

First Steps for Building SAP®UI5 Mobile Apps

Robert Burdwell

Table of Contents

Introduction	9
1 Introduction to Mobile Development	11
1.1 History of smart phones	11
1.2 Native or hybrid	12
1.3 HTML	13
1.4 CSS	14
1.5 JavaScript	15
1.6 XML	16
1.7 Cordova	18
1.8 OData	18
2 SAPUI5 Development	21
2.1 Open UI5	22
2.2 SAPUI5 Development Toolkit	23
2.3 ListBox	24
2.4 SAPUI5 libraries	27
2.5 Themes	30
3 Model View Controller (MVC)	33
3.1 HTML Example	35
3.2 View example	38
3.3 Model example	39
3.4 Controller example	41
4 SAPUI5 apps with Eclipse	43
4.1 Install Eclipse Java Development Kit (JDK)	43
4.2 SAP development tools	44
4.3 Push button and text app	49

4.4	SAP Development tools for Eclipse	55
4.5	Create OData web service	57
5	SAP Mobile Platform Server	61
5.1	SMP server purchase options	63
5.2	SAP Service Marketplace	64
5.3	SMP Server preparation	66
5.4	Connect Eclipse to SAP Mobile Platform Administration and Monitoring	72
5.5	SMP Server new application	73
5.6	Gateway Management Cockpit	76
6	SAP Mobile Platform SDK	79
6.1	SMP SDK for Visual Studio	81
7	SAPUI5 with SAP Cloud Platform	85
7.1	Purchase options	85
7.2	Logging into the SAP Cloud Platform	86
8	Building Apps with the SAP Cloud Platform	95
8.1	Create app – SAP Cloud Platform	95
8.2	Create app in SAP Web IDE	97
8.3	Layout Editor	106
8.4	Odata app	110
8.5	Create, import, route and deploy app	116
8.6	Git clone app	123
8.7	Delete app	124
9	Building SAPUI5 Apps with Visual Studio	127
9.1	New project	128
9.2	Kapsel	135
9.3	Visual Studio market place	140
9.4	Git	141
9.5	OpenUI5-sources	143

10 Resources	145
11 References	146
A The Author	148
B Index	149
C Disclaimer	150

2 SAPUI5 Development

SAP has been around since the 1970s, so there is value and demand for building mobile apps with a focus on SAP transactional data. SAPUI5 uses an SAP library and HTML5 for application and mobile application development. SAPUI5 can support XML, JSON and ODATA. You will find in the development that the views are created with XML.

The examples below show the initial index page when creating an HTML5 or SAPUI5 project. The content in this chapter covers each section in more detail, but it is good to compare the structure of an index.html page with just HTML5 code versus an index.html page with SAPUI5. Notice the additional `<script>` tag, JavaScript variable and CSS needed to develop the project using SAPUI5. Obviously, an HTML5 example would have `<script>` tags, CSS and JavaScript embedded in the code, but the intent here is to show the basic foundation needed to develop an app with SAPUI5. You can see the similarities with the DOCTYPE, html, head and body.

HTML5 example

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title></title>
</head>
<body>

</body>
</html>
```

SAPUI5 Example

```
<!DOCTYPE HTML>
<html>
  <head>
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>

    <script src="resources/sap-ui-core.js"
            id="sap-ui-bootstrap"
            data-sap-ui-libs="sap.m"
            data-sap-ui-theme="sap_bluecrystal">
    </script>
    <!-- only load the mobile lib "sap.m" and the "sap_bluecrystal" theme -->
  >

  <script>
    sap.ui.localResources("starter_app");
    var app = new sap.m.App({initialPage:"idMain1"});
    var page = sap.ui.view({id:"idMain1",
viewName:"starter_app.Main", type:sap.ui.core.mvc.ViewType.XML});
    app.addPage(page);
    app.placeAt("content");
  </script>

  </head>
  <body class="sapUiBody" role="application">
    <div id="content"></div>
  </body>
</html>
```

2.1 Open UI5

SAPUI5 is the focus of app development, but there is also an option to use *OpenUI5*. This is available for everyone, whereas SAPUI5 is available to developers using platforms that support SAP customers. The majority of the controls are available in both environments, with the exception of new editions which are usually featured in the SAPUI5 Development Toolkit for HTML5 (see Chapter 10).

OpenUI5



This is an open source JavaScript library for building enterprise applications. The open source environment provides a software development kit (SDK) for development needs.

SAPUI5 is very flexible because it can be used with *Eclipse*, *SAP Cloud Platform* and *Visual Studio*, along with many other development options. Table 2.1 lists the libraries supported for SAPUI5. The `sap.m` library is used in this book because it contains the necessary controls for mobile app development. Mobile apps also use `sap.ui` for user experience and standard controls.

<code>sap.ui.comp</code>	<code>sap.m</code>	<code>sap.table</code>
<code>sap.ca</code>	<code>sap.me</code>	<code>sap.ushell</code>
<code>sap.uxap</code>	<code>sap.gantt</code>	<code>sap.ui.core</code>
<code>sap.ui.layout</code>	<code>sap.ui.unified</code>	<code>sap.ui.table</code>
<code>sap.viz</code>		

Table 2.1: SAPUI5 libraries

The namespace `sap.m` is used for mobile devices and desktops. A list of all available controls is located under the API Reference tab in the SAPUI5 Development Toolkit.

2.2 SAPUI5 Development Toolkit

SAPUI5 offers a dynamic demo kit for SAP developers. Developers have the option to view code and examples for *Action*, *Chart*, *Container*, *Data Binding*, *Display*, *Layout*, *List*, *Map*, *Popup*, *Routing*, *Testing*, *Theme* or *Tile* using the SAPUI5 Development Toolkit. Samples of code snippets or solutions can be accessed through the SAPUI5 Development Toolkit. The Development Toolkit helps developers test examples and then use the code for their mobile app development. SAP's willingness to provide this demo kit increases the use and acceptance of the SAPUI5 framework. Table 2.2 outlines the tab options available with the SAPUI5 Development Toolkit.

Tab	Description
DEVELOPER GUIDE	Provides a list of supports such as Tutorials, Essentials, Developing Apps Documentation and Templates
EXPLORED	Offers control examples outlining properties, events and methods
API REFERENCE	Provides a list of controls with the namespace <code>sap.ui</code> , <code>sap.m</code> , <code>sap.chart</code> , <code>sap.m.routing</code> , along with many others
DEMO APPS	Provides over twelve sample apps that can be explored, with the option to download the source code to import into the development environment
ICONS	Offers options for common icons used in the SAPUI5 apps
CONTROLS	Provides a gallery of simple to advanced controls

Table 2.2: SAPUI5 Development Toolkit

As you can see, there are plenty of learning opportunities and code examples to build apps. This can sometimes be an issue when considering a new mobile development environment – for instance, there is an abundance of mobile development options, but few offer code examples, demonstrations, templates and libraries to this degree.

2.3 ListBox

List boxes and items allow us to organize content as sections to be displayed on the page. As seen in the example below, the SAPUI5 Development Toolkit provides the structure and example of the ListBox code. It is important to note that this is only the JavaScript, which can be nested in the `index.html` page or listed as a function in the `controller.js` page. We will discuss the controllers more in the Model View Controller (MVC) section.

Exercise – ListBox review



1. Open the *SAP Demo Kit* (see Chapter 10)
2. Click on the CONTROLS tab
3. Click on the VALUEHOLDERS section, expand the selection
4. Click on LISTBOX
5. View the sample code snippets under the Examples section

The example of the ListBox below shows the items, function for the text summary and multi selection checkbox. We will not use this specific ListBox in any tutorials, but it is good know that this code snippet is available for future needs.

ListBox example

```
// Create the ListBox
var oSimpleListBox = new sap.ui.commons.ListBox({
    items: [
        new sap.ui.core.ListItem({ text: 'Spring' }),
        new sap.ui.core.ListItem({ text: 'Summer' }),
        new sap.ui.core.ListItem({ text: 'Autumn' }),
        new sap.ui.core.ListItem({ text: 'Winter' })
    ],
    select: function (e) {
        var a = ["Your favorite season(s): "];
        jQuery.each(oSimpleListBox.getSelectedItems(),
function (idx, item) { a.push(item.getText(), ","); });
        oTextSummary.setText(a.join(""));
    }
}).placeAt("sample1");
var oTextSummary = new
sap.ui.commons.TextView().placeAt("sample1");
var oMultiSelectionCheckBox = new sap.ui.commons.CheckBox({
    text: "MultiSelection",
    checked: false,
```

B Index

A

Agentry 63

C

Cordova 18, 130

CSS 12, 14

E

Eclipse 43, 62, 77

H

HTML 13

HTML5 12, 13, 96

I

IDE 79

Internet of Things 89

J

Java 87

K

Kapsel 63, 135

M

MVC 33

O

OData 18, 57, 60, 115

OpenUI5 22

S

SAP Cloud Platform 77, 85, 95

SAP Gateway 76

SAP Mobile 61

SAP Mobile SDK 79

SAP Mobile Server Toolkit 57

SAP Web IDE 97, 106

SAPUI5 21, 22, 81, 97, 132

security 92

SMP 3.0 62, 64

SMP Mobile Platform Server 83

SMP SDK 64, 79, 83

SMP Server 66, 70, 76

SMP Server Administration and
Monitoring 71

V

Visual Studio 127

X

XML 16