

Custom Grav Theme Modifications Report

Introduction: This report outlines the customizations made to the Grav CMS theme as part of the development process. The modifications include changes to the color scheme, the addition of a custom logo to the header, and the implementation of a feature to display a list of the most recent pages in the sidebar.

Changing the Color Scheme: To enhance the visual aesthetics of the theme, the color scheme was customized. This was achieved by updating the `tailwind.config.js` file with the following changes:

- Extended the theme's color palette with custom primary and secondary colors.
- Defined a custom shade of gray for text and elements.

These changes were applied to the theme's stylesheet to create a visually appealing and consistent color scheme throughout the site.

tailwind.config.js:

```
module.exports = {
  content: [
    '../..//config/**/*.yaml',
    '../..//pages/**/*.md',
    './blueprints/**/*.yaml',
    './js/**/*.js',
    './templates/**/*.twig',
    './custom-theme.yaml',
    './custom-theme.php'
  ],
  darkMode: 'class', //false or 'media' or 'class'
  theme: {
    extend: {
      colors: {
        primary: '#1299A7',
        secondary: '#F87010',
        gray: {
          '900': '#808080'
        },
      },
    },
  },
  fontFamily: {
    sans: ['Hind Vadodara'],
  },
}
```

```

    },
  },
  variants: {
    extend: {}
  },
  plugins: [
    require('@tailwindcss/forms'),
    require('@tailwindcss/typography'),
    require('tailwindcss-debug-screens')
  ]
}

```

Adding a Custom Logo to the Header:

A custom logo was integrated into the theme's header to establish a unique brand identity. The following steps were taken:

- A logo image was sourced from an external provider.
- The logo image was placed in the custom-theme/images folder within the theme directory.
- An HTML element was added to the theme's header template to display the custom logo.

```

<link rel="icon" type="image/png" href="{{
url('theme://images/favicon.png')|e }}" />

```

Furthermore, consideration was given to providing an admin panel option for logo upload, ensuring ease of customization for site administrators.

Modifying the Sidebar to Display Recent Pages:

To improve user navigation and engagement, the theme's sidebar was customized to display a list of the most recent pages visited by users. The implementation involved the use of session management:

- The current page's URL was retrieved.
- Existing session data or an empty array was initialized to track recent pages.
- The current page's URL was added to the array of recent pages while maintaining a maximum limit (e.g., 3 pages).
- The updated list of recent pages was stored in the session for persistence.
- Twig variables were used to pass this data to the theme's templates for rendering.

```
<?php
namespace Grav\Theme;

use Error;
use Grav\Common\Theme;

class CustomTheme extends Theme
{
    public function onThemeInitialized()
    {
        $this->getMostRecentPages();
    }

    protected function getMostRecentPages(): void {
        try{
            // Get the current page URL
            $pageUrl = $this->grav['uri']->url();
            // Get the existing session data or initialize an empty array
            $recentPages = $this->grav['session']->recent_pages ?? [];

            // Remove the current page URL from the array, if it exists
            if (($key = array_search($pageUrl, $recentPages)) !== false) {
                unset($recentPages[$key]);
            }

            // Add the current page URL to the beginning of the array
            array_unshift($recentPages, $pageUrl);

            // Limit the array to a certain number of recent pages (e.g., 5)
```

```

$maxRecentPages = 3;
$recentPages = array_slice($recentPages, 0, $maxRecentPages);

// Store the updated session data
$this->grav['session']->recent_pages = $recentPages;

// Pass the recent pages data to Twig
$this->grav['twig']->twig_vars['recentPages'] = $recentPages;
}
catch(\Exception $e){
    throw new Error($e);
}
}
}

```

After that i used recentPages from twig variable to show the most recent pages

Conclusion: The modifications made to the Grav CMS theme, including the changes in color scheme, addition of a custom logo, and the implementation of a recent pages feature, enhance the visual appeal, branding, and user experience of the website. These customizations align with the theme's purpose and contribute to a more engaging and user-friendly interface.

This report summarizes the key customizations made to the Grav CMS theme, showcasing my ability to enhance both the visual and functional aspects of the theme to create a more polished and user-friendly website.