Unit 7 Reflective Diary

Gloire Loic Nzoussi Moussounda

Athabasca University

Computer Science 266: Wed design

Unit 7 Reflective proposal diary

Unit 7 Reflective Diary – External Services

Unit 7 marked the final stretch of COMP 266 and, in many ways, the culmination of everything I had built in previous units. Where earlier assignments focused on structure, style, scripting, and enhancements, this unit pushed me to integrate **external services**, tools and APIs that extended my website beyond its own self-contained environment.

This was both exciting and intimidating. Suddenly, I wasn't just writing HTML, CSS, or JavaScript that I controlled. I had to connect with outside platforms like **Formspree, Hypothes.is, Giscus, and the Web Share API**, each of which came with its own setup process, IDs, privacy considerations, and quirks. This meant dealing with real network requests, authentication, and the possibility of external failure.

Looking back, I see Unit 7 as the bridge between "a course project" and "a usable product." In this reflection, I'll go over what I implemented, the challenges I faced, how these additions connected back to my Unit 1 personas, and what I learned from the process.

What I Did

1. Formspree Contact Integration

The first major task was upgrading my contact form. Previously, it had only used a mailto: action—functional but clunky and unreliable. With Formspree, I was able to add a **real backend service** without writing server-side code.

- Created a Formspree account and project.

- Generated a secure **form endpoint** (the /f/xxxxxxx URL).

- Replaced the placeholder in my HTML with this endpoint.

- Updated my main-jq.js file to submit data with fetch() and show toast notifications for success or failure.

The result was a contact form that could send structured feedback directly to my Formspree dashboard.

2. Hypothes.is Annotations

The second integration was **Hypothes.is**, a tool for web annotation. Rather than forcing it on users, I added a button that, when clicked, dynamically loaded the Hypothes.is embed script.

- Implemented **lazy loading** so the script only ran if requested.
- Ensured the button disabled after activation, showing "Annotations enabled."

- Added a toast notification for user confirmation.

Now, students can highlight text, make notes, and even collaborate if they join a shared Hypothes.is group.

3. Giscus Comments

The most challenging feature was **Giscus**, a GitHub Discussions–based comment system.

Steps I took:

- Created a public GitHub repository for the project.

- Enabled Discussions and created a “General” category.

- Retrieved the **repo ID** and **category ID** from the Giscus setup page.

Updated my HTML button with:

```
<button id="load-comments"
    class="secondary"
    type="button"
    aria-controls="giscus-thread"
    data-repo="gloireloic1-bot/gloireloic-comp266-companion"
    data-repo-id="R_kgDOPnQMGQ"
    data-category="General"
    data-category-id="DIC_kwDOPnQMGc4Cuy3S"
    data-mapping="pathname"
    data-reactions-enabled="1"
    data-emit-metadata="0"
    data-input-position="top"
    data-theme="preferred_color_scheme"
    data-lang="en">
  Load comments
</button>
```

-

- Configured main-jq.js to insert the Giscus <script> dynamically when the button was pressed.

Now, my site has a **live discussion thread** tied directly to GitHub Discussions.

4. Web Share API

Finally, I added a **native share button** to the site header.

- Used the navigator.share() method when available.

- Built a **clipboard fallback** so unsupported browsers still worked.

- Confirmed it was accessible with a button and ARIA label.

With this, students (or any visitor) can instantly share the page to their device's apps, whether messaging, social media, or email.

Challenges I Faced

Formspree

- **Issue**: At first, I kept using the **project link** instead of the form endpoint. This caused "Network error" messages.

- **Breakthrough**: Once I copied the /f/xxxxxxx endpoint from the Formspree dashboard, everything worked.

Giscus

- **Issue**: I struggled with repo IDs, category IDs, and confusing "X" errors in the setup screen.
- **Breakthrough**: After carefully checking my repo settings and copying the correct IDs, I was able to make it load dynamically.

Hypothes.is

- **Issue**: Technically easy, but conceptually harder—I wasn't sure why I should include it.

- **Breakthrough**: Realizing how it matched my **collaborator persona** made it click. Annotations allow peer-to-peer learning.

Web Share API

- **Issue**: Some browsers don't support it.

- **Breakthrough**: Added a clipboard fallback so no one was left out.

How This Tied Back to My Personas

From Unit 1, I had imagined different types of students who would use this site:

- **The overwhelmed student** → Formspree gave them a quick way to ask questions.

- **The collaborator** → Hypothes.is let them annotate and discuss right on the page.

- **The peer supporter** → Giscus enabled conversations and community.

- **The motivated sharer** → Web Share API gave them an easy way to pass along resources.

What struck me was how **all four personas finally came alive** in Unit 7. Everything I imagined early in the course had a concrete feature by the end.

Lessons Learned

1. **APIs require patience.** Getting IDs and endpoints correct was tedious but crucial.

2. **Privacy matters.** I designed everything opt-in (Hypothes.is, Giscus) so students weren't forced to load third-party scripts.

3. **Accessibility can't be ignored.** Every button had ARIA labels and fallbacks.

4. **Personas are real guides.** They weren't just theory, they actively shaped my design decisions.

5. **Fallibility is normal.** External services can fail. My job is to handle those failures gracefully.

Emotional Takeaways

This unit felt like the moment my site stopped being an **assignment** and started becoming a **real product**. For the first time, I wasn't just coding for marks, I was building something that could live online, accept feedback, host conversations, and be shared with others.

I also felt **proud of my persistence**. At many points, I hit errors and thought I had broken everything. But instead of giving up, I kept debugging, reading, testing, and trying again. When I finally saw “Message sent,” “Annotations enabled,” “Comments loaded,” and “Link copied,” I felt a rush of accomplishment.

Conclusion

Unit 7 transformed my project into a **living hub**.

- Formspree gave students a private way to reach out.
- Hypothes.is turned reading into active engagement.
- Giscus opened real discussions.
- Web Share API extended the site’s reach.

Together, these integrations took my companion site from **a polished course project** to **a platform with real-world functionality**.

By tying everything back to personas, respecting accessibility, and handling external challenges, I feel I’ve grown not just as a web programmer, but as a designer who builds with empathy and foresight.

Unit 7 didn’t just finish my site, it proved that I could take an idea, integrate real services, and make it something people could actually use.

References

- Formspree. (2025). *Formspree Documentation*. https://formspree.io
- Hypothes.is. (2025). *Hypothes.is Developer Docs*. https://h.readthedocs.io/
- Giscus. (2025). *Giscus Setup Guide*. Retrieved from https://giscus.app/
- GitHub Docs. (2025). *About Discussions*.https://docs.github.com/en/discussions
- Mozilla Developer Network (MDN). (2025). *Web Share API*. https://developer.mozilla.org/en-US/docs/Web/API/Navigator/share
- Mozilla Developer Network (MDN). (2025). *Using Fetch*. https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch
- World Wide Web Consortium (W3C). (2018). *Web Content Accessibility Guidelines (WCAG) 2.1*. https://www.w3.org/TR/WCAG21/
- Athabasca University. (2025). *COMP 266 Course Materials*. Athabasca University.