
Skarf Docs

Release 0.2.2

Woofers

Jul 10, 2022

INSTALLING

1	Installation	3
1.1	With Docker	3
2	Configuring Skarf	5
2.1	Examples	5
2.2	What everything means	8
3	Make Skarf Static	11
4	Credits	13
4.1	Contributors	13
4.2	Images	13
4.3	Other Credits	13

Welcome to the skarf docs which teaches you how to install and configure skarf. Project Source: [Github](#)

INSTALLATION

It is relatively easy to setup skarf with docker.

1.1 With Docker

Install Docker: [Get Docker](#)

1.1.1 Docker Compose

`docker-compose.yml`

```
version: '3.3'
services:
  app:
    container_name: skarf
    ports:
      - '8080:80' # To expose skarf onto the host machine (remove this if you are
↪ using a reverse proxy such as traefik)
    volumes:
      - './static:/app/static' # To store data in the [website]/static/ url
      - './config:/app/config' # To store the config.yml config
    image: 'ghcr.io/woooferz/skarf:master'
```

Then run `docker-compose up` and CTRL+C to get out of it and configure it. Help: [Configure Skarf](#)

1.1.2 Docker CLI (Untested) (Not Recommended)

Run this command:

```
docker run --name skarf -p 8080:80 -v /path/to/static:/app/static -v /path/to/config:/
↪ app/config ghcr.io/woooferz/skarf:master
```

To get out you probably need to do `docker stop skarf` from another shell and then you can configure it. Help: [Configure Skarf](#)

CONFIGURING SKARF

You can configure skarf with YAML and JSON. To configure skarf (in docker) you'll need to mount `/app/config` to a local directory, and then you can put a `config.json` or `config.yml` into that folder. You can see examples of configs on this page, and further down there is a section which tells you what everything means.

2.1 Examples

2.1.1 Example YAML (config.yml)

```
version: 5 # pls no touch this
debug: false
settings:
  name: "Person" # Main name
  subtitle: "Some for info" # A little text below name
  font: Roboto, Arial # The font-family to use for the page
  verified: false # adds a cool checkmark
  check-color: '#55FF11' # decides the color of checkmark
  color: "#6F6F6F" # The color of the name and subtitle
  link-color: "#F6F6F6" # The color of the text on the links
  glass-card: true # If the image, name, subtitle and links should be displayed on a
↳ glass-like card
  custom-css: false # OPTIONAL! url to custom css
  kofi:
    status: false # kofi on / off status
    slug: supportkofi # if your kofi was https://ko-fi.com/supportkofi then you would
↳ put 'wooferz'
    text: Support Ko-Fi # the text on the button (Keep under 13 characters as it will
↳ add a scroll bar otherwise)
    background-color: '#00b9fe' # background color of button
    text-color: '#ffffff' # text color of button
  font-awesome:
    pro: false # if you have pro set to true
    pro-url: "/idk/here/maybe" # if you have pro enter url here, either a local one such
↳ as /static/fontawesome-pro.css or an external one such as https://fontawesome.example.
↳ com/pro.css (BOTH OF THESE WON'T WORK, GET YOUR OWN LINK)
  glass:
    color: "#FFFFFFAA" # color of the glass effect simply remove this if you enjoy the
↳ current color
  background: # The background info
```

(continues on next page)

(continued from previous page)

```

type: color # image or color or gradient Decides what to put as the background
content: "#2F2F2F" # url to image or a color code (e.g. /static/bg.png) or the first
↳ color of a gradient
content-2: "#FFFFFF" # only used when there is a gradient, the second color of
↳ gradient
favicon: /static/favicon.png # If you want a favicon
description: Some info in the meta description tag # If you want a meta description
image:
  type: font-awesome # image or font-awesome decides how to display avatar
  style: solid # for font-awesome. Decides what style to display the icon
  content: user # font-awesome icon name, or url to image
links:
- name: Skarf Docs # Friendly Name of link
  link: https://skarf-docs.readthedocs.io/en/latest/index.html # The link to the link
  image: # same as the 'image' item above
    type: font-awesome
    style: solid
    content: book
    color: "#3F3F3F" # the background color of the link
- name: Skarf Github # same as item above
  link: https://github.com/woooferz/skarf
  image:
    type: font-awesome
    style: brands
    content: github
    color: "#3F3F3F"
- name: Support Discord
  link: https://discord.gg/VnskbWb4Ft
  image:
    type: font-awesome
    style: brands
    content: discord
    color: "#454FBF"
mini-links: # different from 'links' as it is only an image/icon with a link. AKA no
↳ text.
- link: https://github.com/woooferz/skarf # Link to the service
  image: # same as the 'image' item above
    type: font-awesome
    style: brands
    content: github
    color: "#3F3F3F" # color of it

```

2.1.2 Example JSON (config.json)

```
{
  "version": 5,
  "debug": false,
  "settings": {
    "name": "Person",
    "subtitle": "Some for info",
    "font": "Roboto, Arial",
    "verified": false,
    "check-color": "#55FF11",
    "color": "#6F6F6F",
    "link-color": "#F6F6F6",
    "glass-card": true,
    "custom-css": false,
    "kofi": {
      "status": false,
      "slug": "supportkofi",
      "text": "Support Ko-Fi",
      "background-color": "#00b9fe",
      "text-color": "#ffffff"
    },
    "font-awesome": {
      "pro": false,
      "pro-url": "/idk/here/maybe"
    },
    "glass": {
      "color": "#FFFFFFAA"
    },
    "background": {
      "type": "color",
      "content": "#2F2F2F",
      "content-2": "#FFFFFF"
    },
    "favicon": "/static/favicon.png",
    "description": "Some info in the meta description tag",
    "image": {
      "type": "font-awesome",
      "style": "solid",
      "content": "user"
    },
    "links": [
      {
        "name": "Skarf Docs",
        "link": "https://skarf-docs.readthedocs.io/en/latest/index.html",
        "image": {
          "type": "font-awesome",
          "style": "solid",
          "content": "book"
        },
        "color": "#3F3F3F"
      },
    ]
  }
}
```

(continues on next page)

(continued from previous page)

```
"name": "Skarf Github",
"link": "https://github.com/wooferz/skarf",
"image": {
  "type": "font-awesome",
  "style": "brands",
  "content": "github"
},
"color": "#3F3F3F"
},
{
  "name": "Support Discord",
  "link": "https://discord.gg/VnskbWb4Ft",
  "image": {
    "type": "font-awesome",
    "style": "brands",
    "content": "discord"
  },
  "color": "#454FBF"
}
],
"mini-links": [
  {
    "link": "https://github.com/wooferz/skarf",
    "image": {
      "type": "font-awesome",
      "style": "brands",
      "content": "github"
    },
    "color": "#3F3F3F"
  }
]
}
```

2.2 What everything means

version Version of renderer to load config with.

debug If debug mode is on which reloads config on every load and shows some version info on page

settings.name The title of the card, usually your name/nickname.

settings.subtitle Text beneath settings.name with a smaller font size.

settings.font Font of the card, needs to be installed on the viewers device.

settings.verified Verification status, if set to true it adds a cool little checkmark

settings.check-color Decides the color of the checkmark if settings.verified is set to true, if not set it will be the color of settings.color

settings.color Text color of settings.name and settings.subtitle.

settings.link-color Text color of all the links.

`settings.glass-card` If the card should have a glass effect. (From `css.glass`)

`settings.custom-css` A link to custom css

`settings.kofi.status` true/false if you want to enable Ko-Fi Integration.

`settings.kofi.slug` Your kofi link, but only the last bit (e.g. `supportkofi`)

`settings.kofi.text` Button text on Ko-Fi Integration.

`settings.kofi.background-color` Background color of Ko-Fi Integration button.

`settings.kofi.text-color` Text color on Ko-Fi Integration button.

`settings.font-awesome.pro` If you want pro url enabled (true/false)

`settings.font-awesome.pro-url` The link to fontawesome pro you want to use only used if `settings.font-awesome.pro` is enabled.

`settings.glass.color` Optional! If you want a custom color on the glass effect

`settings.background.type` Either color or image or gradient. Decides the type of background your card will have.

`settings.background.content` If `settings.background.type` is color then use a hex color code, if it is image then use a image url. If it is gradient then use the first gradient value

`settings.background.content-2` Only used if `settings.background.type` is gradient, sets the second gradient value

`settings.favicon` Path to a favicon.

`settings.description` Meta description for your card.

`settings.image.type` Either font-awesome or image. Decides the type of avatar you will have.

`settings.image.style` Only useful if you are using font-awesome, it will decide what style of icon you will have. It can be solid or brand, assuming you have the free tier of font awesome

`settings.image.content` If `settings.image.type` is font-awesome it is an icon name. If it is image then an image url.

`settings.links` An array/list of links.

`settings.links.name` The display text of a link.

`settings.links.link` The link of a link.

`settings.links.image` Same as `settings.image`.

`settings.links.color` The background color of a link.

`settings.links.text-color` Overrides `settings.link-color`, it sets the color of the link

`settings.mini-links` Like `settings.links` but it only has a icon and appears in one line, it is an array/list

`settings.mini-links.link` The link of the item

`settings.mini-links.image` Same as `settings.image`

`settings.mini-links.color` The color of the link

MAKE SKARF STATIC

Skarf is created with flask and it will take up your server resources as it is being rendered on the server side. To fix this you can make it static with a workaround kinda thing. Do this after you have finished configuring skarf and you just want the static files left.

First you will need `wget`, it can be installed on unix based systems.

Run this command:

```
wget --recursive --no-clobber --page-requisites --html-extension --convert-links -  
↳ restrict-file-names=windows [link to your skarf instance]
```

This will download your skarf instance and you will be able to view it in static form.

CREDITS

4.1 Contributors

- [woooferz](#) - Main Developer

4.2 Images

`/static/images/avatar.jpeg` - Photo by [Shayna Douglas](#) on [Unsplash](#)

`/static/images/waffle.jpeg` - Photo by [Mae Mu](#) on [Unsplash](#)

4.3 Other Credits

- [Font Awesome](#) - For awesome (free) icons