

NBS ITEMLISTS

INTRODUCTION

In NBS, itemlists are lists of items selected by the user and saved as favorites or a wish list.

These existed in previous versions of NBS as cookies, in v3.6.7 the itemlists are saved against the user data in the database and can be retried at any point in the future.

Because the data is saved with the user data, itemlists only work if the user is logged in. This makes implementation a little more complex.

SETUP

A number of templates need to include the itemlists JS and html code to make itemlists work correctly, we'll try and outline how to implement a itemlists version which logs the client in when a favorite button is clicked by the user. We will also show how to use multiple lists. Other methods can be employed to get itemlists to work, but we'll cover all the basic principles in this document.

THE MODULE

To display the clients itemlists you can use the "[NBS_ItemList](#)" module. This module is designed to display all the users itemlists and can be used to add to basket.

THE TEMPLATES

The following template include code for itemlists. These template examples are taken from the "ClassicRazor" Theme, but if you have built your own theme you'll need to select which templates you need to change.

NBS_AJAXPRODUCTDISPLAYLIST_HEAD.CSHTML

NBS_PRODUCTDISPLAYLIST_HEAD.CSHTML

The examples below are taken from the "NBS_AjaxProductDisplayList_head.cshtml" template, but can also be applied to the "NBS_ProductDisplayList_head.cshtml" template.

```
<!-- Colorbox Lightbox -->
<script type="text/javascript" src="/DesktopModules/NBright/NBrightBuy/Themes/ClassicRazor/js/jquery.c
min.js"></script>

<link href="/DesktopModules/NBright/NBrightBuy/Themes/ClassicRazor/css/colorbox.css" rel="stylesheet">
```

Include a color box popup window in this template to display a list request to the user.

Include the JS to action the itemlists events. (Code not related to itemlists has not been displayed)

```
function AjaxView_GetList_nbxgetCompleted(e) {
    if ($('#Model.GetUrlParam("shopitemid")' != '')) {
        // we have shopitemid in url, so display itemlist popup
        $('#shopitemid').val($('#Model.GetUrlParam("shopitemid")');
        $(".shoppinglistadd[itemid=$('#Model.GetUrlParam("shopitemid")']").click();
    }
}
```

Show popup if returning from the login prompt, the "shopitemid" param in the url will trigger this. *In these examples the JS code is included in the ajax return. This is so it processors when the page has the correct html already loaded.*

```
// Shopping List Popup
$('.shoppinglistadd').unbind("click");
$('.shoppinglistadd').click(function() {
    if (@UserController.Instance.GetCurrentUserInfo().UserID <= 0) {
        window.location.replace('/default.aspx?tabid=@(StoreSettings.Current.Get("logintab"))&returnurl=' +
        location.protocol + '//' + location.host + location.pathname + '?shopitemid=' +
        $(this).attr('itemid'));
    } else {
        $('#shopitemid').val($(this).attr('itemid'));
        $(".shoppinglistadd").colorbox({ inline: true, width: "20%", fixed: true });
    }
});
```

Add the code required to display a popup when the add to wish list button is clicked, this also will redirect to the login page if the user is not connected.

```

// Wish List events
$('.wishlistadd').unbind("click");
$('.wishlistadd').click(function () {
    $('#shoplistname').val($('#shoplistselect').val());
    nbxget('itemlist_add', '#nbs_productajaxview'); //apply serverside
    $('#shoppinglistpopup').colorbox.close();
});

$('.wishlistremove').unbind("click");
$('.wishlistremove').click(function () {
    var name = $(this).attr('listname');
    nbxget('itemlist_remove', '.entryid' + $(this).attr('itemid'));
});

$('.wishlistremoveall').unbind("click");
$('.wishlistremoveall').click(function () {
    nbxget('itemlist_delete', '.entryid' + $(this).attr('itemid'));
});

```

Included the button events to add the products to the itemlist

```
$('.shoplistselect').unbind("change");
$('.shoplistselect').change(function () {
    if ($(this).val() == '-1') {
        $('.newlistdiv').show();
        $('.wishlistadd').hide();
        $('.shoplistselect').hide();
    } else {
        $('#shoplistname').val($(this).val());
        $('.newlistdiv').hide();
        $('.wishlistadd').show();
        $('.shoplistselect').show();
    }
});

$('.cancelnewlist').unbind("click");
$('.cancelnewlist').click(function () {
    $('.newlistdiv').hide();
    $('.wishlistadd').show();
    $('.shoplistselect').show();
    $('.shoplistselect>option:eq(0)').attr('selected', true);
});

$('.addnewlist').unbind("click");
$('.addnewlist').click(function () {
    var newlisttext = $('.newlisttext').val();
    //$('.shoplistselect').append('<option value="' + newlisttext + '" selected="selected">' +
newlisttext + '</option>');
    $('.shoplistselect').append( new Option(newlisttext,newlisttext,false,true) );
    $('.newlistdiv').hide();
    $('.wishlistadd').show();
    $('.shoplistselect').show();
});
```

Include the button events for the list actions

```
function loadItemListPopup() {
    $('.processing').show();
    nbxget('itemlist_getpopup', '#nbs_productajaxview', '#shoppinglistpopup');
}
```

The popup html code is loaded via ajax, so we need to include a ajax call to get the html.
This is called from the document ready section of the head template.

```
<!-- Shopping List Popup -->
<div style="display:none">
  <div id="shoppinglistpopup" class="shoppinglistpopup">
  </div>
</div>
```

The above ajax will load html into the section of the page that displays the popup, we need to include the return point into the page via the head template (or another template if preferable)

```
<div id="nbs_productajaxview" style="display: none;">
  <input id="userid" type="hidden" value="@UserController.Instance.GetCurrentUserInfo().UserID"/>
  <input id="uilang" type="hidden" value="@Model.Lang"/>
  <input id="themefolder" type="hidden" value="@Model.GetSetting("themefolder")" />
  <input id="shopitemid" type="hidden" value="" />
  <input id="shoplistname" type="hidden" value="" />
</div>
```

To make the ajax code for the itemlists work, we need to pass parameters to the server side code. To do this we use the "nbs_productajaxview" section of the page, which is defined in the head template. This section is also used to pass other parameters for other operations. Below we have just listed the ones required by the itemlists.

NBS_PRODUCTDISPLAYLISTBODY.CSHTML
NBS_PRODUCTDISPLAYDETAIL.CSHTML

```
<!-- Add to favorites buttons -->
@if (Model.GetSettingBool("enablefavorites"))
{
  <span>

  <!-- Add to Shopping List button -->
```

```

        <a href="#shoppinglistpopup" class="shoppinglistadd" itemid="@{info.ItemID}" ><i class="fa fa-s
        <a class="wishlistremove" itemid="@{info.ItemID}"
style="display:none;">@ResourceKey("ProductView.wishlistremoveicon-lg")</a>

        </span>
    }

```

These templates need to have the add to wish list button added to the product display.

The “#shoppinglistpopup” of the first link will display a popup window which requests a list name from the user. The product will be inserted into the selected item list.