

Search Property Tax

Search Property

Objective

To provide users facility to search properties in the application by providing various search criteria. Search activity can be performed in two ways.

- Search by Property
- Search by Application

Search by Property

User can search properties by providing all mandatory information along with unique Property ID. When user has entered Unique property id then only one record will be displayed if the record associated with the property id exists. Tenant Id is a mandatory field to enter while searching properties. All the fields have pattern validation rule.

Reset operation clears all the data entered in the screen.

Search by Application

This screen provides facility to search property by providing Application no in search criteria. Few fields are common in both the screens like Owner mobile number and Unique Property Id. All the fields in the screen have pattern validation rule.

Reset operation clears all the data entered in the screen.

Technical Implementation Details

All the screens are developed under the egov-pt dev package.

<https://github.com/egovernments/frontend/tree/master/web/rainmaker/dev-packages/egov-pt-dev> - Connect to preview

It follows egov-ui framework principles. All the screens have their configuration in the egov-pt dev package folder. Property search configuration listed in the below file.

<https://github.com/egovernments/frontend/blob/master/web/rainmaker/dev-packages/egov-pt-dev/src/ui-config/screens/specs/pt-mutation/propertySearch.js>

Search screen is divided into two tabs. The configurations for these two tabs are written in property-search-tabs.js file under pt-mutation folder. Tabs internally refer to the Tabs custom component written in rainmaker-pt module. It has a search resource folder which consists of files that has code for calling search API, validating search criteria, building search table etc.

<https://github.com/egovernments/frontend/tree/master/web/rainmaker/dev-packages/egov-pt-dev/src/ui-config/screens/specs/pt-mutation/searchResource> - Connect to preview

MDMS data

Only one MDMS call is made to get the tenantID data to fill the locality dropdown. Below is the snippet from propertySearch.js file

Localization

Localization keys are added under the '*rainmaker-pt*' locale module. In future if any new labels are implemented in the Property search screen that should also be pushed in the locale DB under *rainmaker-pt* locale module. Below is the example of few locale labels.

```
{
  "code": "PT_ACKNOWLEDGEMENT_ID",
  "message": "Application No:",
```

```

      "module": "rainmaker-pt",
      "locale": "en_IN"
    },
    {
      "code": "PT_OWNER_MOBILE_NO",
      "message": "Owner Mobile No.",
      "module": "rainmaker-pt",
      "locale": "en_IN"
    },
  ],

```

Property Search Table

Objective

Property search table is created to display the search results in a tabular format. Property search table contains various columns as per product requirements. Table Columns can be configured from the Table component configuration written in source file.

Table has few features like sorting, pagination, table search, rows limit. All these features are configurable. Default no of rows in the table configured as 10.

Technical Implementation Details

Two table components are created; one for Property search and one for Application search. These two components imported in the propertySearch.js file and used inside Property Search page configuration.

Each table configuration internally uses the custom Table component written in custom-molecules-local folder inside rainmaker-pt module.

<https://github.com/egovernments/frontend/blob/master/web/rainmaker/dev-packages/egov-pt-dev/src/ui-config/screens/specs/pt-mutation/searchResource/searchResults.js>

Table body initializes with the Property search page initialization. By default tables are hidden. Once the search operation completed successfully then all the search records formatted into to expected tabular format and supplied to the table by making Table as visible.

Below is the file link.

<https://github.com/egovernments/frontend/blob/master/web/rainmaker/dev-packages/egov-pt-dev/src/ui-config/screens/specs/pt-mutation/functions.js>

Property ID and Application no in the search table are made as hyperlink. On click of Property ID user navigates to Property Details page where all the data related to Property displayed.

On click of Application no also it navigates to the Property details page. Only difference here is the further options to provide to the user on the Property details page.

If the File Status in InWorkflow, workflow operations can be performed from the File details page if the User is Authorized to do so. All other operations such as Transfer Ownership or Assess Property will be blocked.

MDMS call

No table specific MDMS call.

Localization Implementation

Locale keys for table headers are pushed in the '*rainmaker-common*' locale module. These locale keys are available in the localStorage. We map the Table header text to the locale keys obtained from locale DB.

API Call Role Action mapping

S. No.	API	Action id	Roles
1	/access/v1/actions/mdms/_get	870	CSR, GRO,DGRO,EMPLOYEE,CITIZEN,PGR-ADMIN,EMPLOYEE_FINANCE,TL_APPROVER, TL_CEMP,SUPERUSER,COLL_OPERATOR,BAN K_COLL_OPERATOR,CSC_COLL_OPERATOR,D MA OFFICE OFFICER,ULB OFFICER,COMMISSIONER,

2	/egov-mdms-service/v1/_search	954	LOA_CREATOR,SUPERUSER,WO_CREATOR,AE_CREATOR,WORKS_MASTER_CREATOR,
3	/user/_search	604	SUPERUSER,SYS_INTEGRATOR_FINANCE,SYS_INTEGRATOR_WATER_SEW,EGF_BILL_CREATOR,EMPLOYEE,EMPLOYEE_FINANCE,
4	/localization/messages/v1/_search	1531	SUPERUSER,EMPLOYEE,CITIZEN,GRO,DGRO,
5	/property-services/property/_search	1897	CITIZEN,PT_CEMP,PT_DOC_VERIFIER,PT_FIELD_INSPECTOR,PT_APPROVER