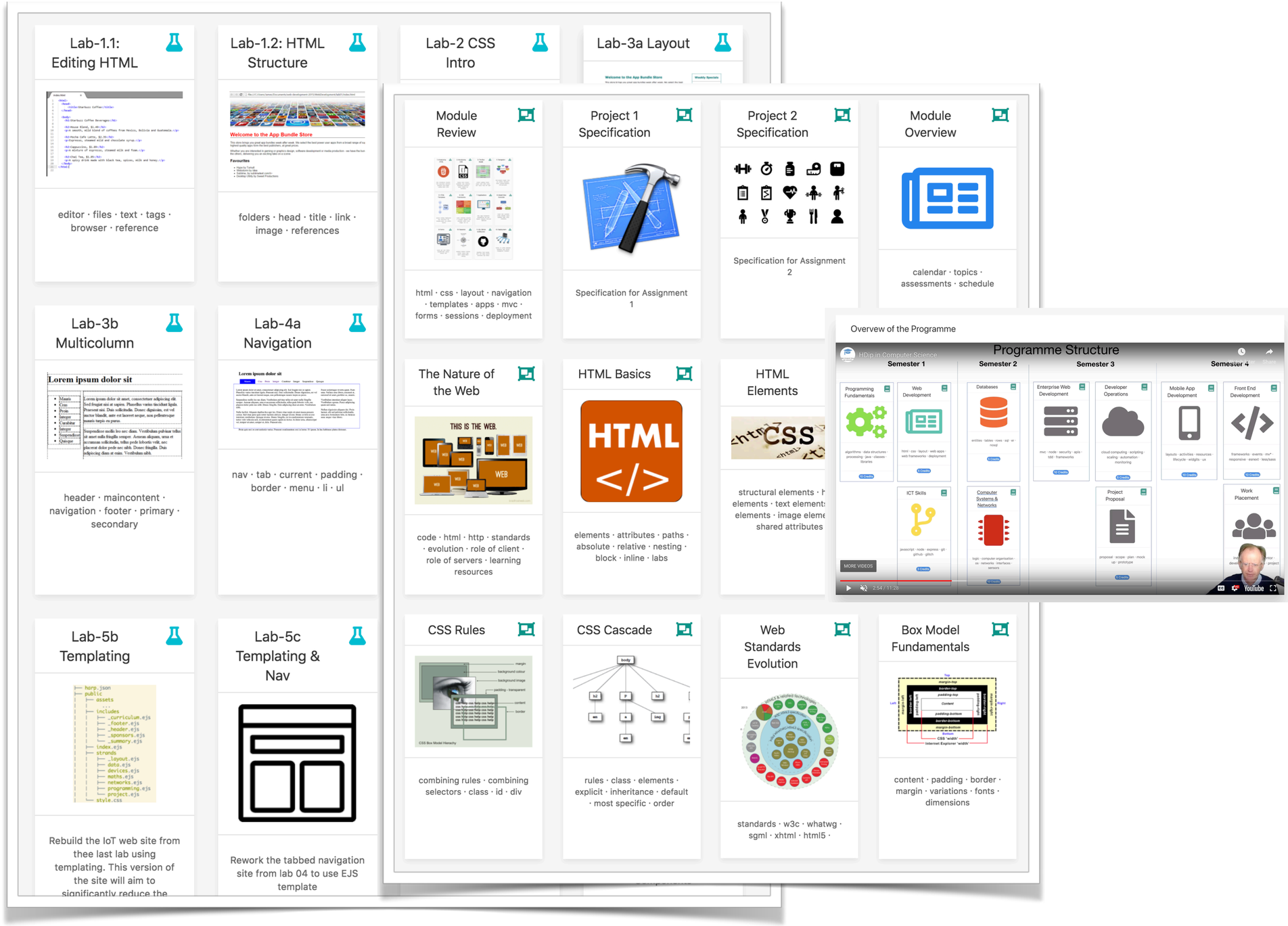
Static Site Generators for Instructional Content



TutorStack

1. Instruction Materials — tutors-ts
2. Community & screen sharing — slack
3. Assessment & Feedback — moodle
4. **Media** — Open Broadcaster Software, YouTube
5. **Broadcast** — Open Broadcaster Software, YouTube

TutorStack

1. Instruction Materials

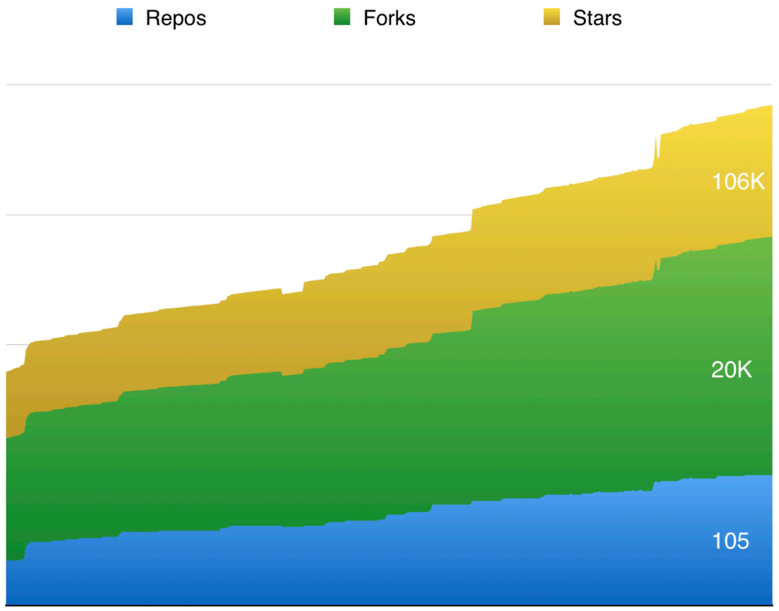
tutors-ts

At StaticGen, our open-source directory of **static website generators**, we've kept track of more than a hundred generators for more than a year now, and we've seen both the volume and popularity of these projects take off incredibly on GitHub during that time, going from just 50 to more than 100 generators and a total of more than 100,000 stars for static website generator repositories.

NOVEMBER 2, 2015 • 99 COMMENTS

Why Static Site Generators Are The Next Big Thing

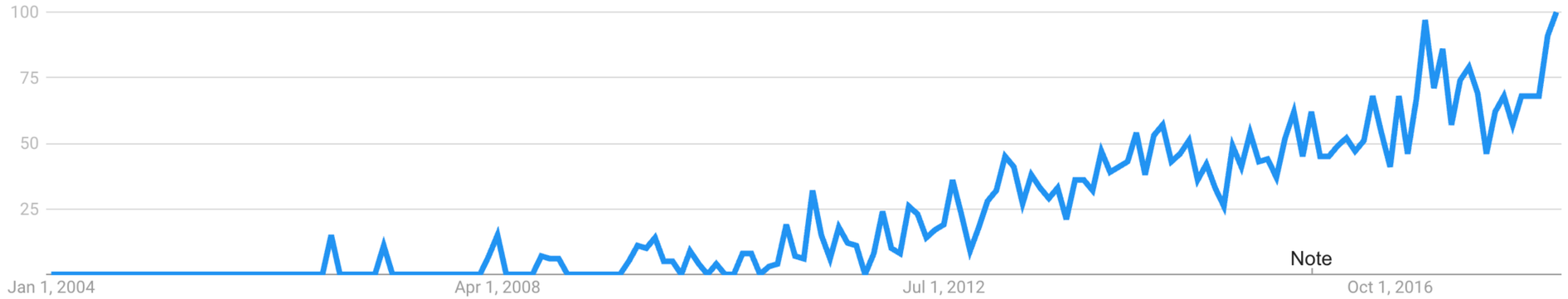
Coding 587 # Tools 173 # Static Generators 9



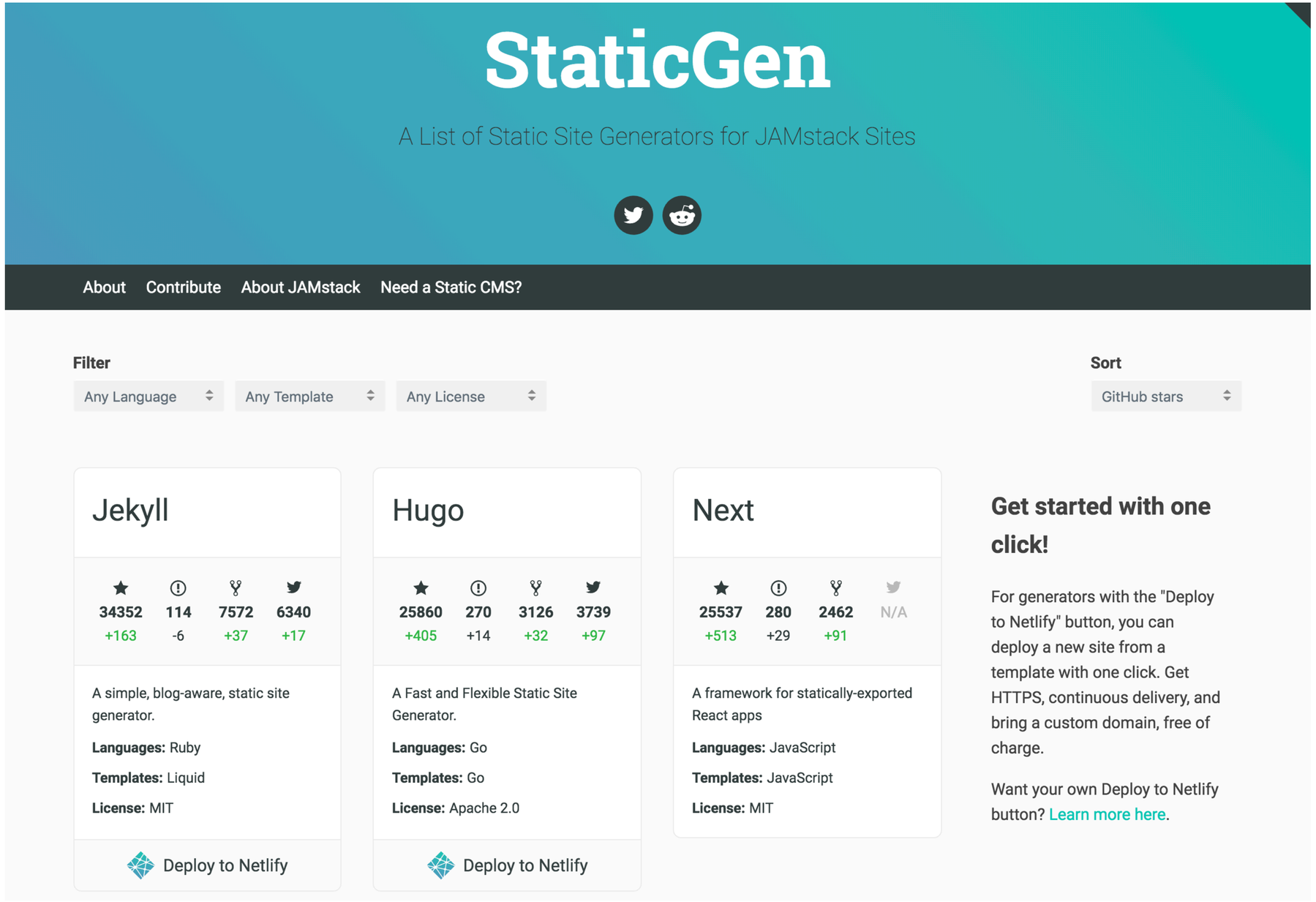
Influential design-focused companies such as Nest and MailChimp now use static website generators for their primary websites. Vox Media has built a whole publishing system around Middleman. Carrot, a large New York agency and part of the Vice empire, builds websites for some of the world's largest brands with its own open-source generator, Roots. And several of Google's properties, such as "A Year In Search" and Web Fundamentals, are static.

https://www.smashingmagazine.com/2015/11/modern-static-website-generators-next-big-thing 2015!

Static Site Generator



Google Trends



https://www.staticgen.com

222 Separate Systems!

Static Site Generator

*“…a hybrid approach to web development that allow you to build a powerful, server-based website locally on your computer but pre-builds the site into static files for deployment”**

*https://wsvincent.com/what-is-a-static-site-generator

Tutors

Eamonn de Leastar (edeleastar@wit.ie)

Setup

Setting up and configuring the tools tutors requires, and tutors itself. These are: git, node.js and the sublime text editor.

Composition

Explore the structure and contents of Labs. Introduce the basics of Markdown and demonstrate the primary features.

Publishing

Publish course to the public Internet using github, and make individual topics available to Moodle.

Portfolios

Aggregating multiple modules into a portfolio.

Example - course with 4 topics

Setup

Eamonn de Leastar (edeleastar@wit.ie)

Introducing Tutors

A quick tour of the purpose, structure and features of a tutors course web site.

Tutors Course Structure

A tour of the structure of a typical tutors course.

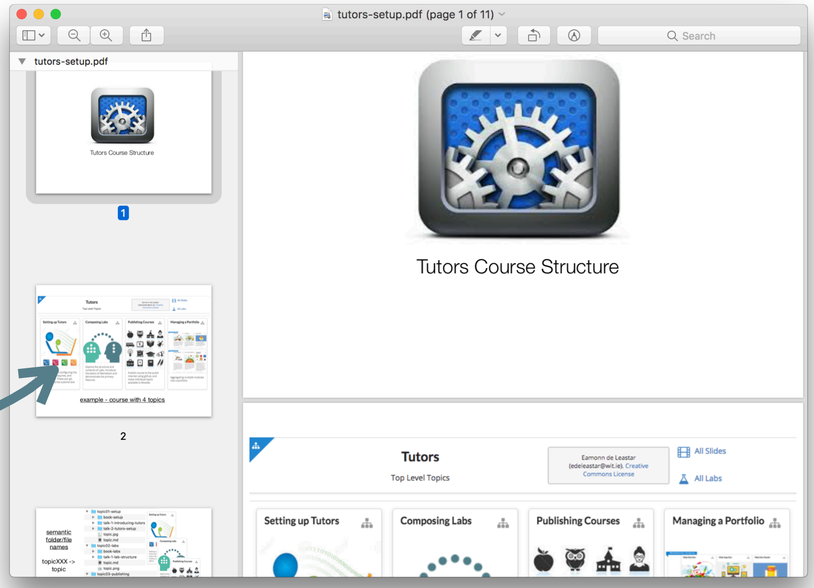
Lab-01-Setup

Install the tutors command line application and take for a first spin.

1 topic = slides + lab

Tutors Course Structure

A tour of the structure of a typical tutors course.



Talk (slides)

Lab-01-Setup

Install the tutors command line application and take for a first spin.

Setup | Lab-01-Setup | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | Exercises

Generate a Course

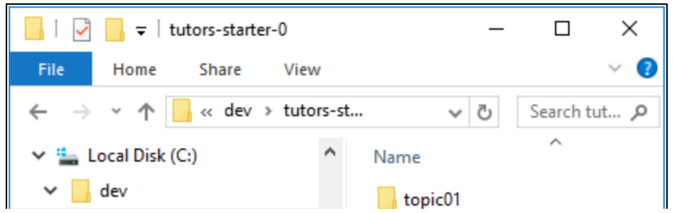
The simplest way to get started with tutors is to use the **new** command to generate a template course:

```
tutors new
```

This should respond with:

```
Creating new template course...
Cloning into 'tutors-starter-0'...
Next steps...
cd into tutors-starter-0 and run "tutors" again
This will generate the course web in "tutors-starter/public-site"
```

The command will have generated a new folder: `tutors-starter-0`, populated with a sample/template course:



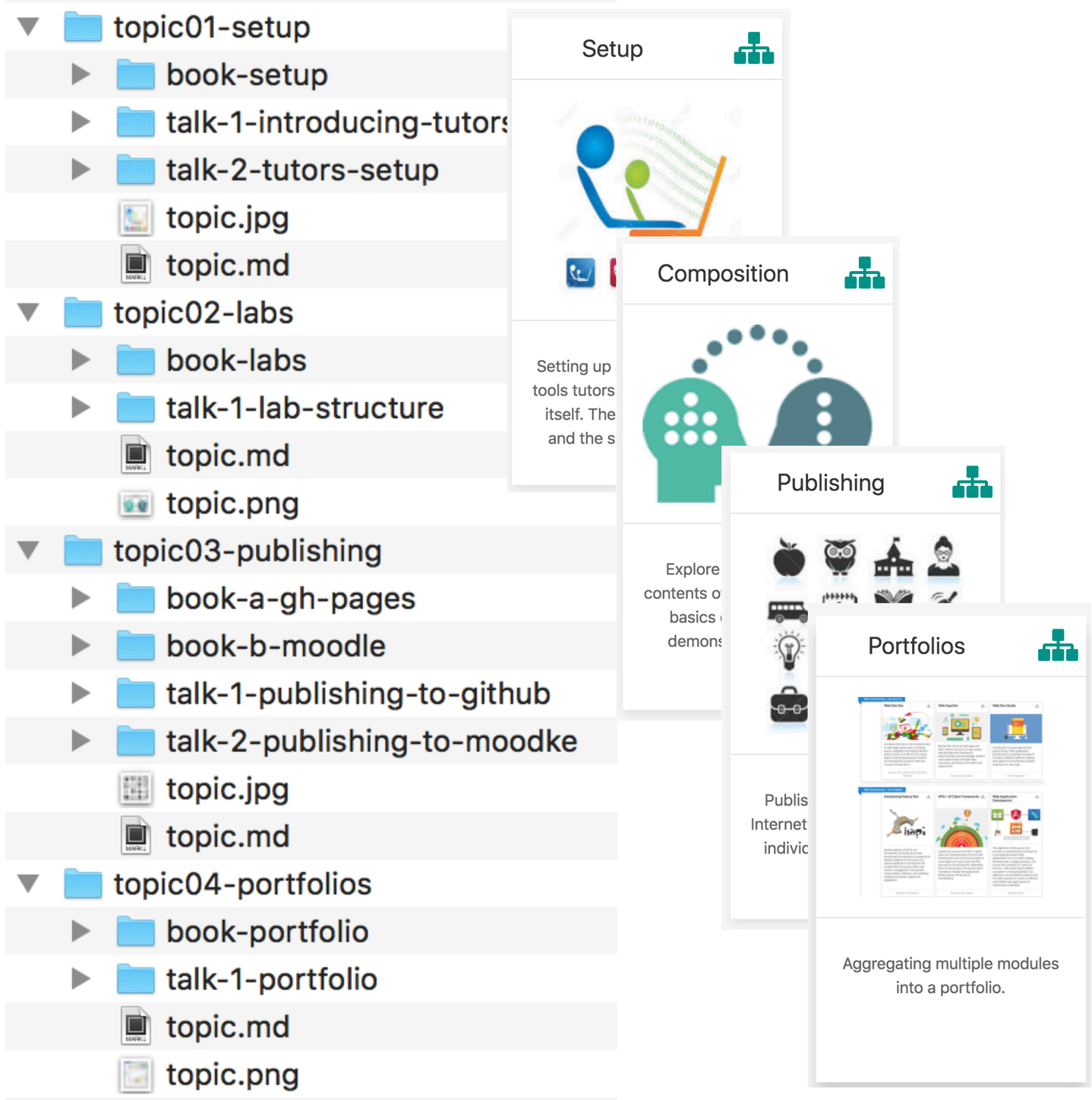
Lab (guides)

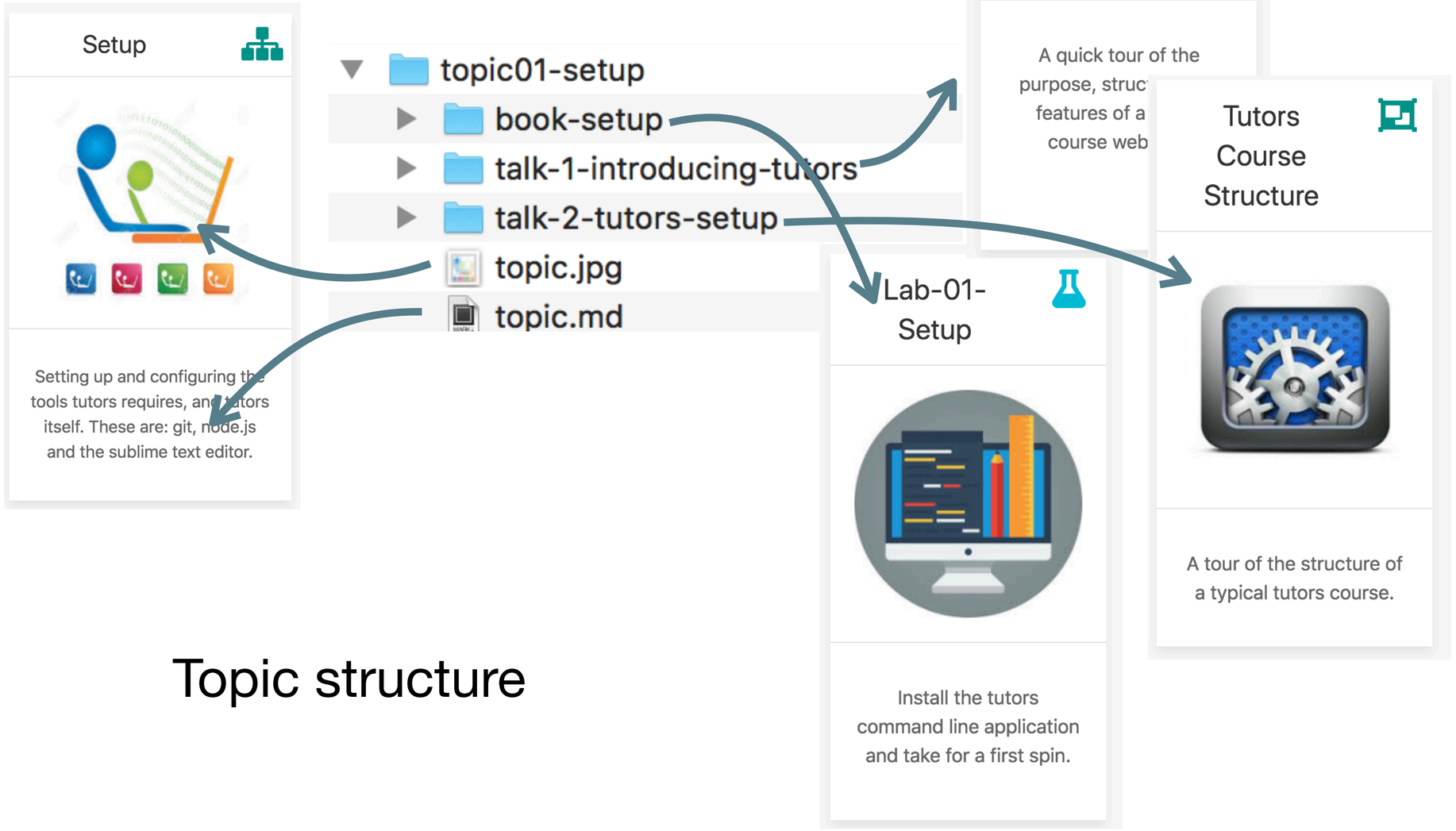
Semantic folder/file names

topicXXX -> topic

bookXXX -> lab

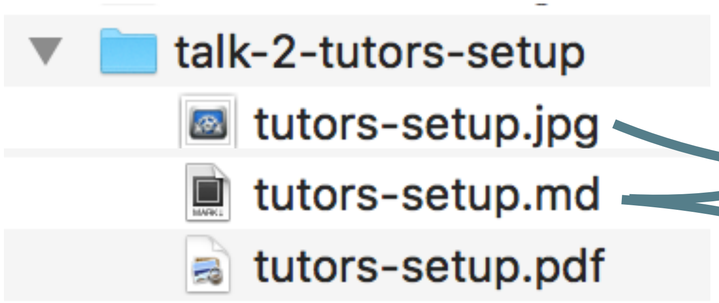
talkXXX -> talk

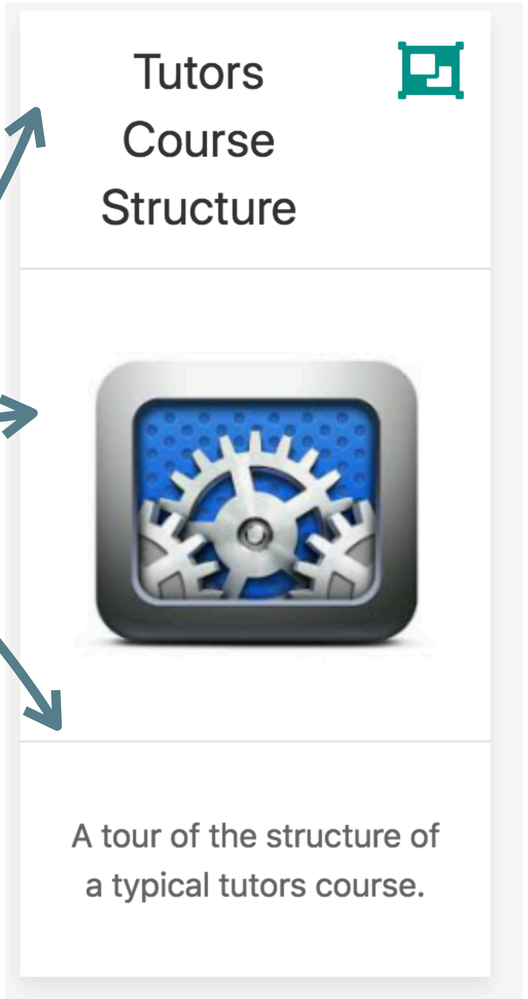




Topic structure

Talk structure



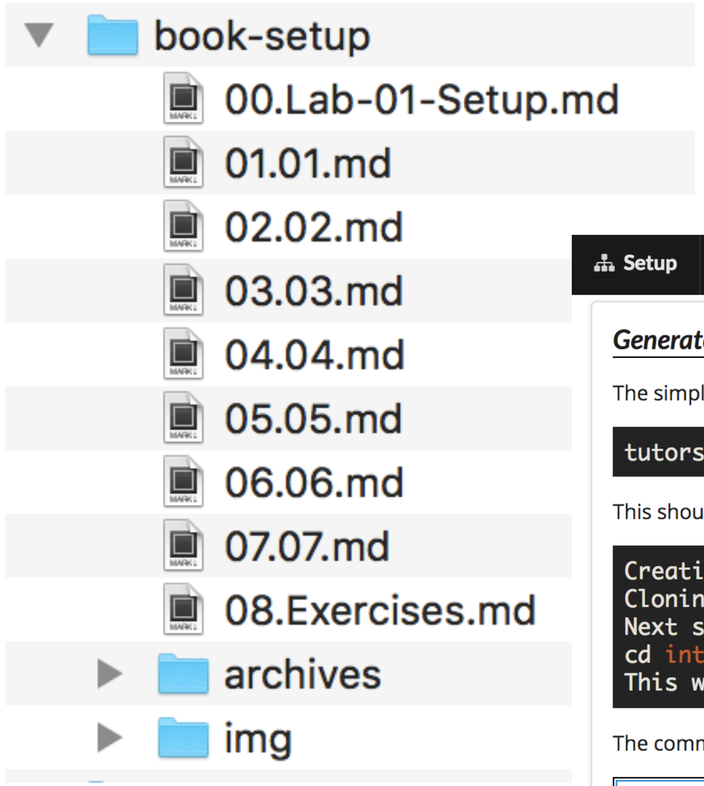


all files
same name

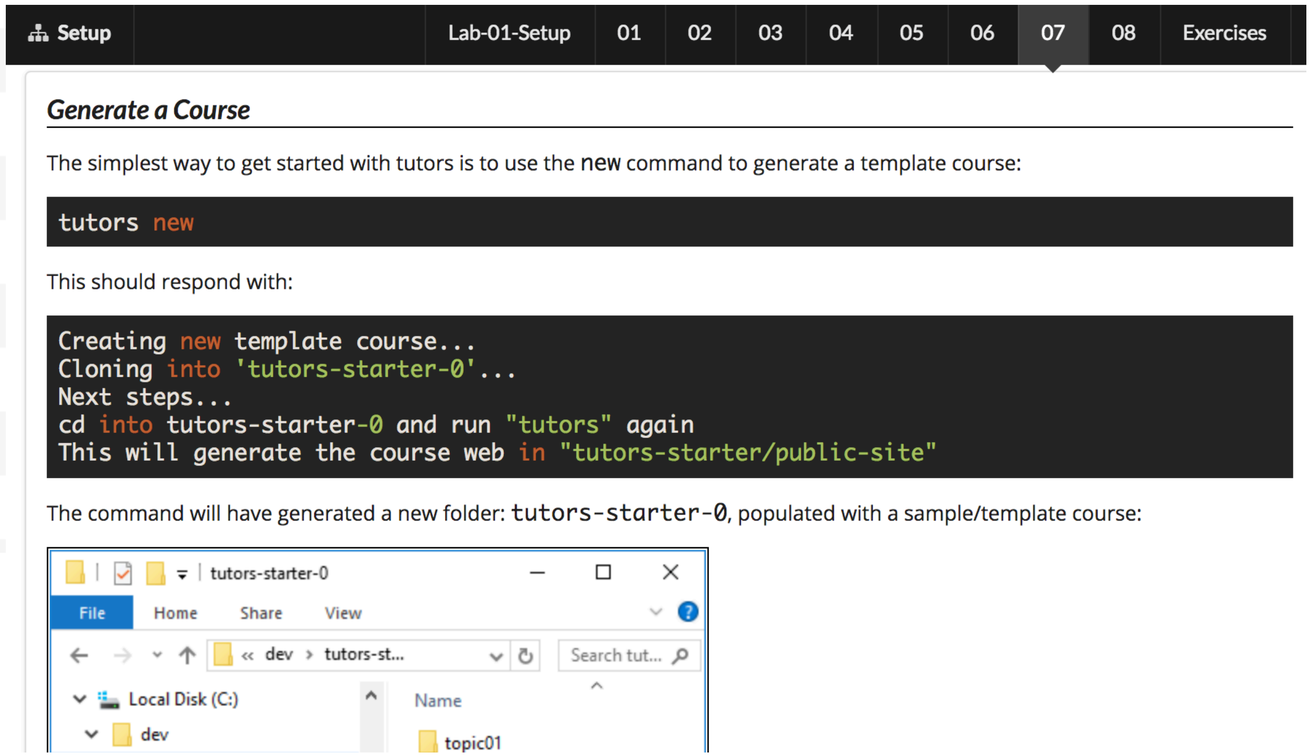
- image
- pdf
- markdown

```
Tutors Setup

Installing the tools needed for
tutors. Taking it for a spin.
```

Lab structure



One *md* file per step



img

- images used in lab

archives

- zipped archive linked to in labs

Setup | Lab-01-Setup | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | Exercises

Generate a Course

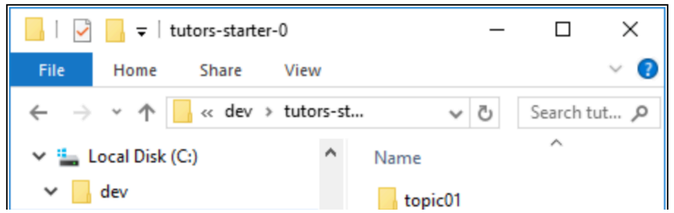
The simplest way to get started with tutors is to use the new command to generate a templa

```
tutors new
```

This should respond with:

```
Creating new template course...
Cloning into 'tutors-starter-0'...
Next steps...
cd into tutors-starter-0 and run "tutors" again
This will generate the course web in "tutors-starter/public-site
```

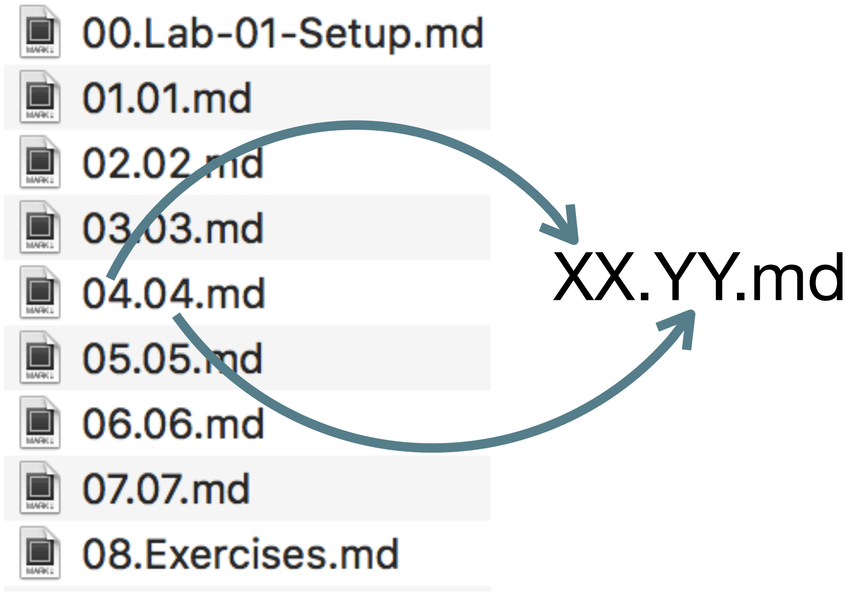
The command will have generated a new folder: `tutors-starter-0`, populated with a sa



Each Lab written in Markdown

```
# Generate a Course

The simplest way to get started with tutors is to use the `new` command to generate a template course:

~~~
tutors new
~~~

This should respond with:

~~~
Creating new template course...
Cloning into 'tutors-starter-0'...
Next steps...
cd into tutors-starter-0 and run "tutors" again
This will generate the course web in "tutors-starter/public-site"
~~~

The command will have generated a new folder: `tutors-starter-0`, populated with a sample/ template course:

![](img/09.png)
```

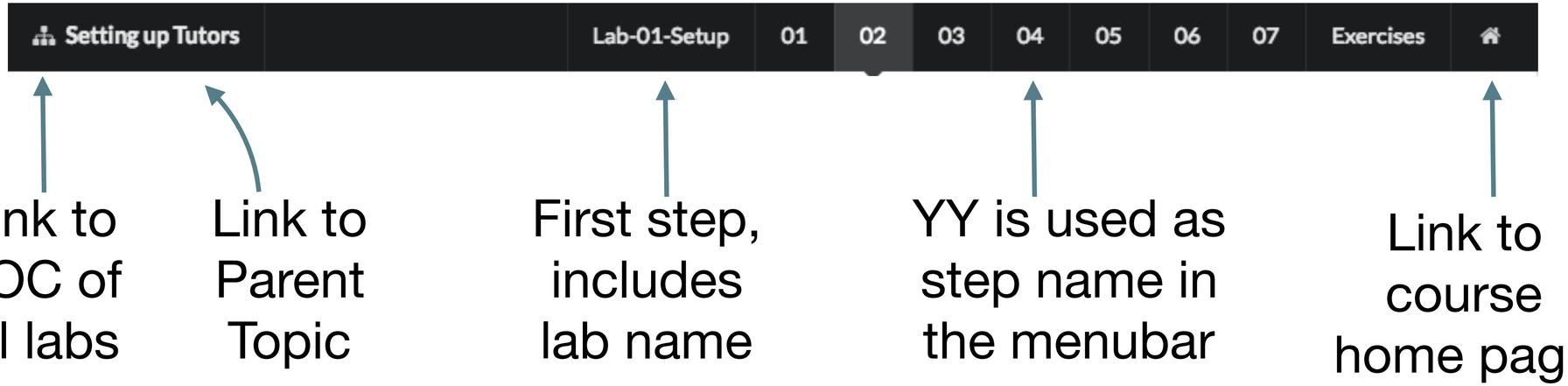
lab menu bar



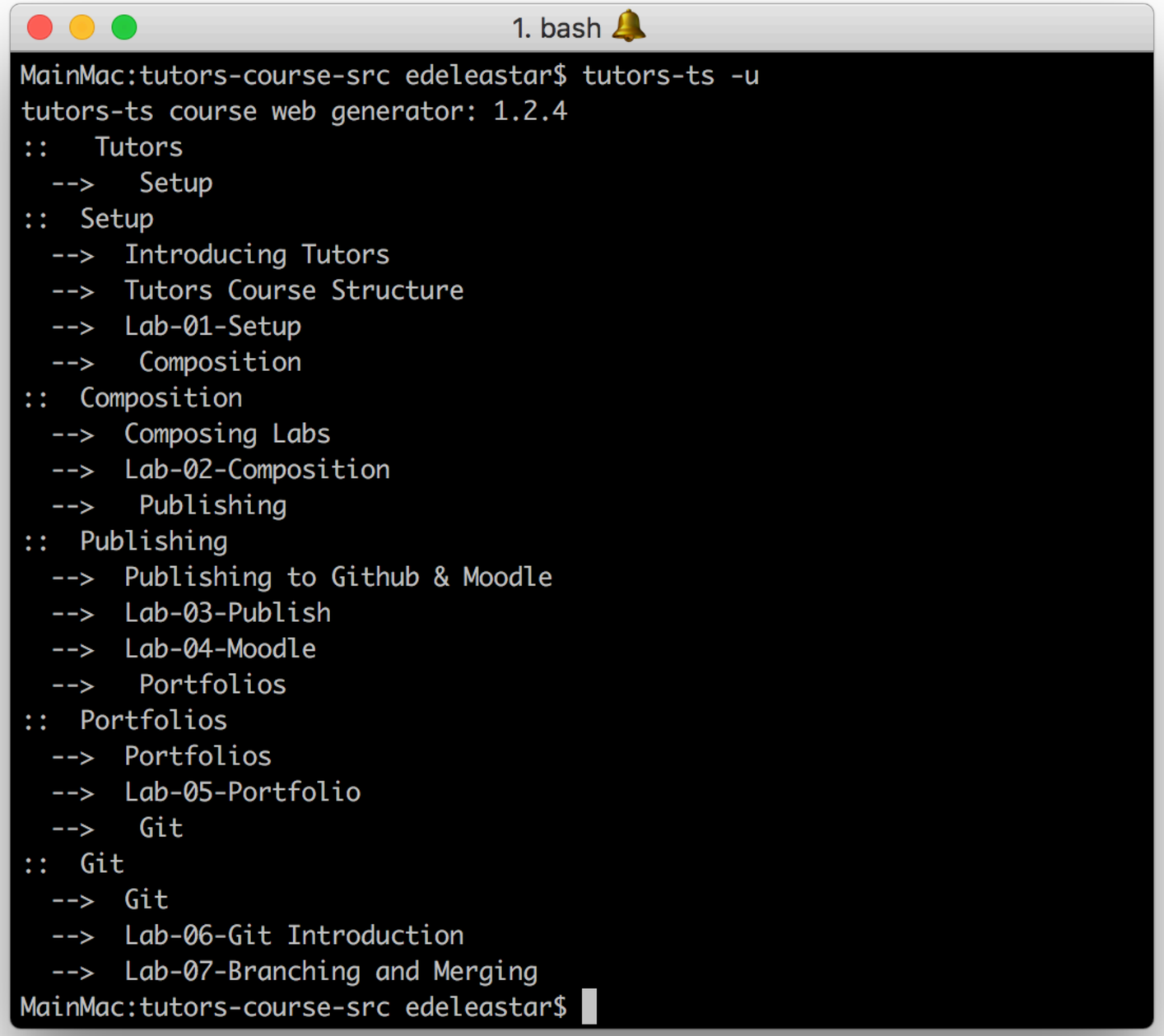
XX
- must be a 2 digit number

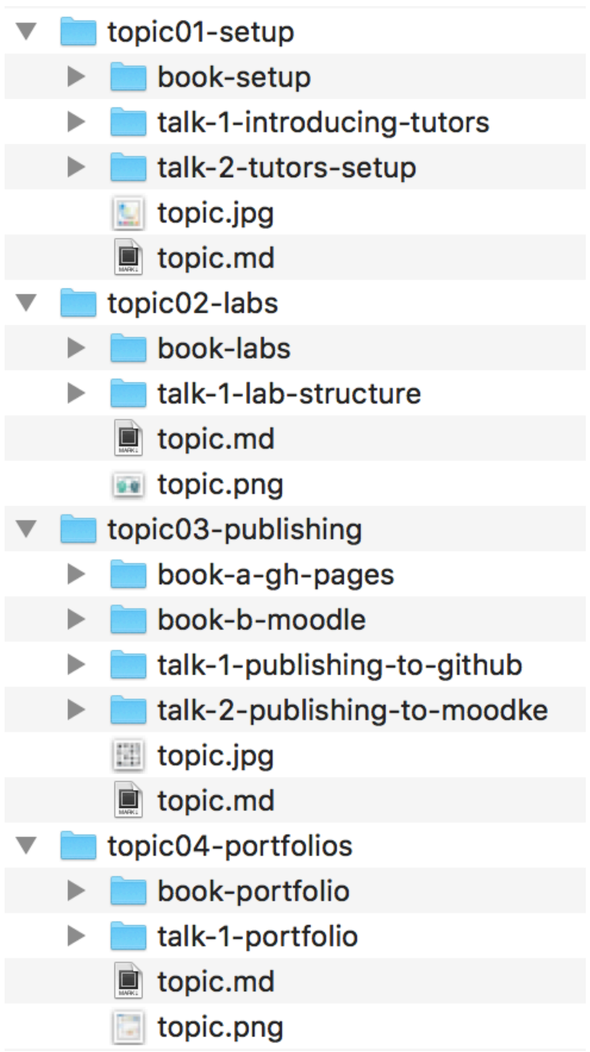
YY
- can be any string

.md
- must me *.md*



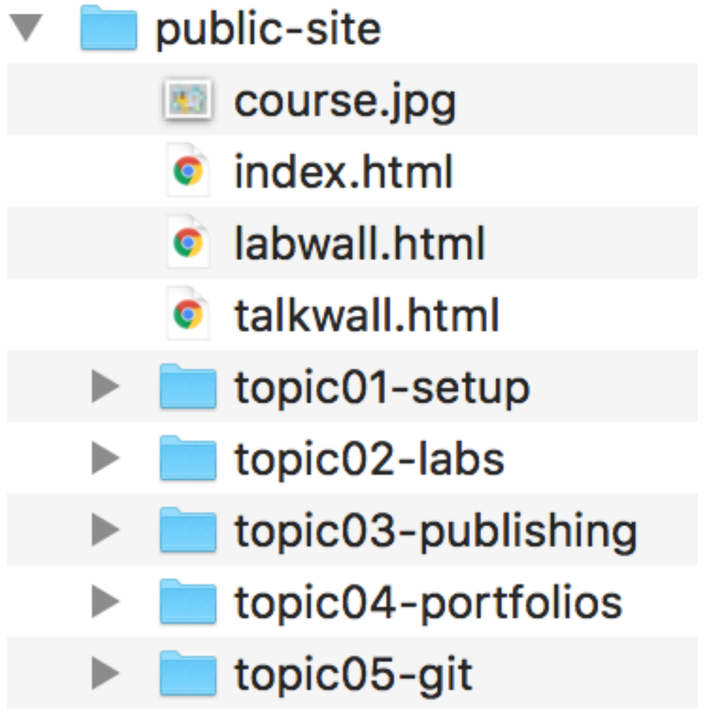
‘tutors’ command generates to ‘public-site’ folder





tutors

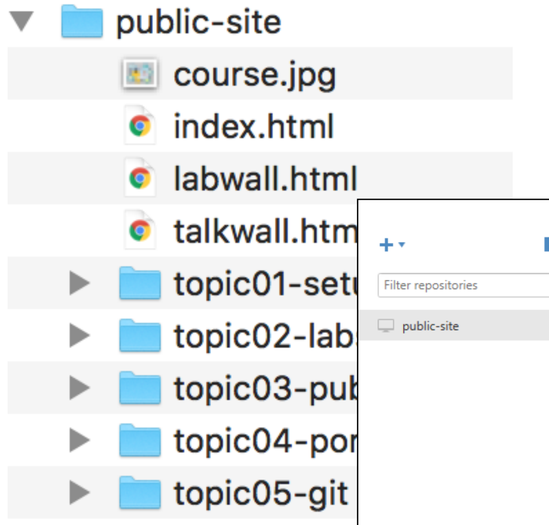
command



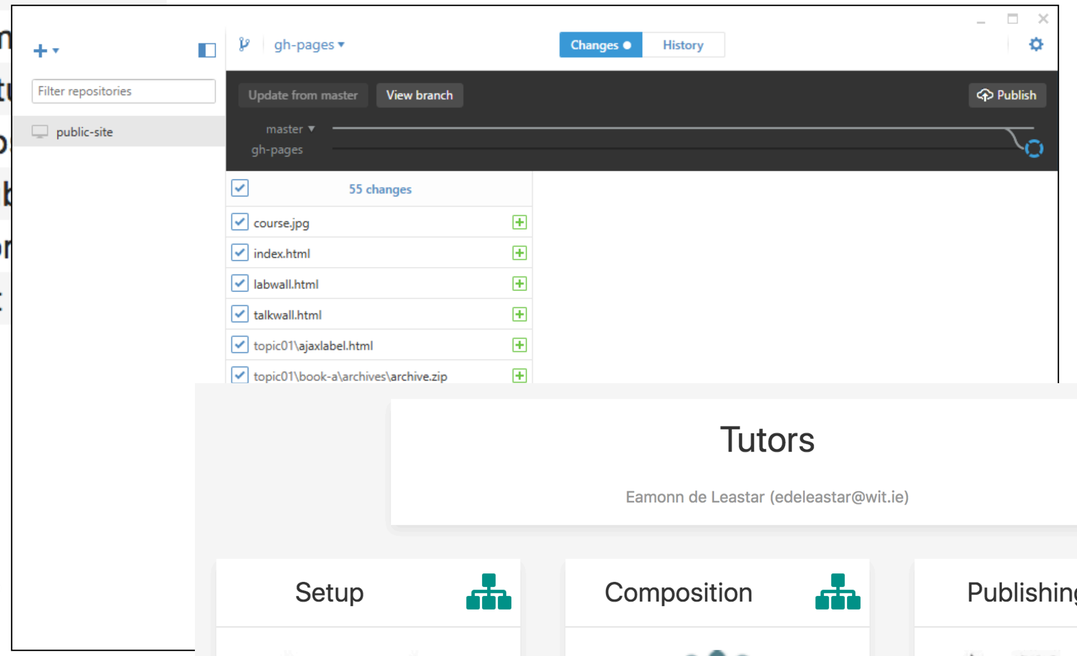
public-site is a ‘static’ web site

easier to publish as it does not require a Content Management System

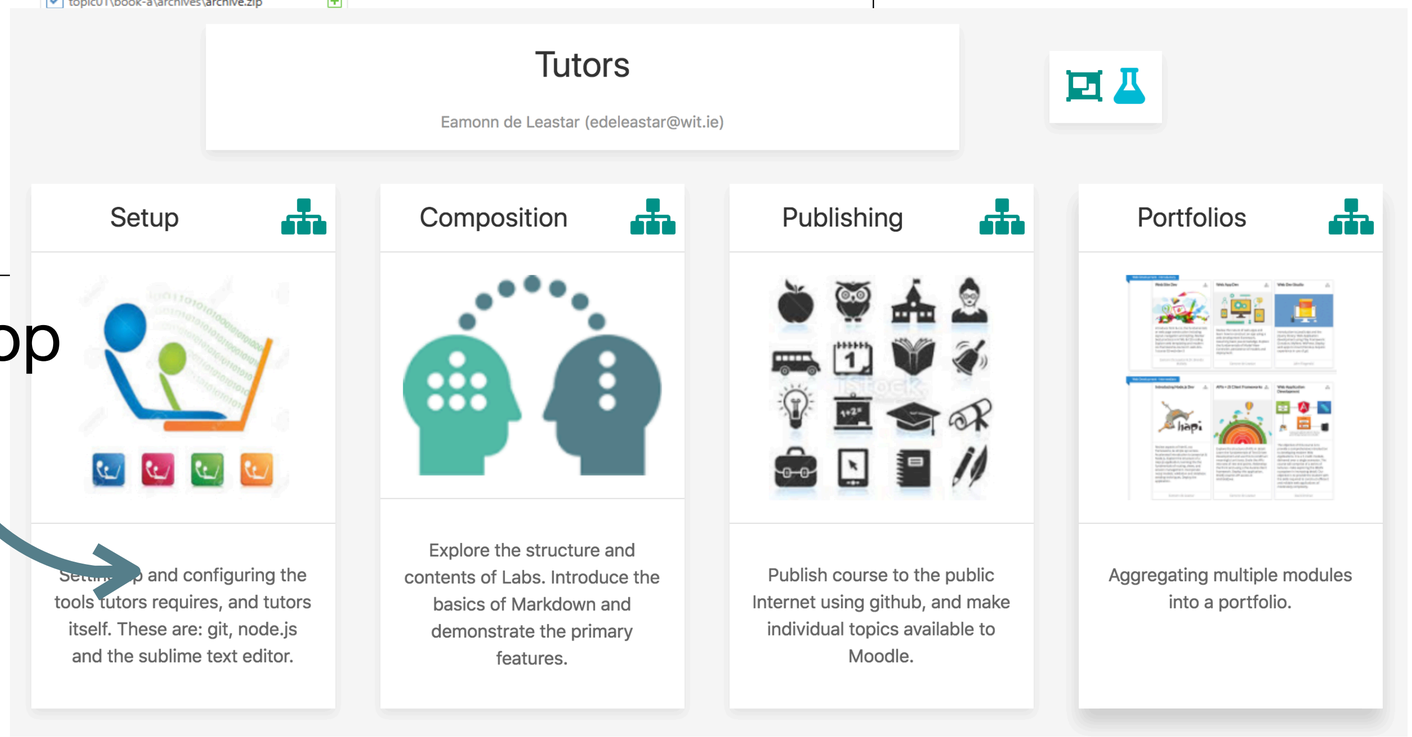
Publishing



folder



git desktop



github pages public site

Key Advantages of Static Site Generators

No Requirement for Content Management System

—> Deploy to low cost/free services

Fast, Reliable & Secure

—> Simplified server infrastructure

Version Control for Content

—> Evolve content independently

Enhanced UX

—> Evolve style independently

Simple Integration

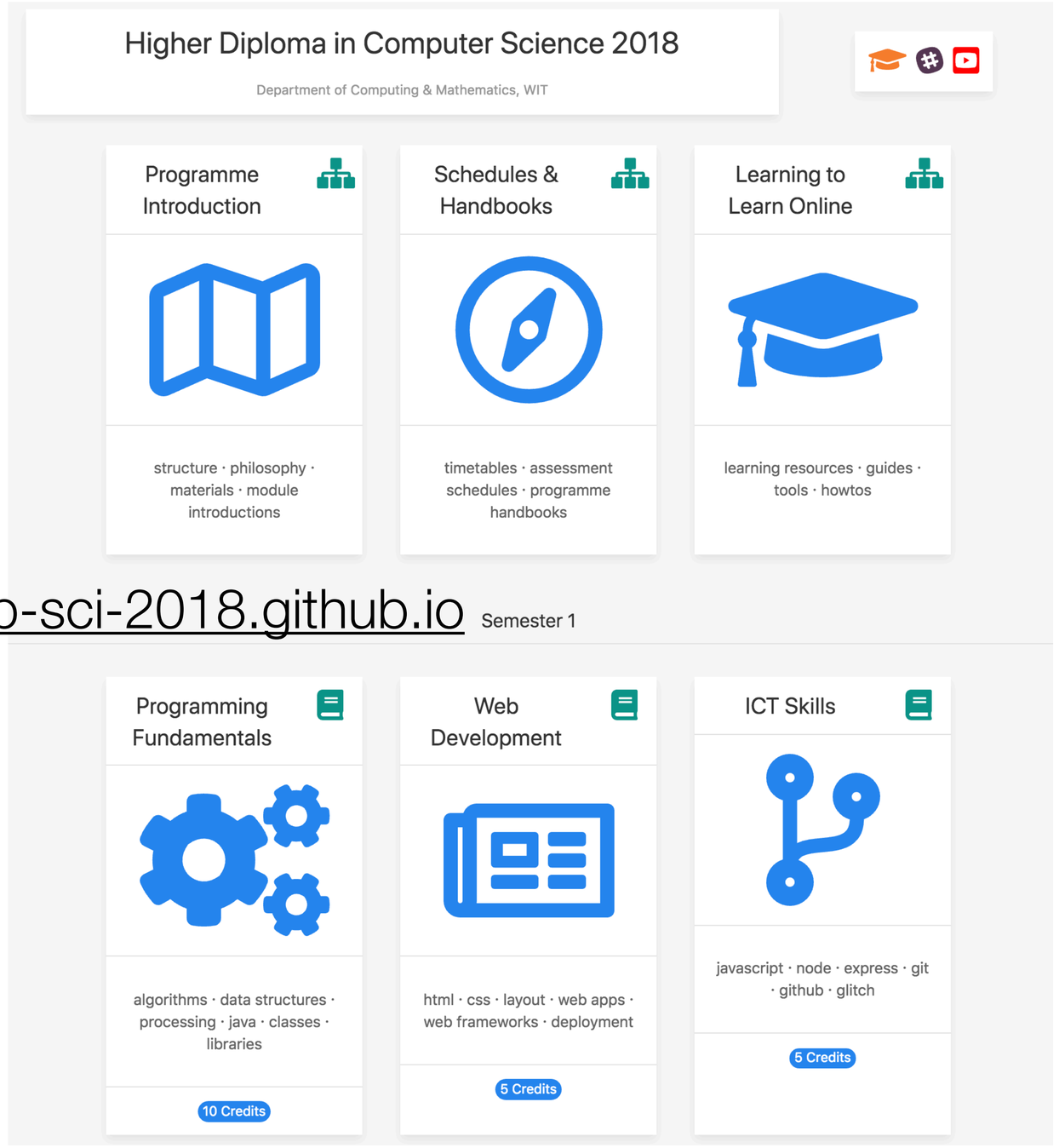
—> Outsource community, assessment and media to external best of breed services

Complete Programme Example

See:

https://wit-hdip-comp-sci-2018.github.io

Comprehensive static site in action



Lab-1.1: Editing HTML

editor · files · text · tags · browser · reference

Lab-1.2: HTML Structure

folders · head · image · ref

Lab-2 CSS Intro

Lab-3a Layout

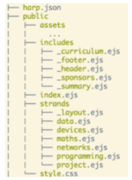
Lab-3b Multicolumn

header · maincontent · navigation · footer · primary · secondary

Lab-4a Navigatio

nav · tab · curre · border · me

Lab-5b Templating



Rebuild the IoT web site from thee last lab using templating. This version of the site will aim to significantly reduce the

Lab-5c Templating Nav

Rework the tabb site from lab 0 templ

Module Review

html · css · layout · navigation · templates · apps · mvc · forms · sessions · deployment

Project 1 Specification

Specification for Assignment 1

Project 2 Specification

Specification for Assignment 2

The Nature of the Web

code · html · http · standards · evolution · role of client · role of servers · learning resources

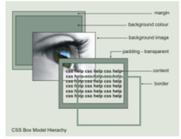
HTML Basics

elements · attributes · paths · absolute · relative · nesting · block · inline · labs

HTML Elements

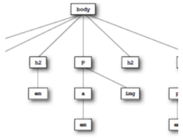
structural elements · head elements · text elements · list elements · image elements · shared attributes

CSS Rules



combining rules · combining selectors · class · id · div

CSS Cascade



rules · class · elements · explicit · inheritance · default · most specific · order

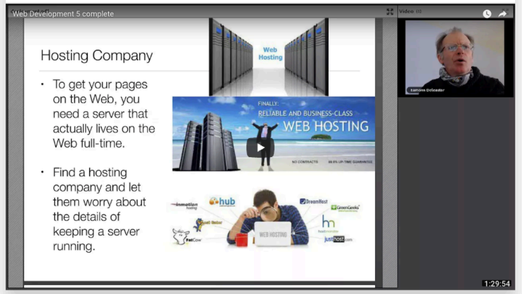
Web Standards Evolution

standards · w3c · whatwg · sgml · xhtml · html5 ·

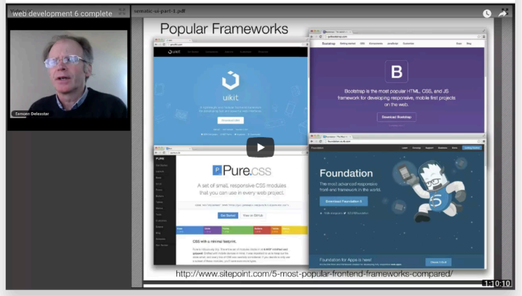
3: Box Model

4: Navigation, Semantics & Style Guides

5: HTML Templates & Deployment



6: CSS Frameworks



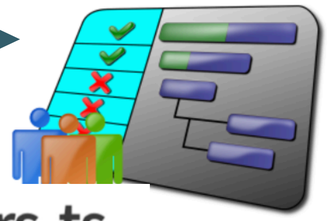
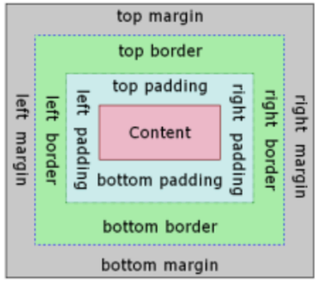
TutorStack

Statically generated Instructional Content + Learning Support Services

- Youtube Channel (**media**)
- Slack (**community & screensharing**)
- Moodle (**assessment & feedback**)
- Youtube Live (**broadcast**)

Web Development

Eamonn de Leastar, WIT

0: Assignments	1: Introducing HTML	2: Introducing CSS	3: The Box Model
			

tutors-ts

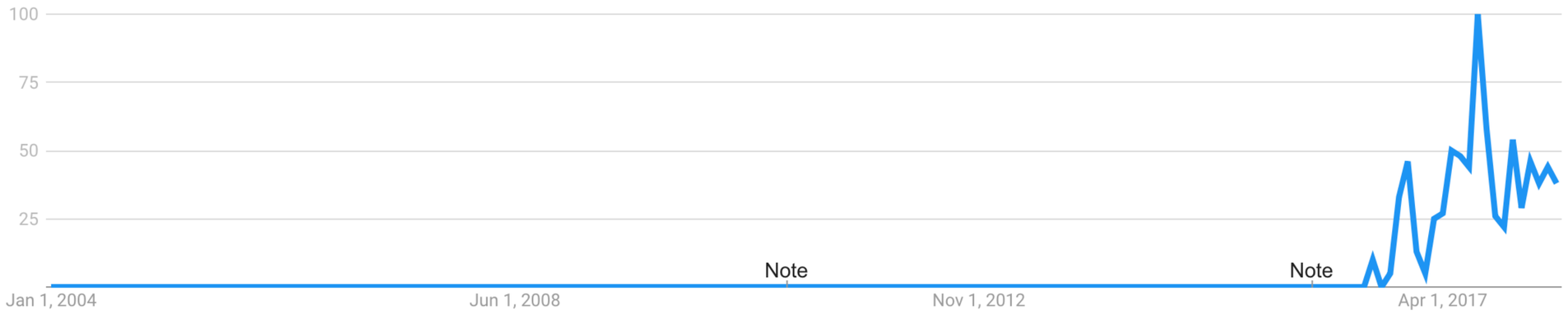
specifications ·

Next Steps….

From Static Site Generators to…

JAMStack
(google it!)

JAMStack



Google Trends

JavaScript

Any dynamic programming during the request/response cycle is handled by JavaScript, running entirely on the client. This could be any frontend framework, library, or even vanilla JavaScript.

APIs

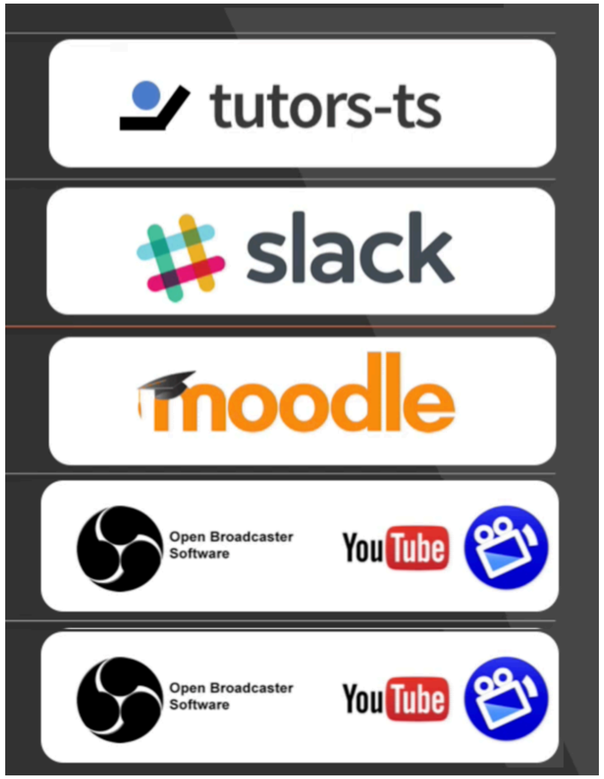
All server-side processes or database actions are abstracted into reusable APIs, accessed over HTTP with JavaScript. These can be custom-built or leverage third-party services.

Markup

Templated markup should be prebuilt at deploy time, usually using a site generator for content sites, or a build tool for web apps.

https://jamstack.org/

Custom JAMStack client to simplify/integrate TutorStack services



JAMSTACK

JAMstack: noun \'jam-stak'\ Modern web development architecture based on client-side JavaScript, reusable APIs, and prebuilt Markup.

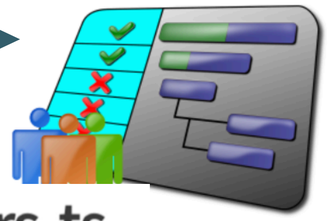
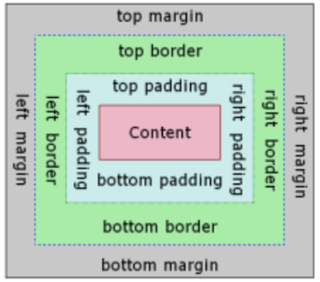
TutorStack

Statically generated Instructional Content + Learning Support Services

- Youtube Channel (**media**)
- Slack (**community & screensharing**)
- Moodle (**assessment & feedback**)
- Youtube Live (**broadcast**)

Web Development

Eamonn de Leastar, WIT

0: Assignments	1: Introducing HTML	2: Introducing CSS	3: The Box Model
			

tutors-ts

specifications ·