ITECH3208 – Project 1

Individual Journal and Investigative Research Report

Submitted to : Nityansh Rattan

Harmanpreet Singh Bedi 10/13/24 30410228

Table of Contents

Individual Journal

Reflection on SFIA/CBOK Skills Development

Throughout the duration of our project, I have actively engaged in developing my SFIA (Skills Framework for the Information Age) skills, particularly in the areas of **Problem Solving and Team Working**. To enhance my understanding of these skills, I sought resources outside of the university curriculum. (*Problem Management*, n.d.)

One valuable resource was a series of webinars focused on Agile methodologies, specifically on the Scrum framework. These sessions provided practical insights into Scrum principles and practices, illustrating their application in real-world scenarios. I learned how iterative development cycles and regular feedback loops are crucial for optimizing project outcomes. Understanding these principles has significantly informed my approach to problem-solving within the project. (*What Is Scrum?*, n.d.)

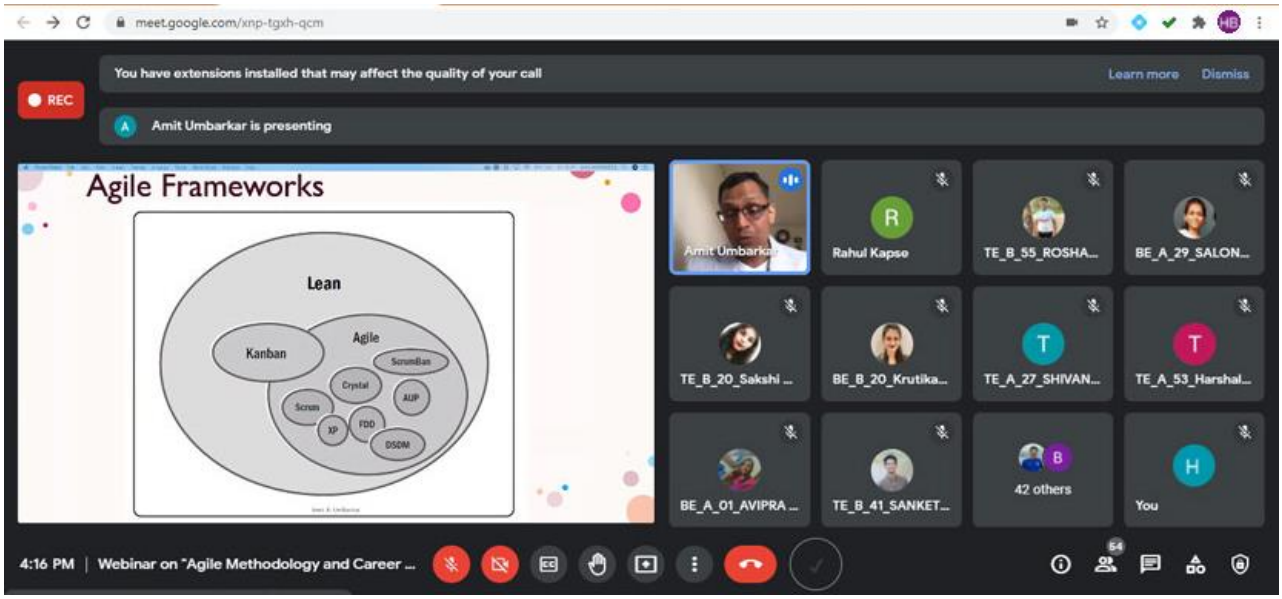

Figure 1. Webinar on Agile Methodologies

Additionally, I explored materials that outline the SFIA Framework. This exploration not only deepened my understanding of the skills required for my role but also illuminated the interconnectedness of various skills within the framework.

For instance, I recognized how developing my leadership capabilities directly impacts my ability to collaborate effectively with team members. Through self-reflection and targeted learning, I was able to identify areas where I could improve my contributions, such as enhancing my communication skills to foster better collaboration within the team.

Individual Status and Contribution

In our project, I primarily served as a Developer and a Team Leader by taking on leadership responsibilities within the team. My contributions were multifaceted, encompassing both technical and non-technical elements.

Technical Contributions

From a technical standpoint, I was responsible for designing several key aspects of our web application, with a significant focus on creating mock-ups and the interactive prototype. This task involved researching best practices for user interface (UI) and user experience (UX) design, ensuring that the layout was intuitive and visually appealing. To enhance usability, I proposed and successfully integrated responsive design principles, ensuring that the application functions smoothly across various devices and screen sizes.

Moreover, I collaborated closely with my team to iterate on the mock-ups based on feedback, making adjustments to improve the overall user flow. My proactive approach in identifying and addressing design inconsistencies was instrumental in maintaining project momentum and aligning the prototype with the project's goals.

Non – Technical Contributions

On the non-technical side, I facilitated team meetings and encouraged open communication among team members. My leadership efforts focused on encouraging an inclusive environment where everyone felt comfortable sharing their ideas and concerns. By implementing structured brainstorming sessions and ensuring that all voices were heard and acknowledged, I was able to enhance team cohesion and collaboration.

However, I encountered challenges in balancing my technical responsibilities with the demands of team coordination. At times, the workload felt overwhelming, and I struggled to keep pace with our project timeline. To address this issue, I adopted a personal task management system using Jira, which helped me prioritize tasks and enhance my productivity. This approach allowed me to maintain focus and contribute effectively to the project while supporting my teammates.

Overall, my contributions were well-documented in our sprint reports, where I provided detailed accounts of my progress, initiatives, evidences and the challenges I faced. This documentation served as a reference point for evaluating my growth throughout the project.

Investigative Research Report

Topic of Research : Designing Web Applications for Accessibility

In my independent research, I explored the essential aspects of designing web applications for accessibility. As we strive to create inclusive digital experiences, understanding and implementing accessibility principles is vital. This investigation not only aligns with ethical standards but also broadens our user base, enhancing the overall user experience. (Vettorino, 2024)

What I Learned

Understanding Accessibility

Web Accessibility refers to the practice of making websites and applications usable by people of all abilities and disabilities. This encompasses a range of considerations, including visual, auditory, physical, and cognitive accessibility. (Wikipedia contributors, 2024c)

Importance of Accessibility in Web Development

1. Legal Compliance : Many countries have laws and regulations (such as the Americans with Disabilities Act in the U.S. and the Web Content Accessibility Guidelines [WCAG] globally) mandating accessibility for digital content. Non-compliance can lead to legal repercussions. (Initiative, n.d.)

2. Enhanced User Experience : Designing with accessibility in mind improves usability for all users, not just those with disabilities. For instance, clear navigation and text alternatives can benefit everyone. (Initiative, n.d.)

3. Broader Audience Reach : By ensuring accessibility, businesses can reach a wider audience, including those with disabilities, thereby increasing potential customer bases. (Initiative, n.d.)

4. Social Responsibility : Committing to accessibility demonstrates a company's dedication to inclusivity and social responsibility, enhancing its brand image and trustworthiness.

Key Considerations for Accessible Web Design

1. Text and Typography (*7 Design Considerations for Accessibility*, n.d.)
    - Contrast Ratios : Ensure sufficient contrast between text and background color to enhance readability. WCAG recommends a minimum contrast ratio of 4.5 : 1 for normal text and 3 : 1 for large text.
    - Font Size and Type : Use legible fonts and allow users to adjust text sizes without losing content or functionality. Avoid overly decorative fonts that can impede readability.
    - Line Spacing and Paragraph Structure : Adequate line spacing (1.5 times the font size) and clear paragraph structure improve readability and make content easier to digest.

2. Semantic HTML (*7 Design Considerations for Accessibility*, n.d.)

    Using semantic HTML elements (like <header>, <nav>, <article>, and <footer>) helps screen readers interpret the structure of web pages correctly. Proper use of HTML elements not only aids in navigation but also provides context to assistive technologies.
    - Headings : Use heading tags (<h1>, <h2>, <h3>, etc.) appropriately to create a logical structure. This hierarchy helps users navigate content efficiently.
    - Lists : Utilize ordered and unordered lists to present related information clearly, which assists screen reader users in understanding content organization.

3. Keyboard Navigation

    Ensure that all interactive elements are navigable and operable using keyboard shortcuts. Many users with disabilities rely on keyboard navigation instead of a mouse. Implementing the following practices can enhance keyboard accessibility :
    - Tab Indexing : Use the tab index attribute to control the order of keyboard navigation. Ensure that all interactive elements are reachable via the Tab key.
    - Visible Focus Indicators : Provide clear visual cues for focused elements to guide keyboard users. CSS can be used to customize focus styles.

4. Alternative Text for Images

    All non-text content, such as images and videos, should include alternative text (alt text) that describes the content and function of the media. This enables users with visual impairments to understand the context of images through screen readers.

- Descriptive Alt Text : Alt text should be concise yet descriptive enough to convey the meaning of the image. For decorative images, use an empty alt attribute (alt="") to avoid unnecessary narration. (*30 Web Accessibility Tips | AccessComputing*, n.d.)

5. Form Accessibility (*7 Design Considerations for Accessibility*, n.d.)

   Forms should be designed with accessibility in mind, ensuring that all elements are labelled correctly. Here are some best practices :

   - Label Elements : Use <label> tags associated with form controls to enhance usability for screen reader users. Ensure that labels are descriptive and clearly indicate the required input.
   - Error Handling : Provide clear error messages and suggestions for correcting errors, ensuring that users understand how to fix issues. For example, specify which field has an error and provide guidance on the correct format.
   - Keyboard – Friendly Navigation : Ensure all form elements are accessible via keyboard and that users can navigate between fields efficiently.

Testing for Accessibility

1. Automated Tools

   Various tools can help assess web accessibility. Some of the examples include :

   - WAVE : A web accessibility evaluation tool that identifies accessibility and WCAG issues. It provides visual feedback on the accessibility of web content.
   - Axe : A browser extension that performs accessibility checks and provides recommendations for improvements. It integrates seamlessly into the development workflow. (Singh, 2024)
   - Lighthouse : A Google tool that audits web pages for performance, accessibility, and best practices. It provides insights into accessibility issues and suggestions for remediation.

2. Manual Testing

   Automated tools can miss certain issues, so manual testing is also crucial. Engage users with disabilities to test your application, gathering feedback on usability and identifying areas for improvement. (Singh, 2024)

   - User Testing : Conduct usability testing sessions with users who have disabilities to gather qualitative feedback on their experiences. (Singh, 2024)

- Checklists : Use accessibility checklists based on WCAG guidelines to systematically review and identify potential issues. (Singh, 2024)

Best Practices for Accessibility

1. Adapt WCAG Guidelines : Familiarize the team with WCAG 2.1 guidelines and implement them throughout the design and development process.

2. Conduct Regular Accessibility Audits : Integrate accessibility audits into our development cycle. Schedule audits at various stages to identify and address accessibility issues early. (*30 Web Accessibility Tips* | *AccessComputing*, n.d.)

3. Provide Accessibility Training : Conduct workshops or training sessions for team members to raise awareness about accessibility principles and best practices.

4. Incorporate User Feedback : Engage users with disabilities in the testing phase. Their feedback is very crucial for understanding real-world challenges and improving usability. (*30 Web Accessibility Tips* | *AccessComputing*, n.d.)

5. Maintain Documentation : Keep detailed documentation of accessibility guidelines, testing results, and user feedback. This will serve as a valuable resource for future projects.

6. Stay Updated : Accessibility standards and technologies are continually evolving. Stay informed about new developments and best practices to maintain an inclusive web environment.

Challenges Faced

1. Lack of Awareness
   - One of the primary challenges in implementing accessibility features is the lack of awareness among team members regarding its importance and best practices. (Cpwa, 2024)
   - Solution : Address this by organizing training sessions to raise awareness about the significance of accessibility and how to incorporate it into the development process.

2. Time Constraints
    - Often, teams may perceive accessibility work as time-consuming and may keep it on a lower priority than other features.
    - Solution : Emphasize that addressing accessibility early in the project lifecycle can save time and resources in the long run, preventing the need for significant rewrites later.

3. Limited Resources
    - Smaller teams may lack the necessary resources or tools to conduct thorough accessibility testing. (Cpwa, 2024)
    - Solution : Utilize free or low-cost accessibility tools and develop a culture of collaboration where team members share resources and knowledge.

Recommendations to the Project Team

Based on my research, I propose the following comprehensive recommendations to ensure our web application is accessible. These recommendations are justified by the findings from my investigation into web accessibility principles and best practices.

1. Adopt WCAG Guidelines
    - Justification :
      Familiarizing the team with the Web Content Accessibility Guidelines (WCAG) 2.1 is crucial for establishing a common understanding of accessibility standards. By aligning our development process with these guidelines, we can ensure our application meets both legal compliance and user needs. (*Accessibility for Web Developers*, 2024)
    - Action Steps :
      → Organize training sessions focused on WCAG principles for all team members.
      → Develop a reference document summarizing key WCAG criteria relevant to our project.

2. Conduct Regular Accessibility Audits
    - Justification :
      Integrating accessibility audits at various stages of development will allow us to identify issues early, reducing the need for significant revisions later. Regular assessments will help maintain compliance and improve the user experience. (*10 Ways to Improve Web Accessibility*, n.d.)

- Action Steps:
  → Schedule audits at key milestones (e.g., after design completion and prior to launch, etc.).
  → Utilize both automated tools (e.g., WAVE, axe, Lighthouse) and manual testing methods to cover a broad range of accessibility issues.

3. Provide Accessibility Training
   - Justification :
     Ensuring that all team members understand accessibility principles is vital for developing a culture of inclusivity. Training can significantly enhance the team's ability to create accessible designs and identify potential barriers during development. (*Accessibility for Web Developers*, 2024)
   - Action Steps:
     → Conduct workshops led by accessibility experts or use online resources to provide ongoing education.
     → Encourage team members to become certified in accessibility standards (e.g., CPACC - Certified Professional in Accessibility Core Competencies).

4. Incorporate User Feedback
   - Justification :
     Engaging users with disabilities during the testing phase is essential for identifying real-world accessibility challenges. Their insights will provide crucial context that automated tools may overlook. (*10 Ways to Improve Web Accessibility*, n.d.)
   - Action Steps:
     → Set up user testing sessions with diverse participants who have varying disabilities.
     → Collect qualitative feedback through surveys and interviews to gather insights into the user experience.

5. Implement a Design System with Accessibility in Mind
   - Justification :
     A design system that incorporates accessibility best practices will streamline the development process and ensure consistency across the application. By establishing a common set of accessible design patterns, the team can improve efficiency and maintain quality. (*Accessibility for Web Developers*, 2024)

- Action Steps:
  → Develop a design system that includes accessible UI components and guidelines.
  → Create templates and patterns that emphasize semantic HTML, proper labelling, and keyboard navigation.

6. Stay Updated on Accessibility Trends and Technologies
   - Justification :
     The field of web accessibility is continually evolving, with new technologies and best practices emerging. Staying informed about the latest developments will help us adapt our approach and ensure ongoing compliance. (*10 Ways to Improve Web Accessibility*, n.d.)
   - Action Steps:
     → Subscribe to accessibility – focused newsletters and blogs to receive updates.
     → Participate in online forums and communities dedicated to accessibility discussions.

7. Maintain Documentation of Accessibility Efforts
   - Justification :
     Keeping detailed documentation of our accessibility guidelines, testing results, and user feedback will serve as a valuable resource for future projects. This transparency enhances accountability and helps ensure ongoing compliance.
   - Action Steps :
     → Create a centralized repository for accessibility – related documents, including audit reports and user feedback summaries.
     → Regularly review and update documentation based on new findings and feedback from team members and users.

8. Consider Limitations and Challenges
   While these recommendations provide a robust framework for ensuring accessibility, it's important to acknowledge potential limitations : (*Accessibility for Web Developers*, 2024)
   - Resource Constraints : Smaller teams may face challenges in allocating time and budget for extensive accessibility training and audits. To mitigate this, prioritize essential training and utilize free resources wherever possible.
   - Balancing Functionality and Accessibility : In some cases, achieving certain functionalities may conflict with accessibility best practices. Continuous dialogue

among team members about prioritizing accessibility during the design process can help navigate these challenges.

Conclusion

1. Importance of Accessibility :
    - Designing with accessibility ensures users of all abilities can engage with digital platforms effectively. It's both an ethical responsibility and a legal requirement, adhering to global standards like WCAG. (Initiative, n.d.)

2. Broader Audience Reach :
    - Prioritizing accessibility expands our audience, including those with physical, sensory, and cognitive impairments. Accessible design also improves usability for all users.

3. User Satisfaction and Engagement :
    - Accessibility leads to more intuitive interfaces, enhancing user satisfaction and engagement. Easy navigation encourages return visits, boosting retention and loyalty. (Wikipedia contributors, 2024c)

4. Compliance with Ethical and Legal Standards :
    - Following accessibility guidelines helps meet legal requirements, avoiding legal risks. It also reflects the organization's commitment to inclusivity and social responsibility.

5. Ongoing Project Benefits :
    - Early adoption of accessibility reduces costly revisions later. It also develops and enhances continuous collaboration and feedback within the team.

6. Future Growth Opportunities :
    - Mastering accessibility prepares the team for future projects, enabling the delivery of compliant, high-quality products and encouraging innovation in future designs. (Wikipedia contributors, 2024c)

References

- Vettorino, M. Z. (2024, April 16). *Web Accessibility: The Ultimate Guide*. https://blog.hubspot.com/website/web-accessibility
- Wikipedia contributors. (2024c, September 25). *Web accessibility*. Wikipedia. https://en.wikipedia.org/wiki/Web_accessibility
- *7 Design considerations for accessibility*. (n.d.). Evolving Web. https://evolvingweb.com/blog/7-design-considerations-accessibility
- Singh, A. P. (2024, May 7). *What is Accessibility Testing, Its types, Test Cases & More | LambdaTest*. https://www.lambdatest.com/learning-hub/accessibility-testing
- *30 Web accessibility Tips | AccessComputing*. (n.d.). https://www.washington.edu/accesscomputing/30-web-accessibility-tips
- Cpwa, P. S. (2024, January 26). Top 7 Web Accessibility challenges for Designers and Developers. *Medium*. https://medium.com/@askParamSingh/top-7-web-accessibility-challenges-for-designers-and-developers-2b58a4d02b19
- *10 ways to improve web accessibility*. (n.d.). https://www2.stardust-testing.com/en/10-tips-for-web-accessibility
- *Accessibility for web developers*. (2024, February 20). web.dev. https://web.dev/articles/a11y-tips-for-web-dev
- *Problem management*. (n.d.). SFIA. https://sfia-online.org/en/sfia-8/skills/problem-management
- *What is Scrum?* (n.d.). Scrum.org. https://www.scrum.org/resources/what-scrum-module