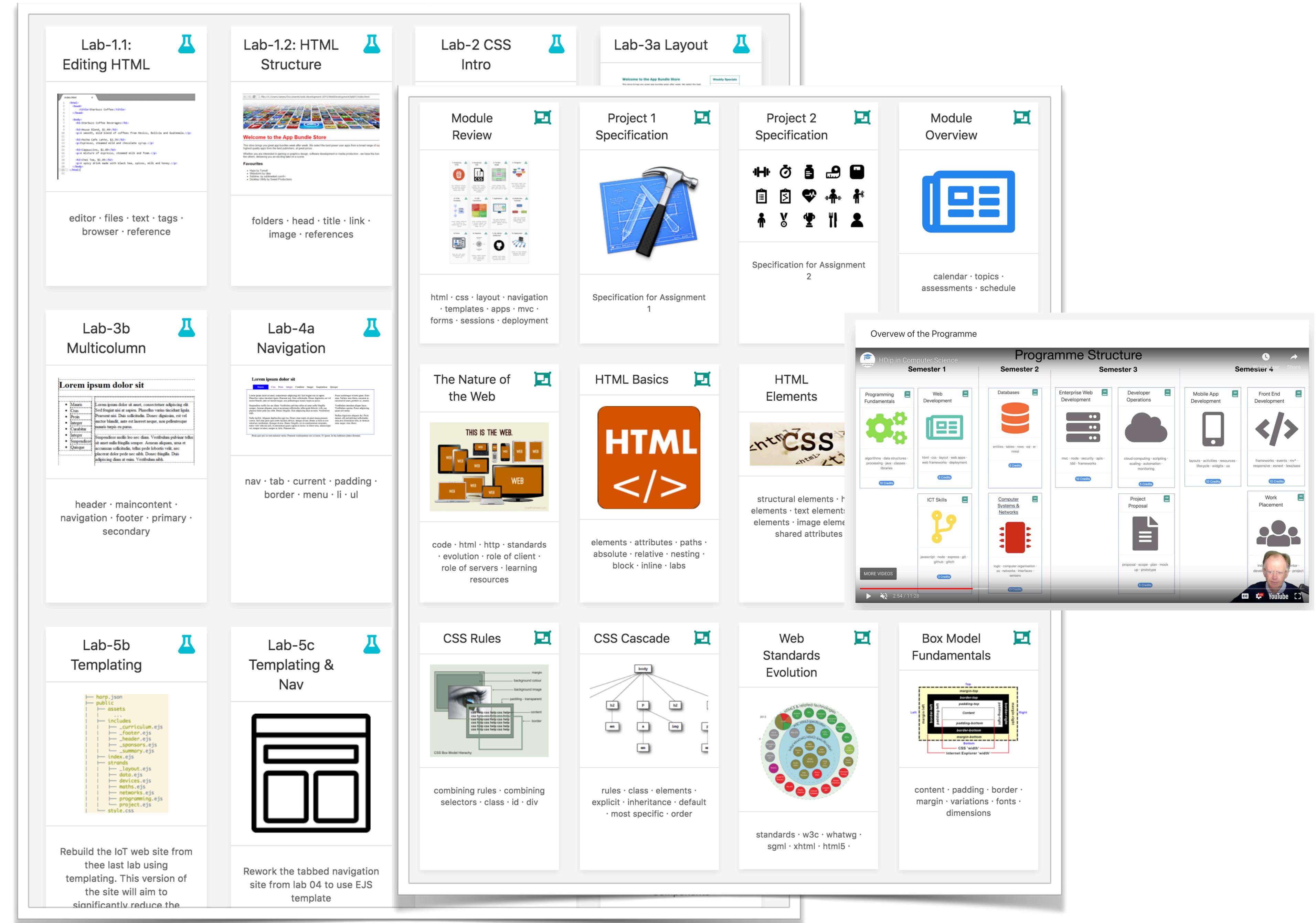


# Static Site Generators for Instructional Content



# TutorStack

1. Instruction Materials



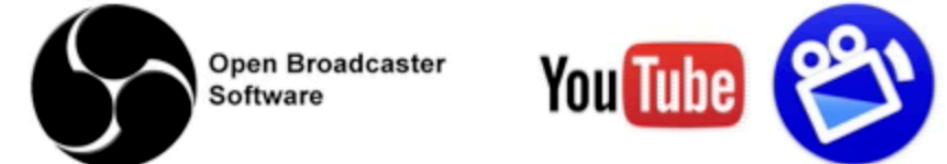
2. Community & screen sharing



3. Assessment & Feedback



4. Media



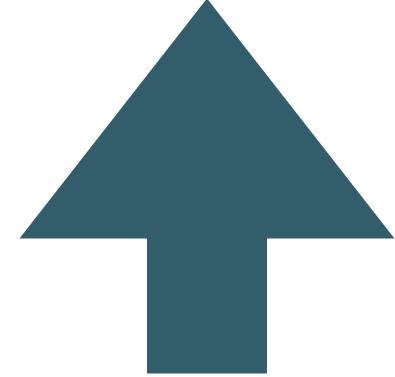
5. Broadcast



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

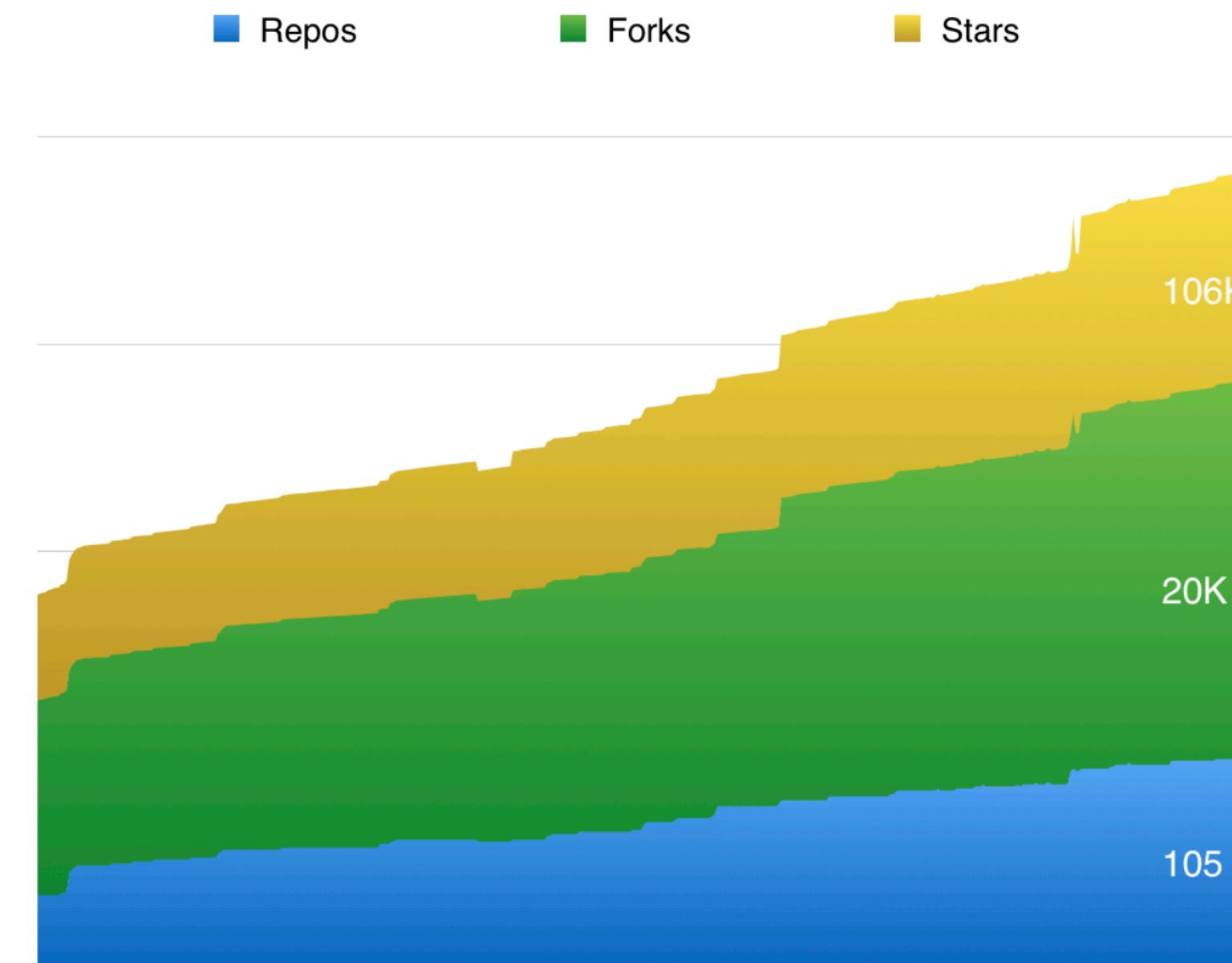
we've seen both the volume and popularity of these projects take off incredibly on GitHub during that time, going from just 50 to more than 100 generators and a total of more than 100,000 stars for static website generator repositories.

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.

NOVEMBER 2, 2015 • [99 COMMENTS](#)

## Why Static Site Generators Are The Next Big Thing

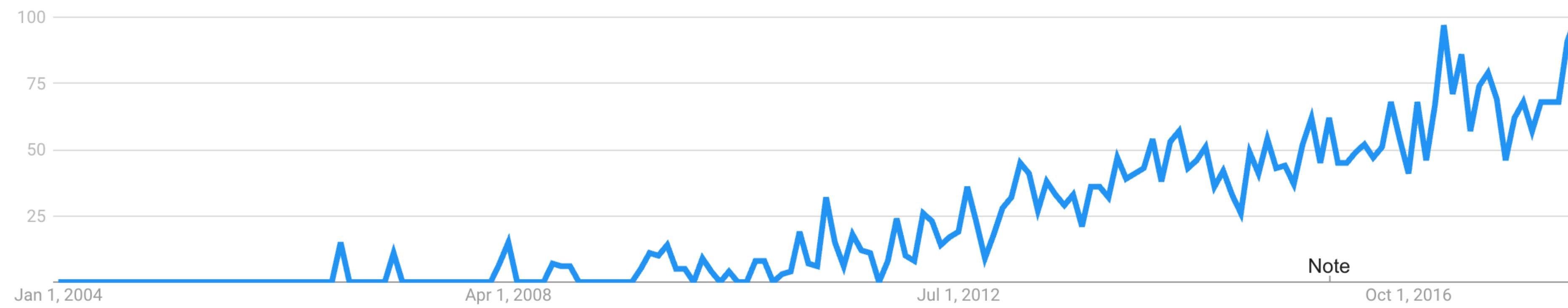
# [Coding](#) 587 # [Tools](#) 173 # [Static Generators](#) 9



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

2015!

# Static Site Generator



Google Trends

# StaticGen

A List of Static Site Generators for JAMstack Sites



About Contribute About JAMstack Need a Static CMS?

Filter

Any Language ▾ Any Template ▾ Any License ▾

Sort

GitHub stars ▾

## Jekyll

★ 34352  
+163  
① 114  
-6  
¥ 7572  
+37  
🐦 6340  
+17

A simple, blog-aware, static site generator.

Languages: Ruby

Templates: Liquid

License: MIT

 Deploy to Netlify

## Hugo

★ 25860  
+405  
① 270  
+14  
¥ 3126  
+32  
🐦 3739  
+97

A Fast and Flexible Static Site Generator.

Languages: Go

Templates: Go

License: Apache 2.0

 Deploy to Netlify

## Next

★ 25537  
+513  
① 280  
+29  
¥ 2462  
+91  
🐦 N/A

A framework for statically-exported React apps

Languages: JavaScript

Templates: JavaScript

License: MIT

**Get started with one click!**

For generators with the "Deploy to Netlify" button, you can deploy a new site from a template with one click. Get HTTPS, continuous delivery, and bring a custom domain, free of charge.

Want your own Deploy to Netlify button? [Learn more here](#).

<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



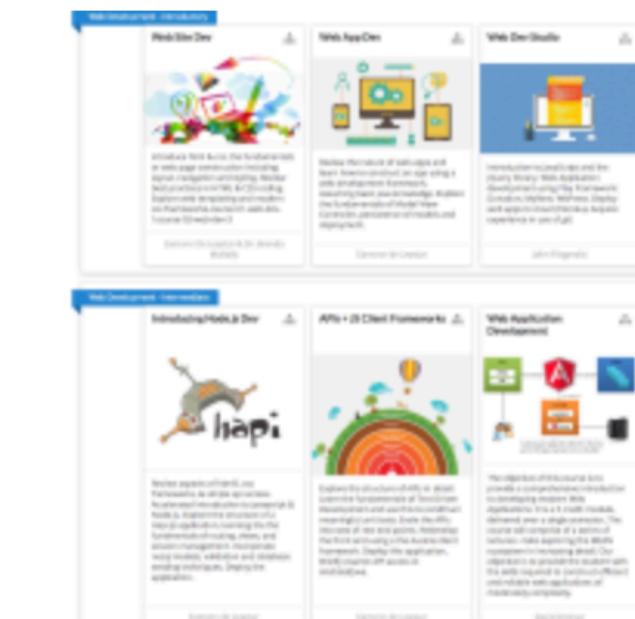
Explore the structure and contents of Labs. Introduce the basics of Markdown and demonstrate the primary features.

## 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



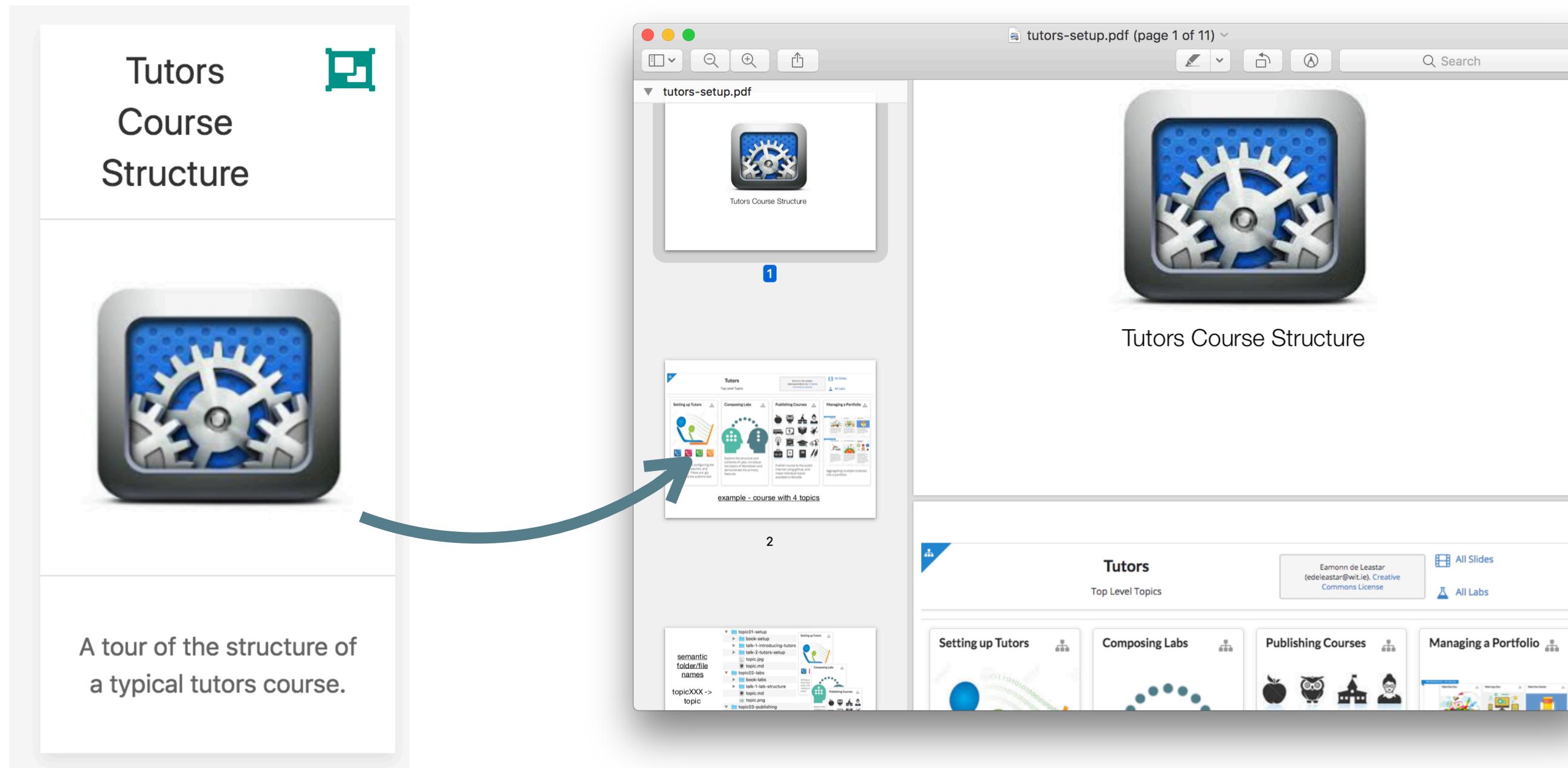
A tour of the structure of a typical tutors course.

## Lab-01- Setup



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

1 topic = slides + lab



## Talk (slides)

A screenshot of a presentation slide titled "Setup". The slide has a dark header with the title and a light-colored body. At the bottom, there's a navigation bar with links: "Lab-01-Setup", "01", "02", "03", "04", "05", "06", "07", "08", and "Exercises".

A screenshot of a presentation slide titled "Lab-01-Setup". The slide features a computer monitor icon and the text "Install the tutors command line application and take for a first spin." A blue arrow points from the monitor icon to the terminal window below.

**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:

A screenshot of a Windows File Explorer window showing the contents of the `tutors-starter-0` folder. The folder structure is as follows:

- File
- Home
- Share
- View
- Local Disk (C:)
- dev
- topic01

## Lab (guides)

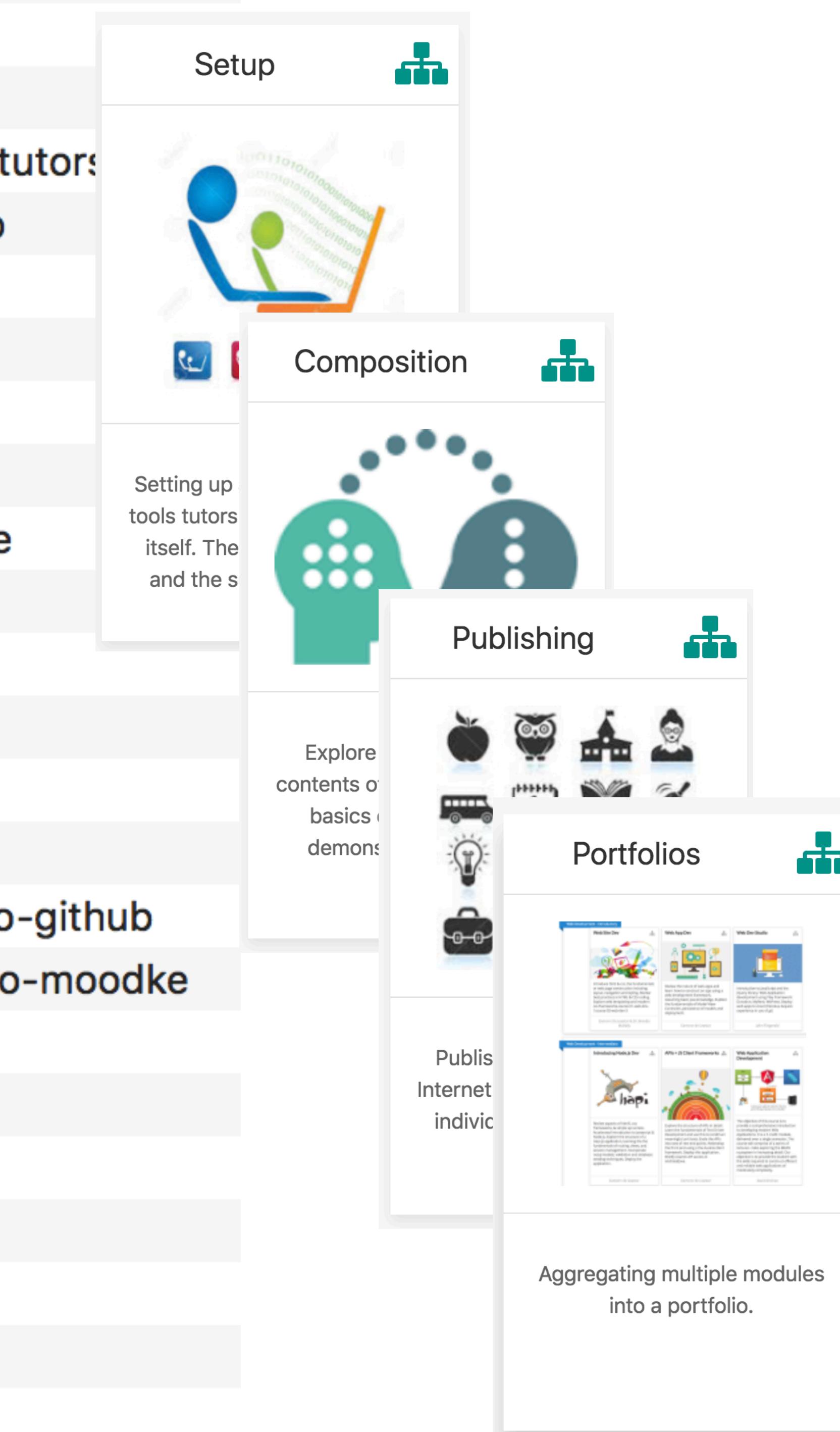
# Semantic folder/file names

topicXXX ->  
topic

bookXXX ->  
lab

talkXXX ->  
talk

▼	topic01-setup
►	book-setup
►	talk-1-introducing-tutors
►	talk-2-tutors-setup
	topic.jpg
	topic.md
▼	topic02-labs
►	book-labs
►	talk-1-lab-structure
	topic.md
	topic.png
▼	topic03-publishing
►	book-a-gh-pages
►	book-b-moodle
►	talk-1-publishing-to-github
►	talk-2-publishing-to-moodke
	topic.jpg
	topic.md
▼	topic04-portfolios
►	book-portfolio
►	talk-1-portfolio
	topic.md
	topic.png



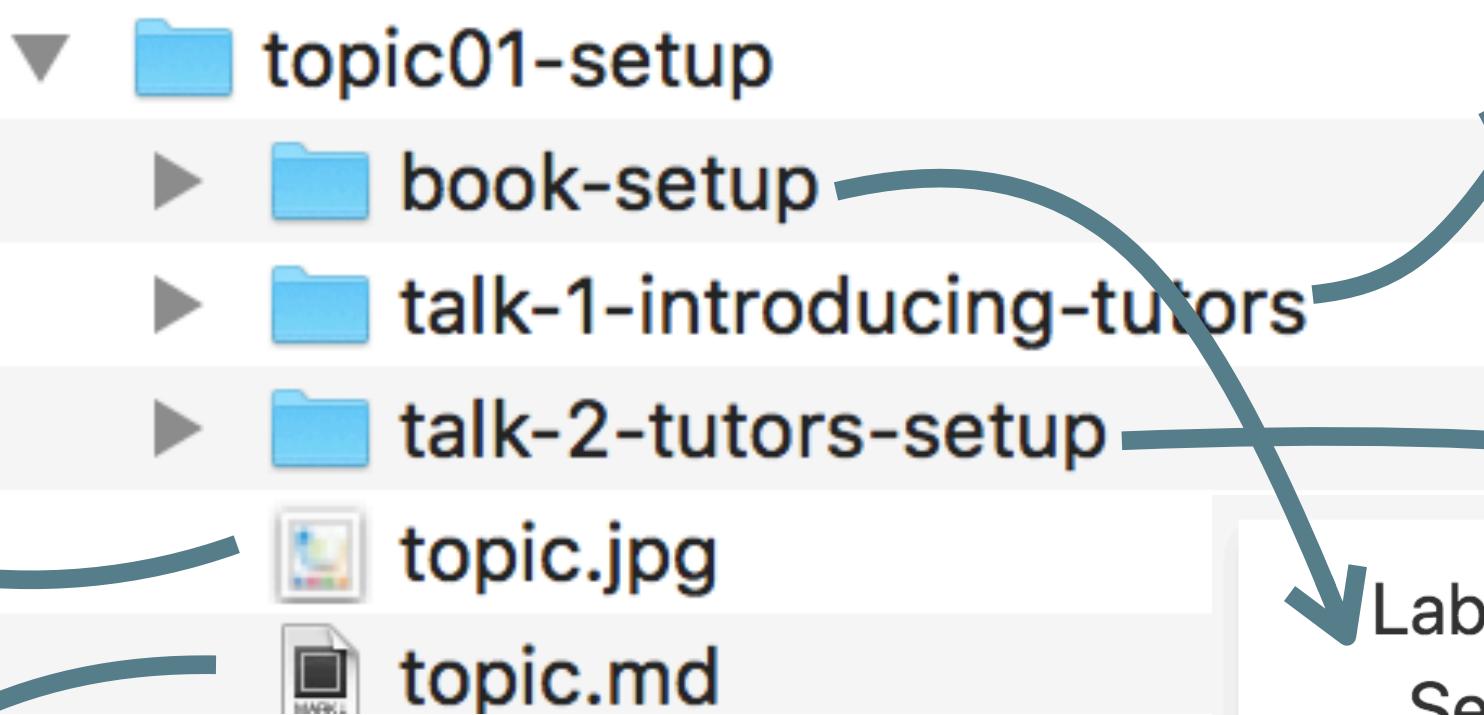
## Introducing Tutors



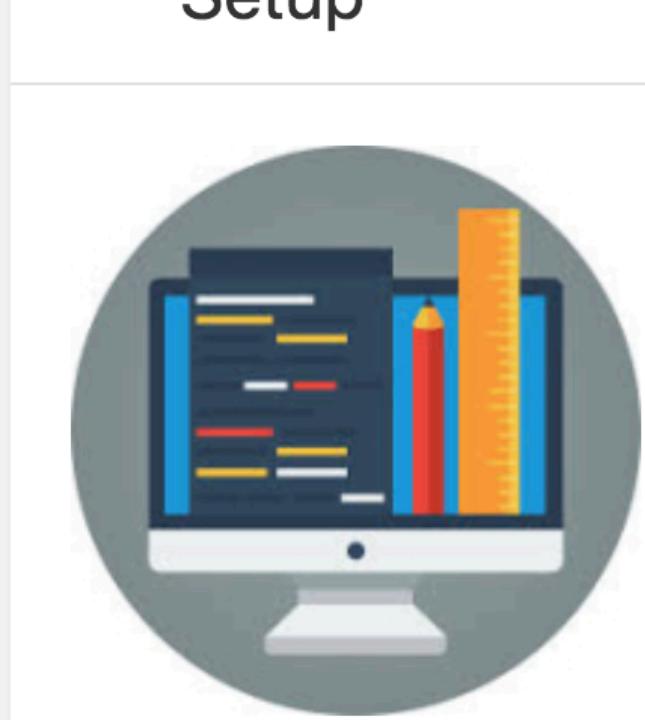
A quick tour of the purpose, structure and features of a course web.

### Setup

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



## Lab-01-Setup



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

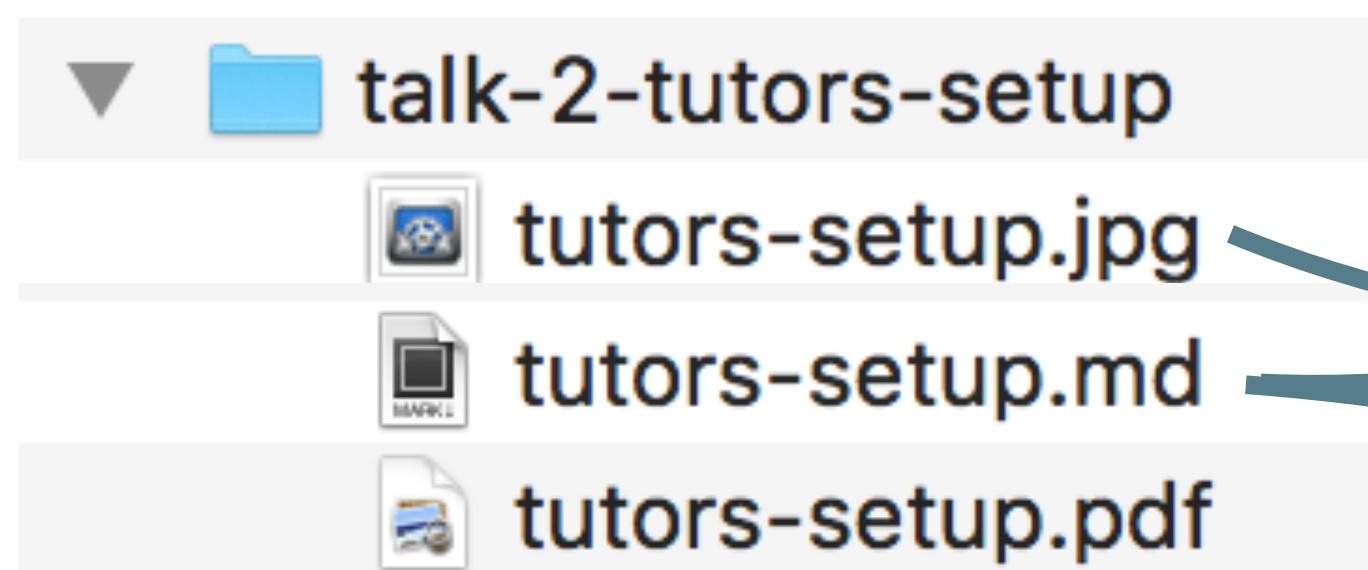
## Tutors Course Structure



A tour of the structure of a typical tutors course.

# Topic structure

# Talk structure



all files  
same name

- image
- pdf
- markdown

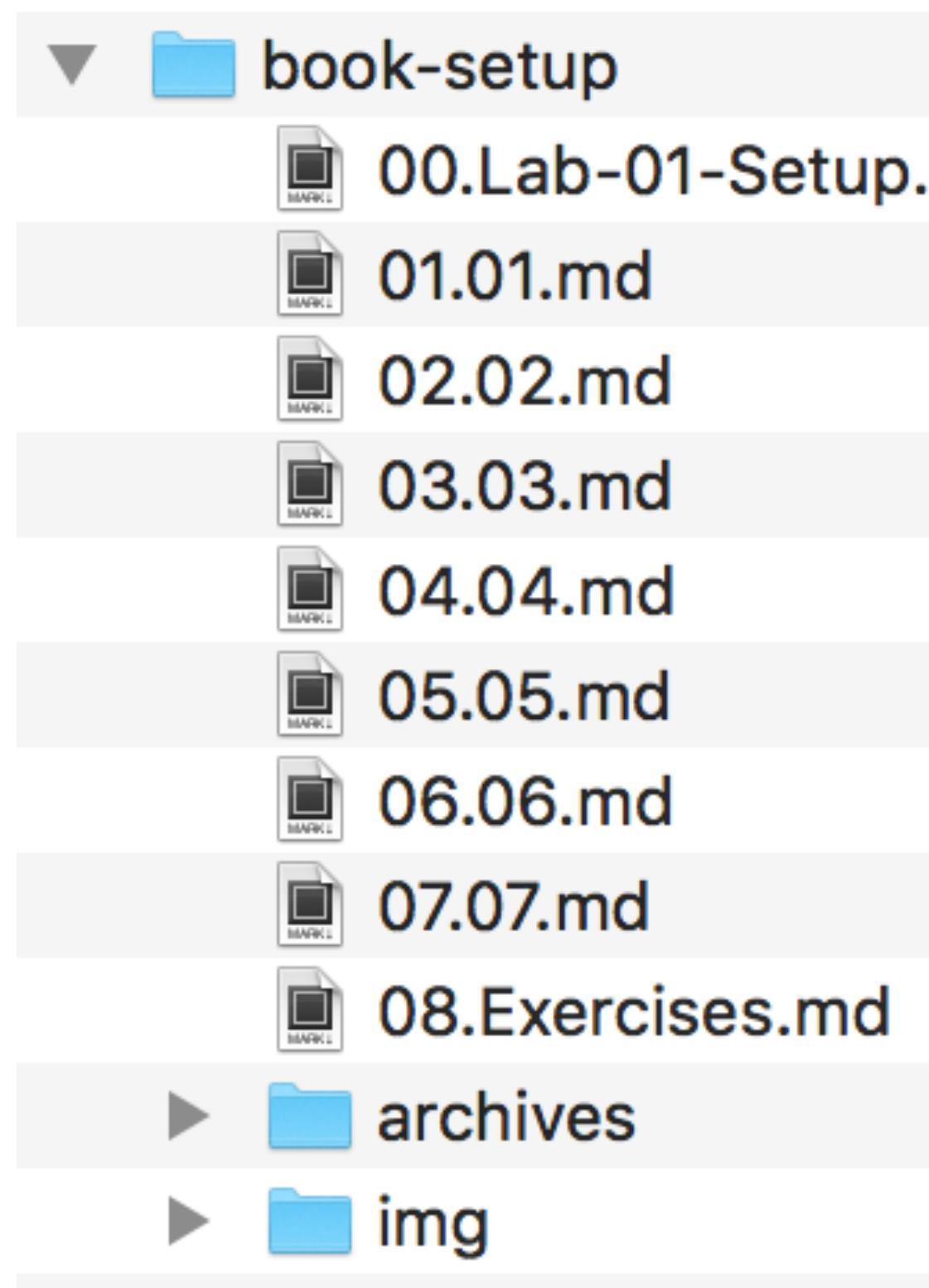
Tutors Course Structure

A tour of the structure of a typical tutors course.

## Tutors Setup

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

# Lab structure



## *img*

- images used in lab

## *archives*

- zipped archive linked to in labs

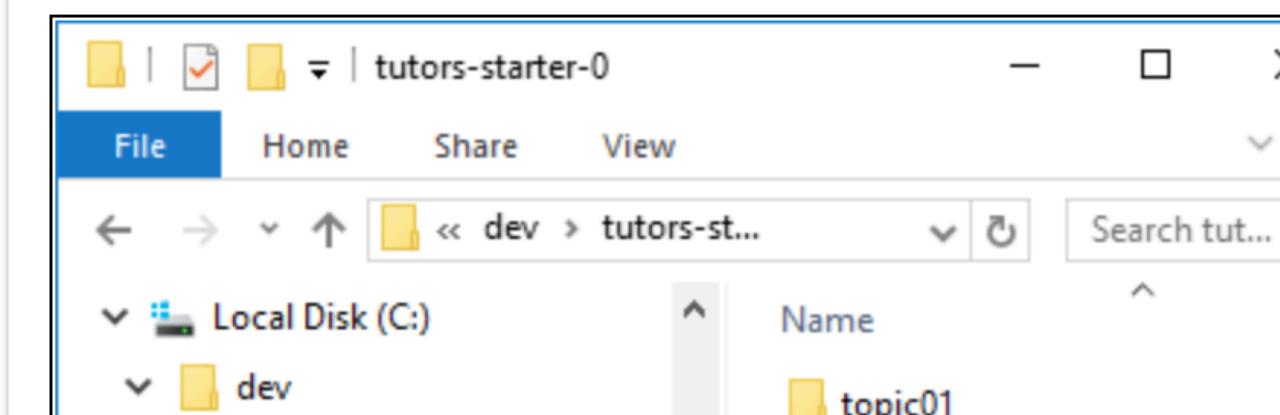
One *md* file per step

```
tutors new
```

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

```
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:



## Generate a Course

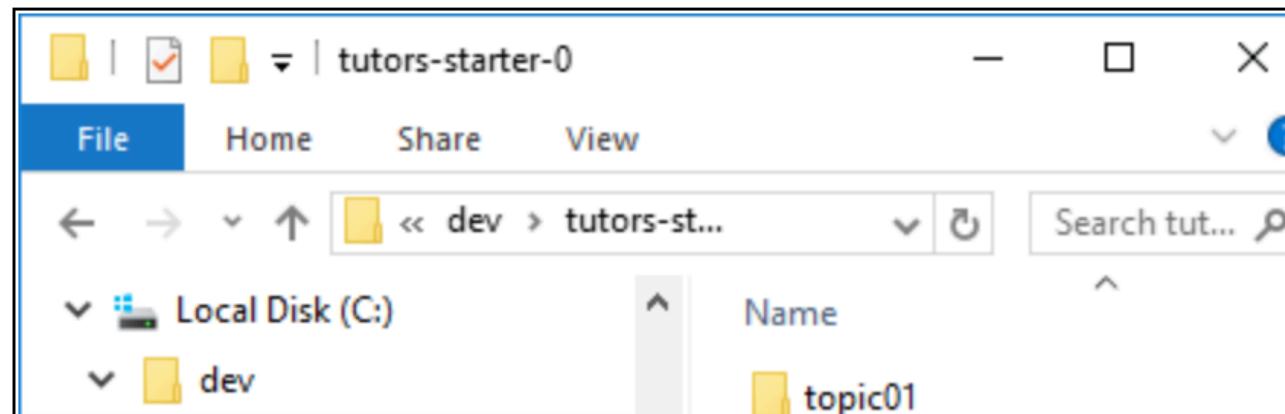
The simplest way to get started with tutors is to use the `new` command to generate a template course:

```
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 course:



# 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
```

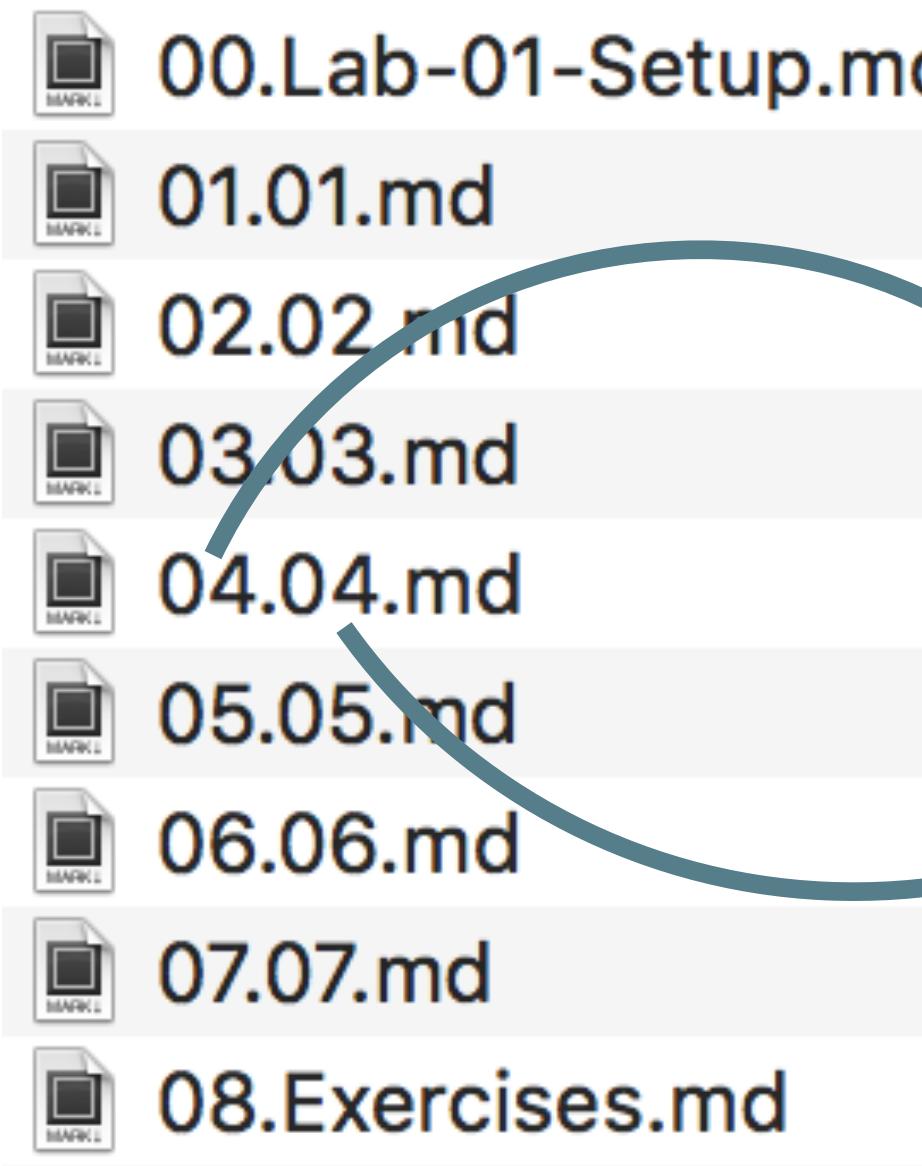
```
This will generate the course web in "tutors-starter/public-site"
```

~~~

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



## lab menu bar



XX

- must be a 2 digit number

YY

- can be any string
- .md
- must me .md



Link to TOC of all labs

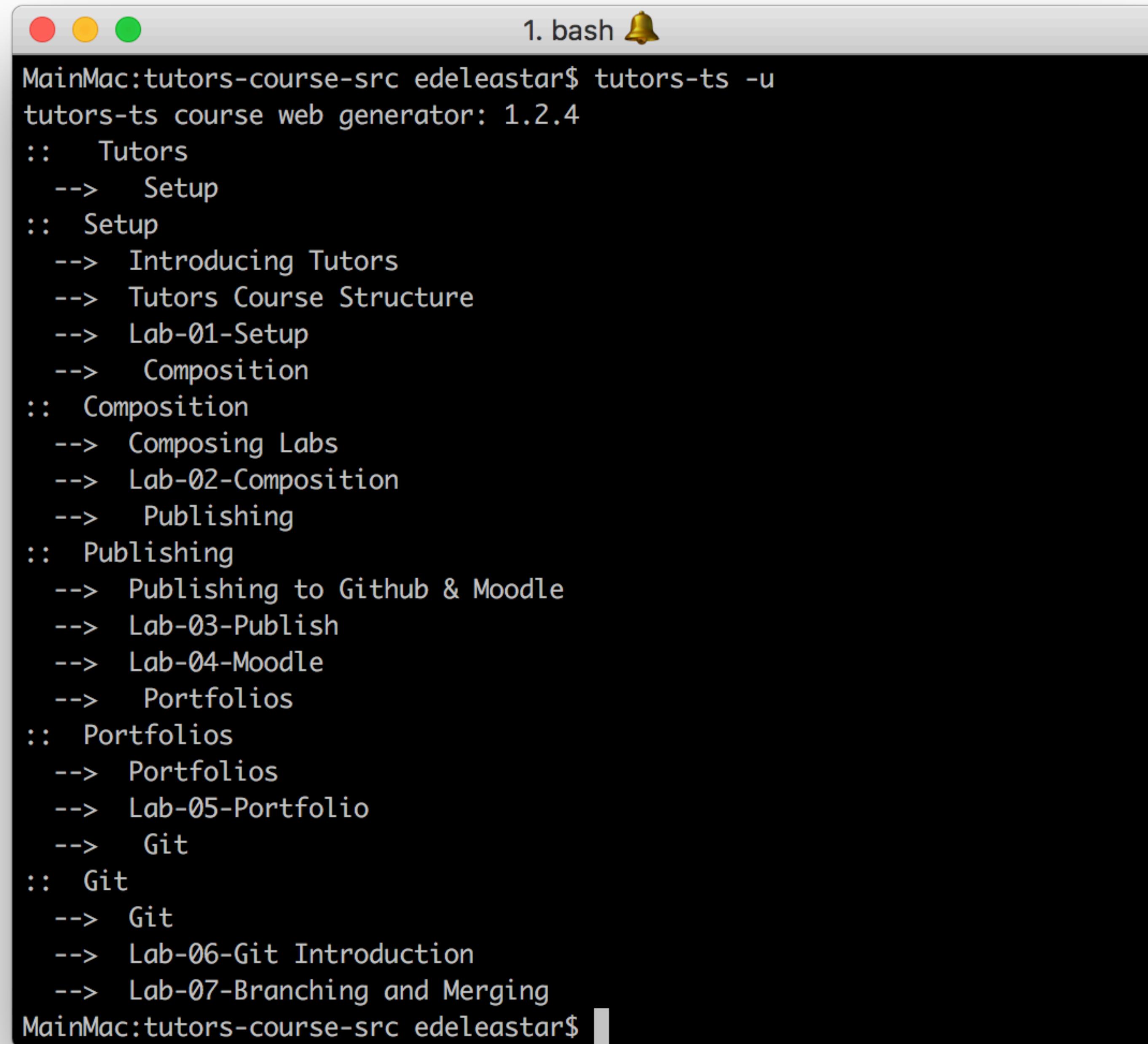
Link to Parent Topic

First step, includes lab name

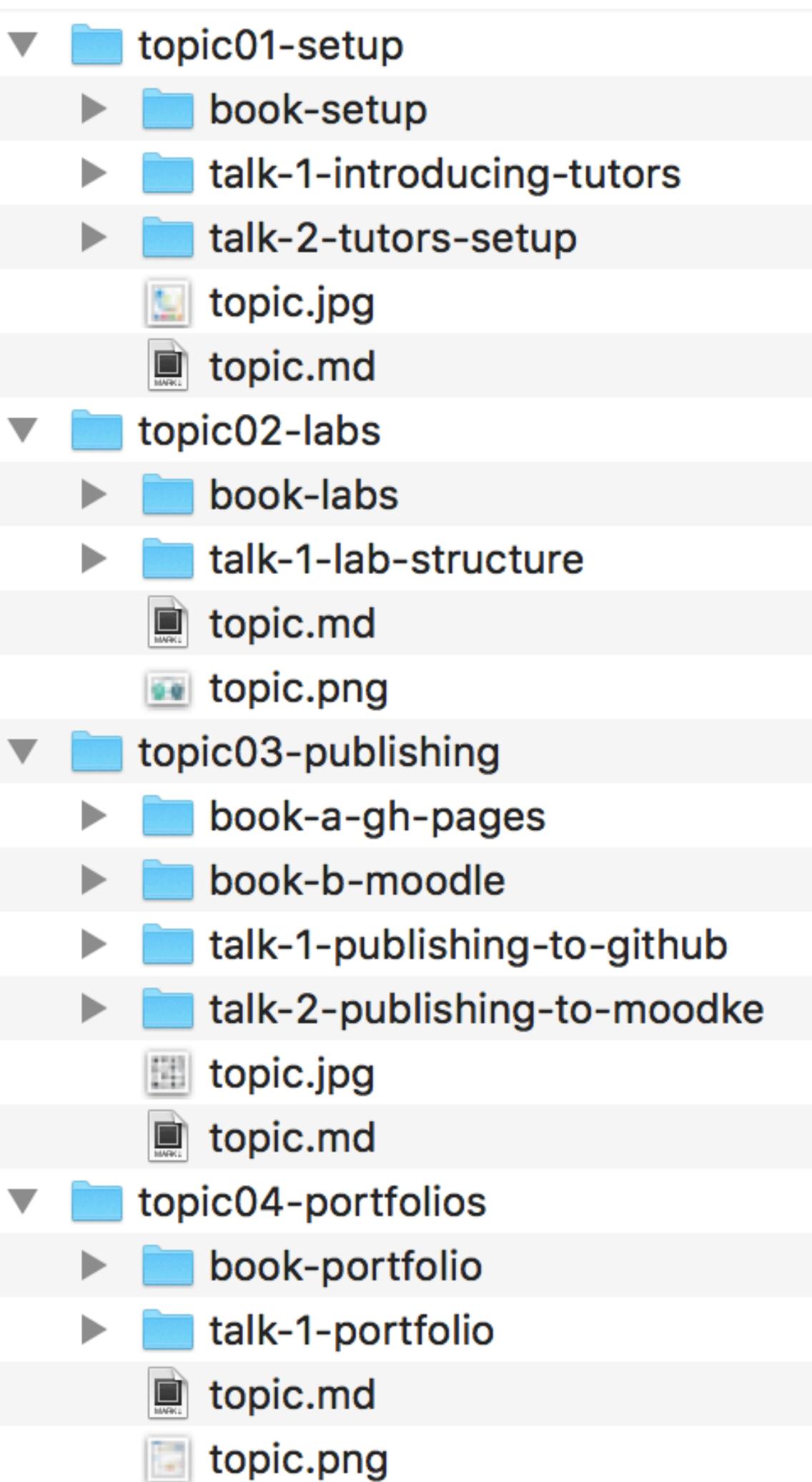
YY is used as step name in the menubar

Link to course home page

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

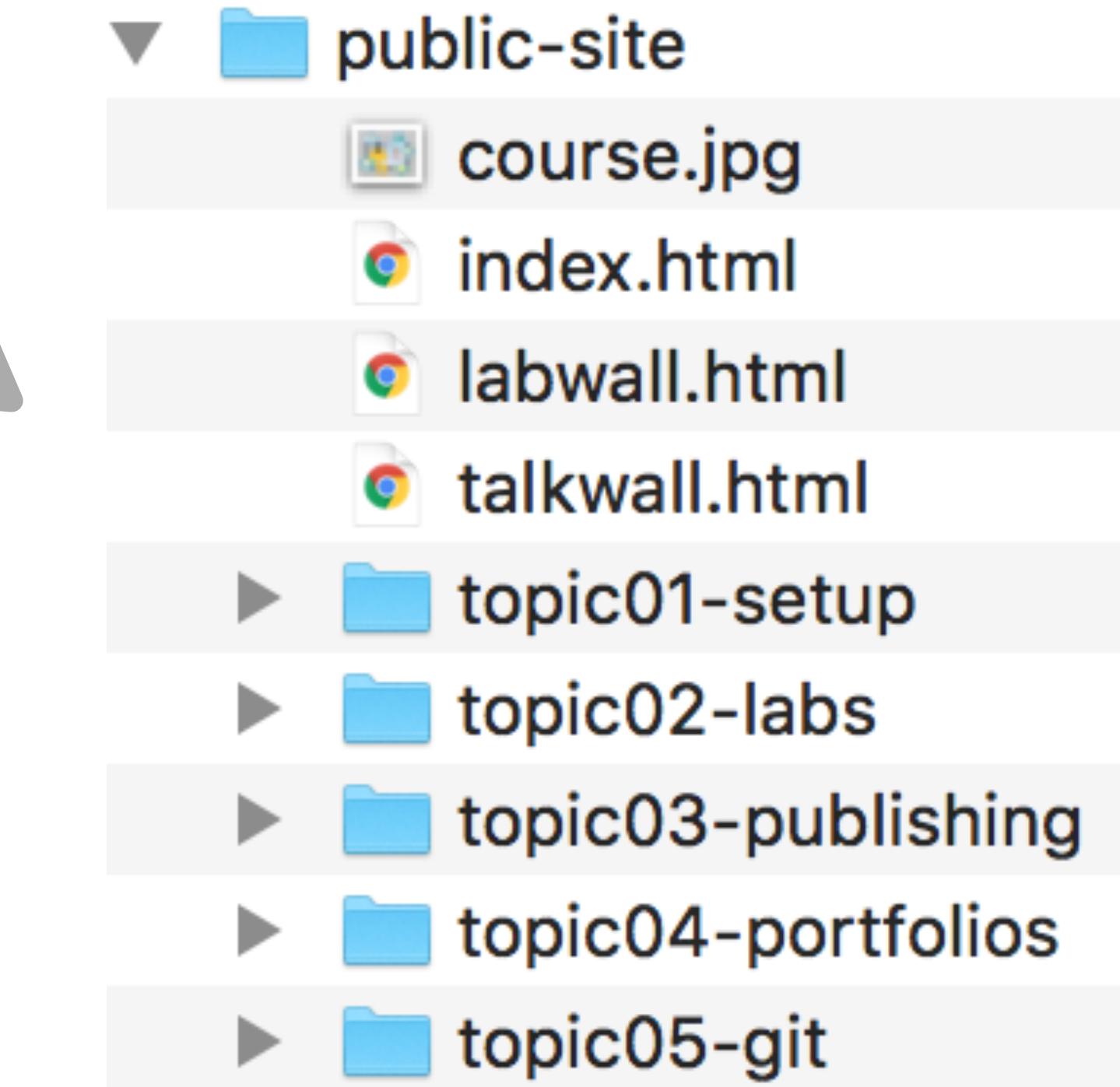


MainMac:tutors-course-src edeleastar\$ tutors-ts -u  
tutors-ts course web generator: 1.2.4  
:: Tutors  
--> Setup  
:: Setup  
--> Introducing Tutors  
--> Tutors Course Structure  
--> Lab-01-Setup  
--> Composition  
:: Composition  
--> Composing Labs  
--> Lab-02-Composition  
--> Publishing  
:: Publishing  
--> Publishing to Github & Moodle  
--> Lab-03-Publish  
--> Lab-04-Moodle  
--> Portfolios  
:: Portfolios  
--> Portfolios  
--> Lab-05-Portfolio  
--> Git  
:: Git  
--> Git  
--> Lab-06-Git Introduction  
--> Lab-07-Branching and Merging  
MainMac:tutors-course-src edeleastar\$



tutors

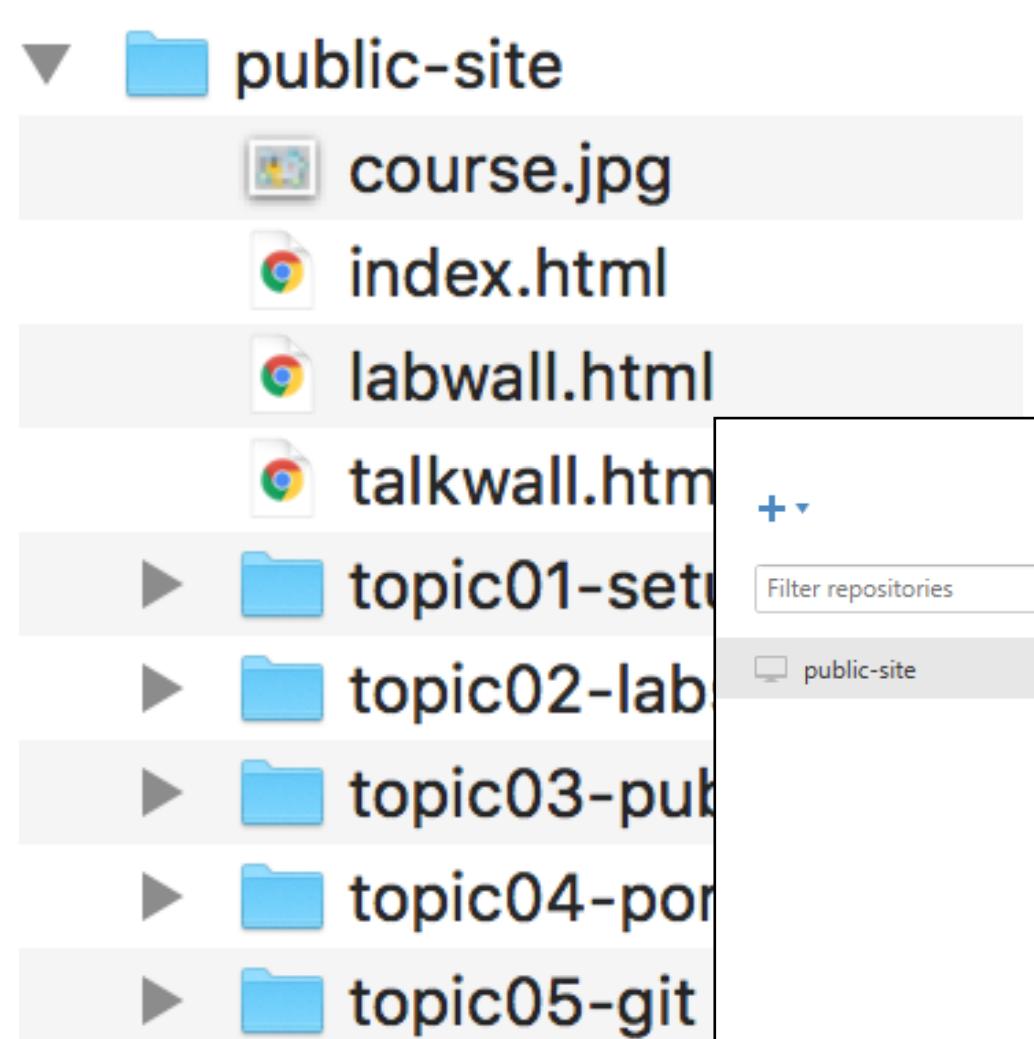
command



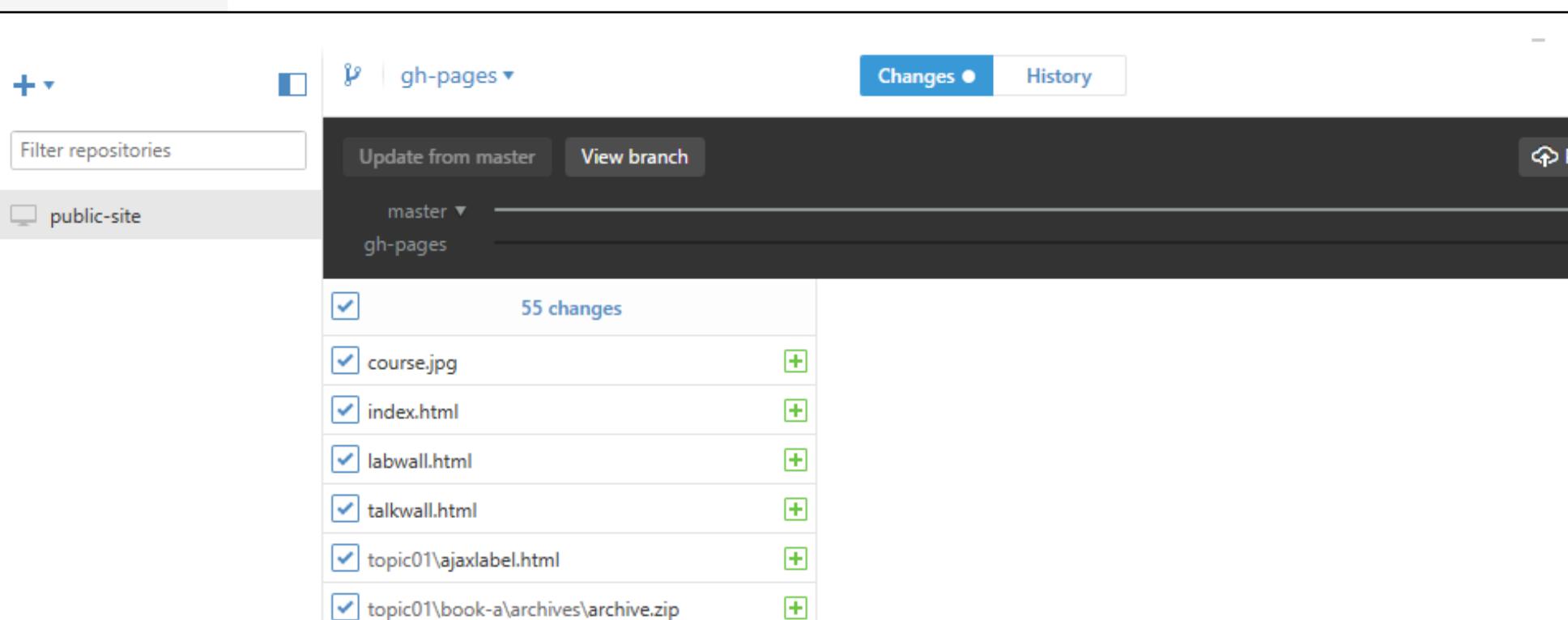
*public-site* is a ‘static’ web site

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

# Publishing



folder



git desktop

The image contains four cards:

- Tutors**: Shows a profile picture of Eamonn de Leastar and the email address edeleastar@wit.ie.
- Setup**: Icon shows two people at a computer. Description: Setting up and configuring the tools tutors requires, and tutors itself. These are: git, node.js and the sublime text editor.
- Composition**: Icon shows two stylized human heads. Description: Explore the structure and contents of Labs. Introduce the basics of Markdown and demonstrate the primary features.
- Publishing**: Icon shows a stack of books. Description: Publish course to the public Internet using github, and make individual topics available to Moodle.
- Portfolios**: Icon shows a portfolio. Description: Aggregating multiple modules into a portfolio.

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

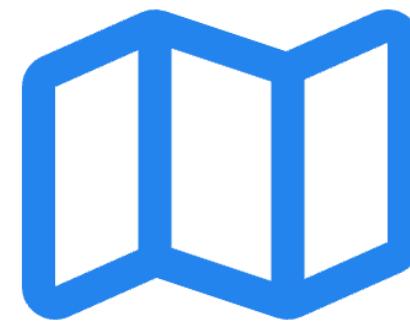
See:

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

Semester 1

## Comprehensive static site in action

Programme  
Introduction



structure · philosophy ·  
materials · module  
introductions

Schedules &  
Handbooks



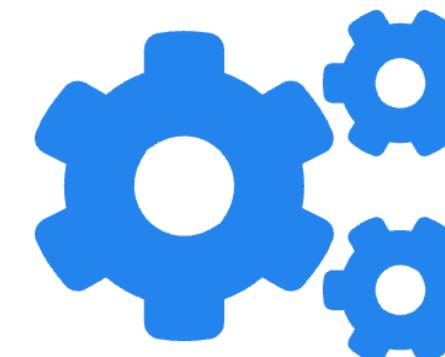
timetables · assessment  
schedules · programme  
handbooks

Learning to  
Learn Online



learning resources · guides ·  
tools · howtos

Programming  
Fundamentals



algorithms · data structures ·  
processing · java · classes ·  
libraries

10 Credits

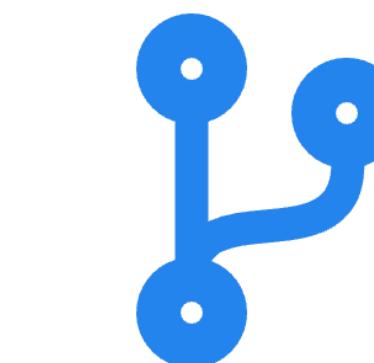
Web  
Development



html · css · layout · web apps ·  
web frameworks · deployment

5 Credits

ICT Skills



javascript · node · express · git  
· github · glitch

5 Credits

# Lab-1.1: Editing HTML

# Lab-1.2: HTML Structure

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>APP Stores</title>
    <link type="text/css" rel="stylesheet" href="style.css" media="screen" />
    <script>
      <!--
        <!--Score: Apps, Movies, Music, Books-->
        <a href="apps.html">Apps</a>
        <a href="music.html">Music</a>
        <a href="movies.html">Movies</a>
      -->
    </script>
  </head>
  <div class="main_panel">
    <ul style="list-style-type: none; padding-left: 0;">
      <li>Clear All</li>
      <li>Google Box</li>
      <li>Squints</li>
    </ul>
  </div>
</body>
</html>
```

# Lab-3a Layout

# Lab-3b

## Multicolumn

---

### **Lore** ipsum dolor sit

<ul style="list-style-type: none"><li>Mauris</li><li>Cras</li><li>Proin</li><li>Integer</li><li>Curabitur</li><li>Integer</li><li>Suspendisse</li><li>Quisque</li></ul>	<p>orem ipsum dolor sit amet, consectetur adipiscing elit. Sed feugiat nisi at sapien. Phasellus varius tincidunt ligula. Praesent nisi. Duis sollicitudin. Donec dignissim, est vel auctor blandit, ante est laoreet neque, non pellentesque mauris turpis eu purus.</p> <p>Suspendisse mollis leo nec diam. Vestibulum pulvinar tellus sit amet nulla fringilla semper. Aenean aliquam, urna et accumsan sollicitudin, tellus pede lobortis velit, nec placerat dolor pede nec nibh. Donec fringilla. Duis adipiscing diam at erat. Vestibulum nibh.</p>
---	--

---

header · maincontent ·  
navigation · footer · primary ·  
secondary

# Lab-4a

## Navigatio

A blue blueprint-style icon showing a drawing of a person's head and shoulders, with a pencil pointing to it. The text "Assignment" is written below the icon.

Document Type

Use HTML5.

HTML5 (HTML syntax) is preferred for all HTML documents: `<!DOCTYPE html>`.

# Lab-5b

## Templating

# Lab-5c

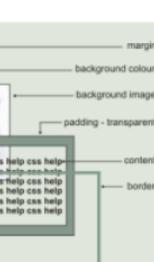
## Templating

### Nav

html · http · standards  
on · role of client ·  
servers · learning  
resources

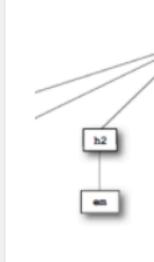
elements · attr  
absolute · rel  
block · in

## Rules



margin  
background colour  
background image  
padding - transparent  
content  
border  
css Help.css Help.css Help.css  
css Help.css Help.css Help.css Help.css  
css Help.css Help.css Help.css Help.css  
css Help.css Help.css Help.css Help.css

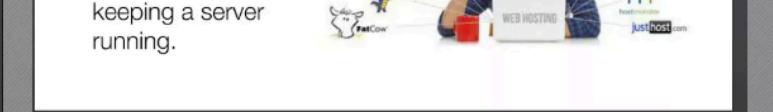
## CSS Cascade



box  
h2  
p  
span

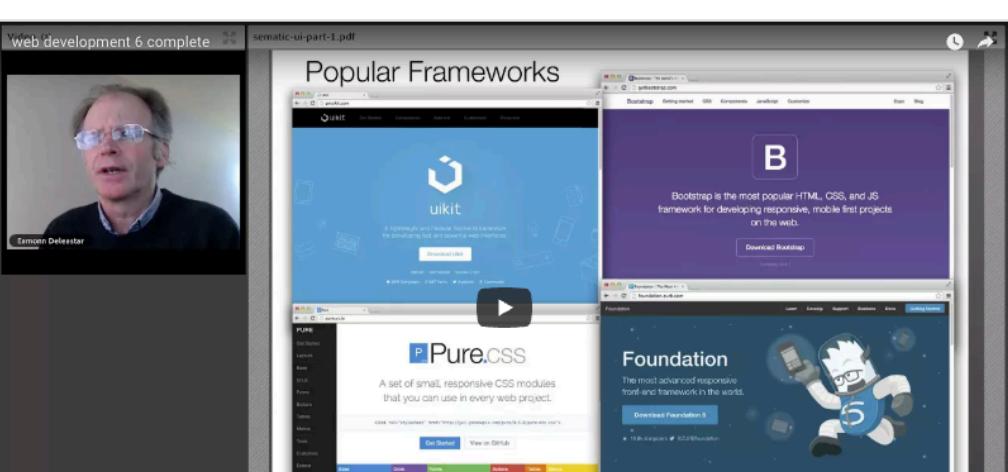
nesting rules · combining  
rules · class · id · div

rules · class  
explicit · inherit  
· most spe



• Find a hosting company and let them worry about the details of keeping a server running.

## 6: CSS Frameworks



The slide displays four popular CSS frameworks side-by-side:

- UIKit**: A framework and toolset for building iOS-like user interfaces.
- Bootstrap**: The most popular HTML, CSS, and JS framework for developing responsive, mobile-first projects on the web.
- Pure.css**: A set of small, responsive CSS modules that you can use in every web project.
- Foundation**: The most advanced responsive front-end framework in the world.

<http://www.sitepoint.com/5-most-popular-frontend-frameworks-compared/>

# TutorStack

Statically generated  
Instructional Content + Learning Support Services →

## Web Development

Eamonn de Leastar, WIT

0: Assignments

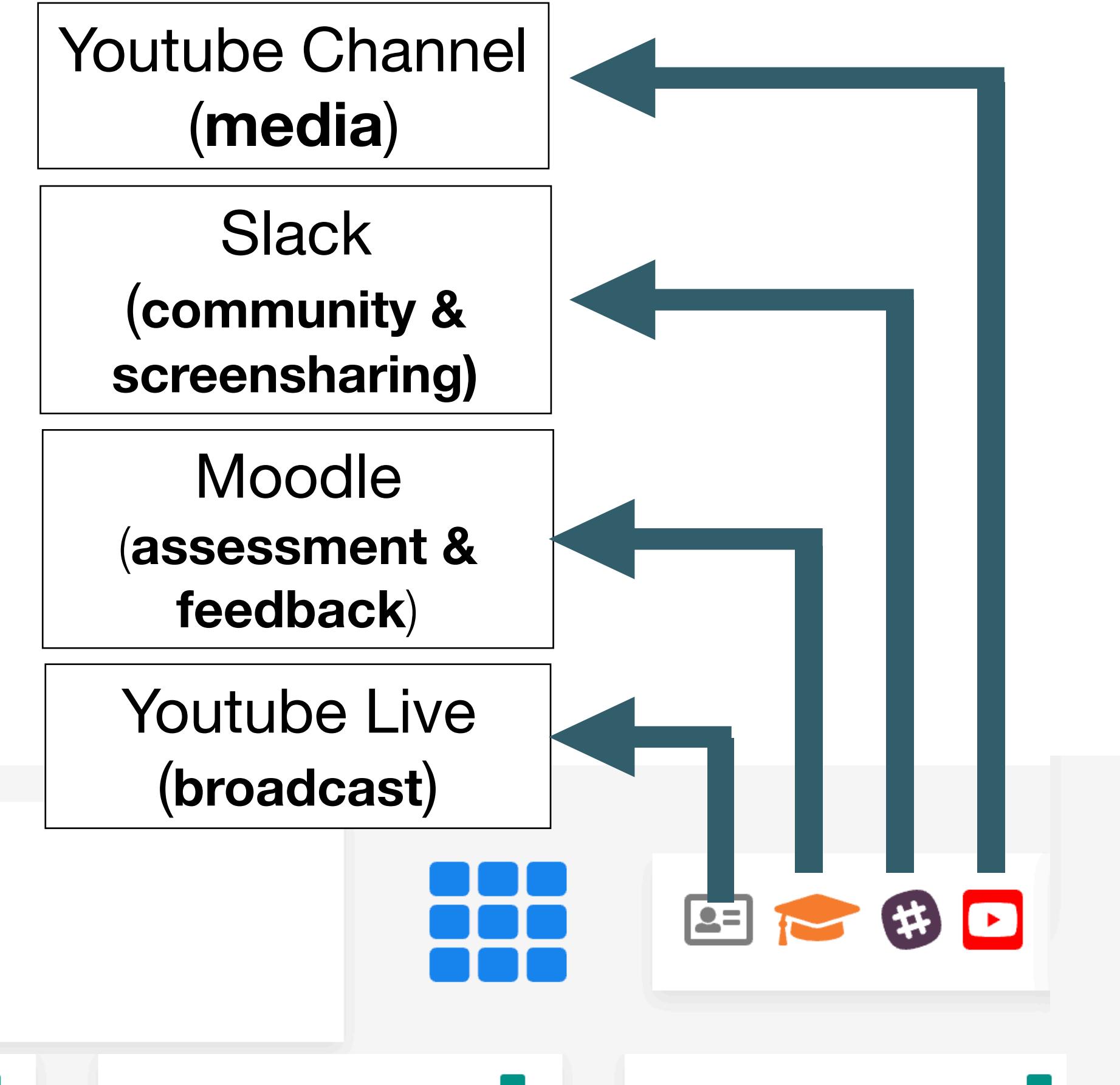
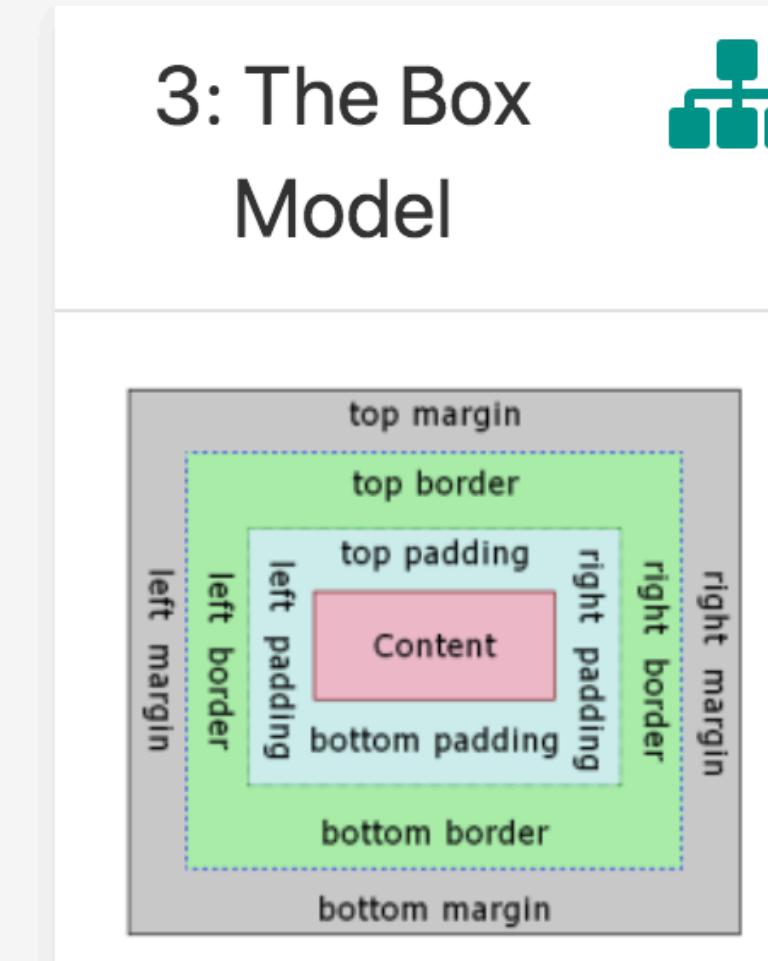


 **tutors-ts**

1: Introducing HTML



2: Introducing CSS

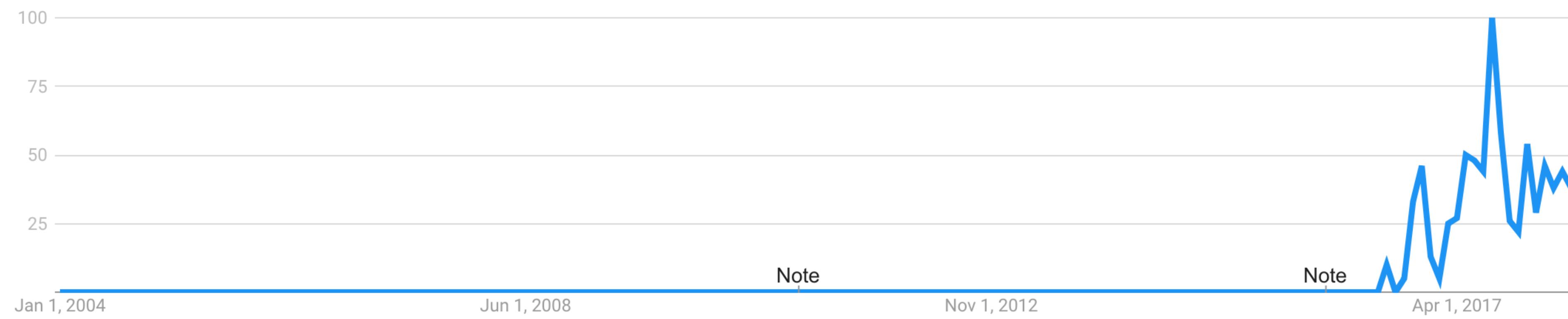


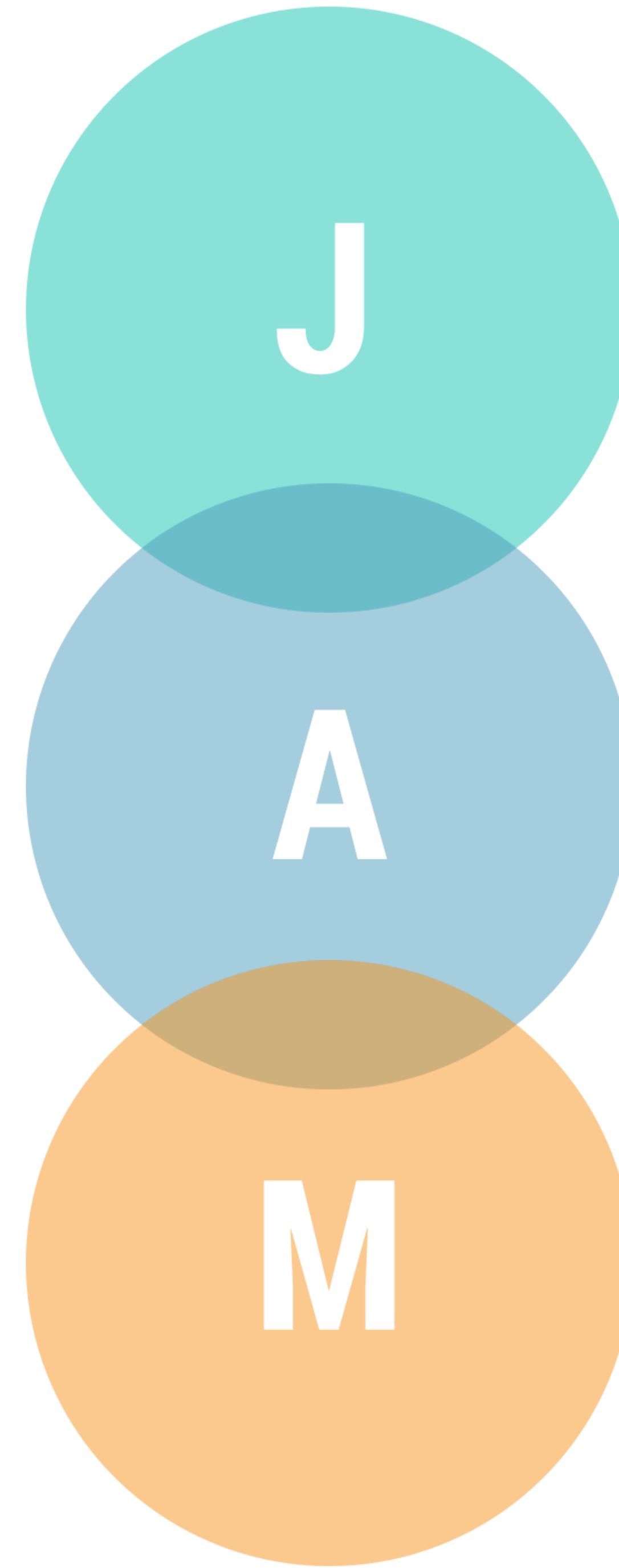
Next Steps....

From Static Site Generators to...

JAMStack  
(google it!)

# JAMStack





## 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

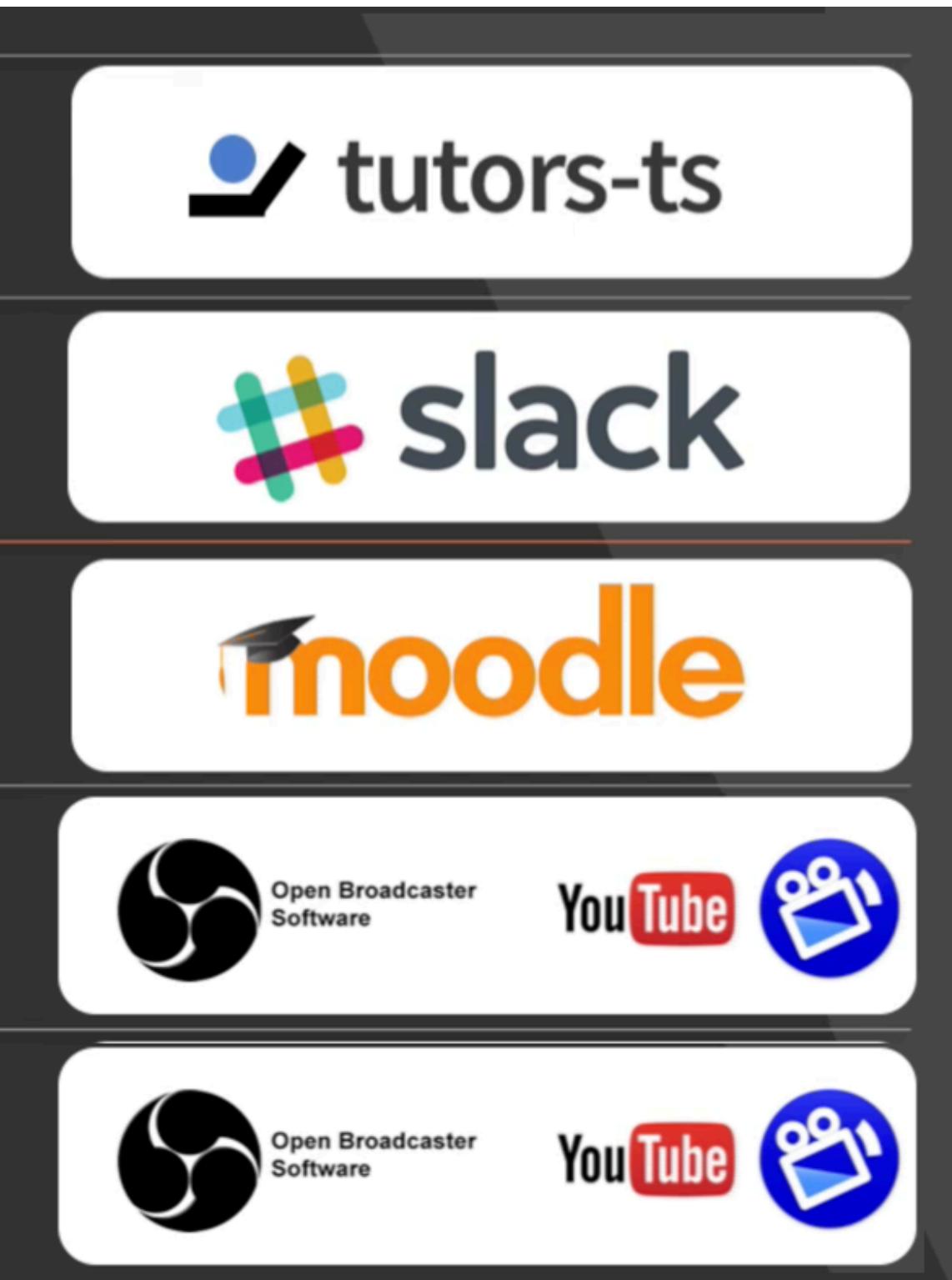
All server-side processes or database actions are abstracted into reusable APIs, accessed over HTTP with JavaScript. These can be custom-built or leverage third-party services.

## 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



JAM<sup>STACK</sup>

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

# TutorStack

Statically generated  
Instructional Content + Learning Support Services →

