Project Documentation

Documentation for Personal Portfolio Project

Overview

This document provides an in-depth explanation of the TH.STE's portfolio. TH.STE is a music producer, sound engineer and DJ who is looking to attract clients. This website is fully responsive. The individual's production skills are proved through the content of the website. The site is easily navigated through and features interactive elements. The website's uncomplicated navigation, clean layout and pleasing aesthetics, ensures that the website is appealing to users.

The Website can be accessed through GitHub Pages. The Link to the GitHub Page and the website's address are provided here:
GitHub Pages URL: Personal Portfolio Website

GitHub Project URL: Stephenlyne62/Course-Final-Project-: This project is the completed assignment work for the course "Professional Academy Diploma in Front-End-Development

---

Project Structure

The project is structured as follows:

```
project-root/
├── index.html   # Main HTML structure
├── style.css    # CSS file for styling
├── script.js    # JavaScript file for interactivity
├── images/      # Directory containing images and video assets
└── README.md    # Project documentation
```

Design

The website design is clean and vibrant. The album art used on the artists latest single is set as the background for the home page. The website has a predominantly pink background with contrasting text colours. This results in a website that is easily readable and aesthetically pleasing to look at. Key design principles include:

- **Consistency:** The colour scheme and font usage has a similar colour pattern throughout.
- **Responsiveness:** A layout that adapts to different screen sizes. This means that the website is attractive and readable on all devices.
- **Intuitiveness:** Easy navigation is delivered through elements such as hover effects and interactive tabs.

The following tools and frameworks were used:

- HTML5 for structure
- CSS3 for styles
- FontAwesome for icons
- Media queries for responsive design

Website Structure

- **Header**
- **About Section**
- **Services Section**
- **Portfolio Section**
- **Contact Section**

- **Footer**

Components

- **Header:** Contains links to each section and header text. Features a responsive design.
- **About Section:** Interactive tabs manufactured through JavaScript. These tabs are named skills, experience, and education. Information is given on each topic when the tab is opened.
- **Services Section:** A grid layout is used to describe each service offered by the individual. A “See more” button, that leads to the Contact section, is added to each grid.
- **Portfolio Section:** The artist’s work, which includes images and video projects, is showcased here. Hover overlays are added, along with links to SoundCloud and YouTube where further content can be viewed.
- **Contact Section:** Formspree API is used to create a contact section that functions correctly. Messages are simultaneously sent to a Formspree account and the artist's email. Links to the individual’s Instagram, LinkedIn, Facebook and X are added as an alternative way of connecting with the composer.
- **Footer:** Contains copyright information.

Code Structure

HTML

- **Semantic Tags:** These tags, such as header, section, nav, and footer, play a crucial role in organizing code, making it more readable and enhancing the structure and meaning of the content.
- **Responsive Meta Tags:** These tags are used in HTML to provide proper display of webpages across different devices and screen sizes. Essential for appropriate layout on mobile devices. Meta information from this project is provided below:

```
html
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible"
content="IE=edge">
<meta name="viewport"
content="width=device-width,
initial-scale=1.0">
```

CSS

- **Grid and Flexbox Layouts:** Used to arrange and create a creative layout for the project's sections, eg. Services, as well as content embedded in these sections.
- **Media Queries:** Allows for a fitting design on mobile devices.
- **Hover Effects:** Makes it obvious to the user what elements of the project are interactive.

Third Party APIs

- **FontAwesome:** Icon drawing in the navigation and services pages is possible as a result of FontAwesome.
- **Formspree:** Manages the server side of the contact form. Results in a platform for messages sent to be stored and viewed by the individual.

Integration Between Frontend and Backend

Frontend

- HTML/CSS for layout and styling.
- JavaScript for dynamic tab functionality.

Backend (Formspree)

- Handles email submissions securely from the contact form.

User Stories

- **As a visitor**, I want to view the artist's portfolio to understand their capabilities.
- **As a client**, I want to easily book services through a contact form.
- **As a collaborator**, I want to download the CV for reference.

Best Practices and Clean Code Structure

- **Separation of Concerns:** HTML, CSS, and JavaScript are modularized.
- **Accessibility:** Includes alt attributes for images and aria-labels for icons.
- **Responsive Design:** Tested across multiple screen sizes.

Website Improvements

Additional Features

- Adding a blog section to highlight achievements and news.
- Incorporating an interactive audio player.

Performance Optimization

- Minify CSS and JavaScript files.
- Use lazy loading for images.

Testing

Testing included:

- **Cross-Browser Testing:** Established compatibility between Chrome, Firefox, and Safari.
- **Responsiveness Testing:** Verified functionality using Chrome DevTools across different devices.
- **Form Validation:** Messages were sent to through the website's contact form. It could then be confirmed that the functionality of this form was correct as these messages could be viewed in both Formspree and the appropriate email address.

Hosting and Deployment

- Hosted using GitHub Pages.
- Deployed by pushing updates to the main branch.

Credits

- **Icons:** FontAwesome
- **Background Images:** Courtesy of Unsplash
- **Form Handling:** Formspree
- **Video Content:** Provided by the artist.

Technical Specifications

JavaScript Features

- **Menu Functionality:** The `openMenu` and `closeMenu` functions dynamically adjust the side menu's position by manipulating its CSS right property.
  Example:

  ```
  1 javascript
  ```

  Copy code

  ```
  function openMenu() { document.getElementById('sidemenu').style.right = '0'; }
  ```

- **Tab Switching Logic:** Utilizes querySelectorAll to manipulate classes for active tabs.
  Event handling makes the tabs interactive, enabling content updates without page reloads.
  Example:

  ```
  1 javascript
  ```

  Copy code

  ```
  function openTab(tabName, event) { document.querySelectorAll(".tab-contents").forEach(tab => { tab.classList.remove("active-tab"); }); document.getElementById(tabName).classList.add("active-tab"); event.currentTarget.classList.add("active-link"); }
  ```

CSS Features

- **Transitions and Animations:** Applied to hover effects and menu sliding to enhance user engagement.
- **Media Queries:** Ensures that the layout adjusts fluidly for screens ranging from smartphones to desktops.

Performance Metrics

- **Loading Speed:** Website scored 90+ on Google Lighthouse for both desktop and mobile performance.
- **Optimization Techniques:** CSS and JavaScript were minified to decrease file sizes.
- Leveraged browser caching for static assets (e.g., images, styles).

Security Measures

- **Formspree Protection:** Form submissions are validated via Formspree, preventing SQL injection and spam.
- **HTTPS:** The website uses GitHub Pages' HTTPS support for secure data transmission.
- **Alt Text and Aria Labels:** Ensures accessibility compliance, avoiding information leakage or phishing risks for screen readers.

User Experience Enhancements

- **Interactive Elements:** Hover effects on portfolio items and navigation links provide immediate visual feedback to users.
- **Keyboard Accessibility:** Focus states are clearly defined for tab navigation.
- **Responsive Design:** The grid layout and media queries ensure seamless usability on mobile devices.

SEO Considerations

- **Meta Tags:**

  ```
  1 html
  ```

  Copy code

  ```
  <meta name="description" content="TH.STE's portfolio: Discover a music producer and DJ's services, achievements, and contact information."> <meta name="keywords" content="Music producer, DJ, Portfolio, Contact TH.STE">
  ```

- **Image Optimization:** Alt attributes are descriptive and aligned with keyword strategy.
- **Structured Data:** Consider adding schema markup (e.g., Person, CreativeWork) to improve search visibility.

Future Development Plans

Audio Player Integration:

- An embedded, interactive audio player will allow users to sample tracks directly on the site.

Blog Section:

- A dedicated section for updates, achievements, and industry insights.

Provide Clear signs of Consistent Commitments

- The consistently pushed to the GitHub repository named assignmentproject. Due to some bugs, the code was edited in GitHub but not in the local server. This became evident that this was bad practice and will not be repeated in future projects as it made things confusing. This resulted in a new repository being set up to upload the final project.

Testing Process

- **Cross-Browser Compatibility:** Verified on Chrome, Firefox, Safari, and Edge.
- **Accessibility Testing:** Tools like WAVE and Axe identified minimal barriers for screen readers.
- **Focus and hover states** were tested to ensure keyboard navigation.
- **Form Validation:** Test submissions confirmed successful delivery via Formspree.

Deployment Process

Hosting:

- The site is deployed via GitHub Pages, with changes automatically reflected when pushed to the main branch.

Steps:

- Clone repository: `git clone <repo-url>`
- Make updates.
- Push changes: `git push origin main`.

Known Issues and Limitations

- **Browser-Specific Quirks:**
  The CSS transitions on tabs may appear slower on older versions of Internet Explorer.
- **Formspree Limitation:**
  The free tier of Formspree restricts submissions to 50/month