**Fiori Documentation for Beginner**

Fiori UI Elements : [UI Elements (SAPUI5) | SAP Fiori for Web Design Guidelines](https://experience.sap.com/fiori-design-web/ui-elements-overview/#quick-access)

Playlist : [Complete playlist for Fiori UI5](https://youtube.com/playlist?list=PL6aK0ufxLukIS5kqHty3E--fnfHxHaGjQ&si=W8iDCvRcqGXIgYE6)

1. Create a header section with 3 buttons:

* Add attribute xmlns:tnt="sap.tnt" to the <mvc:View> tag.
* Then add the following code in the page :

<tnt:ToolHeader id="\_IDGenToolHeader1">

<Button id="\_IDGenButton1" icon="sap-icon://menu2" type="Transparent"/>

<ToolbarSpacer id="\_IDGenToolbarSpacer1"/>

<Text id="\_IDGenText2" text="Login Page"/>

<ToolbarSpacer id="\_IDGenToolbarSpacer2"/>

<Button id="\_IDGenButton2" text="Welcome, Jinoy" type="Transparent"/>

</tnt:ToolHeader>

2. Create a footer with 2 sections:

        <footer>

            <Bar id="\_IDGenBar1">

                <contentLeft>

                    <Text class="redtext" id="\_IDGenText1" text="copyright@jinoy"/>

                </contentLeft>

                <contentRight>

                    <Text id="\_IDGenText3" text="C S Psyco"/>

                </contentRight>

            </Bar>

        </footer>

3. Create a login page:

<content>

    <VBox id="\_IDGenVBox1" fitContainer="true" class="vbox">

    <form:SimpleForm id="addressForm" title="Login" editable="true" layout="ResponsiveGridLayout" xmlns:form="sap.ui.layout.form" class="custform">

        <Label text="User Name :" id="label0"/>

        <Input xmlns="sap.m" placeholder="Enter Your User name" id="input0"  />

        <Label text="Password :" id="label1"/>

        <Input xmlns="sap.m" placeholder="Enter Your Password" id="input1" type="Password" tooltip="Password Field"/>

        <Button id="\_IDGenButton3" text="Login"  type="Emphasized" class="customButton" press="alertButton()" />

    </form:SimpleForm>

    </VBox>

</content>

4. Get input username and password from point 3 and alert inputs.

* Go to your controller home\_view\_01.controller.js and add the following alertButton function:

sap.ui.define([

    "sap/ui/core/mvc/Controller"

],

function (Controller) {

    "use strict";

    return Controller.extend("project1.controller.home\_view\_01", {

        onInit: function () {

        },

        alertButton: function () {

            let username = this.getView().byId("input0").getValue();

            let password = this.getView().byId("input1").getValue();

            alert("Username : "+username+" / Password : "+password);

          }

        }

    });

});

5. Create a new view page and add a button to navigate.

* Add a new button on the current page (home page) to navigate to the next page on press.

<Button id="\_IDGenButton4" text="Go To List" class="customButton" press="onNavigateToList()" />

* Create new view page list\_view\_01.view.xml inside view folder.
* Inside the new view page paste the following:

<mvc:View xmlns:mvc="sap.ui.core.mvc" xmlns="sap.m" xmlns:tnt="sap.tnt" controllerName="project1.controller.home\_view\_01" displayBlock="true">

  <Page id="listPage" title="{i18n>appTitle}" >

        <Text id="\_IDGenText1" text="test page 2" />

    </Page>

</mvc:View>

* Now go to your controller and add the new onNavigateToList function.

onNavigateToList: function () {

            var oRouter = sap.ui.core.UIComponent.getRouterFor(this);

            oRouter.navTo("Routelist\_view\_01");

        }

* Then go to your manifest.json and add target(Targetlist\_view\_01) and route(Routelist\_view\_01) as follows:

"routes": [

        {

          "name": "Routehome\_view\_01",

          "pattern": ":?query:",

          "target": [

            "Targethome\_view\_01"

          ]

        },

        {

          "name": "Routelist\_view\_01",

          "pattern": "list",

          "target": [

            "Targetlist\_view\_01"

          ]

        }

      ],

      "targets": {

        "Targethome\_view\_01": {

          "viewType": "XML",

          "transition": "slide",

          "clearControlAggregation": false,

          "viewId": "home\_view\_01",

          "viewName": "home\_view\_01"

        },

        "Targetlist\_view\_01": {

          "viewType": "XML",

          "transition": "slide",

          "clearControlAggregation": false,

          "viewId": "list\_view\_01",

          "viewName": "list\_view\_01"

        }

      }

* Now click on the new Go To List button to navigate to the next page.

6. Add external style sheet to the Web App:

* Go to your index.html file and add the following line inside the head tag.

<link rel="stylesheet" type="text/css" href="css/style.css">

* Now open style.css inside the css folder and try adding new styles like the following:

.redtext{

    color:rgb(255, 0, 0);

    animation: animate 2s linear infinite;

}

.customButton{

    margin-top: 1rem;

}

.vbox{

    display: flex;

    justify-content: center;

    align-items: center;

}

.custform{

    border: 1px solid black;

    border-radius: 15px;

}

.sapUiRespGridSpanS12{

    width: 0;

}

@keyframes animate{

    0%{

        color:aqua;

    }

    50%{

        color:rgb(0, 255, 0);

    }

    100%{

        color:rgb(204, 0, 255);

    }

}

7. To align elements as rows and columns, we can use VBox and HBox.

E.g.:

        <VBox id="\_IDGenVBox2" class="d-flex justify-content-center align-items-center">

            <HBox id="\_IDGenHbox3" class="mt-2">

                <Text id="\_IDGenText1" text="Data Fetching Page" />

            </HBox>

            <HBox id="\_IDGenHBox1" class="mt-4">

                <Label text=" Order ID : " id="label3" width="10rem"/>

                <Label text="Data 1" id="label5"/>

            </HBox>

            <HBox id="\_IDGenHBox2" >

                <Label text=" Order Date : " id="label4" width="10rem"/>

                <Label text=" Data 2" id="label6"/>

            </HBox>

        </VBox>

8. Transfer data from one view page to another view page.

* To transfer data from home\_view\_01 to list\_view\_01, go to the controller and change the function to get username and pass it as a parameter to Routelist\_view\_01.

          onNavigateToList: function () {

            let username = this.getView().byId("input0").getValue();

            var oRouter = sap.ui.core.UIComponent.getRouterFor(this);

            oRouter.navTo("Routelist\_view\_01",{

                name: username

            });

        }

* Then go to manifest.json and change the pattern of Routelist\_view\_01 to receive the parameter here.

        {

          "name": "Routelist\_view\_01",

          "pattern": "list/{name}",

          "target": [

            "Targetlist\_view\_01"

          ]

        }

* The received parameter is then passed to the controller of the new view. But there both the views are having same controller. So, edit the home controller.
* On onInit function add the following lines:

   onInit: function () {

   var oRouter = sap.ui.core.UIComponent.getRouterFor(this);

   oRouter.getRoute("Routelist\_view\_01").attachPatternMatched(this.\_onObjectMatched, this);

   // Initialize the model

   var oModel = new sap.ui.model.json.JSONModel();

   this.getView().setModel(oModel);

   }

* Then add one more function in the same controller as follows:

 \_onObjectMatched: function (oEvent) {

            var user\_name = oEvent.getParameter("arguments").name;

            this.getView().getModel().setProperty("/username", user\_name);

        },

* You can now use the username variable on the second view page having the value of the username from the first page like as follows:

<HBox id="\_IDGenHBox1" class="mt-4">

          <Label text=" Order ID : " id="label3" width="10rem"/>

          <Label text="{/username}" id="label5"/>

       </HBox>

* Now click on Go To List button will pass the parameter to the second view page.
* If you need to pass one more parameter(password) then the pattern in manifest manifest.json will looks like,

      "pattern": "list/{name}/{pass}",

9. Create a custom popup by using another module called message box.

* Import the module to the controller,

sap.ui.define([

    "sap/ui/core/mvc/Controller",

    "sap/m/MessageBox"

],

* Pass the module to the controller,

function (Controller,MessageBox) {

    "use strict";

* Use the message box,

MessageBox.success("belloo");

10. To call and display a view page inside another view page,

    <mvc:XMLView viewName="project1.view.home\_view\_01" />

11. **Fragmnet:** Creation of fragment and calls the fragment on button press.

* Create a button in the current page to call the fragment

<Button id="\_IDGenText5" text="Policy Document" press="openPolicyDoc()"/>

* Create a new view page policyDoc.fragment.xml and add the following code in it.

<core:FragmentDefinition xmlns="sap.m" xmlns:core="sap.ui.core">

    <Dialog id="frag1" title="Policy Document">

        <Text id="\_IDGenText1" text="Lorem ipsum dolor sit amet"></Text>

        <beginButton>

            <Button id="\_IDGenButton5" text="close" press="closePolicyDoc()" />

        </beginButton>

    </Dialog>

</core:FragmentDefinition>

* Import Fragment module to the controller and pass it to the function

sap.ui.define([

    "sap/ui/core/mvc/Controller",

    "sap/m/MessageBox",

    "sap/ui/core/Fragment"

],

function (Controller,MessageBox,Fragment) {

    "use strict";

* Add 2 more functions in the controller.

        openPolicyDoc: function(){

            var oView = this.getView();

            if(!this.byId("frag1")){

                Fragment.load({

                    id:oView.getId(),

                    name: "project1.view.policyDoc",

                    controller:this

                }).then(function(oDialog){

                    oView.addDependent(oDialog);

                    oDialog.open();

                })

            }

            else{

                this.byId("frag1").open();

            }

        },

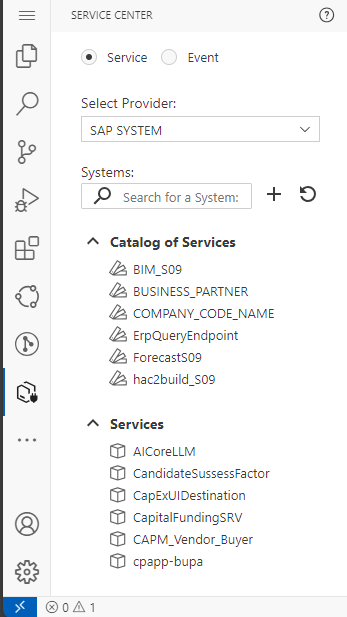
        closePolicyDoc: function(){

            this.byId("frag1").close();

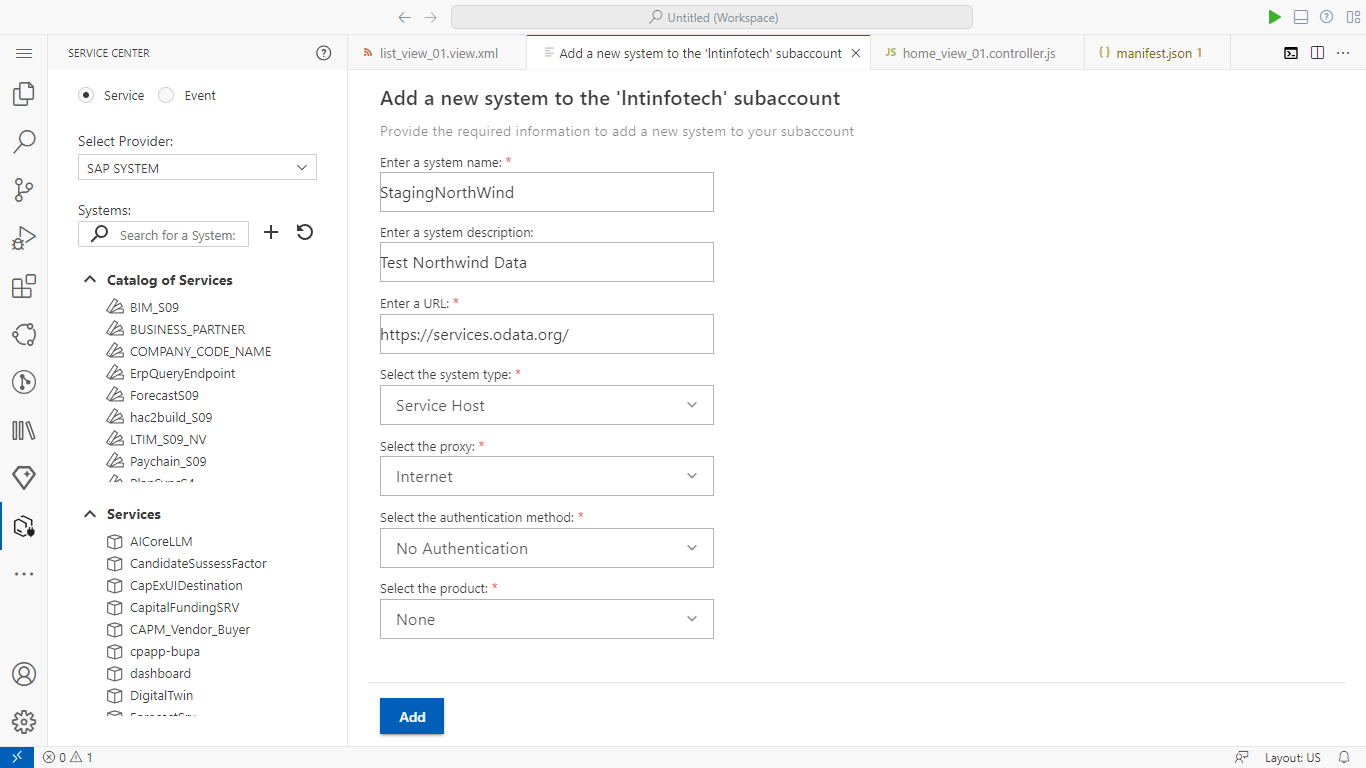
        }

12. Connecting the Northwind oData service in existing application.

* Go to additional Views->Service Center.



* Add a New System



Search: StagingNorthWind and open

Service Url: V2/Northwind/Northwind.svc/

* Click on Add to Project

13. Using Northwind data in View Page

* On view page, add the following:

  <List id="\_IDGenList" items="{/Products}">

  <StandardListItem id="\_IDGenStandardListItem1" type="Active" title="{ProductName}" />

  </List>

* And change the following to

  <Button id="\_IDGenButton" text="Hi, {userModel>/username}"></Button>

* Import     "sap/ui/model/json/JSONModel"
* Change init function and \_onObjectMatched function
* Initialize another model oUserModel since oModel will be used by oData by default.

       onInit: function () {

   var oRouter = sap.ui.core.UIComponent.getRouterFor(this);

   oRouter.getRoute("Routelist\_view\_01").attachPatternMatched(this.\_onObjectMatched, this);

            var oUserModel = new JSONModel();

            this.getView().setModel(oUserModel, "userModel");

        },

        \_onObjectMatched: function (oEvent) {

            var user\_name = oEvent.getParameter("arguments").name;

            var pass\_word = oEvent.getParameter("arguments").pass;

            var oUserModel = this.getView().getModel("userModel");

            oUserModel.setProperty("/username", user\_name);

            oUserModel.setProperty("/password", pass\_word);

        },

14. Login and logout

* Change the login/logout function to the following,

          onNavigateToList: function () {

            let username = this.getView().byId("input0").getValue();

            let password = this.getView().byId("input1").getValue();

            if(username == '' || password == ''){

                MessageBox.error("Invalid Username/Password");

            }

            else{

                let oModel = this.getView().getModel();

                oModel.read("/Employees", {

                    filters: [

                        new sap.ui.model.Filter("FirstName", sap.ui.model.FilterOperator.EQ, username),

                        new sap.ui.model.Filter("Extension", sap.ui.model.FilterOperator.EQ, password)

                    ],

                    success: function (oData) {

                        if (oData.results.length > 0) {

                            // Navigate to the next view if credentials are valid

                            sessionStorage.setItem("loggedIn", "true");

                            var oRouter = sap.ui.core.UIComponent.getRouterFor(this);

                            oRouter.navTo("Routelist\_view\_01", {

                                name: username,

                                pass: password

                            });

                        } else {

                            MessageBox.error("Invalid Username/Password");

                        }

                    }.bind(this),

                    error: function (oError) {

                        MessageBox.error("Failed to fetch data from Northwind service");

                    }

                });

            }

        },

        logout: function(){

            var oRouter = sap.ui.core.UIComponent.getRouterFor(this);

            MessageBox.confirm("Are you sure, you want to logout?", {

                actions: [MessageBox.Action.OK, MessageBox.Action.CANCEL],

                onClose: function(oAction) {

                    if (oAction === MessageBox.Action.OK) {

                        sessionStorage.removeItem("loggedIn");

                        oRouter.navTo("Routehome\_view\_01");

                    }

                }

            });

        },