Working With A Preferences Editor On Your Site

Once User Interface Options or a Preferences Editor is incorporated into your website, it's important to ensure that the site continues to transform correctly as new features are added. As you update your site you will need to test your changes and note how they respond to transformations.

Contrast Themes

The one thing you're probably least accustomed to thinking about is ensuring support for UIO's contrast themes. These themes transform the user interface into one of four high-contrast or one low contrast colour combinations:

- high-contrast
    - black-on-white,
    - white-on-black,
    - black-on-yellow and
    - yellow-on-black.
- low-contrast
    - light grey on dark grey

The themes attempt to remove any other colours from the interface, along with gradients, shadows, background images, etc. This type of contrast interface can be extremely helpful for users with different forms of low vision.

UIO applies contrast themes by adding a special class to the body of the document and providing styles that are scoped to that class. You will likely need to create a few more special classes, scoped to the themes, for some areas of your site.

Support for these contrast themes can be ensured by following a few basic guidelines:

- Implement logos, banners icons, etc. as background image; Create high-contrast versions of them and use CSS to swap in the high-contrast versions when the themes are activated.
- In other places where background images are used, either create high-contrast versions, or switch to a solid colour in high-contrast
- Get rid of shadows.
- Double-check use of CSS "content" property

How to define special styles for contrast themes

UIO will do a pretty good job of transforming your site into the selected contrast, but some parts might need special attention to ensure that they look. If you do need to define styles for something that doesn't transform automatically, here's how:

Scope your styles to the theme name

UIO add the theme name as a class to the body element. Scope your selector using the class name as shown in the following example. This example is for the yellow-on-black theme, which uses the class name `fl-theme-uio-yb`:

```
/* high-contrast background icon for button */
.fl-theme-uio-yb .toolbar .saveButton {
    background-image: url('images/save-yellow.png');
}
```

Make sure you have styles for all the themes

UIO defines five contrast themes:

- yellow-on-black (class name `fl-theme-uio-yb`),

- white-on-black (`fl-theme-uio-wb`),
- black-on-yellow (`fl-theme-uio-by`), and
- black-on-white (`fl-theme-uio-bw`)
- light grey on dark grey (fl-theme-uio-lgdg)

If you need to define a style for a contrast theme, you'll most likely need to define if for all the themes. The styles will likely look very similar to each other, with differences only related to the colours:

```
/* contrast background icon for button. these images have a transparent background */
.fl-theme-uio-yb .toolbar .saveButton { background-image: url('images/save-yellow.png'); }
.fl-theme-uio-wb .toolbar .saveButton { background-image: url('images/save-white.png'); }
.fl-theme-uio-by .toolbar .saveButton,
.fl-theme-uio-bw .toolbar .saveButton { background-image: url('images/save-black.png'); }

.fl-theme-uio-lgdg .toolbar .saveButton { background-image: url('images/save-darkGrey.png'); }
```

Avoid image tags if background images can be used

If you're using images to give your buttons or menu items a unique look, follow these guidelines:

- don't use `<img>` tags; use semantically appropriate tags with background images. Background images can easily be replaced with contrast versions using CSS only.
- ensure background images are sized to scale to fit their container (for more info, see https://developer.mozilla.org/en-US/docs/CSS/background-size)
  - `contain` and `cover` preserve proportions of image
  - a percentage will stretch image
  - use `background-size: 100%;`
- don't build the label text into the image; put it in the tag
- if the 'image' is a gradient, consider using CSS gradients

Testing Your Changes With UIO

Enlarge the font

Open the UI Options interface (be it the Separated Panel or the Full Page version) and use the "Text Size" control (in the "Text and Display" tab) to enlarge the font. Check the site:

- does the text enlarge correctly?
- is the layout of the entire page still reasonable?

Increase the line spacing

Open the UI Options interface (be it the Separated Panel or the Full Page version) and use the "Line Spacing" control (in the "Text and Display" tab) to increase the line spacing. Check the site:

- is there increased space between all of the lines of text?
- is the layout of the entire page still reasonable?

Try each of the themes

Open the UI Options interface (be it the Separated Panel or the Full Page version) and use the "Colour &Contrast" control (in the "Text and Display" tab) to select each of the contrast themes. Check the site:

- is all of the text in the correct colours?
- do buttons and links appear in the selected contrast?
- do icons and logos appear in the selected contrast?
- does anything still have shadows or gradients in the wrong colours?
- are hover and focus styles reasonable?

Try each of the fonts

Open the UI Options interface (be it the Separated Panel or the Full Page version) and use the "Text Style" control (in the "Text and Display" tab) to select each of the different font families. Check the site:

- is all of the text in the correct font?
- is the layout of the page still reasonable?
  some fonts take up more room than others, and sometimes selecting a different font might affect the overall layout of the page.

Enhance links and enlarge inputs

Open the UI Options interface (be it the Separated Panel or the Full Page version) and use the controls in the "Links and Buttons" tab to enhance links and buttons. Check the site:

- are links underlined, bolded and enlarged?
- are inputs (such as text fields and buttons) enlarged?