

# T06M03 – Create an App Using SAP BTP and Gateway System

**Product**

SAP Business Transaction Platform (BTP)

**Level**

Undergraduate/Graduate  
Beginner

**Focus**

Application Development

**Author**

Adriana Trujillo, teaching assistant

**Revision for Uniandes**

Luis Felipe Plazas, teaching assistant

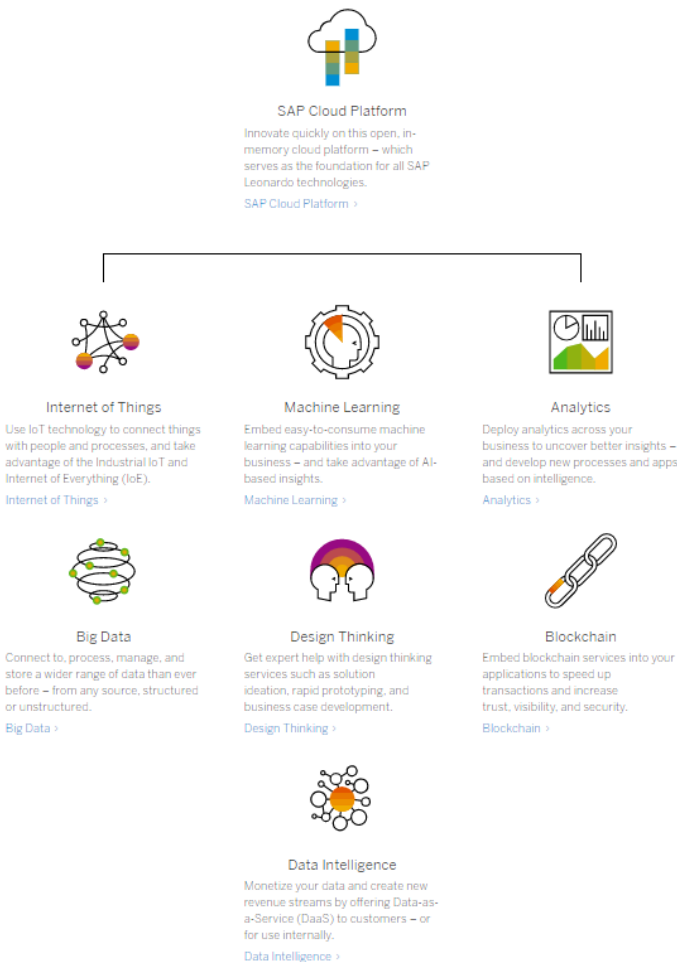
Oscar Avila, PhD, director of the innovation hub in EIS

**MOTIVATION**

This case illustrates many of the features to develop applications on the Business Studio environment of the SAP Business Transaction Platform

**PREREQUISITES**

T05M03



## Create an app using SAP Business Studio

This case illustrates many aspects of developing applications on the SAP Business Transaction Platform including:

- Creating and using destinations to a legacy system (SAP Gateway system)
- Storing code in the git repository
- Using fragments to modularize code

## Create an Account on the SAP Gateway Demo System

The SAP Gateway Demo System provides demo data and services to support new Fiori applications. To access such resources, we will consume/create data of an OData service provided by this system. This is exactly the scenario an integration or technical consultant can have when trying to extend a legacy system such the SAP ERP in which OData services are exposed in order to access data from a new application.

Navigate to <https://register.sapdevcenter.com/SUPSignForms> and log on using your SAP credentials.

Agree to the terms and conditions and click Register.

### SAP Gateway Demo Server - ES5

#### User Information

User ID	P2001276693
First Name	Adriana
Last Name	Trujillo
Email	a.trujilloa1@uniandes.edu.co

#### Terms and Conditions

written consent. If any provision of this T&DE proves to be invalid, this will not affect any other provision of this T&DE. This T&DE does not entitle either party to use the other party's name, trademark or trade designation for purposes of advertising and marketing without prior written consent of this party. The waiver by either party of any of its rights hereunder shall not be construed as a waiver of any subsequent breach.

##### 19. Contact:

If You have any questions or support-issues with regards to the Software please refer to our help portal: <http://scn.sap.com/community/developer-center/netweaver-gateway>

☒ I have read and understood the Terms and Conditions.

When you receive the email confirming your registration, click on the link to log on to the SAP Gateway WebGUI so you can create an account.

**SAP NetWeaver**

System: ES5

Client:\* 002

User:\* |

Password:\*

Language: English

**Log On**

[Change Password](#)

Welcome to ICC Test system, NW AS ABAP 7.51 SAP IDES  
Client for usage is 002

For user creation/password reset:  
<https://register.sapdevcenter.com/SUPSignForms/>

For system related issues contact:

**SAP**

Copyright © 2020 SAP SE. All rights reserved.

## SAP Business Technology Platform

The SAP Business Technology Platform (BTP), formerly SAP Cloud Platform (SCP), is a platform as a service (PaaS) for creating new applications or extending and integrating existing applications in a cloud computing environment. The objective of the workshop is to create a new application by using the SAP Business Studio environment provided by this platform and accessing data from the SAP Gateway Demo Server (see last subsection) by using an OData service. The SAP Gateway Demo Server will make in this workshop the function of a legacy system that you want to extend.

The **SAP BTP trial** account is a free developer account. It has no use time limit but certain limitations such as a 1 GB storage limit. One significant limitation from an instructional point of view is that it only allows one account making it difficult to share the content you create. Otherwise, it provides almost all the capabilities of the full platform. In this section, you will register for an account.

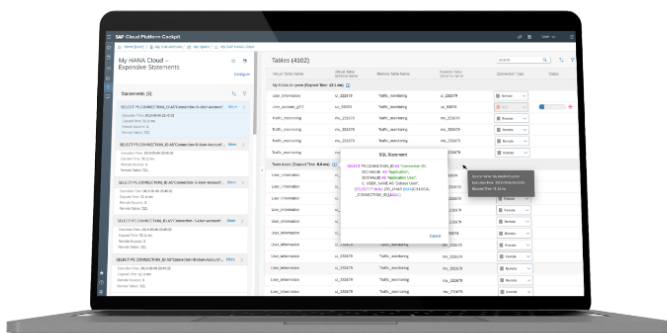
## Registering for an Account

Navigate to the following link:

<https://www.sap.com/cmp/td/sap-hana-cloud-trial.html>

Select the option Start your free trial today

## SAP HANA Cloud Trial



**Sign up for SAP HANA Cloud trial and discover what we can offer**

- Get access to free services up to 90 days\*
- Access to starter scenarios
- Guided tutorials for you to learn the basics

[Start your free trial today](#)

Fill in all the data and verify the e-mail. Once you enter to the trial, it will ask you to introduce and verify your phone number.

## Verify Your Telephone Number

**i** You can change your telephone number. Please type: +(country code) followed by the area code without the leading zero, followed by the subscriber number. Example: +(XXX) XXX XXXXXXXX

The cockpit-trial-cf-eu10 application requires telephone verification. We have sent a code to your telephone number. Please enter the code you have received and choose Continue.

Telephone \*

Code \*

\*Required

[Request New Code](#)

[Continue](#)

**SAP® ID Service**


Select US East as our region.

Welcome to Your Personal Trial

Hi Adriana,

Before you continue, we need to finalize your trial account creation. Choose the region that's closest to you:

☒

 US East (VA) - AWS

☐

 Europe (Frankfurt) - AWS

Create Account

Click on “Go to your Trial Account” to access the Cloud Foundry Environment.


SAP SAP BTP Cockpit

Welcome to SAP BTP Trial


Learn how to create and deploy cloud apps and gain access to a comprehensive set of platform services.

Go To Your Trial Account


Quick Tool Access



SAP Business Application Studio  
Develop business applications using SAP's next-generation, Web-based IDE




CLI for BTP  
Manage your trial account using the command-line interface



APIs for SAP BTP  
Manage, build, and extend the core capabilities of SAP BTP

Navigate through your trial subaccount

Trial Home /  cb972f7dtrial

Global Account: cb972f7dtrial – Account Explorer

All: 0 directories, 1 subaccounts | Subdomain: cb972f7dtrial-ga

Search

Q


All Regions

☐ Hide entities you don't have access


Directories and Subaccounts

Subaccounts (1)

Subaccounts

 trial

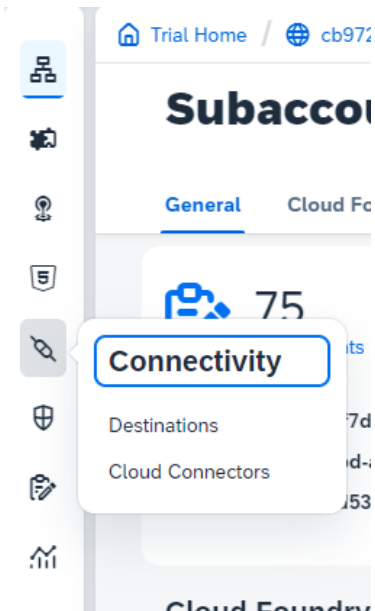
Provider: Amazon Web Services (AWS)

Region:  US East (VA)

Environment: Multi-Environment

Select **Connectivity** on the left-hand menu.

4



Click **Destinations -> Create Destination** and then enter the details shown below:

Name: *	ES5
Type:	HTTP
Description:	SAP Gateway ES5
URL: *	https://sapes5.sapdevcenter.com
Proxy Type:	Internet
Authentication:	BasicAuthentication
User: *	P2001276693
Password:	*****

The **User** is your **SAP Gateway User ID and Password**. Before saving use the **New Property** button to add the following three properties:

WebIDEEEnabled	True
WebIDESystem	Gateway
WebIDEUsage	odata_abap
sap-platform	ABAP
sap-client	002
HTML5.DynamicDestination	true

Additional Properties

HTML5.Dyn...	true	
sap-client	002	
sap-platform	ABAP	
WebIDEEna...	true	
WebIDESyst...	Gateway	
WebIDEUsage	odata_abap	

☒ Use default JDK truststore

Once you save the destination, you can use **the Check Availability of Destination Connection** button to make sure the URL can be reached.



## Check Connection

✓ Connection to "ES5" established. Response returned: "307: Temporary Redirect"

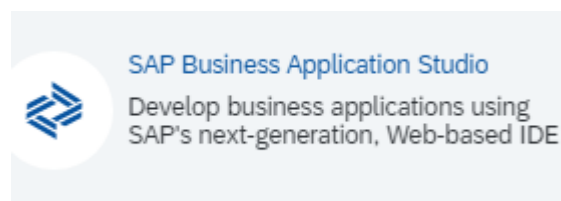
Close

The results are sometimes ambiguous. The only result that is a problem is 404 – Not Found error.

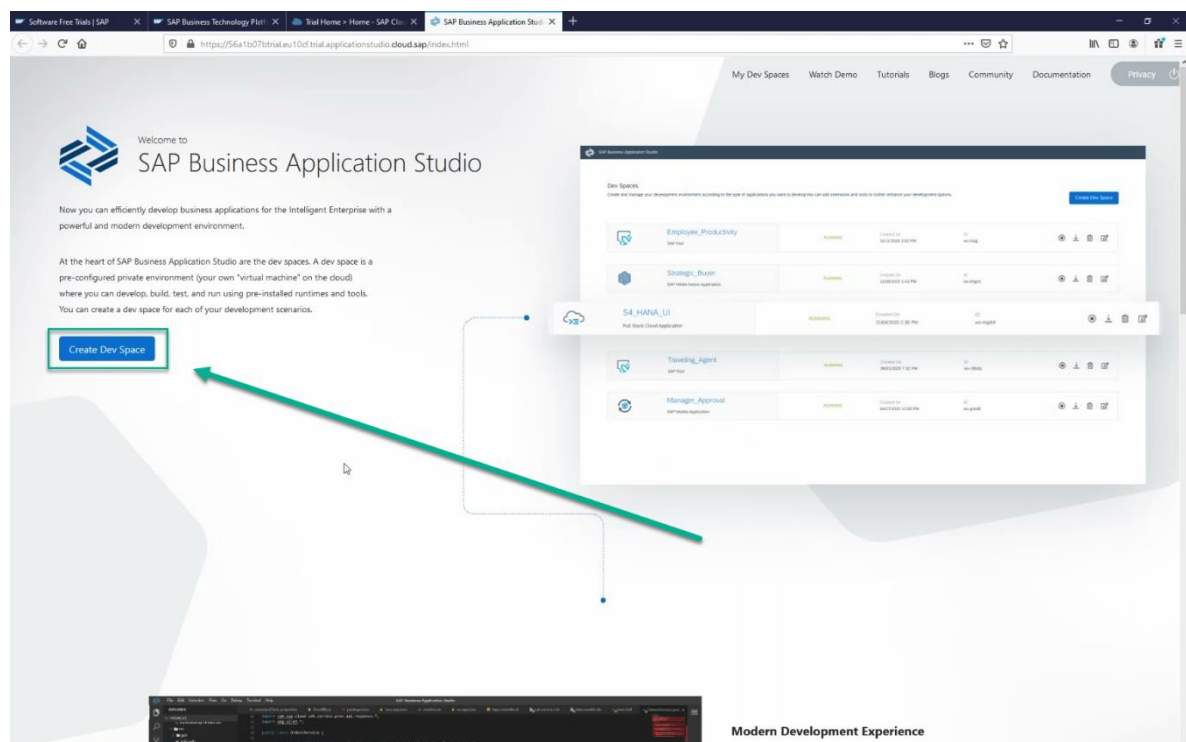
## Create a Dev Space for SAP Fiori Apps

An SAP Business Application Studio dev space is a preconfigured environment with the required tools and extensions tailored for a specific business scenario. Dev spaces are like isolated virtual machines in the cloud that can be instantly spin-up. Each dev space type contains tailored tools and pre-installed run-times for a target scenario such as SAP Fiori or mobile development. This simplifies and saves time in setting up the development environment as there's no need to install anything or upgrade, letting developers focus on their business domain, anytime, anywhere.

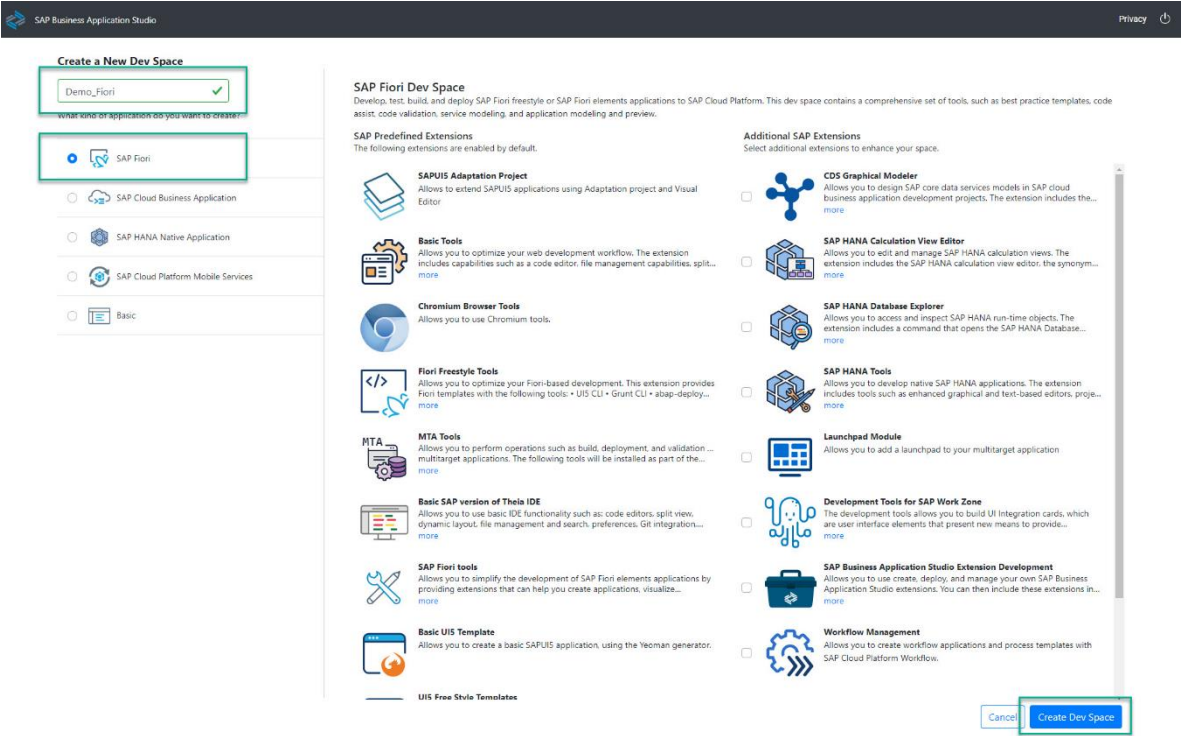
To access the Trial, select the SAP Business Application Studio.



If you have not created a dev space, the welcome page for SAP Business Application Studio loads.

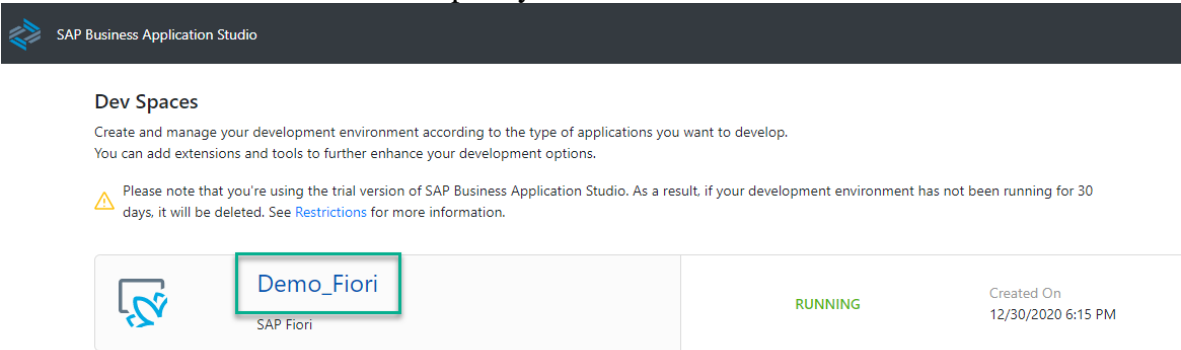


- 1. Enter Demo\_Fiori for your dev space **name**.
- 2. Choose **SAP Fiori** as the application type.
- 3. Click on **Create Dev Space**.

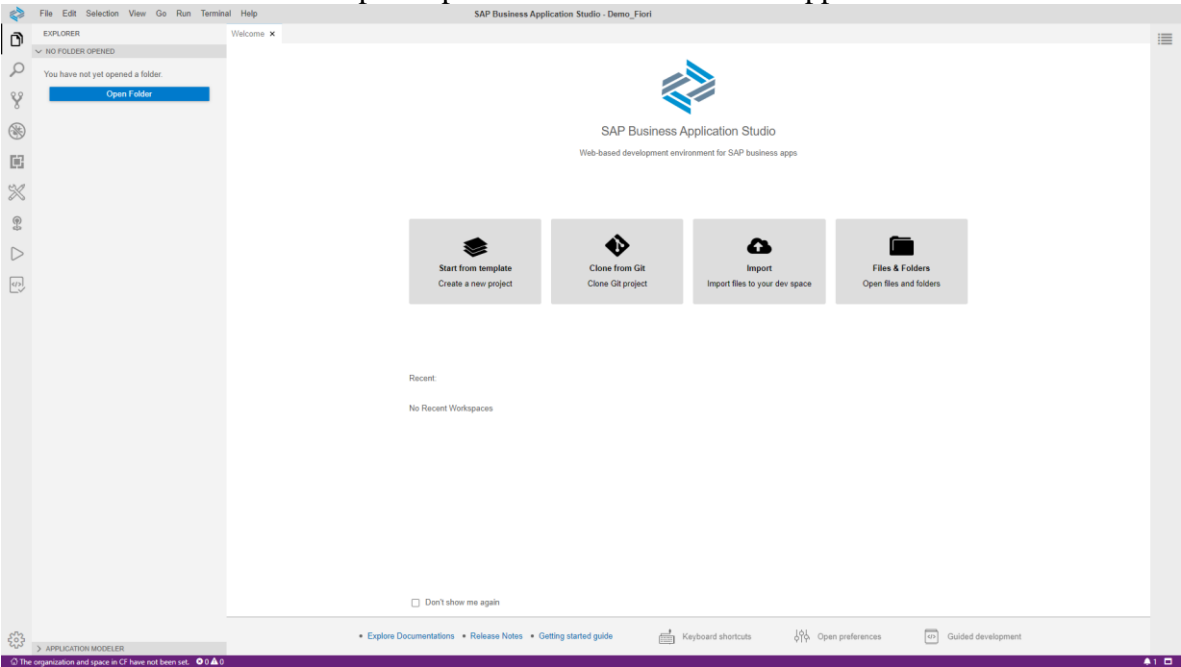


Open the SAP Fiori dev space

- 1. **Click** the name of the dev space you created.



- 2. The SAP Fiori dev space opens and the **Welcome** tab appears.





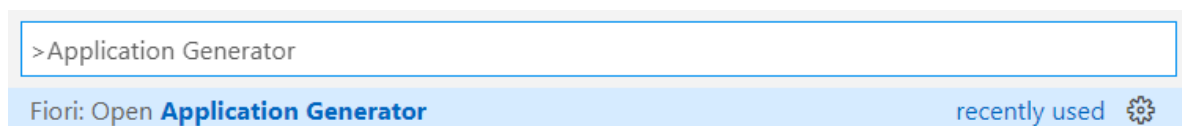
## Set Up and Generate a New SAP Fiori App Project

The purpose of this tutorial is to familiarize you with the process for creating applications using the SAP Fiori tools Application Generator, as well as the process for testing your application with both mock and real data.

In this case, you will create an SAP Fiori element list report page displaying a list of products. This list report will be modified in later tutorials using the other extensions available in SAP Fiori tools.

SAP Fiori tools includes an Application Generator that provides a wizard-style approach for creating applications based on SAP Fiori elements page types. You are going to use it to create your List Report Object Page app.

In Visual Studio Code, open the Command Palette using **CMD/CTRL + Shift + P**, type **Application Generator**, and select **Fiori: Open Application Generator**.



Select the **SAP Fiori Worklist Application** tile and press **Next**.

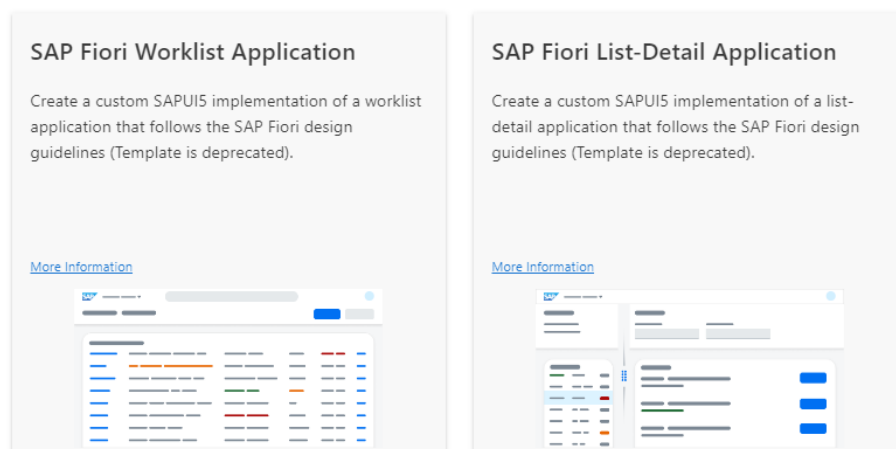
### Template Selection

Choose your application template.

Template Type \*

Deprecated Templates

Which template do you want to use? (Note: these templates are deprecated and will be removed by the end of 2023. Please use the Custom page SAP Fiori template as an alternative) \*



## Configure service for List Report Object Page

With the page type selected, it is time to connect a data source. You will use the OData service mentioned in the prerequisite section of this tutorial.

Select **Connect to an OData Service** from the dropdown menu. A field to enter the OData service URL will appear. Copy and paste the service URL:

[https://sapes5.sapdevcenter.com/sap/opu/odata/sap/SEPMRA\\_PROD\\_MAN/](https://sapes5.sapdevcenter.com/sap/opu/odata/sap/SEPMRA_PROD_MAN/)

You may be prompted to enter credentials to access the service. Enter your username and password, and click the **Login** and then **Next** button



## Data Source and Service Selection

Configure the data source and select a service.

Data source \*

Connect to an OData Service

OData service URL ? \*

https://sapes5.sapdevcenter.com/sap/opu/odata/sap/SEPMRA\_PF

Service username \*

P2001276693

Service password \*

.....

Login

Mandatory field

After successfully connecting to the supplied OData service, click **Next** to customize the template.

On Entity Selection choose the following data and click **Next**:

Object collection \*

SEPMRA\_C\_PD\_Product

Object collection key \*

Product

Object ID \*

Name

Object number \*

None

Object unit of measure \*

None

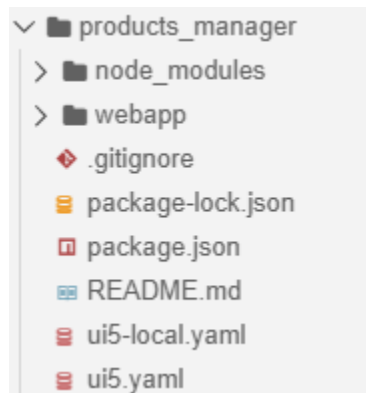
With the page type and data source defined, the next step is to configure the main project attributes:

Field Name	Value
What is the module name for your application?	products_manager
What is the title for your application?	Products Manager
What is the namespace for your application?	Namespace1
What is the description for your application?	SAP Fiori elements application for managing products
Choose your project folder	Click the folder selection icon and select where to save your project.

Click **Finish**.

At this point, your application folder will be generated based on the input from above.

Once your project has been generated, you will have the option to add it to your existing workspace. Verify that your project has been created and looks like this:

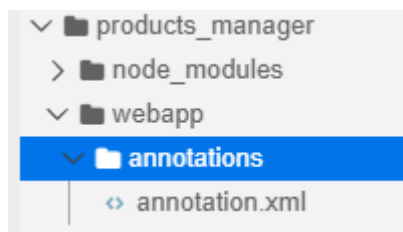


## Editing the Application

1. Go to webapp -> i18n -> i18n.properties and replace the information with the following one:

```
# This is the resource bundle for products_manager
#Texts for manifest.json
#XTIT: Application name
appTitle=Products Manager
#YDES: Application description
appDescription= SAP Fiori elements application for managing products.
```

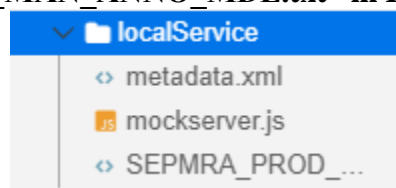
2. Add a new Folder into webapp, called “annotations”, then create a new file “annotation.xml”



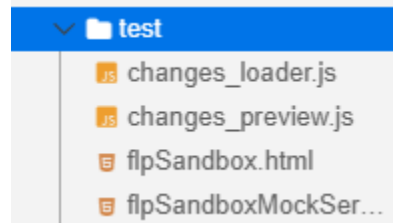
3. Add the following information into “annotation.xml”

```
<edmx:Edmx xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx" Version="4.0">
  <edmx:Reference Uri="https://sap.github.io/odata-vocabularies/vocabularies/Common.xml">
    <edmx:Include Namespace="com.sap.vocabularies.Common.v1" Alias="Common"/>
  </edmx:Reference>
  <edmx:Reference Uri="https://sap.github.io/odata-vocabularies/vocabularies/UI.xml">
    <edmx:Include Namespace="com.sap.vocabularies.UI.v1" Alias="UI"/>
  </edmx:Reference>
  <edmx:Reference Uri="https://sap.github.io/odata-
vocabularies/vocabularies/Communication.xml">
    <edmx:Include Namespace="com.sap.vocabularies.Communication.v1"
Alias="Communication"/>
  </edmx:Reference>
  <edmx:Reference Uri="/sap/opu/odata/sap/SEPMRA_PROD_MAN/$metadata">
    <edmx:Include Namespace="SEPMRA_PROD_MAN" Alias="SAP"/>
  </edmx:Reference>
  <edmx:DataServices>
    <Schema xmlns="http://docs.oasis-open.org/odata/ns/edm" Namespace="local">
    </Schema>
  </edmx:DataServices>
</edmx:Edmx>
```

4. At the folder “localService”, create a new file “SEPMRA\_PROD\_MAN\_ANNO\_MDL.xml” and paste the information you can find “SEPMRA\_PROD\_MAN\_ANNO\_MDL.txt” in BLOQUE NEÓN.



5. In the folder “test” delete all, and create the following files:



- changes\_loader.js

```
//This file used only for loading the changes in the preview and not required to be checked in.
//Loads and extends the openui5 FileListBaseConnector
//For UI5 version >= 1.80, the location of the FileListBaseConnector is different
const connectorPath =
  parseFloat(sap.ui.version) >= 1.8
    ? 'sap/ui/fl/write/api/connectors/FileListBaseConnector'
    : 'sap/ui/fl/initial/api/connectors/FileListBaseConnector';
sap.ui.define(['sap/base/util/merge', connectorPath], function(merge, FileListBaseConnector) {
  var aPromises = [];
  var trustedHosts = [/^localhost$/, /^.*.applicationstudio.cloud.sap$/];
  var url = new URL(window.location.toString());
  var isValidHost = trustedHosts.some((host) => {
    return host.test(url.hostname);
  });
  return merge({}, FileListBaseConnector, {
    getFileList: function() {
      return new Promise(function(resolve, reject) {
        // If no changes found, maybe because the app was executed without doing a build.
        // Check for changes folder and load the changes, if any.
        if (!isValidHost) reject(console.log('cannot load flex changes: invalid host'));
        $.ajax({
          url: url.origin + '/changes/',
          type: 'GET',
          cache: false
        })
        .then(function(sChangesFolderContent) {
          var regex = /(\\changes\\[^\"]*\\.change)/g;
          var result = regex.exec(sChangesFolderContent);
          var aChanges = [];
          while (result !== null) {
            aChanges.push(result[1]);
            result = regex.exec(sChangesFolderContent);
          }
          resolve(aChanges);
        })
        .fail(function(obj) {
          // No changes folder, then just resolve
          resolve();
        });
      });
    }
  });
});
```

- changes\_preview.js you can find de the information **in BLOQUE NEÓN (changes\_preview.txt)**
- flpSandbox.html

```
<!DOCTYPE HTML>
<html>
<head>
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>{{appTitle}}</title>
  <script type="text/javascript">
    window["sap-ushell-config"] = {
      defaultRenderer: "fiori2",
      bootstrapPlugins: {
        "RuntimeAuthoringPlugin": {
          component: "sap.ushell.plugins.rta",
          config: {
            validateAppVersion: false
          }
        }
      },
      renderers: {
        fiori2: {
          componentData: {
            config: {
              search: "hidden"
            }
          }
        }
      },
      applications: {
        "masterDetail-display": {
          title: "Products Manager",
          description: "A Fiori application.",
          additionalInformation: "SAPUI5.Component=productsmanager",
          applicationType: "URL",
          url: "../"
        }
      }
    };
  </script>

  <script src="/test-resources/sap/ushell/bootstrap/sandbox.js" id="sap-ushell-bootstrap"></script>
  <!-- Bootstrap the UI5 core library -->
  <script id="sap-ui-bootstrap"
    src="/resources/sap-ui-core.js"
    data-sap-ui-
libs="sap.m, sap.ushell, sap.ui.core, sap.f, sap.ui.comp, sap.ui.table, sap.suite.ui.generic.template, sap.ui.generic.app,
sap.collaboration"
    data-sap-ui-async="true"
    data-sap-ui-preload="async"
    data-sap-ui-theme="sap_fiori_3"
    data-sap-ui-compatVersion="edge"
    data-sap-ui-flexibilityServices='[{"applyConnector": "productsmanager/test/changes_loader", "custom": true}]'
    data-sap-ui-language="en"
    data-sap-ui-resourceroots={"productsmanager": "../"}'
    data-sap-ui-frameOptions="allow"> // NON-SECURE setting for testing environment
  </script>
  <script src="../utils/locate-reuse-libs.js" data-sap-ui-manifest-uri="../manifest.json">
  </script>
  <!-- relevant for version < 1.78.0 only -->
  <script src="changes_preview.js"></script>
</head>

<!-- UI Content -->

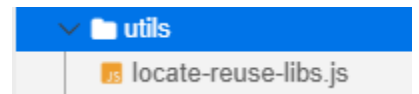
<body class="sapUiBody" id="content">
</body>

</html>
```

- flpSandboxMockServer.html

```
<html>
<head>
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>{{appTitle}}</title>
  <script type="text/javascript">
    window["sap-ushell-config"] = {
      defaultRenderer: "fiori2",
      bootstrapPlugins: {
        "RuntimeAuthoringPlugin": {
          component: "sap.ushell.plugins.rta",
          config: {
            validateAppVersion: false
          }
        },
      },
      renderers: {
        fiori2: {
          componentData: {
            config: {
              search: "hidden"
            }
          }
        }
      },
      applications: {
        "masterDetail-display": {
          title: "Products Manager",
          description: "A Fiori application.",
          additionalInformation: "SAPUI5.Component=productsmanager",
          applicationType: "URL",
          url: "../"
        }
      }
    };
  </script>
  <script src="/test-resources/sap/ushell/bootstrap/sandbox.js" id="sap-ushell-bootstrap"></script>
  <!-- Bootstrap the UI5 core library -->
  <script id="sap-ui-bootstrap"
    src="/resources/sap-ui-core.js"
    data-sap-ui-
libs="sap.m, sap.ushell, sap.ui.core, sap.f, sap.ui.comp, sap.ui.table, sap.suite.ui.generic.template, sap.ui.
generic.app, sap.collaboration"
    data-sap-ui-async="true"
    data-sap-ui-preload="async"
    data-sap-ui-theme="sap_fiori_3"
    data-sap-ui-compatVersion="edge"
    data-sap-ui-
flexibilityServices='[{"applyConnector":"productsmanager/test/changes_loader", "custom":true}]'
    data-sap-ui-language="en"
    data-sap-ui-resourceroots='{"productsmanager": "../'}'
    data-sap-ui-frameOptions="allow"> // NON-SECURE setting for testing environment
  </script>
  <script src="../utils/locate-reuse-libs.js"
    data-sap-ui-manifest-uri="../manifest.json"
    data-sap-ui-componentName="productsmanager"
    data-sap-ui-use-mockserver="true">
  </script>
  <!-- relevant for version < 1.78.0 only -->
  <script src="changes_preview.js"></script>
</head>
<body class="sapUiBody" id="content">
</body>
```

6. Create a new folder called “utils” and create a new file “locate-reuse-libs.js”:



```
(function (sap) {
  fioriToolsGetManifestLibs = function (manifestPath) {
    var url = manifestPath;
    var result = "";
    var ui5Libs = [
      "sap.apf", "sap.base", "sap.chart", "sap.collaboration", "sap.f", "sap.fe", "sap.fileviewer", "sap.gantt", "sap.landvisz",
      "sap.m", "sap.ndc", "sap.ovp", "sap.rules", "sap.suite", "sap.tnt", "sap.ui", "sap.uiext", "sap.ushell", "sap.uxap",
      "sap.viz", "sap.webanalytics", "sap.zen"
    ];
    return new Promise(function (resolve, reject) {
      $.ajax(url)
        .done(function (manifest) {
          if (manifest) {
            if (
              manifest["sap.ui5"] && manifest["sap.ui5"].dependencies && manifest["sap.ui5"].dependencies.libs
            ) {
              Object.keys(manifest["sap.ui5"].dependencies.libs).forEach(function (manifestLibKey) {
                if (!ui5Libs.some(function (substring) { return manifestLibKey === substring || manifestLibKey.startsWith(substring + "."); }))) {
                  if (result.length > 0) {
                    result = result + "," + manifestLibKey;
                  } else {
                    result = manifestLibKey;
                  }
                }
              });
            }
            resolve(result);
          }
          .fail(function (error) {
            reject(new Error("Could not fetch manifest at " + manifestPath));
          });
        });
      sap.registerComponentDependencyPaths = function (manifestPath) {
        return fioriToolsGetManifestLibs(manifestPath).then(function (libs) {
          if (libs && libs.length > 0) {
            var url = "/sap/bc/ui2/app_index/ui5_app_info?id=" + libs;
            var sapClient = jQuery.sap.getUriParameters().get("sap-client");
            if (sapClient && sapClient.length === 3) {
              url = url + "&sap-client=" + sapClient;
            }
            return $.ajax(url).done(function (data) {
              if (data) {
                Object.keys(data).forEach(function (moduleDefinitionKey) {
                  var moduleDefinition = data[moduleDefinitionKey];
                  if (moduleDefinition && moduleDefinition.dependencies) {
                    moduleDefinition.dependencies.forEach(function (dependency) {
                      if (dependency.url && dependency.url.length > 0 && dependency.type === "UI5LIB") {
                        jQuery.sap.log.info(
                          "Registering Library " +
                          dependency.componentId +
                          " from server " +
                          dependency.url);
                        jQuery.sap.registerModulePath(dependency.componentId, dependency.url);
                      }
                    });
                  }
                });
              }
            });
          }
        });
      };
      (sap);
      var scripts = document.getElementsByTagName("script");
      var currentScript = scripts[scripts.length - 1];
      var manifestUri = currentScript.getAttribute("data-sap-ui-manifest-uri");
      var componentName = currentScript.getAttribute("data-sap-ui-componentName");
      var useMockserver = currentScript.getAttribute("data-sap-ui-use-mockserver");
      sap.registerComponentDependencyPaths(manifestUri)
        .catch(function (error) {
          jQuery.sap.log.error(error);
        })
        .finally(function () {
          if (componentName && componentName.length > 0) {
            if (useMockserver && useMockserver === "true") {
              sap.ui.getCore().attachInit(function () {
                sap.ui.require([componentName.replace(/\.\/g, "/") + "/localService/mockserver"], function (server) {
                  // set up test service for local testing
                  server.init();
                  // initialize the ushell sandbox component
                  sap.ushell.Container.createRenderer().placeAt("content");
                });
              } else {
                sap.ui.require(["sap/ui/core/ComponentSupport"]);
              }
            } else {
              sap.ui.getCore().attachInit(function () {
                sap.ushell.Container.createRenderer().placeAt("content");
              });
            }
          }
          sap.registerComponentDependencyPaths(manifestUri);
        });
    });
  };
})(sap);
```



7. Update the information on the “Component.js” with the following:

```
sap.ui.define(['sap/suite/ui/generic/template/lib/AppComponent'], function(AppComponent) {
    return AppComponent.extend("productsmanager.Component", {
        metadata: {
            manifest: "json"
        }
    });
});
```

8. Change “index.html” with the following:

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Manage products</title>
<style>
    html, body, body > div, #container, #container-uiarea {
        height: 100%;
    }
</style>
<script
    id="sap-ui-bootstrap"
    src="resources/sap-ui-core.js"
    data-sap-ui-theme="sap_fiori_3"
    data-sap-ui-resourceroots='{
        "productsmanager": "."
    }'
    data-sap-ui-compatVersion="edge"
    data-sap-ui-async="true"
    data-sap-ui-preload="async"
    data-sap-ui-libs="sap.m">
</script>
<script src="./utils/locate-reuse-libs.js"
    data-sap-ui-manifest-uri="/manifest.json"
    data-sap-ui-componentName="productsanager">
</script>
</head>
<body class="sapUiBody sapUiSizeCompact" id="content">
<div data-sap-ui-component data-name="productsmanager" data-id="container" data-
settings='{ "id" : "products_manager" }' data-handle-validation="true"></div>
</body>
</html>
```

9. Change “manifest.json” with the information you find on the “manifest.txt” document on BLOQUE NEÓN.

10. Change “package.json” with the following:

```
{ "name": "products_manager",
  "version": "0.0.1",
  "private": true,
  "sapux": true,
  "description": "A Fiori application.",
  "keywords": [
    "ui5",
    "openui5",
    "sapui5" ],
  "main": "webapp/index.html",
  "scripts": { "start": "fiori run --open 'test/flpSandbox.html#masterDetail-display'",
    "start-local": "fiori run --config ./ui5-local.yaml --open 'test/flpSandboxMockServer.html#masterDetail-display'",
    "build": "ui5 build -a --clean-dest --include-task=generateManifestBundle generateCachebusterInfo",
    "deploy": "fiori verify",
    "deploy-config": "fiori add deploy-config",
    "start-mock": "fiori run --open 'test/flpSandboxMockServer.html#masterDetail-display' },
  "devDependencies": { "@ui5/cli": "^2.10.1",
    "@ui5/fs": "^2.0.6",
    "@ui5/logger": "^2.0.1",
    "@sap/ux-ui5-tooling": "1",
    "rimraf": "3.0.2",
    "@sap/ux-specification": "latest"},
  "ui5": { "dependencies": [
    "@sap/ux-ui5-tooling" ]}
}
```

11. In the folder “ui5-local.yaml” add on the **libraries** part the following:

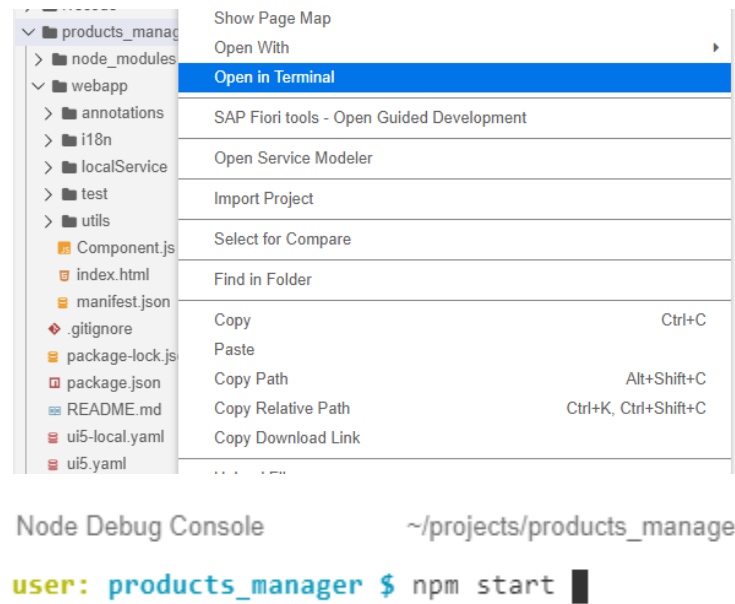
- **name: sap.collaboration**

```
libraries:
- name: sap.m
- name: sap.ushell
- name: sap.ui.core
- name: sap.f
- name: sap.ui.comp
- name: sap.ui.table
- name: sap.suite.ui.generic.template
- name: sap.ui.generic.app
- name: sap.collaboration
- name: themelib_sap_fiori_3
```

## Launch the application

In order to launch the generated app, simply run the following from the generated app root folder:

```
npm start
```



This is the view of the application:

Products Manager

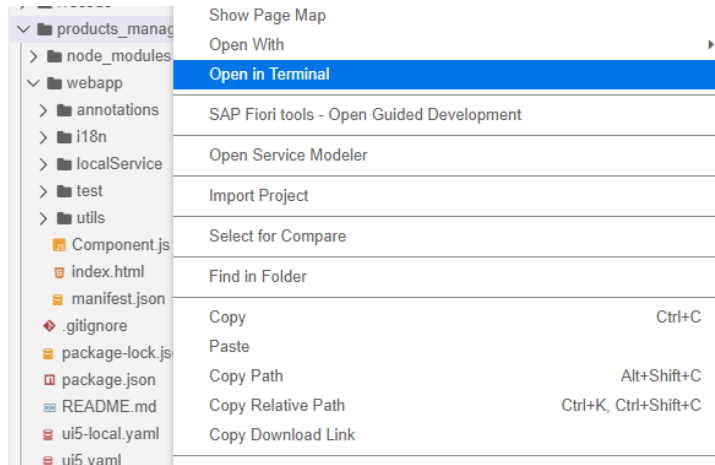
Products (123)							Search		Copy	Create	Delete	↕	⌵	[=]	⚙	Show Details
<input type="checkbox"/>	Image	Product	Category	Availability	Average Rating	Price per Unit										
<input type="checkbox"/>		10" Portable DVD player HT-2001	TV, Video & HiFi		★★★★★	449.99 USD										
<input type="checkbox"/>		7" Widescreen Portable DVD Player w MP3 HT-2000	TV, Video & HiFi		★★★★★	249.99 EUR										
<input type="checkbox"/>		Astro Laptop 1516 HT-1251	Computer Systems		★★★★★	989.00 EUR										
<input type="checkbox"/>		Astro Phone 6 HT-1252	Smartphones & Tablets		★★★★★	649.00 EUR										

Deliverable

The workshop must be imported to github, the link to the repository must be sent through Bloque Neón.

## How to import to github

Put the following into your project's terminal:



git init

git remote add origin (link de su repo en github)

git branch -M main

git push -u origin main

```
user: products_manager $ git init
Initialized empty Git repository in /home/user/projects/products_manager/.git/
user: products_manager $ git remote add origin https://github.com/lplazasp/taller5SAP
```

- user: products\_manager \$ git branch -M main
- user: products\_manager \$ git push -u origin main

```
Enumerating objects: 42, done.
Counting objects: 100% (42/42), done.
Delta compression using up to 8 threads
Compressing objects: 100% (39/39), done.
Writing objects: 100% (42/42), 35.28 KiB | 1.31 MiB/s, done.
Total 42 (delta 8), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (8/8), done.
To https://github.com/lplazasp/taller5SAP
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

Evaluation rubric:

Criteria	Nivel 5	Nivel 4	Nivel 3	Nivel 2	Nivel 1	Nivel 0
i18n <a href="#">Agregar comentarios</a>	2 puntos	1.6 puntos	1.2 puntos	0.8 puntos	0.4 puntos	0 puntos
annotations <a href="#">Agregar comentarios</a>	2 puntos	1.6 puntos	1.2 puntos	0.8 puntos	0.4 puntos	0 puntos
SEMPRA_PROD <a href="#">Agregar comentarios</a>	17 puntos	13.6 puntos	10.2 puntos	6.8 puntos	3.4 puntos	0 puntos
changes_loader <a href="#">Agregar comentarios</a>	4 puntos	3.2 puntos	2.4 puntos	1.6 puntos	0.8 puntos	0 puntos
change_preview <a href="#">Agregar comentarios</a>	4 puntos	3.2 puntos	2.4 puntos	1.6 puntos	0.8 puntos	0 puntos
flpSandbox <a href="#">Agregar comentarios</a>	4 puntos	3.2 puntos	2.4 puntos	1.6 puntos	0.8 puntos	0 puntos

flpSandboxMock <a href="#">Agregar comentarios</a>	4 puntos	3.2 puntos	2.4 puntos	1.6 puntos	0.8 puntos	0 puntos
locate_reuse <a href="#">Agregar comentarios</a>	2 puntos	1.6 puntos	1.2 puntos	0.8 puntos	0.4 puntos	0 puntos
Component <a href="#">Agregar comentarios</a>	15 puntos	12 puntos	9 puntos	6 puntos	3 puntos	0 puntos
index <a href="#">Agregar comentarios</a>	20 puntos	16 puntos	12 puntos	8 puntos	4 puntos	0 puntos
manifest <a href="#">Agregar comentarios</a>	12 puntos	9.6 puntos	7.2 puntos	4.8 puntos	2.4 puntos	0 puntos
package.json <a href="#">Agregar comentarios</a>	12 puntos	9.6 puntos	7.2 puntos	4.8 puntos	2.4 puntos	0 puntos
ui5-local <a href="#">Agregar comentarios</a>	2 puntos	1.6 puntos	1.2 puntos	0.8 puntos	0.4 puntos	0 puntos

That is how we are going to evaluate the changes that you make in the files described along the workshop. You need to fill in the respective files with the information that we provide you, so that you complete the workshop.