



Introduction to OpenUI5

Mobile Web App Development



SAP InnoJam - TechFire



**What is OpenUI5
all about?**

UI development toolkit for HTML5

Open Source Technologies

Built with Leading Web Technologies...

HTML5	CSS3
jQuery	LESS
OData	D3.js
OpenAJAX	ARIA

...and became Open Source @ 11/12/2013



<https://sap.github.io/openui5/>

Difference between SAPUI5 & OpenUI5: no charts in OpenUI5, SAP Web IDE uses SAPUI5

UI development toolkit for HTML5

Responsive Web Design



Examples of SAPUI5 apps

SAP Fiori

←

Item 1 of 1

V CrossArm 65x65x6 mm

62,000.00 USD

100 Each

Standard

Information

Material Group: Mechanics (00104)

Delivery on January 28, 2016

Plant: Atlanta
Address: 867 1 300 Peach tree street South, ATLANTA GA 30310, United States

Media%20Store Sales Orders

Sales Order

Search

SO 17118 20K USD
PO TFG-062615-00
June 26, 2015
Shipped

SO 5003235 4K USD
PO n/a
April 24, 2015
Not Shipped

SO 5003231 2K USD
PO n/a
April 22, 2015
Not Shipped

SO 16762 799.00 USD
PO n/a

Sales Order 17118 19,990.00 USD

PO: TFG-062615-00
Ordered: June 26, 2015
Requested: June 27, 2015
Ship To: Media Store
Contacts

Order Total: 19,990.00 USD
Overall Status: Completed
Delivery Status: Shipped

1 Item

Open In Process Shipped

Description	Ordered	Price	Net Amount	Status Summary	Shipped
10 Notebook Pro 15	10 ST	1,999.00 USD	19,990.00 USD	Shipped	10 ST

Product Details

Samsung 40" Class 720p LCD HDTV 10 ST
Counted Nov 25, 2015

Product Information

Product No.: 411111
GTIN: 2150000000000
Retail Price: 169.00 USD

Count Adjust

My Timesheet

Jan 2016

MON TUE WED THU FRI SAT SUN MON TUE WED THU FRI SAT SUN
25 26 27 28 29 30 31 1 2 3 4 5 6 7

Jan 25 - Jan 31 58:00 hours
2 time assignments
Target: 36:40 hours 16:00 hour...

Feb 1 - Feb 7 65:00 hours
2 time assignments
Target: 36:00 hours 27:00 hour...

☐ Time Assignment Duration Status

Jan 25 (36:00 hours / 08:00 hours)

☐ Attendance Hours 08:00 Sent for approval

Jan 26 (36:00 hours / 08:00 hours)

☐ Vacation 08:00 Sent for approval

Jan 27 (10:00 hours / 07:59 hours)

Purchase Orders (4)

Search

Industrial Suppliers 62,000.00 USD
V CrossArm 65x65x6 mm
Nelson Negotiator
Today

Engineering Suppliers 378,860.00 USD
HVAC Duct Wrap Insulation
Kate Roberts
Today

IDES SUS Vendor 300.00 EUR
11Mbps Wireless PCMCIA...
Matthew Black
Today

C.E.B. New York 4,595.00 USD
2 Items
Paula Purchaser
Today

Purchase Order

Industrial Suppliers 62,000.00 USD
Nelson Negotiator
Today

Information

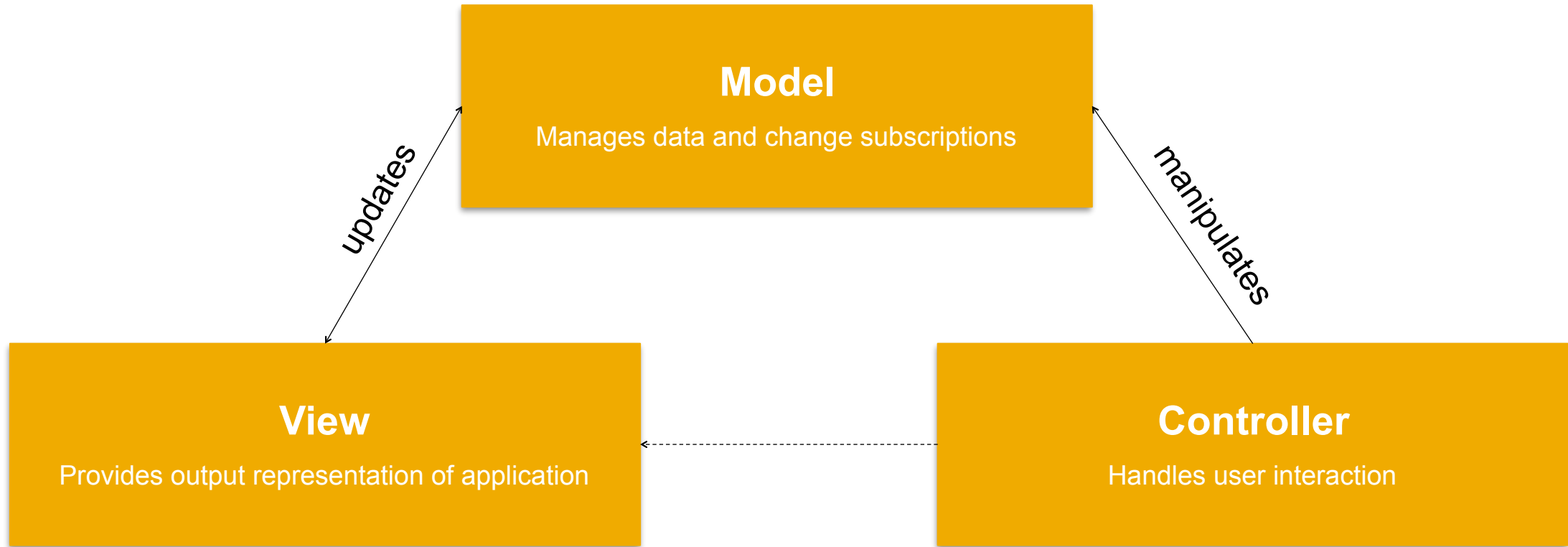
Purchase Order: 4500026217
Account Assignment: Cost Center Purchasing 2 (9362)
Profit Center: G/L Account Raw Materials - Others (400000)
Delivery Date: January 28, 2016
Plant: Atlanta
Incoterms: Costs and freight (CPR), CPR
Company Code: BestRun USA (3000)

Items (1)

Description	Item Category	Quantity	Subtotal
V CrossArm 65x65x6 mm	Standard	100 Each	62,000.00 USD

Approve Reject Forward

Model-View-Controller Concept



Model, View, Controller in detail

XML-based View

Collection of UI controls, positioned using layouts and containers

Automatic re-rendering whenever a property (1:1) or aggregation (1:N) changes

JavaScript-based Controller

Used for handling UI events and for setting up the view

Data Model

Supported data sources: JSON, XML, OData, Resource Bundles (translations)

Loads, holds, and updates all application data

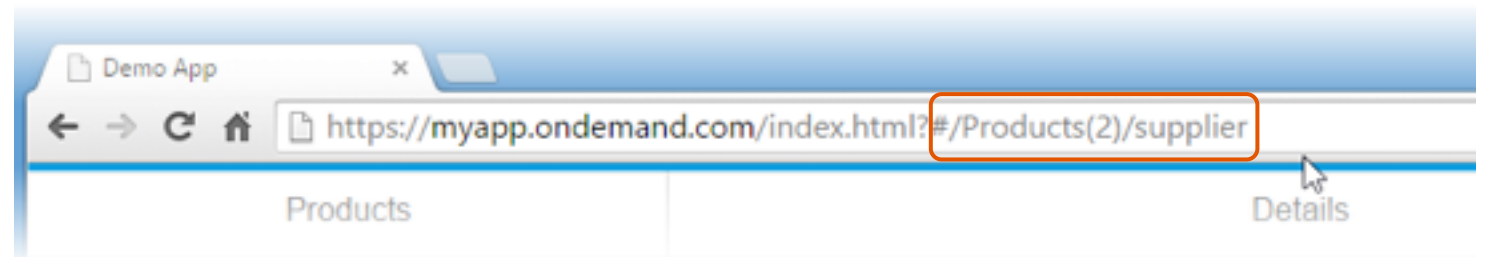
Data is attached (bound) to controls through data bindings.

Routing 101

Routing is a mechanism for in-app navigation for supporting...

... browser history handling

... bookmarks/ deep links



How routing works:

Your web app component will define routes, such as

product → **#/product**

product/{id} → **#/product/5** or **#/product/3**

product/{id}/detail/:detailId: → **#/product/5/detail/3** or **#/product/5/detail**

When a route matches, the shown view is switched out and an event is thrown.



How to start

Prepare Local Development Environment

Text editor / IDE of your choice

```
<!DOCTYPE html>
<html>
<head>

  <meta http-equiv="X-UA-Compatible" content="IE=edge"/>
  <meta charset="UTF-8">
  <title>Hello World App</title>
  <script src="https://sapui5.hana.ondemand.com/resources/sap-ui-core.js"
    id="sap-ui-bootstrap"
    data-sap-ui-theme="sap_bluecrystal"
    data-sap-ui-libs="sap.m">
  </script>
  <script type="text/javascript">
    sap.ui.getCore().attachInit(function () {
      // your coding here
    });
  </script>
</head>
<body class="sapUiBody" id="content">
</body>
</html>
```

Use a http server!!!

(serving sapui5 app, pictures/JSON files)

python -m SimpleHTTPServer

→ **http://localhost:8000**

(or use node.js http-server port 8080)

If you work with Eclipse, use tomcat

Brackets Editor has a built in http-server

Accessing remote resources (Cross origin)

"C:\Program Files (x86)\Google\Chrome\Application\chrome.exe" **--disable-web-security**

If you work with Eclipse, than use a proxy servlet



Example



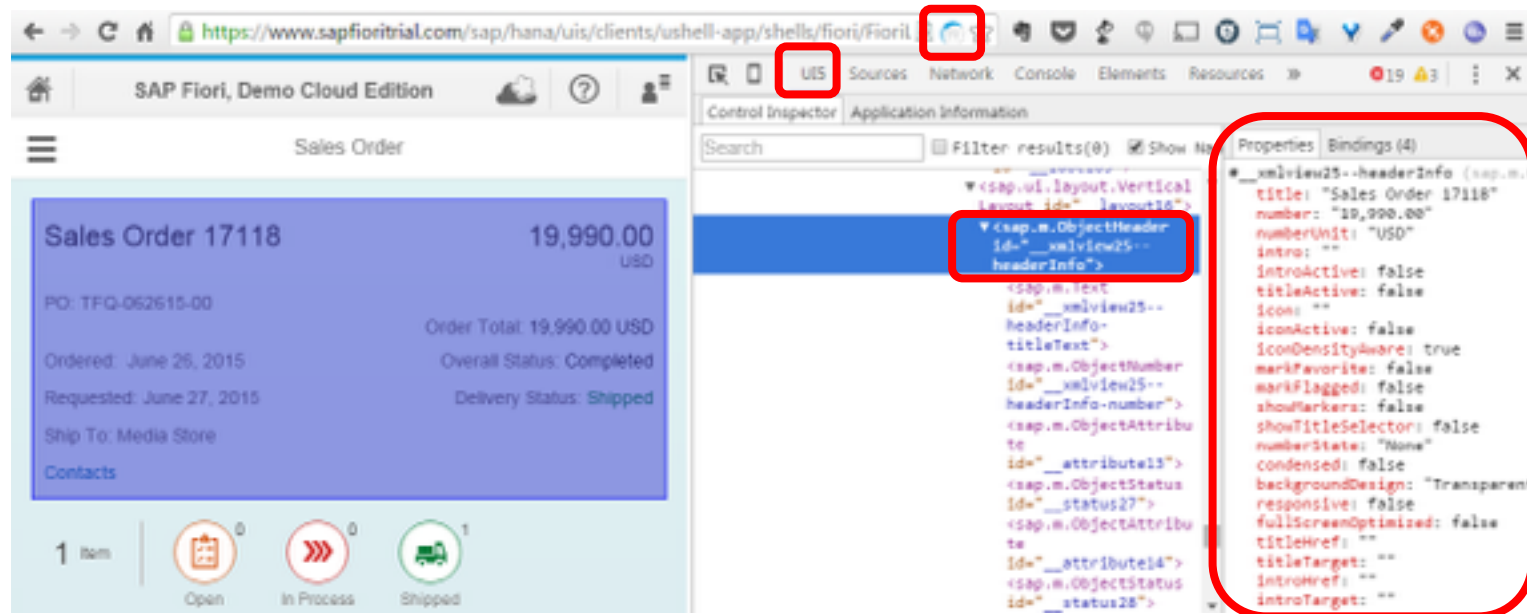
Tips for SAP InnoJam

Development tips 1

Use the SAPUI5 Chrome Inspector

Debug your apps, inspect other apps

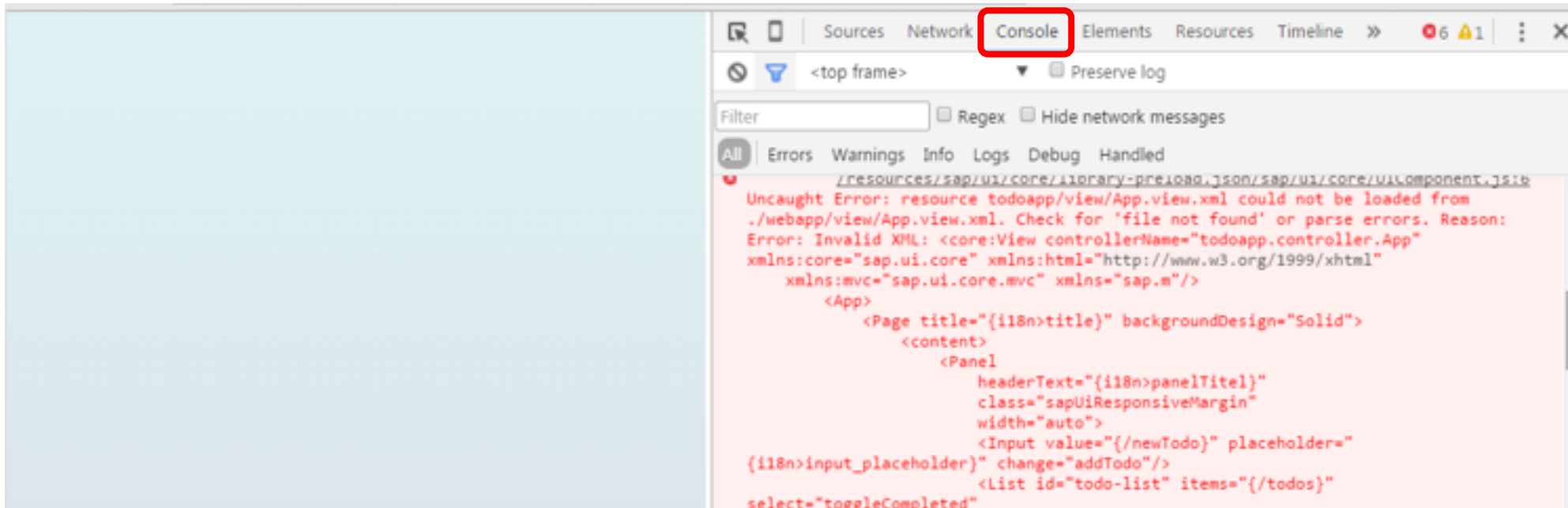
<https://github.com/SAP/ui5-inspector>



Development tips 2

Bluescreen?

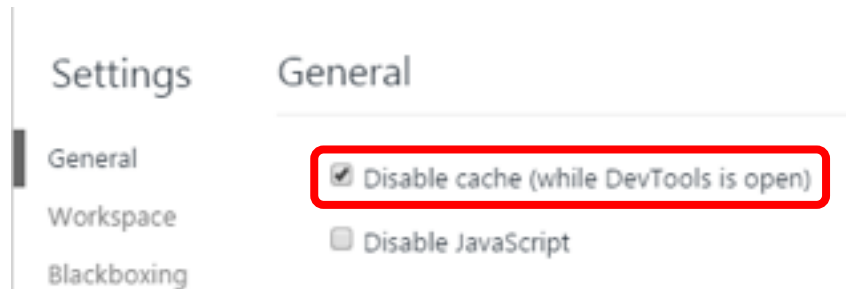
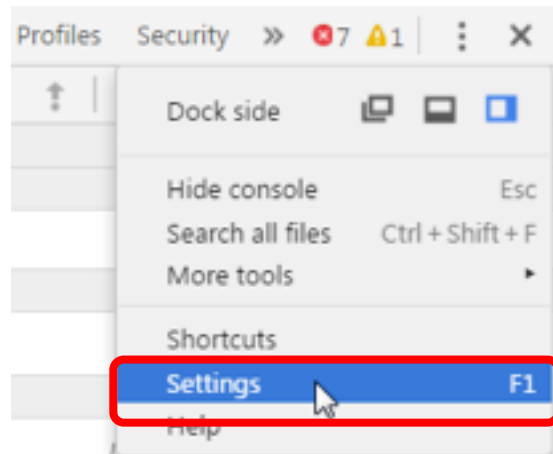
Look into your console!



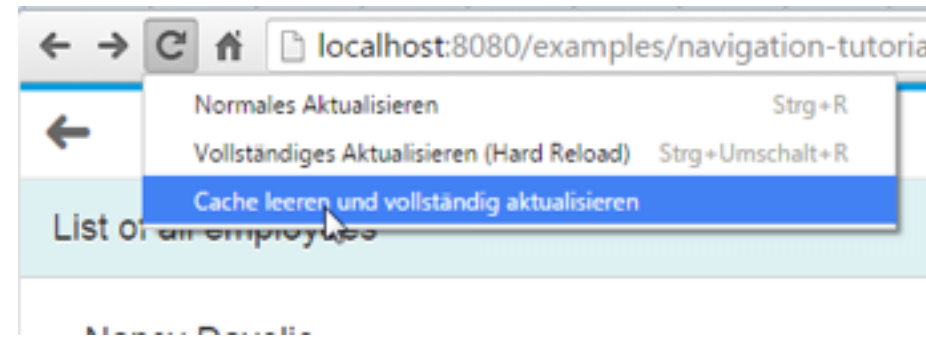
Development tips 3

Caching problems?

Chrome – open DevTools, disable cache



Chrome – open DevTools,
long click on refresh, select clear cache



Development tips 4

Namespaces (SAPUI5)

index.html

```
<script id="sap-ui-bootstrap"
  src="https://sapui5.hana.ondemand.com/resources/sap-ui-core.js"
  type="text/javascript"
  data-sap-ui-libs="sap.m"
  data-sap-ui-theme="sap_bluecrystal"
  data-sap-ui-bindingSyntax="complex"
  data-sap-ui-compatVersion="edge"
  data-sap-ui-preload="async"
  data-sap-ui-resourceroots='{ "todoapp": "../webapp" }' >
</script>
```

```
<script>
  sap.ui.getCore().attachInit(function() {
    new sap.m.Shell({
      app: new sap.ui.core.ComponentContainer({
        height: "100%",
        name: "todoapp"
      })
    }).placeAt("content");
  });
</script>
```

webapp/Component.js

```
sap.ui.define([
  "sap/ui/core/UIComponent",
  "sap/ui/Device",
  "todoapp/model/models",
  "sap/ui/model/json/JSONModel"
], function(UIComponent, Device, models, JSONModel) {
  "use strict";

  return UIComponent.extend("todoapp.Component", {
```

webapp/view/App.view.xml

```
<core:View controllerName="todoapp.controller.App"
  xmlns:core="sap.ui.core" xmlns:html="http://www.w3.org/1999/xhtml"
  xmlns:mvc="sap.ui.core.mvc" xmlns="sap.m">
```

webapp/controller/App.controller.js

```
sap.ui.define([
  "sap/ui/core/mvc/Controller"
], function(Controller) {
  "use strict";

  return Controller.extend("todoapp.controller.App", {
```


Development tips 4

Start with a local JSON Model!

If there is time – or your backend peers work in parallel, you could implement a real backend (REST/OData)

Don't waste too much time on a login screen!

Focus on a **functional app prototype** – something you can pitch and show

Charts are hard

If you use charts, start with the **ChartContainer** example, NOT VizFrame

Use XML views

All of the Explored examples are XML views, easier copy-paste-adapt

Explored app tip

Be aware of the XML namespaces!

Don't forget to declare the layout when you use a layout control!

The screenshot shows the SAPUI5 Explorer interface. On the left, the 'Main.view.xml' file is open, displaying XML code. The code defines a 'sap.ui.core.mvc:View' with 'controllerName="view.Main"' and 'xmlns="sap.m"'. It also declares 'xmlns:sap.ui.layout="sap.ui.layout"' and 'xmlns:sap.ui.core.mvc="sap.ui.core.mvc"'. Inside a '<Page class="marginBoxContent" title="Page">' block, there is a '<content>' block. The code shows two attempts to use 'HorizontalLayout': one with 'xmlns:sap.ui.layout' (commented out as failing) and another with 'xmlns:sap.ui.layout' (commented out as being ok). The correct usage is shown as '<sap.ui.layout:HorizontalLayout>' followed by '<Text text="Hello World!" maxLine...' and '</sap.ui.layout:HorizontalLayout>'. On the right, the 'V.view.xml' file is open, showing the 'sap.ui.layout.sample.HorizontalLayout' controller. The code defines a 'sap.ui.layout.sample.HorizontalLayout' with 'controllerName="sap.ui.layout.sample.HorizontalLayout"', 'xmlns:l="sap.ui.layout"', 'xmlns:mvc="sap.ui.core.mvc"', and 'xmlns="sap.m"'. It then uses '<l:HorizontalLayout>' followed by '<Image'.

```
1 <sap.ui.core.mvc:View controllerName="view.Main"
2   xmlns="sap.m"
3   xmlns:sap.ui.layout="sap.ui.layout"
4   xmlns:sap.ui.core.mvc="sap.ui.core.mvc" >
5   <Page class="marginBoxContent" title="Page">
6     <content>
7       <!-- this would fail because
8        of the namespace! -->
9       <l:HorizontalLayout>
10        <Text text="Hello World!" maxLine...
11      </l:HorizontalLayout>
12      <!-- this would be ok -->
13      <sap.ui.layout:HorizontalLayout>
14        <Text text="Hello World!" max...
15      </sap.ui.layout:HorizontalLayout>
16
17
```

SAPUI5 Explorer

Code: Horizontal Layout

V.view.xml C.controller.js

```
<mvc:View
  controllerName="sap.ui.layout.sample.HorizontalLayout
  xmlns:l="sap.ui.layout"
  xmlns:mvc="sap.ui.core.mvc"
  xmlns="sap.m">
  <l:HorizontalLayout>
    <Image
```

“ Useful links 1/2

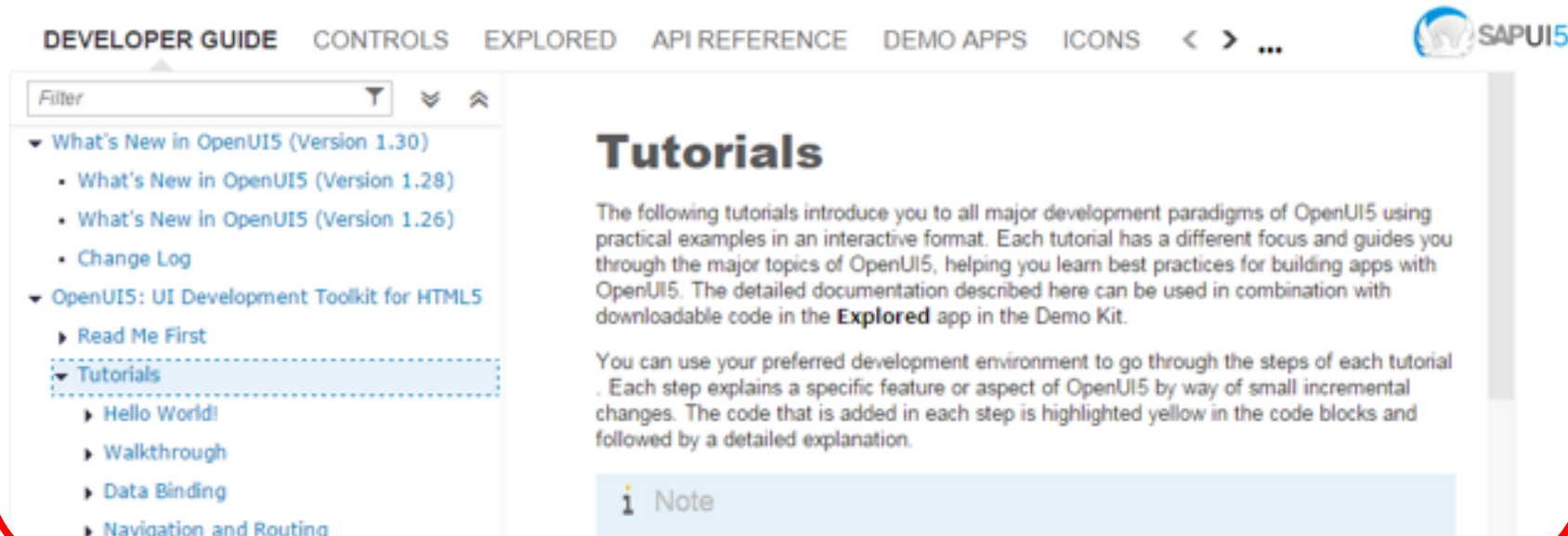
HANA Cloud Platform - Free Developer Version

<https://account.hanatrial.ondemand.com>

OpenUI5 Documentation

<https://openui5.hana.ondemand.com/>

OpenUI5 Tutorials in the Documentation:



Useful links 2/2

OpenUI5 Overview

<http://openui5.org/>

OpenUI5 Explored – Controls overview

<https://openui5.hana.ondemand.com/explored.html>

Free MOOC by SAP: Build your Own SAP Fiori App in the Cloud

<https://open.sap.com/courses/fiux1>

SAPUI5 Space in SAP Community Network (SCN)

<http://scn.sap.com/community/developer-center/front-end>

Fiori Guidelines

<http://experience.sap.com/fiori-guidelines>



SAP WEB IDE Guides: <http://scn.sap.com/docs/DOC-55465>

Examples repositories



Denise Nepraunig: fuzzy-octo-batman

Small self contained examples: MVC, JSON, CRUD, OData

<https://github.com/denisenepraunig/fuzzy-octo-batman>

SAP: OpenUI5 todo app (JSON) with localStorage

<https://github.com/SAP/openui5-sample-app>

Denise Nepraunig: todo app with OData and HANA XS backend

<https://github.com/denisenepraunig/sapui5-hanaxs-examples/tree/master/todoapp>

Denise Nepraunig: chuck_norris_jokes

OpenUI5 app with backend via HCP destination service

https://github.com/denisenepraunig/sapui5-hcp-examples/tree/master/chuck_norris_jokes



Thank you

Denise Nepraunig

SAPUI5 Developer

denise.nepraunig@sap.com

[@denisenepraunig](https://twitter.com/denisenepraunig)

