

1. HW3 Scope and What Was Tested

HW3 expanded the HW2 email campaign prototype by improving button design, responsive hero behavior, single-column layout, background images behind text, updated image assets, and accessibility validation using professional tools.

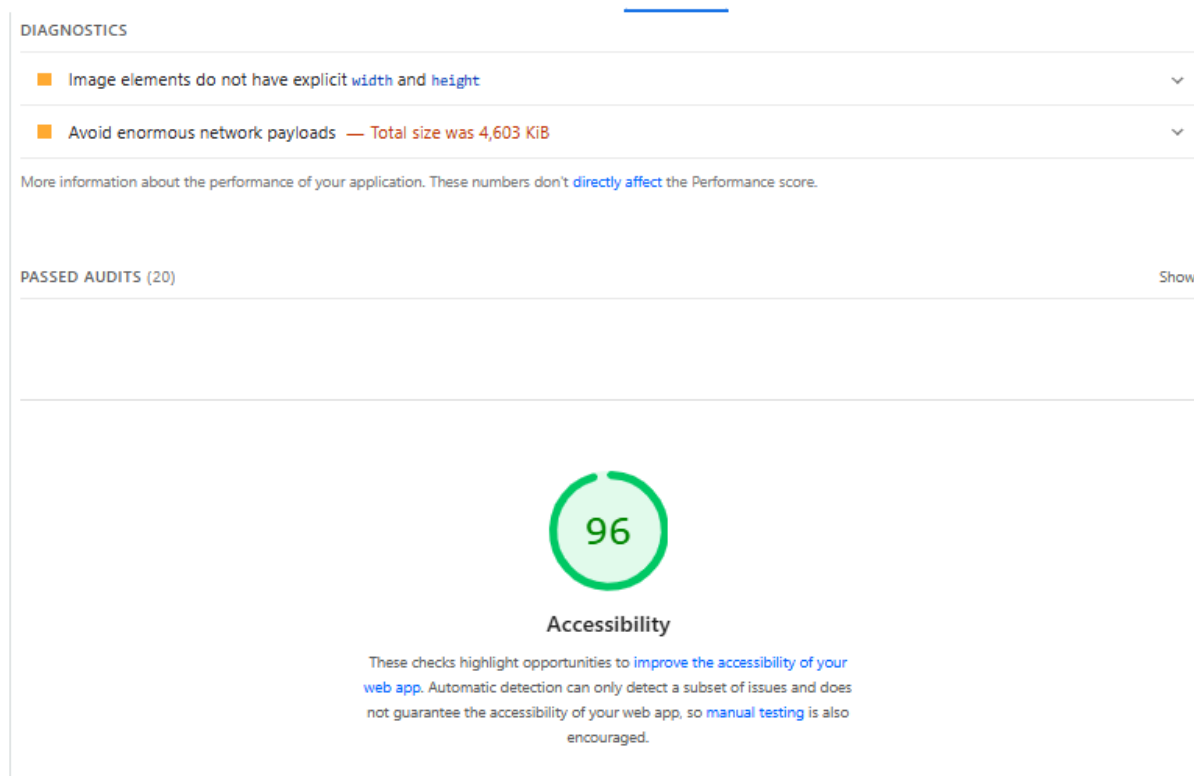
Pages Tested: index.html and signup.html

2. Accessibility Tools Used

Chrome DevTools Lighthouse/PageSpeed Checker (Accessibility)

Pros: Built-in, fast scan, numeric score, highlights common WCAG issues. I used PageSpeed to check by using the pages url (the full report here:

https://pagespeed.web.dev/analysis/https-m-ph484-github-io-comp584-hw3MK/7aw3aa214i?form_factor=desktop)



Cons: Limited contextual understanding; automated checks only.

axe DevTools

Pros: Auto-rule checks, actionable remediation guidance. Very useful for real-time testing, so checking between github and the open window to re-run scans was easy to toggle. Gave informative

feedback after running scans as well, down to what element or child of an element needed to be adjusted.

Cons: Automated; does not replace manual testing. Fixing one thing led to flags saying there were a lot more errors than before. To the best of my ability, these were reduced to the number of errors below, prioritizing the removal of all Critical issues, and almost removing all Serious issues (that last one has been tricky. Anything missed in this HW assignment will be fixed by the next, if we continue working on this specific campaign).

Toggle Dark

Toggle Reader

Toggle Images

Web preview only • Keyboard: Tab to buttons, Enter to toggle

RAGEQUIT PRESS • EMAIL CAMPAIGN PREVIEW

The Week in Gaming, With Zero Mercy

Issue #02 • Click any image to toggle "Evidence Mode."

RAGEQUIT PRESS

WEEKLY ROAST • ISSUE #

QRP

TO TOGGLE GLITCH MODE

Evidence mode (click to swap back).

Under construction: When we add a backend (Django), this page will: (1) create a subscriber profile, (2) store preferences, and (3) trigger email notifications for newsletters and new cross-media uploads.

ax

DevTools

axe-core 4.11.0

Sign up / Sign in

+

Overview

Guided Tests

Test Name

Save Test

Test URL

Re-run scan

https://m-ph484.github.io/comp584-hw3MK/

TOTAL ISSUES

5

Automatic Issues.....

5

Guided Issues.....

0

Manual Issues.....

0

Best Practice.....

4

Critical.....

0

Serious.....

1

Moderate.....

3

Minor.....

1

ax

DevTools

axe-core 4.11.0

Sign up / Sign in

+

Overview

Guided Tests

Test Name

Save Test

Test URL

Re-run scan

https://m-ph484.github.io/comp584-hw3MK/signup.html

TOTAL ISSUES

2

Automatic Issues.....

2

Guided Issues.....

0

Manual Issues.....

0

Best Practice.....

2

Critical.....

0

Serious.....

0

Moderate.....

2

Minor.....

0

Best Practices: ON

WCAG 2.1 AA

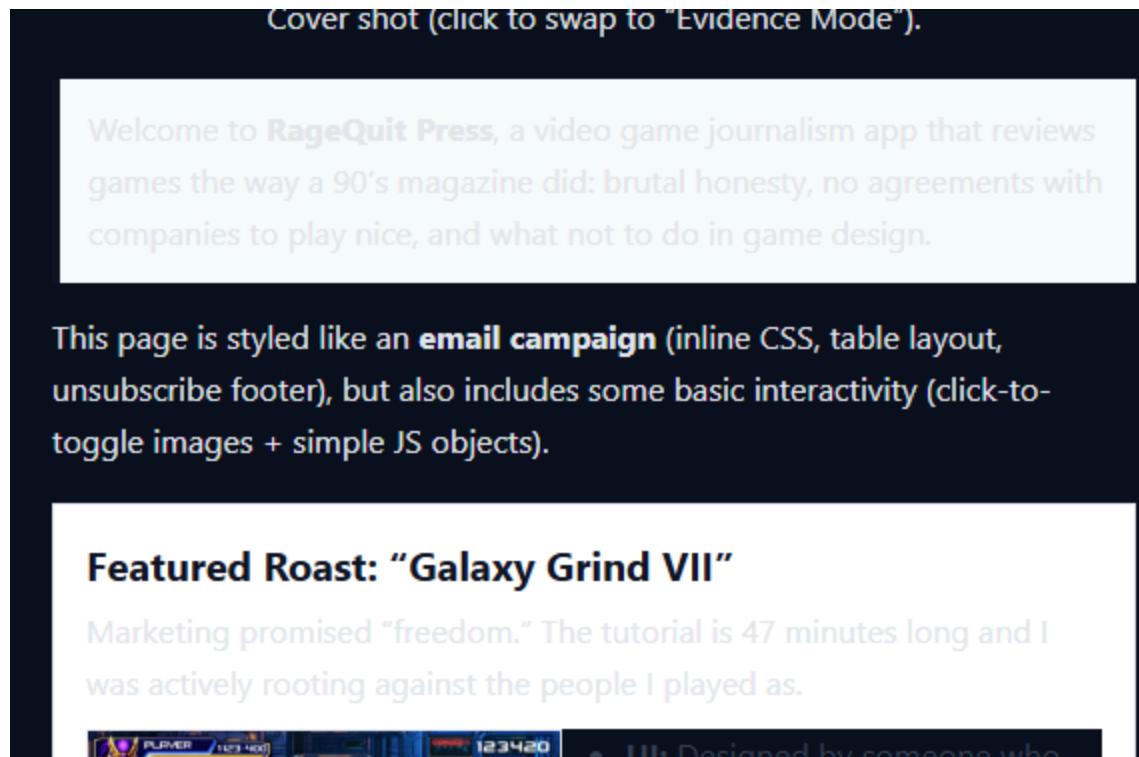
Share

Download

(Some of the recommendations that axe gave were helpful. Both pages were reduced to these issues, a big improvement from before using the dev tools)

Manual Testing

Keyboard navigation validation, focus testing, skip link verification, reader mode testing.



(sometimes glitches happened and tests had to be done to troubleshoot, shown above with the dark mode css conflict)

3. Accessibility Changes Implemented

Skip Link + Main Landmark

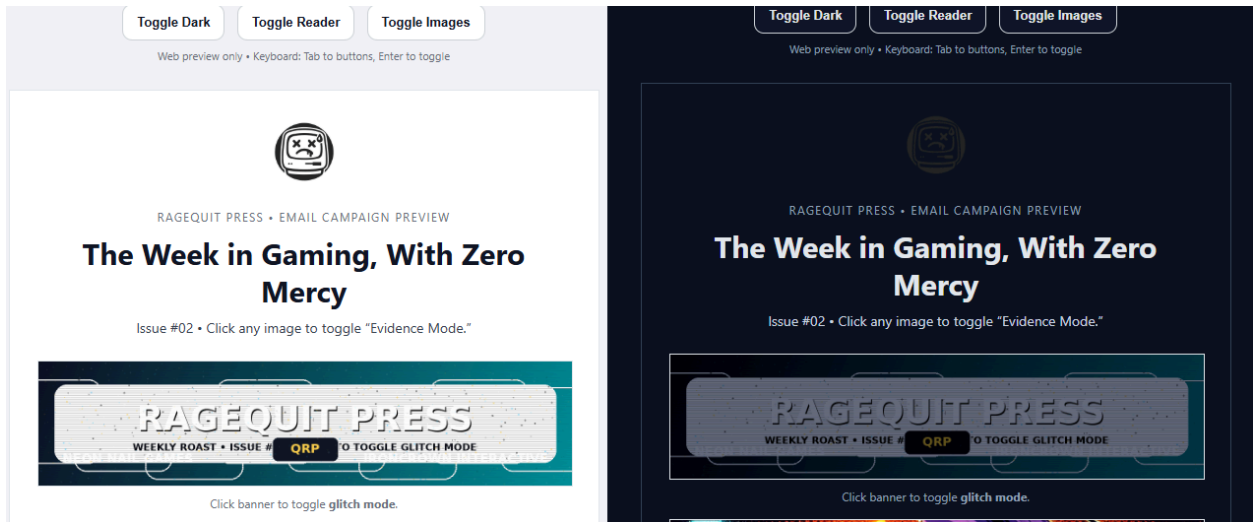
Added skip link and role='main' landmark to improve navigation for assistive technologies. Press tab to focus on main content

```
</head>

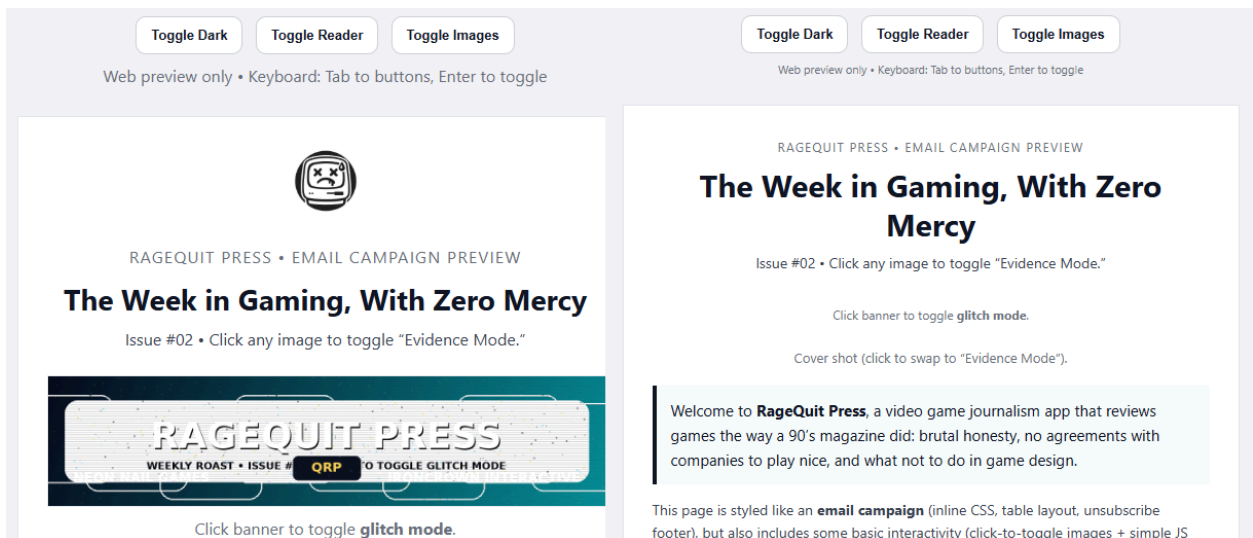
<body id="body" style="margin: 0 !important; padding: 0 !important; background-color: #f3f4f6;">
  <!-- Skip link improves keyboard navigation (especially for screen reader and power users) -->
  <a href="#main" style="position:absolute; left:-9999px; top:auto; width:1px; height:1px; overflow:hidden;">
    Skip to main content
  </a>
```

Dark Mode, Reader Mode, Skip Images Mode

User-controlled accessibility toolbar implemented. Dark mode required CSS overrides due to email-style inline backgrounds.



(Light vs Dark mode - Banner changes too)



(reader-mode on the left, no-image mode on the right)

Dark Banner Fix

Implemented JavaScript-based banner swapping because banner is an ``, not a CSS background.



(banner changes in dark mode, too, and is mindful of keeping the 'darker' theme)

Button Accessibility Improvements

Implemented both bulletproof (table-based) and standard anchor buttons, centered with larger tap targets.

```
<!-- =====
PRIMARY CTA
Purpose:
- Clear conversion action (subscribe) in layout
- When this is made to work for real, this will go to an external landing page
Edit points:
- Change href to a real route later (signup.html / React route / etc.)
===== -->
<div style="margin-top:18px; text-align:center;">
  <a
    href="signup.html"
    id="ctaSubscribe"
    style="display:inline-block; padding:12px 18px; background-color:#111827; color:#ffffff; text-decoration:none; border-radius:6px; font-size:14px; line-height:14px;
    Subscribe for Roasts & Alerts
  </a>
  <div style="margin-top:10px; font-size:12px; line-height:18px; color:#6b7280;">
    (CTA now links to signup.html – a frontend-only landing page preview.)
  </div>
</div>
```

Single-Column Layout

Converted feature section to stacked single-column format for reliability and accessibility.

Click badge to toggle **ROASTED** ⇌ **GODLIKE** (web preview).

Quick Hits

Three headlines, one mood (this block is populated by a JS array of objects):

Patch Notes Apology Tour — “We fixed the bug where the game launched.”

Indie Gem: “Wicker Dungeon” — “Actually fun. Suspicious.”

Sequel Syndrome — “More of the same, priced like innovation.”

Broadcast Channels: Reviews, Shorts, and Live “Game Breaking”

Written reviews stay mostly polite. The **real roasting** happens on video —where we can show the bugs, the jank, and the “feature-not-a-bug” moments in real time. Subscribers can opt in to **email alerts** when new uploads go live on any channel.

(Before it would only turn to single-column mode if window was shrunk)

Form Accessibility (Signup Page)

Proper label-input pairing, clear CTA text, improved spacing, and validation structure.

Subscribe for Roasts & Alerts

(CTA now links to signup.html — a frontend-only landing page preview.)

Subscribe / Email Alerts / Newsletter

Email address

you@example.com

What do you want delivered?

- ☐ Instant alerts (launch train-wrecks)
- ☐ Weekly newsletter (the polite yelling)
- ☐ Video upload alerts (YouTube/TikTok/Twitch)
- ☐ Deals & releases (so you can avoid them faster)

Sign me up

4. Findings Summary from Tools

Contrast issues in early dark mode resolved via override rules. Heading hierarchy validated. ARIA roles and labels confirmed compliant.

5. Accessibility Pros and Cons

Pros: Strong keyboard support, clear landmarks, adaptive viewing modes, improved button sizing.

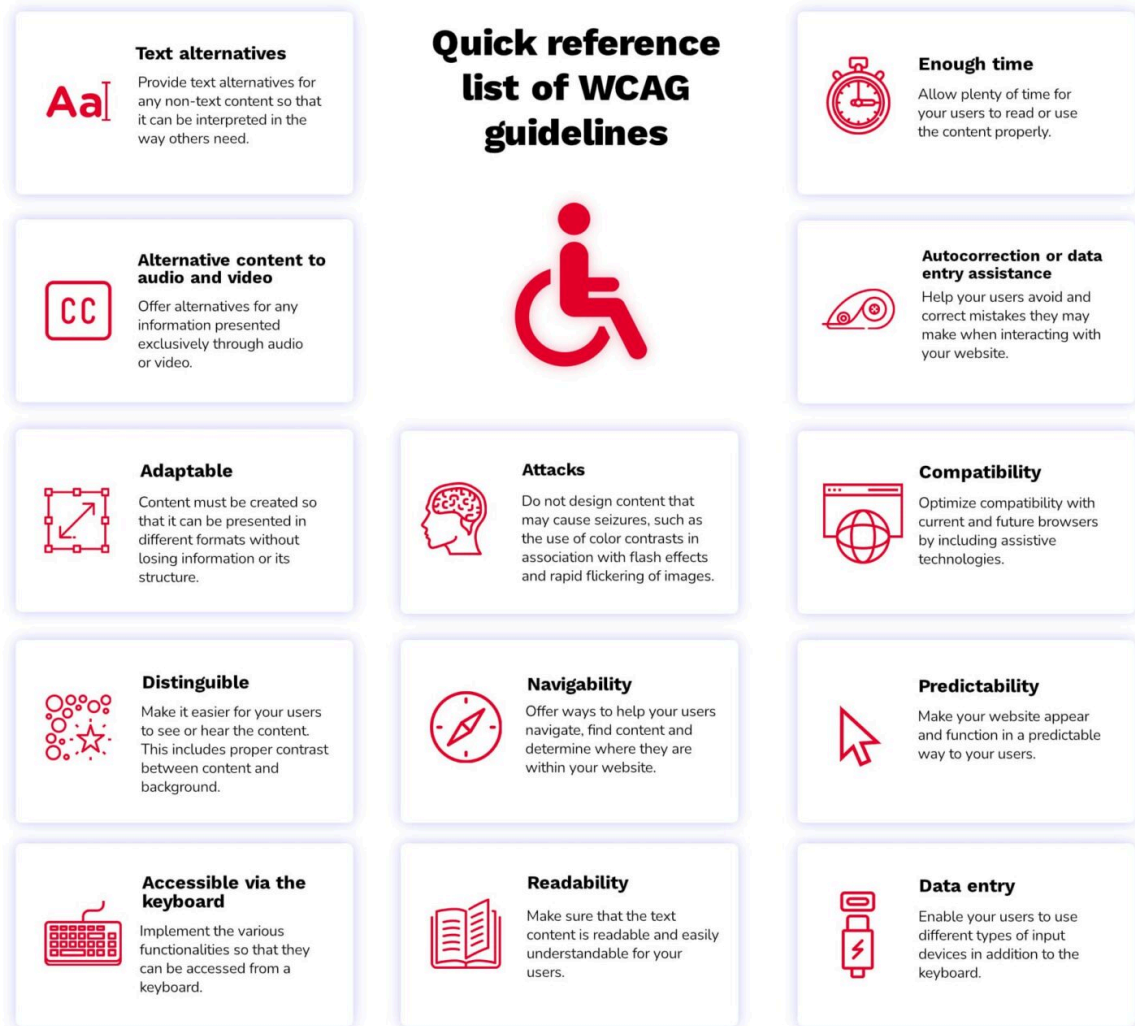
Cons: JavaScript-dependent features not supported in real email clients; inline CSS complicates theming.

6. Connection to Senior Design — WCAG Framework

Inclusion and Accessibility Web Content Accessibility Guidelines



My senior Design course covered WCAG principles: Perceivable, Operable, Understandable, Robust. These were taken into account when doing research and changing the design for HW 3 requirements. ([What is Web Accessibility \(WCAG\) and what are its guidelines? | Human Level](https://www.humanlevel.com/blog/what-is-web-accessibility-wcag-and-what-are-its-guidelines/))



Web accessibility | humanlevel.com



(the full document, separated from comp 491 slides screenshot)

Perceivable

Dark mode, reader mode, alt text, contrast fixes.

Operable

Keyboard support for toggles and navigation.

Understandable

CTA labels, logical heading hierarchy

Robust

Semantic HTML, ARIA roles (especially aria-labelby), form label structure.

7. Influence of External Sources (ByteByteGo + Medium)

Research from ByteByteGo and Medium influenced architecture planning and future stack decisions. These resources were recommended by my senior project professor.

Why MUI Would Be Useful Later

Centralized theming, built-in accessible components, improved dark mode implementation, standardized focus behavior.

Why Django Is Relevant for Accessibility

Server-side validation, accessible error binding, secure form processing, scalable structure.

8. Conclusion

HW3 requirements were satisfied and used some extra interactivity in the campaign. Accessibility tools were applied, WCAG principles followed, and a roadmap for scalable accessibility (MUI + Django) established.