

Intermediate JavaScript Programming

LESSON 9: Responsive Layouts and Interactive Interfaces with Bootstrap and Vue

Learning Objectives:

By the end of this lesson, participants will be able to:

- Understand the role of Bootstrap as a UI layout and styling library and how it supports responsive layouts.
- Understand the concept of reactivity and how Vue.js applies it.
- Combine Bootstrap and Vue to create responsive and interactive web applications.

Lesson Outline:

I. What Is a UI Library vs. a Reactive Framework? (10 min)

- A **UI library** like **Bootstrap** provides styled HTML components (buttons, forms, layouts) and grid systems for consistent design, and adjusts layouts for responsive designs that adjust for various display sizes.
- A **reactive framework** like **Vue.js** tracks application state and updates the DOM automatically when data changes.

Summary:

Purpose	Bootstrap	Vue.js
Focus	Style and layout	Application logic and reactivity
HTML Enhancement	Adds classes to control layout	Adds data bindings and logic
Interactivity	None	Yes
Requires JavaScript	Not strictly	Yes

II. What Is Reactivity? (5 min)

Reactivity means that when the data in your application changes, the view updates automatically — without requiring you to manually manipulate the DOM.

Instead of this (manual approach):

```
let count = 0;
function updateCount() {
  document.getElementById("counter").textContent = count;
}
```

Vue lets you do this:

```
<p>Count: {{ count }}</p>
```

And just update the data:

```
this.count++;
```

The UI updates automatically.

III. Using Bootstrap for Layout and Form Styling (10 min)

Setup Instructions: Include this in your HTML `<head>`:

```
<link  
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.  
css" rel="stylesheet">
```

Example HTML Form:

```
<div class="container mt-4">  
  <h2>Contact Form</h2>  
  <input type="text" class="form-control mb-2" placeholder="Name">  
  <input type="email" class="form-control mb-2" placeholder="Email">  
  <button class="btn btn-primary">Submit</button>  
</div>
```

Commentary:

Bootstrap makes forms and layouts responsive and professional-looking without any custom CSS. It does not handle interactivity.

IV. Using Vue.js for Reactivity (15 min)

Setup Instructions: Add Vue 2 from CDN just before the closing `</body>`:

```
<script src="https://cdn.jsdelivr.net/npm/vue@2"></script>
```

Example: Counter with Vue

```
<div id="app" class="container mt-4">
  <h2>Counter Example</h2>
  <p>Count: {{ count }}</p>
  <button class="btn btn-primary" @click="count++">Increment</button>
</div>

<script>
  new Vue({
    el: '#app',
    data: {
      count: 0
    }
  });
</script>
```

Commentary:

Vue binds UI elements directly to data. This allows you to focus on how the data should behave — not how to update the page.

V. Combined Example: Tip Calculator (Vue + Bootstrap) (15 min)

```
<!DOCTYPE html>
<html>
<head>
  <title>Vue + Bootstrap Tip Calculator</title>
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.
css" rel="stylesheet">
  <script src="https://cdn.jsdelivr.net/npm/vue@2"></script>
</head>
<body>
  <div id="app" class="container mt-4">
    <h2>Tip Calculator</h2>
    <input type="number" class="form-control mb-2" v-model="bill"
placeholder="Bill amount">
    <input type="number" class="form-control mb-2" v-model="tip"
placeholder="Tip %">
    <button class="btn btn-primary" @click="calculate">Calculate
Tip</button>
    <div class="mt-3 alert alert-info">{{ output }}</div>
  </div>

  <script>
    new Vue({
      el: '#app',
      data: {
        bill: '',
        tip: '',
```

```
        output: ''
      },
      methods: {
        calculate() {
          const amount = parseFloat(this.bill);
          const percent = parseFloat(this.tip);
          if (isNaN(amount) || isNaN(percent)) {
            this.output = "Please enter valid numbers.";
            return;
          }
          const tipAmount = amount * (percent / 100);
          const total = amount + tipAmount;
          this.output = `Tip: ${tipAmount.toFixed(2)}, Total:
          ${total.toFixed(2)} `;
        }
      }
    });
  </script>
</body>
</html>
```

VI. Recap & Q&A (5 min)

- Bootstrap is a layout and UI styling toolkit.
- Vue is a reactive framework that automatically updates the view when data changes.
- Together, they make interactive apps that look good and behave intuitively.

Final Multiple-Choice Question:

Which of the following best describes Vue.js? A. A styling library for responsive layouts B. A DOM manipulation tool like jQuery C. A reactive JavaScript framework that updates the UI based on data changes D. A CSS-only framework for building dashboards

(Answer: C. A reactive JavaScript framework that updates the UI based on data changes)