

Fast and simple Web apps with Alpine.js (and Delphi)

Marco Breveglieri

 Software Developer |  Trainer and Consultant |  Tech Content Creator

ROME oct **26/27**

Marco Breveglieri

-  Software Developer
 -  Trainer and Consultant
 -  Tech Content Creator
 -  Community lover
-



 <https://linktr.ee/marco.breveglieri>

Foreword



Web Application $\neq$ SPA

- You are **not required to build a SPA** (*Single Page Application*) everytime
- Angular, React and Vue are **too complex** for some kind of basic needs
- Some «full optional» libraries/frameworks are quite large and difficult to grasp
- Focusing on Delphi developers
 - Some Node-based framework require separated projects
 - JavaScript (and its recent incarnations) are really different from Delphi
 - However, it is recommended to know a little about JS...

Alpine.js to the rescue!

Alpine.js is the «new kid on the block» for JavaScript frameworks.

With its lightweight and easy-to-learn syntax, it's perfect for beginners and experts alike.

Get ready to take your web application to the next level with *Alpine.js*!

Alpine.js



"I hope you find Alpine to be a breath of fresh air. Silence among noise"

Caleb Porzio, creator of *Alpine.js*

Mantras

Simple. 

Lightweight. 

 Powerful
as hell!

What is it?

- **Alpine.js** is a **declarative** and **reactive** JavaScript framework that allows you to add interactivity to your HTML with minimal JavaScript code.
- It is similar to *Vue.js* or *React*, but much **simpler and easier** to learn (it is not meant to replace them!).



Tell me more

- Small and robust **JavaScript library** that expands HTML capabilities and provides a simple API
- Think about JQuery, but for modern web
- Type in your script and... «done»! ⚡

 <https://alpinejs.dev/>

Someone said «jQuery»?



Well, not the same...

Imperative Style (Classic DOM, JQuery, ...)

- Code flow is typically a linear sequence of steps
- You directly specify each step of the process
- Can lead to complex and unmaintainable code as the application grows its size and complexity

```
// Get a reference to an HTML element
var container = document.getElementById("container");

// Create a new element
var newElement = document.createElement("div");

// Set attributes and content
newElement.id = "newDiv";
newElement.textContent = "This is a new div";

// Append the new element to an existing element
container.appendChild(newElement);

// Add event listeners
newElement.addEventListener("click", function() {
    alert("Element clicked!");
});

// Style the element
newElement.style.backgroundColor = "lightblue";
newElement.style.padding = "10px";
```

Declarative Style (Alpine.JS, React, Angular, Vue, ...)

- Code/markup mainly describes what the UI should look like, with data and user interactions
- Allows you to focus on what the result should be
- Makes the code maintainable and easier to understand

```
function App() {  
  // ...  
  return (  
    <div>  
      <h1>Declarative Style with React</h1>  
      <input  
        type="text"  
        value={newText}  
        onChange={e => setNewText(e.target.value)}  
        placeholder="Add a new div"  
      />  
      <button onClick={handleAddClick}>Add</button>  
      <ul>  
        {items.map(item => (  
          <li  
            key={item.id}  
            style={{ backgroundColor: item.backgroundColor, padding: item.padding }}  
            >  
              {item.text}  
            </li>  
          )  
        )  
      }  
    </ul>  
    </div>  
  );  
}
```

Why Alpine.js?

- **Alpine.js** is perfect for small to medium-sized projects that require some interactivity but doesn't need a full fledged framework
- It's easy to learn, has a small footprint and can be added to an existing project with minimal effort.

One library, many benefits

- **Minimal:** it does contain just a few elements
 - 15 attributes
 - 6 properties
 - 2 methods
- **Usable:** download it or include directly in your page
 - Available from CDNs:
🔗 <https://cdn.jsdelivr.net/npm/alpinejs@3.x.x/>
 - No need to use *Node.JS*, *npm* and their friends (if you don't need them)
- **Modern:** support all browser recent features
 - Sync with browser tick to update graphics
 - CSS Transitions
- **Easy:** low learning curve
 - Full documentation available
 - JavaScript newbies and non-lovers can use it
 - Mainly uses declarative programming style

What can you do?

- **Implement some recurrent web page functionality**

- *Form validation*
- *Toggling classes*
- *Showing/hiding elements*
- *Filtering lists*
- *Boost static pages with dynamic interaction*

- **It is suitable for many simple scenarios**

- *Implementing HTML(5) browser games*
- *Collecting data and surveys*
- *Manage an e-commerce shopping cart*
- *Creating engaging landing pages*
- *Prototype web apps and ideas*

Basic concepts



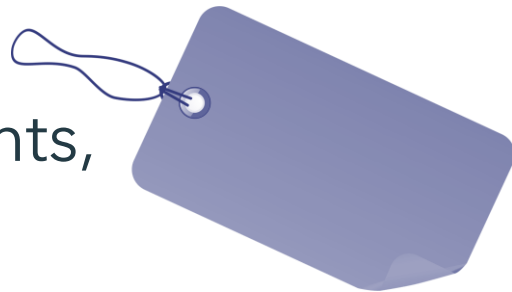
State

- JavaScript data that Alpine watches for changes
- It is at the core of everything you do in Alpine
- It can be
 - **Local** using `x-data`
 - **Global** using `Alpine.store()`
- Alpine.js uses the `x-bind` directive to update HTML in real time based on changes detected on data



Directives

- Alpine.js uses **directives** to add interactivity to your HTML
- Directives are **special attributes** that start with «X-» and allow you to perform actions like showing and hiding elements, toggling classes and more...



Event Handling

- Alpine.js makes it easy to **handle events** in your HTML using the **x-on** directive.
- This allows you to **perform actions** like submitting forms, making API calls and more.



Getting Started



Using the right tools

- **Visual Studio Code**

- Install extensions!
 - [Alpine.js Intellisense](#)
 - [Alpine.js Syntax Highlight](#)
 - [Auto Close Tag](#)
 - [Live Server](#)
 - [Prettier – Code Formatter](#)
- Create a dedicated profile

- **JetBrains WebStorm**

- Alpine.js plugin
<https://plugins.jetbrains.com/plugin/15251-alpine-js-support>



Hello Alpine!

```
<html>
<head>
  <script defer src="https://cdn.jsdelivr.net/npm/alpinejs@3.x.x/dist/cdn.min.js">
</script>
</head>
<body>
  <h1 x-data="{ message: 'I ❤️ Alpine' }" x-text="message"></h1>
</body>
</html>
```

Import the library script

Define your data model

Bind the model to UI elements

Build apps with Alpine.js

1. Setting up a Project

- Use a CDN link or npm to install Alpine.js into your project

2. Defining Data

- Use ``x-data`` to define your component's data properties

3. Binding Data

- ``x-bind`` allows you to bind data to HTML attributes or elements

4. Event Handling

- ``x-on`` helps you manage user interactions and events

5. Conditionals and Loops

- Use ``x-if`` and ``x-for`` to conditionally render or loop through elements

6. Integrating with APIs

- Use JavaScript to fetch data from APIs and update your UI

It's demo time! 🕒



Are you guys ready?

JS

Short Recap

Directives

- x-data
- x-on
- x-text
- x-show
- x-model
- x-for

Events

- @event (x-on:event)
- @event.*modifier*

Functions

- \$store
- \$watch
- \$dispatch

What about Delphi?

- *Delphi*  is **good for the backend** part of your app
- Use the Web development framework of your choice!
 - *DelphiMVCFramework*
 - *RAD Server (from Embarcadero)*
 - *MARS Curiosity*
 - *WiRL*
 - *...and many others!*
- Just remember to enable «static file» support

It's demo time! 🕒



Conclusions



Alpine.js in short

- Alpine.js is a **fun and easy-to-learn JavaScript framework** that can help you add interactivity to your HTML with **minimal effort**
- Whether you are a beginner or an expert, Alpine.js is **definitely worth checking out!**
- **Use it instead of HTMX** if you don't like its syntax and want to keep coding (you're a developer, right?)

More goodies! 🎁

- **Alpine UI Components!**

- Most websites need UI components like modals, dropdowns, tabs, etc. Building them properly yourself is anything but trivial.
- Instead, we've done the hard work and sweated the details, so you can just copy and paste them into your app and spend less time re-inventing the wheel.

- **Screencasts**

- **Third-party Integrations**

- **Tech Support**



Q & A ?

Thanks!



ROME oct **26/27**