

# Apex-Applikationen für Tablet-Devices

Kai Donato, MT AG

Die Grundlage für mobile Applikationen mit Oracle Application Express wurde mit der Version 4.2 implementiert und stellt schon jetzt die nötigen Werkzeuge zur Entwicklung von mobilen Applikationen bereit. Der Artikel zeigt, inwiefern sich diese Werkzeuge für die Entwicklung von Tablet-Applikationen nutzen lassen und beschreibt ein Beispiel aus der Praxis über den Umgang mit diesen Endgeräten, wenn eine Apex-Anwendung darauf laufen sollte.

Zum Erstellen von Apex-Applikationen gibt es seit der Apex-Version 4.2 ein gesondertes Theme, das wiederum für mobile Applikationen angepasste Elemente bereitstellt. Standard-Elemente wie Regions oder Forms-Elemente sind mit dem zugrunde liegenden jQuery-Mobile-Framework für die Darstellung auf mobilen Endgeräten optimiert. Mit der Auswahl des User Interface „jQuery Smartphone“ (Theme 50) wird die Apex-Entwicklungsumgebung insofern angepasst, als nicht für mobile Endgeräte kompatible „Page-Items“ versteckt werden und der Entwickler somit an der Verwendung inkompatibler Elemente gehindert wird.

Darüber hinaus sieht das User-Interface vor, eine mobile Applikation visuell so nah wie möglich an native Apps anzupassen, sodass der Anwender die App intuitiv mit der Hand bedienen und sämtliche Inhalte trotz der mangelnden Darstellungsfläche erfassen kann. Hauptbestandteil des genannten Theme ist zum einen der sogenannte „Header“, der sich ganz nach jQuery-Mobile-Methode aus- und einblenden lässt, und zum anderen der „Footer“, der sich ebenfalls vor dem Inhalt befindet und wahlweise aus- beziehungsweise einblendet werden kann. Das bereitgestellte Interface „jQuery Mobile Smartphone“ lässt sich bei dem Erstellen einer neuen Applikation auswählen (siehe Abbildung 1). Wie der Name dieses User Interface schon verrät, ist es zur Verwendung für Smartphones vorgesehen – doch wie verhält es sich gegenüber Tablets (siehe Abbildung 2)?

Zu erkennen ist das angepasste Layout, aber wie zu erwarten war haben sich die

Elemente der mobilen Website an die Breite des Tablets angepasst und gegenüber dem Smartphone eine ganze Menge Darstellungsfläche in Anspruch genommen. Die Anpassung an die Bildschirmgröße wirkt hierbei eher unvorteilhaft.

## Layout Grids

Für den Konferenzplaner war es aufgrund der oben genannten Problematik nötig, eine weitere Unterteilung der angezeigten Seite vorzunehmen. Für diesen Zweck ermöglicht Apex dem Entwickler, die Strukturierung seiner Applikation anhand von Regionen und des sogenannten „Grid“ vorzunehmen. Für jedes Element, das angelegt wird, kann bestimmt werden, ob es

eine neue Spalte, eine neue Reihe oder gar ein neues Grid innerhalb des vorhandenen Grid einnehmen soll. Dieses Grid beziehungsweise dessen Verhalten wird von dem zugrunde liegenden Theme vorgeschrieben und beeinflusst die Darstellung der Applikation abhängig von der Bildschirmgröße und der Größe der anzuzeigenden Elemente.

An dieser Stelle sei auf das Theme 25 (für das User-Interface „Desktop“) verwiesen, das verschiedene Display-Größen berücksichtigt und sich der Anzeigefläche anpasst (responsive Webdesign). Apex 4.2 greift bei der Erstellung einer mobilen Applikation auf die von jQuery Mobile bereitgestellten Layout-Grids zurück.

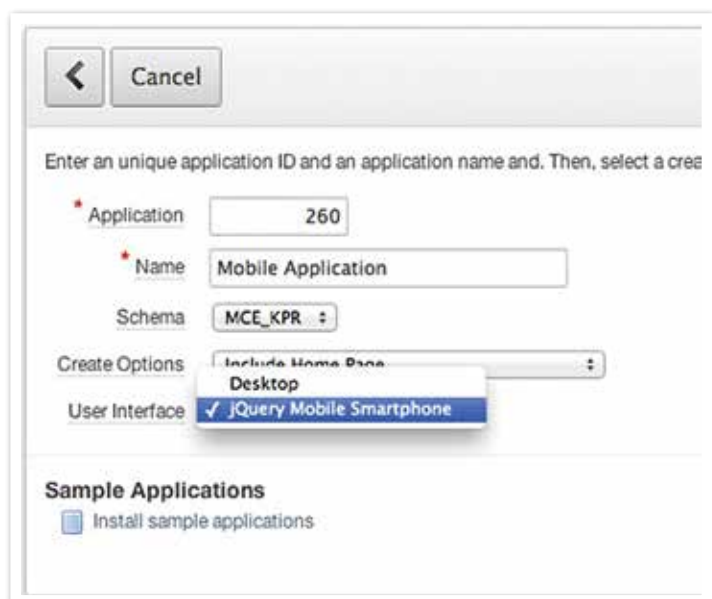


Abbildung 1: Auswahl des User Interface beim Erstellen einer neuen Applikation



Abbildung 2: Vergleich einer jQuery-Mobile-Applikation auf dem Apple iPad und dem iPhone

Die Definition dieses Grid übernimmt Apex für uns, erstellt das passende UI-Grid und fügt wahlweise Spalten hinzu (siehe Listing 1). Die Breite der einzelnen Spalten wird nach dem Standard von jQuery Mobile prozentual gleichmäßig verteilt. Die individuelle Anpassung des Grid kann nur begrenzt vorgenommen und muss je nach Anwendungsfall per CSS geändert werden. Sobald eine Reihe oder Spalte des Grid manuell festgelegt ist, werden die anderen Spalten/Reihen nicht nach der übrigen Breite skaliert, sondern nach wie vor auf die gesamte Höhe/Breite des Anzeigebereichs. Abhilfe schafft an dieser Stelle nur eine pixelgenaue Vorgabe für jeden Teil des Grid.

### Eine Apex-App für Tablets in der Praxis

Für einen Veranstalter sollte eine Applikation entwickelt werden, die den Besuchern einer Veranstaltung als Ersatz für ein Booklet dienen soll. Mit den Standardelementen konnte nach relativ kurzer Zeit ein Prototyp entwickelt werden, der sämtliche Informationen einer Papierbrochure auf den Tablets der Besucher anzeigt (siehe Abbildung 3).

Aufgrund der genannten Layout-Einschränkungen war es notwendig, einige Änderungen an den Standard-Apex-Elementen vorzunehmen, um die Anforderungen für die Tablet-App umsetzen zu können.

Der Prototyp besitzt im Grunde genommen drei Teile, die vertikal angeordnet sind (siehe Abbildung 4).

Hervorgehoben erkennt man den Header, der die Navigation und die Benachrichtigungen (News) beinhaltet, den Content-Bereich, der im mittleren Bereich

angesiedelt ist, und den Footer, der das Logo der MT AG beinhaltet. Im mittleren Teil befindet sich eine weitere Einteilung in drei Bereiche, die horizontal angeordnet sind (siehe Abbildung 5).

Ein sehr wichtiger Bereich des Prototyps ist das personenbezogene Tagesprogramm, das auf der linken Seite angezeigt

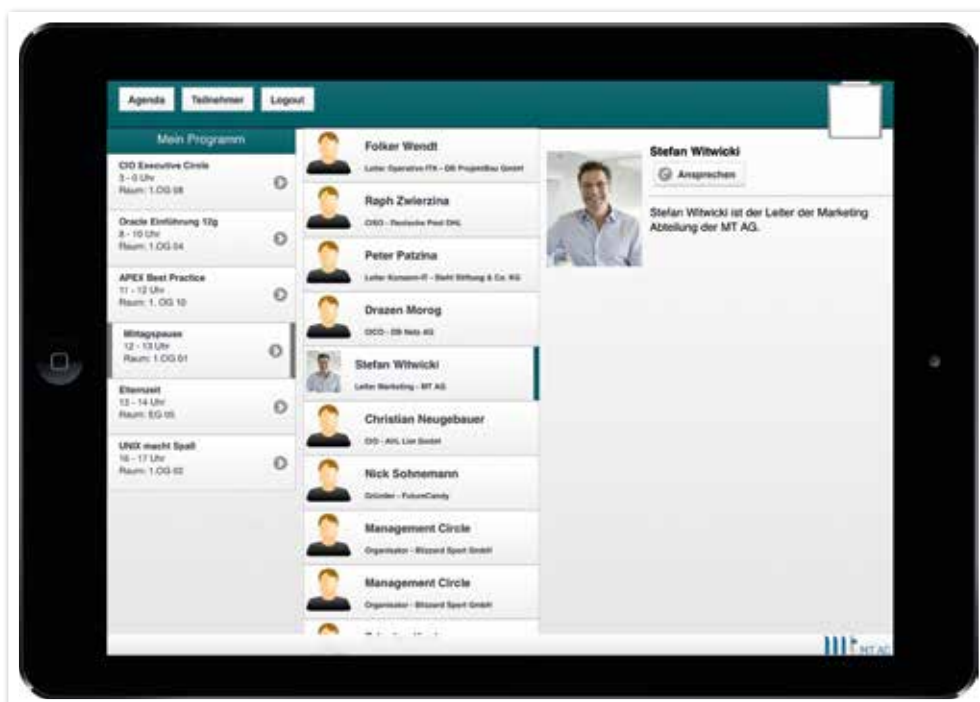


Abbildung 3: Layout des Prototyps für den Veranstaltungsorganisator

```
<div class="ui-grid-a">
  <div class="ui-block-a"> Block A</div>
  <div class="ui-block-b"> Block B</div>
</div>
```

Listing 1: Standard-Grid aus zwei Spalten ohne feste Breite

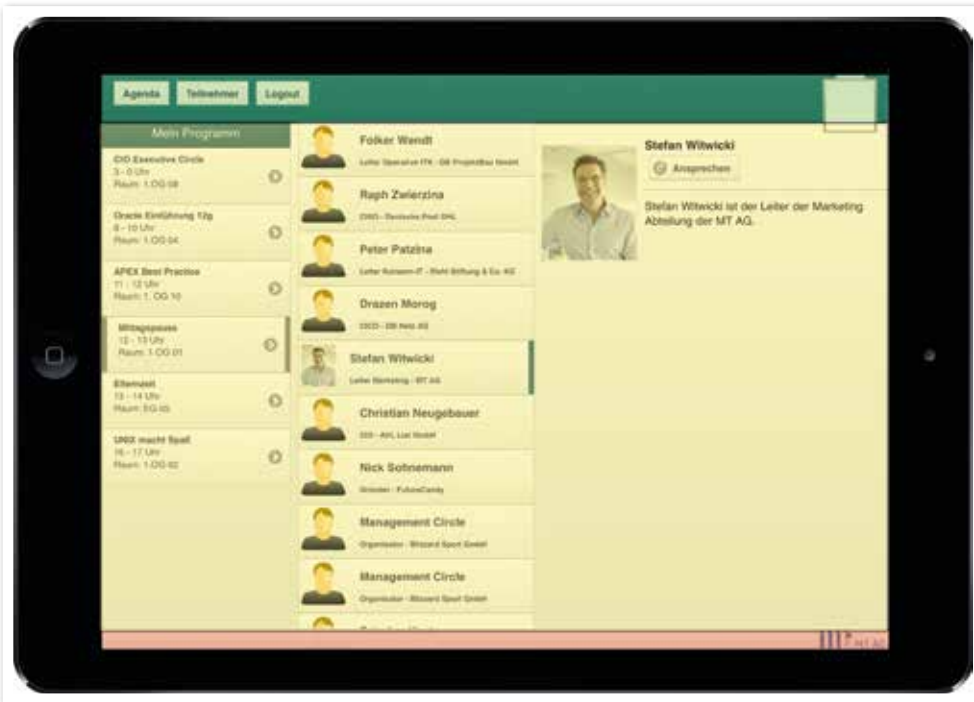


Abbildung 4: Einteilung der Applikation in drei vertikal angeordnete Bereiche

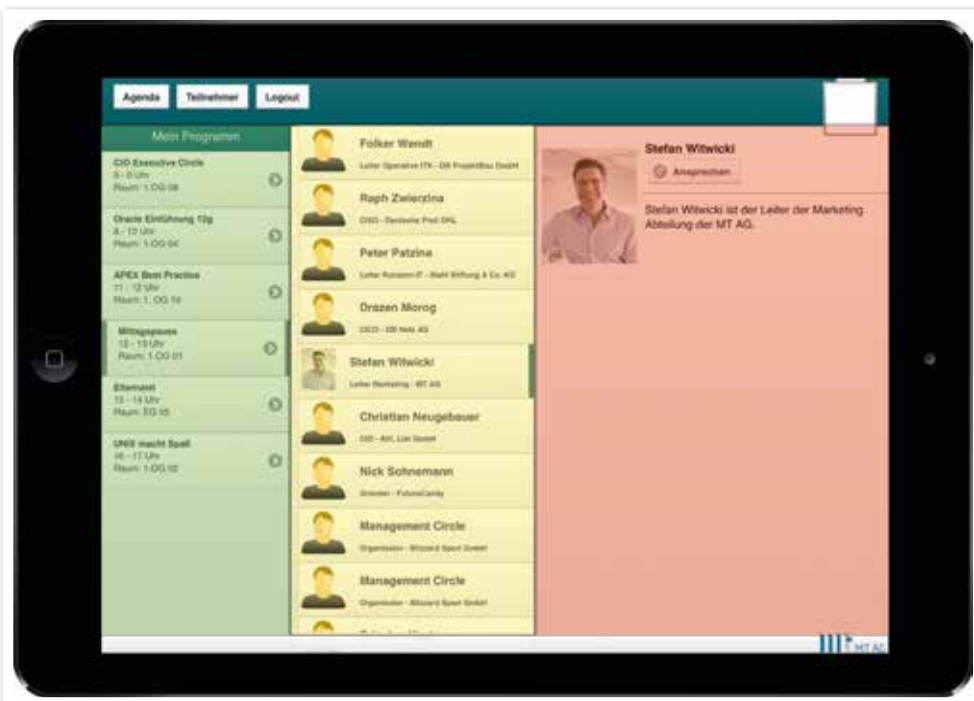


Abbildung 5: Drei-Spalten-Grid im mittleren Teil der Seite

ist. Änderungen im Tagesprogramm werden an dieser Stelle zeitnah angezeigt. Dies spricht deutlich für die Flexibilität, die eine Broschüre in digitaler Form mit sich bringt.

In *Abbildung 5* ist der zentrale Anzeigebereich hervorgehoben, der in dieser Applikation die Liste der Teilnehmer und die Agenda anzeigt. Am rechten Bildrand ist der Detailbereich, der bei ausgewähltem Datensatz ausgeklappt wird und detaillierte Informationen bereitstellt. Bezogen auf die *Abbildung 5* wurde bei diesem Prototyp darauf geachtet, die volle Breite des Tablets auszunutzen und eine unnötige Streckung des Inhalts zu vermeiden.

### Umgang mit Pop-ups auf Tablets

Gerade in einer mobilen Applikation ist es häufig sinnvoll, Pop-ups oder modale Dialoge zu verwenden. Auch für diesen Prototyp war es angedacht, Pop-ups zu nutzen, um beispielsweise Benachrichtigungen oder Sicherheitsabfragen hervorgehoben darzustellen. Leider ist es in Apex 4.2.4 noch nicht möglich, eine sinnvolle Implementation vorzunehmen, da Apex dazu neigt, ein Pop-up auf einer neuen Browserseite zu öffnen. Es besteht an dieser Stelle zwar ein minimaler Einfluss auf die Darstellung der geöffneten Seite, jedoch entspricht dies nicht mehr einem Overlay, das dem Benutzer temporär und ohne die aktuelle Seite zu verlassen, angezeigt wird. Übrigens soll Apex 5.0 sogenannte „Modale Dialoge“ nativ unterstützen und damit diese Maßnahme überflüssig machen.

### Scrolling auf iOS-Geräten

Da die Applikation von ihrem Aufbau und der Funktion her an eine native App angelehnt ist, war der Autor mit einer Problematik konfrontiert, die die Bedienung des Konferenzplaners beeinträchtigt hat. Apple hat in seiner nativen Browser-Engine ein Scrolling eingebaut, um das Scrollen über die Grenzen der eigentlichen Website hinaus zu ermöglichen. Wenn man mit dem iOS-Browser zum Ende einer Website scrollt, führt ein weiteres „Wischen“ in die gleiche Richtung dazu, dass der Bereich, in dem die eigentliche Website angezeigt wird, zur Seite geschoben wird und man quasi hinter die Web-

site sehen kann. Dies führt dazu, dass die Bedienung von Web-Applikationen mit mehreren scrollbaren Bereichen unangenehm beeinflusst wird.

In diesem mobilen Prototyp existiert genau dieses Problem und beeinträchtigt die Usability deutlich. Recherchen haben ergeben, dass man die Möglichkeit hat, das sogenannte „touchOverflow“-Verhalten von Elementen zu verhindern. Dieses vollständig zu deaktivieren, hat jedoch zur Folge, dass jegliche Wischgesten in der Applikation unterbunden sind und somit auch keinerlei Scrolling mehr möglich ist.

Es gibt Lösungsansätze, die sich mit dieser Problematik befassen, jedoch sind diese nicht ohne weitere Probleme zu implementieren. Im Grunde funktionieren diese Lösungen, indem sie das Scrollen unterbinden, sobald man das untere beziehungsweise obere Ende des scrollbaren Elements erreicht hat. Da man bei einer mobilen Applikation aber in der Lage ist, auf jeder beliebigen Stelle zu scrollen, ist es notwendig, jedes dargestellte Element einer mobilen Applikation mit gesonderten Regeln (CSS-Klassen) zu versehen. Wird dies manuell vorgenommen, erschwert es die Erweiterung der Applikation und wirkt sich ebenfalls auf die Kompatibilität zu neueren Apex-Versionen aus.

### „Look & Feel“ mittels Themeroller anpassen

Um die jQuery-Mobile-Applikation nach seinen eigenen Vorstellungen zu gestalten, gibt es ein sehr hilfreiches Tool, das die jQuery-Entwickler ebenfalls kostenlos zur Verfügung gestellt haben. Der Themeroller, den man bequem von der Herstellerseite von jQuery-Mobile erreichen kann, ermöglicht es, für sämtliche Versionen von jQuery-Mobile Themes zu erstellen (siehe Abbildung 6).

Das erstellte Theme lässt sich nach der Erstellung benennen und herunterladen. Eine Einbindung in den Apex-Workspace ist über die „Shared Components“ möglich, nachdem man die sich im Stylesheet befindenden Icon-Verknüpfungen auf die der Apex-Umgebung geändert hat.

Wer seine Mobile-App mit weiteren JavaScript-Funktionalitäten erweitern möch-

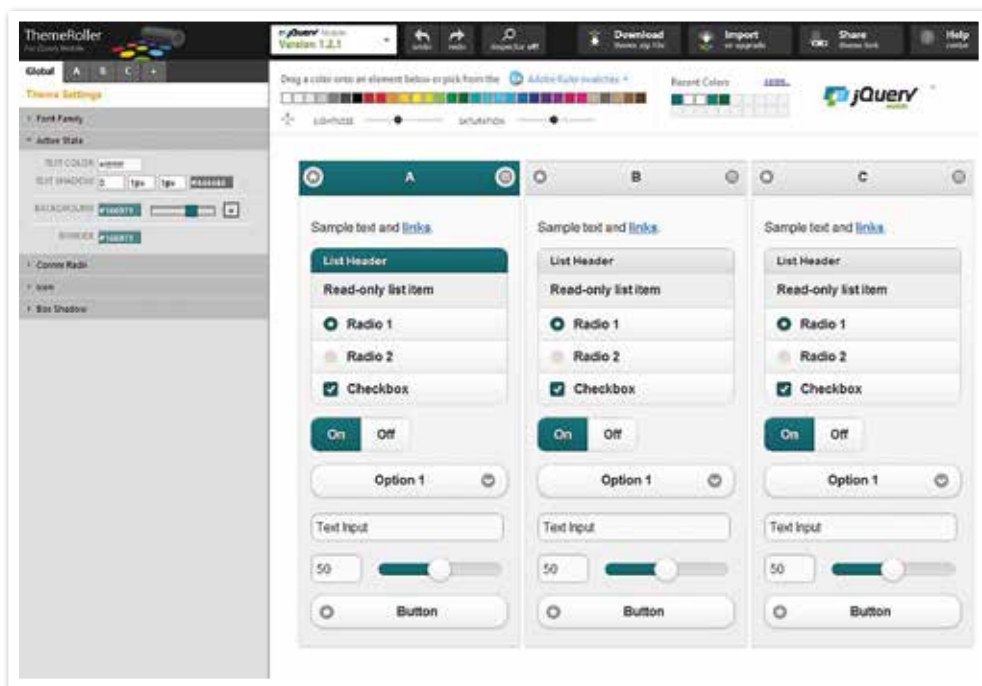


Abbildung 6: Der Themeroller zur Erstellung eigener Stylesheets

te, kann diese ebenfalls in den „Shared Components“ in Apex hinterlegen. Möchte man also JavaScript-Bibliotheken integrieren, kann man diese über das Page-Template in den Kopf der HTML-Seite einbinden und wie gewohnt in der Applikation verwenden. Der Prototyp des Konferenzplaners konnte anhand eines erstellten Theme an die Farbgebung des Kunden angepasst werden.

### Fazit

Die Entwicklung des Konferenzplaners im Apex-Umfeld war bereits mit Apex 4.2.4 komfortabel. Wer in seiner Applikation auf Daten aus der Oracle-Datenbank setzt, ist mit den Bordmitteln von Apex gut bedient und in der Lage, selbst in kurzer Zeit gute Ergebnisse zu erzielen. Mit Kenntnissen im Bereich „CSS und jQuery“ kann man über die Grenzen hinaus Features einbinden und seine Applikationen mit erweiterten Funktionalitäten beleben.

Sowohl die neue jQuery-Version als auch die Unterstützung von modalen Dialogen und Pop-ups ist für die kommende Apex-Version 5.0 angekündigt. Darüber hinaus soll mit der Version 5.1 ein weiteres Mobile Theme gezielt für Tablet-Applikationen kommen, was den Aufwand für das Customizing weiter reduzieren dürfte.

### Weiterführende Links

1. Apex Showcase Portal: <https://apex.mt-ag.com>
2. <http://www.oracle-and-apex.com/improve-the-apex-mobile-theme-part-4/>
3. <http://themeroller.jquerymobile.com>
4. <http://www.kylejlarson.com/blog/2011/1/1/fixed-elements-and-scrolling-divs-in-ios-5>
5. Apex 5.0 Early Adopter 1: <https://apex.oracle.com/i/index.html>



Kai Donato  
kai.donato@mt-ag.com