**Roles**

* This section includes the tab selection between Users, User group and roles and then station type selection dropdown, a search-bar to search for a specific role and the list of roles in card form.
* On clicking user or user group tab, the URL will change to those pages and the page will be redirected.
* On clicking the Roles tab from other pages, the user will be redirected to the roles page.
* Two APIs will be called for the initial data load for category list and role list.
* API for the station category options is: **/stations/options**. This API gets the category and subcategory and is stored in the state in **category\_list**.
* API for getting Roles list is: . And it is stored in **role\_list.**

1. Category / subcategory selection

* The data will be stored in **category\_selected** and **sub\_category\_selected** in state.
* The default value selected is the category and subcategory in the URL **<Project URL>?cat=2&subcat=[].**
* The filter is a multi-select type, i.e. multiple categories and subcategories can be selected.
* The filter is a multi-select type, i.e. multiple categories and subcategories can be selected.
* If no category or subcategory is given in the URL, then the default value selected is Pump Station under Flood.
* On clicking a category, that category is selected along with all the subcategories under it.
* If a subcategory is clicked, if it is already unchecked, then it will be checked, and if already checked, it will uncheck.
* Every time the subcategory or category selected is changed, it is reflected in the URL and accordingly the stations in the table are displayed.
* The **role\_list** is filtered according to the filters applied and stored in **filtered\_roles.**

1. Search

* The result after applying filters can be searched for by Role name or rule.
* The searched text is stored in the key **table\_search** in the state.
* The list after the search from **filtered\_roles** is stored in **searched\_roles.**
* The searched key is reflected in the URL and the result of the search is displayed in the table.
* URL: **<Project URL>settings/roles/?cat=[2]&subcat=[]&search=demo.**

1. Add new role
   * On clicking the “create new role” button, the URL is updated and a popup appears to insert information about the role such as role name, description, and access rules.
   * The URL is: **<Project URL>settings/roles/add/?cat=[2]&subcat=[]**
   * Multiple rules can be selected for a group.
   * If the master access is selected, then all the rules are selected and in API ‘\*’ is sent.
   * The states for the data are:
     + selected\_role\_name
     + selected\_role\_description
     + selected\_role\_access\_rules
   * On saving the information, the data is posted to the API: **stations/groups/new**. And if the data is stored in the data base, we get a response with the group id of the newly created group.
   * A button loader is shown in the submit button while waiting for a response.
   * In successful response, ID and time is received and the new data is pushed to the existing **role\_list** in state.
2. Role listing in card form

* The card shows the role name, info button to view description, menu button (three dots), created by, creation time and rules under that role.
* On clicking the menu button (three dots), option menu opens to delete or edit the group.
* The delete button will be disabled for some roles which are not delete-able.
* On hover over the non-delete able button, a tooltip will open to show that the role is not delete-able.
* On clicking the delete button from the option, the URL will be updated and a confirmation popup will open to confirm the delete. On confirming, an API to delete the role will be called and on success response, the group will be deleted. On cancelling it, the role will not be deleted.
* A button loader will be shown while waiting for the response.
* URL to open the delete confirmation popup:
  + **<Project URL>settings/roles/:role\_id/delete/?cat=[2]&subcat=[]**
* On clicking the edit button from the menu, the URL will be updated and a popup will open to edit the station group information such as role name, description and rules under that role.
* URL to open edit modal:
  + **<Project URL>settings/roles/:role\_id/edit/?cat=[2]&subcat=[]**
* After modifying the information and clicking the save button, an API is called to post the data. API: **/stations/groups/:group\_id/edit**.