Find

Status of Color Modes in Tiki - April 23

Background

Tiki Feature Requests

Support for website color modes in Tiki is coming along at an interesting time. About a year ago there was a feature request by hman ↗ at dev.tiki.org for the ability to switch a Tiki site to black, to have a light-on-dark appearance. The purpose was for the website to be able to make a social statement.

This idea of dark mode options was added to the Tiki 25 UI Improvement Brainstorming ↗ page in March, 2022.

A little later there was another request, by Bernard, ↗ for Tiki to be able to support a set of options for accessibility, including dark mode, high contrast, gray scale, and so on.

It's nice that these ideas will soon be realized. A little after those feature requests were submitted at dev.tiki.org, Bootstrap 5.3 alpha 1 was released with support for color modes, so the means was provided to meet the user requests without us having to create the feature from scratch.

Bootstrap Evolution

In Bootstrap, CSS custom properties, or CSS variables ↗, were playing a bigger role, including in their use in implementing color modes ↗.

We had conversations about color modes in the chat, which was helpful for me personally because I really hadn't given this much thought before. About the same time, I got an external monitor for the laptop that's my everyday PC, which is big and bright, and I learned first-hand the benefits of dark mode, which I now use a lot of the time especially at night and on cloudy days.

File additions and changes

Color modes are done via the Bootstrap stylesheet, and so in Tiki on the theme level. It may be possible to use a custom CSS file or CSS rules in Tiki's Look & Feel CSS Customization textarea to maybe have a global color mode switch that covered all themes, but I didn't test this yet. It does seem necessary for a theme to have the necessary CSS variables, including dark variants, for color modes to work. What I have been doing is adapting the themes in the Tiki package for color modes one theme at a time.

Each theme has had a variables file (_variables.scss), and now each theme also has a _variables-dark.scss file that has dark alternative values for many color-related objects in the original variables file. Both the light and dark variables are ultimately expressed as CSS variables, which is how the site can be switched

instantly from one mode to the other without a page refresh. Modern web browsers are powerful enough to find all the rules using CSS variables in the stylesheet, and replace them with the other-color alternatives instantly and locally.

What page elements are involved?

Of course the body background and foreground colors are switchable as a starting point, but quite a few other page elements including links, cards, dropdowns, and so on are intended to be color-mode responsive (see https://getbootstrap.com/docs/5.3/customize/color-modes/ ⬀. Page elements can be color-mode responsive if they are assigned classes that use the CSS variables for this purpose ⬀.

It addition to the elements covered by Bootstrap variables, we probably will want to shift from SCSS variables to CSS variables in selectors, such as for toolbars and other places where color mode switching isn't working in themes, or to switch to SCSS variables that have a dark variant already. These would use default Bootstrap CSS variables or Tiki-specific ones. I'm experimenting with these details now and will document.

My implementation procedure

I made a master clone where I manually replaced the Bootstrap subdirectory in the vendor directory with the Bootstrap 5.3 files.

I keep the theme files updated along with everything else, and I made duplicates of all the theme file directories and renamed them like "amelia53" (for Bootstrap 5.3). I added these directory names to the exclude file in the .git/info/ directory so they aren't tracked by git.

I tried compiling the base_files and the default53 theme files and got compile errors, not surprisingly, but only a few, and the ones in base_files involved things that could be changed in the master branch, so I updated those in master and now themes in my color mode test instance compile without error once the theme itself has been updated for Bootstrap 5.3..

To update default53, I added Bootstrap's new variables-dark file to the default theme's SCSS directory and tried compiling. There are also several variables that need to be added for each theme. Trying to compile, there would be errors if they variables were missing or not defined, so I addressed each compile error one by one until a default53.css could be compiled without errors.

I continued through the other themes, using the Bootstrap variables-dark values as a starting point, so now for the most part, the dark modes of most of the themes use the same color palette. Later the colors can be adjusted to better match the theme, if we like.

There were problems such as the left-column system menu (or other such menu) not having the correct colors, and I tried one method of adding new rules to tiki-selectors to set colors specifically for vertical navbars. But then I thought it would be better to find a solution that didn't add more CSS rules if possible, so I looked again at Bootstrap's variables file and replicated it regarding colors. I'm analyzing the details and documenting it, as this works with no additional CSS rules. The upshot is that I have the foundation of a method to convert our themes.

In testing, there are some page elements in some themes that have hard-coded colors that don't work well in dark mode, such as cards, tabs backgrounds, toolbars, and so on. These need solutions based on Bootstrap's color utility classes ⬀ and so on, not on new Tiki CSS rules.

Using browser prefers-color-scheme

Bootstrap's color modes enables two states by default — light and dark — that are set by the user's browser light or dark color theme preference. In my test instance, when I specify dark mode in FireFox's settings, for example, then when the Tiki pages use the Bootstrap dark mode, and vice versa. I can also switch to the mode I'm not using, on the page, or switch to "auto", which activates the browser preference again.

Color mode interface in Tiki

The Bootstrap 5.3 docs site suggests using the JavaScript that they use, for the color mode switch. In my test instance, I copied the Bootstrap HTML for the switch and the JavaScript. I put these in a wiki page and then used the Pagemenu module to display the switch in the top module zone. I also put a switch in the Unified Admin Backend bottom module zone so color mode switching would also work there.

Other points

Here are some specific details that need to be worked out or decisions made about:

Themes that are already dark

So far, I haven't made alternative modes for most dark themes, like Darkly or Slate. We could have a light Darkly and Slate, etc. but I don't know what the rationale would be, not accessibility for sure. Maybe if a dark theme has particular nice typography or other design details, it might be good to provide a light version, so I'm open to requests on that point.

Light and dark navbar options

The relationship between navbar background options and color modes: Tiki has had "light" and "dark" options for navbars, such as in the top and topbar module zones, for a while now, which are selected by the site admin and have global effect. And when color modes are implemented, we'll have light and dark options for the page overall, and these will be switchable by the site visitor/user via a switch in a module assigned by the site admin. It's possible from the coding standpoint to specify colors for all of these light and dark combinations but is it useful to do that?

My own tentative conclusion is that both light and dark navbars make sense on a light page, but a light navbar doesn't work well on a dark page, especially if dark mode is chosen by the user for accessibility reasons. If someone wants to make a theme very feature complete, then, for dark color mode, the light/dark navbar options could provide two different dark navbars, maybe a neutral, really dark one and one that is a very dark variant of the site's brand color, for example.

Other color modes

By default, Bootstrap can have light and dark modes. (The third position on the switch, "Auto", is for using the browser theme preference.) More options can be added, though, as shown on the Bootstrap docs page (https://getbootstrap.com/docs/5.3/customize/color-modes/#custom-color-modes).

Full page or content area

In most themes, color mode switching affects the whole page, changing the body background and text colors. But in some themes only the content area(s) change. Darkroom and Darkshine are handled this way. This is because the "non-content" areas are significant parts of the theme and maybe switching the content areas only is sufficient to meet accessibility needs.

References

https://dev.tiki.org/Tiki-25-UI-Improvement-Brainstorming
https://dev.tiki.org/item8096-Feature-request-Switching-a-Tiki-to-black
https://dev.tiki.org/item8311-Accessibility-Adding-accessibility-tools-to-Tiki

https://developer.mozilla.org/en-US/docs/Web/CSS/@media/prefers-color-scheme
https://getbootstrap.com/docs/5.3/customize/color/
https://getbootstrap.com/docs/5.3/customize/color-modes/

https://themes.tiki.org/Color-Modes-in-Tiki
https://themes.tiki.org/CSS-Variables-in-Tiki