Autogrow For Catalog Variables

In the Service Portal Service Catalog(ue), multi-line text variables do not auto expand in the same way large string fields do on the rest of the platform. This update set fixes this by adding a jQuery library **Non-Sucking Autogrow 1.1.1**, and applying it to textarea elements.

This update set does not modify any OOTB records, but does create a relationship to the OOTB Service Catalogue widget.

Contact: Dan Conroy dan.conroy@servicenow.com

Components

- UI Script: jquery.ns-autogrow.min.js - this library does all the work.
- UI Script: sc*cat*item.textarea.autogrow.js - this script applies the above library to textarea elements on the catalog(ue) order page
- Widget Dependency: "Autogrow for SC Catalog Item" - packages the two above UI scripts into a widget dependency.
- M2M Widget Dependency to Widget: Above Widget Dependency added to OOTB "SC Catalog Item" widget.

Instructions

If you are using the OOTB Service Catalog(ue) item form widget ("SC Catalog Item") then this should work right after importing the update set. Any multi-line variables on your service portal should now expand vertically as you type.

If you are using a custom widget to render your catalog(ue) item forms, you will need to add the "Autogrow for SC Catalog Item" dependency to your widget.

If your custom widget is not inside an HTML element with an ID of "sc*cat*item" then you will need to modify the "sc*cat*item.textarea.autogrow.js" UI script. On line 7 the jQuery selector uses "#sc*cat*item textarea" - simply update this so that it selects the input fields you like.

> **Note:** If you want this library to apply to other input fields, and apply vertical as well as (or instead of) horizontal autogrowing, then you should use the jquery.ns-autogrow.min.js library only, and look at the [github repository](#) for instructions on how to apply the autogrow library and how to give it options. Note that if you do this, you need to make sure that the autogrow is applied after all the initial AngularJS has loaded and your DOM is ready. For an example of how this can be done, see the

> sc*cat*item.textarea.autogrow.js UI script. There is probably a better way to do this than using setTimeouts... but we will get to that in v2 of this update set :-)

jQuery Library Details

https://github.com/ro31337/jquery.ns-autogrow

- **Library**: Non-Sucking Autogrow 1.1.1
- **License**: MIT
- **Author**: Roman Pushkin

The MIT License (MIT)

The MIT License (MIT)

Copyright (c) 2015 Roman Pushkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ON INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.