



SAP Fiori[®]

Implementierung und Entwicklung

- › Customizing, Entwicklung und Erweiterung von Oberflächen für SAP S/4HANA
- › Installation und Konfiguration Schritt für Schritt erklärt
- › Inkl. SAP Fiori 3.0, Continuous Integration und Continuous Delivery u.v.m.

3., aktualisierte und erweiterte Auflage

Michael Englbrecht

 Rheinwerk
Publishing

Kapitel 8

Anpassungen mit dem UI Theme Designer

Mit dem UI Theme Designer können Sie das visuelle Design von SAP-Fiori-Apps anpassen. In diesem Kapitel zeigen wir Ihnen, wie Sie mit dem UI Theme Designer arbeiten und welche Möglichkeiten er bereitstellt.

8

Der *UI Theme Designer* von SAP ist ein auf HTML basierendes Werkzeug, mit dem Sie das optische Erscheinungsbild von Webanwendungen an Ihre Wünsche anpassen können. Der UI Theme Designer unterstützt neben SAP Fiori eine Vielzahl von UI-Technologien:

- Web Dynpro für ABAP
- Web Dynpro für Java
- SAP GUI for HTML
- SAPUI5 (inklusive SAP Fiori)
- CRM Web UI
- SAP Enterprise Portal
- SAP Business Client
- SAP GUI

Der UI Theme Designer ist damit ein Werkzeug, das für verschiedene Plattformen und Technologien genutzt werden kann. Sie müssen daher die visuellen Themes, die Sie für das visuelle Erscheinungsbild Ihres Unternehmens erstellen, nur einmal anlegen und können es anschließend für alle Anwendungen nutzen, die auf den genannten Techniken basieren. In diesem Kapitel zeigen wir Ihnen die Verwendung des UI Theme Designers am Beispiel einer SAP-Fiori-App.

Um den UI Theme Designer verwenden zu können, müssen Sie SAP-Hinweis 1852401 implementieren. Danach müssen Sie die folgenden ICF-Knoten aktivieren:

- `/sap/public/bc/themes`
- `/sap/bc/theming`

Administration

Nutzer, die Schreibzugriff zum Erstellen, Ändern und Löschen von Themes haben sollen, benötigen in ihrem Benutzerprofil eine Berechtigung zum Berechtigungsobjekt `/UI5/THEME`. Dieses Berechtigungsobjekt hat die beiden Felder `ACTVT` und `/UI5/THMID`. Das Feld `ACTVT` muss den Wert »02« (Change) haben, und das Feld `/UI5/THMID` muss mit den Werten der änderbaren Theme-IDs belegt sein.

Themes sind mandantenspezifisch. Dies bedeutet, dass der UI Theme Designer im selben Mandanten (und auf demselben Server) gestartet werden muss, in dem sich auch das *UI Theme Repository* befindet.

Für die Arbeit mit dem UI Theme Designer empfiehlt sich Google Chrome, obwohl auch Mozilla Firefox und der Microsoft Internet Explorer ab Version 10 unterstützt werden.

8.1 Konzept des UI Theme Designers

Basis-Theme

SAPUI5-Theming basiert auf CSS und CSS-Parametern. Jedes Theme, das Sie mit dem UI Theme Designer erstellen, baut auf einem Basis-Theme auf. Ihr Theme verwendet die CSS-Dateien des Basis-Themes plus zusätzliche CSS-Dateien, die Sie mit dem UI Theme Designer erstellen.

Die CSS-Dateien eines Basis-Themes sind folgendermaßen aufgeteilt:

- CSS-Dateien für die einzelnen SAPUI5 Controls
- globale CSS-Dateien, die von allen SAPUI5 Controls verwendet werden (`global.css` und `shared.css`)

Zur Laufzeit führt der UI Theme Generator die Dateien des Basis-Themes und Ihres Themes zusammen. Der UI Theme Generator erledigt die folgenden Aufgaben:

- Er ersetzt die CSS-Parameter durch deren Werte.
- Er führt eine optionale Optimierung und Komprimierung der Dateien durch.
- Er erzeugt je SAPUI5-Bibliothek eine Datei namens `library.css`.

8.1.1 CSS und LESS

LESS SAPUI5 verwendet die Sprache *LESS*, um CSS-Parameter zu behandeln. LESS ist eine dynamische Stylesheet-Sprache und erweitert CSS mit dynamischem Verhalten wie Variablen, Mixins, Berechnungen und Funktionen. Diese Merkmale von LESS können Sie in eigenen Anwendungen und Controls nutzen. Mit LESS können Sie CSS effizienter schreiben und Wiederho-

lungen im CSS-Code vermeiden. LESS wird vom UI Theme Generator zu CSS kompiliert. LESS ist ein Open-Source-Projekt, das Sie auf <http://lesscss.org> finden.

In einigen wenigen Fällen müssen Sie zusätzliche CSS-Dateien verwenden, die Teile des Themes anpassen. Dazu gehen Sie in SAPUI5 wie folgt vor:

- Verwenden Sie ein `<style>`- oder `<link>`-Element im HTML-`<head>`-Element, um zusätzliches CSS einzubinden.
 - Die so eingebundenen Stile werden immer nach den Stilen des UI-Themes angewandt.
 - Nach den Regeln von CSS überschreibt bei einer Mehrfachdefinition der zuletzt eingebundene Stil den Basisstil.
- Verwenden Sie die Methode `addStyleClass`, über die jedes SAPUI5 Control verfügt.

Um herauszufinden, welche IDs für die Elemente verwendet werden, deren Erscheinungsbild Sie anpassen möchten, können Sie die Entwicklungswerzeuge Ihres Webbrowsers verwenden, um damit das HTML-DOM zu untersuchen. In diesem Zusammenhang ist es auch nützlich zu wissen, dass die ID eines SAPUI5 Controls, das diesem bei seiner Erzeugung gegeben wird, von dem obersten HTML-Element dieses Controls übernommen wird. Diese ID können Sie daher als Selektor im CSS verwenden.

8.1.2 Parameter

Oft werden in CSS-Dateien Werte an mehreren Stellen verwendet. Typische Beispiele dafür sind Abstände oder Farben. Zu diesem Zweck können Sie mit LESS Parameter definieren, auf die Sie sich dann an anderen Stellen der CSS-Datei beziehen, wie dies Listing 8.1 zeigt.

```
@sapUITextColor: #000000; /* Textfarbe ist Schwarz */

button {
  color: @sapUITextColor; /* Textfarbe wird referenziert */
}
```

Listing 8.1 LESS-Parameter definieren und verwenden

Sie definieren die Parameter nur einmal und referenzieren sie dann an vielen verschiedenen Stellen. Der UI Theme Generator ersetzt anschließend die Referenzen durch die aktuellen Werte und erzeugt so eine gültige CSS-Datei.

Zusatzzdateien

Parameter referenzieren

Sie können die Parameterwerte auch zur Laufzeit mit JavaScript-Funktionen lesen, wie es in Listing 8.2 dargestellt ist.

```
sap.ui.define([
    "sap/ui/core/mvc/Controller",
    "sap/ui/core/theming/Parameters"
], function (Controller, Parameters) {
    ...
    var textColor = Parameters.get("sapLinkColor");
});
```

Listing 8.2 Zugriff auf die Parameterwerte aus JavaScript

Mit dieser Technik können Sie Parameter aus SAPUI5-Themes auf eigene Controls übertragen. Dies ist sinnvoll, wenn Sie APIs wie etwa WebGL oder Canvas einsetzen, in denen Sie das CSS nicht direkt verwenden können.

8.2 Mit dem UI Theme Designer arbeiten

Wenn Sie mit dem UI Theme Designer arbeiten, durchlaufen Sie die folgenden Schritte:

1. Sie starten den UI Theme Designer über Transaktion /UI5/THEME_DESIGNER. Alternativ navigieren Sie mit einem Browser zu <http://<host>:<port>/sap/bc/theming/theme-designer>.
2. Sie wählen ein Basis-Theme aus.
3. Sie geben die URL einer SAPUI5-Anwendung – in diesem Falle die URL des SAP Fiori Launchpads – für die Vorschau an.
4. Sie definieren Ihr Theme, indem Sie Parameterwerte anpassen, Bilder einfügen oder eigenen LESS- bzw. CSS-Code schreiben.
5. Sie speichern Ihr Theme unter einem neuen Namen.
6. Sie publizieren das Theme.

Sobald Sie das Theme publiziert haben, können Sie es in SAP Fiori nutzen.

Basis-Theme wählen

Da jedes Theme auf einem Basis-Theme beruht, müssen Sie zunächst das Basis-Theme auswählen. Je nachdem, ob Sie den UI Theme Designer in der Cloud oder On-Premise-Version verwenden und abhängig von Ihrer Version, stehen Ihnen derzeit folgende Themes zur Verfügung:

■ SAP Belize

Dieses Theme löst mit SAP NetWeaver 7.5 das Theme Blue Crystal ab und wird ab dieser Version von SAP für SAP Fiori empfohlen.

■ Blue Crystal

Dieses Theme wird bis SAP NetWeaver 7.4 von SAP für SAP Fiori empfohlen und ist für SAPUI5 optimiert. Es kann auch für *Unified Rendering (UR)* verwendet werden, das von SAP GUI for HTML, von Web Dynpro für ABAP und von Web Dynpro für Java genutzt wird.

■ SAP Quartz Light

Mit SAPUI5-Version 1.65 wurde dieses neue Theme eingeführt. Es stellt das Basis-Theme für SAP Fiori 3.0 dar. Es ist minimalistischer als seine Vorgänger und soll den Inhalt der Anwendung in den Mittelpunkt stellen. Das Theme ist vollständig kompatibel zum Theme SAP Belize. Eine Umstellung auf dieses Theme stellt daher kein Problem dar, vorausgesetzt, Sie haben sich an die Vorgaben von SAP gehalten (schauen Sie sich dazu die goldenen Regeln in Kapitel 10, »Grundlagen der SAP-Fiori-Entwicklung«, an).

Nach dem Start des UI Theme Designers führt ein Assistent Sie durch das Anlegen eines neuen Themes. Im ersten Schritt werden die vorhandenen Themes aus dem UI Theme Repository geladen und in einer Liste angezeigt (siehe Abbildung 8.1). Aus dieser Liste können Sie dann ein beliebiges Basis-Theme auswählen.

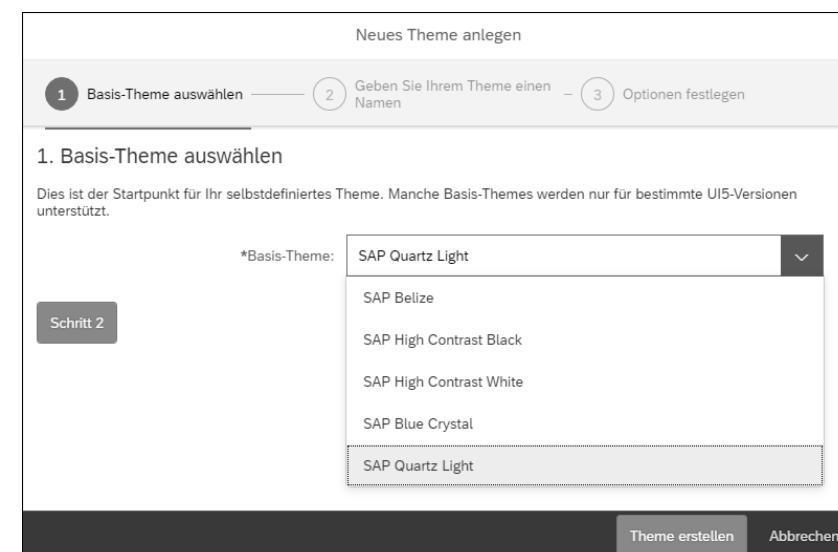


Abbildung 8.1 Basis-Theme auswählen

Nachdem Sie ein Theme ausgewählt haben, können Sie über einen Klick auf den Button **Schritt 2** dem neuen Theme eine eindeutige ID und einen Namen zuweisen. Über den Button **Schritt 3** können Sie anschließend auch

Neues Theme anlegen

einen Hersteller hinterlegen und festlegen, ob das Theme RTL (Right to Left) unterstützen soll (siehe Abbildung 8.2).

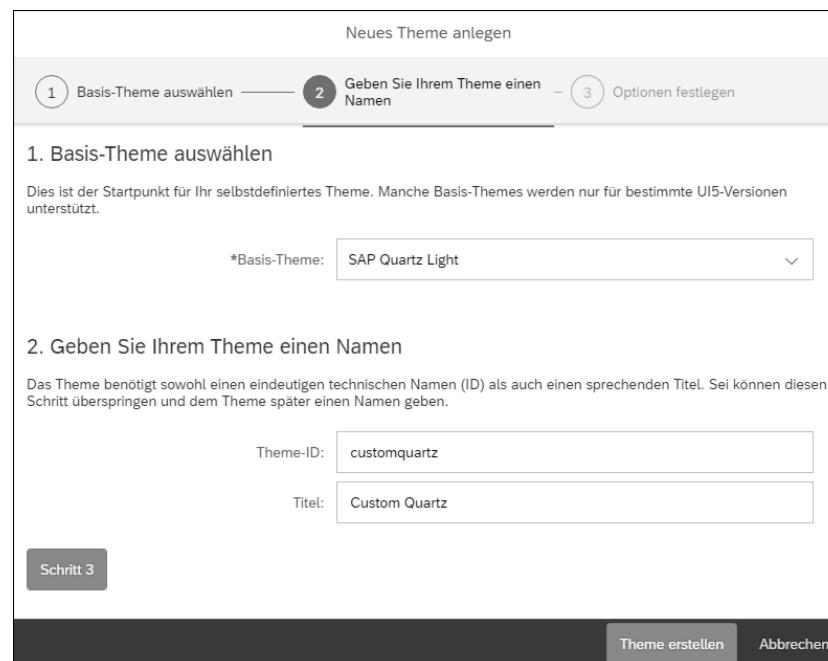


Abbildung 8.2 Theme-Details eingeben

Vorschau

Sie schließen die Theme-Erstellung über einen Klick auf den Button **Theme erstellen** ab. Damit Sie im Vorschaubereich die Auswirkungen Ihrer Änderungen sofort sehen können, geben Sie nun die URL einer Anwendung an. In unserem Fall ist das die URL des SAP Fiori Launchpads: http://<host>:<port>/sap/bc/ui5_ui2/ushell/shells/abap/FioriLaunchpad.html. Sie können auch mehrere Anwendungen angeben, indem Sie auf das Pluszeichen **+** klicken.

Beim Laden der Anwendung für die Vorschau ermittelt der UI Theme Designer, welche SAPUI5-Bibliotheken genutzt werden, und lädt die LESS-Dateien für diese Bibliotheken. Die Anwendung wird im Vorschaubereich angezeigt, und die Parameter der Bibliotheken werden im rechten Bereich des UI Theme Designers zum Editieren angeboten. Abbildung 8.3 zeigt einen Screenshot des UI Theme Designers mit SAP Quartz Light als Basis-Theme und dem SAP Fiori Launchpad als Anwendung.

Die eigentliche Arbeit mit dem UI Theme Designer besprechen wir in den folgenden Abschnitten. Zunächst gehen wir aber noch kurz auf das Speichern und Publizieren eines Themes ein.

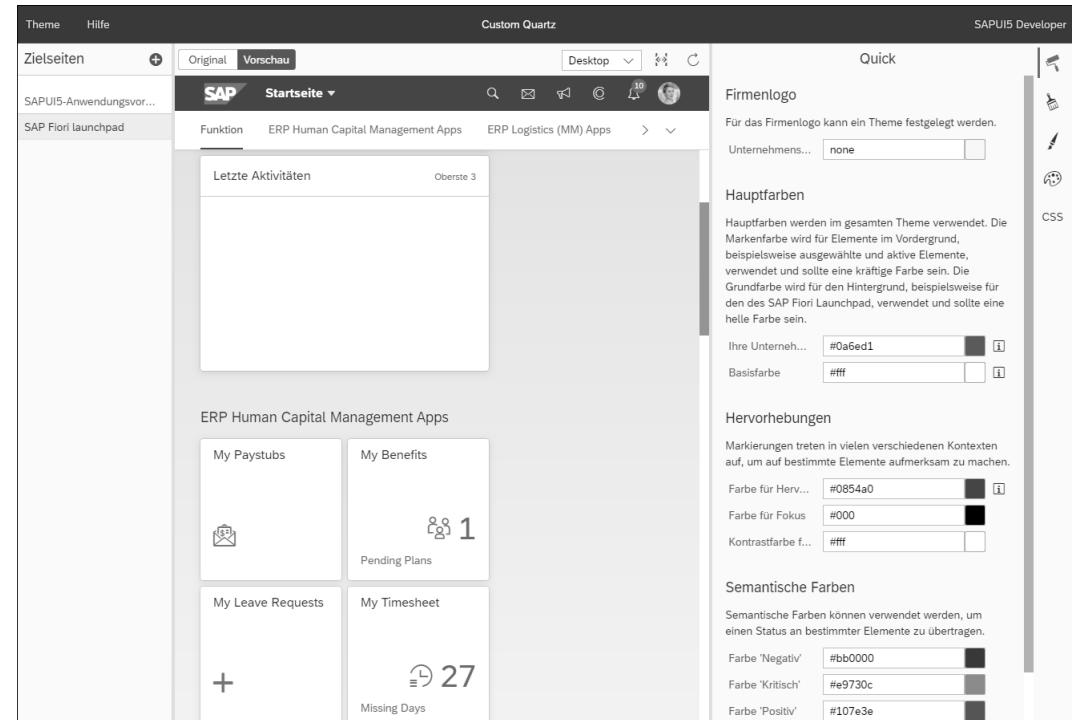


Abbildung 8.3 Das SAP Fiori Launchpad in der Vorschau des UI Theme Designers

Die Änderungen, die Sie an Ihrem Theme vornehmen, werden zunächst nur im Cache des Browsers gespeichert. Es empfiehlt sich deswegen, Ihre Arbeit zwischendurch öfter im Repository zu sichern.

Wählen Sie dazu die Menüpunkte **Theme** • **Sichern**. In der Dialogbox in Abbildung 8.4 tragen Sie die folgenden Angaben ein:

■ Theme-ID

Die Theme-ID ist eine ID, unter der Ihr Theme später referenziert werden kann. Hier sind nur alphanumerische Zeichen (insbesondere auch kein Unterstrich) erlaubt. Die ID darf nicht mit »SAP« beginnen.

■ Titel

Der Titel Ihres Themes ist optional und wird beim Start des UI Theme Designers in der Liste der verfügbaren Themes angezeigt. Auch der Titel darf nicht mit »SAP« beginnen.

■ Hersteller

Der Name der Firma, die dieses Theme vertreibt. Auch dieser darf nicht mit »SAP« beginnen.

Sichern

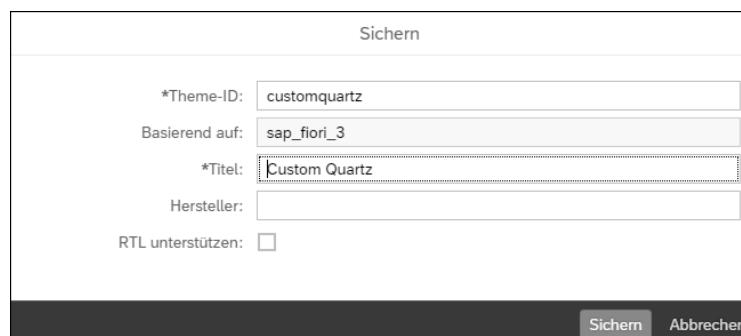


Abbildung 8.4 Sichern eines Themes

Falls Sie eine Right-to-Left-Version (RTL) des Themes benötigen, markieren Sie die entsprechende Auswahlbox. Dadurch werden Stylesheets sowohl für Left-to-Right- (LTR-) als auch für RTL-Sprachen erzeugt.

Bei der Vergabe des Titels und der ID ist es ratsam, sich auf das originale Theme zu beziehen.

Publizieren Beim Speichern werden nur die LESS-Dateien und Bildressourcen gespeichert. Damit Ihr Theme von einer Anwendung genutzt werden kann, müssen daraus CSS-Dateien erzeugt werden. Dazu wählen Sie **Theme • Sichern & erstellen** aus dem Menü aus. Der Dialog sieht genauso aus wie zuvor.

Im rechten Bereich des UI Theme Designers haben Sie über eine Symbolleiste Zugriff auf vier verschiedene Editoren:

■ Schnell

Hier können Sie schnell und einfach eine ausgewählte Anzahl von grundlegenden Parameterwerten ändern, die von allen UI-Technologien und Versionen verwendet werden.

■ Experte

Hier werden Ihnen mehr Parameter als beim Quick Theming angeboten, und Sie haben damit eine größere Kontrolle über das Erscheinungsbild.

■ Palette

Hier können Sie zusätzliche LESS-Parameter für eigene Farben definieren.

■ CSS

Hier können Sie Ihre eigenen LESS- und CSS-Regeln definieren. Leider wird dies für SAPUI5 nicht unterstützt.

In den folgenden Abschnitten geben wir Ihnen einige Beispiele für die Arbeit mit den vier Editoren.

8.2.1 Schnelles Theming

Im Theming-Editor **Schnell** werden Ihnen nur einige ausgewählte und grundlegende Parameter angeboten, die in allen UI-Technologien und -Versionen vorhanden sind. Die Parameter sind auch unabhängig von den Anwendungen, die Sie für die Vorschau ausgewählt haben. Wenn Sie sich nur auf diese Parameter beschränken, haben Sie die größtmögliche Stabilität bei Upgrades. Diese Parameter werden nicht mit ihrem technischen Namen, sondern mit einem nicht technischen Bezeichner benannt.

Wie es Abbildung 8.5 zeigt, können Sie mit dem Editor für schnelles Theming Parameterwerte für unterschiedliche Farben und Bilder anpassen.

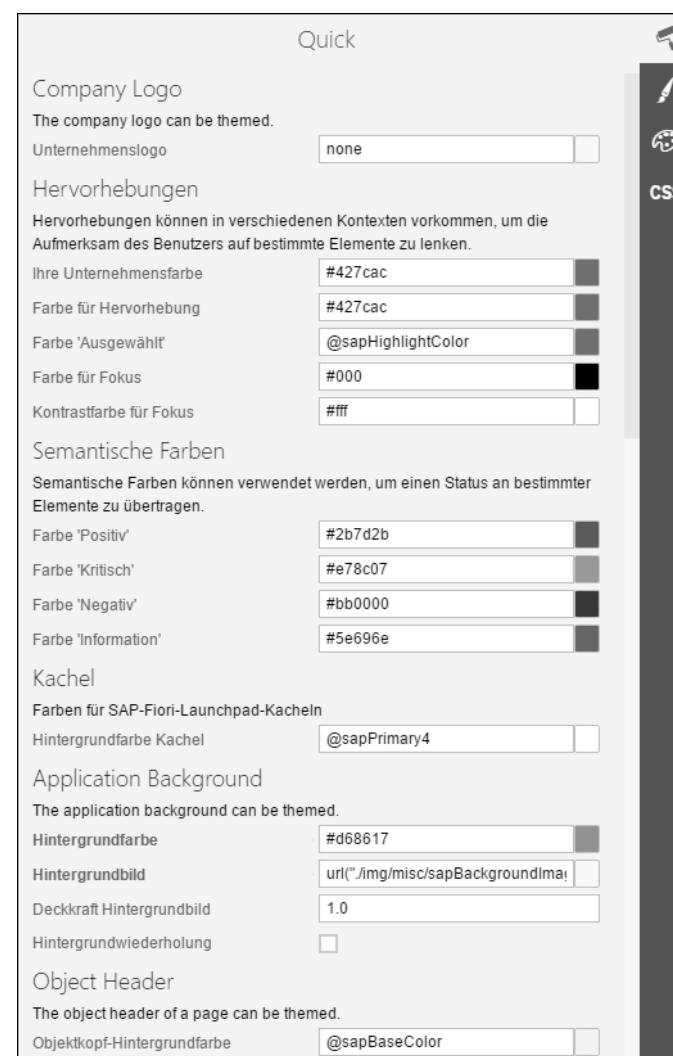


Abbildung 8.5 Der Editor für schnelles Theming

Farbauswahl Den RGB-Code für Farben können Sie direkt in die entsprechenden Textfelder eintippen. Sie können aber auch auf das Icon rechts neben dem Textfeld klicken, um einen Farbauswahldialog wie in Abbildung 8.6 zu öffnen.

Für Farbwerte können Sie anstelle eines hexadezimalen RGB-Codes auch CSS-Funktionen wie `rgba()` verwenden.

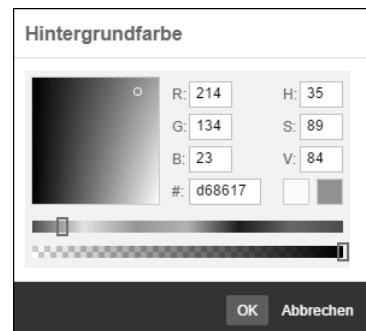


Abbildung 8.6 Farbauswahldialog

Bilder einbinden Um Bilder hinzuzufügen, klicken Sie ebenfalls auf die Schaltfläche neben dem Eingabefeld. So öffnen Sie den Bildauswahldialog aus Abbildung 8.7. Per Drag-and-drop können Sie Bilddateien von Ihrem lokalen Dateisystem auf diesen Dialog ziehen. Alternativ können Sie über das Wolken-Icon auch einen Dateidialog öffnen oder direkt die URL zum Bild in das Eingabefeld des Dateidialogs eintragen.

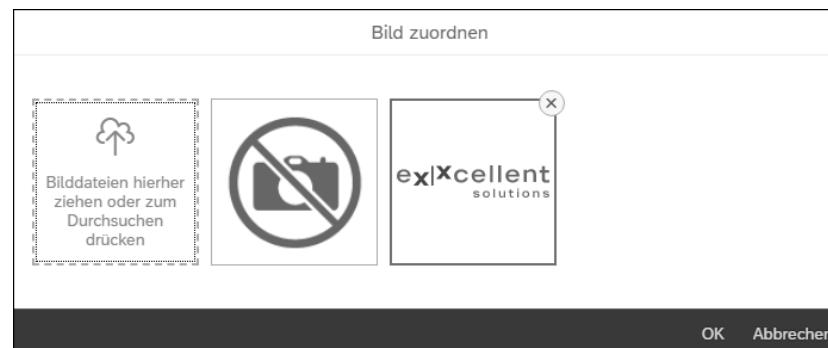


Abbildung 8.7 Der Bildauswahldialog

8.2.2 Expert Theming

Der Theming-Editor **Experte** (siehe Abbildung 8.8) zeigt alle Parameter des gewählten Basis-Themes an, die von den für die Vorschau angegebenen Anwendungen benutzt werden. Die Parameter werden mit ihrer technischen

ID angezeigt. Die Parameter, die im Expert Theming angezeigt werden, sind weit weniger robust gegen Versionsänderungen als die Parameter aus dem schnellen Theming. Sie beziehen sich zum Teil auf einzelne Controls und können auch spezifisch für eine bestimmte UI-Technologie sein.

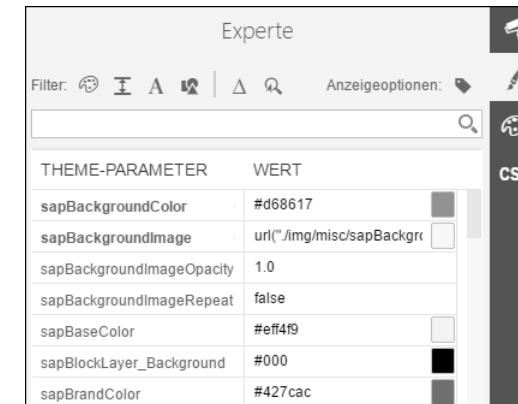


Abbildung 8.8 Experteneditor

In der obersten Zeile des Editors befinden sich fünf Filter-Icons, die in Tabelle 8.1 aufgeführt sind. Wenn Sie auf ein Icon klicken, wird der entsprechende Filter aktiviert, und Sie sehen nur die passenden Parameter. Klicken Sie auf ein weiteres Filter-Icon, wird die Auswahl noch weiter eingeschränkt. Klicken Sie z. B. auf das Paletten-Icon und das Font-Icon, sehen Sie nur Parameter, die sich auf die Farbe von Schriften beziehen. Die Filter werden also mit logischem UND kombiniert. Wenn Sie nochmals auf ein Filter-Icon klicken, wird der entsprechende Filter wieder entfernt.

Filter

Icon	Beschreibung
	zeigt Parameter für Farbwerte
	zeigt Parameter für Fonts
	zeigt Parameter für Höhen, Breiten und Abstände
	zeigt Parameter für Bilder
	zeigt nur gegenüber dem Original geänderte Parameter

Tabelle 8.1 Filter-Icons im Expert Theming

Tags Über das Icon  neben der Beschriftung **Anzeigeoptionen**: können Sie sich oberhalb des Suchfelds eine Liste von sogenannten *Tags* einblenden lassen. Auch diese Tags stellen Filter dar, die Sie zusätzlich aktivieren und wieder deaktivieren können. Abbildung 8.9 präsentiert den Experteneditor mit eingeschalteten Tags. Die beiden Tags **Content** und **Quick** wurden ausgewählt. Daher erkennen Sie nur die Parameter, die mit beiden Tags versehen wurden.

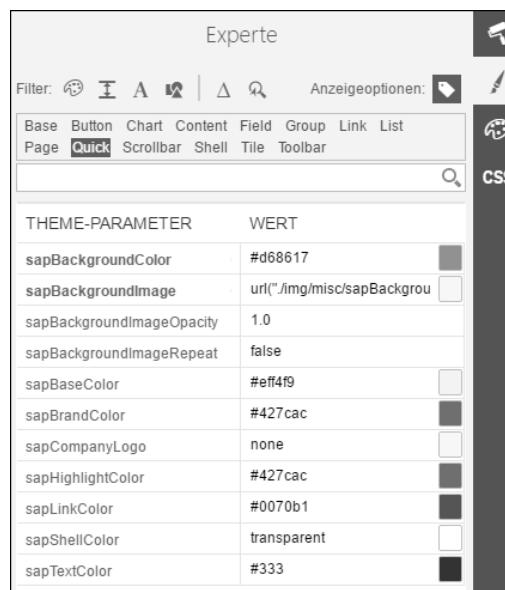


Abbildung 8.9 Filtern nach Tags

Suche Über das Suchfeld können Sie nach Parametern suchen, die einen bestimmten Substring enthalten. Tippen Sie dort z. B. »Global« ein und drücken dann die -Taste, werden nur Parameter angezeigt, die den String Global in ihrer ID enthalten. Um diesen Suchfilter wieder zu entfernen, klicken Sie auf das Kreuz.

LESS verwenden Für die Parameterwerte können Sie neben Literalen und CSS-Funktionen auch LESS-Funktionen verwenden. Damit können Sie den Wert eines Parameters aus den Werten anderer Parameter ableiten. In Abbildung 8.10 wird z. B. der Wert von `sapList_TableGroupHeaderBackground` mit der `darken`-Funktion aus den Werten von anderen Farbparametern abgeleitet.

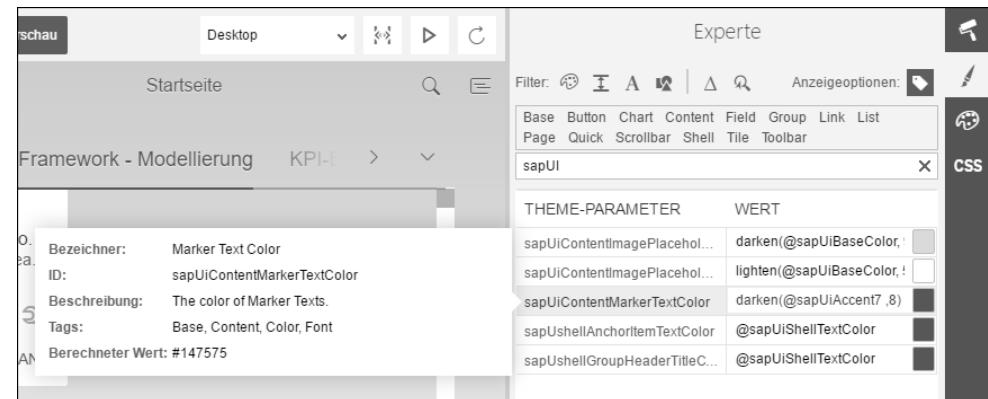


Abbildung 8.10 Verwendung von LESS im Expert Theming

8.2.3 Paletteneditor

Im Farbauswahldialog können Sie auch eigene zusätzliche Parameter für Farben definieren. Wechseln Sie dazu zum Theming-Editor **Palette**; dies ist der dritte der vier Editoren. Geben Sie im ersten Eingabefeld des Editors in Abbildung 8.11 einen Parameternamen ein, und wählen Sie dann einen Farbwert für Ihren neuen Parameter. Achtung: Sie können den Namen später nicht mehr ändern. Stattdessen müssen Sie den Eintrag löschen und einen neuen eingeben.

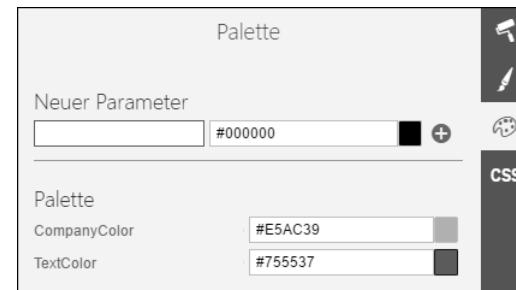


Abbildung 8.11 Paletteneditor

Sobald Sie Ihre zusätzlichen Parameter gepflegt haben, werden sie im Farbauswahldialog angezeigt, und Sie können sie dort auswählen.

8.2.4 Eigenen LESS- und CSS-Code einfügen

Die SAPUI5-Technologie unterstützt leider nicht die Verwendung von eigenem LESS- und CSS-Code im CSS-Editor. Bei anderen Technologien, z. B. Web Dynpro für ABAP, haben Sie damit noch größere Kontrolle über das Er-

scheinungsbild. Sie gehen aber das Risiko ein, dass Ihr Code nicht stabil gegen Versionsänderungen ist, da sich die HTML- und CSS-Definitionen der SAP User Interfaces im nächsten Release inkompatibel ändern können.

Listing 8.3 zeigt z. B., wie Sie die Hintergrundfarbe der CSS-Klasse `.sapUi-Body` anpassen können, sodass der Hintergrund einer Anwendung dieselbe Farbe wie die Markenfarbe hat, jedoch mit einer Opazität von 20 Prozent.

```
.sapUiBody {
    background-color: fade(@sapBrandColor, 20);
}
```

Listing 8.3 LESS-Code zum Anpassen der CSS-Klasse ».sapUiBody«

8.3 Themes verwalten

Es gibt zwei Werkzeuge, mit denen Sie Themes und die dazugehörigen Dateien analysieren und verwalten können:

- Der *UI Theme Repository Browser* ist eine Webanwendung, mit der Sie Verzeichnisstruktur und Dateien des UI Theme Repositorys betrachten können.
- Mit dem Werkzeug zur Verwaltung von kundendefinierten Themes können Sie Themes transportieren, importieren, exportieren und löschen.

In diesem Abschnitt schauen wir uns die wichtigsten Funktionen dieser beiden Werkzeuge an.

UI Theme Repository

Die Dateien für die SAP-Standard-Themes und die kundeneigenen Themes sind im UI Theme Repository gespeichert. Die obersten drei Hierarchieebenen der Verzeichnisstruktur sind folgendermaßen aufgebaut:

■ Framework

Auf der obersten Hierarchieebene befindet sich ein Verzeichnis für jedes UI Framework (UI5, NWBC, UR). Im Verzeichnis *Base* liegen Unterverzeichnisse und Dateien, die von allen Technologien genutzt werden, etwa die technologieübergreifenden SAP-Farben und Symbole.

■ Bibliothek

Auf der zweiten Hierarchieebene befinden sich unterhalb jedes Framework-Ordners die Ordner für die Bibliotheken des Frameworks. SAPUI5 enthält hier z. B. die Ordner *sap/ui/commons* oder *sap/m*. Im Verzeichnis *baseLib* gibt es bibliotheksübergreifende Unterverzeichnisse und Dateien.

■ Theme

Auf der dritten Hierarchieebene befindet sich je ein Ordner für jedes Theme. Die Ordner enthalten die LESS-Dateien des jeweiligen Themes. Ordner für SAP-Themes verwenden das Präfix *sap_*. Das Theme *base* ist ein Theme, auf dem jedes andere Theme direkt oder indirekt basiert.

Mit dem UI Theme Repository Browser unter der URL <http://<host>:<port>/sap/bc/theming/themes/~client-<client>> können Sie sich die Verzeichnisstruktur und die einzelnen Dateien anzeigen lassen. Sie können einzelne Dateien und ganze Verzeichnisse über den **Delete**-Link auch löschen. Unten im Bereich **Tools** können Sie auch neue Verzeichnisse erstellen und Dateien hochladen. Abbildung 8.12 zeigt dieses Werkzeug.

The screenshot shows a file tree structure with the root directory. The tree includes sub-directories like *Base*, *CRMWebUI*, *NWBC*, *UI5*, and *UR*. Each entry has a 'Type' column (DIR), a 'Last Modified' column, and a 'Delete' link. Below the tree is a 'Tools' section with fields for 'Upload' (with a 'Durchsuchen...' button), 'Dir' (with a 'Make Dir' button), and 'Upload' (a button).

Abbildung 8.12 UI Theme Repository Browser

UI Theme
Repository Browser

Verwaltungs-
werkzeug

Das Werkzeug zur Verwaltung von kundeneigenen Themes können Sie mit Transaktion */UI5/THEME_TOOL* starten. Dieser ABAP-Report zeigt die Kunden-Themes und die von diesen Themes verwendeten Verzeichnisse in einer ABAP-Liste an (siehe Abbildung 8.13).

Die IDs der Themes sind grün hinterlegt. Daneben finden Sie in eckigen Klammern die Bezeichnungen **Informationen**, **Transportieren**, **Herunterladen** und **Löschen**. Doppelklicken Sie auf eine dieser Bezeichnungen, um die entsprechende Funktion für das gewählte Theme auszuführen. Alternativ können Sie eine Bezeichnung mit einem einfachen Mausklick auswählen und dann auf die Lupe (F2) klicken.

UI-Theming: Werkzeug zur Verwaltung von Kunden-Themes			
UI-Theming: Werkzeug zur Verwaltung von Kunden-Themes			
[Aktualisieren]	[Cache invalidieren]	[Hochladen]	[Support]
Theme	baseTheme	[Informationen]	[Transportieren]
CRMWebUI/baseLib		7 MEN	23.08.2019
Theme	custom_belize	[Informationen]	[Transportieren]
Base/baseLib	12 MEN	23.08.2019	
CRMWebUI/baseLib	9 MEN	23.08.2019	
NWBC/baseLib	19 MEN	23.08.2019	
UI5/sap/ca/ui	11 MEN	23.08.2019	
UI5/sap/collaboration	10 MEN	23.08.2019	
UI5/sap/f	10 MEN	23.08.2019	
UI5/sap/fileviewer	10 MEN	23.08.2019	
UI5/sap/fiori	14 MEN	23.08.2019	
UI5/sap/gantt	10 MEN	23.08.2019	
UI5/sap/m	12 MEN	23.08.2019	
UI5/sap/mc	10 MEN	23.08.2019	
UI5/sap/ovp	10 MEN	23.08.2019	
UI5/sap/rules/ui	11 MEN	23.08.2019	
UI5/sap/suite/ui/commons	12 MEN	23.08.2019	
UI5/sap/suite/ui/generic/template	13 MEN	23.08.2019	
UI5/sap/suite/ui/microchart	12 MEN	23.08.2019	
UI5/sap/tnt	10 MEN	23.08.2019	
UI5/sap/ui/codemeditor	11 MEN	23.08.2019	
UI5/sap/ui/commons	12 MEN	23.08.2019	
UI5/sap/ui/comp	11 MEN	23.08.2019	
UI5/sap/ui/core	14 MEN	23.08.2019	
UI5/sap/ui/dt	11 MEN	23.08.2019	
UI5/sap/ui/export	11 MEN	23.08.2019	
UI5/sap/ui/f1	11 MEN	23.08.2019	
UI5/sap/ui/layout	11 MEN	23.08.2019	
UI5/sap/ui/mdc	11 MEN	23.08.2019	
UI5/sap/ui/richtexteditor	11 MEN	23.08.2019	
UI5/sap/ui/rta	11 MEN	23.08.2019	
UI5/sap/ui/suite	11 MEN	23.08.2019	

Abbildung 8.13 Werkzeug zum Verwalten von Kunden-Themes

Info-Anzeige Ein Doppelklick auf **Informationen** zeigt Ihnen die URL-Parameter für das Theme an. Ganz oben in Abbildung 8.14 sehen Sie den Parameter **sap-theme**. Diesen Parameter können Sie – über ein Fragezeichen getrennt – an die URL einer beliebigen SAPUI5-Anwendung hängen, um das Theme zu verwenden.

UI-Theming: Werkzeug zur Verwaltung von Kunden-Themes			
URL-Parameter für Theme custom_belize:			
sap-theme=custom_belize@https://sapdemo-s4.excellent.de:8001/sap/public/bc/themes/-client-100/-cache-B-EVsCvg8IQnnz7JulkGORhVMos			
sap-cssurl=https://sapdemo-s4.excellent.de:8001/sap/public/bc/themes/-client-100/-cache-B-EVsCvg8IQnnz7JulkGORhVMos/-legacy/sap_tradeshow/ur/ur_ie6.css			

Abbildung 8.14 Anzeige der URL-Parameter

Transport Um den Inhalt eines Themes in einen Customizing-Auftrag zu stellen, doppelklicken Sie im Startbildschirm auf den Eintrag **Transportieren** neben dem Namen des Themes. Wählen Sie im anschließenden Dialog einen Auf-

trag aus. Nachdem Sie den Transportauftrag freigegeben haben, können Sie die Transportdateien in ein anderes System transportieren.

Doppelklicken Sie im Startbildschirm der Transaktion auf **Herunterladen** neben dem Namen des Themes. Geben Sie im Dialog **Datei Speichern** ein Verzeichnis auf Ihrem lokalen Rechner an, in dem die ZIP-Datei mit dem Theme abgelegt werden soll. Die ZIP-Datei ist gemäß der Verzeichnishierarchie des Themes strukturiert und enthält alle Dateien, die Sie mit dem UI Theme Designer erstellt oder hinzugefügt haben.

Um ein Theme zu löschen, doppelklicken Sie im Startbildschirm des Werkzeugs auf **Löschen** neben dem Namen des Themes. Daraufhin erscheint eine Dialogbox, in der Sie den Löschvorgang bestätigen oder abbrechen können.

Im oberen Bereich des Startbildschirms befinden sich ebenfalls in eckigen Klammern die vier weiteren Bezeichnungen **Aktualisieren**, **Cache invalidieren**, **Hochladen** und **Support**. Ein Doppelklick auf **Aktualisieren** baut die Anzeige neu auf. Ein Doppelklick auf **Cache invalidieren** leert den Memory Cache des Servers. Bei einer anschließenden Anfrage nach dem Theme wird dies von der Datenbank neu in den Speicher-Cache gelesen. Diese Funktion können Sie auf dem Entwicklungssystem nutzen, wenn Sie ein Theme geändert haben und erneut testen wollen.

Ein Doppelklick auf **Hochladen** öffnet den Dialog **Datei öffnen**. In diesem Dialog können Sie eine ZIP-Datei vom Dateisystem Ihres lokalen Rechners auswählen. Diese ZIP-Datei muss ein archiviertes Theme enthalten.

Ein Doppelklick auf **Support** zeigt schließlich Support-Informationen zum UI Theme Designer an.

8.4 Themes verwenden

Um Ihr selbst definiertes Theme mit SAP Fiori zu verwenden, haben Sie verschiedene Möglichkeiten:

- URL-Parameter
- HTML-Attribut
- Customizing

Customizing ist die empfohlene Methode. Verwenden Sie URL-Parameter für Tests und HTML-Attribute für eigene SAPUI5-Anwendungen, die Sie isoliert – d. h. ohne weitere Anwendungen wie z. B. SAP Fiori – einsetzen.

Download

Theme löschen

8

Upload

URL-Parameter Über den URL-Parameter `sap-theme=<theme-id>@<theme-root>` können Sie das Theme festlegen. Dabei ist `<theme-id>` die ID Ihres Themes und `<theme-root>` das Wurzelverzeichnis des UI Theme Repositorys, `http://<host>:<port>/sap/public/bc/theme/~client-<client-nr>`. Diesen Parameter finden Sie über Transaktion /UI5/THEME_TOOL durch einen Klick auf **Info** heraus.

HTML-Attribut Sie können das Theme mit Transaktion /UI5/THEME_TOOL herunterladen und in einem Resources-Ordner einer SAPUI5-Anwendung speichern. Danach können Sie das Theme im Attribut `data-sap-ui-theme` des SAPUI5-Bootstrap-Skripts referenzieren, wie dies in Listing 8.4 gezeigt wird.

```
<script
  id='sap-ui-bootstrap'
  type = 'text/javascript'
  src = 'resources/sap-ui-core.js'
  data-sap-ui-theme = 'my_theme'
  data-sap-ui-libs = 'sap.m,sap.ui.table'
/>
```

Listing 8.4 Einbinden eines eigenen Themes im Bootstrap-Skript

Customizing Mit Transaktion /UI2/NWBC_CFG_CUST legen Sie Konfigurationsparameter für das SAP Fiori Launchpad fest. SAP liefert für einige der Parameter schon vordefinierte Werte aus, die Sie sich mit Transaktion /UI2/NWBC_CFG_SAP anschauen können. In Abbildung 8.15 sehen Sie als letzten Eintrag das Standard-Theme für das SAP Fiori Launchpad.

Sicht "NWBC-Konfiguration: Pflege (SAP)" anzeigen: Übersicht				
NWBC-Konfiguration: Pflege (SAP)				
Pfadfilter	Parametername	Übersetzbar	Parameterwert	Übersetzter Parameterwert
	ATTACH_SAP_PARAMETERS	<input type="checkbox"/>	/SRM-MDM/SRM_MDM	
	FORCE_BROWSER_STANDARD_MODE	<input type="checkbox"/>	^(?:\!bsap-!e=(w.) \$!webdynpro\ !ui5_u5\ !its\ web...	
NWEPM_DEMO	BRANDING_IMAGE	<input type="checkbox"/>	/SAP/PUBLIC/BC/NWDEMO_MODEL/IMAGES/COMPANY_LOGO...	
NWEPM_DEMO	BRANDING_URL	<input type="checkbox"/>	http://scn.sap.com/docs/DOC-31458	
NWEPM_DEMO	COCKPIT_TITLE	<input checked="" type="checkbox"/>		Enterprise Procurement Model (EPM)
NWEPM_DEMO	WELCOME_MESSAGE	<input checked="" type="checkbox"/>		Willkommen im Enterprise Procurement Model
SAP_FLP	THEME	<input type="checkbox"/>	sap_belize	

Abbildung 8.15 Vordefinierte Konfigurationsparameter

Mit Transaktion /UI2/NWBC_CFG_CUST können Sie andere Werte für diese Parameter definieren. Gehen Sie dazu in Transaktion /UI2/NWBC_CFG_CUST in den Änderungsmodus, und klicken Sie auf **Neue Einträge**. Tragen Sie in das Feld **Filter** »SAP_FLP« und in das Feld **Parametername** »THEME« ein (siehe Abbildung 8.16), und drücken Sie die **Enter**-Taste.

Abbildung 8.16 Konfigurationsparameter für das SAP Fiori Launchpad hinzufügen

Geben Sie nun in das Eingabefeld **Wert** die ID Ihres Themes ein, und speichern Sie anschließend (siehe Abbildung 8.17).

Abbildung 8.17 Konfigurationsparameter für das SAP Fiori Launchpad pflegen

Wählen Sie einen Customizing-Auftrag aus, und bestätigen Sie auch diesen Dialog.

8.5 UI Theme Designer der SAP Cloud Platform

Der UI Theme Designer ist seit einiger Zeit nicht mehr nur als On-Premise-Version verfügbar, sondern wird auch als Cloud-Dienst auf der SAP Cloud Platform angeboten. Analog zu den anderen Diensten dort erreichen Sie den UI Theme Designer, wie in Abbildung 8.18 gezeigt, über die Serviceübersicht im SAP Cloud Platform Cockpit. Nach dem Aufruf können Sie den UI Theme Designer wie die On-Premise-Variante verwenden. Alternativ können Sie den UI Theme Designer auch direkt aus SAP Web IDE aufrufen (Tools • UI Theme Designer).

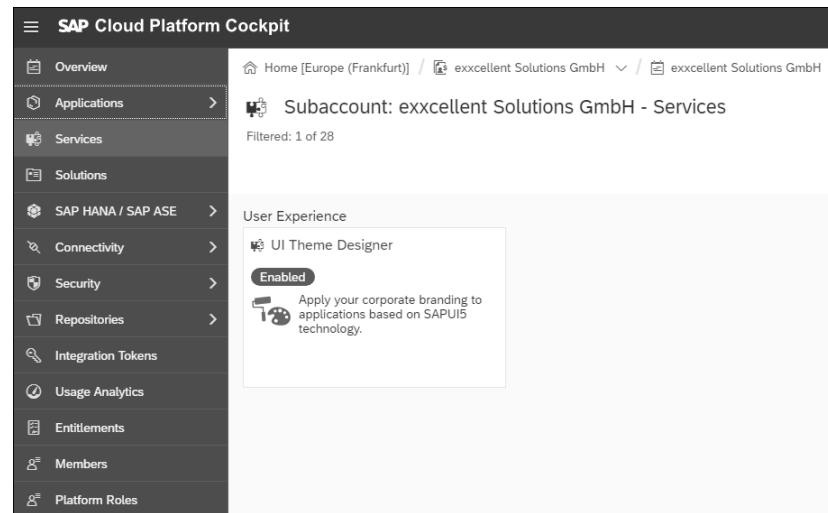


Abbildung 8.18 UI Theme Designer als Service auf der SAP Cloud Platform

Startbildschirm

Wenn Sie den UI Theme Designer auf der SAP Cloud Platform aufrufen, gelangen Sie in den in Abbildung 8.19 gezeigten Startbildschirm. Hier können Sie sowohl neue Themes erstellen als auch existierende bearbeiten. Wie Sie aus der Abbildung entnehmen können, ist es auch möglich, ein Theme zu importieren.

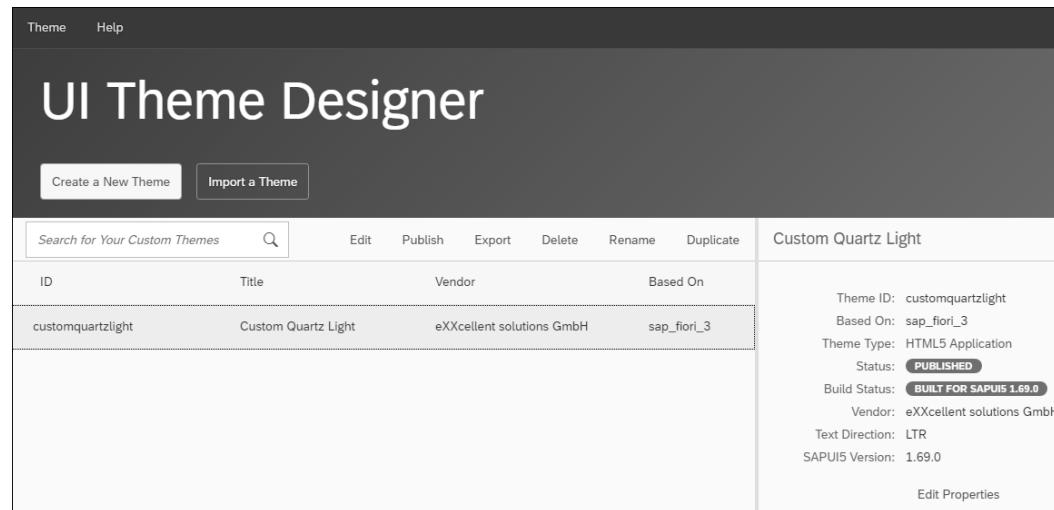


Abbildung 8.19 Startseite des UI Theme Designers der SAP Cloud Platform

Theme verwenden

Damit Sie ein Theme in einer Anwendung auf der SAP Cloud Platform verwenden können, muss es dort verfügbar sein. Dazu müssen Sie es, ähnlich

wie bei Ihrer On-Premise-Lösung, veröffentlichen (siehe Abschnitt 8.2, »Mit dem UI Theme Designer arbeiten«). Anschließend wird das Theme als HTML5-Anwendung auf der SAP Cloud Platform angezeigt (siehe Abbildung 8.20).

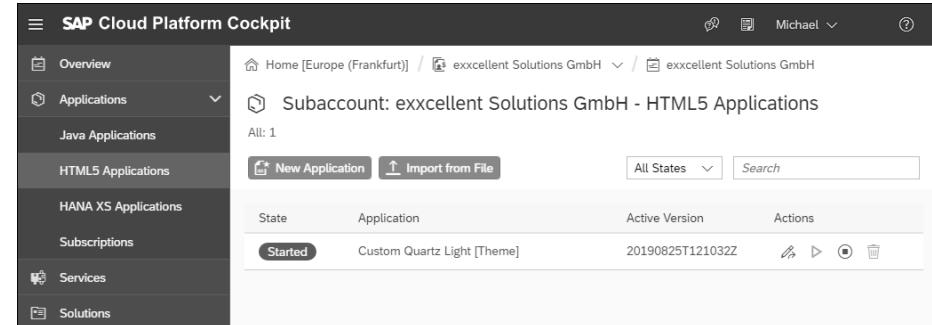


Abbildung 8.20 Veröffentlichtes Theme auf der SAP Cloud Platform

Um sich einen Überblick über den Inhalt des Themes zu verschaffen, können Sie das Theme auch in SAP Web IDE importieren. Wählen Sie dazu in SAP Web IDE **File • Import • Application from SAP Cloud Platform** aus. Sie werden sehen, dass sich der Aufbau nicht von dem zu Beginn des Kapitels beschriebenen unterscheidet.

Anschließend können Sie das Theme in einer Anwendung referenzieren. Verwenden Sie dafür entweder den in Abschnitt 8.4, »Themes verwenden«, vorgestellten URL-Parameter, oder wechseln Sie direkt in die **index.html**-Seite Ihrer Anwendung. Damit Sie das Theme auch in SAP Web IDE nutzen können, während Sie die Anwendung testen, müssen Sie die Datei **neo-app.json** noch mit einem Routing erweitern. Öffnen Sie die Datei dazu in Ihrem Projekt, und fügen Sie, wie in Listing 8.5 gezeigt, ein Routing für den Theme-Service ein. Der Wert des Attributs **path** ist dabei frei wählbar.

```
{
  "path": "/themes",
  "target": {
    "type": "service",
    "name": "theming",
    "entryPath": "/themes"
  },
  "description": "Theming Service"
}
```

Listing 8.5 Routing für Theme-Service in der Datei »neo-app.json«

Theme
referenzieren

Um das Theme direkt zu verwenden, ersetzen Sie den Wert des Attributs `data-sap-ui-theme` und weisen ihm den Namen des Themes zu, den Sie ihm beim Deployen gegeben haben. Als Nächstes müssen Sie ein Mapping definieren, das der Ablaufumgebung sagt, dass Sie ein Theme aus der SAP Cloud Platform verwenden. Definieren Sie dazu das Attribut `data-sap-ui-theme`, und weisen Sie ihm, wie in Listing 8.6 gezeigt, ein JSON-Objekt zu, das ein Attribut-Werte-Paar enthält. Das Attribut trägt die ID unseres Themes. Dem Attribut werden der Anwendungspfad, den Sie in der Datei `neo-app.json` definiert haben, und der Zusatz `/UI5` angehängt. In unserem Beispiel ergibt sich daraus `/themes/UI5`. Zusätzlich geben wir den Parameter `data-sap-ui-versionedLibCss` an und weisen ihm den Wert `true` zu.

```
<script id="sap-ui-bootstrap"
       src="resources/sap-ui-core.js"
       data-sap-ui-versionedLibCss="true"
       data-sap-ui-theme="customquartzlight"
       data-sap-ui-theme-roots=
         '{"customquartzlight" : "/themes/UI5/" }'
       data-sap-ui-resourceroots=
         '{"de.exxcellent.hello.HelloWorld": "./"}'
       data-sap-ui-compatVersion="edge"
       data-sap-ui-preload="async"
       data-sap-ui-libs="sap.m">
</script>
```

Listing 8.6 Ein Theme in der SAP Cloud Platform referenzieren

Einleitung

Im Laufe der letzten Jahrzehnte wurden unterschiedliche Technologien für die Entwicklung von Oberflächen von SAP entwickelt und vorangetrieben. Die anfangs konsolenbasierte Darstellung der Daten veränderte sich mit der Zeit. Heute arbeitet man mit Frameworks zur Erstellung von interaktiven Benutzeroberflächen. Die grundsätzlichen Treiber bei der Entwicklung der Frameworks waren stets die Technologie und der Wandel der Anforderungen im Laufe der Zeit, die jeweils eine neue Technologie und Vorgehensweise bei der Entwicklung von Oberflächen erforderten.

SAP hat in den letzten Jahren die folgenden Technologien für die Implementierung von Benutzeroberflächen für Webanwendungen entwickelt:

- **Internet Transaction Server (ITS)/SAP GUI**

Der Internet Transaction Server (ITS) bietet die Möglichkeit, klassische Dynpros als HTML-basierte Anwendungen darzustellen. Der Funktionsumfang dieser Technik ist sehr limitiert und auch nicht so in das SAP-System integriert, dass ITS von externer Seite (also aus dem Internet) aufgerufen werden kann.

- **Business Server Pages (BSP)**

Bei Business Server Pages handelt es sich um ein mit Release 6.10 des SAP Web Application Servers ausgeliefertes Framework zur Entwicklung von dynamischen Internetseiten. Mithilfe von BSP ist es möglich, ABAP-Anweisungen in HTML-Seiten zu implementieren. Beim Aufruf der Seite durch den Browser wird der dynamische ABAP-Code auf dem Server ausgeführt und integriert das Ergebnis als HTML-Code in die HTML-Seite. Das Resultat wird dann an den Aufrufer gesandt und entsprechend dargestellt.

- **Dynpages und JSPDynpages**

Die Grundidee dieser Technik folgt, ebenso wie BSP, dem Programmiermodell von Servlets und Java Server Pages (JSP) aus dem Java-Umfeld. Dynpages und JSPDynpages wurden im SAP-Enterprise-Portal-Umfeld eingeführt. Durch diese Technik ist es möglich, Java-Code in HTML-Seiten zu integrieren. Die daraus resultierenden Anwendungen können als sogenannte *iViews* in SAP Enterprise Portal integriert werden. Analog zu BSP-Seiten wird ein iView dieser Art auf dem Server generiert und der resultierende HTML-Strom an den Browser zurückgesandt.

■ Web Dynpro für Java

Mit Web Dynpro für Java wurde der damals aufkommende Trend von modellgetriebener Softwareentwicklung aufgegriffen und für die Entwicklung von webbasierten Anwendungen adaptiert.

■ Web Dynpro für ABAP

Web Dynpro für ABAP ist das ABAP-Pendant zu Web Dynpro für Java und der Nachfolger von Web Dynpro für Java.

SAPUI5 Getrieben durch den zunehmend mobiler werdenden Absatzmarkt, musste sich SAP Gedanken darüber machen, wie sich dieser Kanal mit den aktuellen UI-Technologien am besten beliefern lässt. Wie sich herausstellte, war es mit den bis dahin aktuellen UI-Technologien nicht möglich, auf die Vielfalt der mobilen Plattformen adäquat zu reagieren. Die Erstellung eines neuen Frameworks für die Entwicklung von browser-basierten Anwendungen wurde erforderlich. Es entstand das UI Framework *SAP UI Development Toolkit for HTML5* (SAPUI5).

SAP Fiori Es sollte jedoch nicht nur ein neues Framework entstehen, sondern es sollte sich auch die Art der Oberflächenentwicklung verändern. Damit einhergehend sollte auch ein konsistentes und in sich stimmiges Nutzererlebnis umgesetzt und etabliert werden. Das Ziel von SAP ist es dabei, ein einheitliches Bild für alle SAP-Oberflächen auf mobilen und nicht mobilen Endgeräten zu schaffen. Die Technologie rückt dabei in den Hintergrund, und der Nutzer mit seinen Bedürfnissen und Wünschen tritt an ihre Stelle. Das ist also *SAP Fiori*: kein Produkt, sondern ein Paradigmenwechsel in der Anwendungsentwicklung! Mittlerweile ist SAP Fiori in Version 2.0 seit einigen Jahren verfügbar und wird bei einer Vielzahl von Unternehmen eingesetzt. Da SAP in den vergangenen Jahren verschiedene Unternehmen dazugekauft hat und seine Strategie zunehmend auf die Unterstützung und Förderung intelligenter Unternehmen ausrichtet, ist es nun erforderlich geworden, auch SAP Fiori neu zu denken. Das Ergebnis ist SAP Fiori 3.0. Wenn auch zum Zeitpunkt der Entstehung der dritten Auflage dieses Buches (Herbst 2019) noch nicht vollständig veröffentlicht, wird SAP Fiori 3.0 perspektivisch das neue Gesicht aller Standardprodukte von SAP werden. Etwa durch die Einbindung eines digitalen Assistenten stellt es einen weiteren Meilenstein in der Entwicklung hin zu einem besseren Nutzererlebnis dar.

Was ist das Ziel dieses Buches?

Dieses Buch ist kein rein technisches Buch, genauso wenig, wie Sie SAP Fiori aus rein technischer Sicht betrachten können. Das Buch führt Sie in alle

Facetten von SAP Fiori ein und zeigt Ihnen, welchen Einfluss die Einführung von SAP Fiori auf das Design und damit auf die Interaktion zwischen Anwendung und Nutzer hat. Das Buch bietet deshalb neben der technischen Betrachtung von SAP Fiori auch eine Einführung in das Design eigener Nutzer-Erlebnis-Strategien und zeigt, wie das Interaktionsdesign aussehen kann.

Zielgruppe dieses Buches

Dieses Buch richtet sich an Administratoren, Softwareentwickler, technische Berater sowie Entscheider, die SAP-Fiori-Standardanwendungen im Unternehmen einführen möchten und/oder an ihre Unternehmensbedürfnisse anpassen müssen. Hier gehen wir auch auf die Programmierung komplett neuer SAP-Fiori-Apps ein.

Wir richten uns auch an Leser, die sich mit SAP Fiori im Allgemeinen auseinandersetzen müssen und dabei das Thema sowohl aus der infrastrukturellen als auch aus der technischen Sicht betrachten möchten. Einige der technischen Themen erfordern ein gewisses Maß an Erfahrung im SAP-Umfeld und auch in der Programmierung von Anwendungen mit ABAP, JavaScript sowie SAPUI5.

Im Buch werden einige Cloud-Services von SAP vorgestellt, z. B. *SAP Web IDE* oder die *SAP Cloud Platform*. Cloud-Dienste unterliegen einer recht häufigen Aktualisierung. Dies kann dazu führen, dass Screenshots und Beschreibungen im Buch nicht mehr vollständig der Oberfläche entsprechen, die Sie bei der Arbeit vorfinden. Ein Transfer der Ausführungen sollte jedoch kein größeres Problem darstellen. Ähnlich verhält es sich mit den SAPUI5-Versionen. Angesichts der rasanten Entwicklung ist es möglich, dass dargestellte Programmierauszüge nicht mehr in jedem Einzelfall der zu einem späteren Zeitpunkt empfohlenen Vorgehensweise entsprechen.

Zum Aufbau dieses Buches

Das vorliegende Buch gliedert sich in drei Teile: Der erste Teil gibt Ihnen einen Überblick, was SAP Fiori ist und welchen Einfluss SAP Fiori zukünftig auf die Entwicklung von Anwendungen im SAP-Umfeld haben wird. Im zweiten Teil führen wir Sie in die Implementierung von SAP Fiori ein. Wir zeigen Ihnen, welche Infrastruktur- und Systemvoraussetzungen implementiert werden müssen, damit SAP-Fiori-Anwendungen im Unternehmen eingesetzt werden können. Den Abschluss des Buches bildet eine Ein-

führung in die Erweiterung von SAP-Standardanwendungen sowie in die Entwicklung von eigenen SAP-Fiori-Anwendungen. Wir zeigen Ihnen auch, wie SAP Fiori in der Cloud zum Einsatz kommen kann und welche Gedanken man sich hinsichtlich der Entwicklungslandschaft machen muss, um SAP-Fiori-Anwendungen erfolgreich zu implementieren.

Teil I: SAP Fiori im Überblick

Kapitel 1 zeigt Ihnen, was SAP Fiori ist und welche grundsätzlichen Ideen dahinterstehen. Nach der Einführung thematisieren wir die neue User-Experience-Strategie von SAP und skizzieren, wohin die Reise für Unternehmen gehen wird.

Gegenstand von **Kapitel 2** ist der Wechsel der SAP-Strategien in der Oberflächenentwicklung, und wir zeigen Ihnen, welchen Einfluss dieser Strategiewechsel auf ein Unternehmen haben kann. Hier gehen wir auch auf die unterschiedlichen Aspekte bei der Definition von Benutzeroberflächen ein und stellen dar, wie das Umschwenken zum Nutzererlebnis den Softwareentwicklungsprozess beeinflusst.

In **Kapitel 3** stellen wir Ihnen die SAP Fiori Design Guidelines vor. Das Kapitel beinhaltet nach der Einleitung in die Grundkonzepte der Guidelines auch die Designvorgaben von SAP für SAP-Fiori-Anwendungen. Es werden die UI Patterns besprochen und deren Anwendung gezeigt.

Kapitel 4 führt in die Arbeit mit dem SAP Fiori Launchpad ein. Es stellt die Bestandteile des Launchpads sowie die Arbeit mit dem Launchpad aus Nutzersicht vor. Wir zeigen auch, wie sich das SAP Fiori Launchpad mit SAP Fiori 3.0 im Vergleich zu Version 2.0 weiterentwickelt hat. Den Abschluss des Kapitels bildet die Betrachtung des Launchpads aus technischer Sicht.

Teil II: Implementierung von SAP Fiori

In **Kapitel 5** steigen wir in die Architektur von SAP-Fiori-Anwendungen ein. Wir diskutieren die Systemlandschaft und die nötigen Systemkomponenten, um in die vier technischen Typen von SAP-Fiori-Anwendungen (Transactional Apps, Fact Sheets, Analytical Apps und SAP-Smart-Business-Apps) im Unternehmen einzuführen. Es wird dabei auch thematisiert, welche Rolle SAP HANA, SAP HANA XS, SAP Gateway, der SAP Web Dispatcher sowie unterschiedliche Browser und mobile Endgeräte bei der Nutzung von SAP-Fiori-Apps spielen. Das Kapitel gibt Ihnen auch eine kurze Einführung in die Architektur von SAP S/4HANA und die damit verbundenen Programmiermodelle für SAP-Fiori-Anwendungen im Kontext von SAP S/4HANA.

In **Kapitel 6** beschreiben wir die Installation von SAP Fiori. Dabei werden zunächst die Installationsvoraussetzungen auf SAP HANA, auf dem ABAP-Backend-Server und dem SAP-HANA-Frontend-Server beleuchtet. Daran schließt sich die Installation der produktübergreifenden Komponenten an, bevor dann die Installation der Komponenten erläutert wird, die spezifisch für die Produkte der SAP Business Suite (SAP ERP, SAP CRM, SAP SRM, SAP SCM, SAP GRC etc.) und SAP S/4HANA sind. Sie erfahren auch, wie Sie mittels der SAP-Fiori-Frontend-Server-Komponenten alle nötigen Bestandteile des Frontend-Severs installieren.

In **Kapitel 7** gehen wir auf die Konfiguration von SAP Fiori ein, beginnend mit der Konfiguration der Netzwerkinfrastruktur für eine sichere Kommunikation zwischen den einzelnen Komponenten und Single Sign-on und beschäftigen uns dann mit der Konfiguration von SAP Gateway auf dem Frontend-Server. Wir zeigen Ihnen auch, wie Sie unter SAP S/4HANA mittels New Rapid Content Activation schnell und einfach die SAP-Fiori-Aspekte des Systems aktivieren und zur Verfügung stellen.

In **Kapitel 8** beschreiben wir das Konzept des UI Theme Designers. Mit diesem Tool kann das visuelle Design von SAP-Fiori-Apps angepasst werden.

In **Kapitel 9** gehen wir auf die administrativen Aufgaben im Umfeld von SAP Fiori ein. Der Schwerpunkt liegt hier auf dem Monitoring und dem Troubleshooting von SAP Gateway und SAPUI5.

Teil III: SAP-Fiori-Anwendungen entwickeln

Kapitel 10 führt Sie in die Grundlagen und Konzepte von SAPUI5 ein. Neben der Implementierung einer einfachen Anwendung mit SAPUI5 zeigen wir Ihnen anhand von Best Practices, wie SAPUI5-Anwendungen architektonisch aufgebaut werden.

Kapitel 11 beschreibt, wie Backend-Dienste mit SAP Gateway implementiert werden. Neben einer Einführung in die technischen Aspekte der Plattform vermittelt Ihnen dieses Kapitel die Grundanatomie von Backend-Diensten. Wir zeigen Ihnen auch, wie Sie mit ABAP CDS Views arbeiten, OData-Dienste auf der Basis von CDS Views erzeugen und UI-Annotationen verwenden. Den Abschluss des Kapitels bildet eine Einführung in das *ABAP-RESTful-Programmiermodell*.

In **Kapitel 12** führen wir Sie in die Arbeit mit SAP Web IDE ein. Wir stellen Ihnen die Entwicklungsumgebung sowie die Arbeit mit diesem Werkzeug vor. Das Kapitel zeigt Ihnen, wie Sie mithilfe von Projektvorlagen einfach

und schnell eigene Anwendungen erstellen können. Schritt für Schritt entwickeln wir eine einfache master-detail-basierte Anwendung. Wir führen Sie in den Umgang mit SAP-Fiori-Elementen am Beispiel des List Report Patterns ein und zeigen Ihnen, wie Sie UI-Annotationen lokal und serverseitig nutzen. Sie erfahren außerdem, wie Sie SAP Overview Pages anlegen und Cards für Overview Pages erstellen.

Kapitel 13 erläutert die Erweiterungskonzepte von SAP Fiori und insbesondere von OData-Services und SAPUI5. Wir beschreiben anhand konkreter Beispiele, wie Sie Transactional Apps, Fact Sheets und Analytical Apps modifikationsfrei erweitern können. Das Kapitel stellt auch die Erweiterungskonzepte für SAP-Fiori-Anwendungen im Kontext von SAP S/4HANA vor und zeigt Ihnen, wie Sie mit Breakouts Anwendungen erweitern können, die auf SAP Fiori Elements basieren.

In **Kapitel 14** führen wir in SAP Fiori in der Cloud ein. Wir gehen auf die Grundidee von SAP Fiori in der Cloud ein und stellen eine entsprechende Infrastruktur sowie die Implementierung von SAP Fiori Cloud vor.

Kapitel 15 gibt schließlich eine Einführung in die Entwicklungsinfrastruktur und Konzepte, die bei der Entwicklung von SAP-Fiori-Anwendungen wichtig sind. Sie lernen *Continuous Integration* und *Continuous Delivery* kennen und erfahren, welche Empfehlung SAP für die Umsetzung gibt.

Symbolen Um Sie auf wichtige Informationen hinzuweisen und Ihnen so die Arbeit mit diesem Buch zu erleichtern, verwenden wir im Text Kästen mit den folgenden Symbolen:



Tipp: Kästen mit diesem Symbol geben Ihnen Empfehlungen zu Einstellungen oder Tipps aus der Berufspraxis.



Hinweis: Dieses Symbol weist Sie auf zusätzliche Informationen hin oder gibt Ihnen weiteres Hintergrundwissen.



Achtung: Mit diesem Symbol haben wir Warnungen oder Fallen gekennzeichnet.

Wie können Sie mit diesem Buch praktisch arbeiten?

Als Handbuch zum Thema SAP Fiori dient dieses Buch zum einen als Nachschlagewerk zu spezifischen Themenkomplexen und ist zum anderen aber auch so aufgebaut, dass Sie es von der ersten bis zur letzten Seite durcharbeiten können. Von der Webseite zum Buch www.sap-press.de/4974 können Sie sich die Coding-Beispiele herunterladen und so Schritt für Schritt die Beispiele aus dem Buch selbst nachvollziehen. Zusätzlich bieten wir

Ihnen zwei Abschnitte bzw. Kapitel der Vorauflage zum Download an: »Entwicklung von Backend-Diensten mit SAP HANA XS und SAP HANA XSA« und »Entwicklung einer SAP-Fiori-Anwendung mit Eclipse«.

Danksagung

Ich möchte vor allem dem Team beim Rheinwerk Verlag für seine professionelle Arbeit danken. Namentlich erwähnen will ich vor allem das zuständige Lektorat: Janina Karrasch hat das Buchprojekt in der ersten Auflage auf den Weg gebracht und von Anfang an für eine klare Struktur gesorgt. Kerstin Billen möchte ich für die herausragende Arbeit an der ersten Auflage und für ihre Unterstützung zu Beginn der Arbeit an der zweiten Auflage danken. Herrn Sven Clever danke ich für die Arbeit an der zweiten Auflage. Zu guter Letzt gilt mein Dank Daniela Kämmerer. Sie hat mich in der dritten Auflage des Buches tatkräftig unterstützt und aus dem Rohtext ein Buch in verständlicher Sprache gezaubert. Vielen Dank!

Ich möchte mich bei meiner Frau Michaela und meinen Kindern Domenik und Sophie bedanken. Wenn man Menschen nach ihren Hobbies fragt, erhält man oft die Antwort »meine Familie«. Ihr seid weit mehr als ein Hobby, Ihr seid mein Leben. Ihr helft mir, mich nicht in meiner Arbeit zu verlieren, trotz eines vollen Kopfes die Leichtigkeit des Seins nicht zu vergessen. Vielen Dank für eure Geduld, Rücksichtnahme und all die wunderbaren Stunden. Vielen Dank für alles, ohne euch wäre mein Leben sehr viel weniger geprägt von Freude, Lust am Leben und dem Motto, auch einmal Fünfe gerade sein zu lassen.

Ich bedanke mich auch bei meinem Arbeitgeber eXXcellent solutions GmbH für die Aufgaben, Entwicklungsmöglichkeiten und für die unglaublich gute und spannende Arbeitsumgebung. Nicht zu vergessen sind dabei meine Kollegen, die es mir durch ihre Sicht- und Denkweise möglich machen, immer wieder neue Horizonte zu entdecken. Die Zusammenarbeit mit Euch macht immense Freude. Danke.

Ihr Michael Englbrecht

Auf einen Blick

TEIL I SAP Fiori im Überblick

1	Was ist SAP Fiori?	29
2	User-Experience-Strategien entwickeln	65
3	SAP Fiori Design Guidelines	115
4	Das SAP Fiori Launchpad	179

TEIL II Implementieren von SAP Fiori

5	Architektur	207
6	Installation	239
7	Konfiguration	279
8	Anpassungen mit dem UI Theme Designer	357
9	Administration von SAP-Fiori-Anwendungen	379

TEIL III SAP-Fiori-Anwendungen entwickeln

10	Grundlagen der SAP-Fiori-Entwicklung	419
11	Entwicklung von Backend-Diensten	487
12	Entwicklung einer SAP-Fiori-Anwendung mit SAP Web IDE	547
13	Erweiterung von SAP-Fiori-Anwendungen	633
14	SAP Fiori Cloud	735
15	Entwicklungsinfrastruktur	763

Inhalt

Einleitung	19
------------------	----

TEIL I SAP Fiori im Überblick

1 Was ist SAP Fiori?

1.1 SAP Fiori – das Konzept	30
1.1.1 Strategie zur Umsetzung der neuen Nutzerfreundlichkeit	31
1.1.2 Die Grundsätze von SAP Fiori	34
1.1.3 Aus Transaktionen werden Apps	36
1.1.4 Das SAP Fiori Launchpad	37
1.2 SAP Fiori – das Design	39
1.3 SAP Fiori – die Technologie	42
1.3.1 Technische Applikationstypen	44
1.3.2 Das Frontend – SAPUI5	45
1.3.3 Die Kommunikationsprotokolle	53
1.3.4 SAP Gateway	54
1.3.5 Das Backend – SAP HANA und SAP HANA XS/XSA	55
1.4 Das intelligente Unternehmen	56
1.4.1 SAP CoPilot	57
1.4.2 SAP Inscribe	58
1.4.3 SAP Fiori im Kontext des intelligenten Unternehmens	58
1.5 Übersicht über die SAP-Fiori-Apps	59

2 User-Experience-Strategien entwickeln

2.1 Grundlagen	65
2.1.1 Was ist User Experience?	66
2.1.2 Design Thinking	69
2.1.3 UI vs. UX	71

2.2 Benutzeroberflächen gestalten	72
2.2.1 Mensch-Maschine-Kommunikation	72
2.2.2 Interaktionsdesign	74
2.2.3 Interaktionskonzepte	77
2.2.4 Visual Design	77
2.2.5 Einfluss der UX-Konzepte auf den Entwicklungsprozess	80
2.2.6 Mit SAP Build arbeiten	81

2.3 Ihre eigene UX-Strategie entwickeln	105
2.3.1 Was ist eine UX-Strategie?	106
2.3.2 Anwendungsfallzentrierte Strategieentwicklung	110
2.3.3 Analytische Strategieentwicklung	111
2.3.4 Taktische Strategieentwicklung	112
2.3.5 Fazit	113

3 SAP Fiori Design Guidelines	115
--	-----

3.1 Grundlagen	117
3.1.1 Unterstützung verschiedener Gerätetypen	117

3.1.2 Das Grid Layout	119
3.1.3 Letterboxing	120
3.1.4 Cozy Form Factor und Compact Form Factor	121
3.1.5 Schriftart	126
3.1.6 Icons	126

3.2 Floorplans	128
-----------------------------	-----

3.2.1 Seitenlayouts	129
3.2.2 Initial Page	139
3.2.3 List Report	140
3.2.4 Object Page	141
3.2.5 Object View	142
3.2.6 Edit Page	144
3.2.7 Worklist	149
3.2.8 Analytical List Page	150
3.2.9 Wizard	152
3.2.10 Overview Page	153

3.3 Die SAP Fiori for iOS Design Guidelines	155
--	-----

3.3.1 Die Grundlagen	156
3.3.2 Floorplans	156
3.3.3 Views	158

3.4 Die SAP Fiori for Android Design Guidelines	159
--	-----

3.4.1 Die Grundlagen	159
3.4.2 Views	160

3.5 SAP Conversational User Experience	163
---	-----

3.5.1 Designprinzipien	164
3.5.2 UI-Komponenten von SAP CUX	165

3.6 Konzeptionelle SAP-Fiori-Applikationstypen	165
---	-----

3.6.1 Approval App	166
3.6.2 Master-Detail App	170
3.6.3 Shopping App	171
3.6.4 Analysis Path Framework	171
3.6.5 SAP Smart Business Framework	176

4 Das SAP Fiori Launchpad	179
--	-----

4.1 Bestandteile des SAP Fiori Launchpads	179
--	-----

4.1.1 Die Homepage des SAP Fiori Launchpads	179
4.1.2 Das Viewport-Konzept	180
4.1.3 Me Area	182
4.1.4 Benachrichtigungsbereich	184
4.1.5 Der App Finder	185
4.1.6 Kacheln (Tiles)	186
4.1.7 Globale Suche	188
4.1.8 Die Startseite personalisieren	191
4.1.9 Key Performance Indicators modellieren	196

4.2 SAP Fiori Launchpad unter SAP Fiori 3.0	197
--	-----

4.3 Das Launchpad aus technischer Sicht	199
--	-----

4.3.1 Anwendungen integrieren	200
4.3.2 Die Rolle des Shell Containers	200

4.4 SAP-Fiori-Anwendungen bereitstellen	202
--	-----

TEIL II Implementieren von SAP Fiori

5 Architektur 207

5.1 Architekturüberblick und technische SAP-Fiori-Applikationstypen	207
5.1.1 Das SAP Fiori Launchpad	209
5.1.2 App-Typen	211
5.2 SAP HANA XS	212
5.2.1 Virtuelle Datenmodelle	213
5.2.2 OData-Services	214
5.3 SAP NetWeaver AS ABAP und die SAP Business Suite	215
5.3.1 OData-Service-Implementierungen und Geschäftslogik	215
5.3.2 Eingebettete Suche und Fact Sheets	217
5.4 ABAP-Frontend-Server	218
5.4.1 Zentrales UI-Add-on und SAPUI5	218
5.4.2 SAP Gateway und OData	222
5.4.3 Installationsoptionen für den ABAP-Frontend-Server	227
5.5 SAP Web Dispatcher	227
5.6 Endgeräte	228
5.6.1 HTML5-Browser auf Desktop, Tablet und Smartphone	228
5.6.2 Hybrid Apps auf mobilen Endgeräten	229
5.7 Architektur im Kontext von SAP S/4 HANA	231
5.7.1 SAP Fiori in einer SAP-S/4HANA-Systemlandschaft	231
5.7.2 ABAP-Programmiermodell für SAP Fiori	233
5.7.3 ABAP-RESTful-Programmiermodell	236
5.7.4 Transactional Apps	236
5.7.5 Analytical Apps	237

6 Installation 239

6.1 Installation planen	240
6.1.1 SAP Fiori Apps Reference Library	241
6.1.2 SAP Rapid Deployment Solutions	242
6.1.3 SAP-Online-Hilfe	244
6.1.4 SAP-Hinweise	250

6.2 Installationsvoraussetzungen überprüfen

6.2.1 Voraussetzungen für die Installation in SAP HANA	253
6.2.2 Voraussetzungen für den ABAP-Backend-Server	254
6.2.3 Voraussetzungen für den ABAP-Frontend-Server	256

6.3 Softwarekomponentenversionen herunterladen

6.3.1 Software Download Center	258
6.3.2 SAP Solution Manager	258

6.4 Installationswerkzeuge

6.4.1 SAP HANA Application Lifecycle Management	260
6.4.2 SAP Add-on Installation Tool	260
6.4.3 Software Update Manager	261

6.5 Plattformkomponenten installieren

6.5.1 SAP Smart Business for SAP Business Suite Foundation Component auf SAP HANA installieren	264
6.5.2 SAP-Gateway-Komponenten auf den ABAP-Backend-Servern installieren	265
6.5.3 SAP-Gateway-Komponenten auf dem ABAP-Frontend-Server installieren	265
6.5.4 User Interface Add-on for SAP NetWeaver auf dem ABAP-Frontend-Server installieren	266
6.5.5 SAP-Fiori-Frontend-Server installieren	268
6.5.6 SAP Web Dispatcher installieren	271

6.6 Produktspezifische Komponenten installieren

6.6.1 Virtuelle Datenmodelle und OData-Services auf SAP HANA XS installieren	273
6.6.2 Produktspezifische Komponenten auf den ABAP-Backend-Servern installieren	273
6.6.3 Produktspezifische OData-Services und UI-Komponenten auf dem ABAP-Frontend-Server installieren	273
6.6.4 SAP-Fiori-Apps mit dem Maintenance Planner installieren	274

7 Konfiguration 279

7.1 Kommunikationskanäle und Single Sign-on konfigurieren

7.1.1 HTTPS einrichten	280
7.1.2 Routing im SAP Web Dispatcher einrichten	284

7.1.3	RFC-Vertrauensbeziehung vom ABAP-Backend- zum ABAP-Frontend-Server einrichten	286
7.1.4	Initiale Authentifizierung durch den ABAP-Frontend-Server	288
7.1.5	Authentifizierung am ABAP-Backend-Server für Suchanfragen	289
7.1.6	Authentifizierung in SAP HANA XS	290
7.1.7	Benutzerabgleich zwischen den Servern	291
7.2	ABAP-Frontend-Server konfigurieren	291
7.2.1	Spracheinstellungen festlegen	292
7.2.2	SAP-Systemalias einrichten	293
7.2.3	SAP Gateway aktivieren	294
7.2.4	OData-Services für die zentralen UI-Komponenten registrieren	294
7.2.5	SICF-Services für die zentralen UI-Komponenten freischalten	296
7.2.6	Rollen für das SAP Fiori Launchpad kopieren, pflegen und zuordnen	296
7.2.7	Konfigurationswerkzeuge unter SAP S/4HANA	298
7.3	SAP Fiori Launchpad und Transactional Apps	303
7.3.1	Konzept der Launchpad-Konfiguration	303
7.3.2	LPD_CUST Launchpads und -Anwendungen anlegen	304
7.3.3	Semantische Objekte anlegen	308
7.3.4	Kataloge anlegen	309
7.3.5	Zielzuordnungen konfigurieren	312
7.3.6	Statische und dynamische App-Launcher-Kacheln konfigurieren	315
7.3.7	Nachrichtenkachel konfigurieren	316
7.3.8	Gruppen anlegen	320
7.3.9	Kataloge und Gruppen zu Rollen zuordnen	321
7.3.10	SAP Fiori Launchpad Content Manager für SAP S/4HANA	323
7.4	SAP-Fiori-Suche und Fact Sheets konfigurieren	326
7.4.1	Business Functions aktivieren	326
7.4.2	Sekundäre Datenbankverbindung einrichten	327
7.4.3	Konnektorbasierte Berechtigungsprüfungen aktivieren	328
7.4.4	ICF-Services auf dem Backend aktivieren	328
7.4.5	Konnektoren anlegen und Indizierung starten	329
7.4.6	Enterprise-Suche für mehrere Backend-Server konfigurieren	331

7.4.7	Rollen und Benutzerzuordnung auf dem Backend-Server anpassen	331
7.4.8	Rollen und Benutzerzuordnung auf dem Frontend-Server anpassen	332
7.4.9	SICF- und OData-Services auf dem Frontend-Server aktivieren	333
7.5	KPI Modeler und Analytical Apps	334
7.5.1	KPI Framework freischalten	334
7.5.2	Privilegien auf SAP HANA gewähren	335
7.5.3	Konfiguration von Hybrid Apps	336
7.5.4	KPIs anlegen	337
7.5.5	KPI-Kacheln verwenden	344
7.6	KPI Design und Analytical Apps unter SAP S/4HANA	345
7.6.1	Analytical Apps konfigurieren	345
7.6.2	KPI Design Apps verwenden	347

8 Anpassungen mit dem UI Theme Designer 357

8.1	Konzept des UI Theme Designers	358
8.1.1	CSS und LESS	358
8.1.2	Parameter	359
8.2	Mit dem UI Theme Designer arbeiten	360
8.2.1	Schnelles Theming	365
8.2.2	Expert Theming	366
8.2.3	Paletteneditor	369
8.2.4	Eigenen LESS- und CSS-Code einfügen	369
8.3	Themes verwalten	370
8.4	Themes verwenden	373
8.5	UI Theme Designer der SAP Cloud Platform	375

9 Administration von SAP-Fiori-Anwendungen 379

9.1	Aufgaben eines SAP-Fiori-Administrators	379
9.1.1	Periodische Aufgaben	379

9.1.2	Benutzermanagement	380
9.1.3	Transportmanagement	381
9.2	Monitoring	381
9.2.1	Überwachen von SAP Gateway	382
9.2.2	Überwachen des UI-Add-ons für SAP NetWeaver	396
9.3	Troubleshooting	397
9.3.1	SAP Fiori Launchpad und Launchpad Designer	397
9.3.2	SAP-Fiori-Apps	401
9.3.3	SAP Gateway	408

TEIL III SAP-Fiori-Anwendungen entwickeln

10	Grundlagen der SAP-Fiori-Entwicklung	419
10.1	SAPUI5	419
10.1.1	Die Entwicklungsumgebung	420
10.1.2	Aufbau einer mobilen Anwendung	428
10.1.3	Implementierung einer mobilen SAPUI5-Anwendung	430
10.2	Best Practices bei der Entwicklung	453
10.2.1	Das Komponentenmodell von SAPUI5	454
10.2.2	Der Application Descriptor	459
10.2.3	Routing	462
10.2.4	Mit Modellen arbeiten	473
10.2.5	Die Datei »testFLPService.html«	478
10.2.6	Simulation von Backend-Systemen	481
10.2.7	Dos and Don'ts	483

11	Entwicklung von Backend-Diensten	487
11.1	SAP Gateway und OData-Services	487
11.1.1	Beschreibung des Datenmodells	488
11.1.2	Generierung der Laufzeitartefakte	494
11.1.3	Implementierung der Data-Provider-Klasse	498
11.2	ABAP Core Data Services	513
11.2.1	Ein ABAP-Projekt anlegen	514
11.2.2	ABAP CDS View als OData-Service bereitstellen	521

11.2.3	Implementierung eines komplexeren OData-Services	524
11.2.4	Arbeiten mit UI-Annotationen	528
11.3	Das ABAP-RESTful-Programmiermodell	535
11.3.1	Die Architektur	535
11.3.2	Implementierung einer SAP-Fiori-Anwendung	536

12 Entwicklung einer SAP-Fiori-Anwendung mit SAP Web IDE

547

12.1	Einführung in die SAP Web IDE	548
12.2	Mit SAP Web IDE arbeiten	555
12.2.1	Die Versionsverwaltung	555
12.2.2	Konfiguration der Backend-Konnektivität	557
12.2.3	Weiterführende Konfigurationseinstellungen	559
12.2.4	Mit Projekten arbeiten	561
12.3	Eine Anwendung mit SAP Web IDE entwickeln	564
12.3.1	Projekt anlegen	565
12.3.2	Oberfläche entwickeln	572
12.3.3	Anwendung ausliefern	581
12.3.4	Mockup-Möglichkeiten für die Simulation von Backend-Systemen nutzen	584
12.4	SAP Fiori Elements	588
12.4.1	Arbeiten mit dem List Report Template	590
12.4.2	Mit dem Overview Page Template arbeiten	616
12.5	Mit SAP-Build-Prototypen arbeiten	629

13 Erweiterung von SAP-Fiori-Anwendungen

633

13.1	Einführung und Überblick	634
13.1.1	Erweiterungen im Backend	634
13.1.2	Erweiterung von OData-Services	636
13.1.3	Erweiterungen von Fact Sheets	637
13.1.4	Erweiterung des User Interfaces	638
13.1.5	Dokumentation der Erweiterungsmöglichkeiten	638

13.2 Erweiterungen im Backend	639
13.2.1 Erweiterung im ABAP Dictionary	639
13.2.2 Erweiterungen von Analytical Views	640
13.2.3 Erweiterungen der Geschäftslogik	641
13.3 Erweiterung von OData-Services	646
13.3.1 Backend-Server	646
13.3.2 SAP HANA XS	651
13.4 Erweiterung von Fact Sheets	652
13.4.1 Erweiterung des Suchmodells	653
13.4.2 Die Annotationsdatei anpassen	660
13.4.3 Konfiguration des SAP Fiori Launchpads	662
13.5 Erweiterung des User Interfaces	663
13.5.1 Das Erweiterungsprojekt vorbereiten	665
13.5.2 Views erweitern	668
13.5.3 Erweiterung von Texten	671
13.5.4 Konfiguration von Service-URLs	672
13.5.5 Erweiterung von Controllern	673
13.5.6 Views modifizieren	676
13.5.7 Views ersetzen	678
13.5.8 Konfiguration der Navigations-Routings	680
13.5.9 Hochladen und Konfigurieren der Erweiterung	681
13.6 Erweiterung der App »My Inbox« mit BAdls	682
13.6.1 Kurze Einführung zu SAP Business Workflow	683
13.6.2 Implementierung der App »My Inbox«	690
13.6.3 Implementierung der BAdls für das Resultat	697
13.6.4 Implementierung der BAdls für Zusatzanzeigen	703
13.7 Anpassungen mit den SAPUI5 Flexibility Services	706
13.7.1 Einführung in die SAPUI5 Flexibility Services	707
13.7.2 Nutzung der SAPUI5 Flexibility Services	708
13.8 Implementierung von Erweiterungen für das ABAP-Programmiermodell für SAP Fiori	713
13.8.1 Einführung in die Erweiterungskonzepte für ABAP CDS ..	714
13.8.2 Erweiterung einer SAP-Fiori-Anwendung	717
13.9 Ausblick	733

14 SAP Fiori Cloud	735
14.1 Deployment-Optionen	739
14.2 Entwicklung von SAP-Fiori-Cloud-Lösungen	742
14.2.1 Konfiguration von SAP Cloud Platform OData Provisioning	743
14.2.2 Implementierung einer einfachen Master-Detail-Anwendung	749
14.2.3 Konfiguration eines SAP Fiori Cloud Launchpads	753
14.3 Fazit	762
15 Entwicklungsinfrastruktur	763
15.1 Werkzeuge	764
15.1.1 Entwicklungsumgebung	765
15.1.2 Versionsverwaltung	765
15.1.3 Bundling	771
15.2 Einführung in Continuous Integration und Continuous Delivery	775
15.2.1 Continuous Integration	776
15.2.2 Continuous Delivery	776
15.2.3 Zusammenspiel von CI und CD	777
15.2.4 Empfehlungen von SAP	780
Anhang	781
A Quellenhinweise	781
B Der Autor	785
Index	787