**Literature Review Draft: Website Archaeology Tool**

**Introduction**

The Website Archaeology Tool aims to enhance web development education for novices by visualising static websites’ HTML, CSS, and JavaScript as a 3D “excavation” metaphor, promoting STEM engagement. By parsing websites from public repositories (e.g., GitHub, MIT-licensed) and presenting their structure in a 3D view with a 2D timeline, the tool makes abstract concepts like tag hierarchies tangible. This review examines web development education and visualisation tools, identifies gaps, and outlines a methodology to address them, connecting to the project’s research questions.

**Web Development Education**

Web development skills, including HTML, CSS, and JavaScript, are critical for digital literacy and STEM education (W3Schools, 2025). W3Schools provides beginner-friendly tutorials with practical examples, but its linear, text-based format struggles to convey complex relationships like CSS inheritance or DOM structure (W3Schools, 2025). CodePen offers a live coding environment for testing HTML, CSS, and JavaScript, enabling real-time previews (CodePen, 2025). However, its focus on advanced users limits accessibility for novices who need guided, visual learning to grasp web development fundamentals.

**Visualisation Tools**

Visualisation tools enhance coding education by representing code graphically. BeautifulSoup, a Python library, efficiently parses HTML and extracts elements, ideal for analysing website structures (BeautifulSoup, 2025). However, its text-based output lacks educational visualisations for beginners. Three.js, a JavaScript library, supports 3D web visualisations, used in interactive applications like simulations (Three.js, 2025). Yet, it requires significant coding expertise and lacks educational focus. Scratch engages young learners with block-based visual programming, fostering creativity (Resnick et al., 2009). While effective for general programming, it does not address web-specific concepts like HTML/CSS interactions.

**Research Gap**

Current tools lack 3D, metaphor-driven approaches to teach web development to novices, particularly in visualising HTML, CSS, and JavaScript as cohesive layers. W3Schools and CodePen focus on code execution, not structural visualisation, while BeautifulSoup and Three.js are technical tools without educational scaffolds. Scratch’s visual approach inspires engagement but misses web development specifics. The Website Archaeology Tool addresses this gap by parsing static GitHub-hosted websites and presenting them in an archaeological 3D interface, making abstract concepts intuitive.

**Research Questions**

This project explores:

1. How does 3D visualisation enhance novices’ understanding of web development structures?
2. Can an archaeological metaphor increase STEM engagement among learners?
3. Is static website parsing sufficient for creating effective 3D educational visualisations?

**Methodology**

The project employs a design-based research approach to develop and evaluate the Website Archaeology Tool. Three static websites from public GitHub repositories (MIT-licensed) will be selected, each with 2-3 modified versions to simulate development history. BeautifulSoup will parse HTML, CSS, and JavaScript, extracting tags, styles, and functions (BeautifulSoup, 2025). These elements will be visualised using Three.js in a 3D “excavation” view, with color-coded HTML tags (e.g., <div> in blue), CSS samples, and JavaScript functions, alongside a 2D timeline showing version changes (Three.js, 2025). No human participants are involved, as the tool uses public datasets, avoiding ethical concerns. Validation will involve comparing the tool’s output against manual code analysis to ensure accuracy, with iterative refinements based on pilot testing with sample websites. This methodology addresses the research questions by testing 3D visualisation’s impact on learning (RQ1), the archaeological metaphor’s engagement (RQ2), and static parsing’s sufficiency (RQ3).

**References**

* BeautifulSoup (2025) *BeautifulSoup Documentation*. Available at: <https://www.crummy.com/software/BeautifulSoup/> (Accessed: 24 June 2025).
* CodePen (2025) *About CodePen*. Available at: <https://codepen.io/about/> (Accessed: 24 June 2025).
* Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B. and Kafai, Y. (2009) ‘Scratch: Programming for All’, *Communications of the ACM*, 52(11), pp. 60–67. doi:10.1145/1592761.1592779.
* Three.js (2025) *Three.js Documentation*. Available at: <https://threejs.org/docs/> (Accessed: 24 June 2025).
* W3Schools (2025) *HTML Tutorial*. Available at: <https://www.w3schools.com/html/> (Accessed: 24 June 2025).