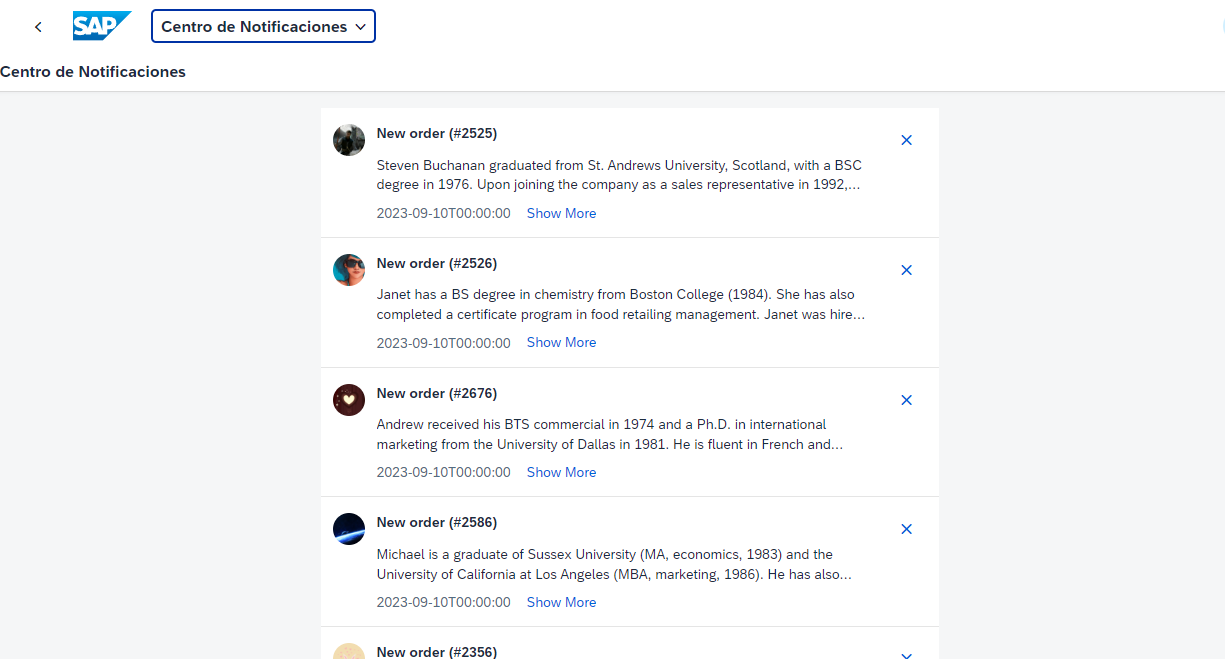
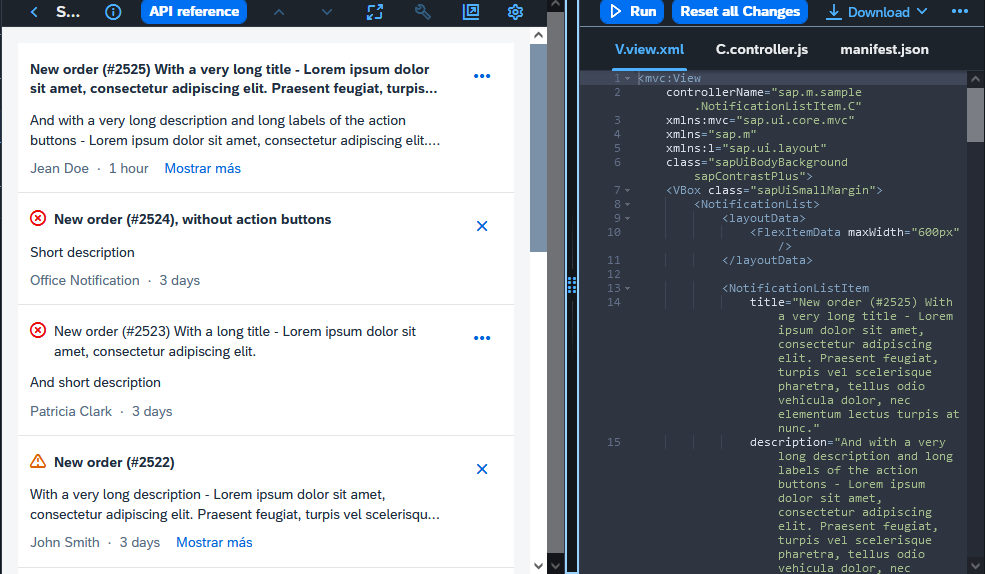
****

### Propuesta Ejercicio 7

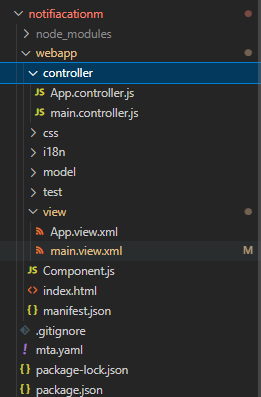
**Creación de la lista de Notification List Item (vista).**   
Ejercicio práctico: Listado de notificaciones. A continuación, se mostrarán los pasos para crear el siguiente programa con SAP FIORI también se utilizaron los componentes de UI5 y haciendo llamado a los servicios de Odata, Mock servers y REST API ****

En el siguiente link se obtuvo la vista el cual se modificó a conveniencia del ejercicio, donde se adapta a los datos que se van a mostrar.

Link : <https://sapui5.hana.ondemand.com/sdk/#/entity/sap.m.NotificationListItem/sample/sap.m.sample.NotificationListItem>



Ya creado el proyecto fiori, dentro de la carpeta de webapp se encuentra las subcarpetas View donde está la vista main.view.xml, controller donde está el archivo main.controller.js y en la raíz del proyecto el manifest.json en estos 3 es donde se van hacer las modificaciones



Main.view.xml

<mvc:View

    xmlns:imageeditor="sap.suite.ui.commons.imageeditor"

    controllerName="techcool.notifiacationm.controller.main"

    xmlns:mvc="sap.ui.core.mvc"

    displayBlock="true"

    xmlns="sap.m"

    height="100%">

    <Shell id="\_IDGenShell1">

        <App id="\_IDGenApp1" busy="{appView>/busy}" class="sapUiSizeCompact">

            <Page id="page" title="{i18n>title}">

                <VBox id="\_IDGenVBox1" class="sapUiSmallMargin">

                    <NotificationList

                        id="NotificationList"

                        items="{

                            path: 'employees>/',

                            templateShareable: true }">

                        <layoutData>

                            <FlexItemData id="\_IDGenFlexItemData1" maxWidth="50%" alignSelf="Center" />

                        </layoutData>

                        <NotificationListItem id="NotificationListItem"

                            title="{employees>Title}"

                            description="{employees>Description}"

                            showCloseButton="true"

                            datetime="{employees>BirthDate}"

                            unread="true"

                            priority="None"

                            close="onItemClose"

                            press=".onListItemPress"

                            authorPicture="{employees>image}"

                            authorAvatarColor="Accent8" />

                    </NotificationList>

                </VBox>

            </Page>

        </App>

    </Shell>

</mvc:View>

En el siguiente código se mostrar el controlador de la vista

main.controller.js

sap.ui.define([

        "sap/ui/core/mvc/Controller",

        "sap/ui/model/json/JSONModel",

        "sap/ui/model/Sorter",

        "sap/ui/model/Filter",

        "sap/ui/model/FilterOperator",

        "sap/ui/model/FilterType",

        "sap/m/MessageToast",

        "sap/m/MessageBox",

    ], function(Controller, JSONModel, Sorter, Filter, FilterOperator, FilterType, MessageToast, MessageBox){

        "use strict";

        var ListController =  Controller.extend("techcool.notifiacationm.controller.main", {

            onInit : function (evt){

                /\*var oJSONData = {

                    busy : false,

                    order : 0

                },

                oModel = new JSONModel(oJSONData);

                this.getView().setModel(oModel, "appView");\*/

                var oModel = this.getView().getModel("employees");

                this.getView().setModel(oModel, "appView");

            },

            onSearch : function () {

                const oModel = [

                    'ID',

                    'FirstName',

                    'LastName',

                    'Address',

                    'HomePhone'

                ];

                var oView = this.getView(),

                    sValue = oView.byId("searchField").getValue(),

                    oFilter = new Filter("Title", FilterOperator.Contains, sValue);

                oView.byId("notificationList").getBinding("items").filter(oFilter, FilterType.Application);

            },

            onItemClose: function (oEvent) {

                var oItem = oEvent.getSource(),

                    oList = oItem.getParent();

                oList.removeItem(oItem);

                MessageToast.show("Item Closed: " + this.\_getText("deletionSuccessMessage", Title));

            },

            \_getText : function (sTextId, aArgs) {

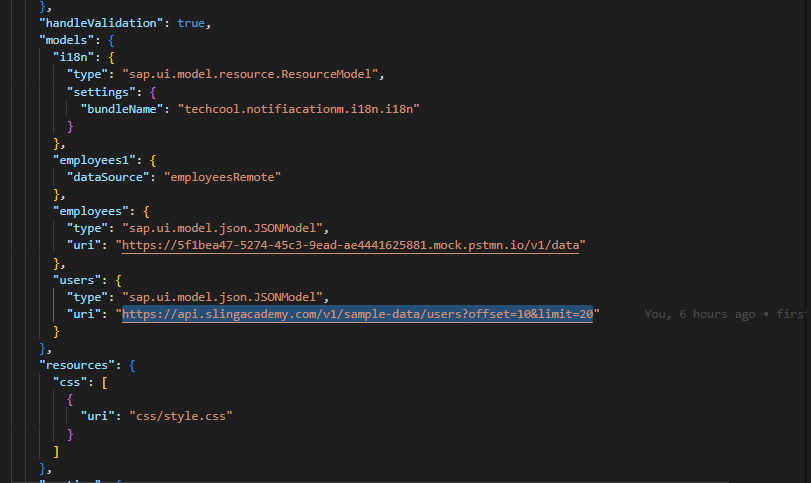
                return this.getOwnerComponent().getModel("i18n").getResourceBundle().getText(sTextId, aArgs);

            }

        });

    });

En el manifest.json se hace el llamado mock servers, odata o a la API



Links adicional  
  
Para hacer el group de las notificaciones

<https://sapui5.hana.ondemand.com/sdk/#/entity/sap.m.NotificationListGroup>