Router set:

\src\components\dictionary\dictionary.js

import createHistory from "history/createBrowserHistory";

const history = createHistory({

    forceRefresh: false

});

    componentDidMount = () => {

        this.props.getTenant(this.state.tenantSelected, true,"");

        history.push("/dictionary"+"/"+this.state.tenantSelected)

    };

    componentWillReceiveProps(nextProps){

        if(!nextProps.SubMenuStatus){

            console.log("if")

            this.setState({

                showStatus:false,

                menuData:nextProps.appsList && nextProps.appsList[this.state.startIndex] && nextProps.appsList[this.state.startIndex].opan.split(" ").join("")

            },()=>{

                history.push("/dictionary"+"/"+this.state.tenantSelected+"/"+this.state.menuData)

            });

        }else if(nextProps.SubMenuStatus === true){

            console.log("else")

            this.setState({

                showStatus:true,

                menuData:nextProps.appsList && nextProps.appsList[this.state.startIndex] && nextProps.appsList[this.state.startIndex].opan.split(" ").join("")

            },()=>{

                history.push("/dictionary"+"/"+this.state.tenantSelected+"/"+this.state.menuData+"/"+this.state.subMenuData)

            });

        }

    }

    handleChangeType = tenantSelected => {

        this.props.getTenant(tenantSelected.value, false,tenantSelected);

        let selectUrl = tenantSelected.value ? tenantSelected.value.split(" ").join("") :"";

        let subMenu = this.props.appsList && this.props.appsList[0].opan.split(" ").join("");

        this.setState({ tenantSelected: selectUrl,showStatus:false },()=>{

            this.props.subMenuChange(false);

            history.push("/dictionary"+"/"+selectUrl+"/"+subMenu)

        });

    ChangeRouter=(i,data)=>{

        this.setState({

            menuData:data,

            startIndex:i,

            showStatus:false

        },()=>{

            this.props.subMenuChange(false);

        });

        history.push("/dictionary"+"/"+this.state.tenantSelected+"/"+data)

    }

    ChangesubMenu=(data)=>{

        this.setState({

            subMenuData:data,

            showStatus:true

        },()=>{

            this.props.subMenuChange(true);

        })

        history.push("/dictionary"+"/"+this.state.tenantSelected+"/"+this.state.menuData+"/"+data)

    }

    getTabsDetails=(data)=>{

        this.props.getSubMenuDetails(data);

        //history.push("/dictionary"+"/"+this.state.tenantSelected+"/"+this.state.menuData+"/"+this.state.subMenuData+"/"+data)

    }

Tab data show:

src\components\dictionary\dictionary.js

line no: 154

                <Col xs={12} sm={12} md={8} lg={10} xl={10} className="table-view">

                    {this.state.showStatus ? ( <div><Tabs getTabsDetails={this.getTabsDetails} TabList={this.props.TabList}/><DatasetList/></div>):(<DefaultTable appList={this.props.appsList}/>)}

                </Col>

    getTabsDetails=(data)=>{

        this.props.getSubMenuDetails(data);

        //history.push("/dictionary"+"/"+this.state.tenantSelected+"/"+this.state.menuData+"/"+this.state.subMenuData+"/"+data)

    }

src\action\TableAction.js

export function getSubMenuDetails(data) {

    //console.log("dasd"+datasetList.mQueryLink)

    let index = 0;

    return function(dispatch, getState) {

        return TableApi.getSubMenuDetailsApi(data)

        .then(response => {

            dispatch({

                type: "GET\_SUB\_MENU\_TABLE\_DATA",

                dataList:datasetList.envInfo[index].tabs[data],

                TabList:Object.keys(datasetList.envInfo[index].tabs)

            });

        })

        .catch(error => console.log(error))

    }

}

src\reducer\TabelReducer.js

export default function TabelReducer(state="" ,action) {

    switch (action.type) {

        case "GET\_TABLE\_DATA":

            return { ...state,

                     tenantList: action.tenantList ? action.tenantList:"",

                     appsList:action.appsList ? action.appsList :"",

                     tenantSelected:action.tenantSelected ? action.tenantSelected:"",

                     defaultStatus:action.defaultStatus

                   };

        case "GET\_SELECTED\_DATA":

            return { ...state,

                tenantSelected: action.tenantSelected ? action.tenantSelected:"",defaultStatus:action.defaultStatus};

        case "CHANGE\_STATUS":

            return{

                ...state,

                tableStatus:action.status,

                CurrentSelect:action.value

            }

        case "CHANGE\_SUB\_MENU\_STATUS":

            return{

                ...state,

                SubMenuStatus:action.status

            }

        case "GET\_SUB\_MENU\_TABLE\_DATA":

            return{

                ...state,

                dataList:action.dataList ? action.dataList:"",

                TabList:action.TabList ? action.TabList:""

            }

        default:

            return state;

    }

}

src\components\dictionary\tabs\tabs.js

import React, { Component } from 'react';

import { Row, Col } from 'reactstrap';

import { Link, NavLink } from 'react-router-dom';

import './tabs.scss';

class Tabs extends Component {

  constructor(props) {

    super(props);

    this.state = {

      tabList: [

        "Base","Aggregate","Cumulative","Since Launch","View","Others"

      ],

    };

  }

  getTabsDetails=(data)=>{

     this.props.getTabsDetails(data);

  }

  render() {

    return (

      <div className="tabs">

        <ul className="">{

            this.props.TabList && this.props.TabList.map((data,i) =>

            <li key={i} onClick={()=>{this.getTabsDetails(data)}}><Link>{data}</Link></li>)}

        </ul>

      </div>

    )

  }

}

export default Tabs;

src\components\dictionary\entityTable\datasetList\datasetList.js

import React, { Component, Fragment } from 'react';

import { Row, Col } from 'reactstrap';

import { Link, NavLink } from 'react-router-dom';

import './datasetList.scss';

import { FontAwesomeIcon } from '@fortawesome/react-fontawesome';

import { faBug } from '@fortawesome/free-solid-svg-icons';

import { faSearch } from '@fortawesome/free-solid-svg-icons';

import \* as TableAction from "../../../../action/TableAction";

import { connect } from "react-redux";

import { bindActionCreators } from "redux";

class DatasetList extends Component {

  constructor(props) {

    super(props);

    this.state = {

      listIndex: null,

      type:"Base",

      list:[

        {

          value:"da"

        }

      ]

    };

  };

  componentDidMount() {

      this.props.getSubMenuDetails(this.state.type);

  }

  render() {

    console.log('aaaaaa', this.state.list);

    return(

        <Row className="dataset-list-row">

        <Col xs={12} sm={12} md={4} lg={4} xl={4} className="dataset-list-col">

          <div className="search-bar">

        <FontAwesomeIcon icon={faSearch} className="search-input-icon"/>

        <input type="search" placeholder={this.props.name} className="search-input"/>

        </div>

        <div className="dataset-list">

        <ul>

          {

            this.props.dataList && this.props.dataList.map((data) =>

            <li><Link><span className="list-index">{this.state.listIndex + 1}</span>{data.datsetNm}</Link></li>)

          }

        </ul>

        </div>

        </Col>

        <Col xs={12} sm={12} md={4} lg={6} xl={6}>

          test

          </Col>

        </Row>

    );

  }

}

const mapStateToProps = state => {

  const { TabelReducer } = state;

    return {

      dataList:TabelReducer ? TabelReducer.dataList:[]

    };

};

function mapDispatchToProps(dispatch) {

    return bindActionCreators(Object.assign({}, TableAction), dispatch);

}

export default connect(

    mapStateToProps,

    mapDispatchToProps

)(DatasetList);

src\components\dictionary\dataset.js

export default {

  "apId": 98,

  "mId": 107,

  "mNm": "Account Services",

  "mDes": "Description of AccountServices",

  "mCrtDt": "2019-11-12",

  "mMdfdDt": "2019-11-12",

  "mUsage": "Usage of AccountServices",

  "mRef": "accountservices",

  "mQueryLink": "https://google.com",

  "envInfo": [

      {

          "env": "mr-storage-prod",

          "src": null,

          "tabs": {

              "Base": [

                  {

                      "id": 6932,

                      "datsetNm": "default.bl\_account\_terms\_events"

                  },

                  {

                      "id": 6934,

                      "datsetNm": "default.bl\_account\_update\_events"

                  },

                  {

                      "id": 7341,

                      "datsetNm": "default.bl\_mme\_assignment"

                  }

                ],

                "Aggregate": [

                    {

                        "id": 6632,

                        "datsetNm": "default.bl\_terms\_events"

                    },

                    {

                        "id": 6634,

                        "datsetNm": "default.bl\_update\_events"

                    },

                    {

                        "id": 6641,

                        "datsetNm": "default.bl\_assignment"

                    }

                  ],

                  "Cumulative": [

                    {

                        "id": 6732,

                        "datsetNm": "default.bl\_account\_events"

                    },

                    {

                        "id": 6734,

                        "datsetNm": "default.bl\_account\_events"

                    },

                    {

                        "id": 6741,

                        "datsetNm": "default.bl\_mme"

                    },                    {

                        "id": 6735,

                        "datsetNm": "default.bl\_events"

                    }

                  ],

                  "Since Launch": [

                    {

                        "id": 6832,

                        "datsetNm": "bl\_account\_terms\_events"

                    },

                    {

                        "id": 6834,

                        "datsetNm": "bl\_account\_update\_events"

                    },

                    {

                        "id": 6841,

                        "datsetNm": "bl\_mme\_assignment"

                    }

                  ],

                  "View": [

                    {

                        "id": 6432,

                        "datsetNm": "bl\_events"

                    },

                    {

                        "id": 6434,

                        "datsetNm": "bl\_events"

                    },

                    {

                        "id": 6441,

                        "datsetNm": "bl\_assignment"

                    }

                  ],

                "Others": [

                      {

                          "id": 7607,

                          "datsetNm": "default.mme\_empcomp\_account"

                      },

                      {

                          "id": 7608,

                          "datsetNm": "default.mme\_empcomp\_from\_ds"

                      }

                  ]

              }

          }

      ]

  }