

Navegación a una página de detalle

Desde la Tabla construida anteriormente vamos a agregar un evento a la fila para que al dar click se tomen los datos y se navegue a una página de detalle.

[https://services.odata.org/Northwind/Northwind.svc/Products\(1\)?\\$expand=Order_Details&\\$format=json](https://services.odata.org/Northwind/Northwind.svc/Products(1)?$expand=Order_Details&$format=json)

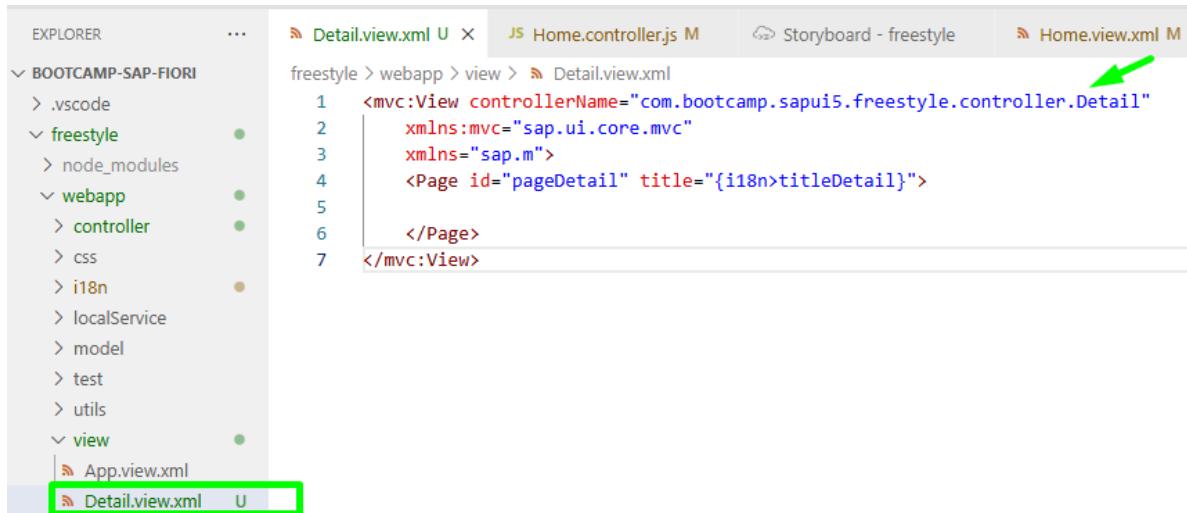
Pasos a realizar:

1. Crear una nueva vista, la llamaremos **Detail**

freestyle/webapp/view/Detail.view.xml

JavaScript

```
<mvc:View controllerName="com.bootcamp.sapui5.freestyle.controller.Detail"
    xmlns:mvc="sap.ui.core.mvc"
    xmlns="sap.m">
    <Page id="pageDetail" title="{i18n>titleDetail}">
        ...
    </Page>
</mvc:View>
```



La vista se le debe asignar su respectivo Controlador, por lo cual debemos agregarlo también, nombrarlo como Detail (**freestyle**/webapp/controller/Detail.controller.js)

```
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.Detail",
    {
        OnInit() {
        }
    });
});
```

```
EXPLORER ... JS Detail.controller.js U X Detail.view.xml U JS Home.controller.js M Storyboard - freestyle
BOOTCAMP-SAP-FIORI
    > .vscode
    < freestyle
        > node_modules
    < webapp
        < controller
            | JS App.controller.js
            | JS Detail.controller.js U
            | JS Home.controller... M
            > css
            > i18n
freestyle > webapp > controller > JS Detail.controller.js > sap.ui.define() callback
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.Detail", {
8         OnInit() {
9
10     }
11 });
```

2. Se debe configurar enrutamiento y la navegación a la nueva vista en el archivo **manifest.json**

The screenshot shows a code editor with several tabs at the top: manifest.json (Working Tree) M, App.view.xml, HomeService.js, metadata.xml, and manifest.json. The manifest.json file contains configuration for SAPUI5 routing. Changes are highlighted in green, indicating they have been staged or committed. The highlighted code defines routes for 'RouteHome' and 'detail', and targets 'TargetHome' and 'detail'. It also specifies view details like 'viewId', 'viewName', and 'viewLevel'.

```

46     "sap.ui5": {
79         "routing": {
80             "config": {
90                 },
91                 "routes": [
92                     {
93                         "name": "RouteHome",
94                         "pattern": ":?query:",
95                         "target": [
96                             "TargetHome"
97                         ]
98+                     }, You, a few seconds ago • Uncommitted changes
99+
100+                 {
101+                     "name": "detail",
102+                     "pattern": "detail/{ProductID}",
103+                     "target": [
104+                         "TargetHome",
105+                         "detail"
106                     ]
107                 ],
108                 "targets": {
109                     "TargetHome": {
110                         "id": "Home",
111                         "name": "Home"
112+                     },
113+                     "detail": {
114+                         "viewId": "detail",
115+                         "viewName": "Detail",
116+                         "viewLevel": 2
117                     }
118                 }
...

```

Como se visualiza en la configuración la vista Detail recibirá un parámetro, el **ProductID**. Este parámetro será tomado al seleccionar un ítem de la tabla que tenemos en la vista **Home**, como veremos a continuación.

JavaScript

```
{
    "name": "detail",
    "pattern": "detail/{ProductID}",
    "target": [
        "TargetHome",
        "detail"
    ]
}
```

```
"detail": {  
    "viewId": "detail",  
    "viewName": "Detail",  
    "viewLevel": 2  
}
```

3. En la vista **Home** Agregar un evento a la fila de la Tabla de Productos para que responda cuando el usuario presione un ítem.

```
<ColumnListItem id="ID_06" type="Active" press="onItemPress">
```

freestyle > webapp > view > Home.view.xml

```
1  <mvc:View controllerName="com.bootcamp.sapui5.freestyle.controller.Home"
4      <Page id="page" title="{i18n>title}">
11         <VBox id="vTest2"> You, 17 hours ago • 4_Binding_Local_Model
12             <Table id="idProductsTable"
16                 <columns>
29                     </columns>
30                     <items>
31                         <ColumnListItem id="ID_06" type="Active" press="onItemPress">
32                             <cells>
33                                 <ObjectIdentifier id="ID_07"
34                                     +title="#{ProductCollection>ProductID1}"
```

4. Modificar el Controller **Home** para agregar la función que tomará los datos de la fila seleccionada y Navegará a la Vista Detail:

```
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         OnInit() {
9             this.oRouter = this.getOwnerComponent().getRouter();
10        },
11    });
12});
```

```

15+     },
16+
17+     onItemPress: function(oEvent){
18+         let oSource = oEvent.getSource();
19+
20+         let oDatos = oSource.getBindingContext("ProductCollection").getObject();
21+
22+         this.oRouter.navTo("detail", {
23+             ProductID: oDatos.ProductID
24+         });
25+     }
26+ );
27 });

```

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", {
        OnInit() {
            this.oRouter = this.getOwnerComponent().getRouter();
        },

        onPress: async function(){
            let oDatos = await HomeHelper.getDataProducts();
            await HomeHelper.setProductModel(this, oDatos[0].results);
        },

        onItemPress: function(oEvent){
            let oSource = oEvent.getSource();

            let oDatos = oSource.getBindingContext("ProductCollection").getObject();

            this.oRouter.navTo("detail", {
                ProductID: oDatos.ProductID
            });
        }
    });
});

```

Ejecutamos la App para probar la navegación.

The screenshot shows a SAP Fiori application titled "Freestyle SAPUI5". The page displays a table of products with columns for Product ID, Name, Unit Price, Units In Stock, and Quantity Per Unit. The products listed are Chai, Chang, Aniseed Syrup, Chef Anton's Cajun Seasoning, Chef Anton's Gumbo Mix, and Grandma's Boysenberry Spread. Each row includes a detailed description of the quantity and packaging.

Product	Unit Price	Units In Stock	Quantity Per Unit
1 Chai	18.00 USD	39	10 boxes x 20 bags
2 Chang	19.00 USD	17	24 - 12 oz bottles
3 Aniseed Syrup	10.00 USD	13	12 - 550 ml bottles
4 Chef Anton's Cajun Seasoning	22.00 USD	53	48 - 6 oz jars
5 Chef Anton's Gumbo Mix	21.35 USD	0	36 boxes
6 Grandma's Boysenberry Spread	25.00 USD	120	12 - 8 oz jars

- Presionamos F12 y agregamos un breakpoint en la función `onItemPress`

The screenshot shows the SAPUI5 developer tools interface. The top navigation bar includes 'DU' and tabs for 'Elements', 'Console', 'Sources' (which is highlighted in green), 'Network', 'Performance', 'Memory', 'Application', 'Security', and 'Lighthouse'. Below the tabs is a code editor window displaying a portion of the 'Home.controller.js' file. The code includes imports for 'jquery-3.1.1.min.js', 'HomeHelper.js', and 'Home.controller.js'. It contains logic for handling item presses on a list item, setting a product model, and navigating to a detail view. To the left of the code editor is a file tree under the 'Page' tab, showing the project structure with files like 'App.controller.js' and 'Home.controller.js' selected. At the bottom of the code editor, there are line numbers from 13 to 27.

Damos Click en alguno de los ítem de la tabla y al entrar en el Debug podemos ir revisando los valores de las variables:

The screenshot shows the Chrome DevTools Sources tab. The left sidebar lists files: top, port8080-workspaces-ws-gl... (selected), controller (App.controller.js, Home.controller.js), css, model, resources, test (flpSandbox.html?sap-ui-xx..., locate-reuse-libs.js), test-resources/sap/ushell/bo..., utils (HomeHelper.js, HomeService.js, Component.js), UI5 Inspector, and port35729-workspaces-ws-gl... . The right pane shows the code for Home.controller.js. Line 18 is highlighted with a yellow box. A tooltip for 'oDatos' shows the following object structure:

```

Object {
  Category: {__deferred: {...}},
  CategoryID: 2,
  Discontinued: false,
  Order_Details: {__deferred: {...}},
  ProductID: 3,
  ProductName: "Aniseed Syrup",
  QuantityPerUnit: "12 - 550 ml bottles",
  ReorderLevel: 25,
  Supplier: {__deferred: {...}},
  SupplierID: 1,
  UnitPrice: "10.0000",
  UnitsInStock: 13,
  UnitOnOrder: 70
}

```

Al terminar la ejecución podremos navegar al detalle:

The screenshot shows a SAP Fiori application window. The header includes the SAP logo and 'Freestyle SAPUI5'. The main content area has a title 'Ejemplo Vista Detalle'. The URL in the browser bar is 'port8080-workspaces-ws-gl.../test/flpSand...'. The page content is currently blank.

Binding en la página detalle

Se hace la implementación de función para obtener el ProductID enviado desde la Vista Home, se hace el registro del evento `_onObjectMatched` para que se ejecute en cuanto se cargue la página Detail:

The screenshot shows a code editor with the file `Detail.controller.js` open. The code is written in JavaScript and extends the `Detail` controller. It includes logic to get the router and attach a pattern match event listener. The `_onObjectMatched` function retrieves the ProductID from the URL and binds it to the view. Three green arrows highlight specific parts of the code: one pointing to the `attachPatternMatched` call, another pointing to the `_onObjectMatched` function definition, and a third pointing to the `bindElement` call within the `_onObjectMatched` function.

```

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.Detail", {
8         OnInit() {
9             let oRouter = sap.ui.core.UIComponent.getRouterFor(this);
10            oRouter.getRoute("detail").attachPatternMatched(this._onObjectMatched, this);
11        },
12
13        _onObjectMatched: function (oEvent) {
14            // Obtener el ProductID de la URL y enlazar el contexto
15            let sProductID = oEvent.getParameter("arguments").ProductID;
16
17            this.getView().bindElement({
18                path: "/Products(" + sProductID + ")",
19                parameters: {
20                    expand: "Order_Details"
21                }
22            });
23        },
24    });
25 });
26 });

```

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.Detail",
    {
        OnInit() {
            let oRouter = sap.ui.core.UIComponent.getRouterFor(this);

            oRouter.getRoute("detail").attachPatternMatched(this._onObjectMatched, this);

```

```
    },  
  
    _onObjectMatched: function (oEvent) {  
        // Obtener el ProductID de la URL y enlazar el contexto  
        let sProductID = oEvent.getParameter("arguments").ProductID;  
  
        this.getView().bindElement({  
            path: "/Products(" + sProductID + ")",  
            parameters: {  
                expand: "Order_Details"  
            }  
        });  
    },  
});  
});
```

Con la función `bindElement` se hace el Binding a la vista con la información del Producto y adicional se hace un `expand` con `Order_Details` y así nos obtenemos el resumen de los pedidos asociados al Producto seleccionado.

Ahora modificamos la Vista Detail para agregar información del Producto y una tabla con el listado `Order_Details`.

Info del Producto:

```
92     "ProductID": 3,  
93     "ProductName": "Aniseed Syrup",  
94     "SupplierID": 1,  
95     "CategoryID": 2,  
96     "QuantityPerUnit": "12 - 550 ml bottles",  
97     "UnitPrice": "10.0000",  
98     "UnitsInStock": 13,  
99     "UnitsOnOrder": 70,  
100    "ReorderLevel": 25,  
101    "Discontinued": false
```

Detail.view.xml M X JS Home.controller.js COMMIT_EDITMSG Storyboard - freestyle

freestyle > webapp > view > Detail.view.xml

You, a minute ago | 1 author (You)

```

1 <mvc:View controllerName="com.bootcamp.sapui5.freestyle.controller.Detail"
2   xmlns:mvc="sap.ui.core.mvc"
3   xmlns="sap.m" You, a few seconds ago • Uncommitted changes
4   xmlns:f="sap.ui.layout.form"
5   xmlns:core="sap.ui.core"
6   xmlns:smartTable="sap.ui.comp.smarttable">
7   <Page id="pageDetail" title="{i18n>titleDetail}">
8     <VBox id="dhb_01">
9       <f:SimpleForm id="SimpleFormDisplay354"
10         editable="false"
11         layout="ResponsiveGridLayout"
12         title="{i18n>Information}"
13         labelSpanXL="3" labelSpanL="3"
14         labelSpanM="3" labelSpanS="12"
15         adjustLabelSpan="false" emptySpanXL="4"
16         emptySpanL="4" emptySpanM="4" emptySpanS="0"
17         columnsXL="1" columnsL="1" columnsM="1"
18         singleContainerFullSize="false" >
19           <f:content>
20             <Label id="ld_01" text="{i18n>ProductName}" />
21             <Text id="nameText" text="{ProductName}" /> ←
22
23             <Label id="ld_02" text="{i18n>QuantityPerUnit}" />
24             <Text id="td_01" text="{Street} {QuantityPerUnit}" /> ←
25
26             <Label id="ld_03" text="{i18n>UnitPrice}" />
27             <Text id="td_02" text="{UnitPrice}" /> ←
28
29             <Label id="ld_04" text="{i18n>UnitsInStock}" />
30             <Text id="countryText" text="{UnitsInStock}" /> ←
31           </f:content>
32         </f:SimpleForm>

```

Detalle de Pedidos:

Detail.view.xml M X JS Home.controller.js COMMIT_EDITMSG Storyboard - freestyle

```
freestyle > webapp > view > Detail.view.xml
You, 4 minutes ago | 1 author (You)
1 <mvc:View controllerName="com.bootcamp.sapui5.freestyle.controller.Detail"
2   xmlns:mvc="sap.ui.core.mvc"
3   xmlns="sap.m"
4   xmlns:f="sap.ui.layout.form"
5   xmlns:core="sap.ui.core"
6   xmlns:smartTable="sap.ui.comp.smarttable"> You, 4 minutes ago • Uncommit
7   <Page id="pageDetail" title="{i18n>titleDetail}">
```

Usamos un control **SmartTable** el cual se va a explicar con más detalle en las próximas sesiones pero básicamente lo que hacemos en este punto es asignarle el BindingPath que en este caso es el Expand que realizamos en la función **_onObjectMatched** (expand: "Order_Details") y también el entitySet (la cual contiene el listado de campos de la entidad) y en la propiedad **initiallyVisibleFields** asignamos el conjunto de campos por defecto que queremos visualizar.

Detail.view.xml M X JS Home.controller.js COMMIT_EDITMSG Storyboard - freestyle Home.vi

```
freestyle > webapp > view > Detail.view.xml
1 <mvc:View controllerName="com.bootcamp.sapui5.freestyle.controller.Detail"
8   <VBox id="dhb_01">
34     <smartTable:SmartTable
35       id="ST_ORDER_D"
36       entitySet="Order_Details" ←
37       tableBindingPath="Order_Details" ←
38       initiallyVisibleFields="OrderID,ProductID,UnitPrice,Quantity,Discount"
39       tableType="ResponsiveTable"
40       enableExport="true"
41       useVariantManagement="false"
42       useTablePersonalisation="true"
43       header="{i18n>headerOrderDetails}" ←
44       showRowCount="true"
45       showFullScreenButton="true"
46       enableAutoBinding="true"
47       class="sapUiResponsiveContentPadding"
48       enableAutoColumnWidth="true"
49       editToggable="false" >
50       <smartTable:customToolbar>
51       </smartTable:customToolbar>
52     </smartTable:SmartTable>
53   </VBox>
54   </Page>
55 </mvc:View>
```

JavaScript

```
<mvc:View controllerName="com.bootcamp.sapui5.freestyle.controller.Detail"
```

```
xmlns:mvc="sap.ui.core.mvc"
xmlns="sap.m"
xmlns:f="sap.ui.layout.form"
xmlns:core="sap.ui.core"
xmlns:smartTable="sap.ui.comp.smarttable">
<Page id="pageDetail" title="{i18n>titleDetail}>
<VBox id="dhb_01">
    <f:SimpleForm id="SimpleFormDisplay354"
        editable="false"
        layout="ResponsiveGridLayout"
        title="{i18n>Information}"
        labelSpanXL="3" labelSpanL="3"
        labelSpanM="3" labelSpanS="12"
        adjustLabelSpan="false" emptySpanXL="4"
        emptySpanL="4" emptySpanM="4" emptySpanS="0"
        columnsXL="1" columnsL="1" columnsM="1"
        singleContainerFullSize="false" >
        <f:content>
            <Label id="ld_01" text="{i18n>ProductName}" />
            <Text id="nameText" text="{ProductName}" />

            <Label id="ld_02" text="{i18n>QuantityPerUnit}" />
            <Text id="td_01" text="{Street} {QuantityPerUnit}" />

            <Label id="ld_03" text="{i18n>UnitPrice}" />
            <Text id="td_02" text="{UnitPrice}" />

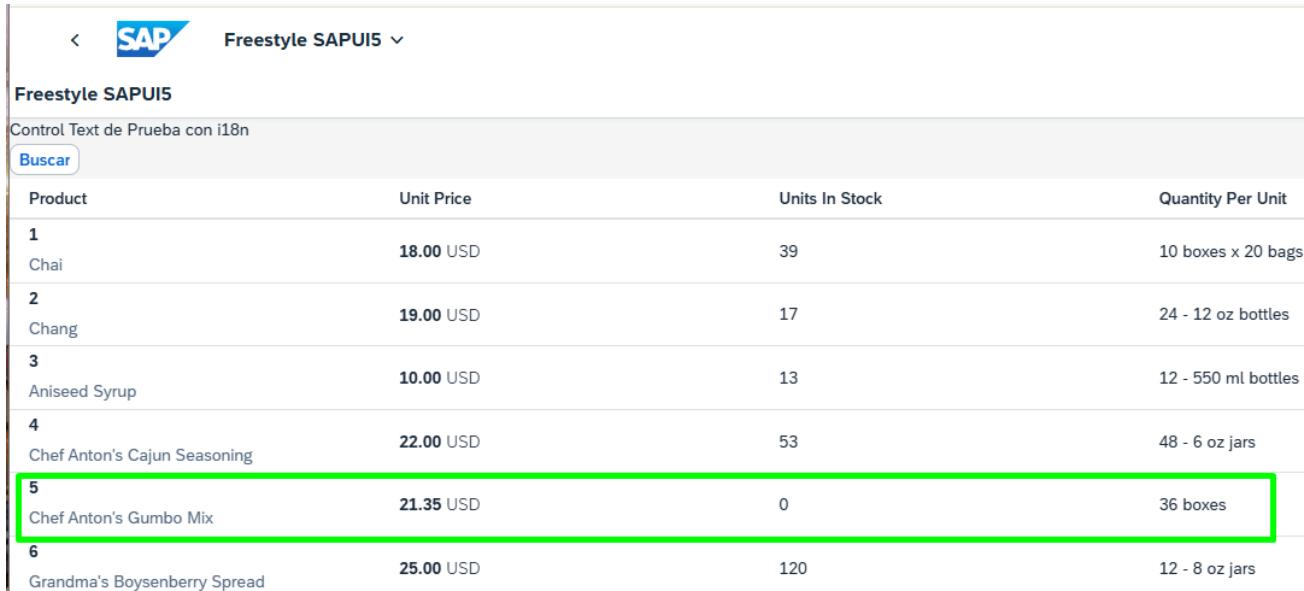
            <Label id="ld_04" text="{i18n>UnitsInStock}" />
            <Text id="countryText" text="{UnitsInStock}" />
        </f:content>
    </f:SimpleForm>

    <smartTable:SmartTable
        id="ST_ORDER_D"
        entitySet="Order_Details"
        tableBindingPath="Order_Details"

        initiallyVisibleFields="OrderID,ProductID,UnitPrice,Quantity,Discount"
        tableType="ResponsiveTable"
        enableExport="true"
        useVariantManagement="false"
        useTablePersonalisation="true"
        header="{i18n>headerOrderDetails}"
        showRowCount="true"
```

```
showFullScreenButton="true"
enableAutoBinding="true"
class="sapUiResponsiveContentPadding"
enableAutoColumnWidth="true"
editToggable="false" >
<smartTable:customToolbar>
</smartTable:customToolbar>
</smartTable:SmartTable>
</VBox>
</Page>
</mvc:View>
```

Ejecutamos la aplicación y se nos debe mostrar de la siguiente manera:



The screenshot shows a SAPUI5 application titled "Freestyle SAPUI5". The page header includes the SAP logo and the title "Freestyle SAPUI5". Below the header, there is a search bar with the placeholder "Control Text de Prueba con i18n" and a "Buscar" button. The main content is a table displaying product information. The table has columns: Product, Unit Price, Units In Stock, and Quantity Per Unit. The data rows are numbered 1 through 6. Row 5, which contains the product "Chef Anton's Gumbo Mix" with a unit price of "21.35 USD", units in stock of "0", and quantity per unit of "36 boxes", is highlighted with a green border.

Product	Unit Price	Units In Stock	Quantity Per Unit
1 Chai	18.00 USD	39	10 boxes x 20 bags
2 Chang	19.00 USD	17	24 - 12 oz bottles
3 Aniseed Syrup	10.00 USD	13	12 - 550 ml bottles
4 Chef Anton's Cajun Seasoning	22.00 USD	53	48 - 6 oz jars
5 Chef Anton's Gumbo Mix	21.35 USD	0	36 boxes
6 Grandma's Boysenberry Spread	25.00 USD	120	12 - 8 oz jars

The screenshot shows a SAP Freestyle SAPUI5 application interface. At the top, there is a header with the SAP logo and the text "Freestyle SAPUI5". Below the header, the title "Ejemplo Vista Detalle" is displayed. Underneath the title, the section "Datos del producto" is shown, which contains a box with the following information:

Product Name:	Chef Anton's Gumbo Mix
Quantity Per Unit:	36 boxes
Unit Price:	21.3500
Units In Stock:	0

Below this, a table titled "Order Details (10)" is displayed, also enclosed in a green box. The table has columns: OrderID, ProductID, UnitPrice, Quantity, and Discount. The data is as follows:

OrderID	ProductID	UnitPrice	Quantity	Discount
10,258	5	17.0000	65	0.2
10,262	5	17.0000	12	0.2
10,290	5	17.0000	20	0
10,382	5	17.0000	32	0
10,635	5	21.3500	15	0.1
10,708	5	21.3500	4	0
10,848	5	21.3500	30	0
10,958	5	21.3500	20	0
11,030	5	21.3500	70	0
11,047	5	21.3500	30	0.25