

How to Guide

PUBLIC

SAP Business One mobile app 1.10.x for iOS and SAP  
Business One mobile app 1.1.5 for Android  
Document Version: 1.1 – 2014-09-03

# How to Create Add-Ons for the SAP Business One Mobile App for iOS and Android

Applicable Releases: SAP Business One mobile app 1.10.x for iOS, SAP Business One mobile app 1.1.5 for Android, SAP Business One 9.0 PL12, SAP Business One 9.0, Version for SAP HANA PL12, SAP Business One Cloud 1.1 PL02

# Typographic Conventions

Type Style	Description
<i>Example</i>	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Textual cross-references to other documents.
<b>Example</b>	Emphasized words or expressions.
<b>EXAMPLE</b>	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
<code>Example</code>	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
<b>Example</b>	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
<b>&lt;Example&gt;</b>	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
<b>EXAMPLE</b>	Keys on the keyboard, for example, <b>F2</b> or <b>ENTER</b> .

---

# Document History

Version	Date	Change
1.0	<2014-06-12>	<New>
1.1	<2014-09-03>	Re-organized the contents and added Android app support

---

# Table of Contents

Introduction .....	5
Prerequisites.....	5
Creating Web Pages for Mobile Devices.....	6
Deploying mobile add-ons on Web servers.....	6
Deploying mobile add-ons on the Integration Framework for SAP Business One Server .....	6
Implementing Back-end Web Services .....	7
Setting up to create Web services in the Integration Framework.....	7
Calling mobile Web services in Web pages .....	9
Deploying Add-ons for SAP Business One mobile app for iOS and Android.....	10
Communicating with the SAP Business One Mobile App for iOS and Android.....	12
Scheme Syntax.....	12
Samples.....	13

---

# Introduction

Add-ons for the SAP Business One mobile app for iOS and Android are Web applications. Through this document, you can learn to create and deploy add-ons.

This document explains the following:

- How to create Web pages for mobile devices
- How to implement back-end Web services
- How to deploy the add-ons for SAP Business One mobile app for iOS and Android
- How to communicate with SAP Business One mobile app for iOS and Android

## Prerequisites

- You know the following coding languages:
  - HTML
  - CSS
  - JavaScript
  - XML
- You know how to create a project and how to implement a responsive UI, such as a list, by using any 3<sup>rd</sup> party library or framework, or by working from scratch.

---

# Creating Web Pages for Mobile Devices

You can take advantage of any 3rd party Web-based mobile framework such as jQuery, Sencha, PhoneGap and so on. Most of these Web-based mobile frameworks provide responsive technologies to help developers easily consume the content for different form factors.

Useful reference links:

- <http://jquerymobile.com>
- <http://www.sencha.com/products/touch>
- <http://phonegap.com>
- <http://www.telerik.com/kendo-ui>
- <http://mobileangularui.com>
- <http://www.appcelerator.com/titanium>

## Deploying mobile add-ons on Web servers

A standard mobile add-on includes several Web pages and one of them should serve as an entry page. We recommend storing these pages in a folder and deploying them on your specified Web server to make sure that all the Web pages can be visited through an external URL in a mobile browser.

## Deploying mobile add-ons on the Integration Framework for SAP Business One Server

On the integration framework server, we recommend locating the mobile add-ons under the *webapps* folder in the integration framework installation folder.

For example: `C:\Program Files\SAP\SAP Business One Integration\B1iServer\tomcat\webapps\<my add-on folder>`

After that, make sure you can access the add-on via a URL in a mobile browser and that this URL is registered in SAP Business One *Administration* → *Add-Ons* → *Mobile Add-On* Module (details provided in a later section).

For example: `http://<my Integration Framework Server IP Address>:<port>/<my add-on folder>/<my entry page>.html`

---

# Implementing Back-end Web Services

This document shows how to implement Web services on top of the integration framework for SAP Business One. If you do not want to develop services on the integration framework, but on top of other Web servers such as Tomcat or IIS, you must know the related technologies, such as Java, .NET or other server-side development languages and related deployment knowledge.

## Setting up to create Web services in the Integration Framework

As of SAP Business One 8.81 release, the integration framework is part of the SAP Business One installation. After installation, it can be launched by clicking *Integration Framework* under *Integration solution for SAP Business One* in Start Menu → All Programs. It is a complete environment for mobile back-end server configuration, maintenance and development. After launching in a Web browser, you see several menus at the top: *SLD*, *Maintenance*, *Scenarios*, *Setup*, *Monitoring*, *Help*, and etc.

The following briefly introduce the functions under the integration framework main menus that are closely related to the mobile Web service setup. For more detailed information, refer to *Help → Documents* in the integration framework.

### SLD

In this page, in the left panel, under *B1i Server* click *WSforMobile*. Make sure that the address of the relevant SAP Business One server appears in the *associatedSrvIP* field

### Maintenance

This page contains an entry in the left panel, *Cfg Dev Environment*, which should be configured before mobile scenario development.

1. For *Mode*, set *Vendor*.
2. Specify values for *Development Prefix* and *Development Long Description*.
3. Define a configuration for further development, such as "Allow step modification in active phase".
4. Choose the *Save* button to finish the preparation.

### Scenarios

This page contains an entry in the left panel, *Package Design*, which deals with scenario definition. All newly created backend interfaces are under this scenario package. Go to this entry, and in the right panel, you can do the following:

1. Specify a new *Scenario Package Identifier*.
2. For *Authentication*, choose *User Authentication (sap.B1Mobile)*.
3. Specify values for *Vendor*, *Version* and *Description*.
4. Choose the *Save* button to create this scenario package.

The left panel contains an entry, [Step Design](#), which is for specific Web service detail design. Go to this entry, and in right panel, you can do the following:

1. Specify a new [Scenario Step Identifier](#).
2. For [Scenario Package Identifier](#), select a newly created vendor scenario package.
3. Choose the [Inbound](#) button and then choose the [Channel](#) button.
4. In the [Scenario Step Definition - INBOUND - CHANNEL](#) window, do the following:
  1. For [Inbound Type](#), choose `Web Service Call`.
  2. For [Process Mode](#), choose `Synchronous`.
  3. For [Identification Method](#), choose `Root Tag`.
  4. For [Identifier](#), specify a tag name\*.
  5. For [Identification Namespace](#), specify a namespace\*.
  6. Choose the [Save](#) and then the [Close](#) buttons to finish the initialization.
5. Click the [Processing](#) button to implement detail business logic by defining BizFlow Atom, such as "call B1 Object", "call SQL", "XSL Transformation", "Conversion XML to JSON", and so on.

\* This tag name and namespace are used for handling corresponding inbound HTTP requests. For example, if the tag name is "GetEmployeeList" and namespace is "xmlns="http://tempuri.org/"", the following request will be handled by this backend interface:

```
<soap:Envelope>
  <soap:Body>
    <GetEmployeeList xmlns="http://tempuri.org/">
    </GetEmployeeList>
  </soap:Body>
</soap:Envelope>
```

For more detailed information, refer to [Help → Online Help → Scenario Development → Develop a Scenario Step](#) and [Help → Online Help → Development Environment](#).

## Setup

In this page, you can do the scenario package setup after creating new backend interfaces in [Step Design](#):

1. Choose the new package identifier.
2. Choose the [Steps](#) button to make sure all the required steps (interfaces) have been checked.
3. Choose the [Sender](#) button.
  - o For [Choose entry](#), choose `Main Sender List` and choose [Select](#).
  - o Select **ws-001sap0003(WSforMobile)**; then choose the [Save](#) and the [Close](#) buttons.
4. If everything is correct from the development phase, choose the [Activate](#) button.

For more detailed information, refer to [Help → Online Help → Scenario Development and Help → Online Help → Operations → Scenarios Administration → Setup & Activation](#).

## Monitoring

In this page, the left panel contains an entry, [Message Log](#), which you can use for log analysis. For more detailed information, refer to [Help → Online Help → Operations → Monitoring](#).



## Calling mobile Web services in Web pages

Refer to *B1Utility.SOAPHelper* interface in the sample `B1Utility.js` file. You can find the file in [https://partneredge.sap.com/content/partnerexp/en/library/assets/products/ent\\_mng/sbo/about/sbo\\_mobile/versions/sbo\\_mobile\\_ios/tr\\_sbo\\_mobile\\_ios\\_sc.html](https://partneredge.sap.com/content/partnerexp/en/library/assets/products/ent_mng/sbo/about/sbo_mobile/versions/sbo_mobile_ios/tr_sbo_mobile_ios_sc.html)

Name	Type	Explanation
Property		
Url (+)	String	Web Service URL. If you deploy the back-end services on the integration framework, there is no need to change the URL provided by default in the sample code. If you want to call other web services (not by the integration framework), you need to change this URL accordingly.
SOAPMethod(*)	String	Name of the Web service operation.
HTTPMethod(+)	String	The common HTTP method. If not specified, POST is the default value.
Args(+)	Array	An array of name-value pairs composed of the parameters intended to be passed to the Web service operation.  Sample: [ <code>{name: arg1Name, value: arg1Value}</code> , <code>{name: arg2Name, value: arg2Value}</code> ]
RequestCallBack(*)	Function	A callback function of the HTTP Request.
Method		
SOAPHelper(string, function, array)	Constructor	To initialize the object. <i>1st argument(*)</i> : Name of Web Service operation <i>2nd argument(*)</i> : The callback function of HTTP Request <i>3rd argument(+)</i> : Name and value of argument requested by the Web Service operation
Start()	Method	Start to send the request

(\*) is mandatory, (+) is optional

# Deploying Add-ons for SAP Business One mobile app for iOS and Android

In order to deploy Web-based add-ons, make the settings described below.

1. From the SAP Business One *Main Menu*, choose *Administration* → *Add-Ons* → *Mobile Add-On*.
2. In the *Mobile Add-On - Setup* window, enter the relevant data in the *Code*, *Name*, and *Provider* fields.
3. In the *Entry URL* field, enter the external accessible URL of the add-on entry Web page that can be loaded from a mobile browser
 

For add-ons hosted on SAP Business One integration server, the URL is similar to <http://<my Integration Framework Server IP Address>:<port>/<my add-on folder>/<my entry page>.html>.
4. In the *Type* field, specify whether the add-on will be accessed from the *Home* tab or from the *Modules* tab on the iPhone (equivalent to the *Home* tab and the vertical modules menu in the menu section on the left side of the iPad).
  - For Android apps, this field is ignored so that all the add-ons on Android will be accessed only from the menu.
5. In the *View Style* field, specify the relevant view:
  - *Page-Universal* - for add-ons that work in iPhone portrait mode, iPad page sheet mode (similar to *Home Type* view), and Android
  - *Full Screen - iPad* - for add-ons that work only in iPad portrait and landscape modes
  - *Landscape Only - iPad* - for add-ons that work only in iPad landscape mode

View Style	iPhone		iPad			Android	
	Home View	Module View	Home View	Module Page View	Module Full Screen View (Landscape Only)	Module Full Screen View (Portrait and Landscape)	Module Page View
Page – Universal	√	√	√	√			√
Full Screen – iPad						√	
Landscape Only – iPad					√		

6. In the *Logon Method* field, specify whether the add-on requires authentication for logon:
  - *B1i Framework* - Select this option if add-on back-end scenarios are implemented on the Integration Framework and the same session ID will be shared with this add-on. Choose this option only if the add-on back-end scenarios are implemented on integration framework.
  - *Standard Logon* - Select this option for add-on that requires user logon authentication, in which case a logon message box appears requesting user name and password.

- 
- o *No Control* - Select this option for add-ons that do not require any authentication to be provided by SAP Business One or for add-ons that control independently everything that has to do with authentication.
7. To save your settings, choose *Update*.

# Communicating with the SAP Business One Mobile App for iOS and Android

SAP Business One mobile app for iOS and Android allows add-ons to communicate with predefined scheme URLs so that an add-on can access existing functions seamlessly. The samples can be found in the `B1Utility.js` file.

## Scheme Syntax

The scheme is as follows:

**sapb1://<operation>? [<command parameter>]**

- `<operation> ::= call`  
operation is to specify the behavior of a general action
- `[<command parameter>] ::= cmd=<Data Code> & [key=<Key Code> | filter=<Filter Options>]`

The command parameter is to specify the data view or a special action. It contains 3 parameters: cmd, key, and filter. cmd is mandatory. In addition, either the key or the filter parameter is required.

Function Type	cmd Parameters	Key Parameters	Filter Parameters
Business Partner	OCRD	BP Code	customer   vendor   lead   all
Inventory Item	OITM	Item Code	all
Opportunity	OOPR	Opportunity ID	open   won   lost
Sales Order	ORDR	Sales Order Doc Key	open   my   all
Sales Quotation	OQUT	Sales Quotation Doc Key	open   my   all
Service Contract	OCTR	Service Contract ID	all
Service Call	OSCL	Service Call ID	open   pending   my   all
Report	RDOC	Report ID	
Customer Equipment Card	OINS	Item Serial Number	
Approval	OWDD	Approval ID	all
Activity	OCLG	Activity ID	my   <BP Code>
Alert	OALT	Alert ID	all
Close the add-on	backToApp		
Handle Time out	timeout		

---

## Samples

- Show business partner view (BP code is C20000)  
sapb1://call?cmd=OCRD&key=C20000
- Show all the customers in a list  
sapb1://call?cmd=OCRD&filter=customer
- Show all the inventory items  
sapb1://call?cmd=OITM&filter=all
- Close the add-on  
sapb1://call?cmd=backToApp

[www.sap.com/contactsap](http://www.sap.com/contactsap)

Material Number

© 2014 SAP SE. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Excel, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, System ads, System i5, System p, System p5, System x, System z, System z10, System z9, z10, z9, ISeries, pSeries, xSeries, zSeries, eServer, z/VM, z/OS, i5/OS, S/390, OS/390, OS/400, AS/400, S/390 Parallel Enterprise Server, PowerVM, Power Architecture, POWER6+, POWER6, POWER5+, POWER5, POWER, OpenPower, PowerPC, BatchPipes, BladeCenter, System Storage, GPFS, HACMP, RETAIN, DB2 Connect, RACF, Redbooks, OS/2, Parallel Sysplex, MVS/ESA, AIX, Intelligent Miner, WebSphere, Netfinity, Tivoli and Informix are trademarks or registered trademarks of IBM Corporation.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, R/3, xApps, xApp, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.