

# Tutors



## Open Web Learning Toolkit

### The Learner Experience



The **Learner Experience** prioritises web interactions that are **engaging, contextual, linkable, searchable, accessible** and **responsive**. In addition the experience should foster a sense of **community** and **connection** among fellow learners.



## What is Tutors?

- A web site?
- A web application?
- A way of creating web sites?
- A way of creating web applications?
- A set of components that can be used in a web site?
- An E-Learning Platform?



Tutors is ...

*The specifications of a set user interface patterns  
for realising a lesson, module or programme.*

*A web based implementation of these patterns in  
a consistent, engaging and accessible manner.*



Tutors Patterns...

## Three Pattern Categories:

- Learning Resources
- Learning Containers
- Learning Navigators



# Tutors Patterns...

- Talk
- Lab
- Video
- Note
- PanelTalk
- PanelNote
- PanelVideo
- Web
- Github
- ProgrammeHome
- ModuleHome
- Topic
- Unit
- Time
- Live
- Card
- BreadCrumb
- Companion
- Filter
- MainNavigator
- TopicNavigator
- ToC

Tutors



TutorsTime



TutorsLive



# Learner

## The Learner Experience



The ***Learner Experience*** prioritises web interactions that are **engaging, contextual, linkable, searchable, accessible** and **responsive**. In addition the experience should foster a sense of **community** and **connection** among fellow learners.

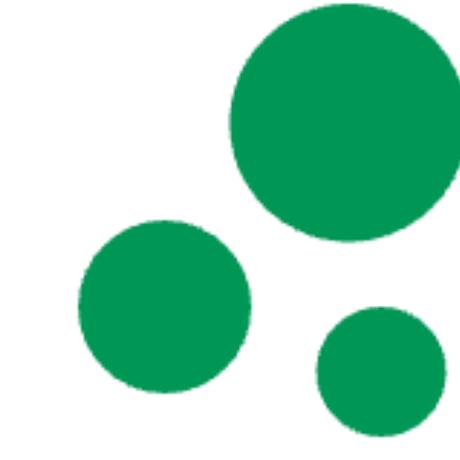
# A programme homepage

Semester One

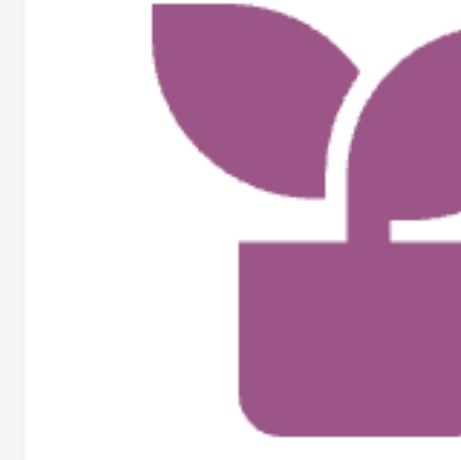
HDIP  
23

**FAQ** 

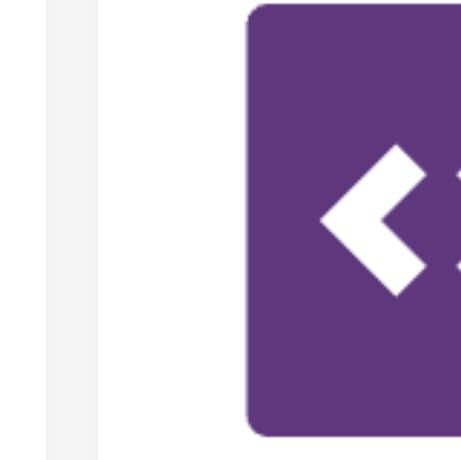
Questions and Answers for new students on the programme

**Workshop One** 

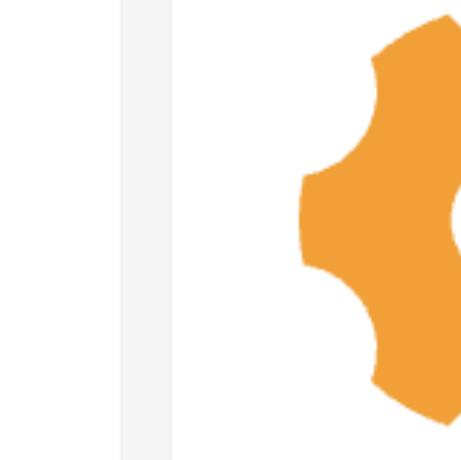
induction · structure · schedules · handbook

**Life etc...** 

time · space · balance · experiences · tips · techniques

**Web Development I** 

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

**Programming** 

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

**Semester 1 Schedule** 

timetables · semester calendars · assessment schedules



Links to individual module  
Home pages

# A course homepage

Full Stack Web Development  
Eamonn de Leastar

Search Layout 0

Home Full Stack Web ...

Overview + Assignments

01: Node

02: Hapi.js

03: Joi

04: TDD

05: Models

06: APIs

07: REST

08: OpenAPI

09: JWT

10: Seeding

11: Deployment & Images

Overview, schedule & assignments.

The fundmendals of node.  
A first node application.

The Hapi application Framework

Validation with Joi

Test Driven Development

Incorporate Mongo Models

Expose an API from an application

Resful API implementations

Documentating OpenAPI/Swagger APIs

Authenticating APIs with JSON Web Tokens

Seeding the Database

Deploying to Monogo Cloud, Heroku & Cloudfinary

Topics - 1 topic/card for each week

# Navigation

The screenshot displays a course management system interface for "Full Stack Web Development" by Eamonn de Leastar. The top navigation bar includes links for "Course Info", "Calendar", "Full Text Search", and "Table of Contents". The main content area shows a timeline of lessons: "Overview + Assignments", "01: Node", "02: Hapi.js", "03: Joi", "04: TDD", and "05: Models". Each lesson card features a title, icon, and brief description. The left sidebar contains "Breadcrumb Navigation" and the right sidebar contains "Wall". The top right corner shows user profile information.

Course Info

Calendar

Full Text Search

Table of Contents

Breadcrumb Navigation

Wall

Layout & Themes

Search

Layout

Profile

Home

Full Stack Web ...

Overview + Assignments

01: Node

02: Hapi.js

03: Joi

04: TDD

05: Models

Companions

Overview, schedule & assignments.

The fundmendals of node. A first node application.

The Hapi application Framework

Validation with Joi

Test Driven Development

Incorporate Mongo Models

# Navigation

Layout & Themes

The screenshot displays a user interface for a "Full Stack Web Development" course by Eamonn de Leastar. At the top, there's a header with a profile icon, the course name, and a search bar. Below the header, a navigation bar shows the current week (9) and a calendar link. On the right side, a dropdown menu titled "Layout & Themes" is open, showing options for "Toggles" (Dark Mode, Compact) and "Themes" (Tutors, Dyslexia). The main content area features a grid of six cards representing different lessons:

- Overview + Assignments**: Overview, schedule & assignments.
- 01: Node**: The fundamentals of node. A first node application.
- 02: Hapi.js**: The Hapi application Framework.
- 03: Joi**: Validation with Joi.
- 04: TDD**: Test Driven Development.
- 05: Incorporate Mongo Models**: Incorporate Mongo Models.

# Theme Support

Layout & Themes

The screenshot displays a dark-themed user interface for a learning platform. At the top, there's a navigation bar with a profile icon, a search bar, and a 'Layout' dropdown. A large downward arrow points from the 'Layout & Themes' text to a 'Themes' section in the sidebar. This sidebar includes a 'Toggles' section with 'Dark Mode' and 'Compact' options, and a 'Themes' section containing 'Tutors' (which is highlighted in teal) and 'Dyslexia'. The main content area shows a grid of course cards. The first card is titled 'Overview + Assignments' and features a 3D character icon. The second card is '01: Node' with a Node.js logo. The third card is '02: Hapi.js' with a Hapi.js logo. The fourth card is '03: Joi' with a Joi logo. The fifth card is '04: TDD' featuring a circular diagram with 'TDD' in the center. The sixth card is partially visible, showing a 3D cube icon.

Full Stack Web Development  
Eamonn de Leastar

Search Layout

0

Search

Layout

Dark Mode

Compact

Toggle Light Mode

Themes

Tutors

Dyslexia

Overview + Assignments

01: Node

02: Hapi.js

03: Joi

04: TDD

05: Incorporate Mongo Models

Validation with Joi

Test Driven Development

Refactor

Write a failing test

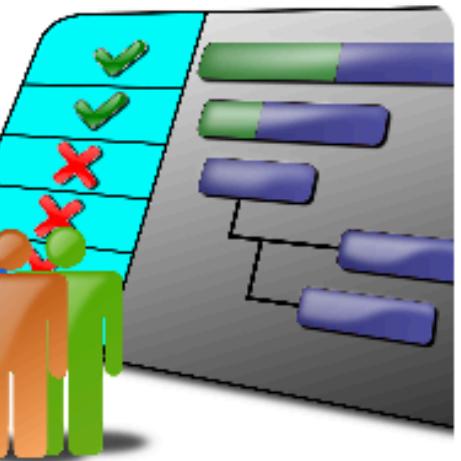
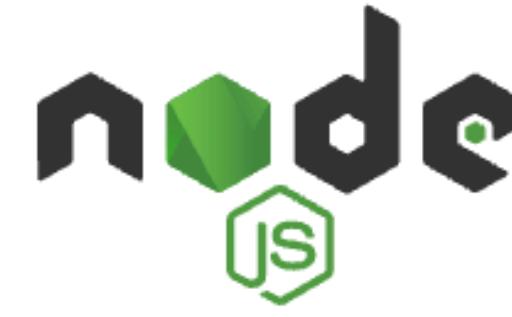
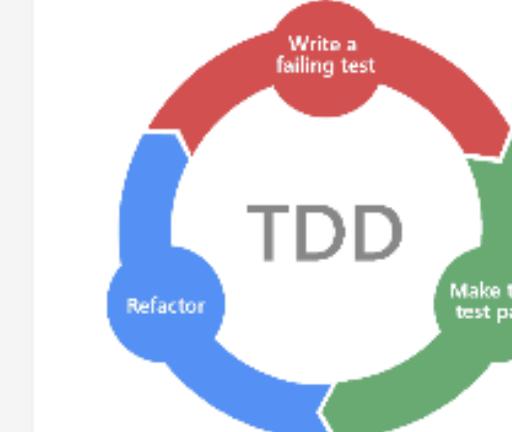
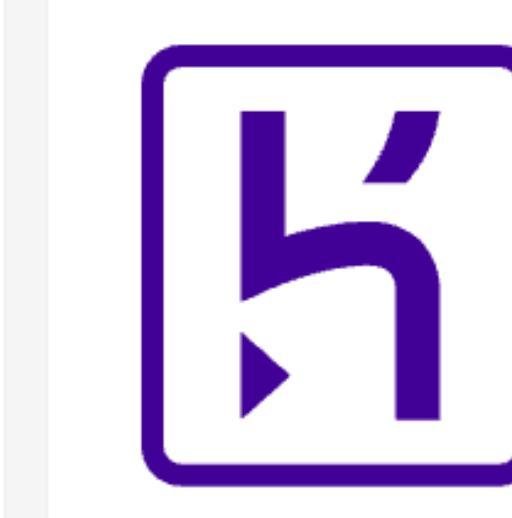
Make the test pass

Overview, schedule & assignments.

The fundmendals of node.  
A first node application.

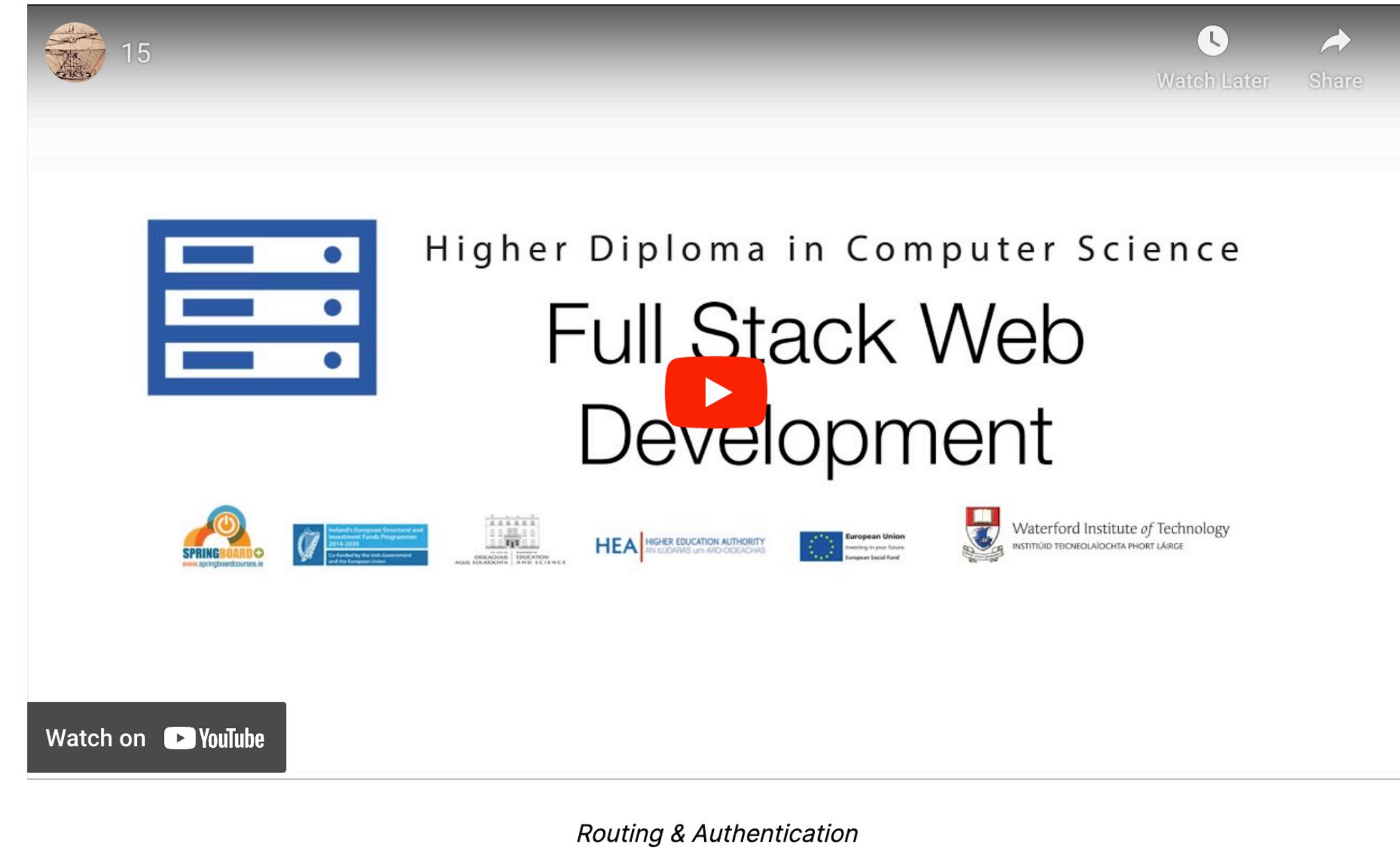
The Hapi application Framework

# Topics - 1 topic/card for each week

<p><b>Overview + Assignments</b></p>  <p>Overview, schedule &amp; assignments.</p>	<p><b>01: Node</b></p>  <p>The fundamentals of node. A first node application.</p>	<p><b>02: Hapi.js</b></p>  <p>The Hapi application Framework</p>	<p><b>03: Joi</b></p>  <p>Validation with Joi</p>	<p><b>04: TDD</b></p>  <p>Test Driven Development</p>	<p><b>05: Models</b></p>  <p>Incorporate Mongo Models</p>
<p><b>06: APIs</b></p>  <p>Expose an API from an application</p>	<p><b>07: REST</b></p>  <p>Resful API implementations</p>	<p><b>08: OpenAPI</b></p>  <p>Documentating OpenAPI/Swagger APIs</p>	<p><b>09 JWT</b></p>  <p>Authenticating APIs with JSON Web Tokens</p>	<p><b>10: Seeding</b></p>  <p>Seeding the Database</p>	<p><b>11: Deployment &amp; Images</b></p>  <p>Deploying to Monogo Cloud, Heroku &amp; Clouinary</p>

# Single Topic

The screenshot shows a top navigation bar with a logo, search icon, layout icon, and user profile. Below it is a breadcrumb navigation path: Home > Full Stack Web ... > 15: Routing & A... > Routing. To the right is a toolbar with icons for file operations.



A list of sub-topics under the main topic "Routing". Each item has a thumbnail, title, and a brief description. The sub-topics are:

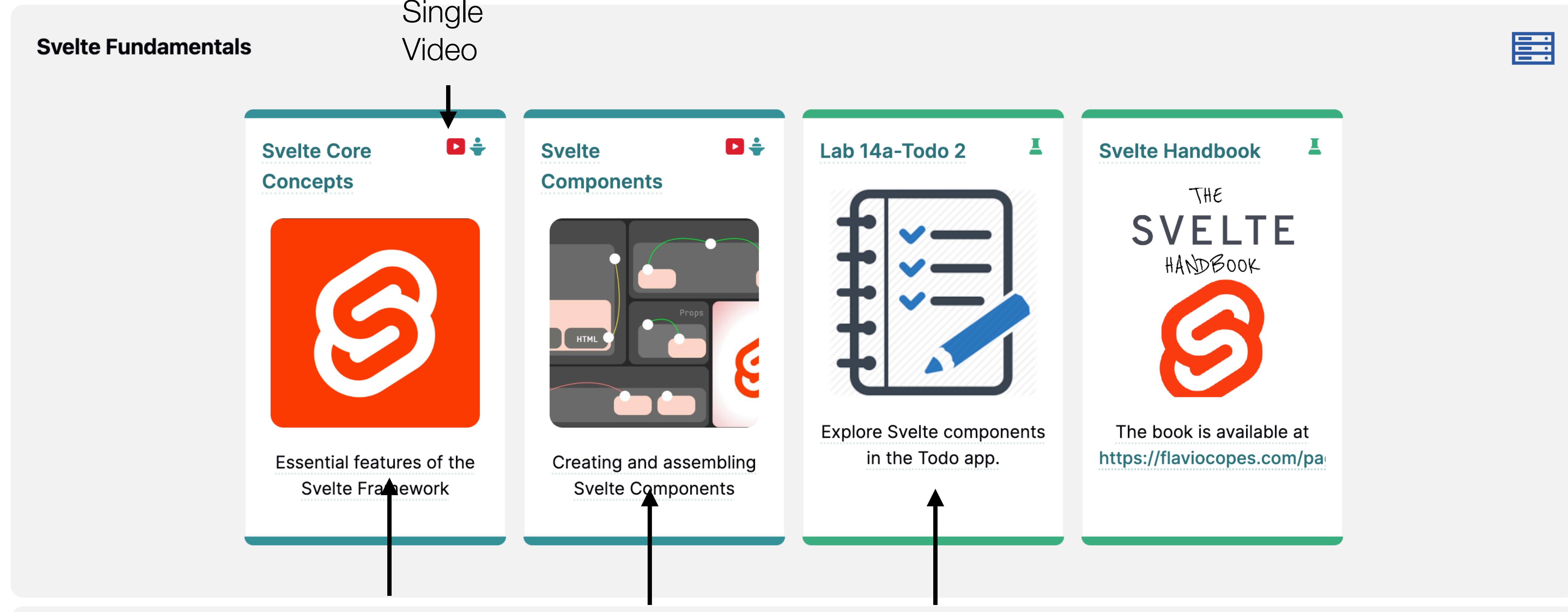
- Routing**: How Routing Works, Hash based routing in Svelte.
- Pages & Components**: Creating and assembling Svelte Components.
- Lab-15a-Donation-Svelte-1**: Front end for Donation - Pages + Components.

## Topic Video

Pre-recorded or live stream

## Sub Topic 1

# Sub Topic



Talk Slide PDFs

Lab

Embedded PDF reader custom with Navigation →

4 of 18

Context

The context API is provided by 2 functions which are provided by the svelte package: getContext and setContext

You set an object in the context, associating it to a key

In another component you can use getContext to retrieve the object assigned to a key:

→ **Context:** A feature to allow a component communicate with multiple descendants

```
<script>
import { setContext } from 'svelte'

const someObject = {}

setContext('someKey', someObject)
</script>
```

```
<script>
import { getContext } from 'svelte'

const someObject = getContext('someKey')
</script>
```

16: Svelte Applications

Talk Context - related talks & videos ←

Svelte Applications

Svelte Applications

Why Svelte? ▶

Donate API ▶

Donate in Svelte ▶

Lab-16a-Donate-Hapi-3 ▶

Lab-16b-Donate-Svelte-3 ▶

# Talk

# Labs

Lab Steps



- Objectives
- Create
- Router**
- Components
- Pages
- Donate & Report
- Login & Signup
- Solutions

## Router

Like web apps, front end applications also have the concept of a router. Svelte itself does not provide one, but there are several popular options:

- <https://www.routify.dev/>
- <https://github.com/AlexxNB/tinro>
- <https://github.com/ItalyPaleAle/svelte-spa-router>

We will use the last one - which is documented in detail in this svelte text book:

- [Svelte 3 Up & Running](#)

Start by installing it:

```
npm install svelte-spa-router
```

To try it out, create three svelte file in a new **pages** folder in the project:

### pages/Main.svelte

```
<div class="notification is-link">
  Hello from Main!
</div>
```

### pages/Login.svelte

```
<div class="notification is-info">
  Hello from Login!
</div>
```

### pages/Signup.svelte

```
<div class="notification is-warning">
  Hello from signup!
</div>
```

This is a revised App.svelte, to include the routing table:

### App.svelte

Markdown  
formatted text

Code block  
syntax  
highlighting



Enter search term:

env

# Search

"env": {  
  
Lab-01-Playtime-0-1-0/ Code Quality Tools

# dotenv environment variables file  
  
Lab-01-Playtime-0-1-0/ Code Quality Tools

# dotenv environment variables file  
  
Lab-01-Playtime-0-1-0/ Code Quality Tools

.env  
  
Lab-01-Playtime-0-1-0/ Code Quality Tools

.env.test  
  
Lab-01-Playtime-0-1-0/ Code Quality Tools

.env.production  
  
Lab-01-Playtime-0-1-0/ Code Quality Tools

In general we should attempt to "externalise" sensitive information, secrets or API details to a text file. This is usually called .env  
  
Lab-02-Playtime-0-2-0/ Exercises

.env  
  
Lab-02-Playtime-0-2-0/ Exercises

<https://github.com/motdotla/dotenv#readme>  
  
Lab-02-Playtime-0-2-0/ Exercises

In general we should attempt to "externalise" sensitive information, secrets or API details to a text file. This is usually called .env  
  
Lab-03-Playtime-0-3-0/ Exercise 3

.env  
  
Lab-03-Playtime-0-3-0/ Exercise 3

<https://github.com/motdotla/dotenv#readme>  
  
Lab-03-Playtime-0-3-0/ Exercise 3

npm install dotenv  
  
Lab-03-Playtime-0-3-0/ Exercise 3

Import the package in server, and check for the presence of the .env file, exiting if not found:  
  
Lab-03-Playtime-0-3-0/ Exercise 3

import dotenv from "dotenv";  
  
Lab-03-Playtime-0-3-0/ Exercise 3

import dotenv from "dotenv";  
  
Lab-03-Playtime-0-3-0/ Exercise 3

const result = dotenv.config();  
  
Lab-03-Playtime-0-3-0/ Exercise 3

If we have included the .env file above at the root of our project, then we can directly access its entries. So, our cookie configuration can now import these values:

# Presence

Learner 'Opt in'  
to Share  
Presence

The screenshot shows a learning management system interface. At the top, there's a header with the course name "Full Stack Web Development" and instructor "Eamonn de Leastar". It also displays "Current Week 9" and a "Calendar" button. Below the header is a navigation bar with icons for Home, Course, and Search.

The main content area features a grid of six cards representing course modules:

- Overview + Assignments**: Shows a grid icon and a brief description: "Overview, schedule & assignments."
- 01: Node**: Shows the Node.js logo and a brief description: "The fundamentals of node. A first node application."
- 02: Hapi.js**: Shows the Hapi.js logo and a brief description: "The Hapi application Framework"
- 03: Joi**: Shows the Joi logo and a brief description: "Validation with Joi"
- 04: TDD**: Shows a circular TDD diagram with steps: "Write a failing test", "Refactor", and "Make the test pass". A brief description: "Test Driven Development"
- 05: Models**: Shows a database icon and a brief description: "Incorporate Mon"

To the right of the main content is a user profile sidebar for "Jordan Harrison". It includes a profile picture, a status message "Logged in as: Jordan Harrison", and a "Share Presence" button with a checked checkbox. A vertical arrow points from the text "Learner 'Opt in' to Share Presence" down to the "Share Presence" button. Other options in the sidebar include "View 0 Online", "Tutors Time", "Github Profile", and "Logout".

# Presence

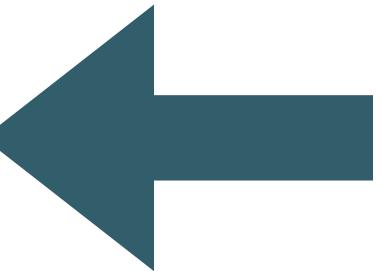
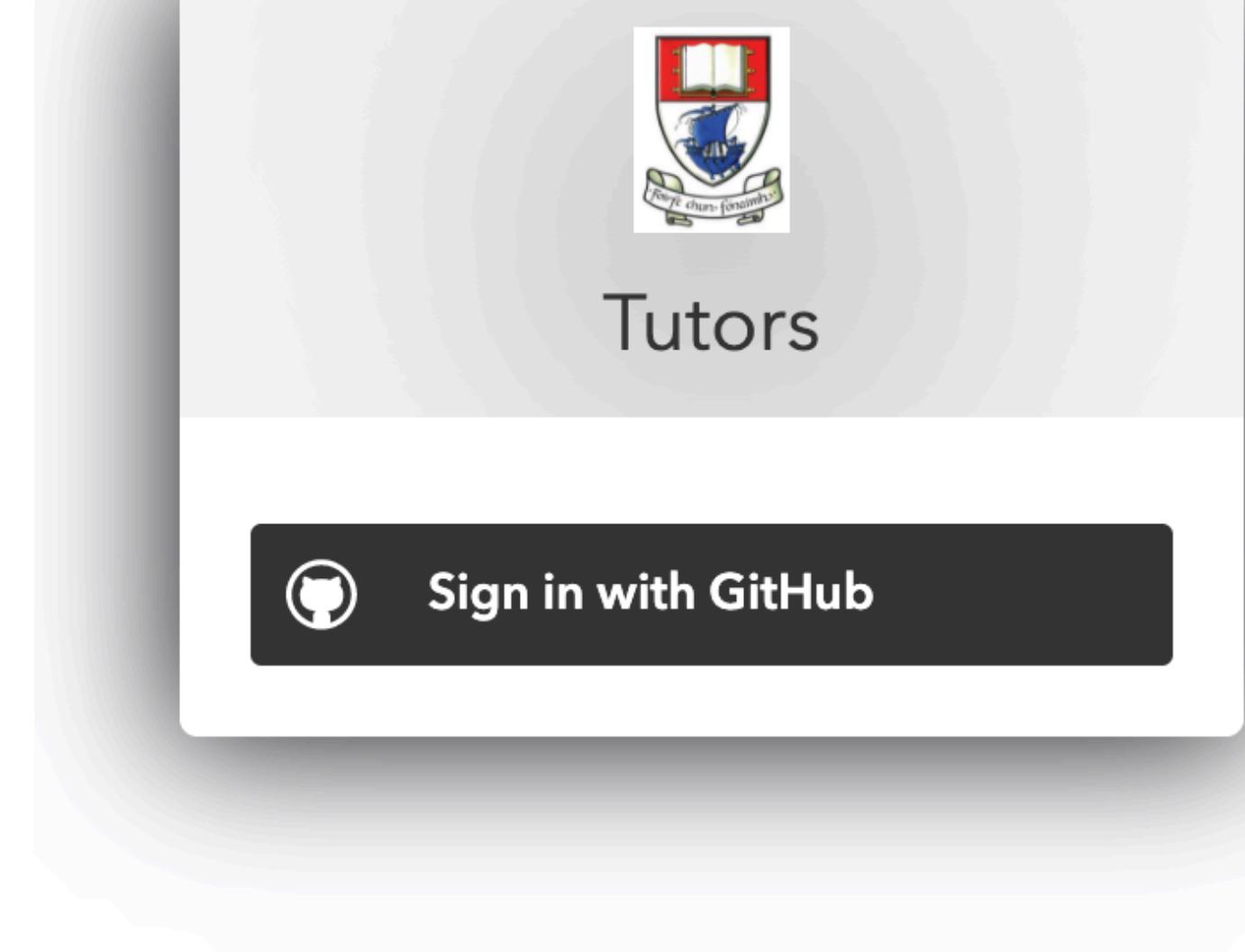
Card for each student 'live' + learning object they are interacting with

The screenshot shows a learning management system interface. At the top, it displays the course title "Full Stack Web Development" by "Eamonn de Leastar", the current week "9", and a "Calendar" button. Below the header, there are five cards representing different lessons:

- Overview + Assignments**: Overview, schedule & assignments.
- 01: Node**: The fundamentals of node. A first node application.
- 02: Hapi.js**: The Hapi application Framework
- 03: Joi**: Validation with Joi
- 04: TDD**: Test Driven Development

To the right of the main interface, three student cards are displayed vertically, each showing a profile picture, the student's name, a network diagram, and a timestamp indicating when they last interacted with a specific lesson:

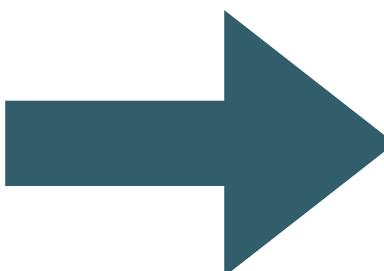
- Jane Doe**: Last interacted at 10:03:28 with lesson 07: Transport Layer & PT IoT. Network diagram shows a Client sending SYN to a Server, which responds with SYN-ACK, and the Client ACKs.
- John Doe**: Last interacted at 10:03:45 with lesson 07: Transport Layer & PT IoT. Network diagram shows a Client sending SYN to a Server, which responds with SYN-ACK, and the Client ACKs.
- Joe Doe**: Last interacted at 10:03:24 with lesson 07: Transport Layer & PT IoT. Network diagram shows a Client sending SYN to a Server, which responds with SYN-ACK, and the Client ACKs.



When accessing module, you will be asked to sign in with Github

A screenshot of a GitHub OAuth authorization dialog. At the top, it shows a user icon with a star and a checkmark, followed by the GitHub logo. The title "Authorize Auth0" is centered below. The main content area shows a user profile picture and the text "Auth0 by auth0 wants to access your edeleastar2 account". It includes a "Personal user data" section with the subtext "Email addresses (read-only)". At the bottom are "Cancel" and "Authorize auth0" buttons, with a note that "Authorizing will redirect to https://login.eu.auth0.com". A footer bar at the bottom contains three items: "Not owned or operated by GitHub", "Created 2 years ago", and "More than 1K GitHub users". A link "Learn more about OAuth" is at the very bottom.

... first time you will have to authorise ...



# Github Profile

Screenshot of a GitHub profile page for user `edeleastar2`.

The top navigation bar includes the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. On the far right are icons for notifications, a plus sign, and a user profile.

A message banner at the top states: "Your profile picture has been updated. It may take a few moments to update across the site." with a close button.

The main content area is titled "Public profile".

**Name:** Eamonn de Leastar

**Profile picture:** A circular photo of a man with glasses and a textured hat, set against a backdrop of hills and water.

**Profile sidebar:** Includes links for Profile, Account, Appearance (marked as New), Account security, Billing & plans, Security log, and Security & analysis.

**Public email:** A dropdown menu showing "Select a verified email to display" with an arrow icon. Below it, a note says: "You have set your email address to private. To toggle email privacy, go to [email settings](#) and uncheck 'Keep my email address private.'"

**Bio:** A blank input field with an "Edit" button containing a pencil icon.



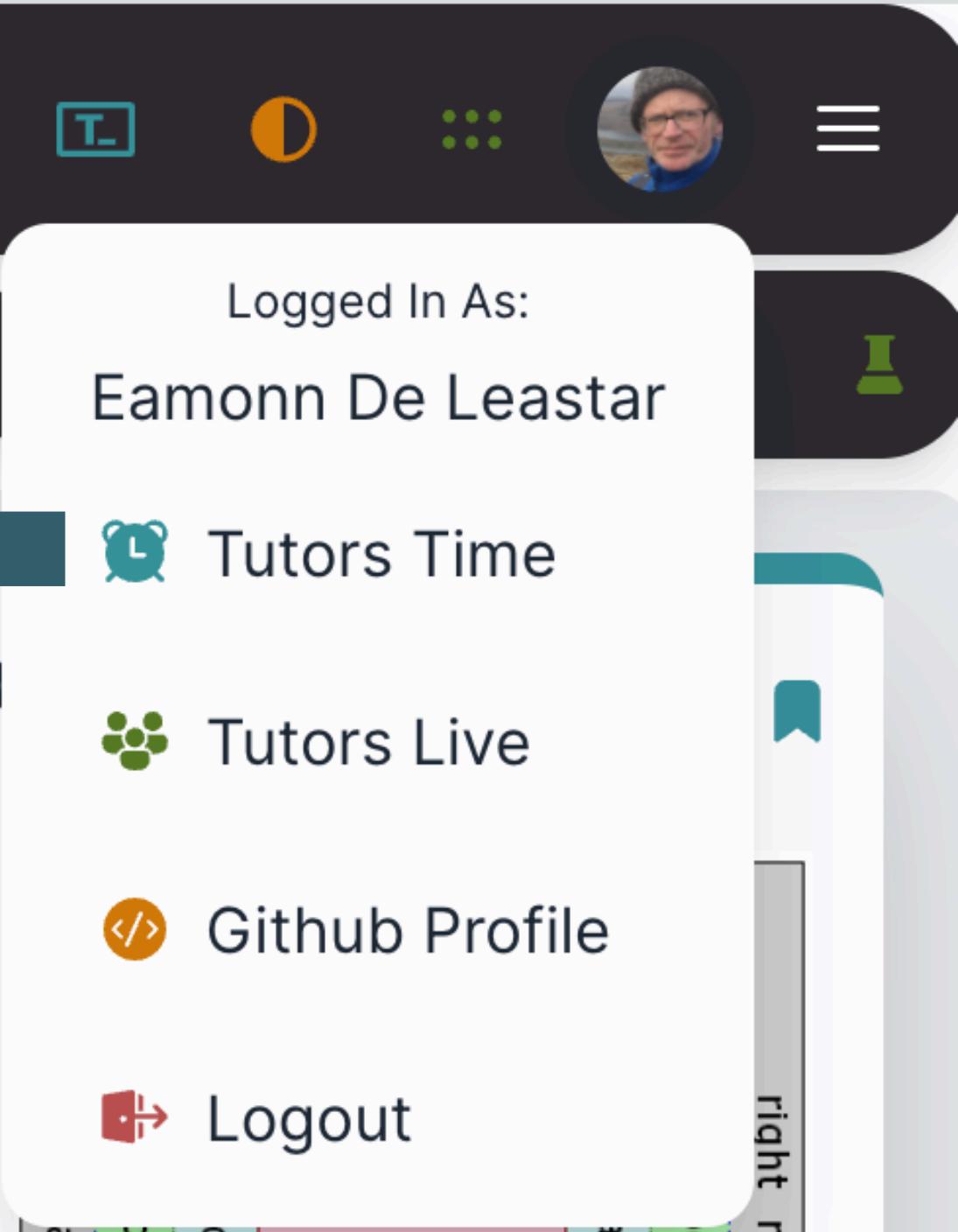
# TutorsTime - Labs

The screenshot shows a digital whiteboard interface. In the top left, there's a user profile for "Eamonn de Leastar" with a photo, followed by "Tutors Time" and "Web Development". A "Appear Offline" button is also present. The top right features a large clock icon. Below the header, a navigation bar has "Labs" (highlighted in blue) and "Calendar". The main area displays a timeline of lab activities with icons and labels:

- Lab-08 Playlist 1
- Lab-07b Play Setup
- Lab-07a Semantic-UI
- Lab-06 Nav Params
- Lab-05b Templates
- Lab-05a Harp & Surge
- Lab-04 Navigation
- Lab-03a Layout
- Lab-02b CSS Paths & Links
- Lab-02a CSS Fonts & Colours
- Lab-01 HTML Structure
- Lab-00 Editing HTML

Below the timeline, a table provides details for each lab, including the user, date last accessed, total time spent, and a progress bar.

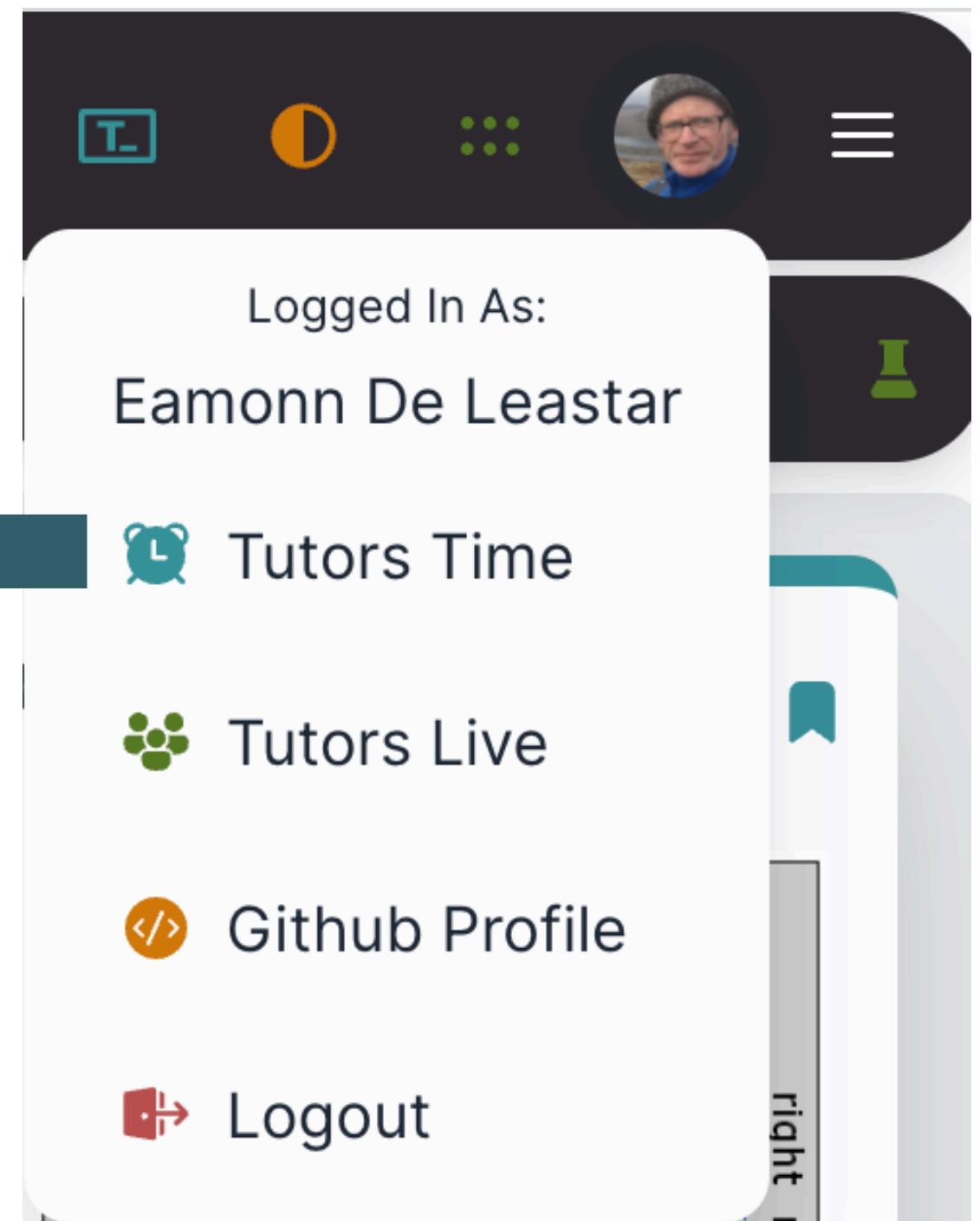
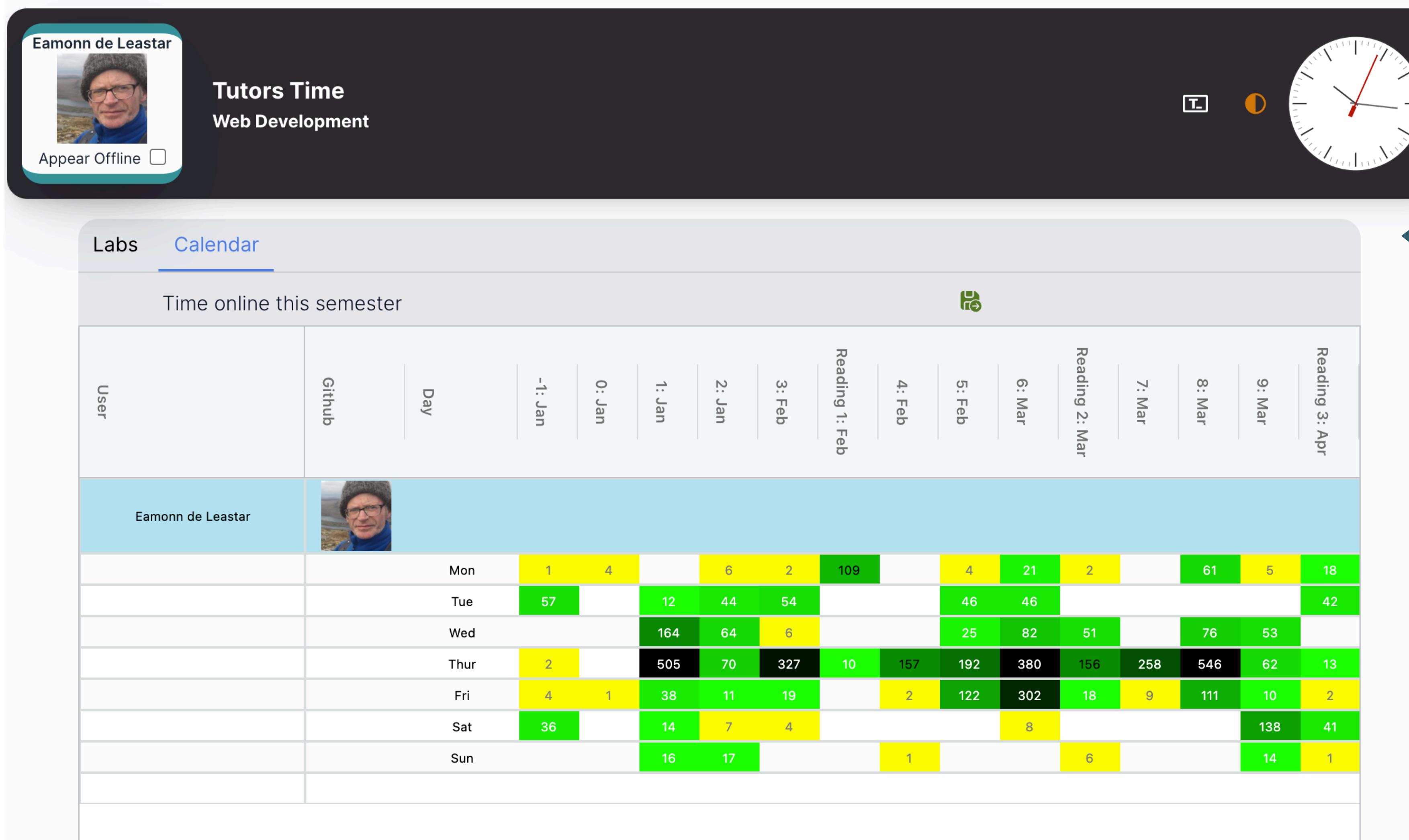
User	Date Last Accessed	Total	Progress Bar
Eamonn de Leastar	2/01/2022, 10:12:3	1337	<div style="width: 50%;">50</div>
Github			<div style="width: 8%;">8</div>
			<div style="width: 11%;">11</div>
			<div style="width: 2%;">2</div>
			<div style="width: 16%;">16</div>
			<div style="width: 28%;">28</div>
			<div style="width: 22%;">22</div>
			<div style="width: 6%;">6</div>
			<div style="width: 41%;">41</div>
			<div style="width: 2%;">2</div>
			<div style="width: 128%;">128</div>
			<div style="width: 47%;">47</div>



Number of minutes the Each lab has been active in your browser (estimate)



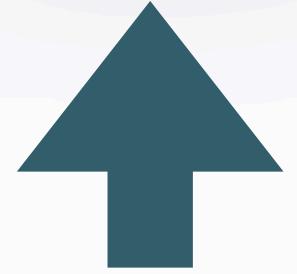
# TutorsTime - Calendar



Number of minutes  
the the Course has  
been active in your  
browser (estimate)

Eamonn de Leastar  
Appear Offline

Tutors Live: Web Development  
Web Development



You may choose to not appear on this page if you prefer to 'appear offline'

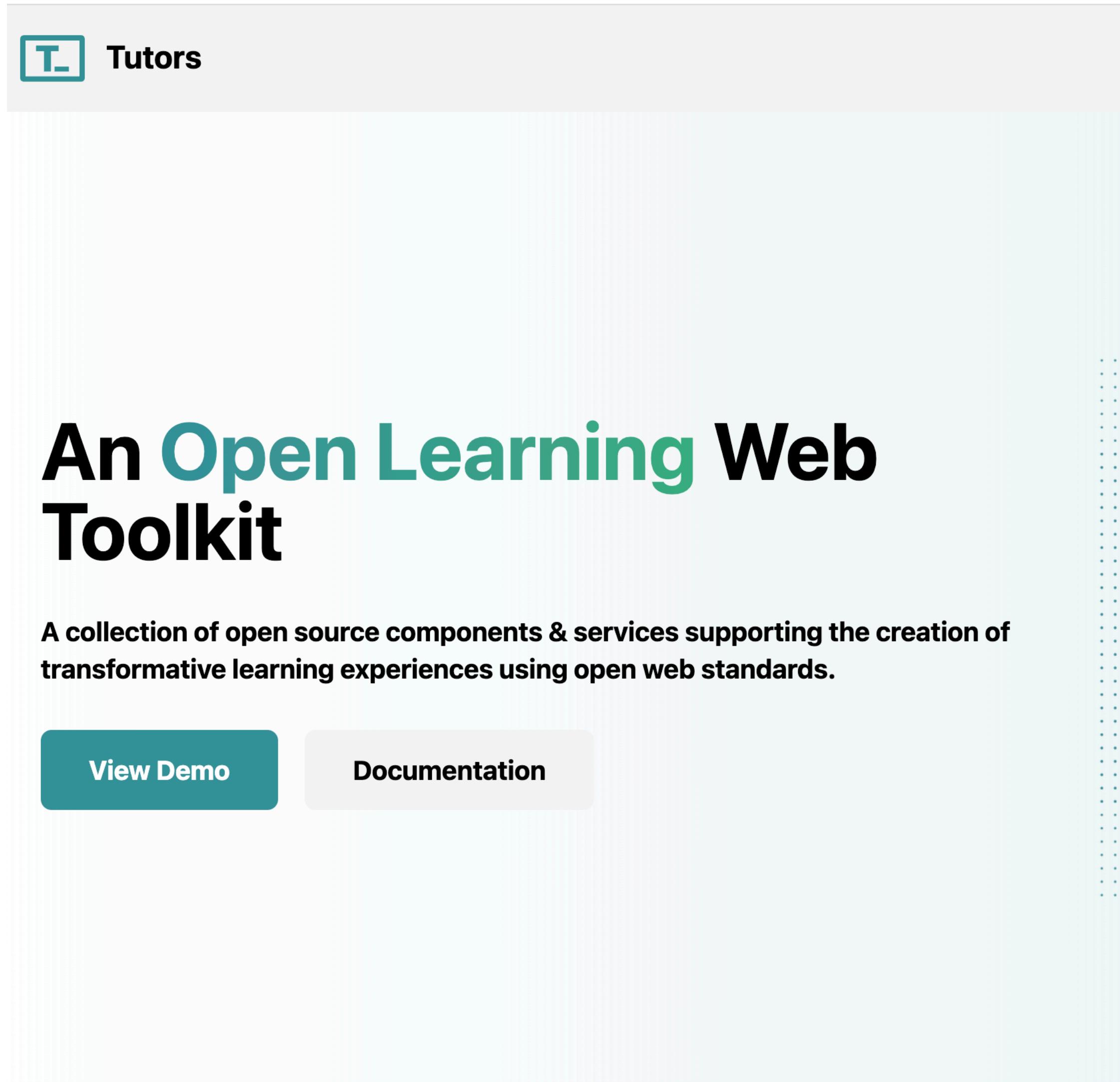
Student	Activity	Last Seen
Joe	Arrays Lab-05a	12:38:55
Mary	Classes Lab-04a	12:47:54
Mike	Arrays Lab-05a	12:39:14
Frank	Arrays Lab-05a	12:47:54
Elizabeth	Arrays	12:38:57
Sean	Arrays	12:47:54

Logged In As:  
Eamonn De Leastar

Tutors Time  
 Tutors Live  
 Github Profile  
 Logout

Any students currently online will appear here, indicating the topic/lab they may be working on

<https://tutors.dev/>



The screenshot shows the homepage of the Tutors website. At the top left is a teal square icon containing a white stylized 'T' and the word 'Tutors' in a black sans-serif font. The main title 'An Open Learning Web Toolkit' is centered in large, bold, black and teal letters. Below it is a subtitle in a smaller black font: 'A collection of open source components & services supporting the creation of transformative learning experiences using open web standards.' At the bottom are two buttons: a teal one labeled 'View Demo' and a white one labeled 'Documentation'.

# An Open Learning Web Toolkit

A collection of open source components & services supporting the creation of transformative learning experiences using open web standards.

[View Demo](#) [Documentation](#)