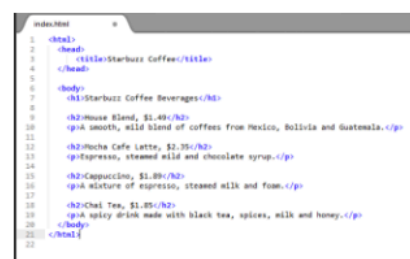


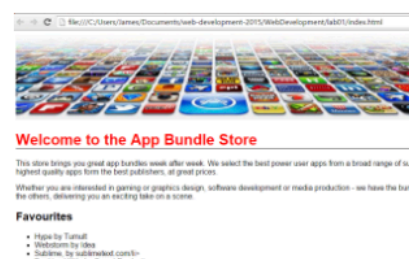
Static Site Generators for Instructional Content

Lab-1.1: Editing HTML



editor · files · text · tags · browser · reference

Lab-1.2: HTML Structure



folders · head · title · link · image · references

Lab-2 CSS Intro

Module Review



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

Lab-3a Layout

Project 1 Specification



Specification for Assignment 1

Project 2 Specification



Specification for Assignment 2

Module Overview



calendar · topics · assessments · schedule

Lab-3b Multicolumn



header · maincontent · navigation · footer · primary · secondary

Lab-4a Navigation



nav · tab · current · padding · border · menu · li · ul

The Nature of the Web



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

HTML Basics



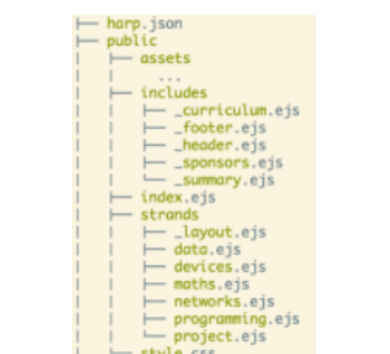
elements · attributes · paths · absolute · relative · nesting · block · inline · labs

HTML Elements



structural elements · elements · text element · elements · image element · shared attributes

Lab-5b Templating



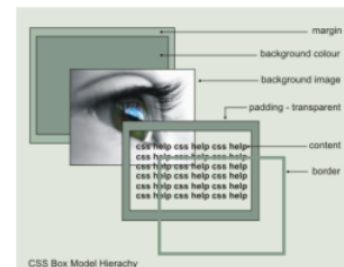
Rebuild the IoT web site from the last lab using templating. This version of the site will aim to significantly reduce the

Lab-5c Templating & Nav



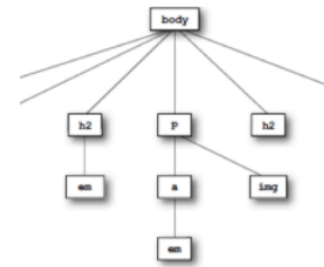
Rework the tabbed navigation site from lab 04 to use EJS template

CSS Rules



combining rules · combining selectors · class · id · div

CSS Cascade



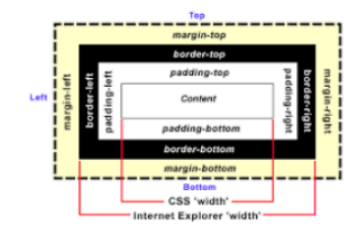
rules · class · elements · explicit · inheritance · default · most specific · order

Web Standards Evolution



standards · w3c · whatwg · sgm1 · xhtml · html5 ·

Box Model Fundamentals



content · padding · border · margin · variations · fonts · dimensions

TutorStack

1. Instruction Materials



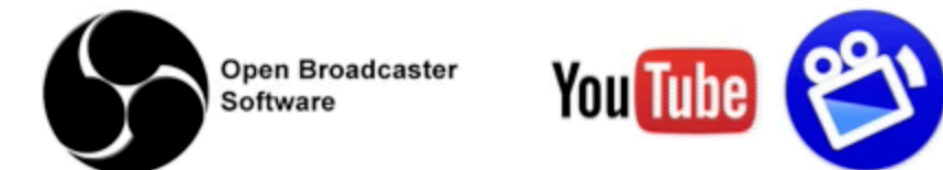
2. Community & screen sharing



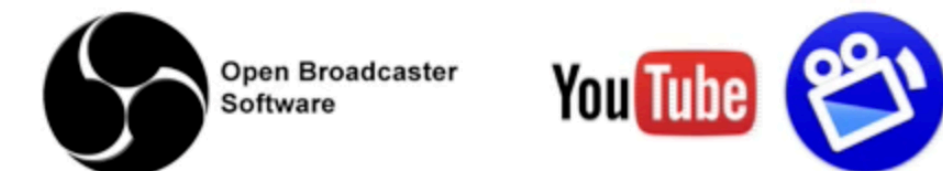
3. Assessment & Feedback



4. Media



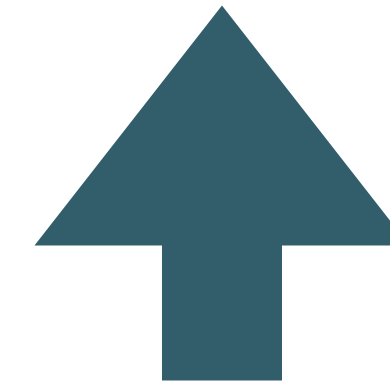
5. Broadcast



TutorStack

1. Instruction Materials

 tutors-ts





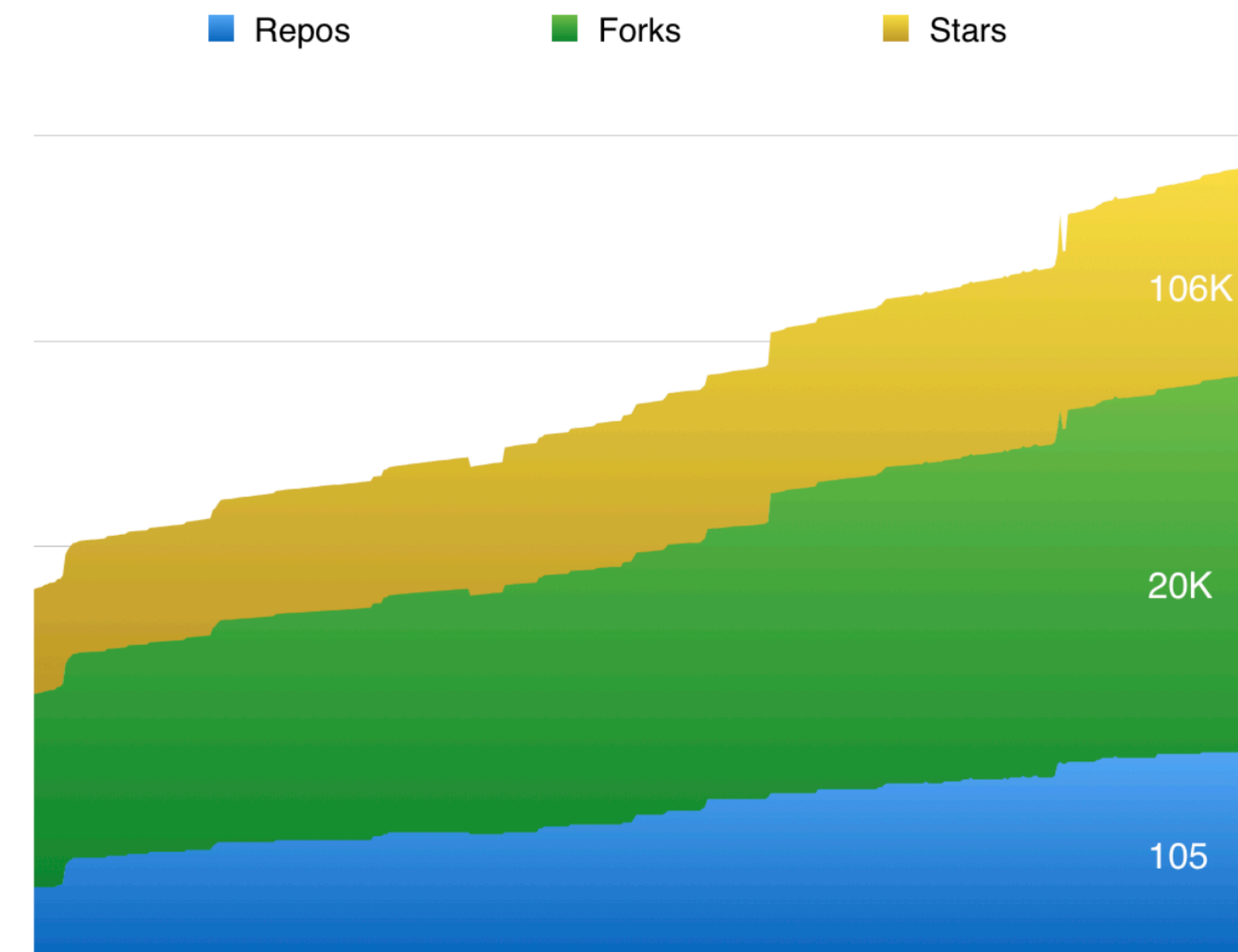
At [StaticGen](#), our open-source directory of **static website generators**, we've kept track of more than a hundred generators for more than a year now, and we've seen both the volume and popularity of these projects take off incredibly on GitHub during that time, going from just 50 to more than 100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

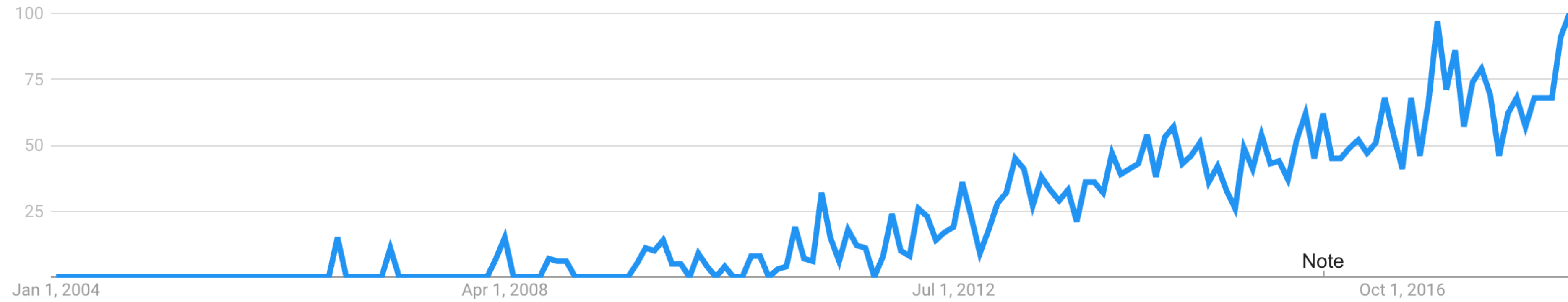
NOVEMBER 2, 2015 • [99 COMMENTS](#)

Why Static Site Generators Are The Next Big Thing

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



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

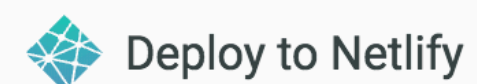
★	ⓘ	🔗	🐦
34352	114	7572	6340
+163	-6	+37	+17

A simple, blog-aware, static site generator.

Languages: Ruby

Templates: Liquid

License: MIT



Hugo

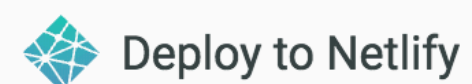
★	ⓘ	🔗	🐦
25860	270	3126	3739
+405	+14	+32	+97

A Fast and Flexible Static Site Generator.

Languages: Go

Templates: Go

License: Apache 2.0



Next

★	ⓘ	🔗	🐦
25537	280	2462	N/A
+513	+29	+91	

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 (edelestar@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



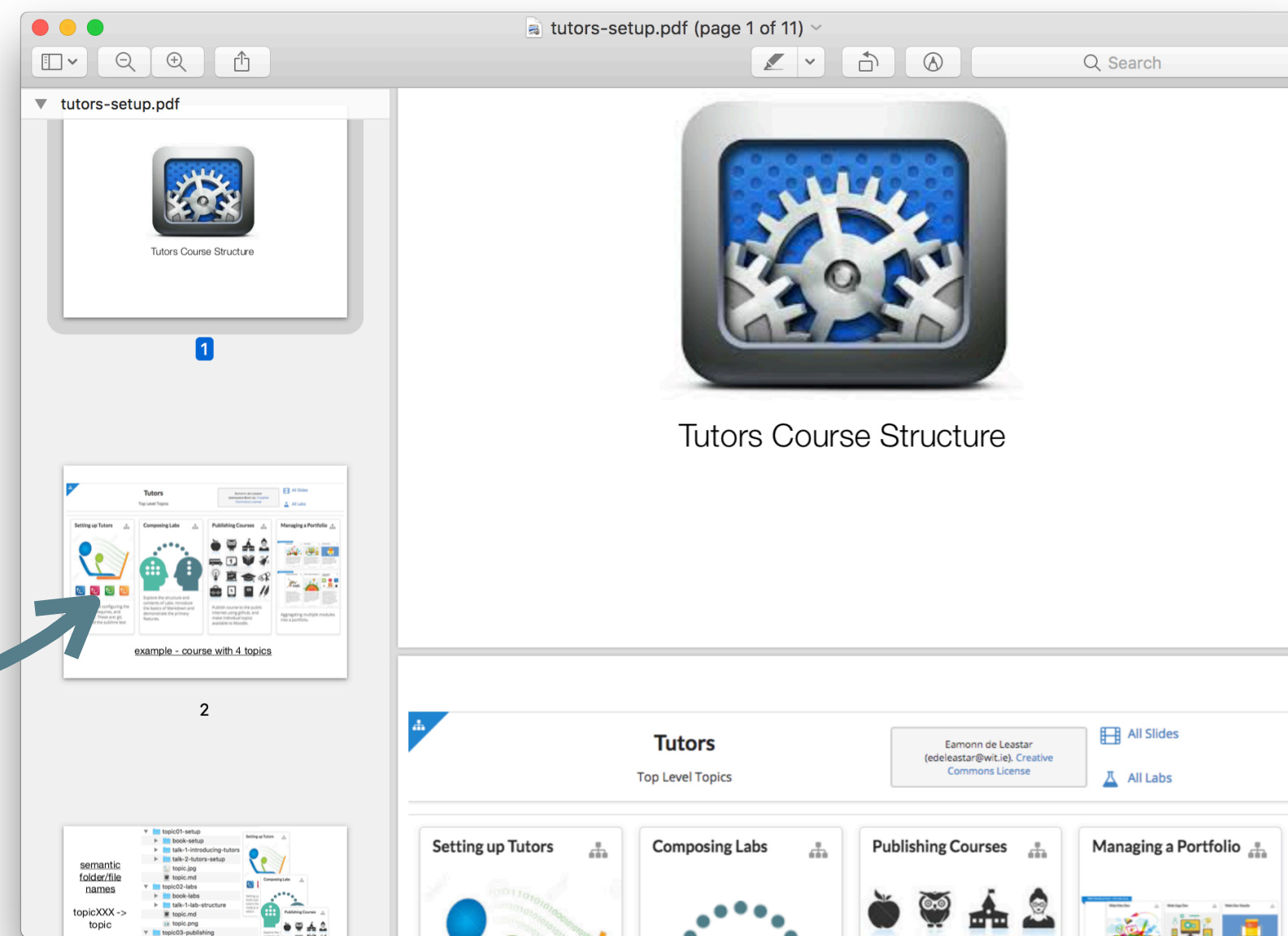
Install the tutors command line application and take for a first spin.

1 topic = slides + lab

Tutors Course Structure



A tour of the structure of a typical tutors course.

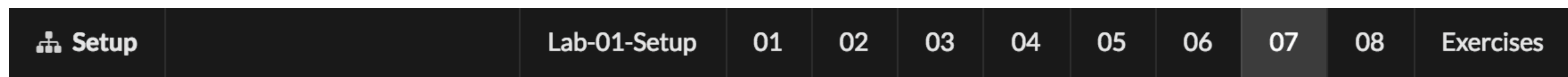


Talk (slides)

Lab-01-Setup



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



Generate a Course

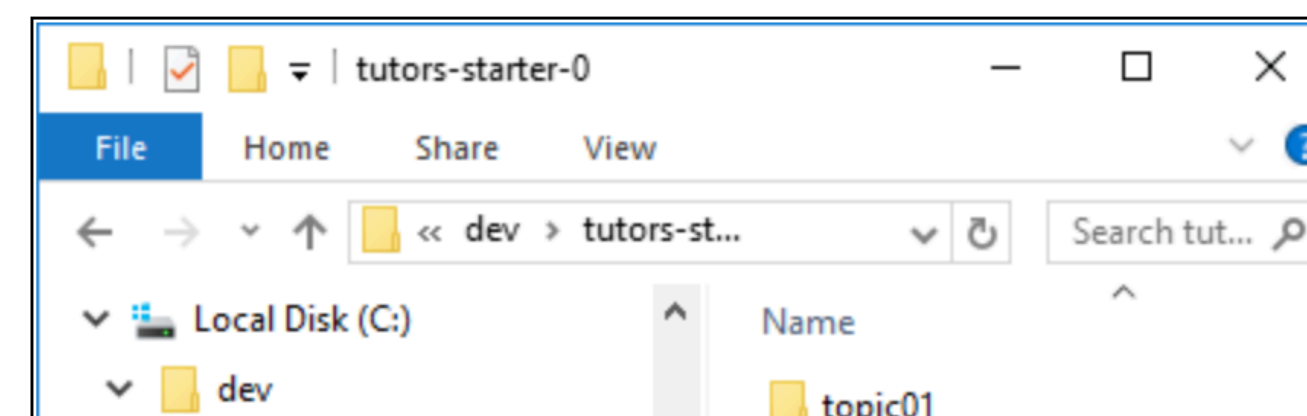
The simplest way to get started with tutors is to use the `new` command to generate a template course:

```
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:



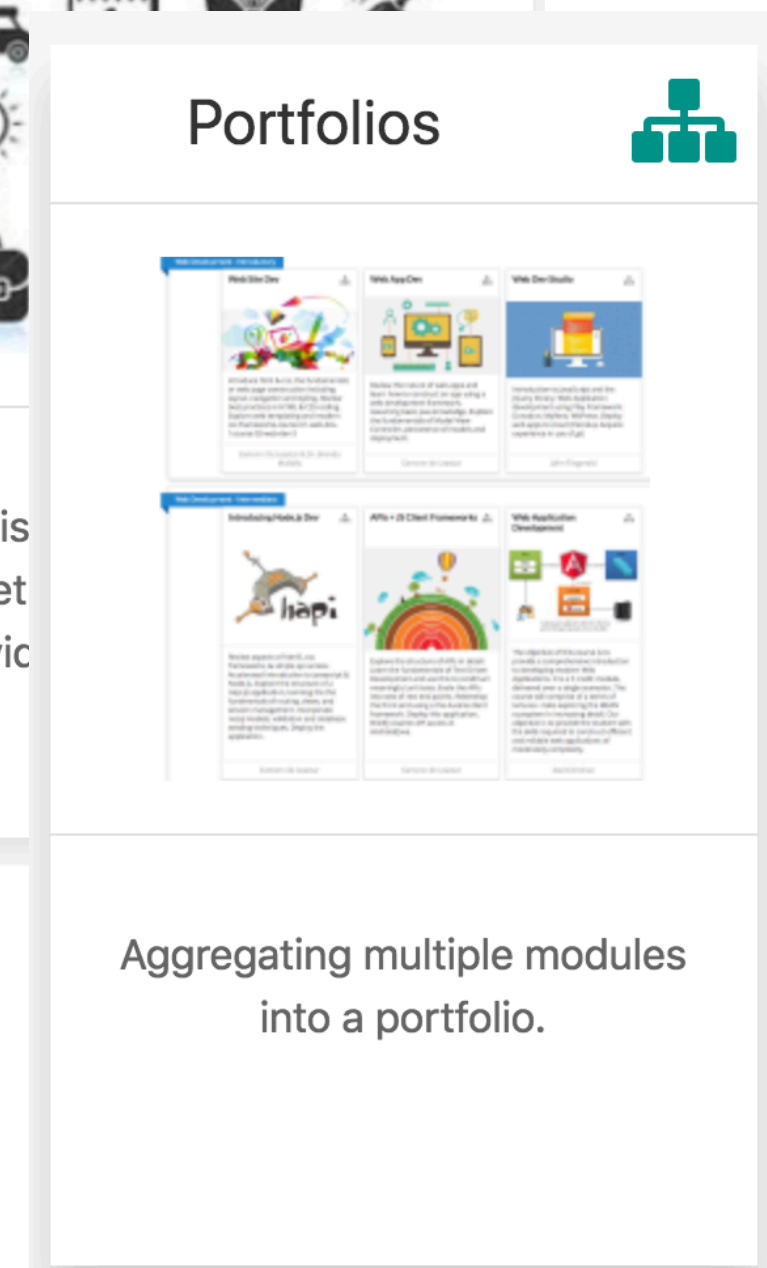
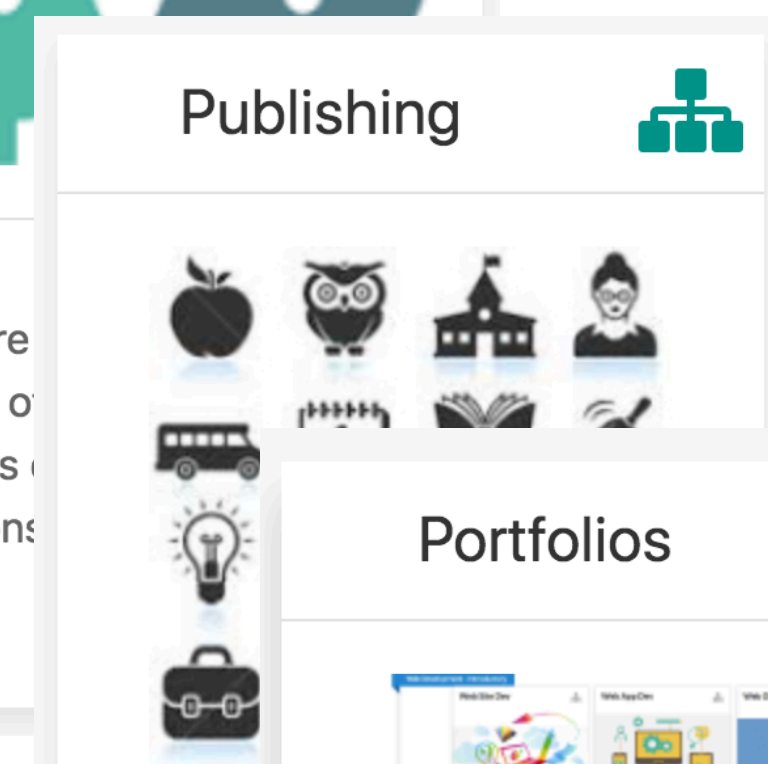
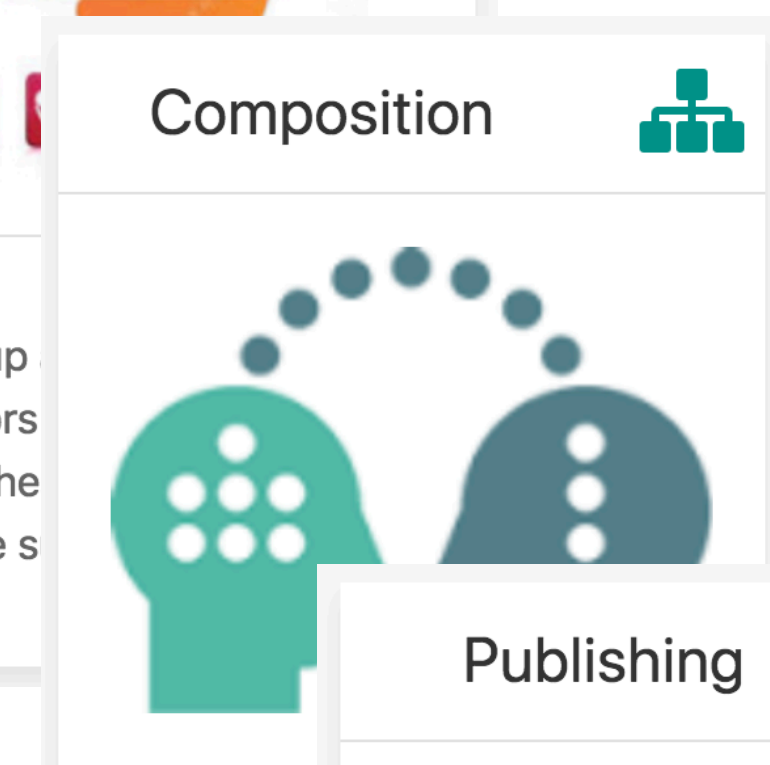
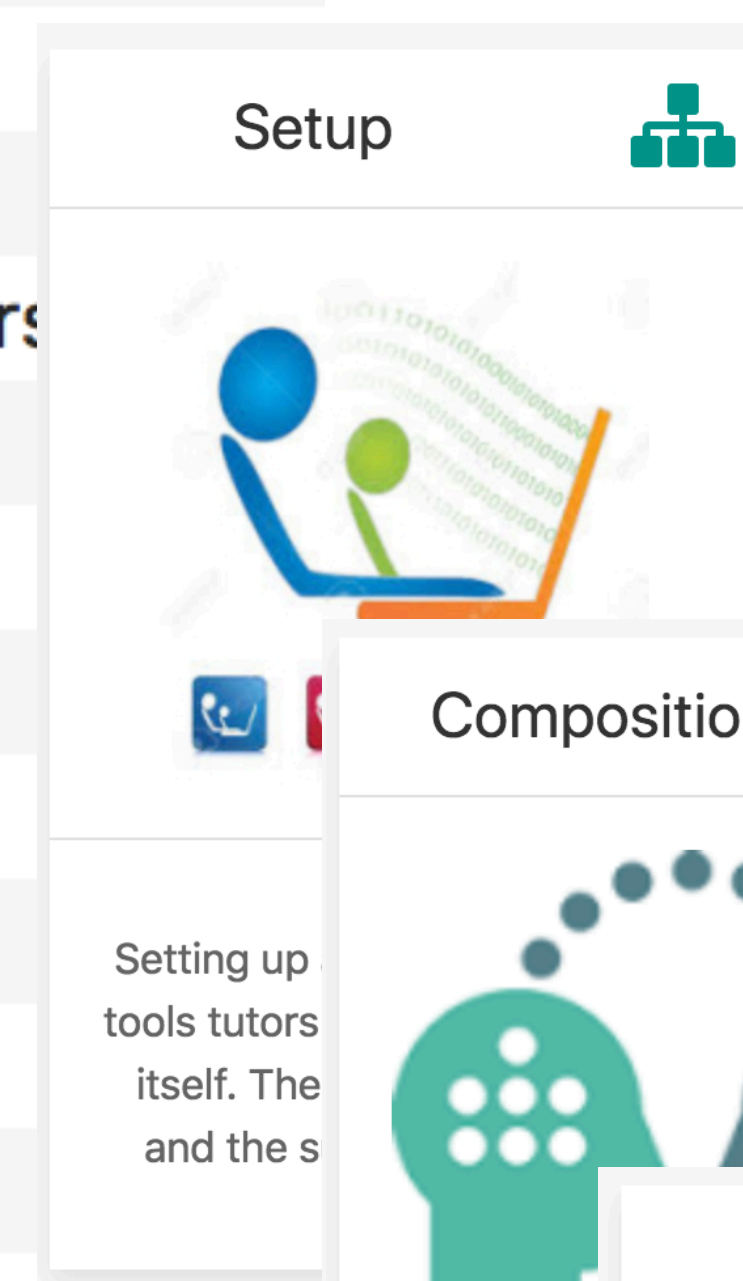
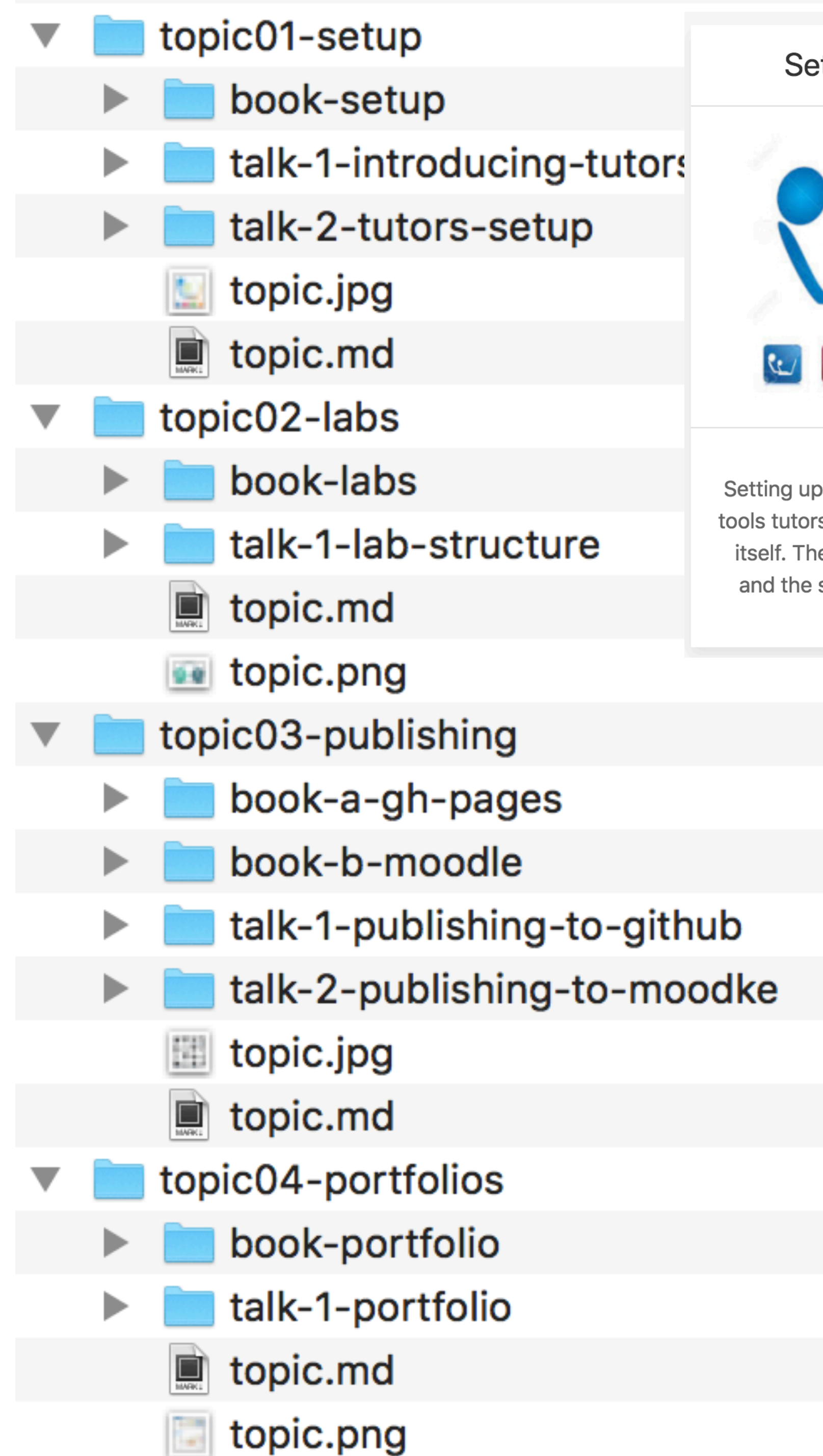
Lab (guides)

Semantic
folder/file
names


topicXXX ->
topic

bookXXX ->
lab



talkXXX ->
talk




Setup



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


- ▼ **topic01-setup**
 - ▶ **book-setup**
 - ▶ **talk-1-introducing-tutors**
 - ▶ **talk-2-tutors-setup**
 -  **topic.jpg**
 -  **topic.md**

Introducing Tutors




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

Tutors Course Structure



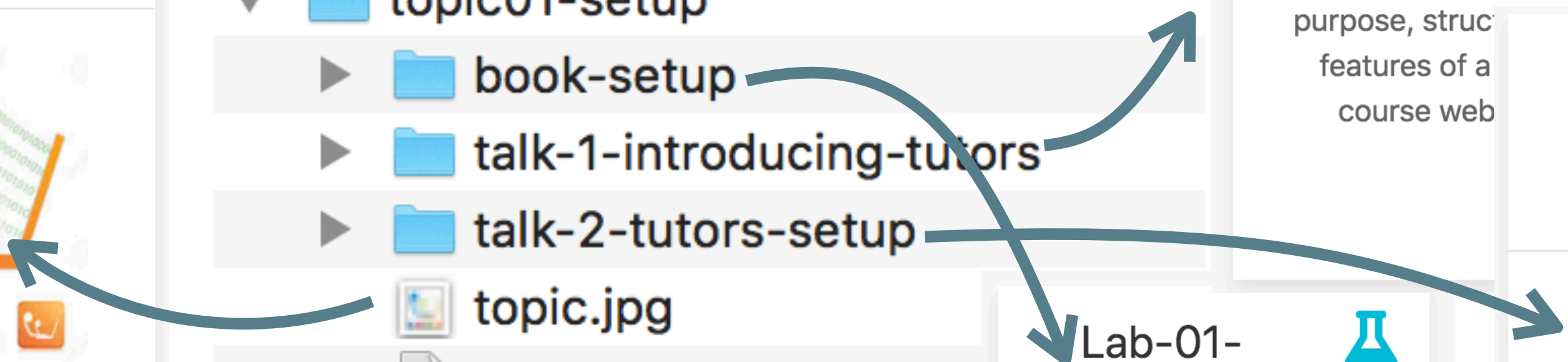
A tour of the structure of a typical tutors course.

Lab-01-Setup

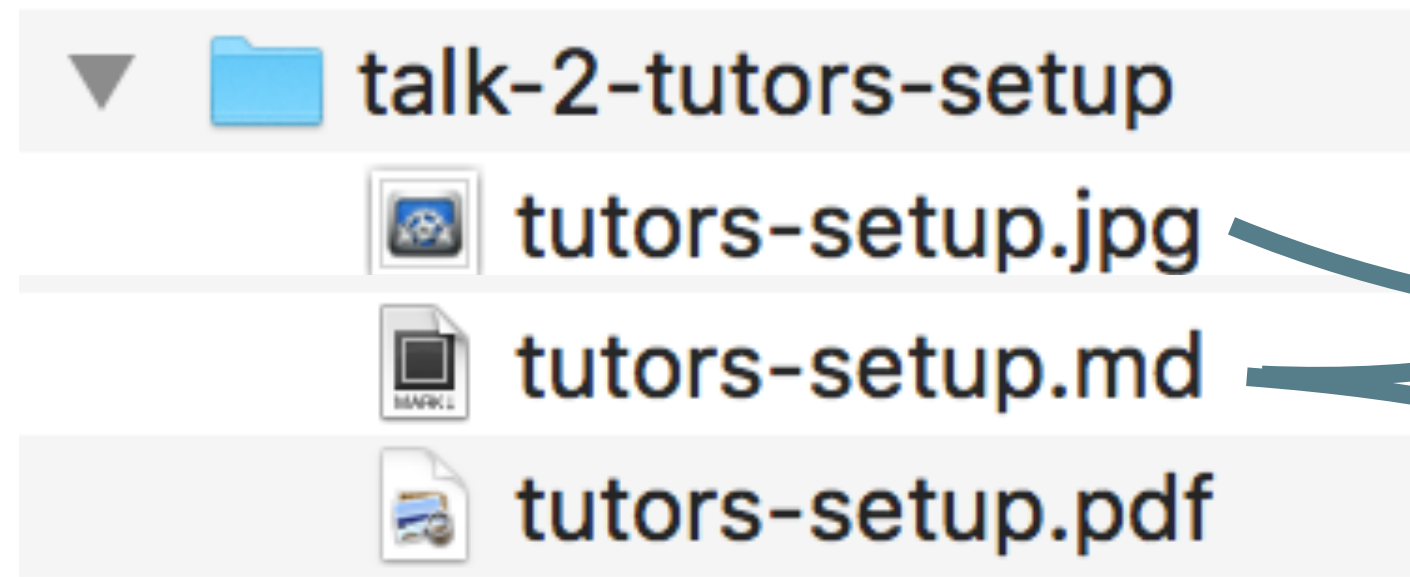


Install the tutors command line application and take for a first spin.


Topic structure



Talk structure



Tutors Course Structure



A tour of the structure of a typical tutors course.

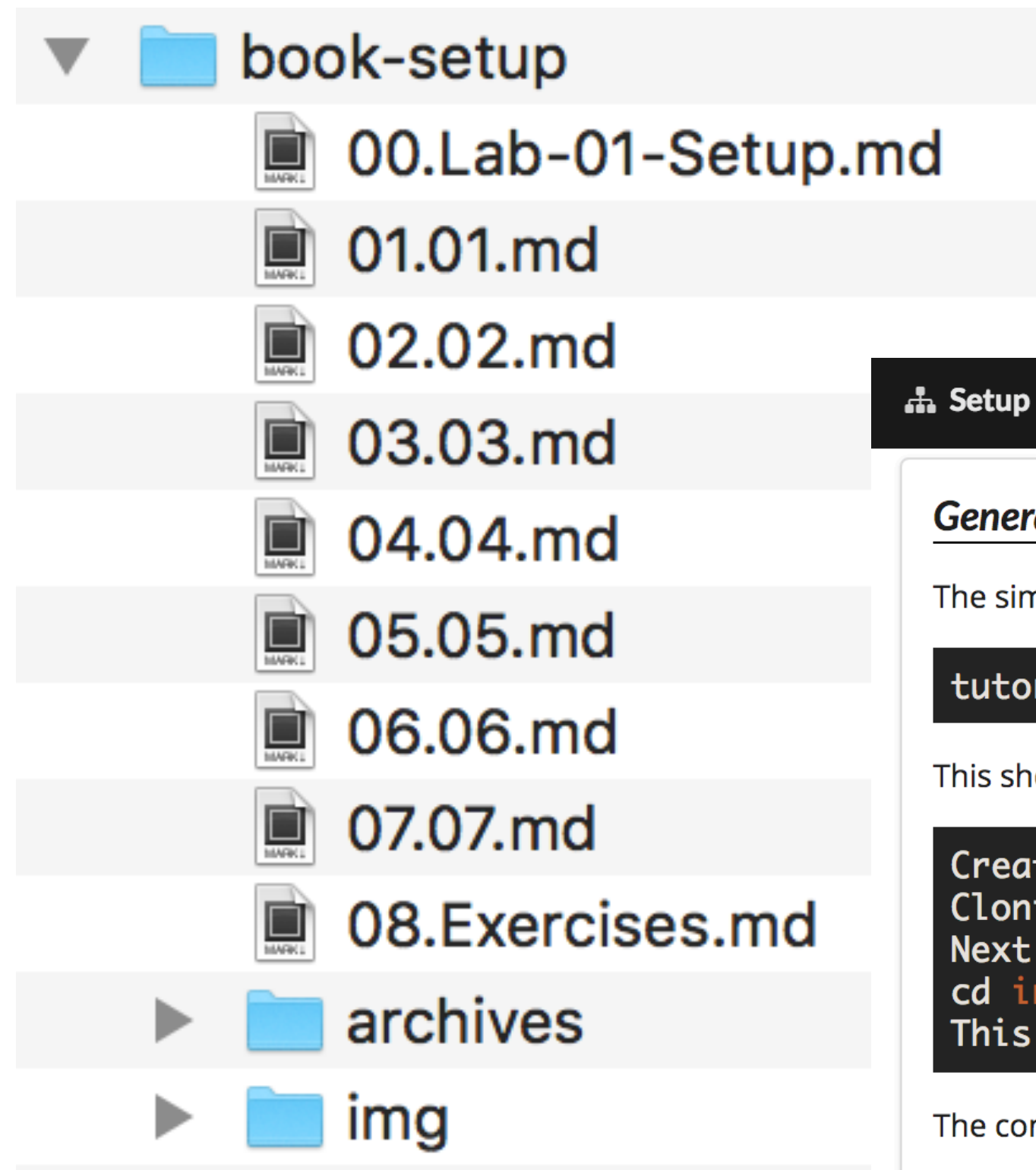
all files
same name

- image
- pdf
- markdown

Tutors Setup

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

Lab structure



One *md* file per step

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:

tutors-starter-0

File Home Share View

Local Disk (C:) > dev > topic01

img

- images used in lab

archives

- zipped archive linked to in labs

Generate a Course

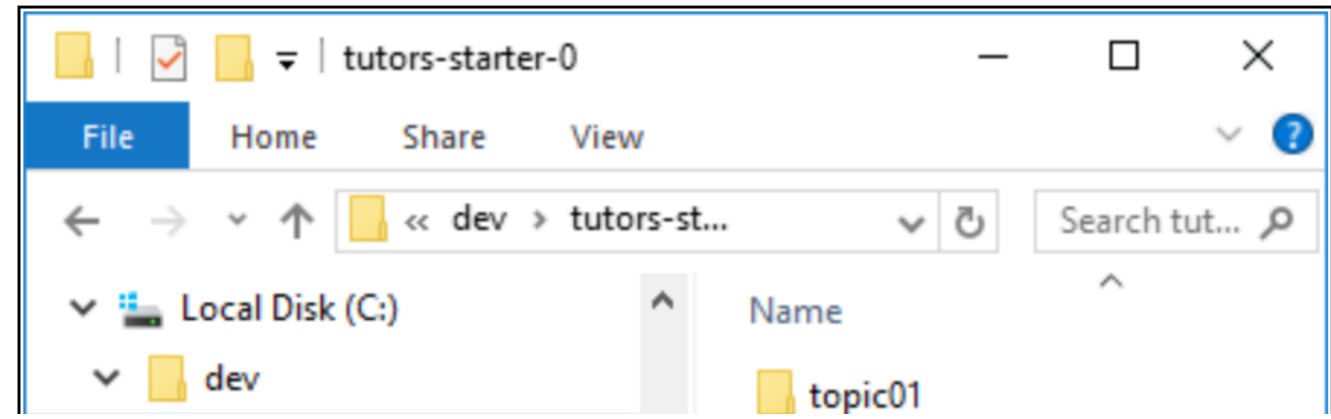
The simplest way to get started with tutors is to use the `new` command to generate a template

```
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



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:

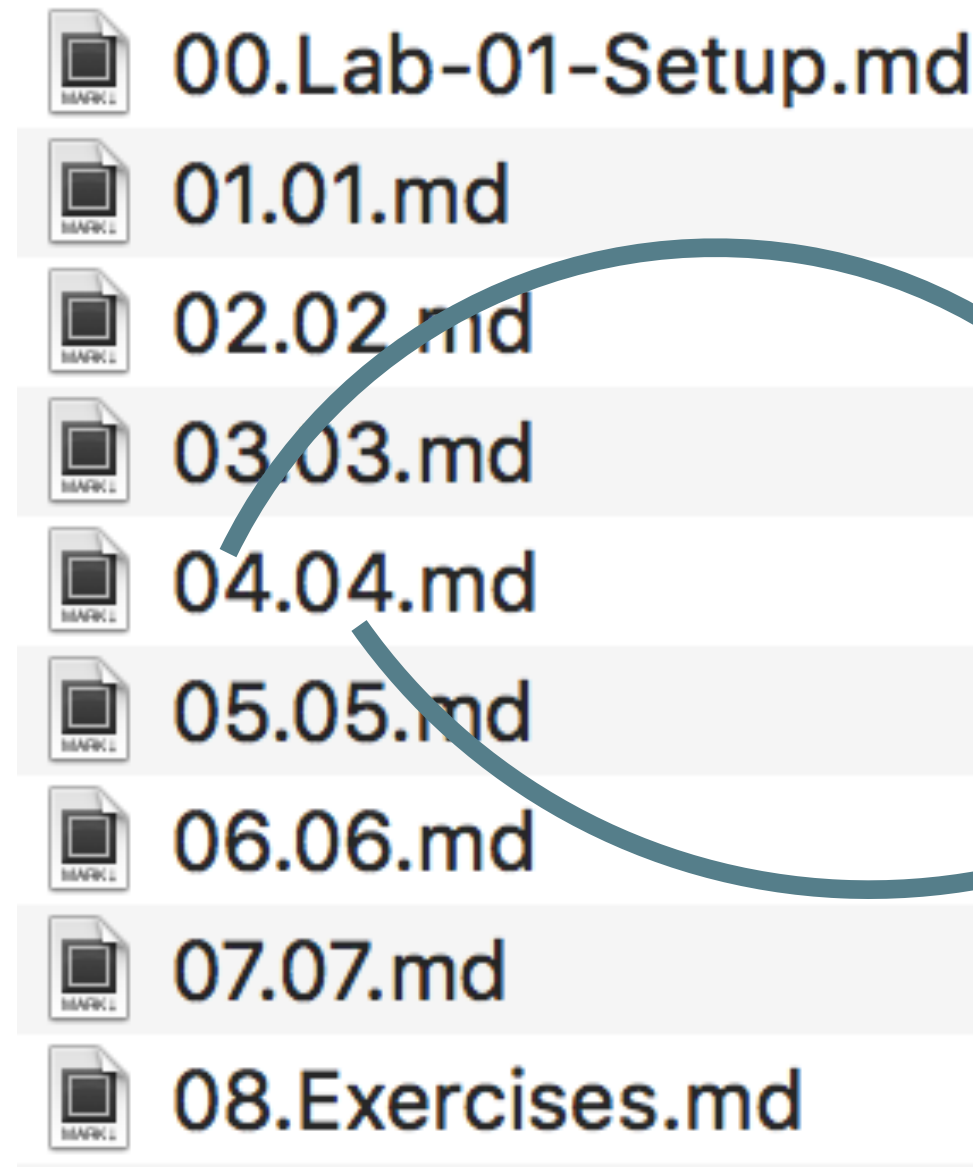
```
~~~
tutors new
~~~
```

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.YY.md

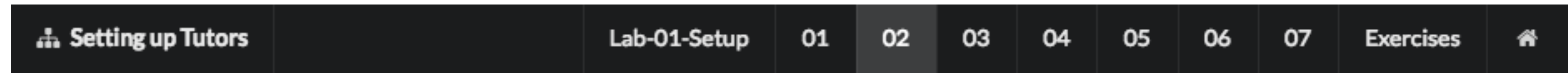
XX

- 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

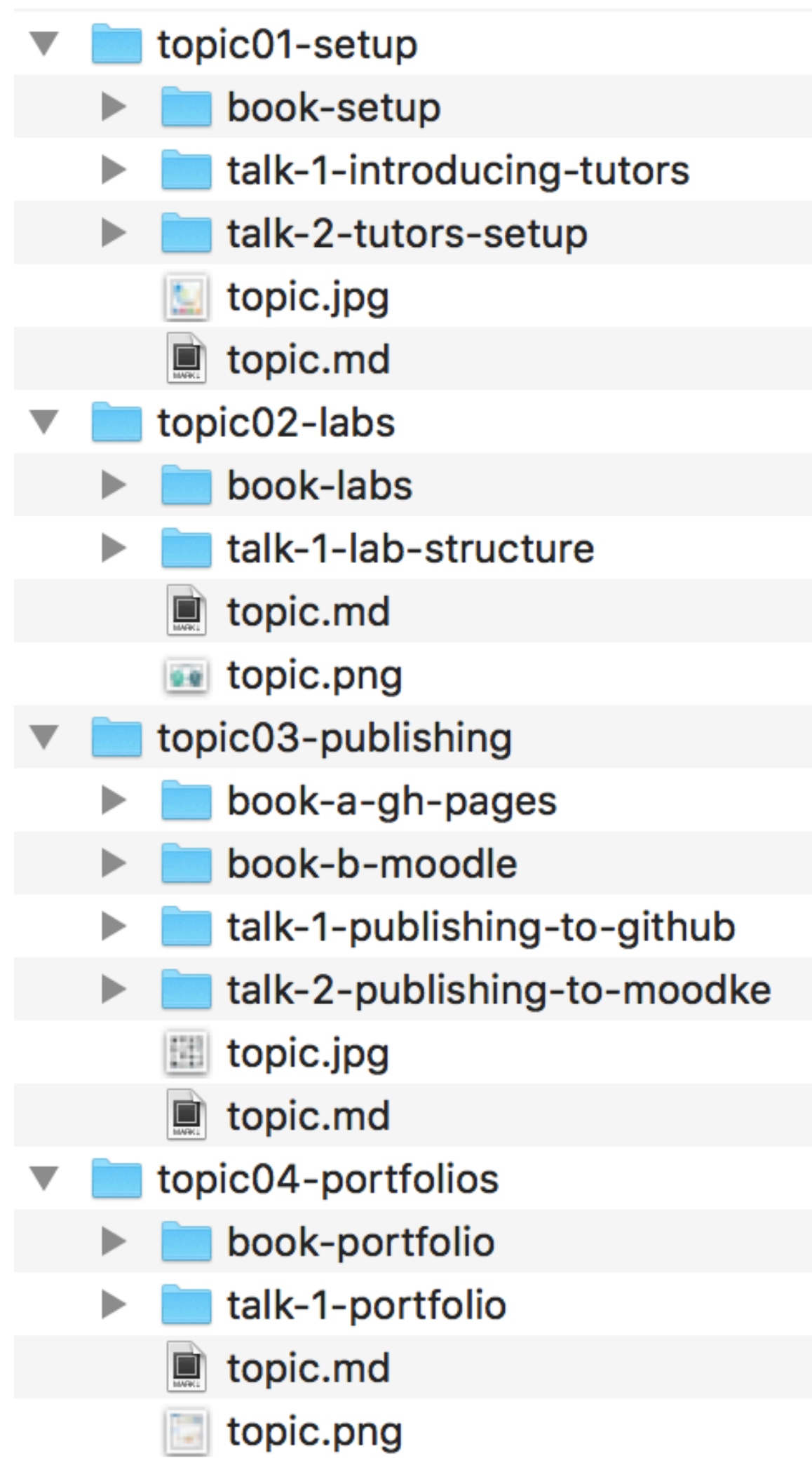
First step, includes lab name

YY is used as step name in the menubar

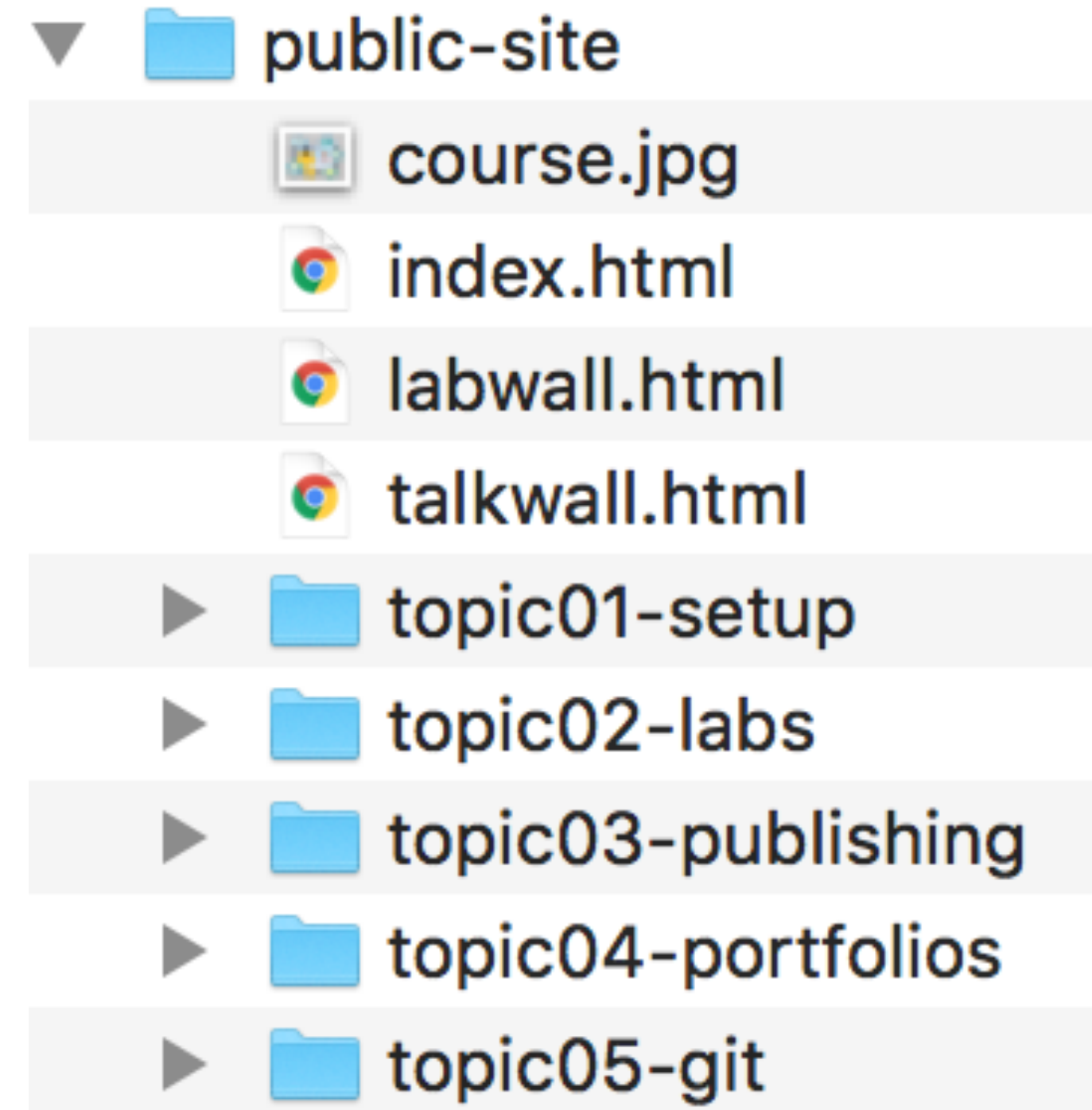

Link to course home page

'tutors' command generates to 'public-site' folder

```
1. bash 🔔
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$
```



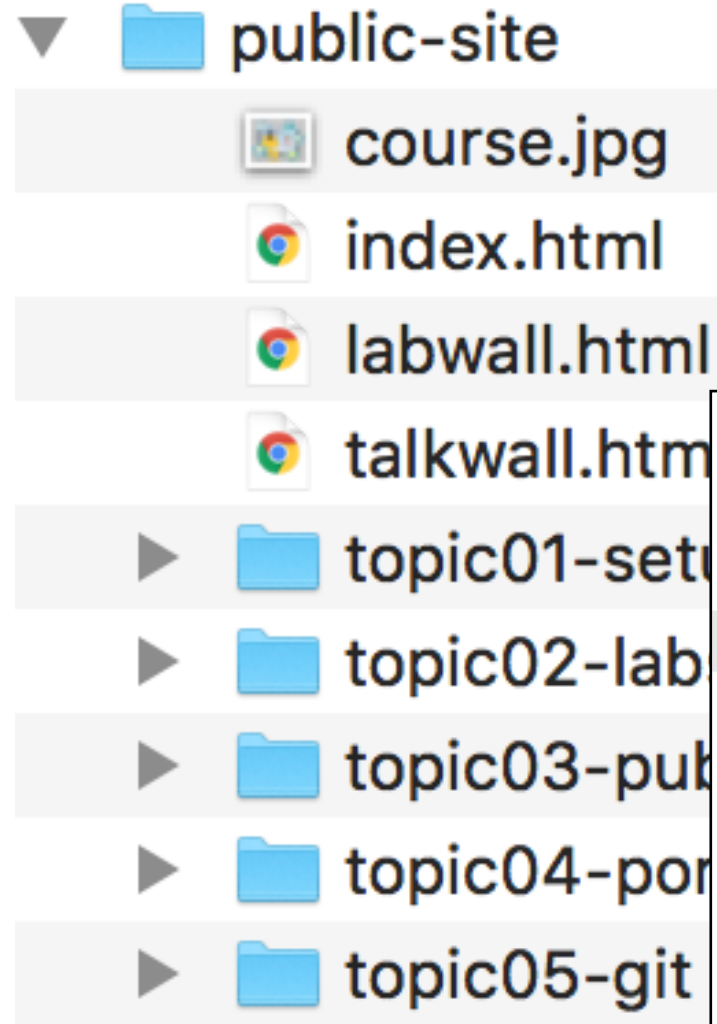
tutors
command



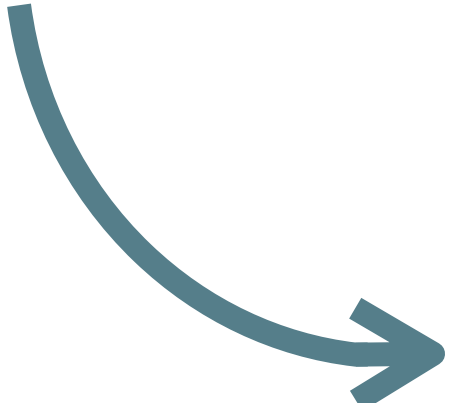
public-site is a 'static' web site

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

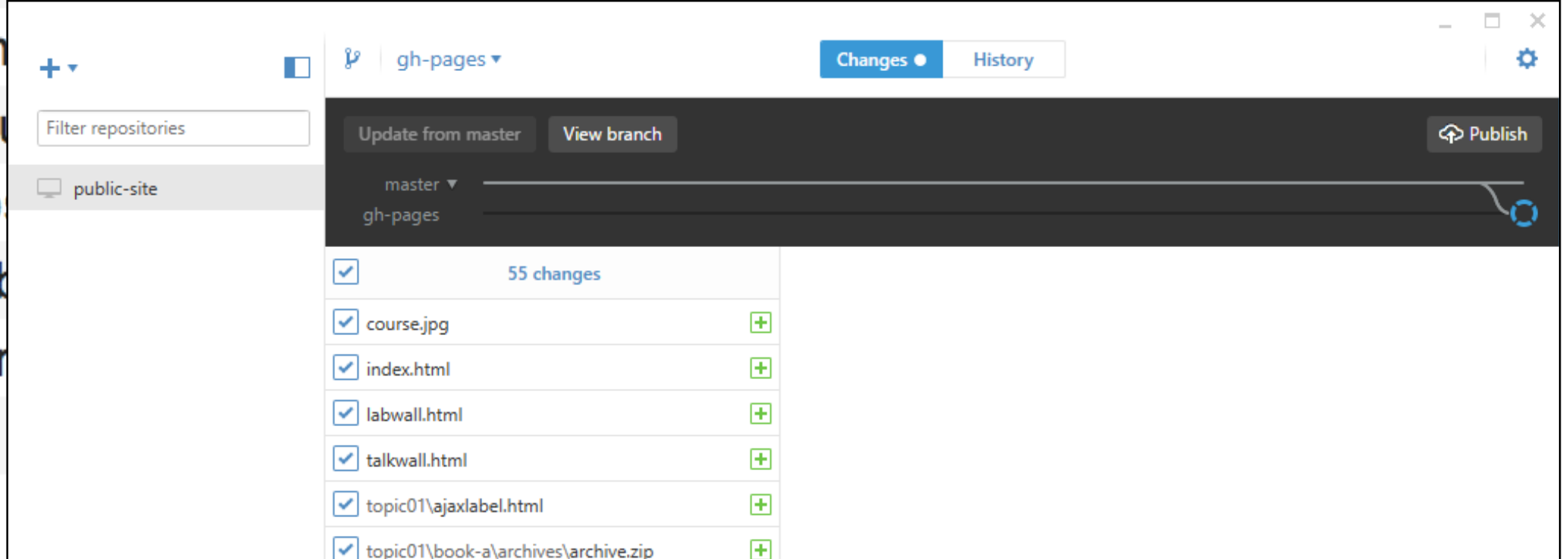
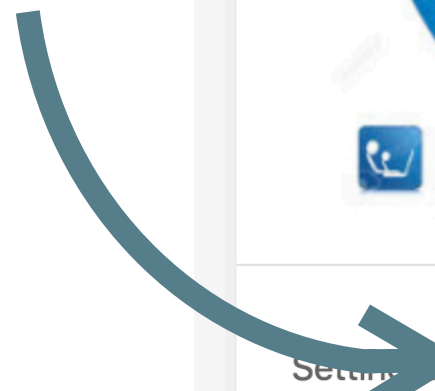
Publishing



folder



git desktop



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.

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

Higher Diploma in Computer Science 2018
Department of Computing & Mathematics, WIT

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



editor · files · text · tags · browser · reference

Lab-1.2: HTML Structure



folders · head · image · ref

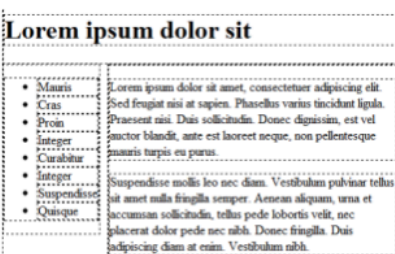
Lab-2 CSS Intro



Lab-3a Layout



Lab-3b Multicolumn



header · maincontent · navigation · footer · primary · secondary

Lab-4a Navigation



nav · tab · current · border · me

Lab-5b Templating



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

Lab-5c Templating Nav



Rework the tab 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



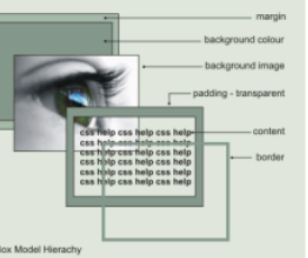
elements · attributes · paths · absolute · relative · nesting · block · inline · labs

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 · shtml · xhtml · html5 ·

3: Box Model

HTML5 Taxonomy & Status on January 20, 2013. W3C Recommendation, Proposed Recommendation, Candidate Recommendation, Last Call, Working Draft, Non-W3C Specifications, Deprecated. Includes a circular diagram of HTML5 related technologies.

4: Navigation, Semantics & Style Guides

Document Type Use HTML5. HTML5 (HTML syntax) is preferred for all HTML documents: <!DOCTYPE html>. Includes a code snippet for the HTML5 doctype.

5: HTML Templates & Deployment

Hosting Company. To get your pages on the Web, you need a server that actually lives on the Web full-time. Find a hosting company and let them worry about the details of keeping a server running. Includes an image of a server room.

6: CSS Frameworks

Popular Frameworks. Includes images for Bootstrap, PureCSS, and Foundation frameworks.

TutorStack

Statically
generated
Instructional
Content

+ Learning
Support Services

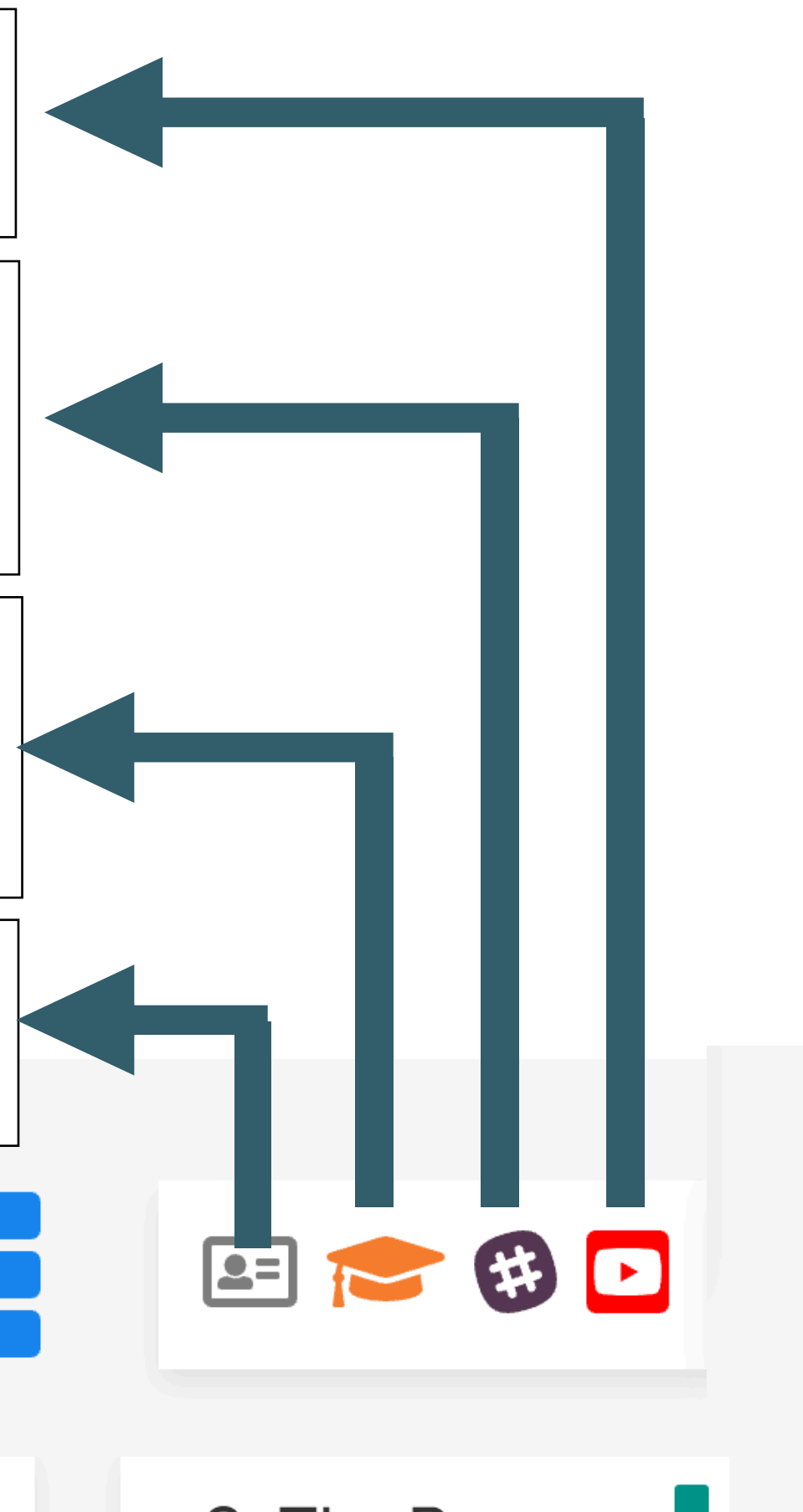


Youtube Channel
(media)

Slack
(community &
screensharing)

Moodle
(assessment &
feedback)

Youtube Live
(broadcast)

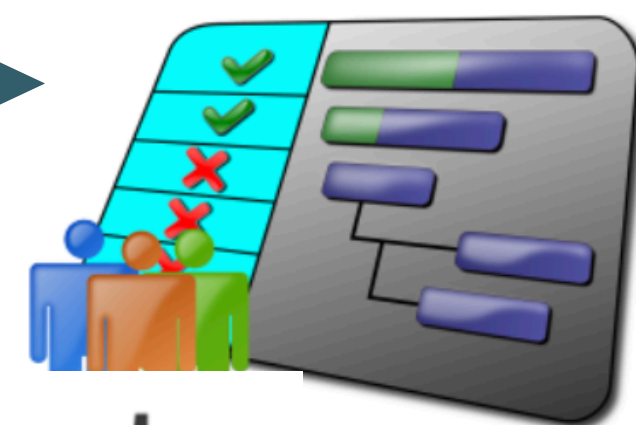


Web Development

Eamonn de Leastar, WIT



0:
Assignments



tutors-ts

specifications ·

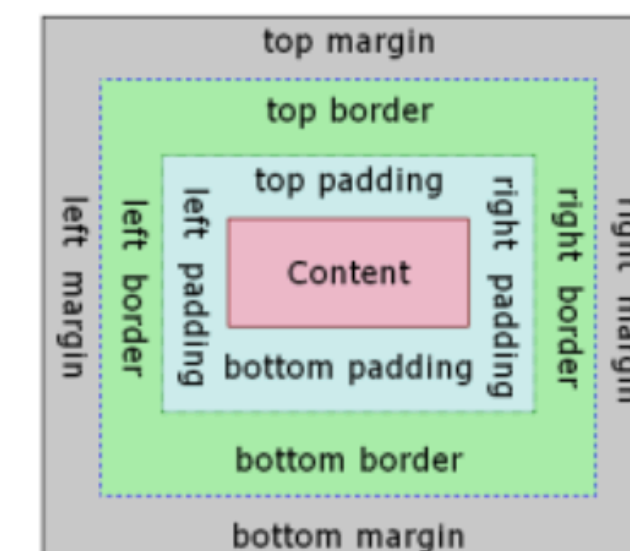
1:
Introducing
HTML



2:
Introducing
CSS



3: The Box
Model

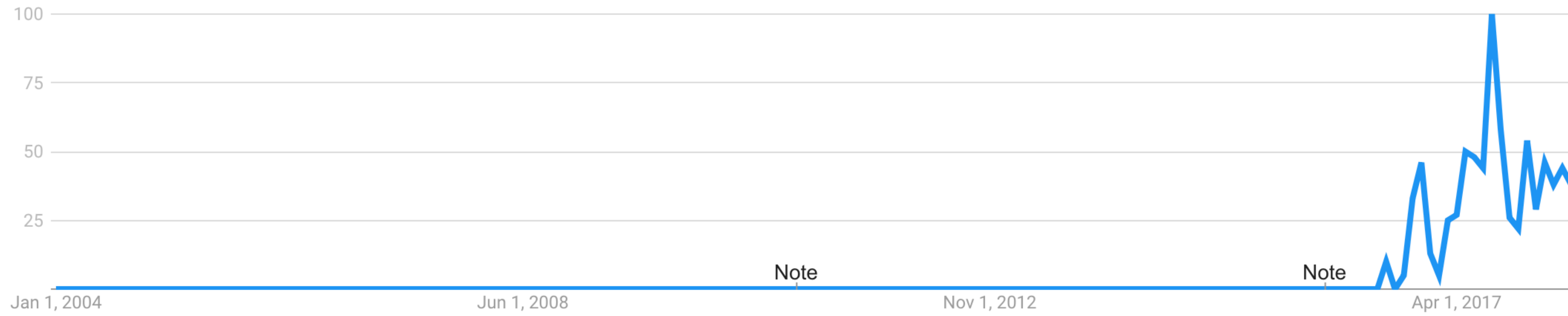


Next Steps....

From Static Site Generators to...

JAMStack
(google it!)

JAMStack



Google Trends



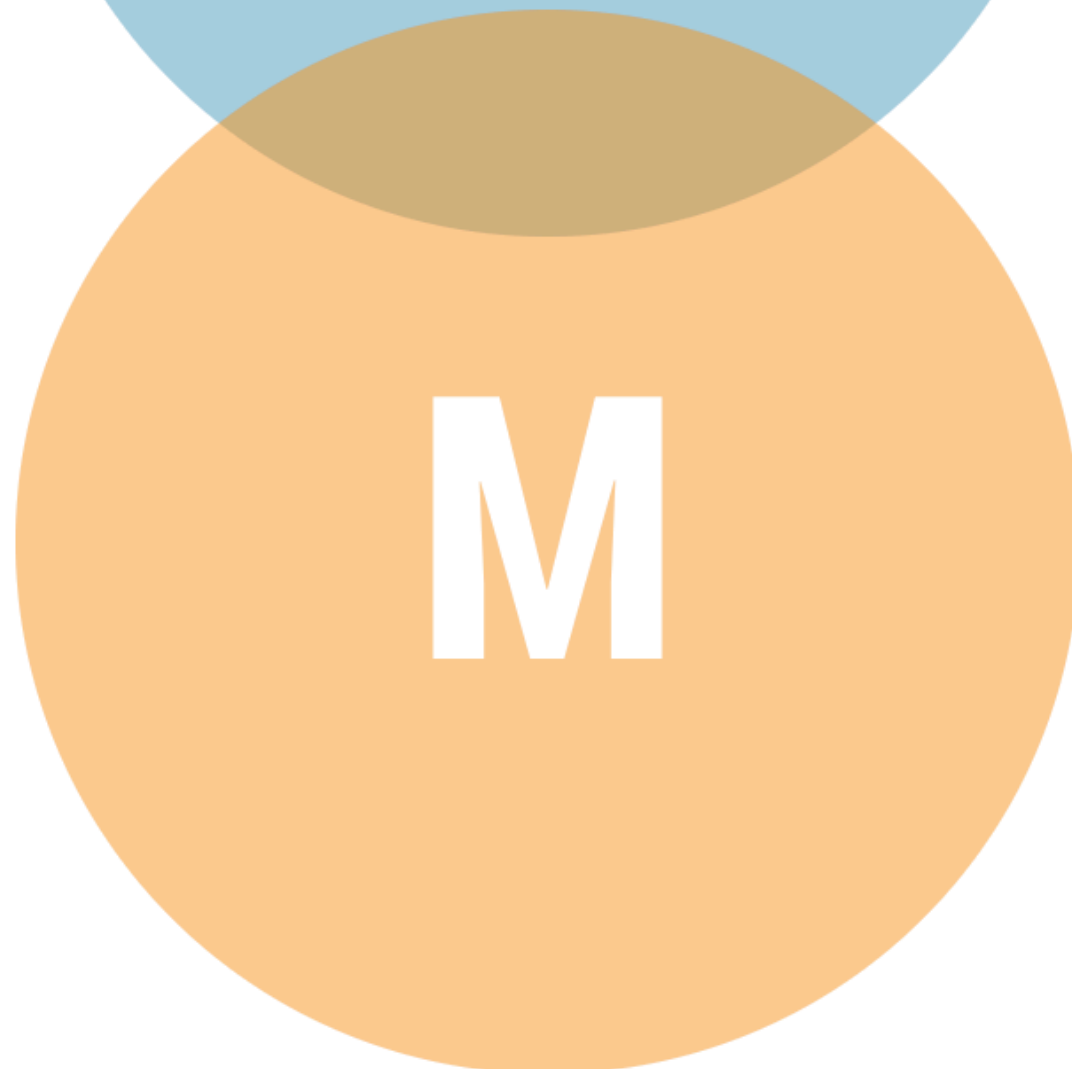
JavaScript

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.

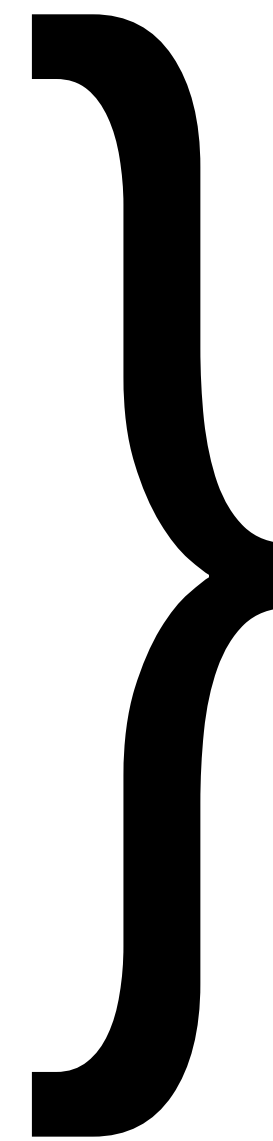
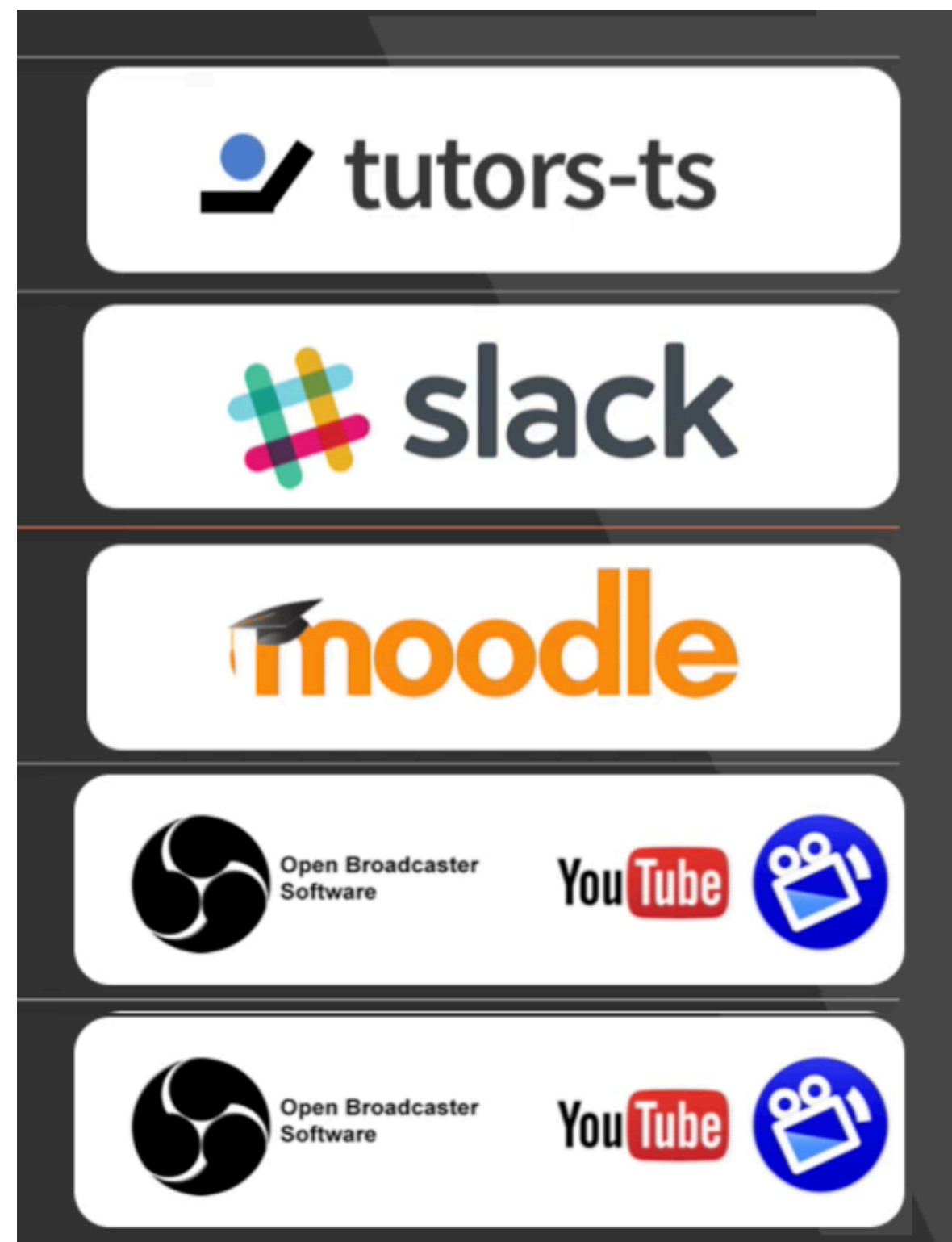


Markup

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)

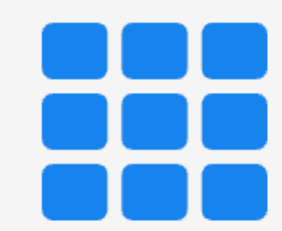
Moodle
(assessment &
feedback)

Youtube Live
(broadcast)

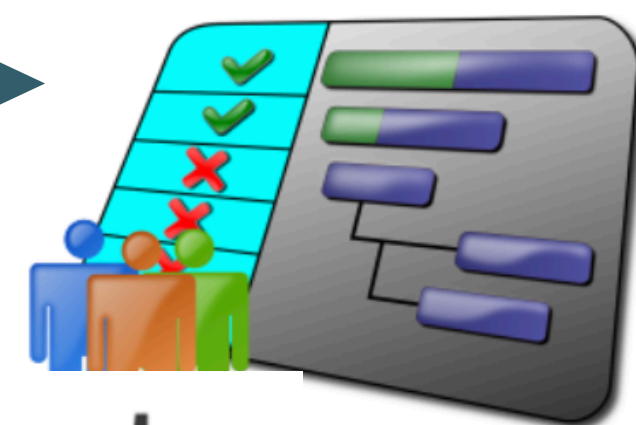


Web Development

Eamonn de Leastar, WIT



0:
Assignments



tutors-ts

specifications ·

1:
Introducing
HTML



2:
Introducing
CSS



3: The Box
Model

