Web Block Craft: web development for children using Google Blockly

Madhumini Gunaratne, Senal Weerasekara, Dehemi Weerakkody, Nisal Sashmitha, Rivoni De Zoysa, Nuwan Kodagoda
Department of Computer Science and Software Engineering, Faculty of Computing, Sri Lanka Institute of Information Technology, Malabe, Sri Lanka

Article Info

Article history:

Received Apr 8, 2024
Revised Jul 3, 2024
Accepted Jul 7, 2024

Keywords:

Backend project
Educational
Frontend project
Learning tool
Web programming

ABSTRACT

Web Block Craft is an innovative educational application that uses the Google Blockly framework to teach web development to children aged eleven and above. The application serves as a comprehensive learning tool, allowing users to explore both frontend project and backend project development. The frontend project includes HTML, CSS, JavaScript, and DOM manipulation, while the backend project covers server building, web app security, application programming interfaces (APIs), and database management. Web Block Craft's unique block-based interface allows users to easily drag and drop components into a dynamic working environment, resulting in an engaging experience with live output display and simultaneous code presentation. A unique feature of Web Block Craft is the integration of a platform within the application, which allows teachers to create lessons with step-by-step instructions for students. This new feature allows for a more structured learning experience, which improves understanding of web development concepts. To enhance the learning experience, the application provides extensive documentation, serving as a valuable resource for users to grasp the intricacies of web programming. By combining the power of Google Blockly with a creative user interface and educational resources, Web Block Craft provides a comprehensive learning environment that empowers creative web programming with confidence.

Corresponding Author:

Madhumini Gunaratne
Department of Computer Science and Software Engineering, Faculty of Computing, Sri Lanka Institute of Information Technology
New Kandy Road, Malabe, Sri Lanka
Email: madhumini.g@sliit.lk

1. INTRODUCTION

Web Block Craft is a revolutionary educational tool designed to meet the increasing demand for coding skills in today's digital economy. It provides an accessible and enjoyable learning experience for web programming, especially targeting children aged eleven and above and beginners. Leveraging the Google Blockly framework [1], Web Block Craft offers a visually driven platform with custom blocks tailored to web programming concepts. In today's digital economy, there is a growing demand for coding skills, especially among children and beginners. Traditional code editors can be complex and intimidating, making it challenging for young learners to grasp web programming concepts. Web Block Craft is strategically designed to simplify coding complexities and make it accessible to all, leveraging Google Blockly's flexibility and open-source nature. It addresses the challenges faced by children and beginners in traditional code editors by transforming the learning experience into something captivating and enjoyable. The aim is to foster a lasting passion for coding by making it accessible and genuinely enjoyable, resonating with curiosity and creativity.

The following paragraphs will explore relevant literature, challenges, and innovations in web programming education and emphasize the need for effective and accessible learning environments. Wang explored the integration of web technologies in computer science courses [2], highlighting the challenges faced by users when adopting new technologies and programming environments. Totan and Korucu [3] demonstrated positive outcomes in middle school students' computational thinking skills using Blockly-based coding education. The findings of the study showed great impact of block-based coding education and how it contributed to students' computational thinking skills and attitudes. Park and Wiedenbeck [4] investigated the challenges faced by students learning web development and how difficult it is to seek help while learning programming. Particularly regarding the challenges in learning HTML, CSS, and JavaScript at young ages. Hwang *et al.* [5] emphasized the need for tools like online coding and debugging support in web-based programming education to address the challenges faced by traditional learning approaches. Nam and Smith-Jackson [6] focus on user interface design and improving the quality of web-based learning and discuss the importance of user-centered design, and how it ensures ease of use. Hadjerrouit [7], Chao [8], and Takacs [9] explain how visual programming environments have evolved with time and the impact of focusing on graphical elements can help problem-solving immensely. While Viktor’s product BlockImpress simplified complex coding concepts, Hjorth [10] explores the strengths and weaknesses of teaching visual programming languages to novices, and benefits within the field of programming.

Kaushal *et al.* [11] focus on the importance of understanding basic concepts for responsive websites and how to bridge the gap between complex web development concepts and beginner-friendly learning experiences. Aggarwal *et al.* [12] introduced a block-programming editor for HTML code, and Queirós [13] introduced LearnJS highlighting a dual-mode approach that allows students to effortlessly switch between block and text-based coding and emphasizing the importance of user-friendly interface and interactive tutorials. Ashrov *et al.* [14] combined Google Blockly with single-threaded JavaScript for reactive system development. Their work contributes to the evolution of technology and design patterns in interactive applications. Xu *et al.* [15] integrated character animation to introduce the application and guide children through the process of learning programming. This study showed significant improvement in learning when the applications are interactive. Marron *et al.* [16] introduced a toolset for creating interactive user interfaces, particularly emphasizing the frontend applications combined with Google Blockly. Miura [17] introduces Block Sweetie, a PHP-based framework with a block-programming editor to aid novice learners in web application development. Both Marron *et al.* [16] and Miura [17] discussed the impact on visual and intuitive environments that minimize syntax errors through block-based programming. Huang [18] and Adam *et al.* [19] explored the impact of backend web development. Huang [18] explored the importance of node.js for backend programming while Adam *et al.* [19] focused on developing a backend system using REST API for better performance and security. Valsamakis *et al.* [20] discussed above collaborative visual programming, and how to incorporate collaborative features. These collaborative tools will aim to promote interactive and engaging educational experiences among groups of students.

Despite the advancements in educational tools, there remains a need for a comprehensive, engaging, and beginner-friendly platform that simplifies web programming concepts for children and beginners. Existing tools often lack interactive features, structured learning environments, simplicity, and the ability to bridge frontend and backend development seamlessly. Web Block Craft addresses these gaps by providing a visually driven, block-based platform using Google Blockly, tailored specifically for web programming. It integrates custom blocks for HTML, CSS, JavaScript, NodeJS, MongoDB, server creation, database handling, security, and application programming interfaces (APIs). When it comes to backend development, there are considerable challenges compared to learning frontend, such as environment setup, working with the command line, and using additional tools for testing. Web Block Craft hides all these complexities and provides users with a visually appealing platform, making backend development accessible and entertaining. This platform transforms the learning experience into an engaging and enjoyable journey, fostering curiosity and creativity while ensuring a smooth transition from drag-and-drop to code implementation. Web Block Craft also offers interactive debugging tools, live code previews, step-by-step lessons, and extensive documentation, making web development accessible and enjoyable for young learners. Web Block Craft not only empowers students to explore and apply their learning in real-time scenarios but also equips educators with a powerful tool to customize lessons, track progress, and provide targeted support. By fostering a dynamic and interactive learning environment, Web Block Craft introduces a new paradigm in web programming education, setting a benchmark for effective, engaging, and accessible learning experiences.

The next sections will detail the development and implementation of Web Block Craft, demonstrating its relevance and impact. The platform's features and functionalities will be explained, highlighting how they address the identified gaps in existing educational tools. Surveys and user feedback will be presented to highlight the effectiveness of Web Block Craft in enhancing web development skills and fostering a passion for web development among children and beginners.

2. PROPOSED METHOD

In order to develop Web Block Craft extensive user research was conducted to delve into users' preferences, learning styles, and expectations. The study included a diverse sample size of seventy-six individuals, encompassing a range of demographics such as age, educational background, and prior experience in web programming. Data collection was conducted through a combination of surveys, interviews, and usability testing sessions, allowing for a comprehensive gathering of user insights. The survey questions were strategically designed to elicit specific feedback on user preferences, ease of use, feature prioritization, and areas for improvement within the application. Table 1 succinctly outlines the percentage distribution of user preferences based on a few questions from the survey.

Table 1. Selected user preferences from the survey

Question	Agree	Neutral	Disagree
Is learning with colorful blocks more engaging than traditional materials like books?	67.1%	4.0%	28.9%
Are you aware that you can create appealing websites using blocks?	26.3%	14.3%	59.4%
Do you like to create your own website using blocks?	63.1%	32.9%	4.0%
Would an interactive, game-like learning platform enhance your enjoyment of the learning process?	75.1%	7.8%	17.1%
Was learning web development challenging with only code editors and documentation?	56.5%	30.8%	12.7%
Are you familiar with Blockly or similar visual programming languages for learning programming?	17.9%	23.1%	59.0%

According to the survey results, participants pointed out a strong preference for interactive and visually pleasing learning techniques, with 67.1% preferring learning with colored blocks over traditional materials such as documents. 75.1% of participants stated a preference for an interactive, game-like learning platform. This is in accordance with the motivation for developing Web Block Craft, which addresses the need for a platform that is interactive and enjoyable. The research highlights the potential of Web Block Craft to address the issues mentioned by 56.5% of respondents, who found learning web development difficult with only code editors and documentation. This survey data indicates a pressing need for a platform that not only facilitates learning but also makes it enjoyable and engaging. The strong consensus among participants emphasizes the importance of overcoming the obstacles associated with traditional coding methodologies. Based on the survey results and the identified learning challenges, visual programming emerged as the optimal approach for teaching web programming and creating tailored lessons to support beginners. This approach not only addresses the specific needs and preferences of learners but also enhances engagement, comprehension, and retention of web development concepts, making the learning process more enjoyable and effective.

In addition to drawing insights from the user survey, Web Block Craft also takes inspiration from a variety of educational tools. Scratch [21] is a renowned block-based educational programming tool that offers a broader range of coding projects to enhance fundamental programming concepts. Similarly, Game Lab [22] is a block-based programming environment tailored for app and game creation promoting storytelling methods. Tynker [23] is a block-based coding platform primarily aimed at children. It offers structured activities and courses in various programming topics, including game design, animation, and app development. Kodu [24], is a visual programming language specifically designed for creating games for kids using a block-based interface to program game behaviors. PictoBlox [25] visual programming platforms aimed at teaching programming concepts, machine learning concepts, and many more focusing on fostering creativity and problem-solving skills through all areas of computer science. The ideas and features from these tools and related research studies formed the backbone of Web Block Craft's innovative approach to teaching web development. By synthesizing the best practices and methodologies from these studies, Web Block Craft emerges as a comprehensive and engaging educational application, to learn web programming.

Drawing from the insights gathered from previous studies, the development of the block creation process commenced. Traditional methods of teaching web development typically involve courses, documentation, and online videos, which often require considerable time and effort to comprehend. This approach can be overwhelming for beginners, potentially leading to a dislike for the field. The primary challenge lay in devising blocks that could simplify the complexities of web programming concepts while still providing comprehensible information. In creating the frontend teaching application, a key challenge was integrating HTML, CSS, and JavaScript elements into blocks in a meaningful way. HTML and CSS blocks needed to generate visual outputs, while JavaScript blocks had to add dynamic functionality. Simultaneously highlighting the output alongside the blocks was essential to demonstrate the correlation between the blocks and their effects. For the backend teaching aspect, hiding complexities was paramount.

Concepts such as server creation with API endpoints, authentication, authorization, and database handling can be daunting for beginners. The conventional method of trial and error, relying heavily on syntax, often consumes significant time and effort. Moreover, setting up backend development environments with command-line interfaces can be discouraging. Balancing the need to simplify without compromising essential information was critical. While considering these information blocks were meticulously designed to provide clear instructions, with inputs and outputs crafted to create a logical pattern for children to follow. The use of vibrant colors and visually distinct shapes further enhanced the attractiveness of the blocks, making the learning experience engaging and accessible to young learners. Each aspect of block design was carefully considered to optimize learning outcomes and foster a positive learning environment for web programming education. In conclusion, the thoughtful design and implementation of Web Block Craft's block creation process not only address the user preferences gathered and the challenges faced by beginners in learning web development but also create an engaging and accessible learning environment, paving the way for a positive and enriching experience in web programming education.

3. METHOD

The overview diagram in Figure 1 illustrates the architecture of the web application. Serving as the user's entry point, the frontend enables seamless interaction with the application. Web Block Craft, built upon Google Blockly's playground, revolutionizes traditional coding methods. A standout feature of Web Block Craft is its integrated platform, empowering users to effortlessly create step-by-step lessons. This functionality proves invaluable for educators, who can tailor lessons to aid students with assignments, fostering dynamic learning experiences in web development. In the backend, development logic and data processing take place, with docker containers managed by the docker daemon, ensuring smooth operation by isolating different components. Within these containers, the Node.js sandbox executes the application's backend code. Users can create both frontend and backend projects within the application, leveraging its intuitive interface and robust features.

To enhance usability, a multifaceted approach was employed, integrating various methods and features into the design of Web Block Craft. Live code outputs were implemented to provide real-time feedback, allowing users to see the immediate effects of their programming actions, and fostering a more dynamic and responsive learning environment. Comprehensive documentation was made available to provide detailed explanations, instructions, and examples, supporting users in understanding and utilizing the platform effectively. YouTube series was created to complement the documentation, offering visual tutorials and demonstrations to further aid users in mastering web programming concepts using Web Block Craft. Tour guides were implemented to provide users with a seamless navigation experience. Animated characters were strategically incorporated to add an element of enjoyment and engagement, making the learning process more interactive and appealing for kids. Proper error handling mechanisms were put in place to restrict block connections to only compatible blocks, minimizing confusion and guiding users towards correct usage. Furthermore, an easy block search feature was integrated to allow users to quickly locate and access specific blocks. By combining these methods and features, usability was significantly improved, making the platform accessible, enjoyable, and conducive to effective learning and experimentation in web development.

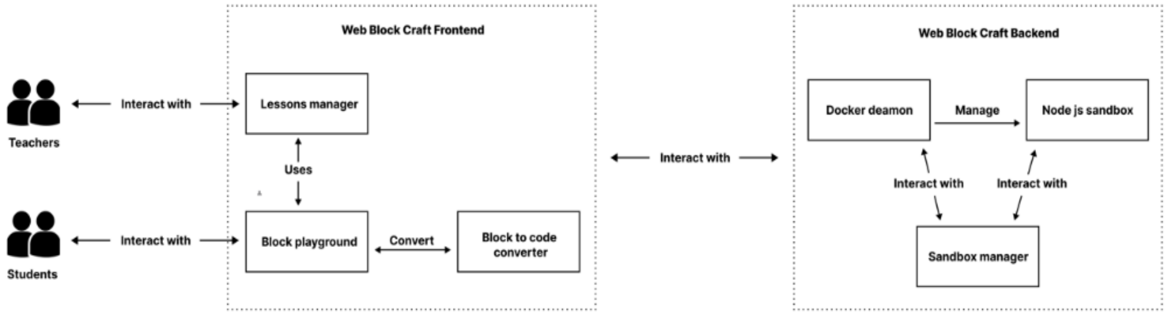


Figure 1. High-level overview diagram of Web Block Craft

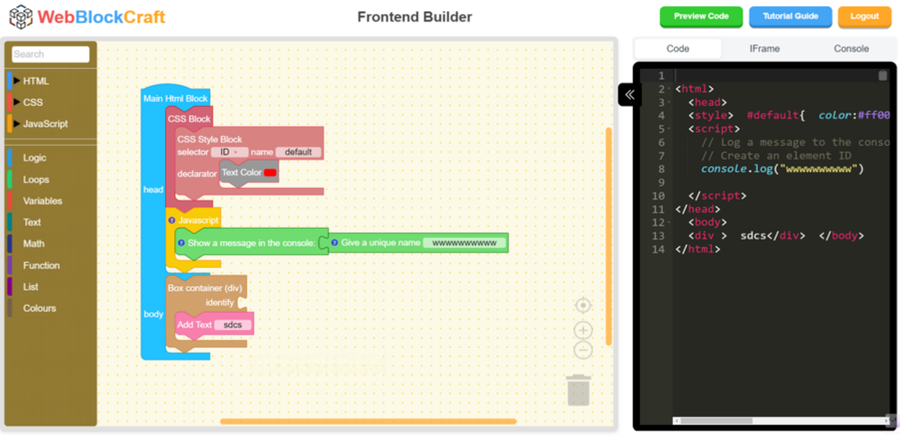
3.1. Frontend implementation

At the heart of Web Block Craft's educational approach is its frontend implementation as shown in Figure 2, where users can seamlessly create frontend web projects using a visual block-based interface.

Incorporating key web development languages such as HTML, CSS, JavaScript, and DOM manipulation, Web Block Craft utilizes the power of Google Blockly to enable users to drag and drop blocks, thereby composing frontend code. The code snippets are then displayed directly within the interface, allowing users to inspect and understand the underlying logic. These codes can also be previewed separately to understand more about the logic behind the added blocks. A distinctive feature of Web Block Craft is the real-time output display. As users assemble blocks to create their frontend projects, the live output is simultaneously displayed within an embedded *iframe*. This not only provides an immediate visual representation of the code execution but also reinforces the connection between the visual blocks and the actual output. Users can witness the direct impact of their coding decisions, enhancing their learning experience. The frontend implementation serves as a dynamic and interactive coding playground.

3.2. Backend implementation

Web Block Craft goes beyond frontend development by introducing a robust backend implementation. Using Google Blockly, users can create server-side applications effortlessly as shown in Figure 3. The backend blocks cover essential aspects of backend development, including server creation, web app security, API management, and database handling. This approach not only demystifies backend development but also seamlessly integrates with the frontend, offering users a holistic understanding of full-stack development. Users have the flexibility to develop frontend and backend projects independently and then connect them through a URL. Each user is allocated a dedicated Docker container for backend development, ensuring a secure and isolated environment. The backend code is subsequently dispatched to a sandbox environment, where it undergoes execution, and users receive timely feedback. This iterative feedback loop enhances the learning process, allowing users to evaluate, debug, and refine their backend code effectively. Web Block Craft enables users to create a fully functional and customizable backend with integrated security and authentication within just minutes, a feature unmatched by any other platform.

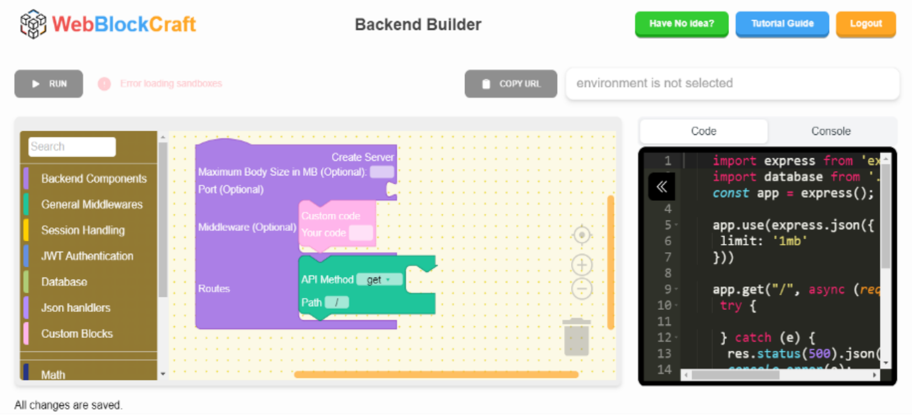


```

<html>
  <head>
  <style>  #default{  color:#ff00
  <script>
    // Log a message to the conso
    // Create an element ID
    console.log("wwwwwwwwww")

  </script>
</head>
  <body>
  <div >  sdcs</div>  </body>
</html>
```

Figure 2. This image represents the frontend builder



Error loading sandboxes

environment is not selected

```
    import express from 'ex
    import database from '.
    const app = express();

    app.use(express.json({
     limit: '1mb'
    }))

    app.get("/", async (req
     try {

     } catch (e) {
      res.status(500).json(
```

All changes are saved.

Figure 3. This image represents the backend builder

4. RESULT AND DISCUSSION

The importance of learning technologies like web development cannot be overstated, especially considering the current demand for skilled professionals in this field. However, there is a significant gap in the availability of easy-to-use and effective learning tools, particularly for backend development. Traditional methods often fall short in teaching these complex concepts, highlighting the need for alternative approaches. The Web Block Craft team conducted user testing on a group of 12 individuals who were a part of the initial survey. This group of 12 individuals aged 11 to 20, encompasses both beginners and those with prior web development experience. The key findings from this testing revealed a high level of satisfaction with the ease and efficiency of creating web applications using Web Block Craft. Participants appreciated the simplicity of dragging and dropping blocks, akin to solving a puzzle, rather than writing each line of code in traditional integrated development environments. The backend builder, in particular, received significant praise as participants were able to configure and launch a fully functional server with proper security and database connectivity in just a few steps, without the need for command-line interfaces or external tools. This evidence underscores the effectiveness of Web Block Craft in simplifying the web development process, making it more accessible and enjoyable for learners.

When compared to previous studies and existing tools, Web Block Craft stands out for its ability to address the complexities of both frontend and backend development through a visual programming approach. Traditional block-based platforms like Scratch and Tynker primarily focus on fundamental coding concepts but do not adequately cover advanced web development topics. In contrast, Web Block Craft provides a holistic solution that includes creating and configuring backend servers with integrated security and database connectivity. Web Block Craft's strengths lie in its ability to simplify complex concepts and provide a user-friendly interface that encourages interactive learning. However, a notable limitation is the current restriction that users can only utilize pre-made blocks, as there has not yet been developed a system for users to create their custom blocks. This limitation suggests an area for further improvement.

The primary purpose of Web Block Craft was to promote web development to young learners and provide a platform that allows them to create web applications without the complexities of traditional coding methods. The importance of this study is underscored by the increasing demand for web development skills in the job market, making it a valuable basic skill for anyone pursuing a career in computer science. While the findings clearly demonstrate Web Block Craft's effectiveness in enhancing the learning experience, there are still several potential areas for future research. Integrating artificial intelligence to provide instant feedback and assistance to students, as well as developing customizable blocks that allow users to create their functionalities, are promising directions for further exploration. Another extension would be introducing collaborative features to work on projects together. This will facilitate teamwork and interaction among users, making it easier for users to collaborate with peers and teachers will be able to help students easily. While the initial goal was to promote web development among young learners, the Web Block Craft framework has the potential to be extended to other areas of programming and computer science education. These advancements could significantly contribute to the educational community, providing a platform that makes learning both enjoyable and accessible.

5. CONCLUSION

Web Block Craft stands as an innovative and comprehensive educational application that leverages the Google Blockly framework to introduce web programming to a diverse audience. After identifying the primary challenges in teaching web programming to children and beginners, Web Block Craft addresses these issues with interactive and engaging solutions. Research findings revealed that Web Block Craft effectively simplifies complex web development concepts through visual, block-based programming, making it accessible and enjoyable for young learners. The integration of custom blocks for frontend and backend development, along with features like live code previews and step-by-step lessons, ensures a smooth transition from block-based learning to traditional coding. These advancements make Web Block Craft a valuable tool for fostering a lasting passion for coding among children and beginners. In conclusion, Web Block Craft represents a significant advancement in educational technology, particularly in the realm of web programming education. Web Block Craft not only empowers the current generation of web developers but also positions itself for future enhancements in the field of education.

ACKNOWLEDGEMENTS

The support and guidance of the Sri Lanka Institute of Information Technology (SLIIT) University and the dedicated staff of the Faculty of Computing are acknowledged. Their expertise, resources, and encouragement have been instrumental in the successful completion of this research project.

REFERENCES

[1] Google, “Blockly | Google for developers,” Google, https://developers.google.com/blockly (accessed Jul. 03, 2024).

[2] X. Wang, “Design, develop and teach the second web programming course in computer science curriculum,” *Journal of Computing Sciences in Colleges*, Apr. 2014, doi: 10.5555/2591468.2591479.

[3] H. N. Totan and A. T. Korucu, “The effect of block based coding education on the students’ attitudes about the secondary school students’ computational learning skills and coding learning: Blocky sample,” *Participatory Educational Research*, vol. 10, no. 1, pp. 443–461, Jan. 2023, doi: 10.17275/per.23.24.10.1.

[4] T. H. Park and S. Wiedenbeck, “Learning web development: challenges at an earlier stage of computing education,” in *Proceedings of the Seventh International Workshop on Computing Education Research, ICER 2011,* Providence, RI, USA, 2011, doi: 10.1145/2016911.2016937.

[5] W.-Y. Hwang, C.-Y. Wang, G.-J. Hwang, Y.-M. Huang, and S. Huang, “A web-based programming learning environment to support cognitive development,” *Interacting with Computers*, vol. 20, no. 6, pp. 524–534, Dec. 2008, doi: 10.1016/j.intcom.2008.07.002.

[6] C. S. Nam and T. L. Smith-Jackson, “Web-Based learning environment: A theory-based design process for development and evaluation,” *Journal of Information Technology Education: Research*, vol. 6, pp. 023–043, 2007, doi: 10.28945/200.

[7] S. Hadjerrouit, “Creating web-based learning systems: An evolutionary development methodology,” in *Proceedings of the 2006 Informing Science and IT Education Joint Conference*, 2006, doi: 10.28945/2997.

[8] P.-Y. Chao, “Exploring students’ computational practice, design and performance of problem-solving through a visual programming environment,” *Computers & Education*, vol. 95, pp. 202–215, Apr. 2016, doi: 10.1016/j.compedu.2016.01.010.

[9] V. L. Takacs, “BlockImpress,” in *IEEE 18th International Conference on Intelligent Engineering Systems INES 2014,* Tihany, Hungary, 2014, pp. 221-226, doi: 10.1109/ines.2014.6909372.

[10] M.-L. Hjorth, “Strengths and weaknesses of a visual programming language in a learning context with children,” *Degree Project in Technology and Learning, Second Cycle*, *30 Credits*, Stockholm, Sweden, 2017.

[11] U. Kaushal, G. Singh, and T. Parashar, “Responsive webpage using HTML CSS,” in *2022 International Conference on Cyber Resilience (ICCR)*, IEEE, Oct. 2022, doi: 10.1109/iccr56254.2022.9995922.

[12] S. Aggarwal, D. A. Bau, and D. Bau, “A blocks-based editor for HTML code,” in *2015 IEEE Blocks and Beyond Workshop (Blocks and Beyond)*, IEEE, Oct. 2015, doi: 10.1109/blocks.2015.7369008.

[13] R. Queirós, “LearnJS - A JavaScript learning playground,” *Open Access Series in Informatics*, vol. 62, Jul. 2018, doi: 10.4230/OASICS.SLATE.2018.2/-/STATS.

[14] A. Ashrov, A. Marron, G. Weiss, and G. Wiener, “A use-case for behavioral programming: An architecture in JavaScript and Blockly for interactive applications with cross-cutting scenarios,” *Science of Computer Programming*, vol. 98, pp. 268–292, Feb. 2015, doi: 10.1016/j.scico.2014.01.017.

[15] S. Xu, S. Lai, and L. Pollacia, “Integrating animation and game-making in teaching JavaScript,” in *2022 IEEE Integrated STEM Education Conference (ISEC)*, IEEE, Mar. 2022, doi: 10.1109/isec54952.2022.10025078.

[16] A. Marron, G. Weiss, and G. Wiener, “A decentralized approach for programming interactive applications with JavaScript and Blockly,” in *Proceedings of the 2nd edition on Programming systems, languages and applications based on actors, agents, and decentralized control abstractions*, in SPLASH ’12. ACM, Oct. 2012, doi: 10.1145/2414639.2414648.

[17] M. Miura, “Block Sweetie: Learning web application development by block arrangement,” in *2018 Thirteenth International Conference on Knowledge, Information and Creativity Support Systems (KICSS)*, IEEE, Nov. 2018, doi: 10.1109/kicss45055.2018.8950653.

[18] X. Huang, “Research and Application of Node.js core technology,” in *2020 International Conference on Intelligent Computing and Human-Computer Interaction (ICHCI)*, IEEE, Dec. 2020, doi: 10.1109/ichci51889.2020.00008.

[19] B. M. Adam, A. Rachmat Anom Besari, and M. M. Bachtiar, “Backend server system design based on REST API for cashless payment system on retail community,” in *2019 International Electronics Symposium (IES)*, IEEE, Sep. 2019, doi: 10.1109/elecsym.2019.8901668.

[20] Y. Valsamakis, A. Savidis, E. Agapakis, and A. Katsarakis, “Collaborative visual programming workspace for Blockly,” in *2020 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, IEEE, Aug. 2020, doi: 10.1109/vl/hcc50065.2020.9127253.

[21] “Scratch - Imagine, program, share,” *Scratch*, https://scratch.mit.edu/ (accessed Jul. 03, 2024).

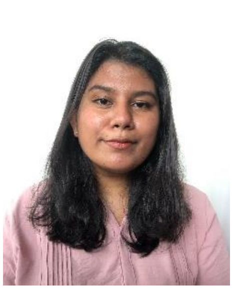
[22] “Game Lab,” *Code.org*, https://code.org/educate/gamelab (accessed Jul. 03, 2024).

[23] “Coding for Kids, Kids Online Coding Classes & Games,” *Tynker*, https://www.tynker.com/ (accessed Jul. 03, 2024).

[24] “Kodu Game Lab,” *KoduGameLab*, https://www.kodugamelab.com/ (accessed Jul. 03, 2024).

[25] “PictoBlox - Python and block-based AI programming software for kids,” *Stempedia*, https://thestempedia.com/product/pictoblox/ (accessed Jul. 03, 2024).

BIOGRAPHIES OF AUTHORS



Madhumini Gunaratne holds a B.Sc. (Hons) degree in Information Technology, specializing in software engineering, from the Sri Lanka Institute of Information Technology (SLIIT). She is currently working as a research assistant at SLIIT. Her research focuses on e-learning and generative AI (GenAI). She can be contacted at email: madhumini.g@sliit.lk.

Senal Weerasekara is an undergraduate at the Sri Lanka Institute of Information Technology (SLIIT), pursuing a B.Sc. (Hons) degree in information technology specializing in software engineering. He has a broad range of interests including electrical engineering, automobile technology, and scientific exploration. With a curiosity for innovation and a commitment to cross-disciplinary exploration, he is set to make a significant contribution to the ever-changing world of technology. He can be contacted at email: it20619794@my.sliit.lk.

Dehemi Weerakkody holds a degree in software engineering from the Sri Lanka Institute of Information Technology (SLIIT) and has accumulated over two years of experience in enterprise software development. He is deeply passionate about sharing his knowledge and expertise with young learners, particularly in software engineering practices. Currently, he continues to pursue his career in software development. He can be contacted at email: it20603618@my.sliit.lk.

Nisal Sashmitha is a final-year undergraduate student pursuing a B.Sc. (Hons) degree in information technology with a specialization in software engineering at SLIIT University, Sri Lanka. He is working as a software engineer at Calcey Technologies, Sri Lanka. His research interests encompass frontend application development and user experience, artificial intelligence, and robotics. He can be contacted at email: it20081034@my.sliit.lk.

Rivoni De Zoysa received her bachelor of information communication technology Honors degree from the Rajarata University of Sri Lanka in 2022. Since 2022, Rivoni has been an assistant lecturer in the Department of Software Engineering at SLIIT. Her research interests include machine learning, human-computer interaction, and IoT systems. She can be contacted at email: rivoni.d@sliit.lk.

Nuwan Kodagoda serves as the pro vice chancellor of the Faculty of Computing at SLIIT, with a PhD from Sheffield Hallam University. His research spans high-performance computing, e-learning, and computer science education pedagogy, focusing on enhancing student engagement and retention. Prof. Kodagoda has published extensively in these areas and has been instrumental in advancing the faculty’s curriculum and research capabilities. His leadership has fostered significant academic growth and international collaboration, aiming to integrate cutting-edge technological and pedagogical innovations in computing education. He can be contacted at email: nuwan.k@sliit.lk.