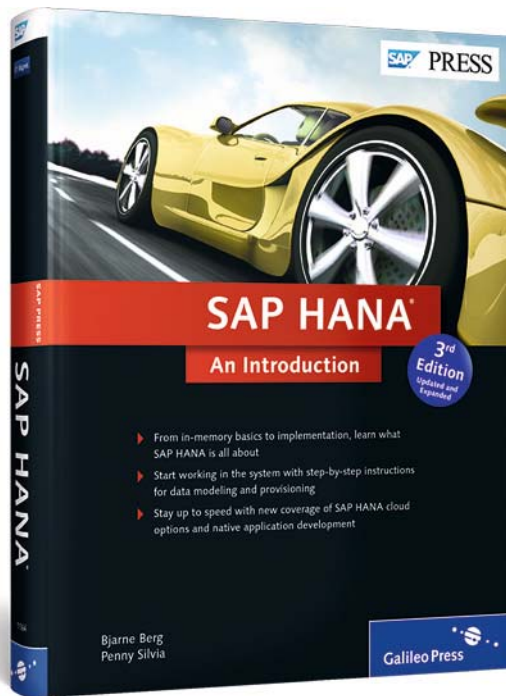


Dr. Berg and Penny Silvia

## Appendix A: SAPUI5 and SAP Fiori

from

SAP HANA®: An Introduction



  
Galileo Press

Bonn • Boston

## A SAPUI5 and SAP Fiori

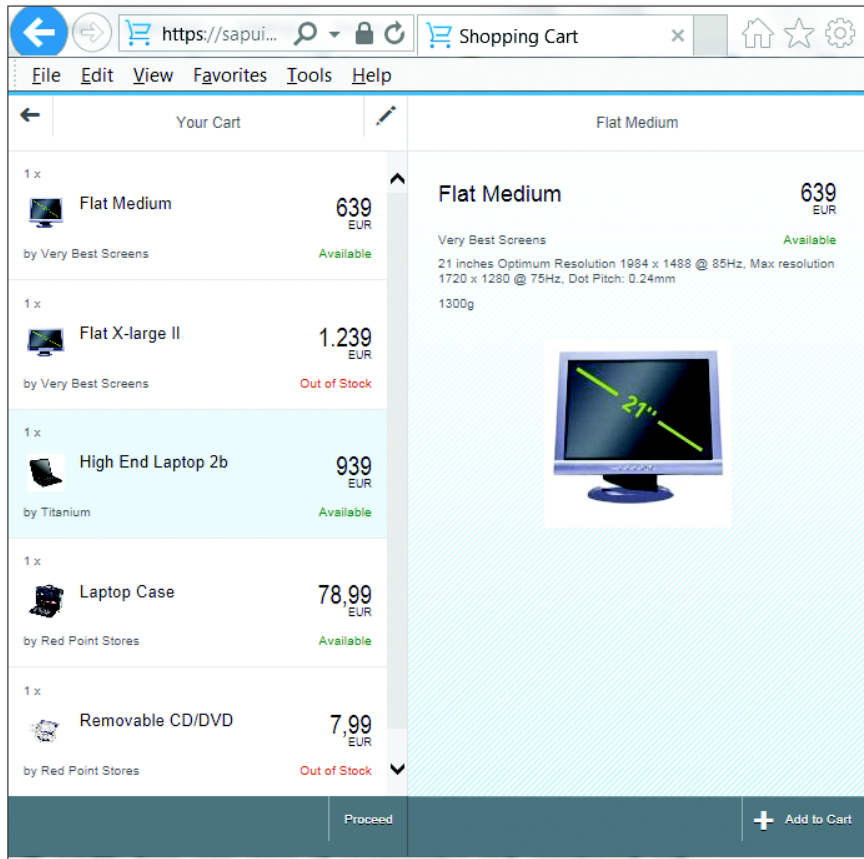
Although not a part of SAP HANA, SAPUI5 applications and SAP Fiori often run on top of SAP HANA. In this appendix, we'll give you some general information about SAPUI5 and SAP Fiori.

### A.1 SAPUI5 Programming Principles

In SAPUI5, a key development concept is that a basic UI element is a reusable entity with properties, relations, methods, and events. For a developer, the most important relations are aggregations to other UI elements. By aggregating, typically a tree structure of UI elements is created. In addition to the many controls, there are also elements that do not belong in the tree structure. Instead, these are dependent parts within the structure. This includes items in a table such as `TableCells` and `TableRows`.

From a data perspective, most properties have an associated data type, a default value, and a name that identifies it. These properties can be changed by the element's application programming interface (API), via *getters* that read properties and *setters* that set them. Some properties, depending on the element, can only be read. For example, in aggregation with a table you can modify the rows and columns in a table.

Websites work based on a similar concept. The highest element may be the web page, and then you can have different areas of the screen for UI purposes (e.g., order screen, display product, navigation bars). Inside the order area you may have a table of what has been ordered and each item with its own properties such as product name, price, image, and description. Even these elements have their own controls. For example, the image can have a label, a help text, and size definitions. It all works like a UI defined tree.



**Figure A.1** Simple Order Application Built by SAPUI5

There is just one way to master this concept, and that is to read the control documentation and then try it in a controlled sandbox environment. Over time you will learn new ways of programming, and experienced web developers often pick up on SAPUI5 in a few days. In addition, SAP offers the class “WDE350—Developing UIs using HTML5 and SAPUI5” for developers and architects.

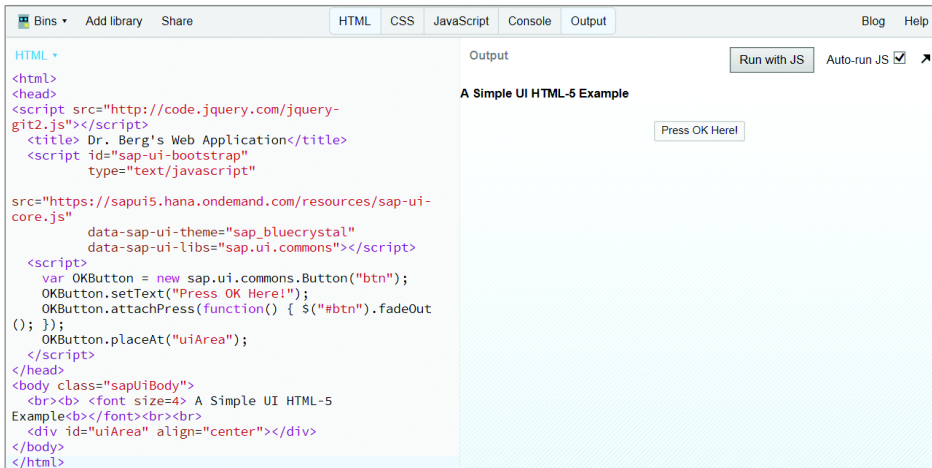
A recent change to SAPUI5 is that the code is now in open source mode. This means that you can extend existing controls, build new ones, and share these with others through the open source code principle. To allow this, SAP actually has created two versions of SAPUI5. The open source version is called *OpenUI5* and is free and available under the Apache 2.0 Open Source license. The other

version, which we have been covering so far, is called *SAPUI5* and is only available to licensed SAP customers.

The library that hosts the open source version is found in the JavaScript library named *OpenUI5*, and SAP is actively soliciting input and developer ideas on how to continue to enhance the code in this library. In addition, some open source libraries are also used in *SAPUI5* and are included with the toolkit. This includes *LESS* for cascading style sheet (CSS) processing and *datajs* as part of the *OData* library. So SAP is committed to the openness of *OpenUI5*, something that developers really seem to appreciate. Others may choose the *SAPUI5* version to get access to the bug-fixes and support that comes with the paid SAP license.

Also, the main *SAPUI5* libraries with its controls, including *sap.ui.commons*, *sap.ui.ux3*, responsive library *sap.m*, and all cores, are made available under the Apache license. While some of the charting libraries are not open source, the libraries above are the same as in the non-open-source versions.

If you are working in *OpenUI5*, you can simply invoke the libraries from an editor and use them as you want. On SAP's Github web page you will find a simple JS Bin editor that can introduce you to the basic concepts. If you are using this editor on the *OpenUI5* sample code we developed in Chapter 8, it would look like Figure A.2.



**Figure A.2** JS Bin Editor and Output with *OpenUI5*

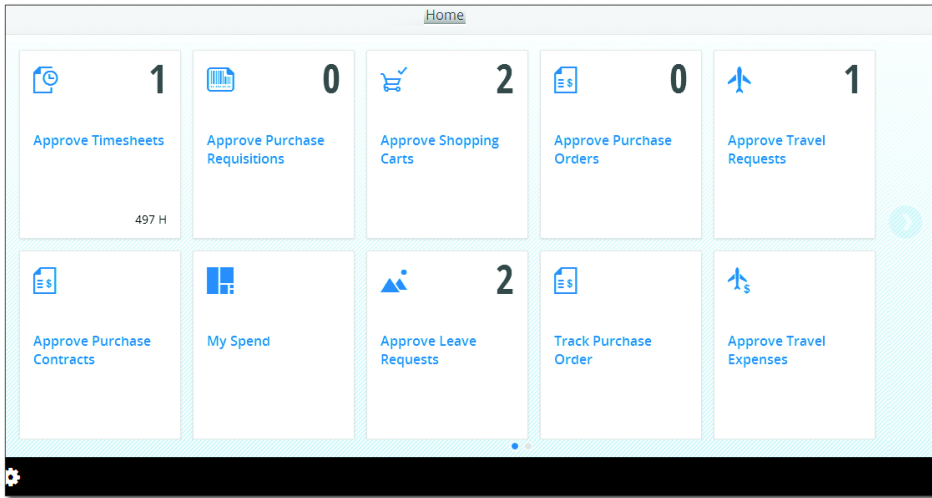
As of summer 2014, there is not yet a formal mechanism for contributing new code to the OpenUI5 library, but SAP is working on deploying the sources to Github, the collaborative source code web site, so that code can be forked and modified.

## A.2 SAP Fiori and Open Design

SAP Fiori is a new application set that was built on SAPUI5 on top of many applications, including SAP BW, SAP Business Suite, and SAP HANA models, and/or the applications already running on SAP HANA.

The idea with SAP Fiori, which means *flowers* in Italian, is that the user interface experience is separated from the backend applications and database functions. The tool simply calls the API's controls and programming elements to build simple end-user applications that can be deployed to laptops, tablets, and mobile phones in days instead of months of development work. The focus is basically on usability instead of system thinking.

This new concept makes sense. SAP has already mastered the system thinking, and now the focus is on *design thinking*. This has a different focus on user experience instead of systems. For example, frontend applications should not have to be constrained by where the data is. An end user does not care if the data is in SAP BW or in SAP Business Suite. The focus is on interaction, simplicity, agility, and fast creation of applications that users can access without having to write functional specifications.



**Figure A.3** SAP Fiori Applications Built with SAPUI5

While SAP has developed many SAP Fiori applications that you can buy, they also have created a framework for thinking about applications. This includes seven types of applications:

► **Analytical applications**

These are applications that show key performance indicators (KPIs) and summarized data on the top tile on the home screen. From there, users can access applications that allow for drilldown, variance reporting, news summaries, and links to other analytical applications to explore business trends and performance measures.

► **Approval applications**

Much of the business interaction for managers includes approving and/or rejecting requests in daily process workflows such as vacation requests, travel requests, expense reports, purchase orders, and so on. This type of application simplifies that by providing alert-based information that is pushed to the decision maker, instead of waiting until the manager is back in the office, logged on, and has run the right report. These types of applications can significantly improve operational efficiencies at a company.

► **Change applications**

Change applications are basically applications that allow you to change the editable fields on an application. For example, you can reach a set of change

actions on the footer of the details page in an application. Once you click on the action, you will see a dialog that shows what fields you can change. This is useful when you need to update master data such as customer-preferred payment methods, addresses, phone numbers, and other contact information. It can also be used for other data updates.

► **Factsheet applications**

A factsheet application is a short summarized sheet on an application that shows the main data points about an object. For example, this can be a simple overview of a customer's total credit line amount, the number of times payments were past due, and the total value of orders being processed but not yet shipped. This gives the user a simple way to access information without being bogged down with too much information. The factsheet can be for any business object, and links can be provided for more drilldown to transactions, reports, and/or other factsheet applications.

► **News applications**

The news application is a list of relevant announcements and contextual news that can be targeted to users by administrators and publishers. News applications can be used to define variants and many different data sources that feed the application. For example, you could use external news feeds and subscriptions, as well as alerts from systems such as SAP BW and SAP Business Suite. As alerts, announcements, and news are published, the feeds are placed on top of the application so that you always see the most recent news.

► **Process applications**

Process applications are often associated with workflow and business processes. These are simply applications that take you through scenarios and tasks that are often repetitive, for example, converting planned production orders to released production orders and turning internal orders into external purchasing orders. In short, process applications are applications that let you do these tasks in smaller dedicated applications through structured step-by-step processes.

► **Shopping applications**

This type of application lets users look at products; search; sort by features, price, and availability; as well as select items for purchase and execute payment transactions. In short, this is the type of application that is most common for e-commerce.

Having a basic idea of the purpose of these many types of applications makes it easier to classify existing applications and to communicate and organize the application development within your organization. It is also the same way SAP communicates their applications in the various modules and solutions.