Breaking Down Barriers

Using Docsify-This As an Entry Point to Markdown Open Publishing

bit.ly/otessa24-docsify-this

Paul Hibbitts
hibbittsdesign.org
🐘 @hibbittsdesign@mastodon.social

Session Topics

1. Open Content Format Considerations
2. Markdown
3. Docsify-This
4. "In the Wild" Examples
5. Hosting Markdown Files
6. Live Demo
7. Online Activities!

1. Open Content Format Considerations

Open Content Format Considerations

- Support for the 5 Rs (Retain, Reuse, Revise, Remix and Redistribute)
    - For example, is content reusable across different platforms?
- Content editing tools
    - For example, are there Web and/or desktop editing options?
- Publishing workflows
    - For example, how is raw content turned into Web pages?
- Collaboration and version control potential
    - For example, can content revisions be managed?

2. Markdown

What is Markdown?

Markdown is a syntax to format the display of content stored as plain text - similar but simpler than HTML formatting

Markdown Examples

Your H1 Text Here
Your H2 Text Here

your italic text here

your bold text here

* your unordered list item here

   1. your numbered (and indented) list item here

[link title](https://www.google.com)

![Image Alt Text](imagefile.jpg)

Tip: To ensure a new paragraph after text in markdown, put two spaces after the end of the line

Why Markdown?

✓ System-independent

✓ Text-only (future proof *and* perfect for version control!)

✓ Separation of content vs. presentation

✓ Human-readable (i.e. more than HTML)

✓ Can also contain HTML snippets

Example HTML Snippets

```
<div style="text-align: center;">

# This h1 Header is Centered

</div>
```

```
<details>
    <summary>Content title</summary>
    This content can be <em>shown/hidden</em>.
</details>
```

Publishing with Markdown

- ■ Markdown Converters
    - ○ For example, Pandoc (pandoc.org)
- ■ Markdown Static Site Builders
    - ○ For example, Jekyll (jekyllrb.com)
- ■ Markdown Dynamic Renderers
    - ○ For example, Docsify (docsify.js.org)

How can we further reduce the barriers to publishing open content with Markdown?

3. Docsify-This

Docsify-This

Docsify-This is an open source Web app, built using the Docsify Open Publishing Kit, that can instantly turn online Markdown files into web pages

Docsify-This Core Design Principles

1. Frictionless Markdown publishing
2. Flexible content display handling
3. Your content, your control
4. Support the 5 Rs of OER
5. Authors helping other authors

Publishing Markdown Content with Docsify-This

1. Create or view a Markdown file on GitHub, Codeberg or a Webserver and copy that URL
2. Go to **Docsify-This.net**, paste the URL into the **Markdown File URL** field, and tap the **Publish as a Standalone Web Page** button
3. *That's it, there is no step 3!*

🟢 Docsify–This

Instantly Turn Online Markdown Files into Web Pages

This open-source web app, built with the magical documentation site generator Docsify, provides a quick way to publish one or more online Markdown files as standalone web pages without needing to set up your own website. Try it out below!

Web Page Builder

Enter the URL of a Markdown file and view that file as a web page in a new Browser tab. The resulting Docsify-This page URL can be copied and shared or used as an iFrame source URL.

Markdown File URL (GitHub, Codeberg or raw source URL):

https://github.com/hibbitts-design/docsify-this-one-page-article/blob/main/home.md
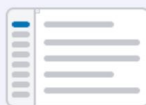
Publish as a Standalone Web Page

Page layout:

Content Only   Table of Contents   Docsify Sidebar

☐ Include 'Edit this Page' link with public GitHub or Codeberg files (default location is bottom of page)

Show More Page Appearance Options »

All About Docsify-This

- What Problem Does Docsify-This Solve?
- Docsify-This Core Design Principles
- How Docsify-This Works
- Read What People are Saying about Docsify-This
- Example Content Workflows
- Ready-to-Use Docsify-This Markdown Templates
- Additional Docsify-This Markdown Examples and Templates
- Page Appearance URL Parameters
- Embedding Docsify-This Pages into Other Platforms

My Open Publishing Space
Create, Share and Collaborate

[Docsify](#) can generate article, portfolio and documentation websites on the fly. Unlike Docusaurus, Hugo and many other Static Site Generators (SSG), it does not generate static html files. Instead, it smartly loads and parses your Markdown content files and displays them as a website.

Introduction

Markdown is a system-independent markup language that is easier to learn and use than **HTML**.

Figure 1: The Markdown Mark

Some of the key benefits are:

1. Markdown is simple to learn, with minimal extra characters, so it's also quicker to write content.
2. Less chance of errors when writing in markdown.
3. Produces valid XHTML output.

Example Docsify-This URL

`https://docsify-this.net/?`**`basePath`**`=https://raw.githubu`
`sercontent.com/hibbitts-design/docsify-this-one-page-a`
`rticle/main&`**`homepage`**`=home.md&`**`sidebar`**`=true`

Time for Questions & Discussion

- What we've covered so far
    - Open content format considerations
    - Markdown
    - Docsify-This
- Coming up
    - "In the Wild" Examples

4. "In the Wild" Examples

OpenArm
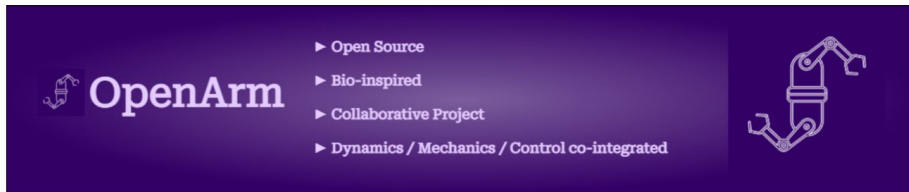
An open-source robot arm

> **PURPOSE: to design a robot arm using bio-inspiration for higher modularity and higher loads than leading commercial designs in 2023.**
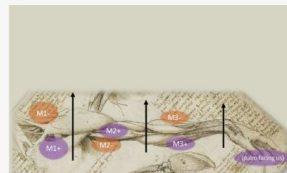
| Github Repo | Website |

Characteristics

Version 1

- Comprised of 3D Printable + OTS parts
- Less than $150 in components
- Using DC motors, M2 Belts, and M2 pulleys like those of 3D printers
- Having no encoders required ❇️
- Bio-inspired - having damping forces and torque relationships similar to a human arm.
- Digital-twin oriented - the simulated design shall represent the real implementation sufficiently to simulate motion.
- Parametric design - the geometry leaves space for variations and modifications of internals & frame.
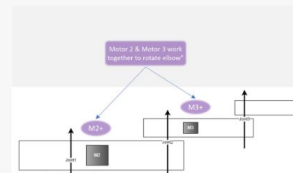
Biological Sketch	Biological sketch with motors identified	Robot Arm concept with motors

Product ▾ Solutions ▾ Open Source ▾ Pricing

Search or jump to... /

Sign in   Sign up

dmalawey / **OpenArm** Public

Notifications   Fork 0   ☆ Star 5 ▾

<> Code   ⊙ Issues   ⑂ Pull requests   ⊙ Actions   ▦ Projects   ⊙ Security   ⌁ Insights

⑂ main ▾   ⑂ 1 Branch   ⊙ 0 Tags

Go to file

<> Code ▾

dmalawey Update README.md

530a5cd · 3 months ago   ⊙ 76 Commits

📁 img	Update readme.md	9 months ago
📄 LICENSE	Initial commit	9 months ago
📄 README.md	Update README.md	3 months ago

📖 README   ⚖ MIT license

OpenArm

An open-source robot arm

> PURPOSE: to design a robot arm using bio-inspiration for higher modularity and higher loads than leading commercial designs in 2023.

| [Github Repo](#) | [Website](#) |

Characteristics

Version 1

About

An Open Source Robot Arm

🔗 qr.page/g/2wY5JrxcciD

education   ai   makers   dynamics

openhardware   3dprinting

📖 Readme

⚖ MIT license

⌁ Activity

☆ 5 stars

👁 3 watching

⑂ 0 forks

Report repository

Scenario

Students don't seem to understand the material in my course. They just don't get it! For some reason they seem to miss the mark in my exams each year no matter what I try to do to prepare them. They seem disinterested and distracted. I want them to understand the relevance and enthusiasm I feel for my content, but I'm out of ideas on how to get through to them.

How do I become a Teacher for Learning?

Watch on ▶ YouTube
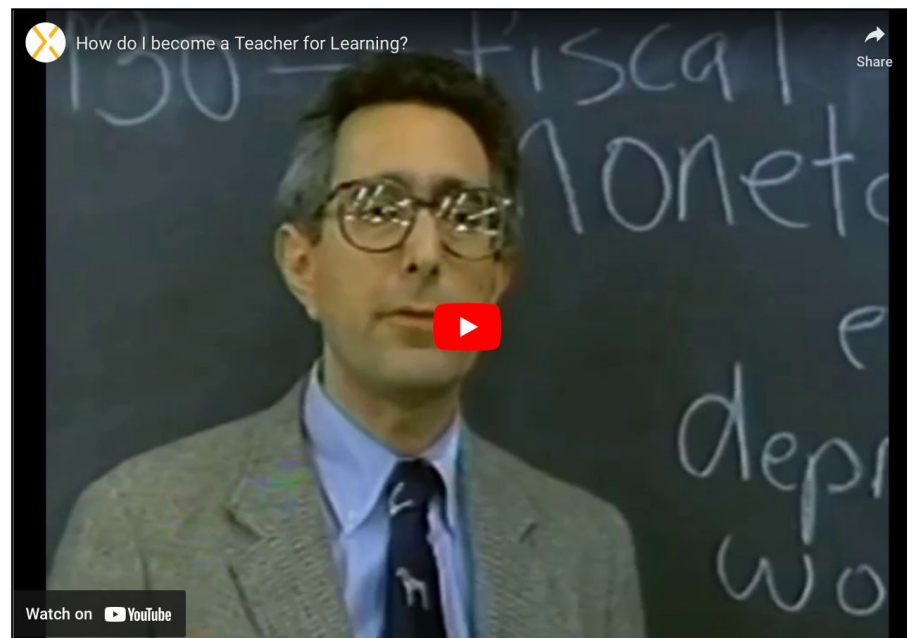
Direct Link: [How do I become a teacher For learning?](#)

Product ⌄ Solutions ⌄ Open Source ⌄ Pricing

Sign in   Sign up

harlows / **waikato-extend** Public

⊘ Notifications   Fork 0   ★ Star 2

<> Code   ⊙ Issues   ⑂ Pull requests   ⊙ Actions   ▦ Projects   ⊙ Security   ⊘ Insights

⊟ Files

waikato-extend / **docs** / **teacher-for-learning** /

⎇ main

🔍 Go to file

⌄ 📁 docs

  › 📁 collaborator

  › 📁 curator

  › 📁 experimenter

  › 📁 scholar

  ⌄ 📁 teacher-for-learning

    › 📁 images

      📄 _footer.md

      📄 _sidebar.md

      📄 climate-of-the-course.md

      📄 feedback-and-practice.md

      📄 how-learning-works.md

      📄 mastery.md

      📄 metacognition.md

      📄 motivation.md

      📄 organise-knowledge.md

      📄 overview.md

      📄 prior-knowledge.md

      📄 scenario.md

      📄 summary.md

  › 📁 technologist

    📄 README.md

Name	Last commit message	Last commit date
👤 harlows Update _footer.md		6fee52a · 3 months ago ⟲ History
📁 ..		
📁 images	Add files via upload	3 months ago
📄 _footer.md	Update _footer.md	3 months ago
📄 _sidebar.md	Update _sidebar.md	8 months ago
📄 climate-of-the-course.md	Update climate-of-the-course.md	3 months ago
📄 feedback-and-practice.md	Update feedback-and-practice.md	3 months ago
📄 how-learning-works.md	Update how-learning-works.md	3 months ago
📄 mastery.md	Update mastery.md	3 months ago
📄 metacognition.md	Update metacognition.md	3 months ago
📄 motivation.md	Update motivation.md	3 months ago
📄 organise-knowledge.md	Update organise-knowledge.md	3 months ago
📄 overview.md	Update overview.md	3 months ago
📄 prior-knowledge.md	Update prior-knowledge.md	3 months ago
📄 scenario.md	Update scenario.md	3 months ago
📄 summary.md	Update summary.md	3 months ago

Spntg 2023

A CSS Primer

This is the smallest, shortest CSS primer you'll ever see.

Basic syntax (note the punctuation):

A CSS file is a collection of declarations about what **things** (er, content objects) look like. We say which thing we mean by using a "selector" and then we tell it what we want by specifying "attributes" and "values". The punctuation is necessary: it basically goes like this:

```markup
selector { attribute: value;  }
```

E.g.,

```markup
h1 { font-size: 18pt; }
```

The **selector:** specifies the thing that we're going to talk about. It can be an HTML **element** like `p` or `h1` or `blockquote`

```markup
p
```

It can also refer to specific kinds of those elements, where they have named **classes** (as in `<p class="whatever">` . Or where they have a unique identifier:

```markup
p.whatever

p#uniqueID
```

You can also specify that the thing only counts if it appears in a particular context. For instance, `p` only if it appears within a `section` element, but not otherwise:

```markup
section > p
```

Resetting the test student will clear all history for this student, allowing you to view the course as a brand new student.

Reset Student | Leave Student View

Product
Solutions
Open Source
Pricing

Search or jump to...

Sign in
Sign up

jmaxsfu / pub607-23   Public

Notifications
Fork 1
Star 3

<> Code
Issues
Pull requests
Actions
Projects
Security
Insights

main
1 Branch
0 Tags

Go to file
<> Code

jmaxsfu Added zoom recorder image   ✓   3402f8f · last year   198 Commits

📁 Alice	Update AW-3.txt	last year
📁 Other	Update Bigoverview2022.md	last year
📁 img	Added zoom recorder image	last year
📄 .gitignore	get rid of DS_store	last year
📄 1.Files.md	Update 1.Files.md	last year
📄 2.Production.md	Removed microphone from Lecture2	last year
📄 3.Ebooks.md	Update 3.Ebooks.md	last year
📄 4.Audio.md	Update 4.Audio.md	last year
📄 Assignments.md	Update Assignments.md	last year
📄 Audacity.md	Create Audacity.md	last year
📄 CSS.md	tweaks	last year
📄 CSSspec.md	Added CSS examples	last year
📄 Commands.md	Pandoc tutorial details	last year
📄 DAW.md	Added zoom recorder image	last year
📄 DownloadPage.md	Update DownloadPage.md	last year
📄 Github.md	Minor edits to Github how tos	last year
📄 GithubPages.md	Minor edits to Github how tos	last year
📄 Pandoc.md	Update Pandoc.md	last year

About

This is material for PUB 607, a graduate project course in Publishing production technologies at SImon Fraser University. This course is new as of spring 2023.

🔗 jmaxsfu.github.io/pub607-23/

📖 Readme
∿ Activity
☆ 3 stars
👁 1 watching
⑂ 1 fork

Report repository

Releases

No releases published

Packages

No packages published

Contributors   2

jmaxsfu John Maxwell

Shaleeta

Languages

● CSS 100.0%

My blog 💬 0 + New Delete Cache LLAR TRU Writer Options

Dashboard
Writings
Media
Pages
Comments
Appearance
Themes
Patterns
Customize
Widgets
Menus
TRU Writer Options
Import/Export TRU Writer Options
Theme File Editor
Plugins
Users
Limit Login Attempts
Tools
Settings
Collapse menu

TRU Writer Documentation

Settings | Documentation | Import/Export

The most current TRU Writer documentation is displayed below (view in a new window). Generated with Docsify This.

TRU Writer Documentation

- Setting Up TRU Writer
- Customize the Writing Form
- Shortcodes
- Optional / Suggested Add-ons
- Updating The TRU Writer theme
- Got Questions

TRU Writer Documentation

These is a reference for the settings and making use of other WordPress features in the TRU Writer SPLOT WordPress theme

Setting Up TRU Writer

Once you have installed TRU Writer and can see the barebones theme staring back at you or maybe dome cheesy demo content.

It is not very interesting. Yet. Get rid of any "Hello World" posts, say bye bye. Set your TRU Theme Options (see below). Set up a menu. Try using the Write page to create some initial content. Poke around the Customizer settings. Fiddle with your footer widgets.

Recommended Plugins for TRU Writer

While your doing all that uploading, you should know that this theme uses the Reading Time WP plugin to insert those commonplace estimates of reading time -- blame Medium if you hate knowing how long a post might take to read.

Page Setup

This theme has one special page for your writing form that must be created; associated with a specific template that provides it's functionality. Activating the theme *should* create the page for you when the theme is activated, but if not, create them as described below. You can edit the content of the **Write** page to customize the welcome seen by writers on your site.

If the theme does not do so automatically (and it should) create this Wordpress **Page**.

- **Write** -- The page that provides the writing form, see http://splot.ca/writer/write. Whatever you include in the body (not required) is added to the top of the form, maybe for extra instructions e.g. for a site at

Product ▾ Solutions ▾ Open Source ▾ Enterprise ▾ Pricing

Search or jump to...   /

Sign in   Sign up

▢ cogdog / truwriter   Public

🔔 Notifications   ⑂ Fork 11   ☆ Star 13   ▾

<> Code   ⊙ Issues 2   ⍁ Pull requests   ⊙ Discussions   ⊙ Actions   ▦ Projects   📖 Wiki   ⊙ Security   📈 Insights

ᛏ master ▾   ⑂ 2 Branches   ⬚ 17 Tags

🔍 Go to file

<> Code ▾

About

Wordpress theme for elegant web publishing without logging in

📁 cogdog Permalink Fix ⋯		57fd7cb · 6 months ago	🕐 **226 Commits**
📁 css	Doc docsified		2 years ago
📁 data	Fixed Widget Export File		3 years ago
📁 images	Link fix		last year
📁 includes	Permalink Fix		6 months ago
📁 js	Tags Suggestion		4 years ago
📄 .gitignore	.DS_Store banished!		8 years ago
📄 LICENSE	Initial commit		10 years ago
📄 README.md	Permalink Fix		6 months ago
📄 archive.php	Archive Vars		2 years ago
📄 class.truwriter-theme-options.php	Permalink Fix		6 months ago
📄 comments.php	Comment Display Fix		3 years ago
📄 content.php	Parent Theme Changed		6 years ago
📄 docs.md	Permalink Fix		6 months ago
📄 examples.md	Author features Docs Reorg		2 years ago
📄 footer.php	Classier Footer		5 years ago
📄 functions.php	Authorless SPLOT		5 years ago
📄 header-write.php	Menu Fixes, Comment Option Enabled		4 years ago
📄 header.php	Menu Fixes, Comment Option Enabled		4 years ago

📖 Readme
⚖ GPL-2.0 license
⟲ Activity
☆ 13 stars
👁 5 watching
⑂ 11 forks

Report repository

Releases 16

🏷 **Just a Few More Bugs** `Latest`
on Oct 12, 2022

+ 15 releases

Packages

No packages published

Contributors 4

👤 **cogdog** Alan Levine
👤 **davidnwright** David N. Wright
👤 **grantpotter** Grant Potter
👤 **cmadland** Colin Madland

Languages

5. Hosting Markdown Files

Hosting Markdown Files

- GitHub or Codeberg
  a. Signup for a GitHub or Codeberg account
  b. Make a new repository, and create or upload your Markdown file
  c. View the Markdown file, and copy/paste that URL into the Docsify-This Markdown File URL field
- Gist (GitHub Gists)
- Personal or Organizational Webhost
- Almost any static file hosting service
- *Or just about anywhere a raw source Markdown file can be accessed!*

Time for Questions & Discussion

- What we've covered in this section
  - "In the Wild" Examples
  - Hosting Markdown files
- Coming up
  - Live demo
  - *And then online activities!*

6. Live Demo

7. Online Activities!

bit.ly/otessa24-docsify-this

Breaking Down Barriers:
Using Docsify-This As an Entry Point to Markdown Open Publishing

Materials for Paul Hibbitts' [OTESSA 2024 Conference](#) online workshop about the open source project [Docsify–This](#).

- [Interactive Worksheet](#) ([PDF](#))
- [Introduction to Docsify-This](#) ([PDF](#))
- [Google Slides](#) ([PDF](#))

©iStockphoto.com/xijian

Markdown export from Google Docs courtesy of the [Docs to Markdown](#) Google Drive extension. These materials are also available on [Codeberg.org](#).

All text of this work at [bit.ly/otessa-2024-docsify-this](#) by Paul Hibbitts is licensed under [CC BY 4.0](#) 🅭🅯

📁[View on GitHub](#)

Displayed by [Docsify-This](#), based on the [Docsify Open Publishing Starter Kit](#).

Next Steps

Markdown Guide
- ■ markdownguide.org

Improving The Accessibility Of Your Markdown
- ■ smashingmagazine.com/2021/09/improving-accessibility-of-markdown

OER on GitHub What, Why, & How (also applies to Codeberg)
- ■ evanwill.github.io/make-oer

Markdown Desktop Editors
- ■ code.visualstudio.com
- ■ typora.io

GitHub Desktop App (can also be used with Codeberg)
- ■ desktop.github.com

Thank You! Questions?

Contact info
- Web: hibbittsdesign.org
- Email: paul@hibbittsdesign.org

OTESSA 2024 online workshop materials
- bit.ly/otessa24-docsify-this (incl. these slides)

Docsify-This
- Web App: docsify-this.net
- Open Publishing Ecosystems - Docsify This: youtube.com/watch?v=EjeqxOJ2ZxE

Want to host your own Docsify website? Check out my Docsify Starter Kits!
- github.com/hibbitts-design