Sprint 4 Report

Front-End – Responsive Web Page Creation

Sprint 3 & 4 : Création d'une page web responsive

Réalisé par :
Karrouach Ansar

Encadré par :
Hamza Bahlaouane
Abdelmajid Bendrif

12 février 2026

VOID

Table des matières

1. Project Setup – Vite and Tailwind CSS

Objective : Initialize a modern front-end project using Vite and configure Tailwind CSS for responsive styling.

1. Create Vite project :

```
npm create vite@latest my-project
cd my-project
```

2. Install Tailwind :

```
npm install tailwindcss @tailwindcss/vite
```

3. Configure plugin in vite.config.js :

```
import { defineConfig } from 'vite'
import tailwindcss from '@tailwindcss/vite'

export default defineConfig({
  plugins: [tailwindcss()],
})
```

4. Import Tailwind in CSS :

```
@import "tailwindcss";
```

5. Start development server :

```
npm run dev
```

Result : Tailwind successfully integrated with Vite. Project ready for responsive development.

2. Responsive Header and Navigation

Objective : Create a fully responsive header with navigation, CTA button, dark mode toggle, and mobile burger menu.

Features implemented :

- Semantic HTML structure
- Responsive navigation (desktop + mobile)
- SVG icons integration
- Sticky header

- Mobile dropdown menu

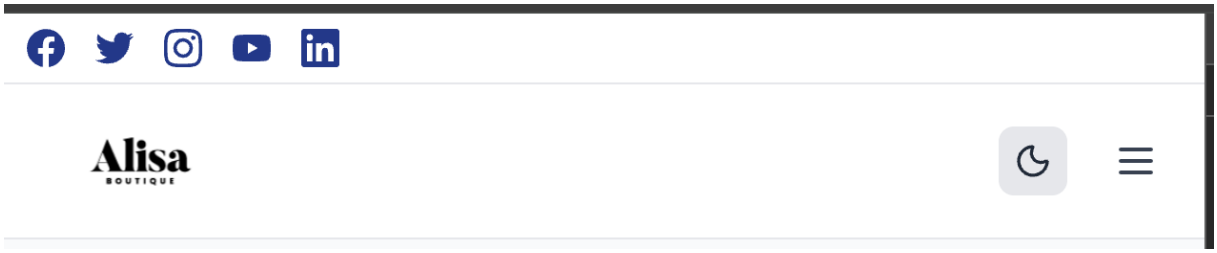

FIGURE 1 – Responsive Header – Mobile View

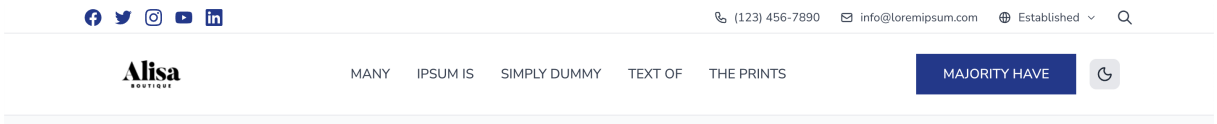

FIGURE 2 – Responsive Header – Desktop View

Result : Header adapts correctly to different screen sizes using Tailwind breakpoints.

3. Dark Mode Implementation

Objective : Implement a dark/light theme toggle using native JavaScript and Tailwind custom variant.

- Custom dark variant in CSS
- Theme persistence using localStorage
- Flash prevention on page load
- Dynamic icon switching

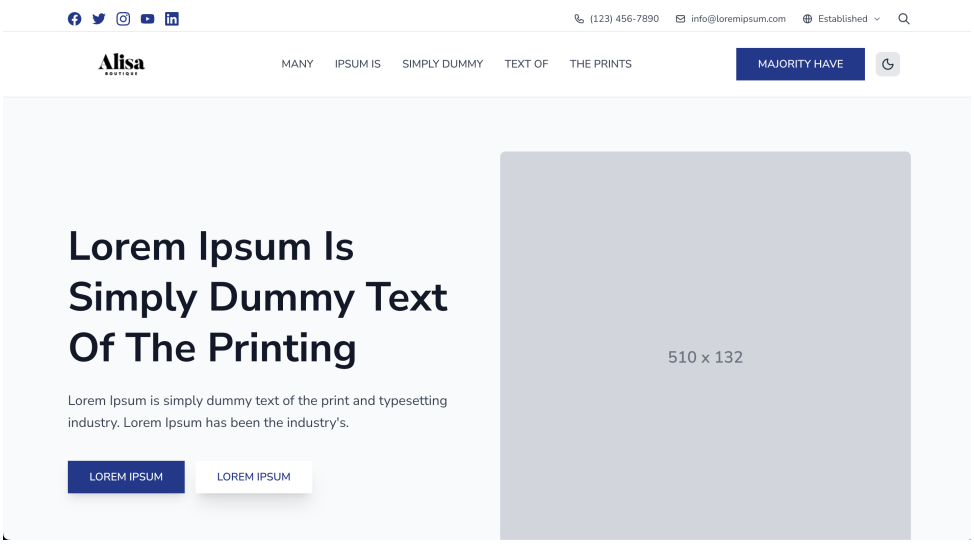

FIGURE 3 – Light Mode Interface

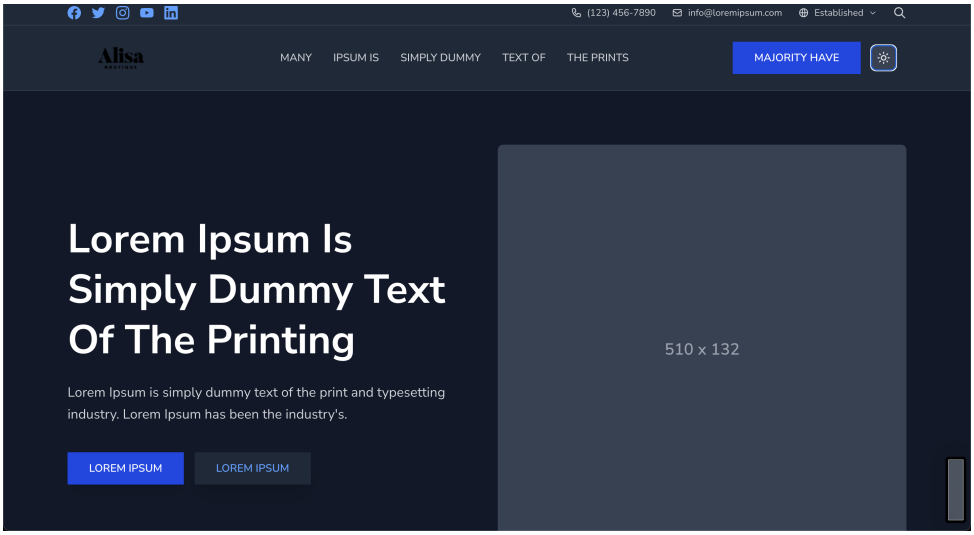

FIGURE 4 – Dark Mode Interface

Result : Smooth dark mode toggle with persistent user preference.

4. Typography and Google Fonts

Objective : Integrate custom Google Font for improved UI design.

```
<link href="https://fonts.googleapis.com/css2?family=Poppins
:wght@300;400;500;600;700&display=swap" rel="stylesheet">
```

```
font-family: 'Poppins', system-ui, Avenir, Helvetica, Arial, sans-serif;
```

Result : Modern typography integrated across the website.

5. Responsive Form with Validation

Objective : Create a responsive form with client-side validation using native JavaScript.

Features :

- Responsive grid layout
- Email validation (regex)
- Moroccan phone number validation
- Error handling with dynamic messages
- Success message and form reset

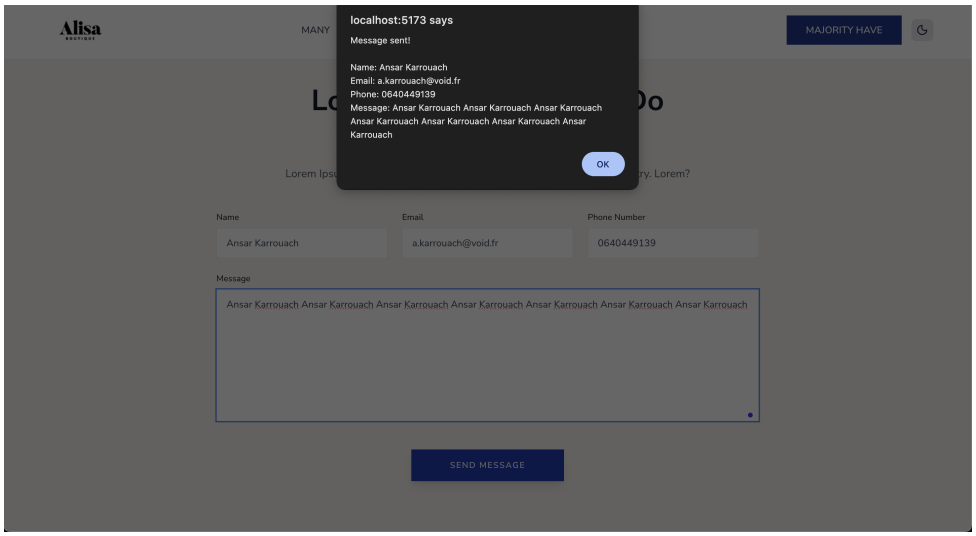

FIGURE 5 – Form – Valid Submission

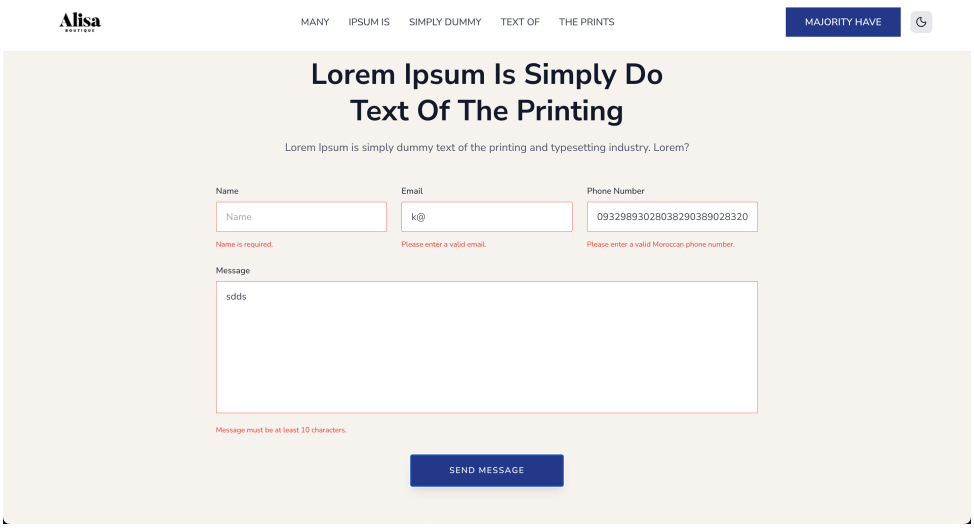

FIGURE 6 – Form – Validation Errors Displayed

Result : Fully functional responsive form with real-time validation feedback.

6. FAQ Interactive Accordion

Objective : Implement interactive FAQ accordion using native JavaScript.

- Toggle open/close behavior
- Auto-close other items
- Dynamic icon switching (+ / -)
- Responsive design

FIGURE 7 – Interactive FAQ Section

Result : FAQ section works dynamically without any framework.

7. Statistics Section – Animated Counters

Objective : Create animated statistics using Intersection Observer and native JavaScript.

Features :

- Count-up animation
- Intersection Observer trigger
- One-time animation execution
- Responsive layout (grid system)

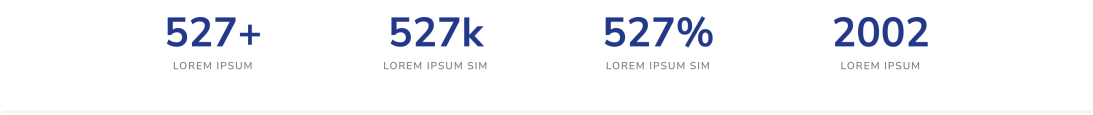

FIGURE 8 – Statistics Count-Up Animation

Result : Animated counters triggered when section enters viewport.

8. SEO Improvements

Objective : Improve search engine visibility and semantic structure.

The following SEO optimizations were implemented :

- Added meta description :

```
<meta name="description"
content="Alisa Boutique offers premium fashion accessories and boutique items.
    Discover our curated collection of stylish products for the modern shopper.
    ">
```

- Improved heading hierarchy (corrected h4 to h3 where necessary)
- Removed duplicate `h1` and replaced second occurrence with `h2`
- Structured semantic layout using :
  - `header`
  - `nav`
  - `main`
  - `section`
  - `footer`

Result : Improved semantic clarity and better SEO Lighthouse score.

9. Accessibility Improvements

Objective : Enhance usability and accessibility compliance (WCAG best practices).

Improvements implemented :

- Added `aria-label` to social media links
- Added `type="button"` to interactive buttons
- Added `aria-label="Toggle mobile menu"` to burger button
- Converted carousel dots to accessible buttons with :
  - `aria-pressed`
  - visually hidden slide labels
- FAQ section improvements :
  - Added unique IDs (faq-toggle-1 to faq-toggle-6)
  - Added `aria-expanded` to buttons
  - Added `role="region"` to panels
  - Linked panels using `aria-labelledby`

Result : Accessibility score improved to 100 in Lighthouse.

10. Performance Optimization

Objective : Optimize loading speed and resource efficiency.

The following performance improvements were applied :

- Converted images from PNG to AVIF format
- Implemented lazy loading :

```
<img src="image.avif" loading="lazy" alt="Product image">
```

- Reduced large images dimensions (1200px to 600px)
- Preloaded Google Fonts :

```
<link rel="preload"
as="style"
href="https://fonts.googleapis.com/css2?family=Nunito+Sans&display=swap">
```

- Optimized font loading strategy
- Reduced unnecessary CSS and unused classes

Result : Significant reduction in Largest Contentful Paint (LCP) and faster page rendering.

11. Lighthouse Performance Results

Objective : Validate performance, accessibility, best practices and SEO using Lighthouse.

After applying optimizations, the following scores were achieved :

- Performance : 99
- Accessibility : 100
- Best Practices : 100
- SEO : 100

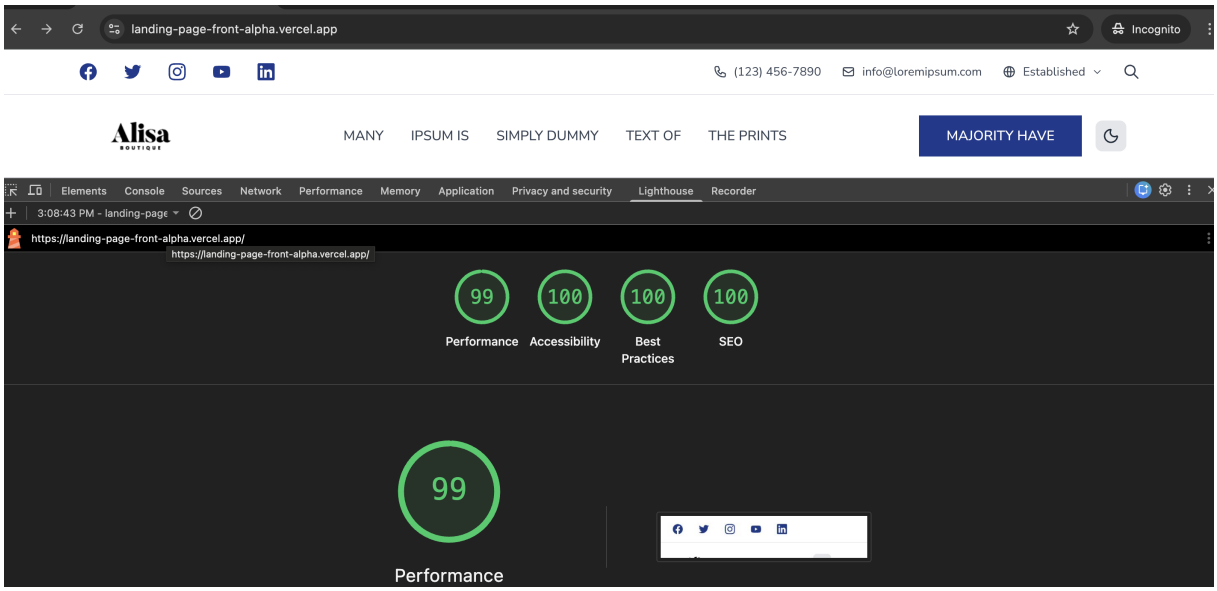

FIGURE 9 – Lighthouse Audit Results

Result : The project meets modern web standards with near-perfect Lighthouse scores.

12. Conclusion

Sprint 4 Progress Summary

During this sprint, a fully responsive and modern web page was successfully developed using semantic HTML, Tailwind CSS, and native JavaScript, in accordance with the project requirements.

The implementation includes :

- Semantic and accessible HTML structure
- Fully responsive design using Tailwind breakpoints
- Interactive slider (carousel) developed with native JavaScript
- Dark mode with persistent user preference
- Responsive and validated form
- Interactive FAQ accordion
- Animated statistics section using Intersection Observer
- SEO improvements and optimized heading hierarchy
- Accessibility enhancements using ARIA attributes
- Performance optimizations (AVIF images, lazy loading, font preloading)

- Lighthouse validation with near-perfect scores

The project demonstrates strong adherence to modern front-end development standards, focusing on performance, accessibility, maintainability, and user experience.

This sprint successfully achieved all required objectives and provides a solid foundation for future enhancements and advanced front-end implementations.