

# **Personal Portfolio Website**

Steven Filbert

September 29, 2025

# Table of Contents

<b>Executive Summary .....</b>	<b>3</b>
<b>1. Introduction .....</b>	<b>4</b>
1.1. Project Motivation .....	4
1.2. Goals and Objectives .....	4
<b>2. Planning and Design Phase .....</b>	<b>4</b>
2.1. Content Strategy .....	4
2.2. Design & User Experience (UX) .....	5
2.3. Technology Stack Selection .....	5
<b>3. Development and Implementation .....</b>	<b>5</b>
3.1. Project Setup .....	5
3.2. HTML Structure .....	6
3.3. CSS Styling .....	6
3.4. Deployment .....	7
<b>4. Challenges and Solutions .....</b>	<b>7</b>
4.1. The Challenge: Responsive Layout .....	7
4.2. Troubleshooting and Solution .....	7
4.3. Key Learnings .....	8
<b>5. Future Improvements .....</b>	<b>8</b>
<b>6. Conclusion and Key Learnings .....</b>	<b>9</b>
<b>References .....</b>	<b>10</b>

## Executive Summary

This document details the planning, design, and implementation of a personal portfolio website created by a recent Bachelor of Cyber Security graduate, **specializing in Governance, Risk, and Compliance (GRC)**, to establish a strong professional brand for the job search. The project was developed using foundational technologies including semantic HTML5, modern CSS3 with Flexbox, and vanilla JavaScript. Version control was managed using Git, and the site was deployed using GitHub Pages. The outcome is a clean, fully responsive website that successfully centralizes all professional assets (CV, LinkedIn, GitHub) and serves as a dynamic showcase of **specialized cybersecurity skills and relevant projects**.

# 1. Introduction

## 1.1. Project Motivation

As a recent graduate with a Bachelor of Cyber Security **specializing in GRC**, entering a competitive technology job market required the creation of a strong, professional online presence. While a traditional resume and LinkedIn profile are essential, they are static and limited in their ability to fully showcase **specialized technical skills relevant to GRC and data privacy**. This project was undertaken to create a dynamic, central hub that links all my professional assets—resume, LinkedIn, and GitHub—into one cohesive platform. It serves as a digital handshake, offering recruiters and hiring managers a more in-depth and interactive look into my capabilities and projects beyond the constraints of a single-page document.

## 1.2. Goals and Objectives

The development of this portfolio website was guided by a set of clear and measurable objectives:

- **To Demonstrate Specialist Technical Proficiency:** The primary goal was to showcase hands-on proficiency in **cybersecurity concepts (GRC, Data Privacy, Forensics)** alongside core front-end technologies (HTML5, CSS3, JavaScript) used to build the platform itself.
- **To Create a Central Professional Hub:** The website is designed to be the single source of truth for my professional profile, providing direct access to my optimized resume, project summaries (including GRC-relevant university work), and professional network links.
- **To Gain Practical End-to-End Development Experience:** This project provided an opportunity to manage a complete development lifecycle relevant to deploying web-based tools or reports, from planning and design, through coding and Git version control, to final deployment on GitHub Pages.
- **To Enhance Professional Branding (as a GRC Specialist):** By designing and building a clean, modern website focused on my GRC specialization, the objective was to create a positive first impression reflecting a commitment to quality and relevant expertise.

# 2. Planning and Design Phase

## 2.1. Content Strategy

The content was strategically chosen to guide a potential employer from a general introduction to specific proof of **GRC-related technical skill and experience**. The information architecture consists of five key sections:

- **Hero Section:** Serves as the immediate introduction, presenting my name and **professional focus (GRC, Data Privacy, Digital Forensics)**, with direct calls-to-action to LinkedIn, GitHub, and my downloadable **GRC-focused CV**.
- **About Section:** Expands on the introduction, offering a brief professional narrative that mirrors my CV and LinkedIn profile, emphasizing my **GRC specialization and relevant internship experience**.

- **Skills Section:** A visually scannable overview of my key **GRC and cybersecurity competencies (e.g., GRC, ISO 27001, Data Privacy, Digital Forensics, Python, Azure, XGBoost)**, designed for quick assessment.
- **Projects Section:** The core evidence, presenting projects like the **SecureLink PPRL Platform** (highlighting XGBoost, LDP, and privacy trade-off analysis) and the SafeUS mobile app. Each includes a description, technologies used, and a link to the relevant **project summary PDF** (for SecureLink).
- **Contact Section:** A final call-to-action.

## 2.2. Design & User Experience (UX)

The design philosophy was centred on creating a clean, professional, and minimalist aesthetic that keeps the focus on the content. A mobile-first, responsive design was a primary technical requirement to ensure a seamless experience across all devices.

- **Layout:** A single-page vertical scroll was chosen for its straightforward user journey, allowing information to be presented in a logical, story-like progression.
- **Colour Scheme:** A simple and professional palette was selected, using a dark charcoal for the background, white for text to ensure high readability, and a single accent colour to draw attention to key elements like links and buttons.
- **Typography:** The font family **Roboto** was chosen for its excellent on-screen readability and modern, neutral feel. A clear hierarchy was established using different font weights (**Bold** for titles) and sizes to guide the user's eye through the content.

## 2.3. Technology Stack Selection

The technologies for this project were selected to demonstrate a strong command of foundational web development principles.

- **Front-End:** The website was built using foundational HTML/CSS and JavaScript [9] without a front-end framework. This approach was deliberately chosen to showcase a deep understanding of core web technologies, including semantic HTML structure, modern CSS for layout and responsiveness, and vanilla JavaScript [9] for interactivity.
- **Hosting:** GitHub Pages was selected as the hosting platform. While I have familiarity with other platforms like Firebase and Microsoft Azure [8], GitHub Pages was the ideal choice for this project due to its seamless integration with the Git version control workflow, its zero-cost hosting for public static sites, and its widespread use for developer portfolios. The more powerful features of Firebase or Azure, such as backend services and databases, were not required for this static portfolio site.

# 3. Development and Implementation

## 3.1. Project Setup

The development environment was chosen for efficiency and adherence to modern web development standards. The primary code editor used was Visual Studio Code (VSCode [5])

[5] (VSCode [5]), selected for its robust features, integrated terminal, and extensive extension support.

Version control was managed using **Git**, with a remote repository hosted on **GitHub**. This allowed for systematic tracking of changes, a clear commit history, and a straightforward deployment pipeline. The project was organized with a clean and logical file structure to ensure maintainability:

```
/vennfilbert.github.io
├─ /assets/
│  ├── Steven Filbert - CV.pdf
│  └── Steven Filbert - Data Privacy Project Summary.pdf
├─ /css/
│  └── style.css
├─ /images/
│  ├── /safeus/
│  │  ├── HomeScreen.png
│  │  └── (Other SafeUS images...)
│  ├── /securelink/
│  │  ├── SecureLink.png
│  │  └── (Other SecureLink images...)
│  └── logo.png
├─ /js/
│  └── script.js
├─ index.html
└─ README.md
```

### 3.2. HTML Structure

The website was built using semantic HTML5 [1] to create a well-structured, accessible, and SEO-friendly document. Rather than relying on generic `<div>` tags for all containers, semantic elements were used to define the purpose of each section of the page.

- `<header>` was used for the introductory hero section containing my name and title.
- `<nav>` was implemented for the main navigation links.
- `<main>` wrapped the primary content of the page, including the distinct About, Skills, and Projects sections.
- `<section>` tags were used to group the content within each thematic block (e.g., `<section id="projects">`).
- `<footer>` contained the final contact information and social media links.

This approach not only improves the machine readability of the site for search engines but also enhances accessibility for users relying on screen readers.

### 3.3. CSS Styling

A mobile-first approach was adopted for styling, ensuring the website would be perfectly functional and readable on small screens before scaling up to larger displays. The layout was

primarily managed using CSS Flexbox [6], which provided a powerful and efficient way to align and distribute space among items in a container.

Responsiveness was implemented using media queries [7] targeting standard device breakpoints (e.g., tablets at 768px and desktops at 1024px). These queries adjust CSS properties like font sizes, container widths, and Flexbox layouts to optimize the user experience for different screen sizes, ensuring the design never appears broken or crowded.

### 3.4. Deployment

The website was deployed using GitHub Pages [4], a static site hosting service directly integrated with the project's GitHub repository. The process was streamlined and efficient:

1. A new public repository was created on GitHub with the specific name `vennfilbert.github.io`.
2. The local project folder was initialized as a Git [3] repository.
3. The remote GitHub repository was added as the origin.
4. All project files were committed to the main branch.
5. The local commits were pushed to the remote origin on GitHub using `git push origin main`.

Within minutes of the final push, GitHub Pages [4] automatically built and deployed the site, making it publicly accessible at the URL.

## 4. Challenges and Solutions

### 4.1. The Challenge: Responsive Layout

During the development of the portfolio website, a significant technical challenge arose in implementing a fully responsive layout for the "Projects" section. This section details the problem, the troubleshooting process, and the key learnings from the experience.

While the project cards were correctly styled and aligned on a desktop view, the layout would break on smaller screen sizes, particularly on mobile devices. The cards, arranged horizontally, would overflow their container rather than re-stacking vertically. This resulted in a horizontal scrollbar appearing and made the content unreadable, creating a poor user experience which undermined the project's goal of appearing professional and polished.

### 4.2. Troubleshooting and Solution

The troubleshooting process began by using the browser's built-in developer tools to inspect the CSS properties on a simulated mobile viewport. This analysis revealed that the container for the project cards was using a fixed layout that was not adaptable to smaller screens.

The problem was resolved by refactoring the CSS to adopt a mobile-first design philosophy, using modern layout techniques.

1. Initial State (Mobile-First): The CSS for the project container was initially styled for mobile devices, using CSS Flexbox [6] with `flex-direction: column`; This ensured that on small screens, the project cards would stack vertically by default.

2. **Implementing a Breakpoint:** A **media query** was then introduced to handle larger screens. The query (`@media (min-width: 768px)`) targets tablets and desktops.
3. **Final Solution (Desktop State):** Inside the media query, the container's styles were overridden. The flex-direction was changed to row, and flex-wrap: wrap; was added. This allowed the project cards to arrange themselves in a flexible, horizontal grid on larger screens, wrapping to the next line if space was limited.

This solution not only fixed the layout issue but also resulted in a more robust and maintainable stylesheet that was optimized for a wide range of devices.

#### 4.3. Key Learnings

This challenge provided several valuable lessons in front-end development:

- It offered deep, practical experience in implementing a mobile-first responsive design strategy, which is a modern best practice.
- It reinforced the critical importance of using browser developer tools for real-time inspection and debugging of CSS layouts.
- It strengthened my skills in advanced CSS properties, specifically Flexbox and the use of media queries [7], which are fundamental to creating any modern, user-friendly website.

## 5. Future Improvements

While the current version of the portfolio website successfully meets its primary objectives, several potential enhancements have been identified for future development. These improvements would add more functionality, improve user experience, and provide opportunities to demonstrate a broader range of technical skills.

- **Dark Mode Toggle:** Implement a user-selectable dark mode. This would not only be a popular user experience feature but would also provide an opportunity to demonstrate more advanced JavaScript [9] and CSS skills, such as using CSS variables and handling user preferences with local storage.
- **Blog Section:** Add a simple blog where I can write about technical projects, cybersecurity topics, and key learnings from certifications like the **SC-900 or CompTIA Security+**. This would showcase my ongoing engagement with the tech industry and provide a platform to demonstrate my expertise in specific subjects.
- **Dynamic Contact Form:** Upgrade the current contact section to include a functional contact form. This would involve using a serverless function, potentially hosted on Microsoft Azure [8], to process the form data and send it as an email. This would add dynamic functionality to the site and demonstrate experience with cloud services.
- **Project Filtering:** As the number of projects grows, a filtering system could be added to allow visitors to sort projects by category (e.g., "Cybersecurity," "Web Development," "Data Analysis"). This would improve the user experience and demonstrate more advanced JavaScript [9] DOM manipulation skills.



## 6. Conclusion and Key Learnings

The personal portfolio website project successfully met all its initial objectives. The result is a clean, fully responsive, and professionally branded website that serves as a central hub for my professional profile. It effectively showcases my technical projects and skills, providing a dynamic and interactive alternative to a traditional resume.

The process of building this website from planning to deployment was a valuable learning experience that strengthened a range of technical and professional skills:

- **Responsive Web Design:** Gained significant practical experience in implementing a mobile-first design philosophy using modern CSS techniques like Flexbox and media queries [7] to ensure a seamless user experience across all devices.
- **Version Control Workflow:** Solidified my understanding of **Git** and GitHub by managing the project through a standard workflow of committing changes, pushing to a remote repository, and deploying directly from the main branch.
- **End-to-End Project Management:** This project provided hands-on experience in managing a project through its entire lifecycle, from the initial blueprint and design phase to development, debugging, and final deployment.
- **Technical Problem-Solving:** Enhanced my debugging skills by using browser developer tools to diagnose and resolve layout and styling issues, reinforcing a systematic approach to troubleshooting.

# References

- [1] W3C, \*HTML5 Specification\*. World Wide Web Consortium, 2017. [Online]. Available: <https://www.w3.org/TR/html5/>
- [2] MDN Web Docs, \*Responsive Design\*. Mozilla Developer Network, 2024. [Online]. Available: [https://developer.mozilla.org/en-US/docs/Learn/CSS/CSS\\_layout/Responsive\\_Design](https://developer.mozilla.org/en-US/docs/Learn/CSS/CSS_layout/Responsive_Design)
- [3] S. Chacon and B. Straub, \*Pro Git\*, 2nd ed. Apress, 2014. [Online]. Available: <https://git-scm.com/book/en/v2>
- [4] GitHub Docs, \*GitHub Pages\*. GitHub, 2024. [Online]. Available: <https://docs.github.com/pages>
- [5] Visual Studio Code Docs, \*Visual Studio Code Documentation\*. Microsoft, 2024. [Online]. Available: <https://code.visualstudio.com/docs>
- [6] MDN Web Docs, \*CSS Flexible Box Layout (Flexbox)\*. Mozilla, 2024. [Online]. Available: [https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_Flexible\\_Box\\_Layout](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout)
- [7] MDN Web Docs, \*Using Media Queries\*. Mozilla, 2024. [Online]. Available: [https://developer.mozilla.org/en-US/docs/Web/CSS/Media\\_Queries/Using\\_media\\_queries](https://developer.mozilla.org/en-US/docs/Web/CSS/Media_Queries/Using_media_queries)
- [8] Microsoft Azure Docs, \*Azure Functions Overview\*. Microsoft, 2024. [Online]. Available: <https://learn.microsoft.com/azure/azure-functions/>
- [9] MDN Web Docs, \*JavaScript Guide\*. Mozilla, 2024. [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>