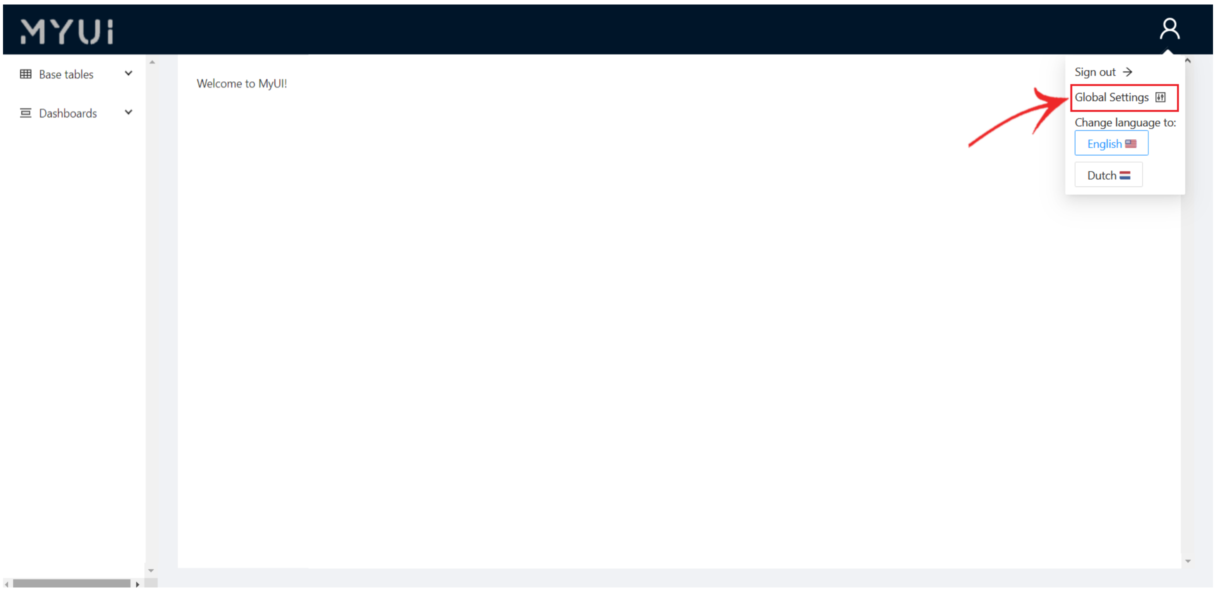

Figure 3.82: The *Global Settings* button

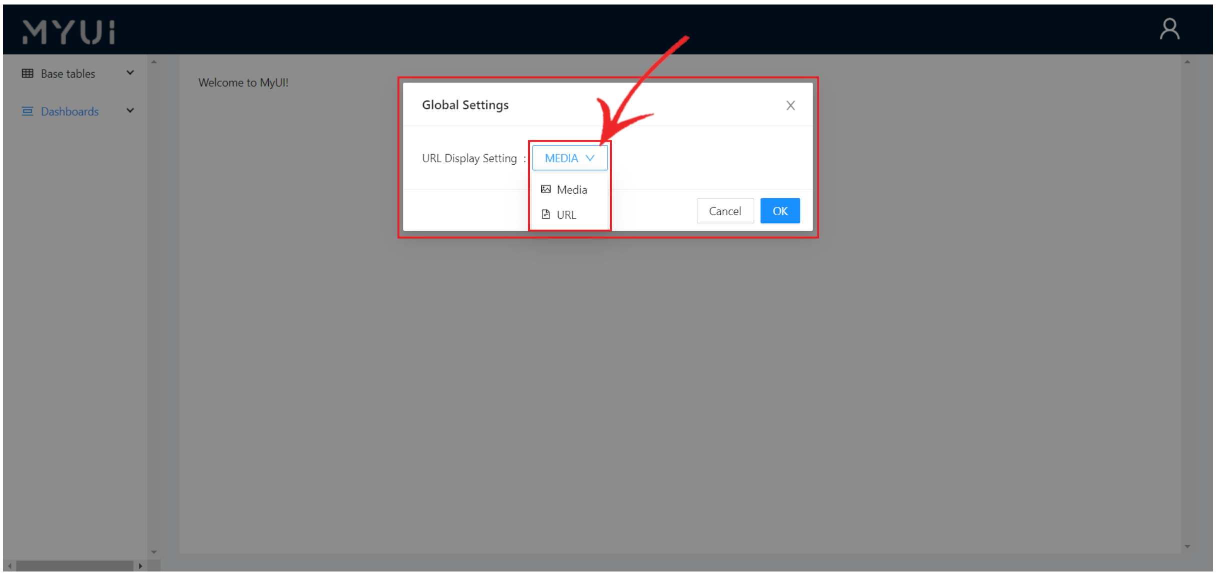

Figure 3.83: The media display options

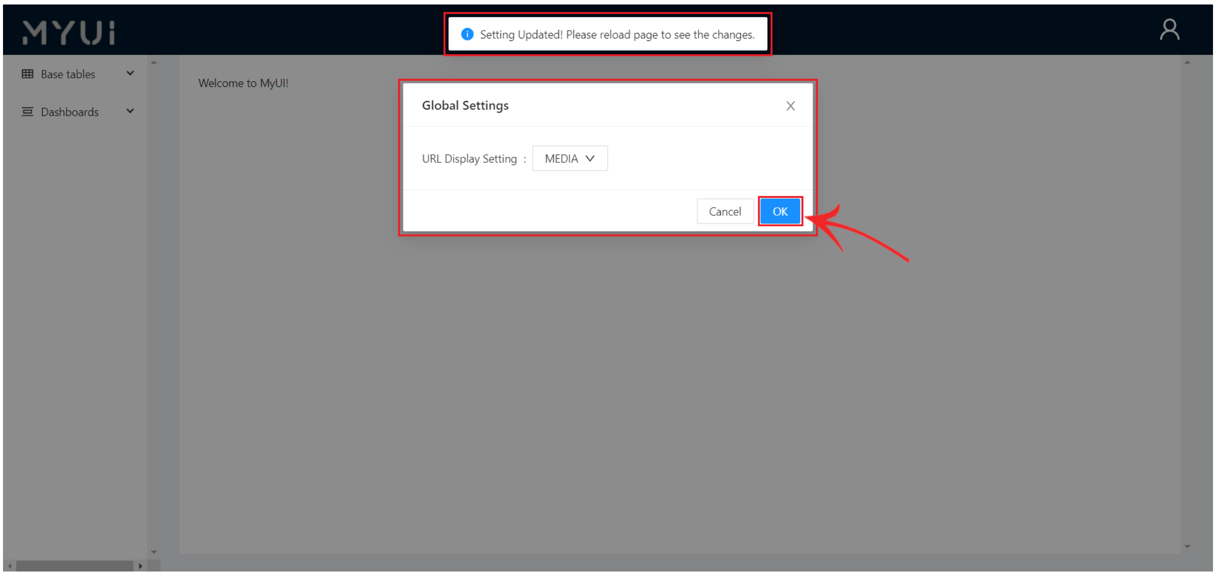

Figure 3.84: The selected desired setting and the *OK* button

Expected outcome: After clicking on the desired setting, a notification is shown stating that the setting was updated. Clicking *OK* closes the *Global Settings* prompt. In order to see the changes, the user has to refresh the page (Figure 3.85).
To cancel the action, click on the *Cancel* button or on the *X* button in the prompt header. The *Global Settings* prompt will disappear (Figure 3.86).

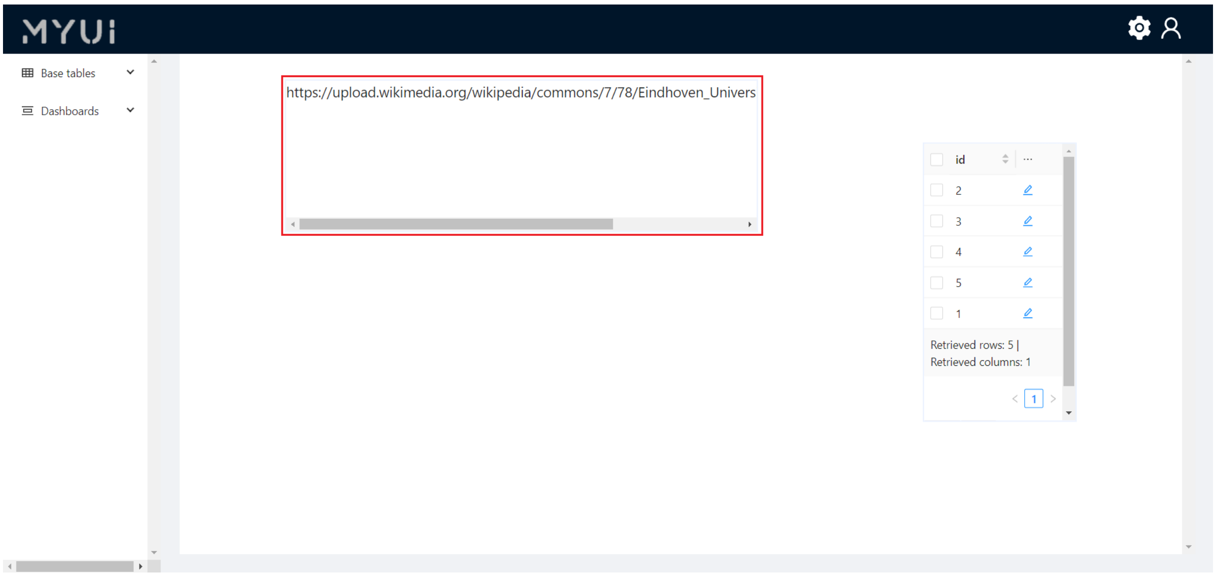

Figure 3.85: The media is displayed as a URL when selecting URL for the display setting

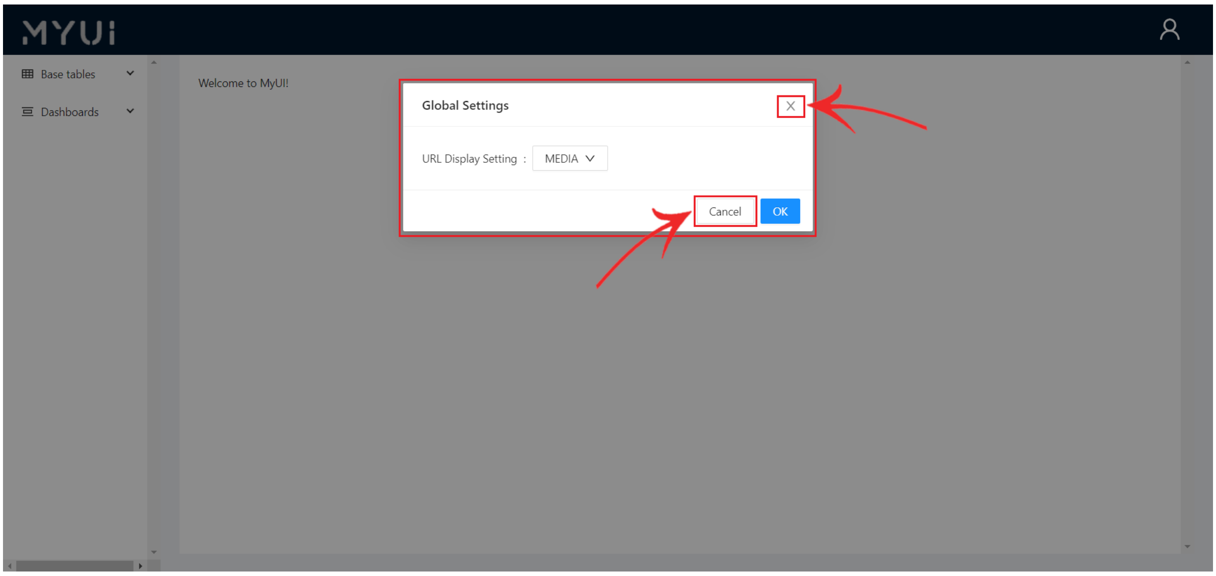

Figure 3.86: The *Cancel* button and *X* button for canceling the change

Likely errors:

None.

4 | Reference

This section provides descriptions of each user interface view and its subviews. A subview is a view that does not show the whole page. An example of a subview is a view of the sidebar of a page rather than the entire page view. This section enables the reader to understand the working of each view, the functioning of its components, and the errors they may get for each operation.

Each view shown in this section includes a functional and formal description, possible errors, images of its view, cautions and warnings, and related references.

The functional description gives an outline of the operations a user can perform on a specific view. The formal description explains each one of these possible actions, how to perform them, and their results. The possible errors, cautions, and warnings list the errors, their causes, and the precautions users should take before performing specific actions. Related references are given to link the different views, making it easier to understand the functionalities of these views.

4.1 | Login

Functional Description: The *Login* page (Figure 4.1) allows the user to fill in their credentials to sign into their MyUI account. It is not possible to create a new account in the application itself.

Cautions and Warnings: None.

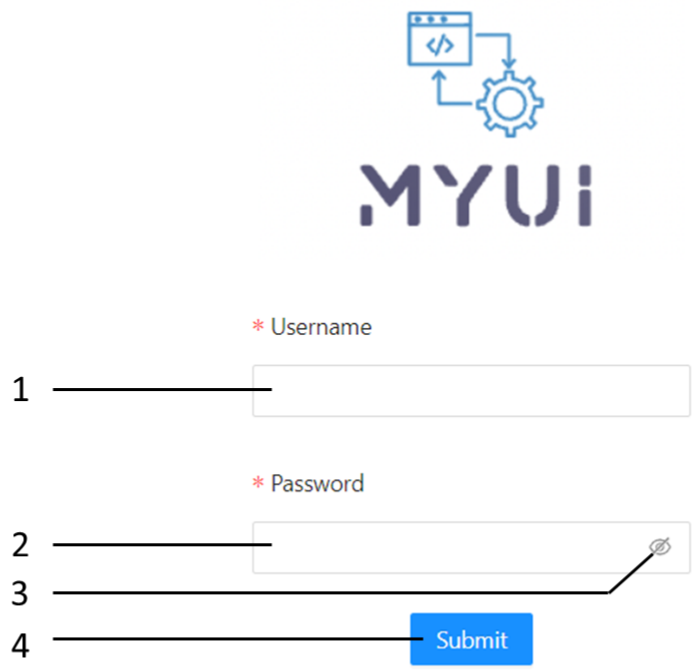

Figure 4.1: The login page of MyUI

Formal Description:

	Operation	Description	Result
1	Enter username	Click on the *Username* text field and fill in the <username>	<username> is displayed in the *Username* text field
2	Enter password	Click on the *Password* text field and fill in the <password>	<password> is displayed in the *Password* text field. By default this password is masked, but the user can choose to unmask the password
3	Mask/Unmask password	Click on the *Eye Icon* to enable/disable masking the password	If enabled, the contents of the *Password* text field will be shown masked. If disabled, the contents of the *Password* text field will be shown unmasked
4	Attempt to sign in with the filled-in credentials	Click on the *Submit* button	If the username is incorrect, the page will reload to let the user fill in the credentials again. If the user has entered an incorrect username and a password, the application will show an error page. If the credentials are correct, the user will be directed to the homepage

Examples:

	Operation	Example
1	Enter username	A user logs into the application with the username ***user1***
2	Enter password	User ***user1*** logs into the application with the password ***sep2022***
3	Mask/Unmask password	User ***user1*** views their password to check if they have entered the password ***sep2022*** correctly
4	Attempt to sign in with the filled in credentials	User **user1** logs into the application with the username ***user1*** and password ***sep2022***

Possible Errors:

Error	Cause(s)
Please input your username!	The user has not filled in the username
Please input your password!	The user has not filled in the password
Page reload	The user fills in an invalid password or fills in invalid inputs for both fields
Error page	The user fills in an incorrect username and a password.

Related References: None.

4.2 | MyUI User Interface

4.2.1 | MyUI Sidebar

Functional Description: This component of the application's user interface consists of two dropdown menus that the user can click on. The first dropdown menu is *Base Tables* and the second dropdown menu is *Dashboards* (Figure 4.2).

Cautions and Warnings: None.

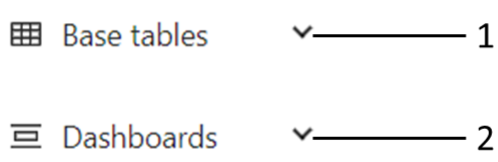

Figure 4.2: The MyUI Sidebar

Formal Description:

	Operation	Description	Result
1	Open list of base tables	Click on the *Base Tables* menu	The user is presented with a list of base tables
2	Open list of dashboards	Click on the *Dashboards* menu	The user is presented with a list of dashboards as well as the option to create and delete dashboards

Examples:

	Operation	Example
1	Open list of base tables	***user1*** opens the list of base tables containing the tables ***Customer***, ***Inventory*** and ***Purchase***
2	Open list of dashboards	***user1*** opens the list of dashboards containing the dashboards ***Customer Information***, ***Inventory Information*** and ***Purchase Information***

Possible Errors: None.

Related References: 4.2.4, 4.4.1

4.2.2 | MyUI Header

Functional Description: This component of the application's user interface consists of two icons that the user can click on. When a user logs in, the header contains only the user button. When a user views a dashboard, the user can see two icons - the user button and the change-mode button. The change-mode button can be clicked on to switch to edit mode (Figure 4.3).

Cautions and Warnings:

Caution	Cause(s)
Global Settings button is not present	User does not have administrator permissions
Change-mode button is not present	User is not viewing a dashboard

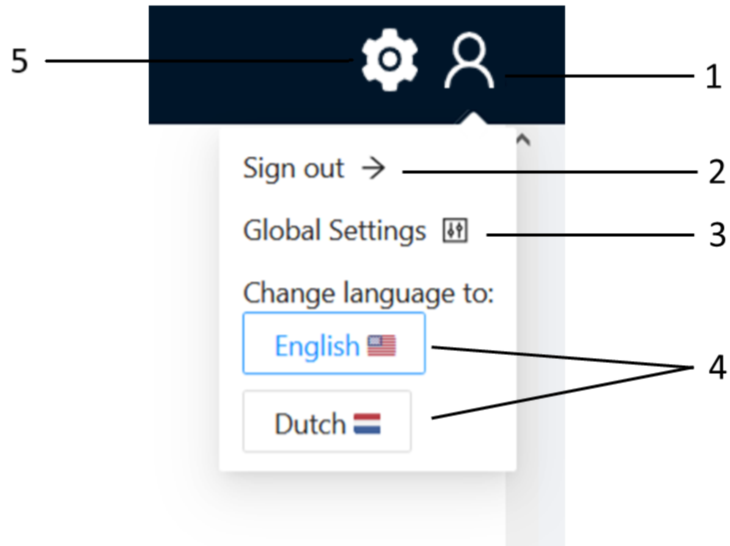

Figure 4.3: The MyUI Header

Formal Description:

	Operation	Description	Result
1	View options for profile settings of the account	Click on the user button	The user is presented with a dropdown menu with options for the languages to view the application in, the global settings button, and the sign out button. If these options have already been displayed on the screen, clicking on the user button again will close the list of options
2	Sign out	Click on the *Sign Out* button	The user is signed out of the application and is redirected to the login page
3	Change global settings	Click on the *Global Settings* button	The dropdown menu to change the media display settings is displayed

4	Change the language	Click on the language button corresponding to the desired language	The application will be displayed in the selected language and the dropdown menu will close. The user will be redirected to the homepage
5	Change the mode	Click on the change-mode button	The user will enter edit mode. If a user clicks on this button while in edit mode, they will exit edit mode

Examples:

	Operation	Example
1	View options for profile settings of the account	An user ***user1*** clicks the user button to see the user menu.
2	Sign out	***user1*** signs out of the application
3	Change global settings	An admin ***admin1*** changes the media setting such that the media in the application is displayed as a *URL* instead of *Media*
4	Change the language	***user1*** to changes the display language of the application from *Dutch* to *English*
5	Change the mode	***user1*** enters edit mode for the dashboard ***Customer Information***

Possible Errors: None.

Related References: 4.1, 4.2.3, 4.5.1, 4.5.6

4.2.3 | Change media settings

Functional Description: This feature allows an administrator to change the media settings of the application for all users. By clicking on the *Global Settings* button, a prompt appears where an administrator can switch the settings for the media between *URL* and *Media* through a dropdown menu (Figure 4.4). When an administrator changes the setting to *URL*, the media for all users will be displayed as links instead of being rendered. When an administrator changes the setting to *Media*, the media will be displayed for all users as the rendered media.

Cautions and Warnings: None.

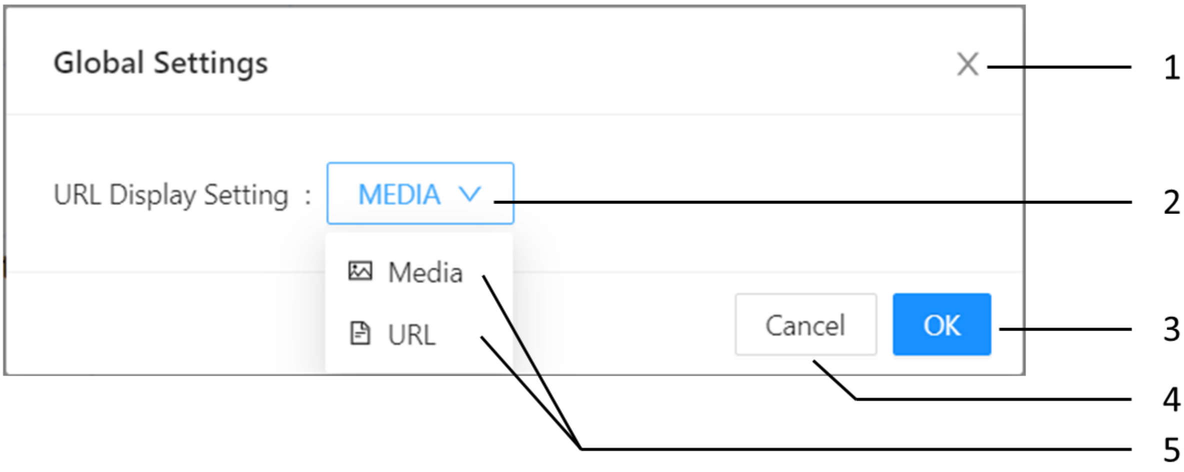

Figure 4.4: The media settings prompt

Formal Description:

	Operation	Description	Result
1	Cancel the changes made for the media setting	Click on the 'X' at the top right of the dropdown menu	The changes made will not be saved and the prompt for the media settings will close
2	View the options for the media settings	Hover over the dropdown menu in the global settings menu	The dropdown menu for the media settings is displayed for the administrator to choose between
3	Save media settings and exit the dropdown menu	Click on the *OK* button at the bottom right of the media-setting menu	The media-setting menu will close and the dropdown menu for the media settings will close
4	Cancel the changes made for the media setting	Click on the *Cancel* button at the bottom right of the media-setting menu	The changes made will not be saved and the dropdown menu for the media settings will close
5	Select a media setting	Click on an option to set the media setting	The administrator gets a confirmation message that the setting has been updated. To see the changes in the media display, the administrator and users must reload the page

Examples:

	Operation	Example
1	Cancel the changes made for the media setting	***admin1*** does not change the media display setting from *Media* to *URL*
2	View the options for the media settings	***admin1*** checks to confirm that the media settings are correctly set to *Media*
3	Save media settings and exit the dropdown menu	***admin1*** saves the media display setting as *URL*
4	Cancel the changes made for the media setting	***admin1*** does not change the media display setting from *Media* to *URL*
5	Select a media setting	***admin1*** selects the *URL* display setting instead of *Media*

Possible Errors: None.

Related References: 4.2.2

4.2.4 | View base table

Functional Description: The *Base Tables* dropdown menu contains the list of the various base tables accessible to the user. Once they have opened the menu, users can click on the different names in the dropdown menu to view their corresponding base tables (Figure 4.5).

Cautions and Warnings: None.

Figure 4.5: The *Base Tables* dropdown menu

Formal Description:

	Operation	Description	Result
1	View the base table names	Click on the *Base Tables* dropdown heading	The list of base table names will appear below the *Base Tables* heading
2	View a base table	Click on a base table name	The corresponding base table will load in the main page

Examples:

	Operation	Example
1	View the base table names	***user1*** views the names of the base tables: ***Customer***, ***Inventory*** and ***Purchase***
2	View a base table	***user1*** views the base table ***Customer***

Possible Errors: None.

Related References: 4.2.1

4.3 | Grid View Operations

4.3.1 | Viewing a grid view

Functional Description: Users can perform many operations on a grid view. They can select, edit, insert and delete rows in a grid view. The delete-row button can only be clicked on if at least one row in the grid view has been selected. In addition to these row operations, users can also change how a grid view is displayed. The rows of a grid view are originally ordered according to how the data has been fetched from Hasura. To change the ordering of a column in a grid view, a user can click on the arrow button in the header of that column. This will change the ordering of a column to be ascending, descending, or to remove the ordering from the column. The pagination settings of a grid view can also be modified by changing the settings that are provided at the bottom of each grid view (Figure 4.6).

Cautions and Warnings:

Caution	Cause(s)
+ button is not present	User does not have insert-access permission
- button is disabled	No row has been selected in the grid view
The next page button is disabled	There are no succeeding pages that can be displayed
The previous page button is disabled	There are no preceding pages that can be displayed

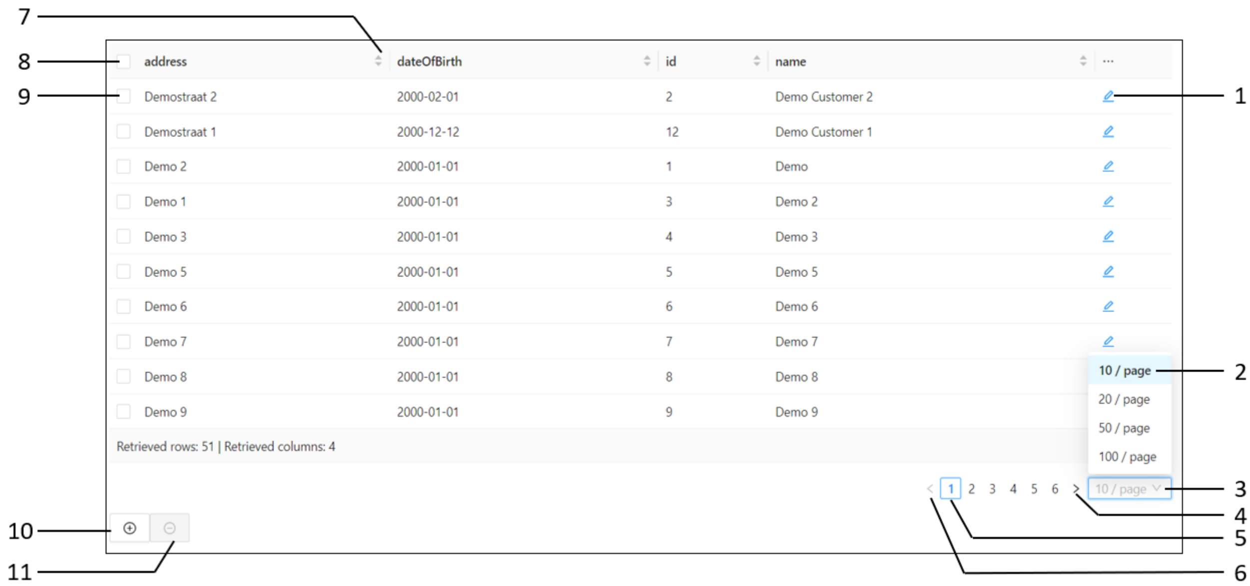

Figure 4.6: Elements of a grid view

Formal Description:

	Operation	Description	Result
1	Edit a row	Click on the edit-row button displayed at the end of the row to be edited	Each attribute of the row being edited will be temporarily replaced by an input field containing the current attribute value which the user can edit
2	Select the number of rows displayed per page	Click on one of the options displayed by the pagination dropdown menu	The grid view will update to group its rows according to the selected number of rows per page. The user will then be shown the grid view with the selected pagination setting
3	View the options for the number of rows displayed per page	Click on the pagination dropdown menu on the bottom right of the grid view	The user will be able to see the various options for the number of rows that are displayed for each page
4	View the next page of a grid view	Click on the arrow to the right of the displayed page numbers	The user will be shown the next page of the grid view
5	Select the page of the grid view to be viewed	Click on a page number at the bottom right of the grid view to view its corresponding page	The user will be shown the selected page of the grid view
6	View the previous page of a grid view	Click on the arrow to the left of the displayed page numbers	The user will be shown the previous page of the grid view

7	Order a column	Click on the ordering button next to the column name of the column to be ordered	If there was no previous ordering on that column, clicking on the ordering button will result in the column being ordered in ascending order. If the column was previously ordered in ascending order, clicking on the ordering button will result in the column being ordered in descending order. If the column was previously ordered in descending order, clicking on the ordering button will remove the ordering from the column. Removing the ordering results in the column being ordered in terms of how the data has been fetched from Hasura. Once the ordering has been changed, the grid view is sorted accordingly and displayed
8	Select and deselect all rows in a grid view	Click on the checkbox displayed to the left of the column headers	All the rows and their corresponding checkboxes will be selected. If all the rows were already selected, clicking on the checkbox in the column header will deselect all the rows. The checkboxes for all the rows as well as the checkbox in the column header will then be deselected
9	Select and deselect a row	Click on the checkbox displayed at the beginning of the row to be selected	The corresponding row and its checkbox will be selected. If the row was already selected, clicking on the checkbox will deselect the row and the checkbox will no longer be selected
10	Insert a row	Click on the insert-row button at the bottom of the grid view	The insert-row menu will appear on the screen for the user to fill in
11	Delete a row	Click on the delete-row button at the bottom of the grid view	A warning prompt asking the user to confirm the deletion will appear on the screen

Examples:

	Operation	Example
1	Edit a row	***user1*** changes the name of ***Customer X*** to ***Customer Y*** in the ***Customer*** base table
2	Select the number of rows displayed per page	***user1*** changes the pagination setting to **50** rows per page for the ***Customer*** base table
3	View the options for the number of rows displayed per page	***user1*** checks the maximum number of rows that can be displayed per page for the ***Customer*** base table
4	View the next page of a grid view	***user1*** selects Page ***2*** of the ***Customer*** base table after viewing Page ***1***
5	Select the page of the grid view to be viewed	***user1*** selects Page ***5*** of the ***Customer*** base table

6	View the previous page of a grid view	***user1*** selects Page ***1*** of the ***Inventory*** base table after viewing Page ***2***
7	Order a column	***user1*** clicks the header of the ***Product Names*** column in the ***Inventory*** base table to view the table in alphabetical order on ***Product Names***
8	Select and deselect all rows in a grid view	***user1*** selects all the rows in the ***Inventory*** base table to delete them
9	Select and deselect a row	***user1*** selects a row in the ***Inventory*** base table to delete a product ***Chairs*** that is no longer in stock
10	Insert a row	***user1*** adds a row in the ***Inventory*** base table for a new product ***Tables***
11	Delete a row	***user1*** deletes a product ***Chairs*** from the ***Inventory*** base table

Possible Errors: None.

Related References: 4.3.2, 4.3.3, 4.3.4

4.3.2 | Insert a row

Functional Description: In addition to viewing grid views, users can perform extra functions on grid views. One of these functions is inserting a row. To insert a row, users can click on the insert-row button and fill in the attribute values of the required fields (Figure 4.7). The new changes to the grid view will be saved to the Hasura database.

Cautions and Warnings: None.

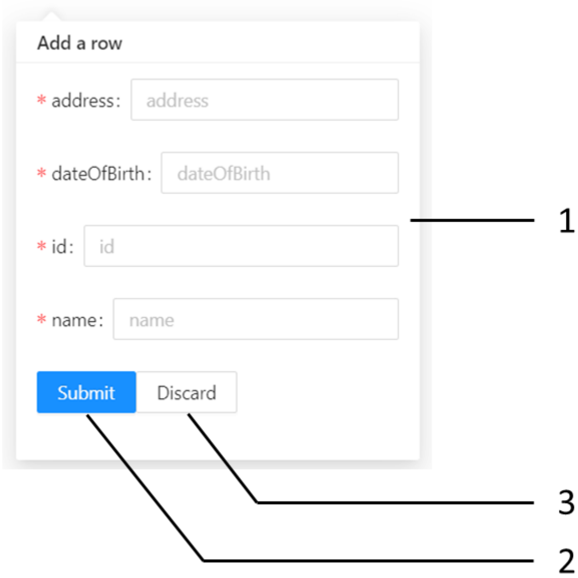

Figure 4.7: The *Add a Row* prompt

Formal Description:

	Operation	Description	Result
1	Enter the new row's details	Fill in the details of each required field for the row	The text fields contain the contents of the new row
2	Submit changes	Click on the *Submit* button	The new changes will be saved to the grid view. The *Add a Row* prompt will close and the new row will be displayed
3	Discard changes	Click on the *Discard* button	The *Add a Row* prompt will be closed and no new changes will be made to the grid view

Examples:

	Operation	Example
1	Enter the new row's details	***user1*** enters the *Product Name* and *Quantity* of the product ***Chairs*** to add to the ***Inventory*** base table
2	Submit changes	***user1*** adds and saves the information about the product ***Chairs*** to the ***Inventory*** base table
3	Discard changes	***user1*** does not add information about the product ***Chairs*** to the ***Inventory*** base table

Possible Errors:

Error	Cause(s)
General Hasura error	The input field for an attribute has been filled with a value whose type does not match the column type
This is a required field	The input field for an attribute is empty

Related References: 4.3.1

4.3.3 | Edit a row

Functional Description: Another function that a user can perform on a grid view is to edit rows. To edit a row, users can click on the edit button of the row they would like to edit, where they can enter new values for the row's attributes (Figure 4.8). The new values of the grid view will be saved to the database in Hasura.

Cautions and Warnings:

Caution	Cause(s)
Edit-row button is absent	User does not have update permission

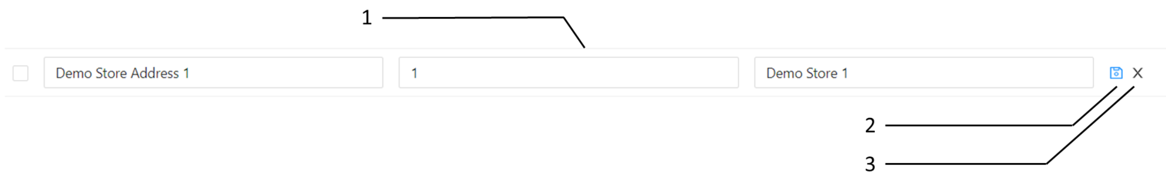

Figure 4.8: Editing a row

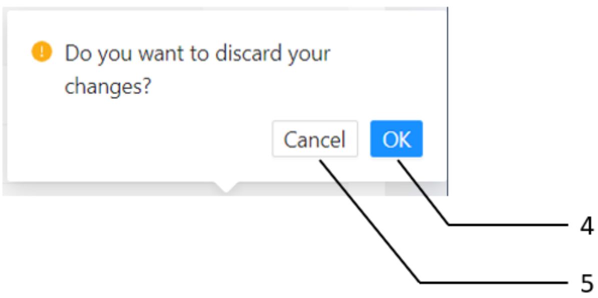

Figure 4.9: Edit row warning prompt

Formal Description:

	Operation	Description	Result
1	Enter new attribute values	Click on the text field that each attribute is temporarily replaced with and enter the attribute's new value	The text fields will display the new values entered for the row
2	Save changes	Click on the *Save* button at the end of the row	The changes made to the row are saved to the database. The text fields close and are replaced with the new attributes entered in by the user
3	Discard changes	Click on the *X* button at the end of the row	A warning prompt appears asking the user if they want to discard their changes
4	Confirm discarding changes	Click on the *OK* button on the warning prompt	The active changes are not applied to the grid view. The warning prompt and the input fields will close and the input fields will be replaced with the previous row values
5	Cancel discarding changes	Click on the *Cancel* button on the warning prompt	The warning prompt will close and the input fields will continue to be displayed

Examples:

	Operation	Example
1	Enter new attribute values	***user1*** changes the name of ***Customer X*** to ***Customer Y*** in the ***Customer*** base table
2	Save changes	***user1*** saves the changes made to the name of ***Customer Y*** from ***Customer X*** in the ***Customer*** base table
3	Discard changes	***user1*** does not change the name of ***Customer X*** to ***Customer Y*** in the ***Customer*** base table
4	Confirm discarding changes	***user1*** confirms the warning prompt to not change the name of ***Customer X*** to ***Customer Y*** in the ***Customer*** base table
5	Cancel discarding changes	***user1*** cancels discarding the change of ***Customer X*** to ***Customer Y*** in the ***Customer*** base table

Possible Errors:

Error	Cause(s)
General Hasura error	The input field for an attribute has been filled with a value whose type does not match the column type
<column name> cannot be empty	The input field for an attribute is empty

Related References: 4.3.1

4.3.4 | Delete a row

Functional Description: The last function that can be performed is the deletion of rows. After selecting a row/rows, users can click on the - button at the bottom of a grid view where a warning prompt will be displayed (Figure 4.10). If a user confirms the deletion, the row will be removed and the changes to the grid view will be saved to the Hasura database.

Cautions and Warnings: None.

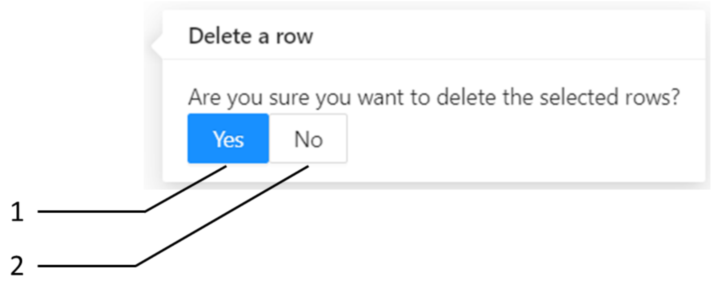

Figure 4.10: The *Delete a row* warning prompt

Formal Description:

	Operation	Description	Result
1	Confirm deletion	Click on the *Yes* button on the warning prompt	The new changes will be saved to the grid view. The warning prompt will close and the row will be deleted from the grid view
2	Discard deletion	Click on the *No* button on the warning prompt	No new changes will be made to the grid view and the warning prompt will close

Examples:

	Operation	Example
1	Confirm deletion	***user1*** confirms the deletion of ***Chairs*** from the ***Inventory*** base table
2	Discard deletion	***user1*** cancels the deletion of ***Chairs*** from the ***Inventory*** base table

Possible Errors:

Error	Cause(s)
General Hasura error	The row has already been deleted from the database and the user's table was not updated

Related References: 4.3.1

4.4 | Dashboard Operations

4.4.1 | View dashboard

Functional Description: The *Dashboards* dropdown menu contains the list of the various dashboards accessible to the user. Once a user has opened the menu, they can click on the different names in the dropdown menu to view their corresponding dashboards (Figure 4.11).

Cautions and Warnings: None.

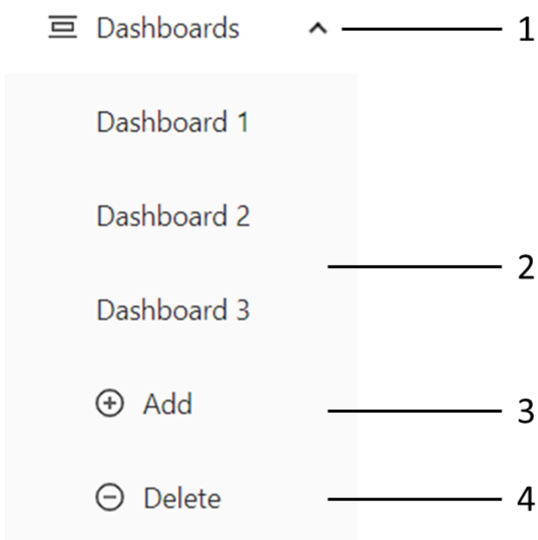

Figure 4.11: The *Dashboards* dropdown menu

Formal Description:

	Operation	Description	Result
1	View the dashboard names	Click on the *Dashboards* dropdown heading	The list of dashboard names will appear below the *Dashboards* heading. If the list of dashboard names was already shown, the dropdown menu closes
2	View a dashboard	Click on a dashboard name	The corresponding dashboard will load in the working area
3	Add a dashboard	Click on the + button	The *Add a Dashboard* prompt will appear on the screen
4	Delete a dashboard	Click on the - button	The *Remove a Dashboard* prompt will appear on the screen

Examples:

	Operation	Example
1	View the dashboard names	***user1*** views the names of the dashboards: ***Customer Information***, ***Inventory Information*** and ***Purchase Information***
2	View a dashboard	***user1*** views the dashboard ***Customer Information***
3	Add a dashboard	***user1*** adds a dashboard ***Store Information***
4	Delete a dashboard	***user1*** deletes a dashboard ***Purchase Information***

Possible Errors: None.

Related References: 4.2.1, 4.4.2, 4.4.3

4.4.2 | Add a dashboard

Functional Description: In addition to viewing dashboards, users can also add dashboards. Users can do this either by creating a new empty dashboard or by uploading a dashboard configuration file.

Cautions and Warnings:

Caution	Cause(s)
Application stops responding	An unsupported file has been submitted while submitting a dashboard configuration file
JSON.parse error	The submitted JSON file is in the wrong format. A submitted JSON file is in the right format when it has been downloaded from MyUI

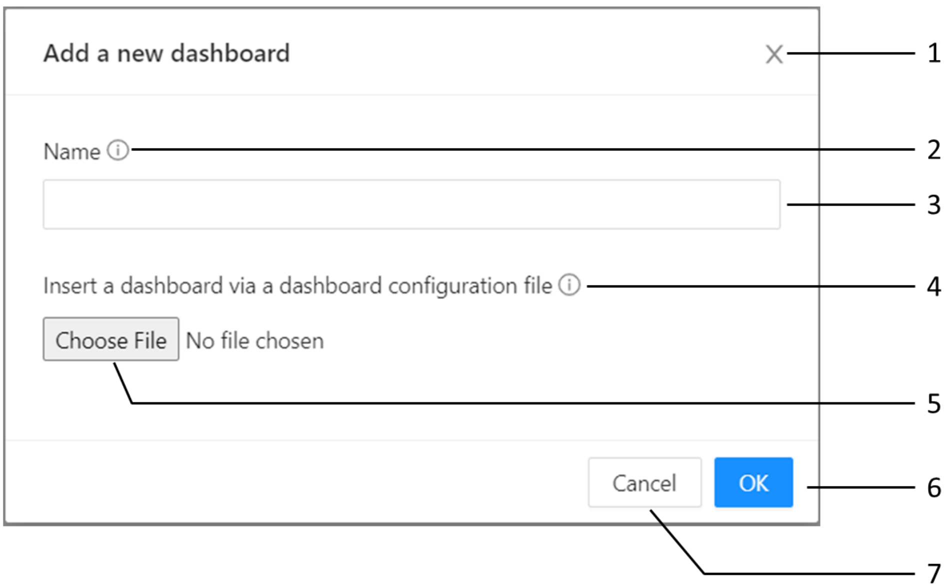

Figure 4.12: The *Add a Dashboard* prompt

Formal Description:

	Operation	Description	Result
1	Cancel addition of dashboards	Click on the 'X' button at the top right of the *Add a Dashboard* prompt	No dashboard will be added and the *Add a Dashboard* prompt will close
2	View information about the name of the dashboard	Hover over the info button next to the *Name* heading	Information about the dashboard name will be displayed as a message
3	Create a name for dashboard	Click on the text field under *Name* and fill in the name of the dashboard	The input field for the dashboard name will contain the new dashboard name
4	View information about the format of the dashboard configuration file	Hover over the info button next to the heading *Insert a dashboard via a dashboard configuration file*	Information about the dashboard configuration file will be displayed as a message
5	Upload dashboard configuration file	Click on the *Choose File* button to upload a valid JSON configuration file	The name of the uploaded file appears next to the *Choose File* button
6	Confirm adding dashboard	Click on the *OK* button at the bottom of the *Add a Dashboard* prompt	The new dashboard will be created and its name will be added to the names of the dashboard menu. The user will then be taken to the new dashboard
7	Cancel addition of dashboards	Click on the *Cancel* button at the bottom of the *Add a Dashboard* prompt	No dashboard will be added and the *Add a Dashboard* prompt will close

Examples:

	Operation	Example
1	Cancel addition of dashboards	***user1*** cancels adding a dashboard
2	View information about the name of a dashboard	***user1*** views the information on how the new dashboard should be named
3	Create name for dashboard	***user1*** names the new dashboard ***Store Information***
4	View information about the format of dashboard configuration file	***user1*** views the information on what type of configuration file can be uploaded for a dashboard
5	Upload dashboard configuration file	***user1*** uploads a dashboard configuration file named ***storeInformation.json***
6	Confirm adding dashboard	***user1*** adds the dashboard ***Store Information***
7	Cancel addition of dashboards	***user1*** cancels adding a dashboard

Possible Errors:

Error	Cause(s)
Sorry, you cannot use this name	The input field for the dashboard name is empty or contains the following: only spaces, a backlash, or double quotation marks. Naming the dashboard *dashboardAdd* or *dashboardDelete* will also cause the application to throw this error
JSON.parse error	The submitted JSON file is in the wrong format. A submitted JSON file is in the right format when it has been downloaded from MyUI

Related References: 4.4.1

4.4.3 | Delete a dashboard

Functional Description: If a user has a dashboard that they no longer wish to view, they have the option to remove the dashboard. To do this, the user can click on the - button. A *Remove a Dashboard* prompt will then be displayed where the user can enter the details of the dashboard to be deleted (Figure 4.13).

Cautions and Warnings: None.

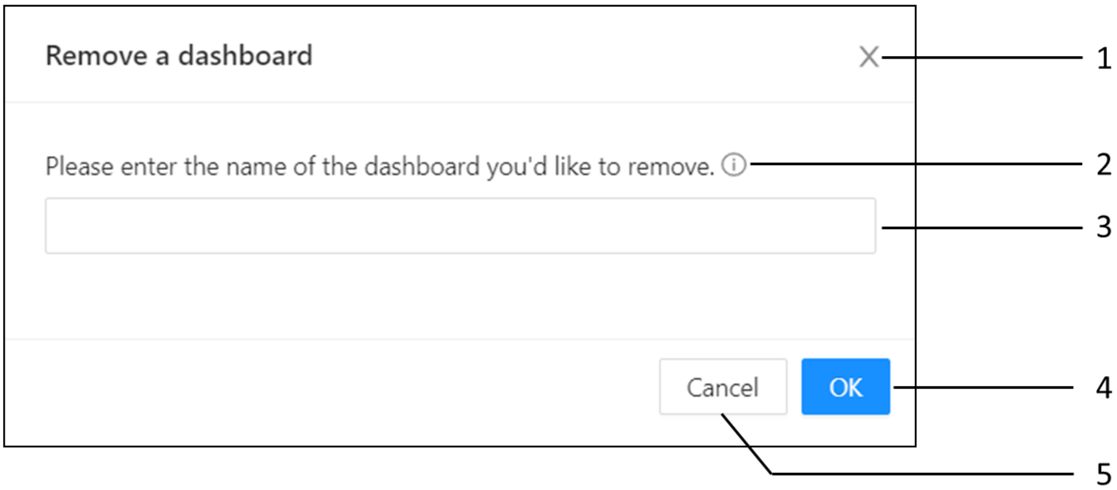

Figure 4.13: The *Remove a dashboard* prompt

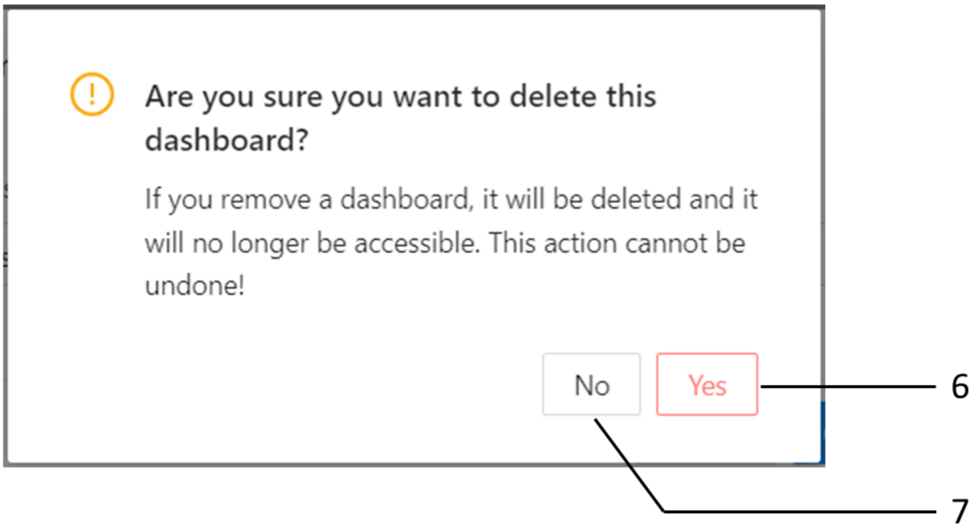

Figure 4.14: Deleting a dashboard warning prompt

Formal Description:

	Operation	Description	Result
1	Cancel deletion	Click on the *X* button at the top right of the *Remove a Dashboard* prompt	The prompt is closed without deleting the dashboard
2	View information about the *Remove a Dashboard* prompt	Hover over the info button	The details of the prompt will be displayed
3	Enter the name of the dashboard to be deleted	Click on the text field and fill in the name of the dashboard to be deleted	The text field displays the entered dashboard name
4	Confirm dashboard name	Click on the *OK* button at the bottom of the prompt	A warning prompt is displayed asking for the user's confirmation to delete the dashboard
5	Cancel deletion	Click on the *Cancel* button at the bottom of the prompt	The prompt is closed without deleting the dashboard
6	Confirm deletion	Click on the *Yes* button at the bottom of the warning prompt	The *Remove a Dashboard* prompt and confirmation prompt are closed, the dashboard is deleted, and the dashboard name is removed from the dashboard names dropdown menu. If the user deleted the dashboard that they are currently viewing, the user will be taken to the homepage
7	Cancel deletion	Click on the *No* button at the bottom of the warning prompt	The confirmation prompt to delete the dashboard is closed without deleting the dashboard

Examples:

	Operation	Example
1	Cancel deletion	***user1*** does not delete any dashboard
2	View information about the *Remove a Dashboard* prompt	***user1*** would views information on the conditions to delete a dashboard ***Purchase Information***
3	Enter the name of the dashboard to be deleted	***user1*** enters the name ***Purchase Information*** in the text field to delete the **Purchase Information** dashboard
4	Confirm dashboard name	***user1*** confirms deleting the dashboard ***Purchase Information***
5	Cancel deletion	textit**user1** does not delete any dashboard
6	Confirm deletion	***user1*** confirms the deletion of the dashboard ***Purchase Information***
7	Cancel deletion	***user1*** does not delete the dashboard ***Purchase Information***

Possible Errors:

Error	Cause(s)
This is not a valid dashboard	The user enters a name in the remove-dashboard text field that does not exist

Related References: 4.4.1

4.5 | Edit Mode

4.5.1 | Edit mode sidebar

Functional Description: After creating or viewing a dashboard, users can make changes to the dashboard by adding different elements. These changes can be made by entering a different mode known as edit mode. While a user is viewing a dashboard, they can click on the change-mode button that appears in the header. By clicking this button, the user will enter edit mode. Once they have entered edit mode, the edit mode sidebar will appear where they can add static and grid view elements, save their dashboard, and download the dashboard's configuration file as a JSON object (Figure 4.15).

Cautions and Warnings: None.

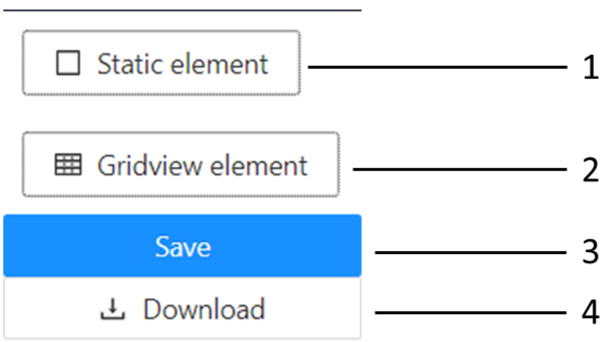

Figure 4.15: The edit mode sidebar

Formal Description:

	Operation	Description	Result
1	Add static element	Drag the static element button to the working area	The static element edit menu appears
2	Add grid view element	Drag the grid view element button	The query edit menu appears
3	Save changes	Click on the *Save* button	The current dashboard configuration is saved and a confirmation of the save is displayed
4	Download dashboard configuration file	Click on the *Download* button	The current dashboard configuration is downloaded as a JSON file

Examples:

	Operation	Example
1	Add static element	***user1*** adds a static element containing the text ***Inventory Information - June 2022*** to the dashboard ***Inventory Information***
2	Add gridview element	***user1*** adds a grid view with data from the columns ***Product Name*** and ***Quantity*** from the ***Inventory*** base table to the dashboard ***Inventory Information***
3	Save changes	***user1*** saves the dashboard ***Inventory Information*** for future reference
4	Download dashboard configuration file	***user1*** downloads the dashboard configuration file of the dashboard ***Inventory Information*** for other users to view the dashboard details

Possible Errors: None.

Related References: 4.2.2, 4.5.2, 4.5.3

4.5.2 | Add a static element

Functional Description: After a dashboard has been created and the user is in edit mode, the user can add different application elements to the dashboard working area. Application elements are of two types - static and grid view. To add a static element, a user can drag the static element button to the working area. Once the static element has been dragged to the desired point on the working area, the static element edit menu appears on the screen (Figure 4.16). The user then fills in the contents of the static element, and on saving, the element will be added to the working area. The content of a static element can be either text or a media link. Media will only be rendered as the correct type if one media link has been entered. Unsupported links will be displayed as text in the static element.

Cautions and Warnings: None.

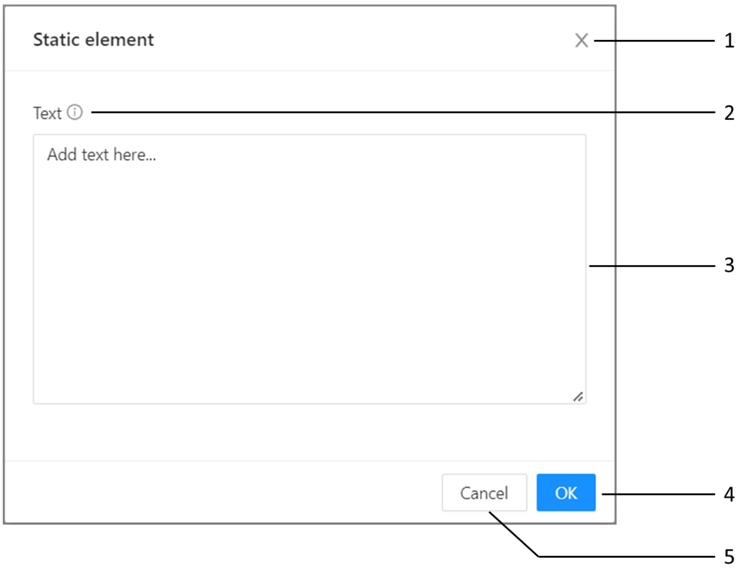

Figure 4.16: The edit menu for a static element

Formal Description:

	Operation	Description	Result
1	Cancel changes	Click on the *X* button at the top of the static element edit menu	No changes will be saved and the edit menu will close. The static element added will contain the placeholder text *Add text here...*
2	View information about the static element edit menu	Hover over the info button	The details of the input field for the static element will be displayed
3	Enter input for static element	Click on the input text field and fill in the input for the static element. The input can be either text or a URL	The text field will display the input entered by the user
4	Save changes	Click on the *OK* button at the bottom right of the static element edit menu	The new static element will be saved and added to the dashboard working area
5	Cancel changes	Click on the *Cancel* button at the bottom right of the static element edit menu	No changes will be saved and the edit menu will close. The static element added will contain the placeholder text *Add text here...*

Examples:

	Operation	Example
1	Cancel changes	***user1*** does not add any text in the newly created static element ***Inventory Information***
2	View information about the static element edit menu	***user1*** views the information on what can be added to a static element
3	Enter input for static element	***user1*** enters the text ***Inventory Information - June 2022*** in the input text field
4	Save changes	***user1*** adds the text ***Inventory Information - June 2022*** to the static element
5	Cancel changes	***user1*** does not add any text in the newly created static element

Possible Errors:

Error	Cause(s)
Broken image icon	The link to the image entered could not be found
These characters are not allowed	There is at least one double quotation mark or backslash in the input text field

Related References: 4.5.1

4.5.3 | Add a grid view element

Functional Description: The second type of application element that can be added to the working area is a grid view element. To add a grid view element, a user can drag the grid view element button to the working area. Once the grid view element has been dragged to the desired point on the working area, the grid view element edit menu appears on the screen (Figure 4.17). The user then fills in the contents of the grid view element, and on saving, the element will be added to the working area. The content of a grid view must be a GraphQL query that is syntactically correct.

Cautions and Warnings: None.

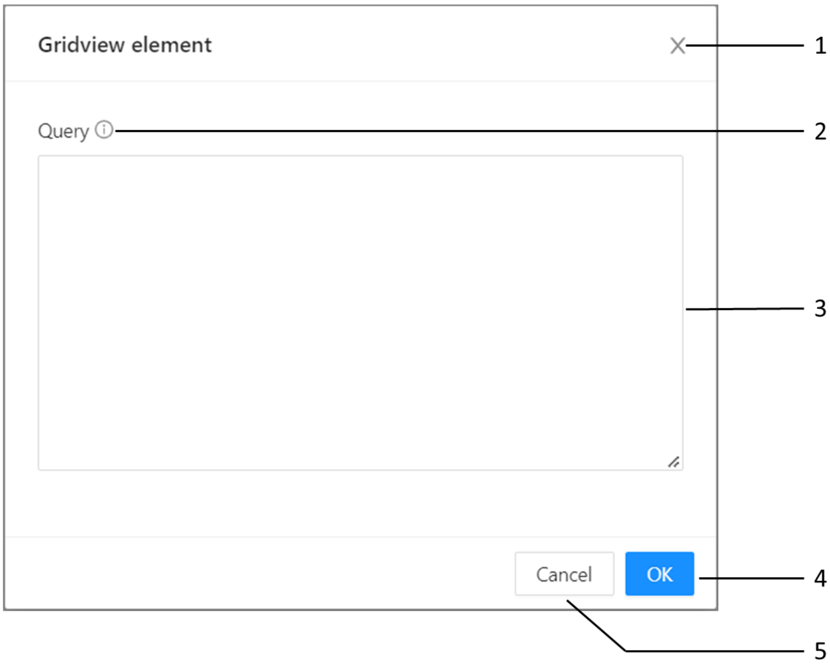

Figure 4.17: The edit menu for a grid view element

Formal Description:

	Operation	Description	Result
1	Cancel changes	Click on the *X* button at the top right of the gridview element edit menu	No changes will be saved and the edit menu will close. The gridview element added will contain the text *There is no data available*
2	View the input query information of the grid view element edit menu	Hover over the info button	The details of the input query for the grid view element will be displayed
3	Enter query for grid view element	Click on the text field and fill in the query for the static element	The text field will display the query entered by the user
4	Save changes	Click on the *OK* button at the bottom right of the grid view element edit menu	The new grid view element will be saved and added to the dashboard working area. The edit menu is closed
5	Cancel changes	Click on the *Cancel* button at the bottom right of the grid view element edit menu	No changes will be saved and the edit menu will close. The grid view element added will contain the text *There is no data available*

Examples:

	Operation	Example
1	Cancel changes	***user1*** does not add a query to the added gridview.
2	View the input query information of the grid view element edit menu	***user1*** views information on how to write add a query to the gridview.
3	Enter query for gridview element	***user1*** enters a query to obtain data from ***Product Name*** and ***Quantity*** in the ***Inventory*** base table to add to the dashboard ***Inventory Information***
4	Save changes	***user1*** saves the query to obtain data from the ***Inventory*** base table to add to the dashboard ***Inventory Information***
5	Cancel changes	***user1*** does not add a query to the added gridview.

Possible errors:

Warning	Cause(s)
The input does not seem a valid GraphQL query	The query entered is syntactically incorrect

Related References: 4.5.1

4.5.4 | Changing the physical appearance of application elements

Functional Description: Once users have added grid view or static elements, they can also change the physical appearance of the elements on their dashboard. Users can resize and delete elements by clicking on the *Resize* button and the *Trash Icon*.

Cautions and Warnings:

Error	Cause(s)
Application element cannot be resized	The new size of the element causes two elements to overlap on top of each other

Figure 4.18: Changing the physical appearance of an application element

Formal Description:

	Operation	Description	Result
1	Delete application element	Click on the *Trash Icon*	The warning prompt asking for confirmation of the deletion of the application element appears on the screen
2	Resize application element	Drag the mouse inwards or outwards while clicking and holding on the *Resize button*	The element will resize to the size set by the user

Examples:

	Operation	Example
1	Delete application element	***user1*** deletes the static element containing the text ***Inventory Information - June 2022*** in the dashboard ***Inventory Information***
2	Resize application element	***user1*** increases the size of the static element containing the text ***Inventory Information - June 2022*** in the dashboard ***Inventory Information***

Possible Errors: None.

Related References: 4.5.2, 4.5.3

4.5.5 | Delete application element

Functional Description: Once users have added gridviews or static elements, they can also remove them from their dashboard. This can be done by clicking on the delete element button at the top right of the respective element. A warning prompt will then be displayed on top of the screen asking for the user's confirmation to delete the application element. The user can confirm or cancel the deletion (Figure 4.19).

Cautions and Warnings: None.

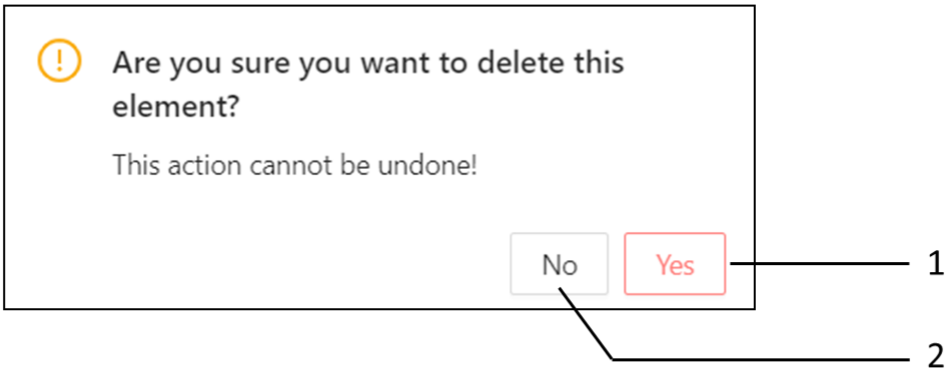

Figure 4.19: Deleting an application element warning prompt

Formal Description:

	Operation	Description	Result
1	Confirm deletion	Click on the *Yes* button at the bottom of the warning prompt	The element is deleted from the dashboard and the warning prompt is closed
2	Cancel deletion	Click on the *No* button at the bottom of the warning prompt	The warning prompt is closed without deleting the element

Examples:

	Operation	Example
1	Confirm deletion	***user1*** confirms the deletion of the application element containing the text ***Inventory Information - June 2022*** in the dashboard ***Inventory Information***
2	Cancel deletion	***user1*** cancels the deletion of the application element containing the text ***Inventory Information - June 2022*** in the dashboard ***Inventory Information***

Possible Errors: None.

Related References: 4.5.4

4.5.6 | Exit edit mode without saving

Functional Description: A user can exit edit mode if they no longer wish to edit their dashboard. This can be done by clicking on the change-mode button in the header. If the user has saved their changes, they will leave edit mode. If the user has not saved their changes, a prompt will be displayed asking the user if they wish to save their changes (Figure 4.20).

Cautions and Warnings: None.

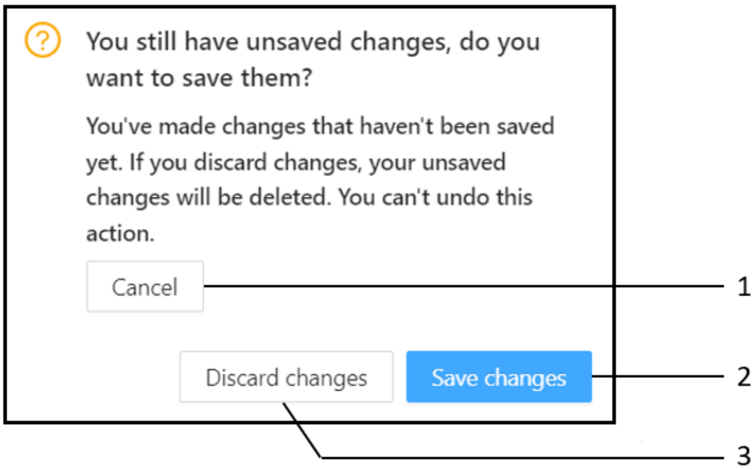

Figure 4.20: The prompt when exiting edit mode without saving dashboard changes

Formal Description:

	Operation	Description	Result
1	Cancel changes	Click on the *Cancel* button at the bottom of the warning prompt	The prompt is closed and the user remains in edit mode
2	Discard changes	Click on the *Discard changes* button at the bottom of the warning prompt	The new changes made to the dashboard will not be saved and the user will exit edit mode. They will return to the previously saved version of that dashboard
3	Save changes	Click on the *Save changes* button at the bottom of the warning prompt	The new changes made to the dashboard are saved and the user will exit edit mode. A confirmation message regarding the newly saved dashboard layout will be displayed

Examples:

	Operation	Example
1	Cancel changes	***user1*** cancels their exit from edit mode and stays in edit mode to make more changes to the dashboard ***Inventory Information***
2	Discard changes	***user1*** exits edit mode as they no longer want to edit the dashboard ***Inventory Information*** and does not save the new changes that were made to the dashboard
3	Save changes	***user1*** exits edit mode as they no longer want to edit the dashboard ***Inventory Information*** and saves the new changes made to the dashboard

Possible Errors: None.

Related References: 4.2.2

A | Error messages and recovery procedures

This section contains detailed descriptions of all error messages the user might encounter while interacting with MyUI and their respective recovery procedures.

A.1 | Login errors

Error Message: *Please input your username!*
Diagnosis: The user did not enter a username when logging in to MyUI.
Recovery Procedure: The user has to enter a username.

Error Message: *Please input your password!*
Diagnosis: The user did not enter a password when logging in to MyUI.
Recovery Procedure: The user has to enter a password.

Error Message: Login page refreshes.
Diagnosis: The user entered a correct username and an incorrect password.
Recovery Procedure: The user has to enter the correct password.

Error Message: Login error message: "Error localhost:3000".
Diagnosis: The user entered an incorrect username.
Recovery Procedure: The user has to head back to the login page and enter the correct username.

A.2 | Grid view errors

Error Message: *This is a required field*
Diagnosis: The user did not fill in a required text field when adding a new row in a grid view.
Recovery Procedure: The user has to fill in all required fields.

Error Message: *<colum name> can not be empty*
Diagnosis: The user did not fill in a required text field when editing a row in a grid view.
Recovery Procedure: The user has to fill in all required fields.

Error Message: *You are not allowed to edit this column*
Diagnosis: The user tries to edit a column in a grid view to which he does not have update permission.
Recovery Procedure: The user has to insert the old value of the column.

Error Message: *Foreign key violation*
Diagnosis: The user inputs a foreign key that does not exist
Recovery Procedure: The user has to insert a value that exists.

Error Message: Base table not loading in the working area.
Diagnosis: At random, the selected base table does not render in the working area and a loading animation is shown indefinitely.
Recovery Procedure: The user has to refresh the page.

A.3 | Dashboard errors

Error Message: *Sorry, you cannot use this name*
Diagnosis: The user entered an invalid dashboard name in the *Name* text field when adding a new dashboard.
Recovery Procedure: The user has to enter a valid unique name for the new dashboard.

Error Message: *This is not a valid dashboard*
Diagnosis: The user entered an invalid dashboard name in the *Name* text field when deleting a dashboard.
Recovery Procedure: The user has to enter a valid dashboard name in the text field.

Error Message: Uploading dashboard file with the wrong structure

Diagnosis: The user uploads a file that is not in .JSON format.
Recovery Procedure: No recovery, contact system administrator.

A.4 | Application elements errors

Error Message: *These characters are not allowed*
Diagnosis: The user enters quotation marks (") or backslashes (\) to the text field of a static element.
Recovery Procedure: The user has to delete the quotation marks and backslashes from the text field.

Error Message: *The input does not seem a valid GraphQL query*
Diagnosis: The user enters a syntactically incorrect GraphQL query to the text field of a grid view element.
Recovery Procedure: The user has to enter a GraphQL query with valid semantics.

Error Message: *There is no data available*
Diagnosis: The user enters a GraphQL query that requests a table or column that does not exist.
Recovery Procedure: The user has to enter a GraphQL query that requests valid tables and columns.

Error Message: Broken image icon.
Diagnosis: The user enters a URL of a media that cannot be found.
Recovery Procedure: The user has to enter a valid URL in the static element text box.

A.5 | Hasura error

Error Message: Hasura error message returned.
Diagnosis: The user performs an action that causes Hasura to return an error.
Recovery Procedure: The user has to check Hasura documentation.

B | Glossary

Term	Definition
Application element	A component of the front-end of the application. This includes all elements the user sees on their screen when using the application, such as text and images.
Base tables	A set of pure tables visualized in MyUI by default. There is precisely one base table for each table in the database that the user currently logged in to MyUI has access to. Each base table is visualized on a dedicated page via a grid view.
Dashboard	A particular page of which the contents and layout are rendered based on the stored preferences for a particular user (Dashboard Configuration).
Dashboard configuration	The set of preferences constructed in MyUI used to render a dashboard.
Grid view	A table data representation element.
Hasura	Hasura GraphQL Engine. It is a web application server that generates APIs instantly and stores the data access permissions of the users.
Homepage	The page of the application the user sees when logging in to MyUI.
Invalid dashboard name	A dashboard name that is non-unique, contains (double) quotation marks ("), contains backslash (\) or is equal to *dashboardAdd* or *dashboardDelete*.
Permission profile	Set of permissions to access information in the database assigned to a specific role.
Pure table	A table whose structure is contained in one of the tables on the database and where each row can be traced back to its original record in the database. In essence, the column names of this table are a subset of the column names of one of the tables in the database, and the table includes the primary key of the database.
Static element	An application element that contains static data. Examples include text fields, images, or headers/footers. These elements do not interact with the contents of the database or the application.
Working area	Area of the UI where all application elements are added, excluding the sidebar and header.

C | Index