ICT 581 014: Assignment 3: Create your first mobile-fit website
Student: Marcin M. Malec

Description Section

"What do you want to make?"

Build a mobile friendly "To Do" list. The user should be able to create list items, delete list items, and edit list items.

Users will be able to use a form at the top of the application to create new items. The form will only accept text input which will act as the title of the "To Do" item. Submission will be conducted via a submit event listener rather than a submit button. New items will be appended to the bottom of the list located below the form field. Each item will also receive a child element that will act as the close button that will delete the item from the list when pressed. When the user clicks on the list item, they will be presented with a dialog box that will ask for new input to replace the list item.

Each action (create, delete, and edit list items) trigger a store function where the list will be stored in local storage that will be persistent between sessions. At every page load the page is populated with the list currently stored in local storage.

"What technology will you use?"

The project will use the following languages:

- HTML5
- CSS3
- Javascript 9

To extend Javascript capability the following jQuery libraries are utilized:

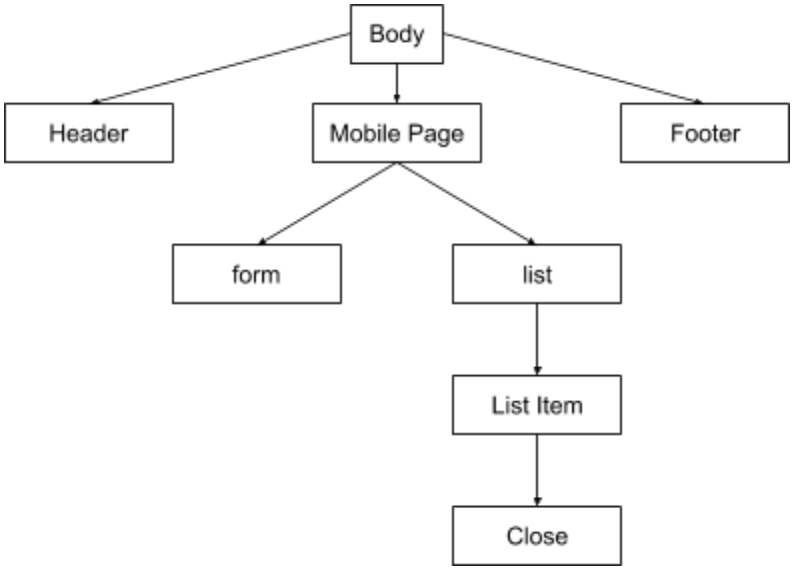
- jQuery 2.1.0
- jQuery UI 1.12.1
- jQuery Mobile 1.4.5

The project will be tested on Google Chrome. Google Chrome Dev will not be utilized as local storage does not function with it.

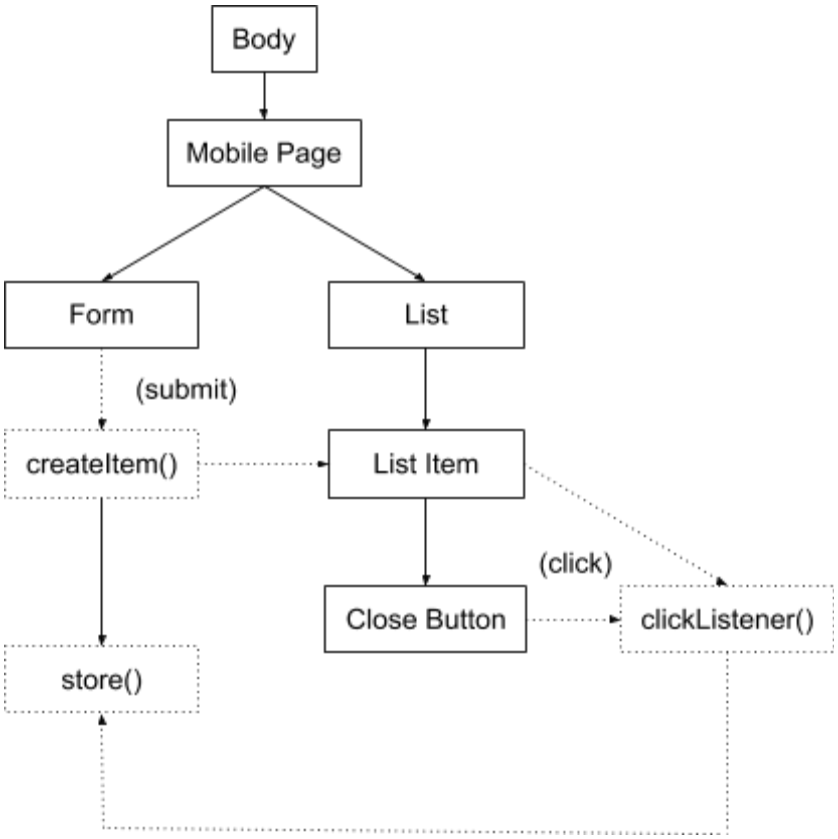
Feature Section

HTML features ():

ICT 581 014: Assignment 3: Create your first mobile-fit website
Student: Marcin M. Malec



Javascript Features ():



Prototype Section

Beginning with an empty list where only a header, footer, the form, and an empty list are present in the DOM. The first script to run is the $(document).ready(function(){}) function which sets up variables for the list element, form element, and input element (a

child of the form element). Two event listeners are established. The first listener is attached to the form element based on the “submit” event. The second listener is attached to the list element based on the “click” event. The purpose of these listeners will be explained later. Finally, the getValues(var list) function is called extracting the list currently stored in local storage. If nothing is stored, then a default list is printed. At this time the default is just a blank list.

Text is inputted into the form field and on submit the createItem(var list, var item) function is called. This function uses the text from the form (via the item variable) and appends a new list item into the list (that is stored in the list variable). After this the store(var list) function is called which adds the current list into local storage. This process creates a list item element that has two parts: the text from the item variable and a child span element that will act as the close button. Clicking on the parent element or the child element will produce different results.

When either element is clicked on, the clickListener(event e, var list) function is called. This function encapsulated both the edit and delete functions. If the parent element (list item) is clicked on, new event listeners are created for the edit list item dialog box which triggers on submit and replaces the text in the list item element. If the child element (close button) is clicked on the list item (with child) are removed from the list. In both cases store(var list) is called updating local storage.