A guide for maintaining *oshchip.org*

Mahesh Venkitachalam

oshchip.org is hosted at github, and is based on *jekyll*. This is a brief guide to maintaining the site. You can read about *jekyll* here:

http://jekyllrb.com/

Main Configuration file

The main configuration file for the site is *_config.yml*. It's already set up for you, but here are some parts you may like to modify:

Local vs. Remote

While testing locally (we'll come to that in a bit) you need to change the url field:

```
#url:          http://localhost:4000
url:          http://oshchip.org
```

Owner

You can change the fields under owner to suite your needs. For example, here's the bio field:

```
owner:
  bio: "Consultant for FPGA design, especially high end DSP type applications with Xilinx products."
```

Configuring the look of oshchip.org

Here are some notes on configuring the look of the website.

Portrait

The portrait used at the top left comes from the *images/portrait.jpg* file. To change, just replace this file, ensuring that the **width** of the image remains the same.

Welcome Message

The welcome message at the home page, which currently starts with “Welcome to the home of OSHChip...” can be changed by editing *index.md* at the top level.

Header Image

Every post or page has a banner image on top. These can be customized on a per-page basis by setting it in the header of the page. For example, here's the header from the top level *index.md*.

```
---
layout: home
excerpt: "A blog on Electronics & Programming."
tags: [Jekyll, theme, responsive, blog, template]
image:
  feature: header.jpg
---
```

You can specify the path of the image. By default it picks up from the *images* directory, but you can specify a full path starting from the top level instead.

Uploading content to oshchip.org

Here are some instructions on uploading content to the jekyll based oshchip.org website.

Creating a Post

Here is how you create a post. Go into the *_posts* directory, and create a file of the form **2015-10-01-test.md**. The naming of the file matters, since that determines the creation timestamp of the post. You can rename the file later, and correspondingly the creation time will be changed. The string after the date part doesn't matter – just use something that helps you identify the article later. The *.md* extension indicates that the file uses Markdown format.

Header

Every *.md* file begins with a header like the following:

```
---
layout: post
title: Hello from OSHChip
excerpt: "Hello page"
tags: [nRF51822, BLE]
categories: [Electronics]
comments: true
modified: 2015-10-15
thumbnail: images/2015/10/oshchip.jpg
image:
  feature: header.jpg
---
```

The *layout* field says what type of page it is - post, page, home, etc. The *title* is the web page title, and that's what you will see in the browser if you try to bookmark this page. The *excerpt* field is an abstract of the content, and it will appear on the appropriate index page – in this case, on the home page. The *tags* field is important – it indicates to search engines what keywords are relevant to your web page. *categories* is for your internal use – you may later organize your pages via these. You can enable/disable comments on any page using the *comments* field. The modified field lets you set the “last modified” date, and that information will appear at the end of your article. The *thumbnail* field is very useful for external websites such as reddit, facebook, etc. It creates the metadata necessary to pick the right thumbnail image from with your webpage. The *image/feature* field can be used to set the image on a per-page basis.

Formatting Content

I've illustrated the most common formatting options in the *hello* page. Please refer to the associated *_posts/2015-10-15-hello.md* file.

Here's the complete reference for Markdown. Not all features of this spec are supported by Jekyll.

http://daringfireball.net/projects/markdown/

Also, you can intersperse your Mardown post with HTML code. So if you don't know how to do something in Markdown, just use HTML.

Creating a Page

A page is a place for showcasing more prominant parts of your website. What appears on the top menu (All Posts, Contact, etc.) is determined by what is set in the *_data/navigation.html* file. To create a page, just create a directory with the name at the top level, and put an *index.md* file into it. You can copy one from an existing page, such as *about*.

Testing Locally

To test locally, you have to install jekyll. Instructions are here:

http://jekyllrb.com/docs/installation/

I've written two scripts (requires *sh*) that let you test locally – build.sh and serve.sh. First, change your URL setting in *_config.yml* to:

```
url:         http://localhost:4000
#url:          http://oshchip.org
```

Now, in a shell do the following in the top level directory in the git repository:

```
./build.sh
./serve.sh
```

The above will start serving a page. Now, open a browser window and go to http://localhost:4000/ - you should be able to see a preview of your web site. You can keep editing files using another shell, and serve.sh will automatically update the web pages – just give it a few seconds. Once you are done, press Ctrl-C to exit the server.

Deploying

Before you do a git push, make sure that your URL setting in _config.yml is correct! It should be:

```
#url:          http://localhost:4000
url:         http://oshchip.org
```

You need to wait (typically under a minute) for github to build your page and have changes available at the website. At times you may need to clear the browser cache to see the changes.

Good Luck!