



Menschen Methoden Lösungen

CREATING APPS WITH UI5

Me

Tobias Hofmann
Senior Consultant

PROFILE

- Fiori
- Cloud
- Portal
- Mobile
- Architect & Developer

CONTACT

- Twitter: @tobiashofmann
- E-Mail: tobias.hofmann@bridging-it.de

■ Seit 2008

Gründung der BridgingIT GmbH Anfang 2008 als IT-Beratungsunternehmen

■ Beratungspartner

Neutral, ohne Produkt- oder Vertriebsbindung

■ Über 520 Mitarbeiter

Kundennähe durch ausgeprägten regionalen Fokus. Standorte: Mannheim, Karlsruhe, Frankfurt, Stuttgart, Köln, München, Zug (CH), Nürnberg und Berlin

■ Netzwerk

Gut ausgebautes Netzwerk aus Applikations- und Technologiepartnern

■ Erfolg

Unabhängigkeit und Stabilität durch Eigenfinanzierung und durchgängigen wirtschaftlichen Erfolg

■ Preis-/Leistung

Attraktives Preis-/Leistungsverhältnis durch geringen Overhead, komfortable Mitarbeiter-Self-Services und konsequentes Outsourcing bei Support Prozessen

■ CraftingIT

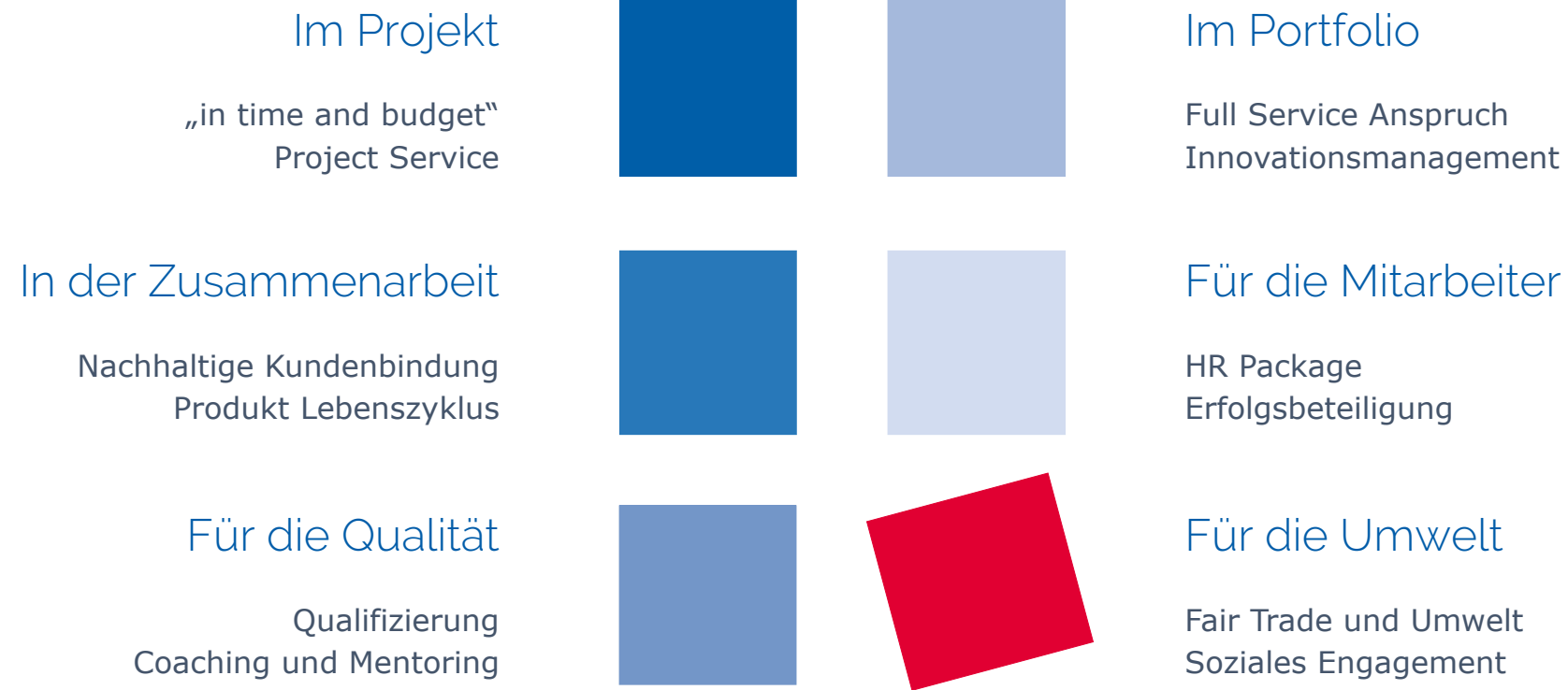
Gründung der Tochter CraftingIT GmbH Anfang 2014 als IT-Service Unternehmen in Magdeburg

■ Qualität

Fokus auf Qualität und Nachhaltigkeit durch hohe Beteiligung der Mitarbeiter an der Unternehmensentwicklung

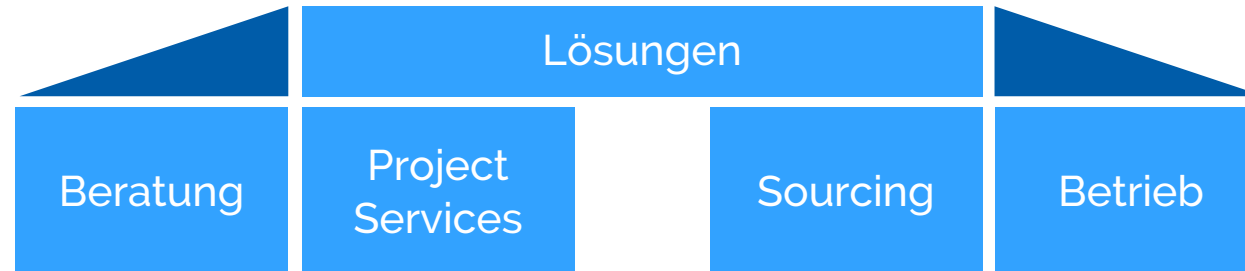
Wir übernehmen Verantwortung

aufmerksam – positiv – zeitgemäß



Portfolio

Unser Full-Service-Ansatz



Beratung

Beratung von der Strategie über die Gestaltung von Geschäftsprozessen bis hin zur Bewertung und Einführung von neuen IT-Systemen und -Anwendungen

Project Services

Unterstützung und Projektsupport bei der Gestaltung von Projekten über den gesamten Lebenszyklus durch Anforderungs-, Projekt- und Testmanagement

Lösungen

Umsetzung von fachlichen und technischen Anforderungen unserer Kunden in bedarfsgerechte, kosteneffiziente und langfristig stabile Lösungen auf der Basis von Standardtechnologien- und Produkten

Sourcing

Unterstützung von der Bereitstellung und Ergänzung von Projektteams über die Beratung in Near-/Offshore-Vorhaben bis hin zum umfassenden Lieferantenmanagement durch CraftingIT GmbH

Betrieb

Dienstleistungen rund um den Betrieb von IT-Systemen und Anwendungen wie Betriebsprozesse, operative Betriebsführung, Beratung zu Fragen der Betriebssicherheit und Wartung

Kunden, Mitgliedschaften, Partner

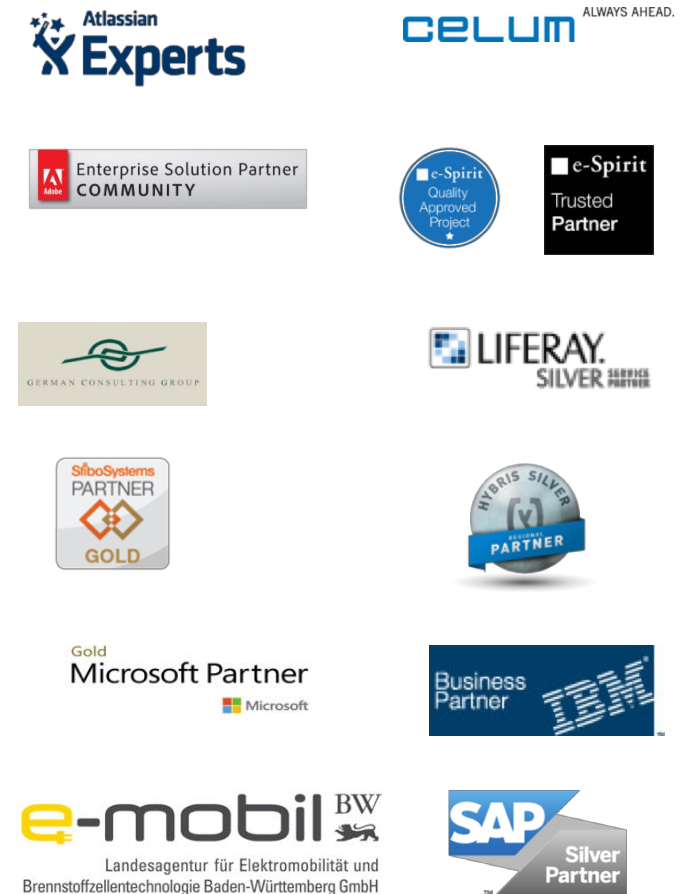
Kunden *



Mitgliedschaften *



Partner *



(* ein Auszug)

Agenda

01

Basics

02

Creating apps

03

Backend

04

Examples

05

Testing

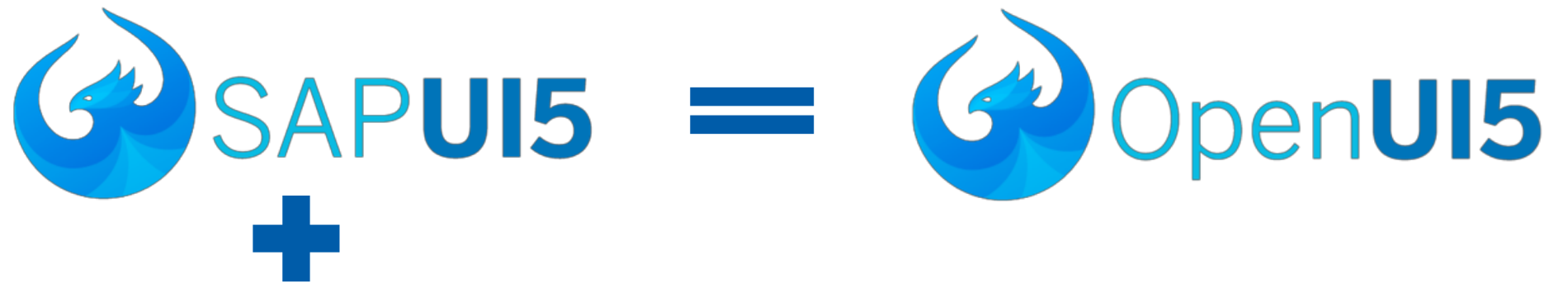
BASICS

How to code

Tabs vs. Spaces

SPACES!!!

UI5 ... Where the cool shit is



Analysis Path Framework
Fiori Elements
Charts / Data Visualization
Process Flow
Geo / Analytical Map
3D

SAPUI5
vs.
OpenUI5



SAPUI5: Where the cooler shit is

Integrated Development Environment (IDE):

If you really need one

- Official answer: use SAP Web IDE
- Fiori customization: use SAP Web IDE

Everyone else:

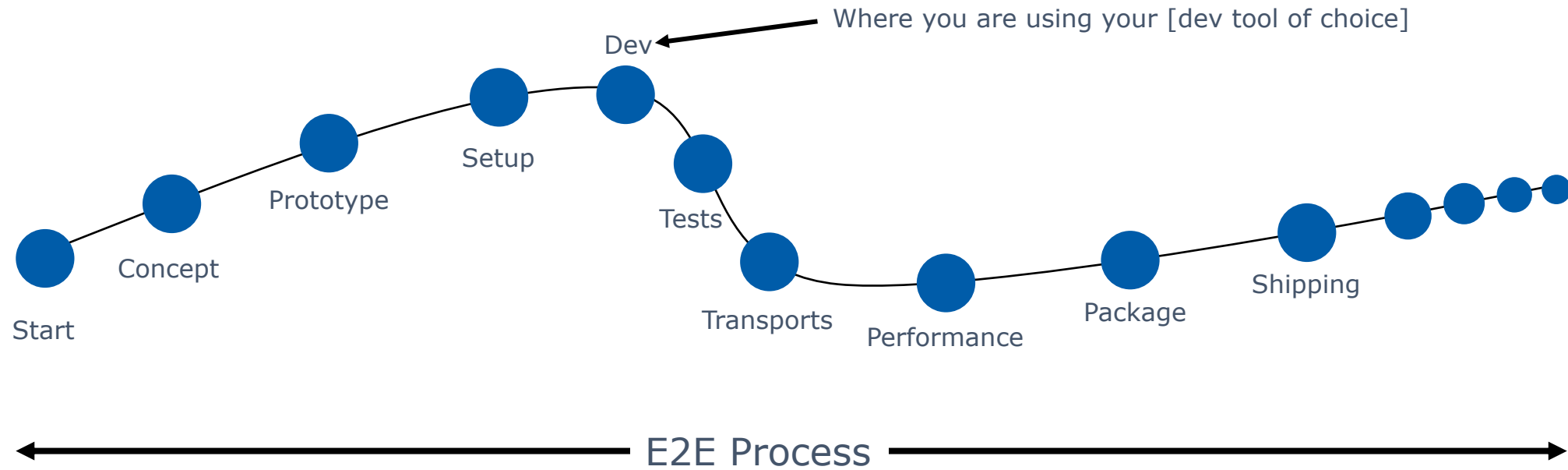
- **Use whatever helps you**
- You are going to write ASCII. Any editor can do this (echo „hello world“ >> file.js)
- Building the app is NOT the job for an IDE
- We are not in the 90s anymore ...

Dev tools

Learn different tools accross complete app lifecycle

Understanding concepts is more important than knowing a feature

In case of doubt: Google



Dev tools

Editor	Eclipse	Give VIM a chance Exit: ESC + : + q + ! + enter	Anything you like
Run local	Static web site? serve	UI5 app with backend data? Wiremock	Simple JSON backend? json-server
	UI5 NPM tools	HTTP server	
Dev tools	Git	Continuous Integration	SonarQube
	GitLab	Nexus	Grunt
	Docker	Apache jMeter	npm
	Jenkins	ATC	Whatever fits your case
Tests	Karma	Istanbul	Webdriver.IO
	Qunit	OPA5	Selenium
	<i>PhantomJS</i>	jUnit	Appium
	Headless Chrome	TestNG	Many more

CREATING APPS

Creating apps

Your app > your child

Run and use the app as often as possible: make it part of your daily routine

Use valid backend data

App with easy & intuitive usage that works → holy grail

Minimum viable product

Just enough features

It works and gets the job done

Early adopters / users



<http://www.bawarchi.com/recipe/lychee-sponge-cake-oetnMcjgaedij.html>



Minimum lovable product

Focus on the features that matter

Purpose driven

Small group that *loves* your product



<https://www.momlovesbaking.com/kit-kat-cake/>

**PICK
TWO**



<https://logosbynick.com/fast-good-cheap-pick-two/>

Creating apps

SAP Fiori Guidelines

Very good starting point

Covers complex scenarios: Draft, Validation, Process

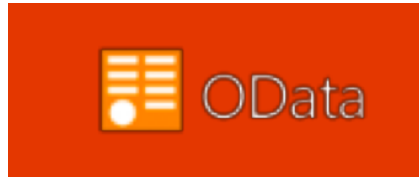
May be too complex: made for ERP apps

Don't forget

Use your brain



BACKEND



VS.

Everything else



Simplified

SAP World: OData

The rest of the world uses boring simple REST with JSON

Why? Because of great tools

The image shows the Swagger Editor interface. On the left, the Swagger JSON schema is displayed in a code editor. On the right, the Swagger Petstore API documentation is shown, including a list of endpoints and their methods.

```
1 swagger: "2.0"
2 info:
3   description: "This is a sample server Petstore server. You can find out more about
4     Swagger at [http://swagger.io](http://swagger.io) or on [irc.freenode.net, #swagger]
5     (http://swagger.io/irc/). For this sample, you can use the api key 'special-key'
6     to test the authorization filters."
7   version: "1.0.0"
8   title: "Swagger Petstore"
9   termsOfService: "http://swagger.io/terms/"
10  contact:
11    email: "apiteam@swagger.io"
12  license:
13    name: "Apache 2.0"
14    url: "http://www.apache.org/licenses/LICENSE-2.0.html"
15  host: "petstore.swagger.io"
16  basePath: "/"
17  tags:
18    - name: "pet"
19      description: "Everything about your Pets"
20    - name: "store"
21      description: "Access to Petstore orders"
22    - name: "user"
23      description: "Operations about user"
24  externalDocs:
25    description: "Find out more about our store"
26    url: "http://swagger.io"
27  schemes:
28    - "http"
29  paths:
30    /pet:
31      post:
32        summary: "Add a new pet to the store"
33        description: ""
34        operationId: "addPet"
35        requestBody:
36          description: "Pet object to add to the store"
37          content:
38            application/json:
39              schema:
40                $ref: "#/components/schemas/Pet"
41          required: true
42        responses:
43          201:
44            description: "Pet successfully added"
45            content:
46              application/json:
47                schema:
48                  $ref: "#/components/schemas/Pet"
49          405:
50            description: "Invalid input"
51      put:
52        summary: "Update an existing pet"
53        description: ""
54        operationId: "updatePet"
55        requestBody:
56          description: "Pet object to update in the store"
57          content:
58            application/json:
59              schema:
60                $ref: "#/components/schemas/Pet"
61          required: true
62        responses:
63          200:
64            description: "Pet successfully updated"
65            content:
66              application/json:
67                schema:
68                  $ref: "#/components/schemas/Pet"
69          400:
70            description: "Invalid input"
71          404:
72            description: "Pet not found"
73          405:
74            description: "Invalid input"
75      get:
76        summary: "Find Pets by status"
77        description: ""
78        operationId: "findPetsByStatus"
79        parameters:
80          - name: "status"
81            in: "query"
82            description: "Status values that need to be considered to filter Pets"
83            type: "string"
84            required: true
85        responses:
86          200:
87            description: "list of pets"
88            content:
89              application/json:
90                schema:
91                  type: "array"
92                  items:
93                    $ref: "#/components/schemas/Pet"
94          400:
95            description: "Invalid input"
96          404:
97            description: "No pets found"
98      delete:
99        summary: "Delete a pet"
100       description: ""
101       operationId: "deletePet"
102       parameters:
103         - name: "petId"
104           in: "path"
105           description: "ID of pet to delete"
106           type: "string"
107           required: true
108       responses:
109         204:
110           description: "Pet successfully deleted"
111         404:
112           description: "Pet not found"
113     /pet/{petId}:
114       get:
115         summary: "Find pet by ID"
116         description: ""
117         operationId: "findPetById"
118         parameters:
119           - name: "petId"
120             in: "path"
121             description: "ID of pet to return"
122             type: "string"
123             required: true
124         responses:
125           200:
126             description: "Pet with ID"
127             content:
128               application/json:
129                 schema:
130                   $ref: "#/components/schemas/Pet"
131           400:
132             description: "Invalid input"
133           404:
134             description: "Pet not found"
135       post:
136         summary: "Update a pet in the store with form data"
137         description: ""
138         operationId: "updatePetWithForm"
139         parameters:
140           - name: "petId"
141             in: "path"
142             description: "ID of pet to update"
143             type: "string"
144             required: true
145           - name: "name"
146             in: "query"
147             description: "Name of pet to update"
148             type: "string"
149             required: false
150           - name: "status"
151             in: "query"
152             description: "Status of pet to update"
153             type: "string"
154             required: false
155         requestBody:
156           description: "Form data to update the pet"
157           content:
158             application/x-www-form-urlencoded:
159               schema:
160                 type: "object"
161                 properties:
162                   name:
163                     type: "string"
164                   status:
165                     type: "string"
166               required: false
167         responses:
168           200:
169             description: "Pet successfully updated"
170             content:
171               application/json:
172                 schema:
173                   $ref: "#/components/schemas/Pet"
174           400:
175             description: "Invalid input"
176           404:
177             description: "Pet not found"
178     /pet/{petId}/tags:
179       post:
180         summary: "Add tags to a pet"
181         description: ""
182         operationId: "addTagsToPet"
183         parameters:
184           - name: "petId"
185             in: "path"
186             description: "ID of pet to add tags"
187             type: "string"
188             required: true
189           - name: "tags"
190             in: "query"
191             description: "Tags to be added"
192             type: "array"
193             required: true
194             items:
195               type: "string"
196         requestBody:
197           description: "List of tags to add"
198           content:
199             application/json:
200               schema:
201                 type: "array"
202                 items:
203                   type: "string"
204           required: true
205         responses:
206           204:
207             description: "Tags successfully added"
208           400:
209             description: "Invalid input"
210           404:
211             description: "Pet not found"
212     /pet/{petId}/uploadImage:
213       post:
214         summary: "Upload image to a pet"
215         description: ""
216         operationId: "uploadImage"
217         parameters:
218           - name: "petId"
219             in: "path"
220             description: "ID of pet to upload image"
221             type: "string"
222             required: true
223           - name: "image"
224             in: "query"
225             description: "Image file to upload"
226             type: "string"
227             required: true
228         requestBody:
229           description: "Image file to upload"
230           content:
231             multipart/form-data:
232               schema:
233                 type: "object"
234                 properties:
235                   image:
236                     type: "string"
237                   additionalMetadata:
238                     type: "string"
239                 required: false
240         responses:
241           200:
242             description: "Image successfully uploaded"
243             content:
244               application/json:
245                 schema:
246                   type: "object"
247                   properties:
248                     code:
249                       type: "string"
250                     message:
251                       type: "string"
252                   required: false
253           400:
254             description: "Invalid input"
255           404:
256             description: "Pet not found"
257           500:
258             description: "Image could not be uploaded"
259             content:
260               application/json:
261                 schema:
262                   type: "object"
263                   properties:
264                     code:
265                       type: "string"
266                     message:
267                       type: "string"
268                   required: false
```

Swagger Petstore

This is a sample server Petstore server. You can find out more about Swagger at <http://swagger.io> or on irc.freenode.net, #swagger (http://swagger.io/irc/).

pet Everything about your Pets

- POST** /pet. Add a new pet to the store
- PUT** /pet. Update an existing pet
- GET** /pet/findByStatus. Finds Pets by status
- GET** /pet/{petId}. Find pet by ID
- POST** /pet/{petId}. Updates a pet in the store with form data

Generate Server **Generate Client**

ada-server	jaxrs-resteasy	restbed
aspsnetcore	jaxrs-resteasy-eap	rust-server
erlang-server	jaxrs-spec	scala-lagom-server
finch	kotlin-server	scalatra
go-server	lumen	sinatra
haskell	msf4j	slim
inflector	nancyfx	spring
java-pkmst	nodejs-server	undertow
java-play-framework	php-silex	ze-ph
java-vertx	php-symfony	
jaxrs	pistache-server	
jaxrs-cxf	python-flask	
jaxrs-cxf-cdi	rails5	

And UI5?

Creating great UX and apps without using Odata?



<https://imgur.com/gallery/64M1g>

Example

Azure Cognitive Service: sentiment analysis

<https://westus.api.cognitive.microsoft.com/text/analytics/v2.0/sentiment>

POST

```
{ "documents": [ { "language": "en", "id": "1", "text": "Hello world. This is some input text that I love." }, { "language": "fr", "id": "2", "text": "Bonjour tout le monde" }, { "language": "es", "id": "3", "text": "La carretera estaba atascada. Había mucho tráfico el día de ayer." } ] }
```

Response

```
{ "documents": [ { "score": 0.92, "id": "1" }, { "score": 0.85, "id": "2" }, { "score": 0.34, "id": "3" } ], "errors": null }
```


Not everything in the SAP world is Odata

Example

SAP Translation Hub

GET

<https://saptranslation-s00000000001trial.hanatrial.ondemand.com/translationhub/api/v1/domains>

Response

```
{"domains": [{"id": "AL", "name": "ADL"}, {"id": "AI", "name": "AII"}, {"id": "AC", "name": "Accounting - General"}, {"id": "AM", "name": "Additive Manufacturing"}]}
```

Examples,
anyone?



<http://www.everydayunittesting.com/blog/page/2>

EXAMPLES

Private cloud



Raspberry Pi 3

- Web
- Apps
- VPN

Intel NUC i7 32GB RAM

- Docker
- VMs

Storage

- Main
- Backup
- Snapshots

Wake on Lan

See status of server

Start server + container

Access services

Wake on Lan

Search for server or IP

Servers (7)

Server	192.168.0.164	Running
Intel NUC i7	192.168.0.2	Down
Lenovo	192.168.0.6	Down
Kodi	192.168.0.4	Running

Server Details

192.168.0.2

Down

Proxmox	Open service
Portainer	Start Open service
Gitlab	Start Open service
Continuous Integration	Start Open service
Code quality analysis	Start Open service

Learning assets

Find learning material

Access learning

The screenshot displays the SAP Learning Hub interface. At the top, there is a search bar with the placeholder text "Search for title or description". Below the search bar, a navigation bar shows "936 Learnings" and six filter icons: "All trainings", "E-Books", "Handbooks", "Workshops", "Learning map", and "PDF". The main content area features a grid of six learning asset cards. Each card includes a title, a course ID, a brief description, and a version number. The cards are:

- SAP Activate Methodology** (ACT100): This course is designed to give participants an introduction to the new innovation adoption... Ver. EN_Col05_v3. eBook.
- SAP HANA Introduction** (HA100): Gain Knowledge about SAP HANA SPS10 and in-memory computing. Ver. EN_Col13. eBook.
- SAP HANA Cloud Platform (HCP)** (HCP100): This course will introduce and inform you about the vast potential of SAP HANA Cloud Platform with... Ver. EN_Col02_v2. eBook.
- AS ABAP - Advanced Administration** (ADM103): Gain deeper knowledge with the tasks in ABAP Administration to provide accurate support on... Ver. EN_Col15_v2. eBook.
- AS Java - Administration** (ADM800): Provides the various tasks of the system administration for systems based on SAP NetWeav... Ver. EN_Col16_v2. eBook.
- SAP Identity Management** (ADM900): Gain knowledge about SAP Identity Management 8.0 SP3. Understand how SAP Identity manages users ... Ver. EN_Col15_v2. eBook.

SAP Notes Search

Search SAP Notes

Search for notes with Star or Flag set. Star ☐ NO Flag ☐ NO

Component Release Status Priority Category Rating Projects Years

Took: Results

SAP Notes

No SAP Notes found. Please search.

Search SAP Notes

Search for notes with Star or Flag set. Star ☐ NO Flag ☐ NO

Component Release Status Priority Category Rating Projects Years (2017)

Took: 50 Results 5613

SAP Notes

2506466: Release of Fiori Product

You have installed SAP Fiori Product Versions for the Business Suite based on either • • • • SAP UI Addon 1.0 (Frontend Server 4.0) SAP Fiori Frontendserver 2.0 (SAP_UI Addon 2.0 or SAP_UI 7.50) SAP Fiori Frontendserver 3.0 (SAP_UI Addon 3.0 or SAP_UI 7.50) SAP Fiori Frontendserver 4.0.

XX-SER-REL 14-12-2017 Version: 1 <https://launchpad.support.sap.com/#/notes/2506466>

2570306: Push Notifications not received when Fiori Client in

Push Notifications not received when Fiori Client in Foreground on iOS and Android. Environment: Fiori Client in Foreground. Cause: This is by design, Fiori Client will not provide notifications if it is in foreground, user could use custom d

Release of Fiori Product

Priority: XX-SER-REL Priority: Recommendations / Additional Info

Category: Release planning information Release Status: Released for Customer

Date: 14-12-2017 Version: 1

SYMPTOM

NOTE CONTENT

You have installed SAP Fiori Product Versions for the Business Suite based on either • • • • SAP UI Addon 1.0 (Frontend Server 4.0) SAP Fiori Frontendserver 2.0 (SAP_UI Addon 2.0 or SAP_UI 7.50) SAP Fiori Frontendserver 3.0 (SAP_UI Addon 3.0 or SAP_UI 7.50) SAP Fiori Frontendserver 4.0.

bridgingIT

BridgingIT GmbH | SITWDF 2019 31

Cognitive Leave Request

Combine Azure Cognitive Services with SAP Cloud Platform

SCP Workflow and SCP API Management

bridgingIT - Menschen Methoden Lösungen

Create Leave Request

Start >> Select date >> Take picture >> Justification >> Summary

Time to convince your manager...

Show some love to your manager and come up with a good excuse. Good luck, you will need it!

This text will be automatically translated by SAP Translation Hub

Target language for manager:

Vacations!

Text translated in realtime to: de

Result of real time translation

Urlaub!

Sentiment score: 55.74%

SAP

Sapphire Now 2018 Workflow Additional Information

Leave Request Cognitive Leave Req... Azure + SCP

SAP API Management Requests overview

Workflow

My Inbox All Tasks 0

Monitor Workflows Workflow Instances

Monitor Workflows Workflow Definitions

My Inbox All Tasks 0

Alle Elemente (1)

Suchen

Approve leave for S0018293263 Mittel

Employee provided information

Mitarbeitername:

Beginndatum Urlaub:

Anzahl Tage:

Rechtfertigung:

Demos

Project Wizard

Simplify App creation process

Project templates with automatic Continuous Integration / Delivery setup

Ensure each new project follows certain standards

The image displays two overlapping web interfaces. The background interface is the 'Project Creation Wizard' from GitLab, showing a sidebar with navigation options like Project, Details, Activity, Cycle Analytics, Repository, Issues, Merge Requests, CI / CD, Operations, Wiki, Snippets, and Settings. The main content area shows project details for 'Test2', including a description, project ID, and repository information. The foreground interface is the Jenkins 'Pipeline Test2' page. It features a sidebar with links like Back to Dashboard, Status, Changes, Jetzt bauen, Pipeline löschen, Konfigurieren, Full Stage View, Open Blue Ocean, Rename, and Pipeline Syntax. The main content area shows the 'Stage View' of the pipeline, with a table of stage times and a 'Build-Verlauf' section.

Project Creation Wizard (Background)

Suchen

UI5 Test Basic Master Detail stable

UI5 Master Detail stable

Project data

Project name: Test1

App Title: Test App

Namespace: de.itsfullofstars.sap.test.flori

Description: Lorem ipsum app description is useful and makes sense but must be not too long.

Open UI5 Version: latest

Karma: ☒ AN ☐ QUnit: ☐

Jenkins Pipeline Test2 (Foreground)

Back to Dashboard

Status

Changes

Jetzt bauen

Pipeline löschen

Konfigurieren

Full Stage View

Open Blue Ocean

Rename

Pipeline Syntax

Build-Verlauf

suchen

1 25.01.2019 12:10

Started by GitLab push by Tobias Hofmann

RSS aller Builds RSS der Fehlschläge

Stage View

Average stage times:

Declarative: Checkout SCM	SCM	Preparation	Build	Test	Analyse
4s	2s	19s	1s	10s	2s
4s	2s	19s	1s	10s	

Jan 25 13:10 No Changes

Permalinks

- Letzter Build (#1) vor 45 Sekunden

Demos

Speech 2 Text

Combine Azure Cognitive Services with SAP Cloud Platform

Add subtitle to video stream in real time

Enhance accessibility: non-native speaking audience, hearing disability, etc.

Get text version of a talk (PDF)

The image shows two overlapping windows. The top window is the 'Speech 2 Text App' by bridging IT. It features a 'Select speaker language' dropdown set to 'English - US', a 'Name of recording' field, and 'Start recording' and 'Stop recording' buttons. Below these are tabs for 'Recording', 'Speech2Text', 'Word Cloud', and 'Sentiments'. The 'Speech2Text' tab is active, showing a 'Subtitle' section with a black box containing the text 'looking for' and a 'Transcribed text of speech' section with a paragraph of text. The bottom window is an OBS 21.0.3 (mac) interface showing a video stream titled 'Intelligent Suite'. The video content features a diagram with 'Intelligent Suite' at the center, surrounded by icons for 'Manufacturing & Supply Chain', 'Digital Core', 'People Engagement', 'Customer Experience', and 'Network & Sped Management'. The SAP logo and 'TECHEDLIVE' are in the bottom right of the video frame, and the text 'automate the day-to-day business processes' is at the bottom. The OBS interface includes a 'Szenen' list, 'Quellen' (sources), a 'Mixer' with volume sliders, and a 'Steuerung' (controls) panel on the right.

Examples

Find the most lovable feature

Does not need to be necessarily a UI5 app

App	Loved feature	Technology
Control deploy of employees	Employee and supervisor can see if deploy is against company rules (2014)	UI5
Approval app	Approve all type of workflows in one app (2010+)	JS, UI5, Hybrid
Portal customization	Responsive (2009+)	JS
Incident Management	Offline (2013)	Hybrid
Inventory Management	Input via bar code scanner (2012+)	ITSmobile
Contract management	Combine several screens into one (2016)	Personas
SSO on mobile device	SSO with user certificate (2014+)	Afaria

TESTING



Test Personas

I guess, no Problem?

This can't be complicated!

Should not take too much time.

Junior

Good idea, do we have test data?

Let's add it to the backlog for the released API.

Senior

With visual regression tests incl. legacy code!

Of course, part of our portfolio!!

In time and in budget!!!

Sales

80% code coverage is a minimum requirement.

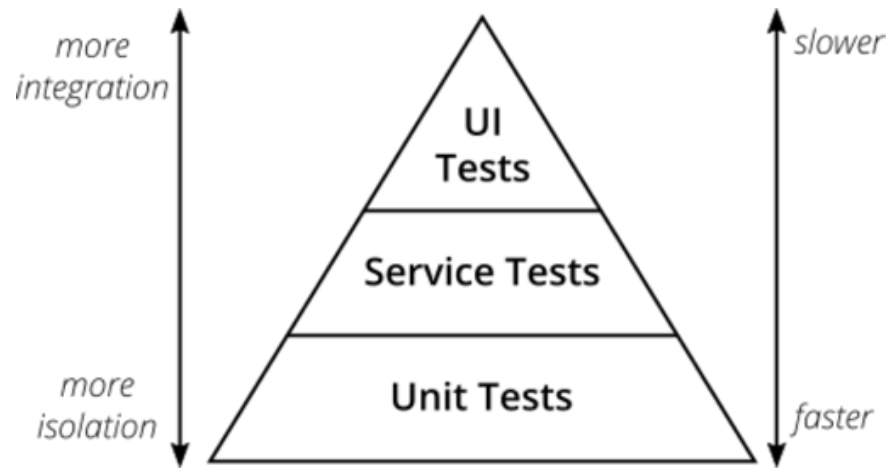
I need nice reports!

Client project manager

No technical details please. Is the app working?

End User

Tests



REALITY

We let the client do the testing for us. In production.

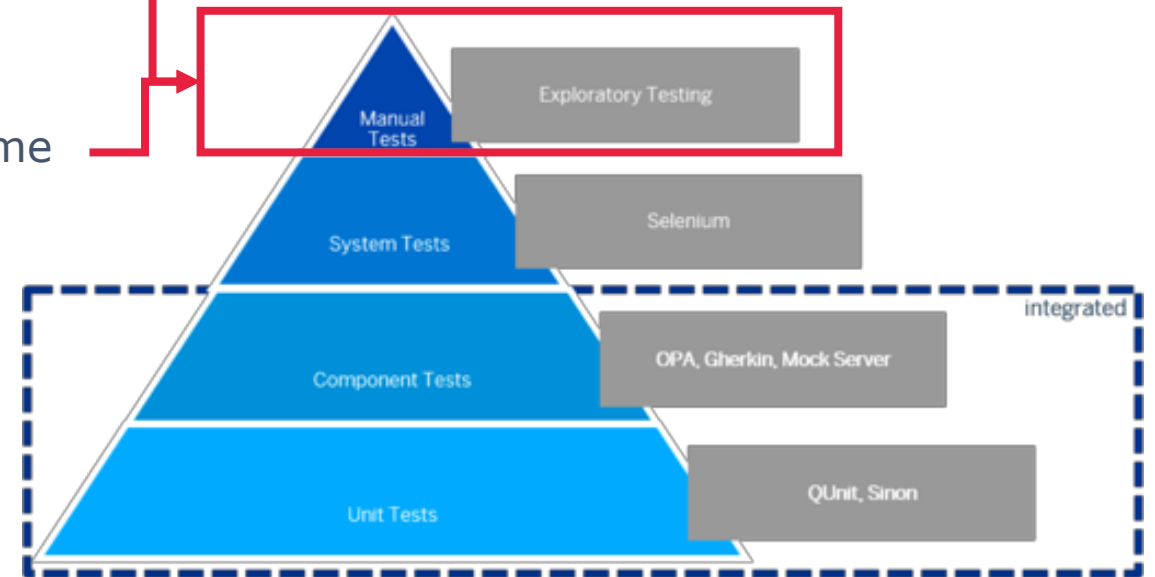
Benefit

faster shipping, tests with real world data

THEORY

Non-SAP world: We test all the time

SAP world: We test all the time



<https://martinfowler.com/articles/practical-test-pyramid.html>

<https://sapui5.hana.ondemand.com/#/topic/7cdee404cac441888539ed7bfe076e57>

Rules

#1	Just do it!!!
#2	You will always meet at least 2 times.
#3	Just like with documentation, you write tests for your future self.
#4	Testing is not free. 100%+ time effort is not unrealistic. Learn how to write testable code.
#5	No excuses: If you do not write tests, it's because of you.
#6	Something is better than nothing.
#7	Unit tests are not enough

UI5 apps support qunit, OPA5, Istanbul, Karma, mocha, chais, selenium, etc.

Tests can be written and run using a wide range of tools

Transparent reporting of test runs and results

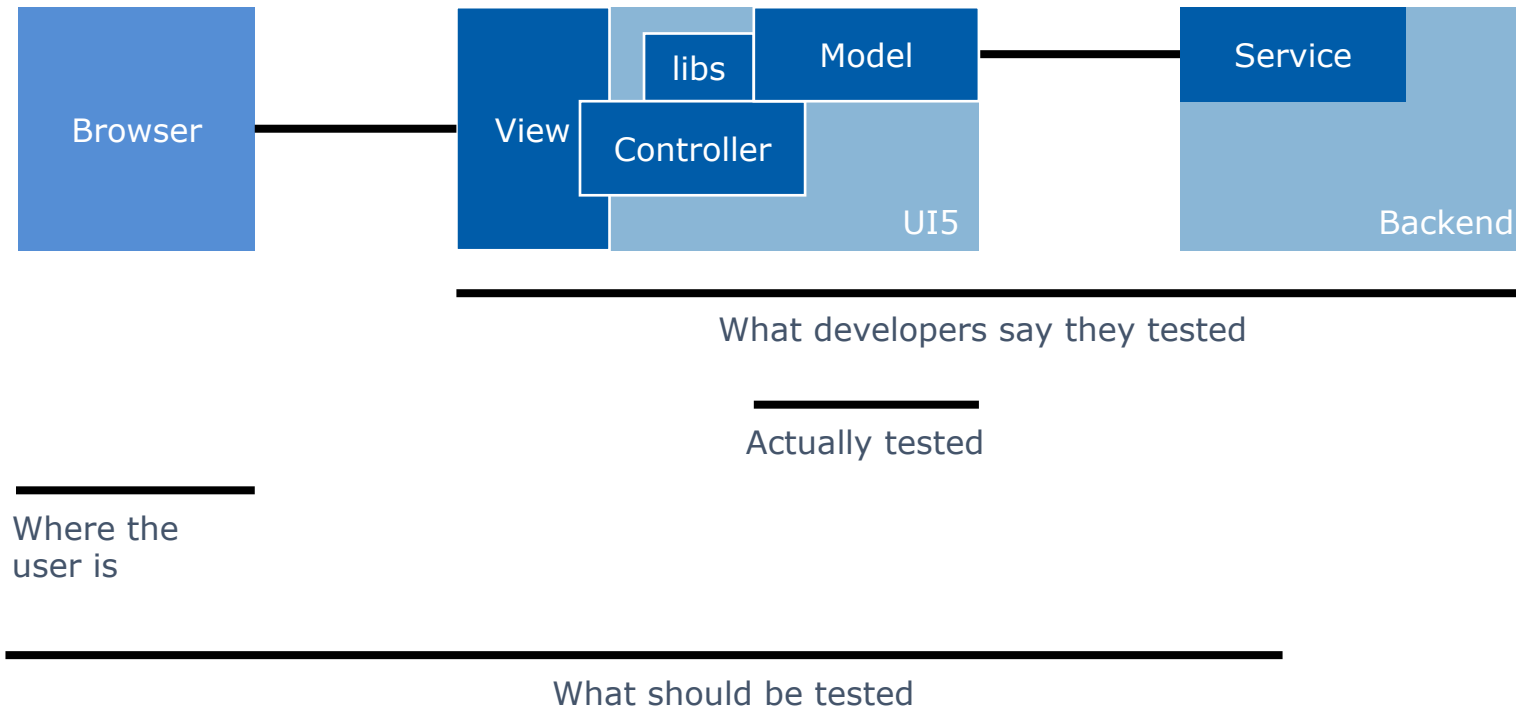
Tests

Choose your battle wisely

Live is too short to spend time on testing the wrong component with the wrong tool

Don't test what's out of your scope

If you can't control it, ignore it



Focus: user journey

Mock oData backend, not UI controls

Given, then when

- Given: I am on the start page
- When: I see a list
- Then: I search for Fiori and get one search result

Isolate user actions

```
opaTest("Search", function (Given, When, Then) {  
  // Arrangements  
  Given.iStartMyApp();  
  //Actions  
  When.onTheWorklistPage.iSearchForTheFirstObject();  
  // Assertions  
  Then.onTheWorklistPage.theTableShowsOnlyObjectsWithTheSearchStringInTheirTitle();  
});
```

Can get complicated

Won't help with visual regression tests

```
Opa5.createPageObjects({
  onTheWorklistPage : {
    iSearchForTheFirstObject: function() {
      var sFirstObjectTitle;
      return this.waitFor({
        id: sTableId,
        viewName: sViewName,
        matchers: new AggregationFilled({
          name: "items"
        }),
        success: function(oTable) {
          sFirstObjectTitle = oTable.getItems()[0].getCells()[0].getTitle();
          this.iSearchForValue(sFirstObjectTitle);
          this.waitFor({
            id: [sTableId, sSearchFieldId],
            viewName: sViewName,
            check : allItemsInTheListContainTheSearchTerm,
            errorMessage: "Did not find any table entries or too many while trying to search for the first object."
          });
        },
        errorMessage: "Did not find table entries while trying to search for the first object."
      });
    },
    iSearchForValue : function (sSearchString) {...}
  },
  function allItemsInTheListContainTheSearchTerm (aControls) {...}
  theTableShowsOnlyObjectsWithTheSearchStringInTheirTitle : function () {...}
});
```

Integration tests

Testing your app forces you to follow certain standards

Stable Ids	Assign an Id to UI elements to make them accessible for UI tests. Avoid automatically assigned {__xmlview0} as this Id can change.
UI patterns	Allows to build tests based on templates. Form validation, wait dialog, lists, etc. Speeds up tests development and assures a coherent UX of your apps.
One root	Make UI elements findable by using a coherent tree of UI elements.
Follow possible user interactions and treat them	Write your tests to cover the user journey. Include alternative paths and possible exceptions. The more alternatives you cover, less your users have to test for you in production.
I18n for texts	Make texts i18n and check the translation for each one.
Discover not handled exceptions	Use code coverage to find not covered use cases and identify why these are not covered.
Many more	This list will never be finished.

Stable Ids

Important: always use stable Ids!

Manifest.json

```
"rootView": {
  "viewName": "sitmuc.SITMuc.view.Home",
  "type": "XML",
  "id": "sitmuc"
}
"targets": {
  "TargetHome": {
    "viewType": "XML",
    "viewName": "Home",
    "viewId": "home"
  }
}
```

Index.html

```
new sap.m.Shell({
  app: new sap.ui.core.ComponentContainer({
    id : "sitmuc2018App",
    name : "de.itsfullofstars.sitmuc.SITMuc"}}))
```

Test

```
browser.element ('#sitmuc2018---sitmuc--TileContainerLayout');
```

View.xml

```
<mvc:View
xmlns:html="http://www.w3.org/1999/xhtml"
controllerName="sitmuc.SITMuc.controller.Home">
<App id="idAppControl">
  <pages>
    <Page id="homePage" title="{i18n>title}">
      <content>
        <l:HorizontalLayout
          id="TileContainerLayout"
          content="{/links}">
```

Component.js

```
constructor: function(sId, mSettings) {
  UIComponent.call(this, "sitmuc2018", mSettings);}
```



Tests with wdio

OPA5 is fine

Consider an alternative

Selenium is widely adopted. There is a reason to it.

Try to use it

Try webdriver.io

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Lines
controller/ Overview.controller.js	0	0	0	0	... 80,83,86,89
model/ models.js	55.56	100	66.67	55.56	17,18,19,20
All files	9.09	0	25	9.09	



You can too!

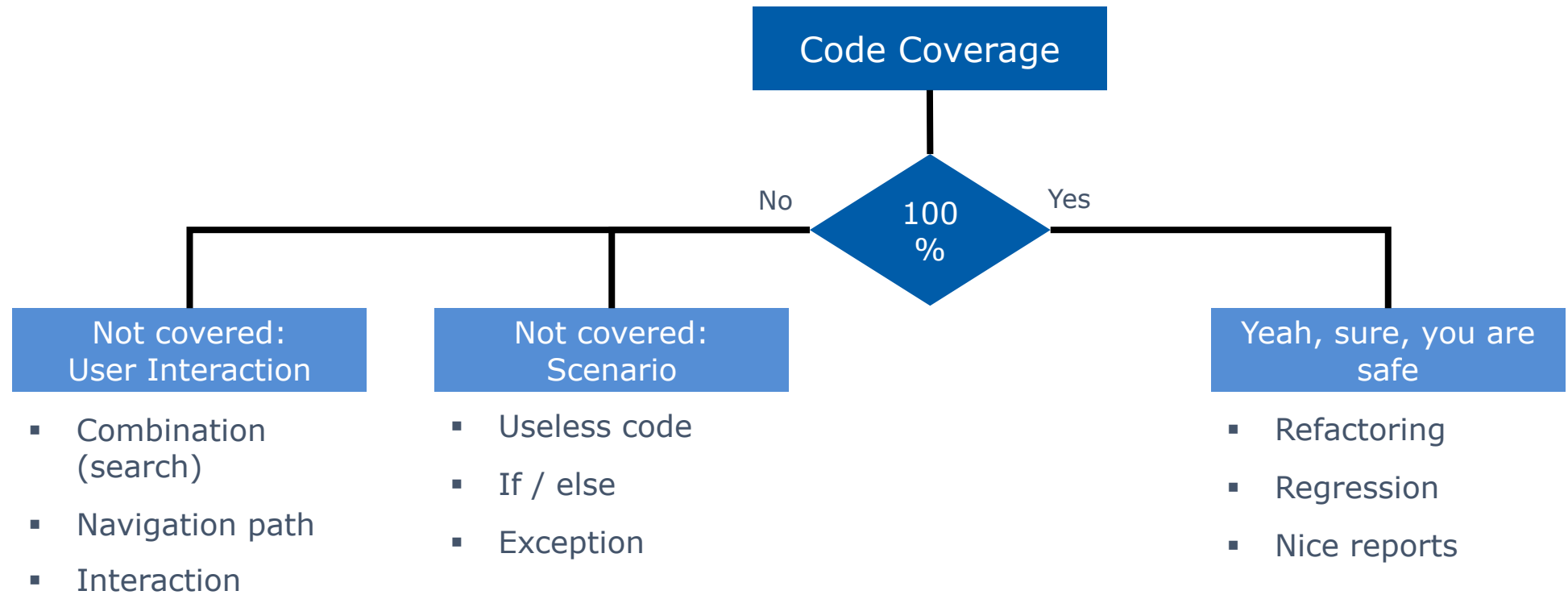
File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Lines
webapp/ Component.js	100	100	100	100	
webapp/controller/ Overview.controller.js	100	100	100	100	
webapp/model/ models.js	100	100	100	100	
All files	100	100	100	100	

Combination of integration tests with code coverage

Free extra: go to 100% code coverage

Cover possible paths

Find not covered user interactions







Wir freuen uns auf Sie!

Mannheim

N7, 5-6
68161 Mannheim

Karlsruhe

Rüppurrer Str. 4
76137 Karlsruhe

Frankfurt

Solmsstraße 4
60486 Frankfurt

Stuttgart

Marienstraße 17
70178 Stuttgart

Köln

Martinstraße 3
50667 Köln

München

Riesstraße 12
80992 München

Zug (CH)

Baarerstraße 14
CH-6300 Zug

Nürnberg

Königstorgraben 11
90402 Nürnberg

Berlin

Friedrichstr. 68
10117 Berlin