
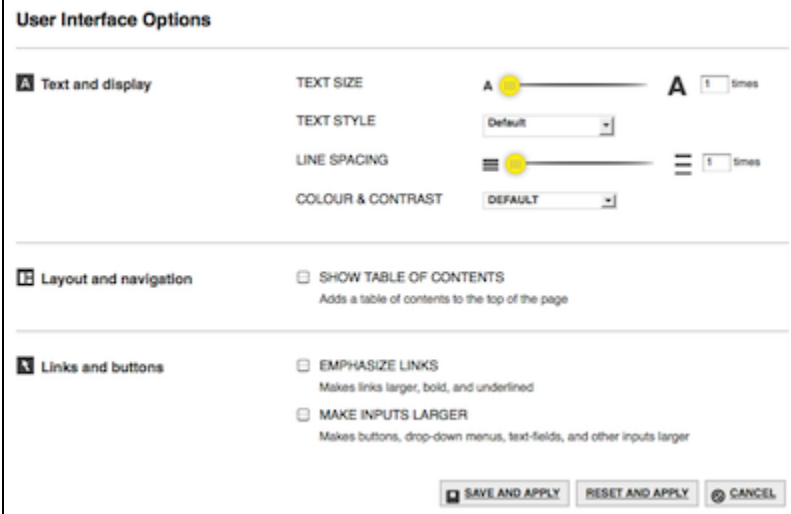


Full Page UI Options

 This documentation applies to v1.4 of UI Options.
For earlier versions, see [UI Options API](#).



User Interface Options

Text and display

TEXT SIZE: A A 1 times

TEXT STYLE: Default

LINE SPACING: 1 times

COLOUR & CONTRAST: DEFAULT

Layout and navigation

SHOW TABLE OF CONTENTS
Adds a table of contents to the top of the page

Links and buttons

EMPHASIZE LINKS
Makes links larger, bold, and underlined

MAKE INPUTS LARGER
Makes buttons, drop-down menus, text-fields, and other inputs larger

SAVE AND APPLY RESET AND APPLY CANCEL

Production Status: PREVIEW

The *User Interface Options* component (*UI Options*) works in conjunction with the *User Interface Enhancer* component (*UI Enhancer*) and the *Fluid Skinning System* (*FSS*) to adjust the user interface based on personal preferences:

- UI Options presents the user with controls for adjusting preferences.
- UI Enhancer is the engine that transforms pages according to the saved preferences.

The **Full Page** version of UI Options displays the UI Options controls in a standalone page.

See Also

[Tutorial - Full Page UI Options](#)
[UI Options Instructional Demos](#)
[Full Page UI Options \(with Preview\)](#)
[Fat Panel UI Options](#)
[UI Enhancer](#)

New in 1.4

(October, 2011)

The UI Options component has been completely refactored in version 1.4:

- The component is now available in three versions: Full Page, Full Page with Preview, and Fat Panel.
- The interface has been completely redesigned.
- The API has been updated.

#Creator	<code>fluid.uiOptions.fullNoPreview(container, options)</code>
Supported Events	<i>none</i>
Methods	<i>none</i>
#Options	<code>#prefix</code> A relative path to templates.

#Configurable Subcomponents	#layoutControls The <code>layoutControls</code> subcomponent renders the controls for customizing the layout of the page.
	#linksControls The <code>linksControls</code> subcomponent renders the controls for customizing the appearance of links on the page.
	#templateLoader The <code>templateLoader</code> subcomponent is responsible for fetching HTML templates for the various controls.
	#textControls The <code>textControls</code> subcomponent renders the controls for customizing the appearance of text on the page.
	#uiOptions The <code>uiOptions</code> subcomponent is responsible for the main rendering of the controls.

Creator

[back to top](#)

Use the following function to create a Full Page UI Options component:

Method	<code>fluid.uiOptions.fullNoPreview(container, options);</code>
Description	Instantiate a full-page version of the UI Options component, with no Preview.
Parameters	<p><i>container</i> A CSS-based selectors, single-element jQuery object, or DOM element that identifies the root DOM node where the UI Options interface should be placed.</p> <p><i>options</i> An optional data structure that configures the UI Options component, as described below.</p>
Returns	The Full Page UI Options component
Examples	<pre>var myFullPageUIO = fluid.uiOptions.fullNoPreview("#myContainer", { prefix: "../components/uiOptions/html/" });</pre>
Notes	
See also	Full Page UI Options Fat Panel UI Options

Options

[back to top](#)

The second argument to the creator function is the options argument. This is a JavaScript object containing name/value pairs: The name is the name of the option and the value is the desired setting. Components define their own default values for options, but integrators can override these defaults by providing new values using the options argument. For technical information about how options are merged with defaults, see [Options Merging](#).

```
var fullUIO = fluid.uiOptions.fullNoPreview(".myContainer", {
    <option1Name>: <option1value>,
    <option2Name>: <option2value>
    ...
});
```

The options supported by Full Page UI Options are described below.

prefix

Description	The <code>prefix</code> option allows you to specify a custom relative path to the templates used by the UI Options interface. These templates can be found in the source tree of the Infusion distribution. The value of the <code>prefix</code> option will be passed on to the Template Loader subcomponent.
Default	<code>"../html/"</code>

Example	<pre>fluid.uiOptions.fullNoPreview("#myContainer", { prefix: "../../components/uiOptions/html/" });</pre>
Notes	Note that if the Template Loader values do not use the %prefix notation, the value of the prefix option will be ignored.
See also	Template Loader

Configurable Subcomponents

Configurable subcomponents can be configured through the parent component's options by passing a JavaScript object containing options for the subcomponent:

```
parent.component.name(".myContainer", {
  <subcomponentName>: {
    options: {
      <subcomponent options>
    }
  }
});
```

To learn about the specific options for each subcomponent, visit the default implementation's API page, as described below.

layoutControls

Description	The <code>layoutControls</code> subcomponent renders the controls for customizing the layout of the page.
Default implementation	fluid.uiOptions.layoutControls
Example	<pre>fluid.uiOptions.fullNoPreview(".myContainer", { layoutControls: { options: { listeners: { onUIOptionsRefresh: myRefreshListener } } } });</pre>
See also	Layout Controls

linksControls

Description	The <code>linksControls</code> subcomponent renders the controls for customizing the appearance of links on the page.
Default implementation	fluid.uiOptions.linksControls
Example	<pre>fluid.uiOptions.fullNoPreview(".myContainer", { linksControls: { options: { listeners: { onUIOptionsRefresh: myRefreshListener } } } });</pre>
See also	Links Controls

templateLoader

Description	The <code>templateLoader</code> subcomponent is responsible for fetching HTML templates for the various controls.
Default implementation	fluid.uiOptions.templateLoader
Example	<pre>fluid.uiOptions.fullNoPreview(".myContainer", { templateLoader: { options: { templates: { linksControls: "../templates/CustomLinksTemplate.html" } } } });</pre>
See also	Template Loader

textControls

Description	The <code>textControls</code> subcomponent renders the controls for customizing the appearance of text on the page.
Default implementation	fluid.uiOptions.textControls
Example	<pre>fluid.uiOptions.fullNoPreview(".myContainer", { textControls: { options: { listeners: { onUIOptionsRefresh: myRefreshListener } } } });</pre>
See also	Text Controls

uiOptions

Description	The <code>uiOptions</code> subcomponent is responsible for the main rendering of the controls.
Default implementation	fluid.uiOptions
Example	<pre>fluid.uiOptions.fullNoPreview(".myContainer", { uiOptions: { options: { autoSave: false } } });</pre>
See also	UI Options