



© Sertavio, Shutterstock.com

Mehr als eine technische Fragestellung

HTML5 vs. SAPUI5

Nach wie vor beherrschen alt wirkende, blau dominierte Oberflächen die Standard-SAP-Welt. Sie orientieren sich weniger am individuellen Nutzen als vielmehr an der Abbildung von allgemeingültigen Prozessen. Aber: SAP Dynpro stirbt – der Trend geht zu WebDynpro und SAPUI5/HTML5.

Von Dirk Hirschmann und Elias Mistler, Pikon

Es hat lange gedauert, bis auch SAP moderne UI-Technologien auf den Weg gebracht hat. Schwierig ist vor allem die Anpassungsfähigkeit an individuelle Anforderungen und die wenig intuitive Bedienbarkeit. Im Rahmen einer Evaluierungsphase für HTML5-UI-Technologien entstand die Idee, einen Artikel über Oberflächentechnologien zu schreiben. Vor allem über solche, die im SAP-Umfeld typischerweise eingesetzt werden.

Das WebDynpro Framework ist mittlerweile bewährt und bietet auch interessante Möglichkeiten für Eigenentwicklungen. So kann man vordefinierte Templates für Benutzeroberflächen und eine komfortable Entwicklungsumgebung mit einem WYSIWYG-Editor nutzen.

Wie bei allen geschlossenen Frameworks ist SAP als Hersteller der Rahmengeber und gibt somit die Grenzen strikt vor. Somit wird die gesamte Steuerung der Kommunikation mit dem Backend-System auf dem Server definiert. Es gibt keine Möglichkeit für ein individuelles Coding auf der Client-Seite, wie es zum Beispiel bei der Einbindung bestimmter multimedialer Inhalte (z. B. Video) notwendig wäre. Andererseits bieten diese Grenzen auch Vorteile, gerade bei Themen, wie Releaseplanung, Support sowie der Datensicherheit und Stabilität. Aufgrund der beschriebenen Begrenzungen geht SAP neue Wege und stellt ein noch relativ neues UI-Framework zur Verfügung: SAPUI5!

Bisher ist SAPUI5 hauptsächlich durch die Fiori-Apps bekannt, aber es handelt sich hierbei um ein offeneres Konzept mit mehr Eingriffsmöglichkeiten im Vergleich zu WebDynpro.

SAPUI5

SAPUI5 ist ebenfalls ein Framework und bietet eine Sammlung von Design- und Steuerungselementen auf Basis von HTML5. Weiter stellt SAP mit SAPUI5 ein Konzept zum Data Binding zur Verfügung, wie es bereits in ähnlicher Form aus WebDynpro bekannt ist. Damit wird der Datenaustausch zwischen Oberfläche und Backendsystemen (mit SAPUI5 lassen sich auch Non-SAP-Systeme anbinden) gewährleistet. Hierzu hält das SAPUI5-Framework eine Form des Model View Controller (MVC) bereit, wodurch geregelt wird, wie die Daten von der Oberfläche in die Businesslogik bzw. von der Businesslogik in die Oberfläche gelangen. Mithilfe dieses Prinzips können schnell und einfach Oberflächen produziert werden, um erste Vorhaben umzusetzen und Eindrücke der Lösung zu vermitteln. Das Framework setzt sich dabei aber nicht nur aus SAP-eigenen Bibliotheken zusammen, sondern bedient sich weiterer Open Source Libraries wie jQuery, Bootstrap, Cordova/Phonegap.

Aktuell gibt es zwei Ausprägungen (auch Kombinationsvarianten genannt): eine für mobile Anwendungszwecke und eine für den Desktop-Bereich.

Ein Unterschied zwischen den beiden Varianten liegt darin, dass die Mobile-Variante aus Performancegründen eine kleinere und auf mobile Anwendungen hin optimierte Bibliothek umfasst. Durch die leistungsstarken mobilen Endgeräte, die immer besser werdende Netzabdeckung und die höheren mobilen Datenraten wird der Unterschied zwischen diesen Variationen vernachlässigbar. Lediglich die Anpassung an unterschiedliche Bildschirmgrößen mithilfe von Responsive Design ist wirklich relevant.

Frameworks

Bei SAPUI5 handelt es sich, wie gesagt, um ein Framework. Zur Einordnung wollen wir deshalb kurz auf die Vor- und Nachteile solcher Frameworks eingehen.

Bei einer Eigenentwicklung steht man immer vor der Frage, ob man bereits existierende Frameworks verwendet oder ob alles selbst entwickelt wird. Dazu hört man als Entwickler oft die Aussage: „Verwende doch lieber ein vorhandenes Framework und erfinde nicht immer das Rad neu.“ Es gibt aus unserer Sicht zwei Indikatoren, die Aufschluss für die Nutzung eines Frameworks geben können: erstens der Einsatzzeitraum der Lösung, zweitens die Flexibilität im Hinblick auf die Anpassbarkeit an neue Anforderungen.

Will man schnell etwas Vorzeigbares erzeugen, können Frameworks mit Sicherheit das Mittel der Wahl sein. Will man die

Lösung auch langfristiger einsetzen oder sogar dauerhaft für Änderungen offenhalten, sind beim Einsatz von Frameworks weitere Maßnahmen nötig, um deren Nachteile zu mildern. Grundsätzlich sollte man Frameworks oder verwendete Komponenten immer so einbinden, dass sie nicht elementarer Bestandteil der Lösung sind, um funktionale Abhängigkeiten zu vermeiden. Dies kann durch eine Softwarearchitektur erreicht werden, die von Fremdkomponenten bestmöglich abstrahiert, damit diese austauschbar bleiben. Zum Beispiel sollte es in der Ablauflogik keine Rolle spielen, welche Eingabemöglichkeiten die GUI zur Verfügung stellt. Dies alles führt bei uns zu dem Entschluss, dass wir uns gerne anschauen, wie ein Framework arbeitet, um dann ggf. einzelne Konzepte daraus zu übernehmen. Natürlich ist dieses Vorgehen auf den ersten Blick aufwändig, doch für Erweiterungsanforderungen sind dann auch die entsprechenden Möglichkeiten bekannt und die Anpassungen können wesentlich kontrollierter und damit sicherer erfolgen.

Frameworks setzen, wie der Name schon sagt, einen Rahmen, um Lösungen zu erstellen. Diese Einschränkungen können dabei helfen, schnelle Ergebnisse zu produzieren, da ein Framework uns viele Entscheidungen (bewusst oder unbewusst) abnimmt. Aber wer entscheidet sich im Endeffekt für oder gegen ein Framework? Projektleiter? Entwickler? Die optimale Lösung wäre: beide miteinander. Hier bestehen allerdings oft sehr unterschiedliche Meinungen und Interessenlagen:

Risikominimierung gegen Flexibilität, kurzfristige Ergebnisse gegen langfristige Optionsvielfalt.

Nur wenn allen dies in genügender Tiefe bewusst ist, kann auch hier ein bewusster und richtiger Entschluss getroffen werden.

HTML5

Im Gegensatz zu den Frameworks (WebDynpro, SAPUI5) ist HTML5 lediglich eine Auszeichnungssprache und stellt die aktuelle Version (gemäß der W3C-Spezifikation) von HTML (Hypertext Markup Language) dar. HTML dient an sich zur Beschreibung und zum Verknüpfen/Verbinden von Webinhalten (wie z. B. Texten, Bildern, Videos) in elektronischer Dokumentenform. Hierzu zählen Webseiten und andere webbasierten Lösungen mit grafischer Oberfläche, die mithilfe von Webbrowsern oder sonstigen Browser-Engines dargestellt werden. Bei der Entwicklung einer HTML-Lösung wird in der Regel auf CSS (Cascading Style Sheets) für die Ausgestaltung von HTML-Dokumenten zurückgegriffen. Dies war auch schon in früheren HTML-Versionen der Fall und wird mit der aktuellen HTML-Version weiterverfolgt. HTML und CSS allein reichen aber noch nicht aus, um den Leistungsumfang von dynamischen und komplexeren HTML5-Lösungen abzubilden. Dazu wird die weitere Komponente JS (JavaScript) benötigt. JavaScript ist eine Skriptsprache, mit der Dynamik in HTML-Dokumenten realisiert wird. Da sich die maßgeblichen

Ziele einer Lösung aus den fachlichen Anforderungen ergeben, bietet sich HTML5 gerade bei Oberflächen mit einem hohen Maß an Flexibilität und individueller Gestaltung an.

Anwendungsszenarien

Pikon-typische Anwendungsszenarien sehen einen Datenaustausch mit einem Backend-System (z. B. SAP ERP) vor. In SAP kann hierzu eine HTML5-Lösung mithilfe einer Business Server Page (BSP) oder auch in Kombination mit einer WebDynpro-Anwendung umgesetzt werden. Solch eine Anwendung läuft nicht ausschließlich auf dem SAP-Server ab, sondern in erster Linie auf einem mobilen Client, und es muss ein Weg für die Kommunikation mit SAP gefunden werden. Den klassischen Weg im Webumfeld stellt dafür das Internet Communication Framework (ICF) im SAP NetWeaver dar. Hierbei kann ein Client z. B. mithilfe eines ICF-Service Daten zum SAP-Server senden und der SAP-Server entsprechende Daten zurückliefern. Diesen Weg erspart uns SAPUI5 mit dem SAP NetWeaver Gateway und oData Services.

Fazit

Zusammenfassend kann man sagen, dass eine reine HTML5-Entwicklung mit einer Mischung aus selbst erstellten und bewusst genutzten Fremdkomponenten für uns das Mittel der Wahl ist, auch wenn hier zur Umsetzung einiges an Wissen, Disziplin und Aufwand nötig ist. Aber gerade wenn ein ganzes Entwicklerteam an einer Sache arbeitet, ist es sinnvoll, an der Strukturierung sowie Verständlichkeit einer Lösung zu arbeiten, und der Aufwand wird mit jedem Änderungswunsch, der später folgt und umgesetzt wird, lohnender.

Wer also mit den Einschränkungen eines Frameworks nicht leben kann, dem steht mit HTML5 auch im SAP-Umfeld ein durchaus gangbarer Weg zur Verfügung.

Insgesamt lässt sich die Abwägung, ob und welches Framework eingesetzt werden soll, immer nur vor dem Hintergrund der konkreten Anforderung vor allem an die Individualität und die Anpassungsfähigkeit der Software entscheiden.



Elias Mistler zählt zu den jungen Talenten von Pikon. Als Berater im Bereich Business Intelligence bewegt er sich schwerpunktmäßig in den Themenbereichen Analytical Frontend Design, Business Objects sowie Abap- und HTML5-Entwicklung.



Dirk Hirschmann ist Senior-Entwickler bei Pikon für Business Integration. Seine Aufgaben umfassen die Projektleitung sowie Entwicklungstätigkeiten bei der Individualentwicklung mit Abap OO, WebDynpro und HTML5.

Bitte beachten Sie auch den Community-Info-Eintrag ab Seite 99

