

Binding OData Solicitudes al servicio (Botón Buscar) - Primera Parte

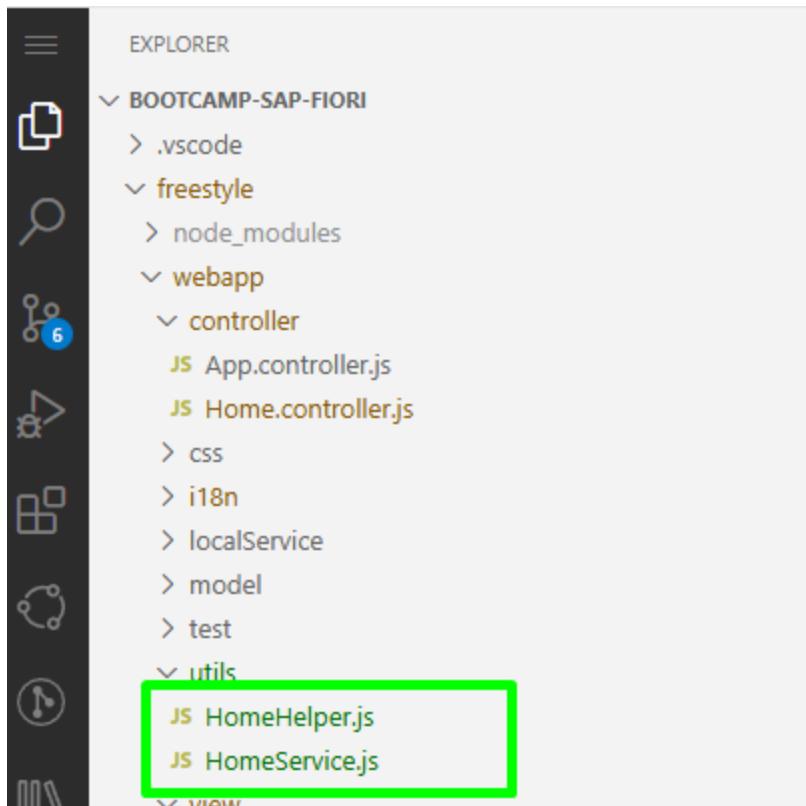
Agregamos un nuevo control a la vista Home, un botón:

JavaScript

```
<VBox id="vTest">
    <Text id="textPrueba" text="{i18n>textPruebai18n}"></Text>
    <Button id="btnBuscar" text="{i18n>textBuscar}" press="onPress" />
</VBox>
```

```
freestyle > webapp > view > Home.view.xml
You, 3 minutes ago | 1 author (You)
1 <mvc:View controllerName="com.bootcamp.sapui5.freestyle.controller.Home"
2     xmlns:mvc="sap.ui.core.mvc"
3     xmlns="sap.m">
4     <Page id="page" title="{i18n>title}">
5
6         <VBox id="vTest">
7             <Text id="textPrueba" text="{i18n>textPruebai18n}"></Text>
8             <Button id="btnBuscar" text="{i18n>textBuscar}" press="onPress" />
9         </VBox>
10
11     </Page>
12 </mvc:View>
```

Modificada la vista vamos a implementar la función en el controlador para ello vamos a hacer uso de Helper:



HomeService.js : En este Helper se deben implementar las llamadas al Servicio Odata, en este caso implementamos(función `readProducts`) la llamada a la Entidad **/Products** del servicio **Northwind**

```
JavaScript
sap.ui.define([
], function () {
    "use strict";

    return {
        readProducts: async function (oModel, oFilter) {
            const aRequestsPromises = [
                new Promise(function (resolve, reject) {
                    oModel.read('/Products', {
                        filters: oFilter,
                        success: resolve,
                        error: reject,
                    })
                }).bind(this))
            ];

            return Promise.all(aRequestsPromises);
        }
    };
});
```

```
        },
    }
});
```

HomeHelper.js: Esta librería es la intermedia entre los llamados a la base de datos(**HomeService.js**) y el controlador de la vista(para este caso **Home.controller.js**), la idea es que en este JS se puedan hacer transformaciones de datos o aplicar reglas de negocio en caso de ser necesario y no carguemos el controlador de la vista con demasiado código.

Tenemos 2 funciones implementadas:

init: Recibe el Odata de la aplicación para tener una referencia global y usarlo cada vez que se van a realizar llamados al Odata.

getDataProducts: Función que permite el llamado al Helper de Servicios y para este caso obtiene el listado de Productos.

```
JavaScript
sap.ui.define([
    "com/bootcamp/sapui5/freestyle/utils/HomeService",
], function (HomeService) {
    "use strict";

    return {
        init: function (oNorthwindModel) {
            this._oNorthwindModel = oNorthwindModel;
        },

        getDataProducts: async function() {
            let oFilters = [];
            return HomeService.readProducts(this._oNorthwindModel, oFilters);
        }
    );
});
```

Desde el archivo **Component.js** hacemos referencia a esta librería para inicializar el modelo de la aplicación con el Servicio configurado inicialmente y que podemos visualizar en el archivo **manifest.json**

Todo lo que está en **este color** fue lo que se agregó en el archivo Component.js:

JavaScript

```
sap.ui.define([
    "sap/ui/core/UIComponent",
    "com/bootcamp/sapui5/freestyle/model/models",
    "com/bootcamp/sapui5/freestyle/utils/HomeHelper"
], (UIComponent, models, HomeHelper) => {
    "use strict";

    return UIComponent.extend("com.bootcamp.sapui5.freestyle.Component", {
        metadata: {
            manifest: "json",
            interfaces: [
                "sap.ui.core.IAsyncContentCreation"
            ]
        },
        init() {
            // call the base component's init function
            UIComponent.prototype.init.apply(this, arguments);

            // set the device model
            this.setModel(models.createDeviceModel(), "device");

            // enable routing
            this.getRouter().initialize();

            this.setInitModel();
        },
        setInitModel: function () {
            HomeHelper.init(this.getModel());
        }
    });
});
```

```

manifest.json ×
freestyle > webapp > manifest.json > sap.app > dataSources > mainService
  "sap.app": {
    "sourceTemplate": {
      },
      "dataSources": {
        "mainService": {
          "uri": "/northwind/northwind.svc/",
          "type": "OData",
          "settings": {
            "annotations": [],
            "localUri": "localService/metadata.xml",
            "odataVersion": "2.0"
          }
        }
      }
    },
  }
}

manifest.json ×
freestyle > webapp > manifest.json > sap.ui5 > models > "" > dataSource
  "sap.ui5": {
    "models": {
      },
      "": {
        "dataSource": "mainService",
        "preload": true,
        "settings": {}
      }
    },
  }
}

```

Finalmente agregamos la función en el controlador que responde al Botón agregado en la vista Home:

The screenshot shows the SAP UI5 IDE interface. On the left is the Explorer view, which lists the project structure under 'BOOTCAMP-SAP-FIORI'. The file 'Home.controller.js' is selected and highlighted with a green box. The main area shows the code for 'Home.controller.js'.

```

sap.ui.define([
    "sap/ui/core/mvc/Controller",
    "com/bootcamp/sapui5/freestyle/utils/HomeHelper"
], (Controller, HomeHelper) => {
    "use strict";

    return Controller.extend("com.bootcamp.sapui5.freestyle.controller.Home", {
        ...
        onPress: async function(){
            let oDatos = await HomeHelper.getDataProducts();
        }
    });
});

```

Two specific parts of the code are annotated with green arrows:

- An arrow points to the import statement for 'HomeHelper': `"com/bootcamp/sapui5/freestyle/utils/HomeHelper"`.
- An arrow points to the call to `HomeHelper.getDataProducts()` in the `onPress` event handler.

```

JavaScript
sap.ui.define([
    "sap/ui/core/mvc/Controller",
    "com/bootcamp/sapui5/freestyle/utils/HomeHelper"
], (Controller, HomeHelper) => {
    "use strict";

    return Controller.extend("com.bootcamp.sapui5.freestyle.controller.Home", {
        ...
        onPress: async function(){
            let oDatos = await HomeHelper.getDataProducts();
        }
    });
});

```

Ejecutamos la aplicación y presionamos F12 y hacemos un Debug para ver la respuesta del servicio:

```

1 sap.ui.define([
2   "sap/ui/core/mvc/Controller",
3   "com/bootcamp/sapui5/freestyle/utils/HomeHelper"
4 ], (Controller, HomeHelper) => {
5   "use strict";
6
7   return Controller.extend("com.bootcamp.sapui5.freestyle.controller.Home", {
8     ...
9     ...
10    ...
11    ...
12    ...
13    ...
14    ...
15  });
}

```

```

1 sap.ui.define([
2   "sap/ui/core/mvc/Controller",
3   "com/bootcamp/sapui5/freestyle/utils/HomeHelper"
4 ], (Controller, HomeHelper) => {
5   "use strict";
6
7   return Controller.extend("com.bootcamp.sapui5.freestyle.controller.Home", {
8     ...
9     ...
10    ...
11    ...
12    ...
13    ...
14    ...
15  });
}

```

Object { length: 20 }
 0: Object { metadata: ..., Category: ... }
 1: Object { metadata: ..., Category: ... }
 2: Object { metadata: ..., Category: ... }
 3: Object { metadata: ..., Category: ... }
 4: Object { metadata: ..., Category: ... }
 5: Object { metadata: ..., Category: ... }
 6: Object { metadata: ..., Category: ... }
 7: Object { metadata: ..., Category: ... }
 8: Object { metadata: ..., Category: ... }
 9: Object { metadata: ..., Category: ... }
 10: Object { metadata: ..., Category: ... }

El resultado de lo anterior es que en la variable oDatos podemos visualizar el listado de Productos, aca otra vista por consola:

