

Multi-Level Navigation Menus for Experience Cloud

As of: Winter '23

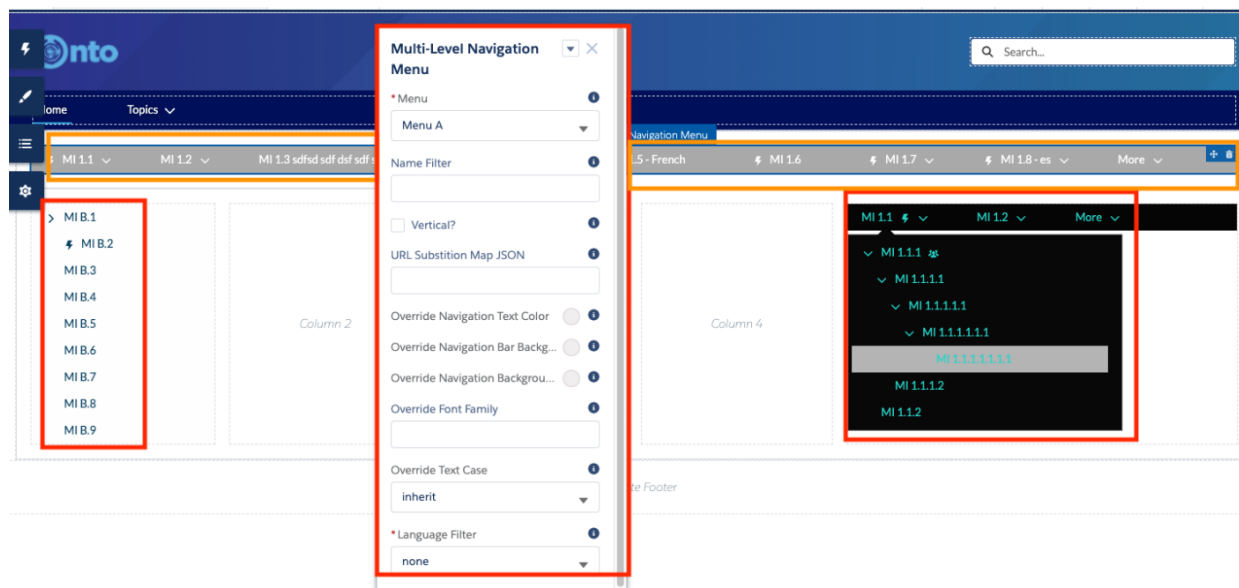
Authored By: George Abboud

Last Updated: 10/12/2022

Reviews and Contributions:

Craig Johnson, Manish Aggarwal, Yelena Kamyshina, David Goldbrenner

Overview:



![[image]](<https://user-images.githubusercontent.com/8514282/153256304-0be3c849-8e05-467c-96ca-db7b51edb77c.png>)

AppExchange Listing:

<https://appexchange.salesforce.com/appxListingDetail?listingId=a0N4V00000FZbCXUA1>

Open-Source GitHub Repository:

<https://github.com/SalesforceLabs/MultiLevelNavigationMenus>

<https://github.com/salesforce-experiencecloud/MultiLevelNavigationMenu>

Description

The Multi-Level Navigation Menu for Experience Cloud app offers components to manage multi-level navigation menus, and render them in your Experience Cloud site. You can configure up to 6 levels of depth in your menus, control the look and feel with styling configuration, horizontal vs vertical presentation, language filtering support, and much more!

DISCLAIMER:

This package is free to use, but is not an official [salesforce.com](https://www.salesforce.com) product, and should be considered a community project. The functionality is not officially tested or documented, and does not come with any support, warranties, or updates. It is offered as-is.

Usage and Configuration

Community Menus App (Menus Manager component for App Builder)

Access the Lightning App called “Community Menus” from the Lightning experience as an admin or a user who is given access to that app and underlying objects (ccnavmenus__Menu__c, ccnavmenus__Menu_Item__c). The home page of the app will run the Menus Manager component (menusManager), which uses an apex controller called “menusManagerController.cls”. Ensure that any user needing to run this functionality also has access to that apex class to be able to run the Menus Manager component.

Community Menus

Home

Menus

Menu Items

* Choose Menu

Menu A - test 1

New Menu

Import Menu

Edit Menu

Delete Menu

Export Menu

Create Menu Item

Filter By Language

none

Name	Label	URL	Position	Opens in...	Public	Language
MI 1.1	MI 1.1	javascript:void(0);	1	No	Yes	
MI 1.1.1	MI 1.1.1	javascript:void(0);	1	No	Yes	
MI 1.1.1.1	MI 1.1.1.1	javascript:void(0);	1	No	Yes	
MI 1.1.1.1.1	MI 1.1.1.1.1	https://ccnavmenus-dev-ed.preview.salesforc...	1	No	Yes	
MI 1.1.1.1.1.1	MI 1.1.1.1.1.1	https://www.google.com	1	Yes	Yes	
MI 1.1.1.1.1.1.1	MI 1.1.1.1.1.1.1	/s/	1	No	Yes	
MI 1.1.1.1.1.2 - French	MI 1.1.1.1.1.2 - French	javascript:void(0);	2	No	Yes	fr
MI 1.1.1.2	MI 1.1.1.2	javascript:void(0);	2	No	Yes	
MI 1.1.2	MI 1.1.2	javascript:void(0);	2	No	Yes	
MI 1.2.1	MI 1.2.1	javascript:void(0);	1	No	Yes	
MI 1.3 sdfsd sdf dsf sdf sdfsd	MI 1.3 sdfsd sdf dsf sdf sdfsd	javascript:void(0);	3	No	Yes	
MI 1.3.1	MI 1.3.1	javascript:void(0);	1	No	Yes	
MI 1.3.2	MI 1.3.2	javascript:void(0);	2	No	Yes	
MI 1.4	MI 1.4	javascript:void(0);	4	No	Yes	
MI 1.5 - French	MI 1.5 - French	javascript:void(0);	5	No	Yes	fr;fr_FR
MI 1.6	MI 1.6	javascript:void(0);	6	No	Yes	

![[image]](<https://user-images.githubusercontent.com/8514282/153256405-76d78f53-46ad-4ac7-ae96-4662c8a0a3ad.png>)

When a menu is not selected, you will have two enabled buttons:

- New Menu: Opens a modal in which you can create a new menu record (layout is based on assigned page layout)
- Import Menu: Opens a modal in which you can select a .json file that you had previously exported, to import the menu based on

When a menu is selected, you will have a few additional buttons enabled:

- Edit Menu: Opens a modal in which you can edit the selected menu record (layout is based on assigned page layout)
- Delete Menu: Opens a modal in which you are asked to confirm the intent to delete the selected menu record. Note: This will also delete all child menu items
- Export Menu: Prompts user to save a .json file that is an export of the selected menu in JSON format.
- Create Menu Item: Opens a modal in which you are able to create a first-level menu item record (layout is hard coded and NOT based on page layout)

Creating menu items:

- Create the first menu item by clicking Create Menu Item after a Menu is selected.
- Create child menu items by clicking the arrow to the right of the parent menu item.
- The position assigned to the menu item is always in relation to sibling menu items

Using Translations:

- There are 2 options for multi-lingual Nav Menus: 1. Create separate menu item per language or 2. Create a single menu with multiple languages

Using User field substitutions:

The multi-level navigation menu also supports user field substitutions by defining the JSON substitution map with one of the supported tokens below:

- [@User.Id]
- [@User.AccountId]
- [@User.ContactId]
- [@User.FirstName]
- [@User.LastName]
- [@User.CommunityNickname]

For example, you might have a menu item defined with a url or label that contains your own token to be replaced with the user's Id, such as:

- URL: /profile/[user-Id]

Then you would have to define the **URL Substitution Map JSON** property in builder with the following JSON:

```
[
  {
    "replaceThis": "[user-id]", //your custom token from the menu item
```

```

        "replaceWith": "[@User.Id]" //from the supported tokens list in the component
    }
}

```

You can follow the example above and modify it to replace with the user's Account Id, Contact Id, First Name, Last Name, and Community Nickname.

Here's an example of each:

```

[
  {
    "replaceThis": "[UserCommunityNickname]", //your custom token from the menu item
    "replaceWith": "[@User.CommunityNickname]" //from the supported tokens list in the
  },
  {
    "replaceThis": "[UserLastName]",
    "replaceWith": "[@User.LastName]"
  },
  {
    "replaceThis": "[UserFirstName]",
    "replaceWith": "[@User.FirstName]"
  },
  {
    "replaceThis": "[User.Id]",
    "replaceWith": "[@User.Id]"
  },
  {
    "replaceThis": "[ContactId]",
    "replaceWith": "[@User.ContactId]"
  },
  {
    "replaceThis": "[AccountId]",
    "replaceWith": "[@User.AccountId]"
  }
]

```

Navigation Menu component for Experience Builder

Component Label: CC Navigation Menu

Component Aura API Name: navMenu

Component LWC API Name: nav-menu

Component Namespace: ccnavmenus

Component Properties:

```

<property name="overflowLabel" type="String" label="Overflow Label" required="true" def
    description="Text to display for the overflow menu item." />

<property name="drillDownBackButtonLabel" type="String" label="DrillDown Ba
description="Label to display for the back button in drill down menu." />

<property name="allLabel" type="String" label="DrillDown Go To Label" requi
description="Label to display for going to menu item that has children and

```

Property Label	Aura Property API Name	LWC Property API Name	Type	Description
Menu	menuId	menu-id	String	(required) Choose which menu to display. If "[Name Filter]" is chosen, you have to provide a matching value in the Name Filter property to match the Menu record's Name field. Defaults to "[Name Filter]"
Name Filter	nameFilter	name-filter	String	(optional) Provide a value to match a menu's Name on. The value can be hardcoded, or a merge field such as {!recordId}, {!recordName} {!urlParameter}. Using merge fields can facilitate creating dynamic menu's based for pages with dynamic variables.
Menu Alignment	menuAlignment	menu-alignment	String	(required) Horizontal Alignment of Navigation Bar. Choices are: Left, Center, Right Default: Left
Menu Mode	menuMode	menu-mode	String	(required) Default Drop Down Menu that keeps all items open, or Drill Down that hides other menu items except what is currently active. Choices are: Default, Drill Down Default: Default
Vertical Mode	verticalMode	vertical-mode	String	(vertical) On to force vertical mode always, off to never render vertical mode, mobile only to render on mobile. Choices are: mobile-only, on, off Default: mobile-only
Hamburger Menu Mode	hamburgerMenuMode	hamburger-menu-mode	String	(required) Mode for when to hide menu behind hamburger. Defaults to 'mobile-only' which will render a hamburger menu only on mobile devices. 'on' will render a hamburger menu all the time. 'off' will never render a hamburger menu. Hamburger menu will take precedence over vertical mode.
Language	language	language	String	(required) Provide a language to filter menu on. If 'auto' is chosen, for unauthenticated / guest users the language is populated based on the language picker component ('language' url parameter), otherwise for authenticated users the language is based on their language / locale selection in their user settings. If translation is not needed, set to 'none' value. Valid options are "none", "auto"
Overflow Label	overflowLabel	overflow-label	String	(required) Text to display for the overflow menu item. Default: More
DrillDown Back Label	drillDownBackButtonLabel	drill-down-back-button-label	String	Label to display for the back button in drill down menu. Default: Back
DrillDown Go To Label	allLabel	all-label	String	Label to display for going to menu item that has children and is linked. Default: Go to
Override Navigation Text Color	brandNavigationColorText	brand-navigation-color-text	String	(optional) Provide color code to override the Navigation Text Color inherited from the theme

SHARING, ACCESS, AND SECURITY REQUIREMENTS

The Multi-Level Navigation Menu lightning component and underlying data model follow the Salesforce platform's security and sharing concepts.

1. In order for a user to be able to run the component, the user has to be granted access to the Apex Controller "*cnavmenus__menusController*" (either via profile or permission set).
2. Since the apex controller runs in "*with sharing*" mode, in order for the user to be able to successfully query and render a menu, the following needs to occur:
 - a. The user must be granted "*Read*" CRUD permission to both the "*Menu*" and "*Menu Item*" objects (either via

Override Navigation Bar Background Color	brandNavigationBarBackgroundColor	brand-navigation-bar-background-color	String	(optional) Provide color code to override the theme
Override Navigation Background Color	brandNavigationBackgroundColor	brand-navigation-background-color	String	(optional) Provide color code to override the theme
Override Font Family	fontFamily	font-family	String	(optional) Provide Font Family to override the theme
Override Text Case	textTransform	text-transform	String	(optional) Provide Text Case to override the theme. The values are "inherit", "none", "capitalize", "lowercase". The default is "inherit".
Top Level Item Spacing	topLevelItemSpacing	top-level-item-spacing	Integer	(required) Set spacing in pixels between Top Level Items. The default is 20.
Menu Style Classes	navMenuClassNames	nav-menu-class-names	String	(optional) Class names to uniquely target a navigation menu. The default is "nav-menu".
URL Substitution Map JSON	urlSubMapJson	url-sub-map-json	String	(optional) Provide a JSON map for substituting values defined in the map being passed. Example: [{"replaceThis":"[!recordId]", "replaceWith":"[!recordId]"}] This would replace urls that have [!recordId] with [!recordId] populated.
Vertical? (Deprecated)	isVertical	is-vertical	Boolean	If checked (true) the menu will render in vertical. Defaults to false and renders horizontally.

```
urlSubMapJson="{!v.urlSubMapJson}"
brandNavigationColorText="rgba(255, 255, 255, 1)"
brandNavigationBarBackgroundColor="rgba(255, 255, 255, 1)"
brandNavigationBackgroundColor="rgba(255, 255, 255, 1)"
fontFamily="Salesforce Sans"
textTransform="uppercase"
topLevelItemSpacing="20"
/>
```

LIMITATIONS

From a declarative perspective, this component cannot be used to swap out the nav menu in the header from theme settings. This is due to the unavailability of a nav menu interface to implement that would allow for such a swap. The availability of this interface is currently not on the product roadmap.

From a development perspective, including this component in a custom LWC behaves differently, depending on if you're using your custom component in LWR vs Aura Runtime. In LWR, you can use a custom LWC which includes the "*cnavmenu-nav-menu*". On the other hand, in Aura Runtime you cannot use a custom LWC that includes other LWCs from different namespaces due to Locker Service limitations.

UN-LIMITATIONS (USE CASES)

- This component can be called in the markup of any Aura lightning component. An Aura lightning component serving as a custom theme layout would make it possible to swap out the standard navigation menu with this navigation menu component instead.
- This component can be dragged / dropped into a "shared" header region of the theme, declaratively, if such a region is available in the selected theme.
- This component can be dragged / dropped into any page in builder to be used at a page level if needed.

Features

1. Multi-level support for up to 6 levels deep
2. URLs - supports fully qualified and relative urls
3. Control whether each menu item opens in the same or a new window
4. Control whether each menu item is available for the public / guest / unauthenticated user
5. Support for languages and localization
6. Choose an icon for each menu item and control whether it is placed on the left or right of the label, or no icon at all
7. Export / Import menus for ease of maintenance and deployment from one org to another
8. Display a menu vertically or horizontally. When vertical, if url of page matches a menu item's url, it is automatically expanded and highlighted
9. (Advanced) Use Name Filters and hardcoded values or merge fields to match menu names and render different menus dynamically.
10. (Advanced) Provide JSON token maps to replace tokens you leave in menu item urls, with the values specified in your map (including support for merge fields)
11. Use the power of audience targeting to target different menus for different audiences
12. LWR Ready

Release Log

VERSION 2.6

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005mS8m>

- Added RelaxedCSP Support
- Added component icon

VERSION 2.5 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005mS54>

- Added options for user field substitutions using the JSON Substitution map
- Renamed LEX app from **Community Menus** to **Experience Cloud Menus**

VERSION 2.4 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005mS4u&isdtp=p1>

- Added drill down menu option
- style fixes

- url json substitutions work for both label and url now
- bug fixes
- accessibility fixes

VERSION 1.27 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005mS1q&isdtp=p1>

- Fixed bug with builder mobile preview and CSS for close button on hamburger menu

VERSION 1.25 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005mRfy&isdtp=p1>

- Fixed bug with text transform not applying to menu items

VERSION 1.24 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005mRMB>

- Removed the feature restriction preventing the app from being installed in orgs with no external user licenses

VERSION 1.23 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005mRLm>

- LWR Ready
- Style inheritance and fixes

VERSION 1.22 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005b3Yx>

- Fix for updating public property in renderedCallback causing infinite loop

VERSION 1.21 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005b3Ys>

- browser console error fixes, nav menu style classes property

VERSION 1.18 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005b3H2>

- Bug fix for click on parent div of a menu link

VERSION 1.17 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005b38J>

- Bug fix for chevron icon click next to a linked menu item not expanding/collapsing child items

VERSION 1.16 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005b389>

- Bug fix for click on icon in a menu item not going to url

VERSION 1.15 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005b2oQ>

- Accessibility Fixes

VERSION 1.14 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005b2oL>

- Styling Fixes
- Added RelaxedCSP capability

VERSION 1.11 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000005b2kE>

- Fixed errors from locker service

VERSION 1.9 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000004Gpxg>

- Fixed issue with hamburger menu not closing upon navigation

VERSION 1.8 (DEPRECATED)

Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t5w000004Gps1>

- Fixed issue with Winter '21 input field's readonly attribute not passing values to record being created / updated

VERSION 1.7 (DEPRECATED)

Installation URL: <https://login.salesforce.com/package/installPackage.apexp?p0=04t5w000004Gprw>

- Styling updates and fixes

VERSION 1.2 (DEPRECATED)

Installation URL: <https://login.salesforce.com/package/installPackage.apexp?p0=04t5w000005uZyE>

- Added CRUD / FLS checks based on security review feedback

VERSION 1.1 (DEPRECATED)

Installation URL: <https://login.salesforce.com/package/installPackage.apexp?p0=04t5w000005uZds>

- Fixed clicking on url to close the popup when menu is in horizontal mode
- Fixed double row actions in menu manager
- Fixed action popup in menu manager to be to the left of the button icon
- Changed language selection when creating or editing menu item to be within accordion

VERSION 1.0 (DEPRECATED)

Installation URL: <https://login.salesforce.com/package/installPackage.apexp?p0=04t5w000005uZdi>

- Initial Release