



TechNotes

Funktionalität des Expanders erweitern



Einleitung

In einer TD-Mobile-Anwendung soll ein *Listen-Control* – eine Liste – mit relativ vielen Einträgen angezeigt werden. Da die angezeigte Liste in den meisten Fällen den gesamten Bildschirm – vermutlich muss der Anwender sogar scrollen – in Anspruch nehmen würde, soll die Liste zunächst hinter einem *Expander* verborgen werden. Auf diese Art und Weise könnte die Liste zunächst verborgen bleiben und würde erst sichtbar, wenn der Anwender den *Expander* antippt.

Ein Problem bleibt allerdings: standardmäßig wird beim *Expander* nur das Event **Create** unterstützt. Es würde bedeuten, dass die Liste beim Seitenaufbau mit umfangreichen Daten erzeugt würde. Das hätte zur Folge, dass der gesamte Seitenaufbau verzögert wird, obwohl der Anwender möglicherweise die Liste gar nicht anschauen will. Es sollte möglich sein, die Liste erst aufzubauen und anzuzeigen, wenn der Anwender den *Expander* antippt.

Lösungsansatz

Da es sich beim *Expander-Control* in TD Mobile um das reguläre *JQuery Mobile Control* handelt, bietet auch das *TD-Mobile-Control* die Möglichkeit, auf die (jQuery mobile) Ereignisse **expand** und **collapse** zu reagieren. Allerdings stellt sich die Frage, ob eine derartige Erweiterung in TD Mobile möglich ist. Die kurze Antwort lautet: es ist möglich. Die genaue Vorgehensweise wird im nächsten Abschnitt erläutert.

Lösung

Das *Listen-Control* wird über einen (komplexen) Bindungsarray und eine (ebenfalls komplexe) Bindung gefüllt, wobei die Bindungen über den Rückgabewert der entsprechenden Operation mit Werten versorgt werden. Diese Funktionalität bleibt erhalten. Entfernt wird lediglich der Aufruf dieser Operation **Invoke** beim **Create**-Ereignis.

Im nächsten Schritt muss der zusätzliche Event-Händler für das **expand**-Ereignis des Expanders implementiert und die Funktionalität, die beim Eintreten dieses Ereignisses ausgeführt werden soll, hinterlegt werden. Da beim Antippen des Expanders das *Listen-Control* mit den Daten aus der Datenbank gefüllt werden soll, muss als Aktion die im Quellcode hinterlegte Operation aufgerufen werden. Dieser Job wird durch Verwendung der TD Mobile-API im entsprechenden JavaScript erledigt.

```
Td.Data.Operation.get('name_der_operation').invoke();
```

Die Bezeichnung `name_der_operation` muss beim Nachvollziehen dieses Beispiels noch an die tatsächliche Operation auf der entsprechenden Seite angepasst werden.

Die Hinzufügung des Event-Handlers geschieht zum Zeitpunkt des Seitenaufbaus. Beim **Create**-Ereignis der Seite wird das JavaScript aufgerufen. Die Ausführung dieses Scripts bewirkt, dass der zusätzliche Event-Handler für das `expand`-Ereignis und die dann auszuführende Funktionalität zur Seite hinzugefügt wird. Wenn der Anwender jetzt die Seite aufruft, wird die JavaScript-Funktionalität mit dem **expand**-Ereignis erweitert: die Bindings, die die im *Listen-Control* zu visualisierenden Daten enthalten, werden durch den Aufruf der Operation mit Daten versorgt und das *Listen-Control* wird im Expander angezeigt.

Die gesamte JavaScript-Funktion hat folgenden Inhalt:

```
$("#name_des_expander_controls").collapsible ({  
  expand: function(event, ui) {  
    Td.Data.Operation.get('name_der_operation').invoke();  
  }  
});
```

In diesem Beispiel wird der Name des Expander-Controls im Quellcode referenziert. Standardmäßig legt TD Mobile die Controls mit dem Namen `exp` und einer laufenden Nummerierung an, z.B. `exp1`.

Der Aufruf dieser Funktion erfolgt beim Seitenaufbau, also beim **Create**-Ereignis auf der entsprechenden Seite. Hierdurch wird die zusätzliche Expander-Funktionalität der Seite hinzugefügt.

On Create

...

JavaScript `name_der_JavaScript-Funktion`

Das Hinzufügen der zusätzlichen Funktionalität für den Expander erfolgt beim **Create**-Ereignis der Seite.

Fazit

Es ist mit TD Mobile relativ einfach möglich, zusätzliche Funktionalität, die bereits in JQuery Mobile hinterlegt ist, aber nicht an TD Mobile durchgereicht wurde, hinzuzufügen. In diesem Beispiel wurde die Expander-Funktionalität so erweitert, dass auch auf das Ereignis des Antippens des Controls von Programmierer definierte Aktionen ausgeführt werden. In diesem Beispiel wurde der Event-Handler für das Ereignis „expandieren“ des Expanders und die dann auszuführende Operation hinzugefügt.