

# A critical reflection on refactoring the Web Witchcraft and Wizardry project

Edmund Mulligan

December 17, 2025

## Abstract

*Abstract goes here.*

## Introduction

Web Witchcraft and Wizardry (Mulligan, 2024) is a website for children to learn the basics of web development through fantasy-themed metaphors. The original version of the website was neither accessible (except by accident) nor responsive. This essay reflects on the design and implementation of a refactored version of the project, focusing on accessibility, usability, and quality of the new code. Detailed decisions are documented as comments in the source code, which should be read alongside this essay, so the essay focuses on higher level design decisions and reflections.

The development process used was iterative and incremental (Beck and Andres, 2004), and while there are many criticisms of this approach, particularly around its lack of documentation, endless changes and lack of scalability to large teams (see, for example, (Boehm and Turner, 2004)), it was well suited to this small project with a single developer fulfilling all team roles.

## Responsiveness and Accessibility

A primary goal of the refactor was to make the website responsive and accessible.

## Site Architecture and Navigation

information architecture and layout choices navigation design site navigation vs page navigation

## Third party credits and resources

All code submitted for assessment is my own work. No code has been copied from any external sources, such as online tutorials or code. I have, however, relied heavily on two YouTube channels, those of Kevin Powell (Powell, 2025) and Coding2GO (Coding2GO, 2025), for guidance on best practices in responsive web design. I have also used Stack Overflow (Stack Overflow, 2025) extensively to troubleshoot specific issues that I encountered.

The automation tools used for testing accessibility and performance (Lighthouse, axe, pa11y, Wave) are all open source tools developed by third parties and the scripts used to run them were

adapted from examples found in their respective documentations. I was assisted by a colleague at my employer, Transport for London, in writing the playwright tests used for testing browser compatibility. The automation scripts are included in the git repository but are not part of the submission for assessment.

## Testing

## Outstanding issues and Future Improvements

The project submitted for assessment is phase one of a larger project to refactor and improve the Web Witchcraft and Wizardry website. Some content was deliberately omitted from this phase as it will be developed in phase 2 using JavaScript. There are also some known issues and areas for improvement that could not be addressed within the time constraints of this phase.

Bug Id	Description	Symptoms	Work around
1	Reactive code does not work well at extreme viewport sizes	Text overflows boxes or content does not fill screen properly.	Warning displayed when viewport is too small.

Table 1: Known issues and workarounds

responsive design techniques Fluid grids, flexible images, and media queries (Marcotte, 2010)  
prefer responsive Viewport cutoffs ; 80px too small – unusable 80px – 200px very small – give warning but still usable 200px – 800px small – mobile layout – prefer vertical stacking ; 800px – desktop layout – prefer horizontal layout  
accessibility features implemented  
grid vs flexbox use of media queries semantic HTML ARIA roles and attributes colour contrast keyboard navigation alt text for images form accessibility  
performance improvements (minification, image optimisation)  
dry code (Hunt and Thomas, 1999)  
automated testing (Lighthouse, axe, pa11y, Wave) manual testing (screen readers, keyboard only navigation)

## Conclusion

## Word count

Word count: 460 words (excluding references)

## References

- Kent Beck and Cynthia Andres. *Extreme Programming Explained: Embrace Change*. Addison-Wesley Professional, Boston, MA, 2nd edition, 2004.
- Barry Boehm and Richard Turner. Balancing agility and discipline: A guide for the perplexed. *IEEE Software*, 21(6):57–58, 2004.
- Coding2GO. Coding2go, 2025. URL <https://www.youtube.com/@Coding2G0>. Accessed: December 2025.

- Andrew Hunt and David Thomas. *The Pragmatic Programmer: From Journeyman to Master*. Addison-Wesley Professional, Boston, MA, 1999.
- Ethan Marcotte. Responsive web design. *A List Apart*, 306, May 2010. URL <https://alistapart.com/article/responsive-web-design/>. Accessed: December 2025.
- Edmund Mulligan. Web witchcraft and wizardry. <https://github.com/edmundmulligan/Witches-And-Wizards>, 2024. Accessed: November 2025.
- Kevin Powell. Kevin powell — css evangelist, 2025. URL <https://www.youtube.com/@KevinPowell>. Accessed December 2025.
- Stack Overflow. Stack overflow. <https://stackoverflow.com>, 2025. Accessed: December 2025.