

iGEM Wiki Web Workshop 1



Jonas Johansson
Jul 10 · 11 min read

Hi, I'm Jonas Johansson, a developer and designer. As a member of the Stockholm iGEM team, part of my responsibility is to *work* the wiki. This post is an extension of a session I held within our team and will go through the following:

1. Background of the wiki
2. How to create and edit a page
3. How custom styles are implemented
4. Why it's a great idea to use a text editor before updating the wiki.
5. Why it's a great idea to use a Markdown tool for bridging the gap between content preparation and content presentation.
6. Why debugging is important and how you can do it without leaving the browser.

To learn about how you can increase your understanding of web technologies and create your own personal digital space read my second post.

With that said, let's start at the beginning...

What is a wiki? 🙋

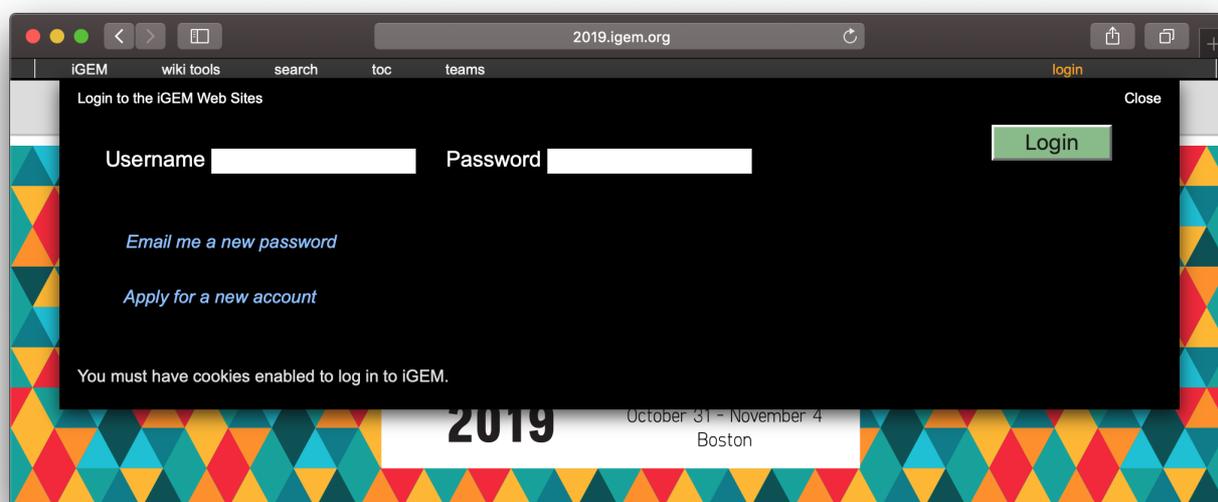
A wiki can be understood as a library where, in our case, each team has their own book which only they can edit (but others can read). Within that book, there's pages and references, in web lingo these references are “hyper references” or simply, “links”.

A website or database developed collaboratively by a community of users, allowing any user to add and edit content.

Technically speaking it's similar to a CMS (Content Management System), a term that you might have heard before as there are other CMS out there, such as Wordpress. What signifies a wiki is that all registered users can edit the content, by default, and it follows a streamlined architecture of pages and links. For the end viewer, the wiki is *just a website* that runs on traditional HTML (content), CSS (style) and Javascript (function), the technical components that make all web experiences.

Getting started

Visit 2019.igem.org and log in using the button found top right in the menu. In order to access the wiki, you must be a registered user, check-in with your team if you have yet to receive the proper credentials.



If you can't see the "login" button, increase the width of your browser window and it should appear.

By logging in you will be granted access to edit existing pages and to create new ones, but only within the team which your account is connected to, Stockholm in my case.

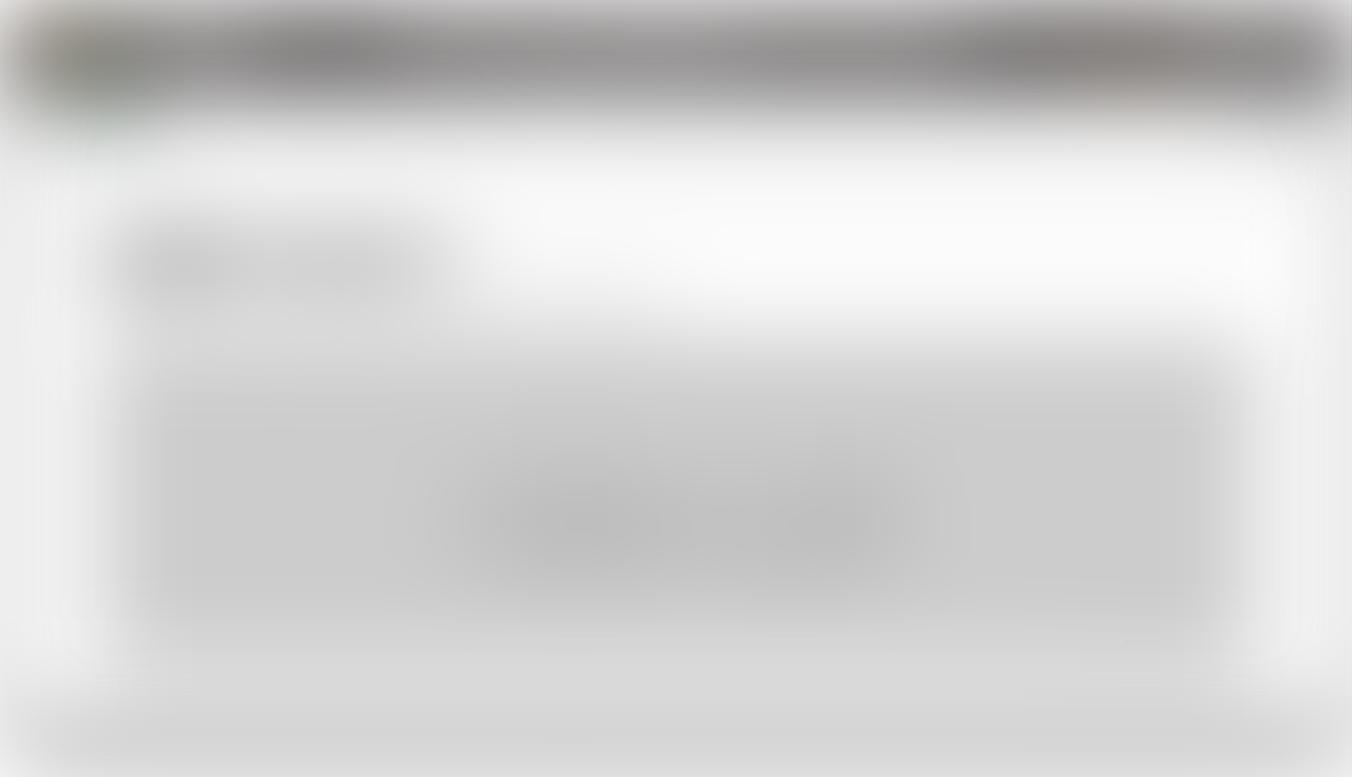




Good job! Well, no point in lingering, let's boogie down to the nitty-gritty.

Editing the page!

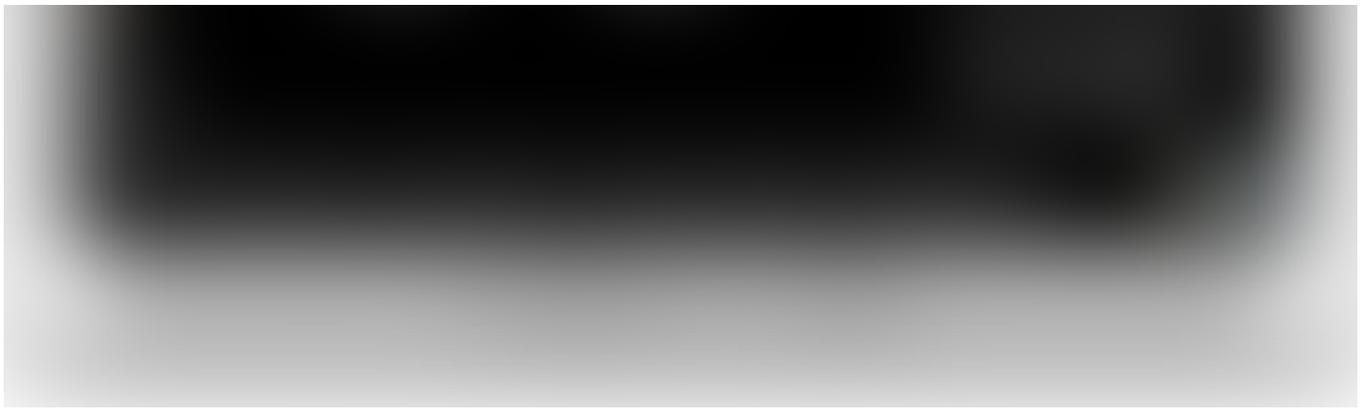
Each team has its own main page which can be found at 2019.igem.org/Team:Stockholm. Replace **Stockholm** with your team identifier.



As you can see there is already default content provided by iGEM here. Hover over “wiki tools” in the top left and click “Edit”. This will open the wiki editor for this particular page.

You can also add “?action=edit” in the URL so instead of 2019.igem.org/Team:Stockholm you would type 2019.igem.org/Team:Stockholm?action=edit.





You've now arrived at the iGEM wiki editor! So, what's going on here?



While I will try to not be overly critical of the wiki, one large downside is that all “web code” is in a text box that does not contain line numbers nor syntax highlighting. Line numbers are helpful as it helps communicate wherein the code attention is required. Syntax highlighting is a visual aid that helps the editor quickly scan through the contents, it's also extremely useful for debugging ie. finding and solving errors. *I will return to this shortly.*

```
{{Stockholm}}
```

```
<html>
```

The first line refers to a “template” named Stockholm. *Ignore this for now.* The next piece of content is the beginning of our HTML (HyperText Markup Language) document.

What happens in <html> stays in </html>

The <html> tag signifies the beginning of an HTML document. The way it is written also informs us that it is the “opening tag” meaning that whatever is written after is nested inside, making <html> the parent. Words such as parent, child, sibling etc. are very common in web development.

If you were to scroll down to the very bottom of the box you would find the “closing tag” which is represented with a forward slash after the left arrow.

```
</html>
```

Directly after <html> comes <hr> which stands for Horizontal Rule and when used adds a horizontal border.

```
<hr>
```

This tag along with several others are “self-closing” meaning they can not be parents to other tags, they both open and close!

```
<div class="column full_size">
```

Following is a <div> tag which contains additional information. There’s an attribute named “class” (as in classification) which has the value of “column full_size”. The class is useful in our template when we use CSS (Cascading StyleSheet) as we can then pinpoint how specifically this tag should be visualised.

I will not go into much detail how this actually works in this article, other than briefly mention it here and reference it later. Please look at the list of resources at the end for further reading.

There are many other tags to learn, those that you will find if you continue scrolling in the text box are:

- h1,h2,h3(differently sized headings)
- p (for paragraphs)
- ul, ol (for unordered lists and ordered lists)
- a (for links)
- img (for images)

A favourite of mine is the old classic `<marquee>` tag, why don't you try it out? Add the snippet below into your page and save.

```
<marquee>I was cool in the 90s!!</marquee>
```

I did it my way...

I will not make it a secret that I personally do not edit directly in this box when I make changes. Instead, I keep a separate file which I edit in my favourite text editor Sublime. When I've made the edits that I like, offline, I simply copy all of it and paste into the iGEM wiki text box. I then click "save page" at the bottom.

As a precaution, I advise that you copy all the contents of the text box and save it in a text file, just in case something goes awry.



Sublime editor with line numbers, syntax highlighting and intending

This file that you just saved, in my case *test.html* can be opened in any browser, simply drag and drop it and you will see it's contents displayed. Whenever you change the contents of the file you may refresh the browser, try the code below and see what happens!

```
<h1>Hello iGEM world!</h1>
```

I got 99 problems but creating a page ain't one

You should now understand how to edit your team's main page. You may not understand the HTML involved but that's where the fun part and learning arrives!

But let's say you want to create a new page. It's actually quite simple, you only need to add the name of the page preceded by a forward slash, for instance, 2019.igem.org/Team:Stockholm/About.



If no page exists at this location you have the opportunity to “edit this page”, which is at the end of the message. It will take you to the editor site that you saw previously. This one works just like the previous one, except that it is blank, so go ahead and play around and once you save it it's public and can be viewed by anyone!

Let's look at how we can change the visual representation of all this content.

Now sissy that walk!

Remember the `{{Stockholm}}` at the top? That refers to a default template which you can also see is referenced at the bottom of the edit page where it says “Templates used in this page”.

You can find your template by going to 2019.igem.org/Template:Stockholm which will not show much, but add the `?action=edit` and voila! Obviously, replace Stockholm with the name of your default template.

It's important to know that whatever changes you make to the template there is a backup which you can find here: 2019.igem.org/Template:Example (you have to go into “edit” mode in order to see the contents).

Instead of editing the default template and taking the risk of accidentally deleting something important, I encourage writing a new template and including it on the pages that should use it. You follow the same procedure as for a page. In my case, I created the template “Stockholm/CSS” by going to 2019.igem.org/Template:Stockholm/CSS. I could then include it by typing `{{Stockholm/CSS}}` on the pages that should use it. **Don't forget this step, otherwise, you won't see your updates take action.**



Since templates are also HTML pages they must have `<html>` at the top and `</html>` at the bottom. In my example here I am also doing a big no-no by linking an external file. The correct way is to upload files and then reference them internally, read about that in the [iGEM Wiki Editing Help](#).

Don't know CSS? It's a fair amount of learning... But it's very fun! For instance, if we want to make all the `<p>` tags red we can add (into the Stockholm/CSS template):

```
p { color: red; } // this will make all paragraphs red!
```

In CSS you write the name of the tag followed by opening curly brackets, the actual style data, and finally closing curly brackets. Most tags actually come with pre-defined style data, and all browsers handle this slightly different. For instance, a website may look one way in Safari and another in Chrome.

```
.column { width: 400px; }
```

Remember that “column full_size” from before? We can reference specific classes by typing a dot followed by the name of the class, in this case, we can choose from either “column” or “full_size”. It’s up to you to decide on the names when writing the HTML. You can then, in CSS, say that all tags with the class of “column” should be 400 pixels wide.

The point here is not for you to understand all the aspects of this, but realise that there is indeed an architecture in place.

Just like tags, there are many CSS attributes to learn, here’s a few of the basic:

- background, color
- font-family, font-size
- width, height
- margin

Prototyping the wiki

It’s hugely inefficient to use the wiki system to test out one’s content, even a minor change could take several seconds to be implemented and minor changes is really at the core of web development. My suggestion is to try and replicate the wiki structure elsewhere in a way that enables the rest of the team to effectively contribute while keeping it easy to transfer to the wiki. It would also mean that all your work is private for as long as it needs to be.

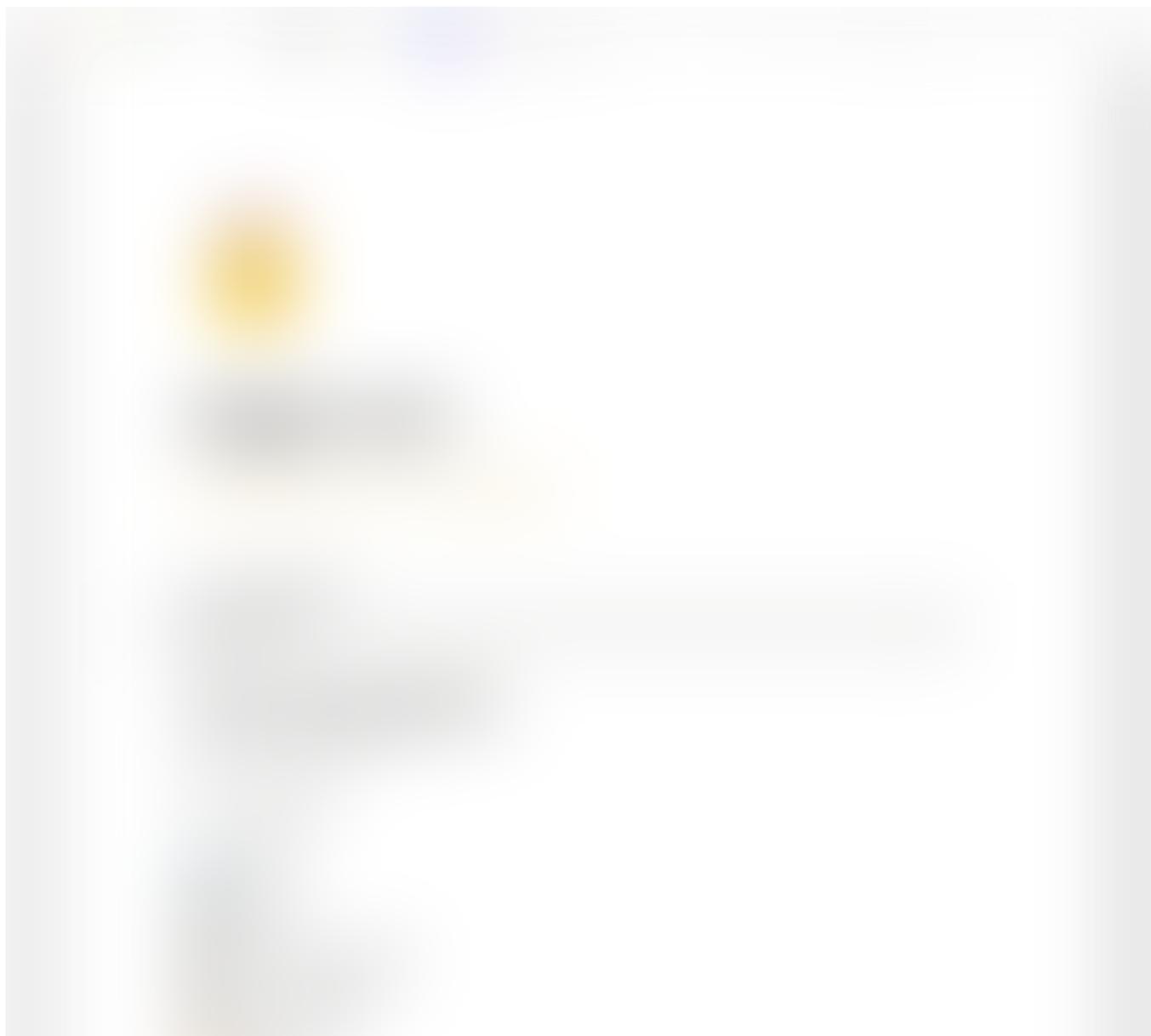
If there was only a way...

Introducing Markdown

Markdown is used in a lot of web services and is a dead-simple syntax for writing rich text. The way it works is that instead of tags you write special characters before and after, like `_italic_` or `*bold*`. Written this way they will be presented like *italic* and **bold** to the reader. You can try it yourself in Facebook Messenger for instance.

You could create a Google Doc structure that reflects the wiki outline but jumping between different documents disrupts the sense of overview, it's simply too distant in design from the wiki.

We use Notion, a tool that acts as a wiki itself, but with very limited design options. I will let you explore it yourself, but what it means to our team is that all members can easily add content, track changes and get an overview. On my end, I can simply convert the Markdown to HTML and paste it in the wiki. This saves us a lot of time whilst also increasing the quality of the end result. Or, at least, that's what we hope.

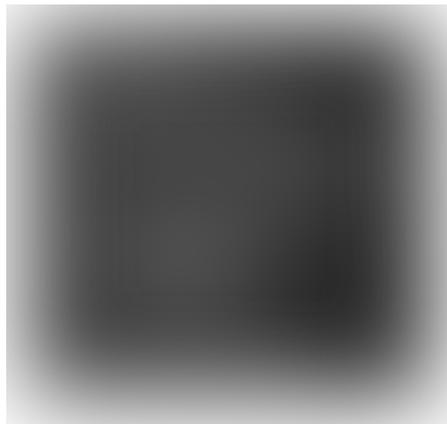




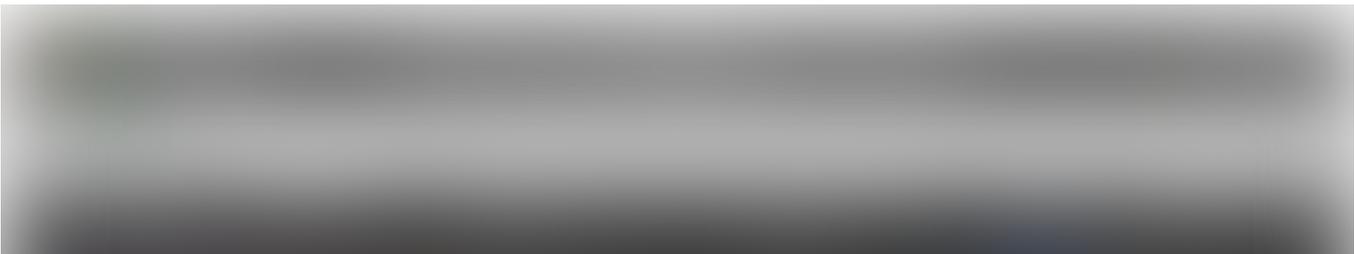
Screenshot of our Notion page for the wiki

Debugging 🐛

You may have heard this term used before (perhaps even in this post!) as someone says that the code is “filled with bugs” or that the software is “buggy”. Everyone makes mistakes, even seasoned developers, so when something does not appear the way you expect it to, it’s likely due to human error. Unfortunately, the wiki does very little to provide assistance but luckily the browser comes packed with great tools which you can make use of. In most browsers, there is a function that allows you to look under the hood of any website and see what’s going on. You can try this out by right-clicking on a page and select “inspect” or “inspect element” from the menu.



Found it? Great. What you will see now is the “backside” of the web site you are visiting. Here you can directly edit the site to test ideas, both for style and for content. Inspect is the greatest debugging tool, besides a great editor, that you can use. It’s perhaps not for the beginner but important to be aware of nonetheless.



Bonus 🎉 Password protect your wiki 🔒

You may be in a situation where you want to upload sensitive content to the wiki, to see how it looks. Is there a way to protect it so that other's can't "be inspired"? Sort of, once anything is published in the wiki it's open-source, but we can make it complicated to access.

```
<style>
  div#content.hidden {
    display: none;
  }
</style>
<script>
  var content = document.querySelector('#content');
  content.classList.add('hidden');
  var pw = '1234';
  var access = prompt('Enter password');
  if (access == pw) {
    content.classList.remove('hidden');
  } else {
    window.alert('Wrong password 😞 ');
  }
</script>
```

Add this snippet into any page (after `<html>`) that you think should only be viewed by those who have the password. Remember that by using debugging tools previously mentioned it's possible to find out the password.

Summary

1. Use an external editor for a more efficient workflow, simply copy and paste into the editor.
2. Create your own templates instead of building on top of the existing one.

3. Use a separate system for managing team content contribution and information overview, I am suggesting Markdown as syntax and Notion as the tool.

I hope this was useful and that it can serve as an easy to grasp starting point for continued learning. Find me at jonasjohansson.se and reach out with any questions.

Resources

The US AFRL CarrollHS team have created a series of video tutorials, check them out, as well as a complete wiki guide. Right on!

Wiki

- <https://2019.igem.org/>
- <https://2019.igem.org/Team:Stockholm>
- <https://2019.igem.org/Team:Stockholm?action=edit>
- <https://2019.igem.org/Template:Stockholm>
- <https://2019.igem.org/Template:Stockholm?action=edit>
- https://2019.igem.org/Resources/Wiki_Editing_Help

Learning

- <https://www.superhi.com/> (learn to write HTML, CSS and Javascript!)
- <https://css-tricks.com> (articles on CSS and more)
- <https://alistapart.com/> (articles on web design thinking and more)

Software

- <https://www.sublimetext.com/> (great editor)
- <https://www.google.com/chrome/> (great browser)
- <https://notion.so> (great tool)

