Kendo UI Builder Grid Test Plan

Best way to store the Test Plans/Scenarios representing atomic business cases (normally containing several test cases per functionality) is always by using a web based system such as e.g **TestLink** ( <http://testlink.org/>. More visual representation here: <https://www.google.com/search?q=testlink&source=lnms&tbm=isch&sa=X&ved=0ahUKEwj7kYnQkcvcAhVFa1AKHWZ6DbgQ_AUICigB&biw=1366&bih=654> ).

In this manner the test cases can be seen by the Quality Assurance guys (who created them), the BA/PO (who is best to make a review after the QAs create them), the developers, guys from other teams which use the same product – such as support engineers (to use them as a regression pack) and if needed – the management.

The test cases should be based on the detailed Acceptance Criteria created by an experienced Bussiness Analyst/Product Owner (depending on the company structure).

Acceptance criteria of each and every task should be strictly followed by both developers and testers.

Manual test plan can be created even before the feature is ready, working on it in parallel with the developers, in order to optimize the process time.

Manual testing should be performed after that, in which process we can enrich/alter/add our initial vision of the test plan, related to a particular feature. We can also create the test cases entirely in parallel with the manual testing if we have such opportunity.

Automated tests can be created as a final step to this process. We could have several test suites, such as a regression and smoke pack. Regression pack includes all of the automated tests covering the main functionalities on a varieties of ways, while the smoke tests can be following just the positive/straight/happy path.

QAs could go one sprint after the developers in order to be sure that no functionality will be changed and their test plans and automated tests will not be reworked, which will lose time. The drawback of this is that in this manner the QAs will miss the opportunity to give proactive ideas (on the planning of the new features) to the whole team how a particular functionality is best to be created from a user perspective (mainly focusing on if the functionality is rich enough for the particular feature, usability, performance)

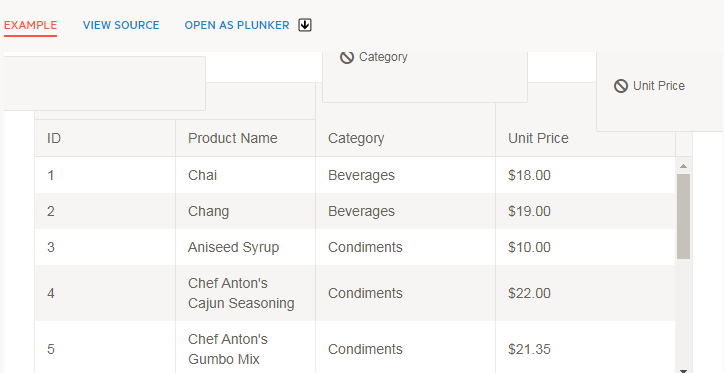
**Grid Reordering Test Plan**

Following the angular implementation of the Grid, here is the test plan of the reordering feature:

<https://www.telerik.com/kendo-angular-ui/components/grid/columns/reordering/>

Using the latest version of Chrome, we got the following issue below. Using Mozilla we have no issues on first glance over the reordering. Seems like the reordering is tested more over the Mozilla browser.

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case Number | Action | Expected Result | Actual Result |
| 1 | Switch the places of the Product Info column with the Category one | All of the sub-columns of the Product Info one, such as ID and Product Name to be moved together with the Product Info column staying at the same structure and order. Product Info column order should be switched from 0 to 1 and the Category column order should be switched from 1 to 0. | Matches |
| 2 | Switch the places of the Product Info column with the Unit Price one | Same as in Step 1, but for the Unit Price column, just to check the reording with it also works as expected. | Matches |
| 3. | Switch the places of the Category column with the Unit Price one | Verify the both columns order have changed in the grid properly. | Matches |
| 4. | Switch the places of the ID and the Product Name columns | ID and Product Name stay below the Product Info column and get properly reordered switching places. | Matches |
| 5. | Try switching the places of the ID and Product Name columns with any other column | Switching/reordering not possible since both columns are siblings of the Product Info column and therefore are tied to it in a downstream hierarchy. | Matches |
| 6. | Move any of the three main column headers out of the view area | The column headers should go back to their place (with the “forbidden sign” over them) or they shouldn’t be allowed to get out of the view area. | Issues raised – {issue\_number\_jira\_link}.  Take a look at the screenshot below. |



Actual behaviour: All of the three column headers stay on place when the mouse left button is released, after dragged out of the visible view area. A corresponding usability issue needs to be raised accordingly.

**Grid Resizing Test Plan**

Following the angular implementation of the Grid, here is the test plan of the resizing feature:

<https://www.telerik.com/kendo-angular-ui/components/grid/columns/resizing/>

Using the latest version of Chrome, we got many issues listed below, some of which are leading to inadequate behavior of the grid. Using Mozilla we have just one issue on first glance. Seems like the reordering is tested over the Mozilla browser, but looks like no or little testing in regards of the resizing has been carried over the Chrome one.

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case Number | Action | Expected Result | Actual Result |
| 1 | Mozilla, Chrome: Try resizing to the left and right (reduce and enhance column width) of any of the columns inside the visible part of the view area | After releasing the left mouse button, the columns width to stay fixed and no mouse over any of the column boundaries to resize the column, acting like a mouse hover | Matches |
| 2 | Chrome: Resize the Category column, getting out of the visible view area on the right. Mouse over the right boundary of the Category column. | To be able again to resize the Category column to the left, by dragging its right boundary (the left one of the Unit Price one). | Issues raised – {issue\_number\_jira\_link}.  Mouse over the left boundary of the Unit Price column header makes the whole grid to fit the visible area, reducing automatically the Category column size accordingly.  Not possible to click the boundary in question in order to perform a normal drag again.  Same behavior observed with the ID column, Product Info column and when getting the left boundary of the Category column out of the visible view area on the right.  Unfortunately further mouse moves over the grid lead to illogical columns width change. |
| 3 | Mozilla: Resize the Category column, getting out of the visible view area on the right. Scroll to the left boundary of the Unit Price column and start dragging it to the left. | The width of the Unit Price column to be made bigger and its left boundary to move left accordingly | Issues raised – {issue\_number\_jira\_link}.  The width of the Category column reduces and the mouse becomes not over the left Unit Price column header boundary any more (but over the Category column). Unit Price column width remains the same up until a certain position when the Category column stops reducing its size and the Unit Price starts to grow in width. |
| 4 | Double click any column span resizer element | The column content fits into the width – the widest string is taken into account | Matches on both Chrome and Mozilla. |

Note that if we have the issues raised in JIRA, no need of the bug description to appear at the right most column of the table. Description now appears just to illustrate the further challenges the software developer guys might have, in order to maintain the software quality at a high level.