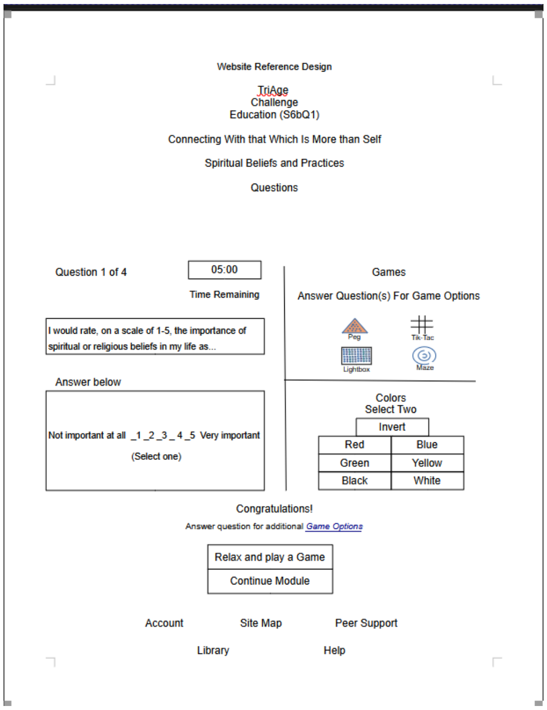
Howard Mahaffey W13D4 Homework The Human Side of Tech Overview

The first time I encountered JavaScript, it was during a pivotal moment in my transition from hardware networking to software development. As a USAF nine year veteran with a background in secure communications and 15 years of experience as an entrepreneur in Silicon Valley, I was no stranger to technical challenges. However, the world of coding felt daunting but exciting. I remember working on a personal community project named TriAge in early 2022, where the goal was to build an interactive webpage to facilitate volunteer sign-ups. Despite my struggles with debugging, I found myself captivated by the potential of JavaScript to bring static designs to life. I started the project in PUG. However, the experience expanded my passion for software development, leading me to explore JavaScript's many capabilities.

Reflecting on this journey, I often think back to basic layout I made using open source in a restrictive environment and brainstorming solutions for that project.



Website Reference Design

TriAge
Challenge
Education (S6bQ1)

Connecting With that Which Is More than Self

Spiritual Beliefs and Practices

Questions

Question 1 of 4

05:00

Time Remaining

I would rate, on a scale of 1-5, the importance of spiritual or religious beliefs in my life as...

Answer below

Not important at all _1 _2 _3 _ 4 _5 Very important

(Select one)

Games

Answer Question(s) For Game Options

Colors
Select Two

Invert

Red	Blue
Green	Yellow
Black	White

Congratulations!

Answer question for additional *Game Options*

These rough, LibreOffice ideas became the blueprint for several features I eventually implemented in various code bases. In this essay, I will detail my personal journey with JavaScript, its impact on goals for my community, the technical insights I've gained, and the innovative solutions I've envisioned based on my experiences.

My journey with JavaScript… it began with an equal measure of frustration and fascination. Early on, I struggled to understand coding logic in general and learned eventually through daily coding challenges. This laid the groundwork for understanding synchronous and asynchronous coding paradigms, including JavaScript. Concepts like callbacks and promises seemed abstract, and debugging felt like an insurmountable challenge. This was all part of a specific turning point occurred in mid-2021 while I was in a transitional phase of my life, including a period spent incarcerated. During my time in a correctional facility, I was part of a tech education program that introduced coding as a rehabilitative tool. My first interaction with JavaScript took place in a small classroom where I wrote my first lines of code. This foundational experience helped me realize the transformative potential of technology, even in restrictive environments.

Later that year, while serving as a teacher's aide I first wrote the application in PUG for content but applied JavaScript to other areas of the project like validating user input online, and grade tracking for the education modules. I spent hours troubleshooting why the validation logic failed to execute as expected. Eventually, through trial, error, MDN, and guidance from classmates and the instructor, I successfully implemented a feature that provided pre-loading a JSON vendor database that was crafted manually. This breakthrough boosted my confidence and underscored the practical value of JavaScript in solving future real-world problems.

As my understanding deepened, I started applying JavaScript to other projects. One memorable example involved developing an interactive dashboard for tracking donation metrics. The project required creating dynamic charts that were updated in real time. By leveraging JavaScript libraries like Chart.js, I created a visually appealing and functional tool that helped the organization make data-driven decisions. These experiences taught me the importance of persistence and adaptability in mastering new skills.

Learning JavaScript has also played a significant role in my efforts to give back to the local community. In 2022, when I became a teacher's aide, which allowed me to learn through the learning experiences of others. My daily routine often included using JavaScript for daily exercises with the cohorts that would follow. As someone passionate about teaching and mentorship, I've used my growing knowledge to help others. For instance, I would help others with their capstone projects, while introducing them to multi-factor authentication programming concepts using JavaScript. One student, initially intimidated by the MVC model, created a site with games from the curriculum. Seeing their sense of accomplishment reinforced my belief in the transformative power of technology.

However, I've also observed disparities in access to technology. I created a program in python and wrote pseudo code with py extensions and had it all deleted because of classroom protocols that would seemingly change over time. In underprivileged situations, limited resources often prevent individuals from exploring programming opportunities. During the project, I was eager to learn but lacked access to many internet resources. This experience motivated me to advocate for programs that bridge the digital divide, ensuring that second chance opportunities exist to harness technology's potential.

One of my most challenging yet rewarding experiences involved trying to convert PUG templates into JavaScript-based templates. While working on the DW_Pug_A project (https://github.com/easymahaffey/DW_Pug_A), I encountered the elegance of PUG's concise syntax and its ability to streamline HTML generation. However, adapting this approach to a JavaScript framework required a deeper understanding of modular design principles.

For example, translating nested PUG loops into JavaScript posed a unique challenge. In one instance, a data-driven list rendered seamlessly in PUG but resulted in broken layouts when implemented in JavaScript. This is a project that still needs to be converted. This experience highlighted the importance of bridging conceptual gaps between different technologies.

Another technical milestone involved managing asynchronous operations. While developing a prototype for real-time data visualization, I encountered race conditions that disrupted functionality. Initially, I struggled to identify the root cause of intermittent failures. By leveraging JavaScript's async/await syntax and understanding event loops, I resolved these issues and improved the application's reliability. This deep dive into asynchronous programming not only enhanced my technical skills but also prepared me to tackle more complex challenges in the future.

My training in JavaScript and Full Stack development provided invaluable insights into creating robust applications. For example, I integrated a Node.js backend with a React frontend to build a small-scale multi-factor authentication system. This project required seamless communication between the client and server, which I achieved using RESTful APIs. Debugging issues like CORS errors and optimizing data flow were key learning moments that solidified my understanding of full-stack principles.

One recurring challenge I've faced is the learning curve when using AJAX to create a test bed for an employment coding challenge. To address this, I developed a UI that would render outputs from the coding challenge for beginners. This tool would feature a dual-pane interface, allowing users to input JavaScript code on one side and view the corresponding JavaScript output on the other. Additionally, it could provide real-time error feedback. This would not only accelerate learning but also empower users to experiment with confidence.

For instance, during one project, I spent hours manually translating the input from JSON's indentation-based structure into the proper AJAX async calls. Having a tool that automated this process while offered and IDE that helped save significant time and frustration.

Drawing from my background in secure communications, I've identified a need for simplified authentication solutions for front-end developers. JavaScript's existing libraries, such as Axios and Redux, provides a simple robust functionality but often require a steep learning curve. I created a lightweight module focused on ease of use, with pre-configured settings for common use cases like securing user challenge requests.

One potential application of this library could be in the development of web sites that handle sensitive data. For example, I prototyped an asynchronous event-based authentication form with built-in hashing functions to generate Luhn based tokens during the authentication process. By abstracting a simple cryptographic operation into reusable modules, this innovation could make secure coding practices more accessible to developers of all skill levels.

Imagine a scenario where a small non-profit organization needs to build a secure and user-friendly platform for tracking donations. With limited technical expertise, they struggle to implement encryption protocols effectively authenticate users. My proposed modular application would simplify this process, allowing them to focus on their mission without compromising security. Similarly, the built-in security question module would allow the client administer their own password resets in a secure fashion with the need to system administrator intervention or assistance, enabling users to work more efficiently.

Controller Luhn function example:

```
createTwoFactor: function (length) {
    let code = []
    while (code.length < length) {
      code.push(Math.ceil(Math.random() * 9))
    }
    let checkDigit = code.map((dig, i) => i % 2 == 0 ? dig : dig * 2)
      .map(num => num.toString())
      .map(num => num.length > 1 ? num.split('') : num)
      .map(num => num.length > 1 ? num.reduce((a, b) => +a + +b, 0) : +num)
      .reduce((a, b) => a + b, 0) * 9 % 10
    code.push(checkDigit)
    let twoFactorToken = code.join('')
    return twoFactorToken
  }
```

Reflecting on my journey, I am struck by the transformative impact of JavaScript on both my personal growth and my ability to contribute meaningfully to my community. From overcoming technical hurdles to empowering others through mentorship, JavaScript has been a catalyst for change. It has taught me the value of persistence, collaboration, and creative problem-solving.

Looking ahead, I am committed to leveraging my skills to drive innovation and inclusivity. In a period where DEI is being rolled back, this is important. Whether by refining my prototypes, mentoring aspiring developers, or advocating for equitable access to technology, I aim to leave a lasting impact. My ultimate goal is to bridge the gap between technology and humanity, ensuring that the tools we build serve not only functional purposes but also foster connection and opportunity. “It’s incredible what you can do with a few lines of code,” a friend once said, and those words resonate deeply with me as I continue to explore the boundless possibilities of JavaScript. By continuing to learn, innovate, and share knowledge, I hope to inspire others to explore the boundless possibilities of coding.