

Telefonbuch – Eine Fingerübung

Realisiert wird hier ein browserbasiertes Telefonbuch. Als Programmiersprache wird Angular-Javascript benutzt. Der Nutzer sieht alle Einträge seines Telefonbuchs in einer Tabelle getrennt nach Name und Telefonnummer. Er hat die Möglichkeit nach einem Eintrag zu suchen, neue Einträge hinzuzufügen und diese zu editieren oder zu löschen. Beim Schließen soll die Liste erhalten bleiben.

1. GUI / Index.html

Das GUI wurde mit HTML und CSS realisiert. Ich habe mich beim Design an der Vorlage von »TodoMVC.com« orientiert.



Name	Telefonnummer
André	05507401436
Peter	0551501555
Frauke	05518015

Die Tabelle mit den Einträgen wird über die Eingabe in der Textbox gefiltert. Realisiert wird dies durch die *ng-model* direktive »Suchfilter« der Inputbox, welche in der Tabelle als Filter genutzt wird.

Die Zeilen der Tabelle werden mit der *ng-repeat* Direktive aus dem Objektarray *Phonebook.Entries* gefüllt.

Der Inhalt der Zeilen wird durch drei Templates vorgegeben, die der Zeilendeklaration durch die Funktion *getTemplate()* übergeben werden. Bei den drei Templates handelt es sich einmal um die normale Tabellenansicht (*display*), bei der die Daten als Text angezeigt werden. Macht der Nutzer einen Doppelklick auf den Namen oder die Telefonnummer, so wird das jeweilige andere Template geholt, und der Name (bzw. die Telefonnummer) wird in einer Textbox angezeigt, die vom Nutzer bearbeitet werden kann.

Bewegt der Nutzer den Mauszeiger über einer Zeile wird rechts neben der Zeile ein Tooltip angezeigt, der Informationen über das Erstellungsdatum, des Eintrags gibt. Außerdem wird rechts in der Zeile ein Button zum Löschen des Eintrags angezeigt.

2. Controller

Gesteuert wird die App vom Controller-Objekt »*phoneCtrl*«. Beim Starten der App wird versucht das Objekt-Array »*phonebook*« aus dem lokalen Speicher zu lesen. Hierbei handelt es sich um die Liste unserer Telefonbucheinträge. Wird diese nicht gefunden, wird die Liste mit drei Dummy-Einträgen vorgefüllt. Wichtig ist, dass die Liste vom JSON-Dateiformat geparsed werden muss. Normalerweise werden alle Daten im lokalen Speicher als String abgelegt. Auf diese Weise erkennt die App aber nicht, dass es sich bei den Daten um ein Array, bzw. ein Objekt handeln kann. Bei jeder Änderung der Daten (löschen, editieren, hinzufügen neuer Daten) wird die Tabelle im JSON-Dateiformat in den lokalen Speicher abgelegt.

Editieren eines Eintrages:

Bei einem Doppelklick auf einen Eintrag wird die Funktion *editName*, bzw. *editNumber* aufgerufen. Hier wird das ausgewählte Objekt in das Array *selected* kopiert. Außerdem wird die boolesche Variable *showEditName*, bzw. *showEditNumber* auf true gesetzt.

In der Funktion *getTemplate* wird geprüft, ob die jeweilige Zeile im *selected* Array liegt. Wenn das so ist, wird das Zeilentemplate geändert, damit der Nutzer eine Eingabe machen kann. Bestätigt der Nutzer seine Eingabe mit Enter, wird die geänderte Zeile in der Funktion *Save* von *selected* wieder an die richtige Stelle im *Entries*-Array kopiert. Danach wird die *selected* Zeile gelöscht und das Template wieder auf normale Ansicht zurückgesetzt.