Site customization

Contents

Site-wide changes

Each site is configurable by root and admin users in the console (`/user/site` or `admin/sites`).

However, part of the configuration should be done adding some files to the `site_files` directory in the `repo/SITE_ID/` directory, which is the git repository of the site. See `offline-editing-via-git` for the workflow's details.

If you installed Amusewiki via the Debian packages, this directory is located in `/var/lib/amusewiki/repo/SITE_I` (where `SITE_ID` is the site code).

`favicon.ico` If present, this will be used as favicon.

`local.css` Additional CSS rules. Always loaded.

`local-alt.css` Same as above, but loaded only if the alternate theme is in use (so-called dark-mode).

`local.js` Local JS file. Here you can do whatever you want without tampering with the code. Open an issue on GitHub if you think you miss IDs in the HTML to hook on.

`navlogo.png` If present, it will be placed on the navigation bar as logo. Height is enforced to 50px.

`navlogo-alt.png` Same as above, but used when the alternate theme is in use.

`widebanner.png` If present, it's used to place a banner below the navbar (which is fixed). It should have a width of 1170px for optimal results.

`pagelogo.png` If present, it will be placed before the latest entries on the special pages. The assigned ids are "special-pages-logo" for the container and "special-pages-logo-img" for the image itself.

`opengraph.png` If present, it will be used for the opengraph protocol (used on social networks when sharing a link). The image opengraph.png should have dimension 300x300 for better effect. If not present, pagelogo.png will be used, otherwise navlogo.png will be used, with often odd results.

The normal way to add these files would be to use Git, but if you place them manually in the tree, just remember to run `amusewiki bootstrap-archive --refresh SITE_ID` (or `scripts/amusewiki-bootstrap-archive --refresh SITE_ID` if not installed by .deb, from the amusewiki installation directory)

Per-text changes

For maximum flexibility, you should use the `repo/SITE_ID/site_files/local.js` file. Every text has an API endpoint which you can hit to get the values of the header of the text file, regardless if the field is supported or not.

In this example, we don't want to show the body when we use in the header:

```
#hidebody 1
#title My title

Text....
```

We can do this in the `local.js` file:

```
$(document).ready(function () {
    // check if we are in a text page
    var api = $('#page').data('text-json-header-api');
    if (api) {
        $.ajax({
            url: api,
            dataType: 'json',
            success: function(response) {
                // log what we get
                console.log(response);
                if (response.hidebody) {
                    $('.page-header').remove();
                    $('#htmltextbody').remove();
                }
            }
        });
    }
});
```

If the changes don't show up immediately try first to force reload the browser, there is probably cache, while if the file is not loaded at all, try to run `amusewiki bootstrap-archive --refresh SITE_ID` again.

Site customization

amusewiki.org