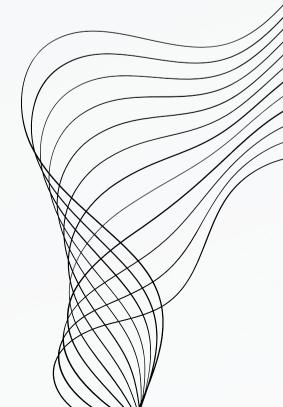
# **AN INTRODUCTION TO** K / S H T M X

**GUSTAVO LINS & SACHA DAVIDSON NBCC WMAD SR A 2024 ADVANCED TOPICS - WEB DEV RESEARCH PROJECT** 





## **AGENDA**

WHO ARE WE?

- WHAT IS HTMX?
- WHY USE HTMX?
- HOW TO INSTALL HTMX
- HTMX ATTRIBUTES
- DEMO: INTRO TO HTML
- EXERCISE: YOUR TURN
  - WHERE TO LEARN MORE?
    - Q&A SESSION



# WHO ARE WE?

### **Gustavo Lins**

Native from Brazil with a background in engineering, now an IT Student at NBCC Moncton!

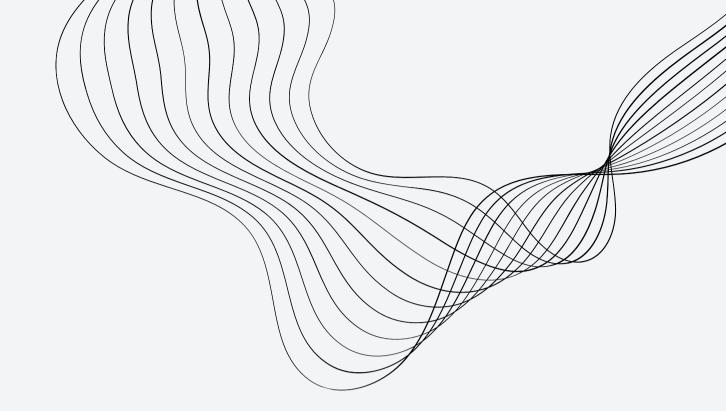
I will be leading the POST portion of our demo.



Sacha Davidson

Some call me Sacha. Some call me 'efficiency Pete'. Call me whatever, just don't call me late for dinner!

I will be leading the GET portion of our demo.



# WHAT IS HTMX?

## HTMX is a Library

HTMX is a compact library that provides developers with direct access to AJAX, CSS Transitions, WebSockets, and Server-Sent Events within HTML.

### Hypermedia As The Engine Of Application State

HTMX emphasizes one of the four principles of the Uniform Interface constraint of **REST Architecture**: Hypermedia as the Engine of Application State (HATEOAS)

HTMX expects hypermedia (like HTML) as a response from the server.

In HTML, the two major hypermedia controls are anchor tags and form tags. These elements allow us to issue GET or POST requests from our HTML.

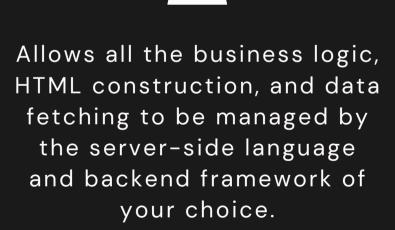
With HTMX, we are now able to issue AJAX requests from almost any HTML element.

We achieve this using special attributes, like hx-get and hx-post. It also allows us to replace specific elements instead of reloading our entire page



## Generalizing Hypermedia

## WHY USE HTMX?



### FLEXIBILITY



HTMX simplifies development with declarative attributes in HTML, reducing boilerplate code and eliminating the need for complex JavaScript frameworks

SIMPLICITY



By dynamizing only specific parts of a webpage, HTMX provides a smoother user experience, faster load times, and enhanced overall performance.

### **USER EXPERIENCE**

## HOW TO INSTALL HTMX?

### Easiest Way to Add HTMX to a Project

• Using a CDN! You can simply add this to your head tag and get going

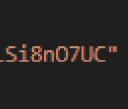
### script

src="https://unpkg.com/htmx.org@1.9.10" integrity="sha384-D1Kt99CQMDuVetoL11rYwg5t+9QdHe7NLX/SoJYkXDFfX37iInKRy5xLSi8n07UC" crossorigin="anonymous" ></script>

Note: While using a CDN is the easiest way to get started, they are not recommended for production

### More Ways of Adding HTMX to a Project

- For npm-style build systems, you can install HTMX via npm
- You can also download HTMX and copy it into your project ~47kb
- Another method for installing is to use Webpack



# HTMX ATTRIBUTES

### HTMX Attributes

An HTMX tag, or attribute, is an HTML attribute that enables dynamic updates and interactions in web pages through AJAX requests.

### **Common HTMX Attributes**

| hx-get:  | hx-post:             |
|--|----------------------|
| Specifies the URL for GET requests.            | Specifies the URL fo |
| hx-target:                                     | hx-swap:             |
| Defines the HTML element where the response    | Determines how the   |
| will be inserted.                              | target element.      |
| hx-trigger:                                    | hx-indicator:        |
| Specifies the event that triggers the request. | Specifies the HTML   |
|  | processing status d  |



or POST requests.

e response replaces the content of the

element used to indicate loading or during a request

# DEMO: INTRO TO HTMX

### **Overview**

- The demo will be two parts:
  - $\circ$  GET and POST
- We will cover some fundamental HTMX attributes and their uses/use cases
- We have chosen to use Node.js + Express for this project
- It is worth noting that you are not required to use Node.js.
- Any server or REST API that responds with HTML markup should work.
- HTMX is often used with GO

## **Part One: GET**

- Part One focuses on making GET requests to our server using HTMX
- HTMX Attributes Used:
  - hx-get
  - hx-trigger
  - hx-swap
  - hx-target
  - hx-confirm
  - $\circ$  hx-vals
- This portion of the demo will introduce students to common mistakes and use cases for working with HTMX.
- We'll work with attributes, using some of the server-side code as a reference.

### Part Two: POST

- Part Two focuses on working with **POST requests using HTMX**
- HTMX Attributes Used:
  - hx-get
  - hx-post
  - hx-trigger
  - hx-target
  - hx-indicator
- This portion will teach the users how to do post operations in HTMX
- We will work with several attributes, representing different cases of posting to a back-end.



## EXERCSE: YOUR TURN

### Mission

Using what you've learned, implement the following:

- An input that issues a POST request to the server to add a name to the list
- An unordered list that displays all the names stored in the server via a GET re

### Steps:

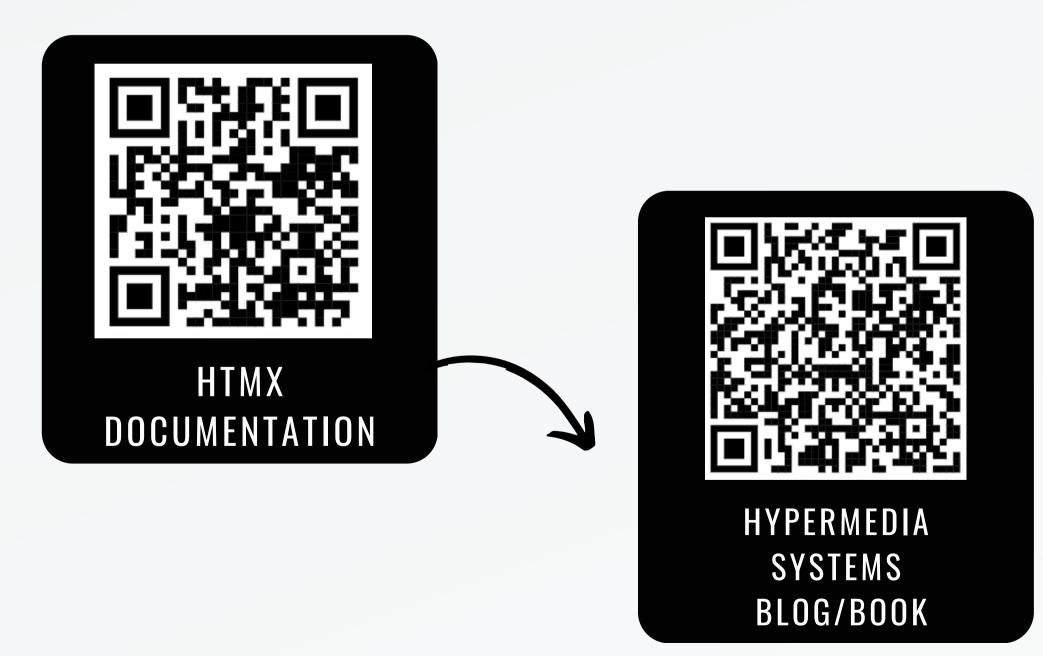
- 1. Add the HTMX CDN to the exercise.html page
- 2. Add the required HTMX attributes to get the nav when the form loads
- 3. Add the required HTMX attributes to post names when the form submits
  - a. Use an hx-indicator for while the request processes
  - b. Target the div with an id of result
- 4. Add the required HTMX attributes to get the names
  - a. When the page loads
  - b. When the page loads & every two seconds (Challenge 1)
  - c. When the page loads & every two seconds & when the form submits (Ch

### Tips

|             | <ul> <li>All of the server-side code</li> </ul> |
|-------------|---|
| equest      | has been written for you.                       |
|             | <ul> <li>To implement the POST</li> </ul>       |
|             | functionality, you will need to                 |
|             | issue a request to "post-                       |
|             | name"   |
|             | <ul> <li>To implement the GET</li> </ul>        |
|             | functionality, you will need to                 |
|             | issue a request to "get-                        |
|             | names"  |
|             | <ul> <li>To complete Challenge 1 and</li> </ul> |
|             | Challenge 2, you will need to                   |
|             | go to htmx.org and research                     |
| nallenge 2) | <ul><li>"polling" and</li></ul>                 |
|             | <ul> <li>"standard event modifiers"</li> </ul>  |

# WHERE TO LEARN MORE?

Here are some links that were really helpful throughout our journey:





Another great source has been ThePrimeagen on YouTube

# Q&A SESSION