Engineering Educators Bringing the World Together
2025 ASEE Annual Conference & Exposition
Palais des congrès de Montréal, Montréal, QC • June 22–25, 2025
Paper ID #46632

Designing a Virtual World Experience to Foster Computational Thinking in Young Learners: An Hour of Code Initiative

Dr. Safia Malallah, Kansas State University

Dr. Safia Malallah is a teaching assistant professor at Kansas State University, where she completed her Ph.D. in Computer Science and early childhood. Her research is dedicated to advancing computer science and data science education across the PreK-12 and undergraduate levels. Dr. Malallah is particularly passionate about designing innovative and accessible learning experiences that cultivate essential computational skills in students

Dr. Ejiro U Osiobe

.

Lior Shamir, Kansas State University

Associate professor of computer science at Kansas State University.

Dr. David S. Allen, Kansas State University

David is an Associate Professor in the Department of Curriculum and Instruction at Kansas State University and the Director of the Center for STEAM Education. His work involves professional development for K-12 schools in STEAM related areas, and he is currently focused on on-line programing development in mathematics and computer science education.

Designing a Virtual World Experience to Foster Computational Thinking in Young Learners: An Hour of Code Initiative

Safia A. Malallah, Kansas State University, safia@ksu.edu
Ejiro Osiobe, dr.o@aneosiobe.org
Lior Shamir, Kansas State University, lshamir@ksu.edu
Allen S. David, Kansas State University, dallen@ksu.edu

Abstract:
The commercial gaming industry has significantly shaped virtual worlds that captivate millions of young users, often without an educational focus. A prime example is Roblox, a platform boasting over 35.5 million daily active users under the age of 13. The swift adoption of such platforms by children has far outpaced the creation of safe, educational alternatives, leaving a critical gap in free, accessible resources that foster controlled and purposeful engagement. This project addresses this gap by offering an "Hour of Code" experience where children assist island inhabitants in solving computer science problems through structured tasks, fostering an understanding of engineering, data science, coding, and artificial intelligence (AI) in a fun and engaging way. "The Hidden Island of Computational Thinking" is an immersive, narrative-driven virtual world built on Roblox, designed to teach children ages 6-12 the fundamentals of computational thinking through diverse computer science pedagogies. The educational island features five main sections, each representing a distinct CS field, offering tailored, age-appropriate challenges that engage children by allowing them to collect points. Additional areas include a store where points can be redeemed, a pet house where children can adopt companions, and a playground where they can interact with other players and their pets, making the learning experience both social and rewarding.

Introduction

Today's children are digital natives, growing up immersed in technology. Generation Alpha, born from 2010 to mid-2025, seamlessly integrates technology into their daily lives [1]. Platforms like the Roblox Studio, a popular online game with more than 35.5 million daily users (including approximately 8 million children under 9 [2]), showcase this trend [2]. While Roblox offers immense potential for learning, many similar platforms lack structured educational content. Furthermore, the educational landscape has evolved regarding technology use. Early childhood educators are moving away from restrictive approaches and toward actively guiding and monitoring children's digital interactions [15-17]. However, a gap exists between this evolving understanding and the availability of high-quality educational resources for young children within these digital platforms. This work aims to bridge this gap by leveraging the popularity of Roblox to create an engaging and educational experience for early childhood learners. By developing a Roblox-based "Hour of Code" experience, we aim to provide a safe and controlled environment for children aged 6–12 to learn computational thinking (CT). This project is guided by the Computational Thinking Pedagogical Framework for Early Learners (CTPF+) [3] and best practices in technology integration [4, 5]. The game design follows the Game Development Life Cycle (GDLC) [6] and game studies [7] principles, ensuring a structured and engaging experience. The virtual world (VW) integrates STEM-based missions into interactive, narrative-driven gameplay. The missions in coding, engineering, data science, and artificial intelligence align with the cognitive development of young learners. Through immersive narratives—such as protecting Mr. Elephant from disasters or teaching a recycling machine to sort trash—children engage with computational thinking concepts such as sequencing, pattern recognition, decomposition, and abstraction. A qualitative pilot testing with a small group of children demonstrated the potential of the VW to serve as an educational tool, though further development and evaluation are needed. By addressing gaps in early childhood education and leveraging popular platforms like Roblox, this project bridges the divide between learning and play, offering a scalable solution to teach modern concepts such as artificial intelligence (AI) and data science to young learners.

Background and Related Work

The Role of Virtual Worlds in Early Childhood Learning

Virtual worlds—immersive, interactive, digital environments where users can explore, socialize, and engage in various activities—have increasingly been recognized as powerful tools for learning in early childhood education. Research has shown that young children benefit from the dynamic and engaging nature of VWs, which allows them to explore concepts through hands-on, experiential learning. While concerns about screen time and content safety are valid, carefully designed VWs can offer a safe, educational space that supports the holistic development of young children, preparing them for future academic challenges [8]. These findings underscore the potential for VWs, such as those used in platforms like Roblox, to serve as valuable educational tools when combined with thoughtful pedagogical strategies. This paper explores the use of learning through VWs for children.

Roblox

Roblox, a popular online gaming platform, has become a prominent tool for VW experiences, especially among younger audiences. With millions of active users under age 13, Roblox provides an engaging and immersive environment where children can explore virtual worlds, create games, and socialize with peers [2]. This platform has attracted significant attention from both educators and researchers for its potential as a learning tool. Studies have shown that Roblox fosters creativity and collaboration, allowing children to experiment with game development, storytelling, and design. Its user-friendly interface and wide range of games appeal to children, while its open-world nature allows for exploration and interaction within a safe, controlled environment. However, despite its entertainment value, concerns have arisen regarding the platform's unregulated content and the risks associated with young children's online interactions [9]. As a result, there is a growing interest in leveraging Roblox's popularity to create educational experiences that are both engaging and safe. This paper explores the use of Roblox's sandbox environment to create a virtual world for children to learn modern concepts such as AI and data science.

Virtual Worlds Framework for Early Childhood Education (VWF-ECE)

Malallah's proposed a comprehensive Virtual Worlds Framework for Early Childhood, developed through a systematic process that included expert feedback, qualitative analysis, and iterative refinement. The framework consists of 21 items grouped into 13 distinct elements, each addressing critical aspects necessary to create virtual worlds that are engaging, inclusive, and developmentally appropriate for young children. These elements are: Engagement and Motivation (EM), Collaboration and Teamwork (CT), Creativity and Problem-Solving (CPS), Communication and Interaction (CI), Inclusivity, Accessibility, and Age-Appropriate (IAA), Design and Environment (DE), Data Security and Privacy (DSP), Safety and Technical Security (STS), Evaluation and Feedback (MEF), Cultural Responsiveness (CRR), Community Building (CB), Facilitation and Educator Tools (FET), Ethics, Empathy, and Decision-Making (EDM) [4]. This paper uses these elements to develop the virtual world environment in Roblox.

Table. 1 Elements for Virtual World [4].

The VW integrates STEM-based missions and scenarios encouraging problem-solving and exploration of solutions.
The VW offers timers or limited-time control features for structured activities.
The VW includes multiplayer activities with structured collaboration and clear communication channels.
The VW supports communication with age-appropriate tools (emojis, voice, pre-set phrases) and accessibility features (multilingual, adjustable fonts, contrast, audio).
The VW adapts difficulty levels based on the player's performance to keep them challenged but not frustrated.
The VW features activities that promote sharing, turn-taking, and negotiation.
The VW includes a structured tutorial to help children learn how to communicate within the platform.
The VW allows facilitators or parents to monitor and guide player interactions in real time.
The VW offers support or tutorials for children who may need additional assistance.
The VW offers customizable avatars that reflect diverse cultural, gender, and physical representations.
The VW allows for low-bandwidth modes to ensure inclusivity across technological constraints.
The VW adheres to child data protection laws and safeguards the environment by filtering inappropriate content and offering real-time monitoring tools for facilitators.
The VW provides clear and understandable privacy policies for parents and educators.
The VW provides comprehensive parental control options for managing settings and privacy.
The VW conducts regular third-party audits to ensure compliance with security and privacy standards.
The VW uses age-appropriate visuals animations and layouts to match developmental stages.
The VW incorporates scenarios that allow players to reflect on the consequences of their decisions.
The VW fosters empathy through activities that encourage understanding and includes age-appropriate ethical dilemmas to support moral development in young children
The VW provides real-time feedback and visual progress indicators to engage players and track achievements. It rewards positive behaviors and discourages harmful actions through constructive feedback.
The VW provides session reports and saves artifacts for assessment and reflection.
The VW hosts virtual events that bring players parents and educators together.

The besTech Framework II for Early Childhood

Malallah's proposed "besTech" framework considers the good, bad, and ugly aspects of technology. It addresses how bad aspects can be converted into good through good practices and ugly aspects can be eliminated with the implementation of technology best practices. It includes eight elements that need facilitator consideration: content, environment, pedagogy, evaluation, child, content, context, and screen time [5].

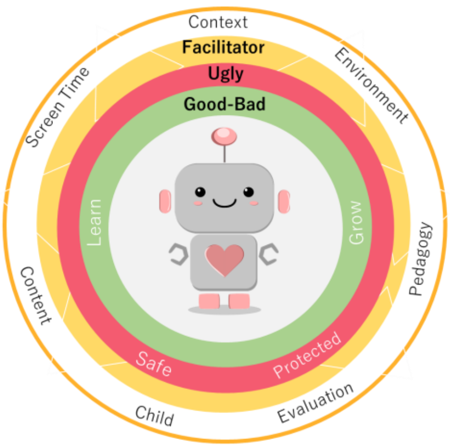


Figure 1. "besTech" *Framework II* [6]

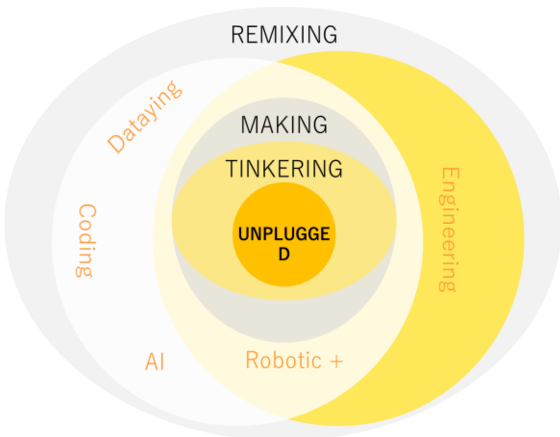


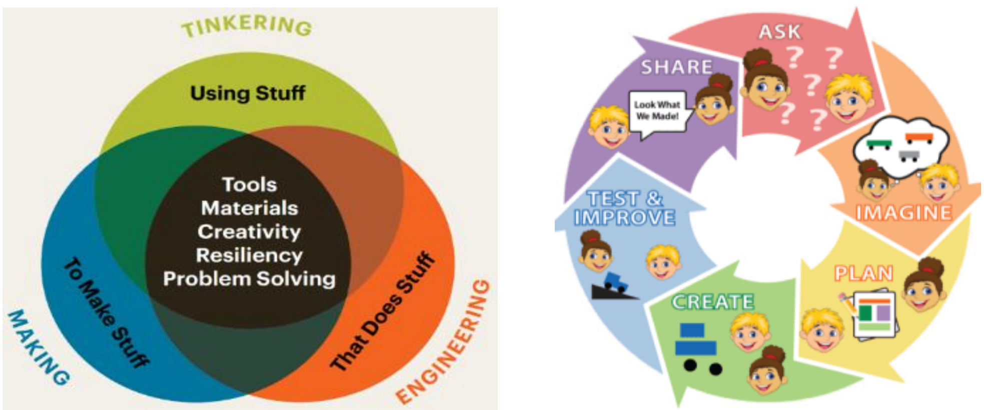
Figure 2. CTPF+ [5]

Computational Thinking Pedagogical Framework + for Early Learners (CTPF+)

Malallah's proposed Computational Thinking Pedagogical Framework+ (CTPF+) for Early Learners to help educators identify available CT experiences to incorporate them into their lessons. The framework includes nine pedagogical experiences: (1) unplugged, (2) tinkering, (3) making, (4) remixing, (5) robotics+, (6) engineering, (7) coding, (8) datatying, and (9) AI [4]. This paper uses these pedagogical concepts to develop and design the VW missions.

Tinkering, Making, and Engineering for Early Childhood

Tinkering, making, and Engineering are creative processes that promote exploration and innovation, and they can be seamlessly applied in VWs. Tinkering involves taking objects apart and modifying or repurposing them, emphasizing experimentation and adaptation. As NAEYC described, it is "using stuff" to explore and change components. Making focuses on building new objects, either by following instructions or designing freely, described as "using stuff to make stuff" that may be functional or purely creative. Remixing involves reconfiguring existing objects or ideas for new purposes, combining elements to create something unique [10]. In VWs, these activities enable learners to explore, construct, and innovate using digital resources, fostering creativity and problem-solving. Although the NAEYC provides a description of engineering for early childhood, this work adopts Ber's Engineering framework instead. In virtual worlds, these activities enable learners to explore, construct, and innovate using digital resources, fostering creativity and problem-solving.



a. NAEYCs' Tinkering, Making and Engineering Model **b. Bers' Engineering Model [12]**

Figure 3. Engineering

Engineering for Early Childhood

Engineering in early childhood involves using materials to build physical items that address specific problems or needs. It is a process driven by purpose and creativity, often requiring children to define a problem based on criteria such as available resources and time constraints. Bers outlined an engineering cycle for young learners that includes six steps: suggesting possible solutions, selecting the most suitable one, creating a prototype, testing it, and refining the design. These steps encourage problem-solving and iterative thinking, making engineering an effective hands-on learning approach for early childhood education [11].

Dataying – Data Science literacy for Early Childhood

Malallah proposed a dataying framework to teach data science concepts to young children ages 4–7 years old. The framework development included identifying K–12 data science elements and then validating element suitability for young students. Six cycled steps were identified: identifying a problem, questioning, imagining and planning, collecting, analyzing, and story sharing [12]. This paper utilizes the dataying framework to develop data science missions within the VW environment.

Figure 4. Datayng [13]

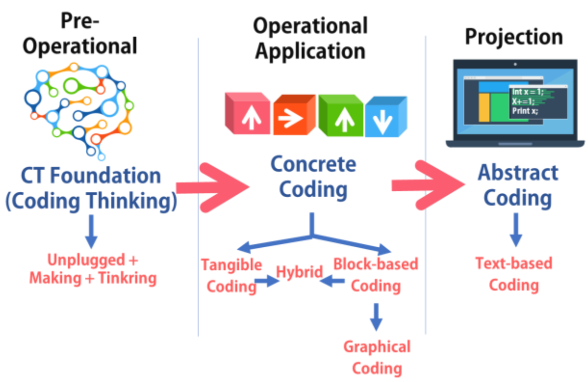


Figure 5. Coding [14]

Coding literacy for Early Childhood

Coding for young children introduces programming concepts tailored to their developmental stages, focusing on concrete and visual methods rather than abstract written syntax. It emphasizes developing computational thinking (CT) skills such as sequencing, pattern recognition, and decomposition. Malallah outlined three stages for teaching coding: pre-operational (CT foundation), operational (concrete coding), and projection (abstract coding). In the early stages, children use tangible and visual tools to interact with coding concepts, enabling them to grasp algorithmic principles and build problem-solving confidence. Coding methods include graphical-based coding, which uses icons and images for pre-literate children, block-based coding for assembling text-based blocks, and tangible coding with physical objects providing immediate feedback. Hybrid coding blends these approaches, catering to diverse learning needs [14]. This paper focuses on a task where children code the movement of an object from point A to point B on a grid, introducing fundamental CT concepts such as spatial reasoning and algorithmic thinking in an engaging and developmentally appropriate manner.

AI literacy for Early Childhood

AI literacy for early childhood refers to the ability to understand, use, and engage with AI in a knowledgeable and ethical manner. It involves equipping individuals, even at an early age, with the foundational understanding of AI's basic functions and applications, as well as the critical thinking skills to use AI responsibly in everyday life. According to a recent research review, AI literacy encompasses knowing and understanding AI, using and applying AI, evaluating and creating AI, and considering the ethical implications of AI, such as fairness, privacy, and responsible use [13]. This paper focuses on introducing concepts of supervised learning, particularly decision trees, as a foundation for AI literacy in early childhood. The AI missions are designed to help students explore and apply these concepts in an interactive and age-appropriate way. Through the missions, children collect data and build a classifier to predict outcomes based on input features by following a series of decision rules.

Using Hour of Code to learn

In recent years, the "Hour of Code" initiative has become pivotal in promoting computer science education to young audiences. Launched by Code.org, this program provides accessible, short coding tutorials aimed at demystifying computer science and encouraging students, especially those in elementary and middle school, to engage with the subject. The initiative is designed to be inclusive, catering to all skill levels, and is often integrated into classroom activities as well as extracurricular programs. Research on the effectiveness of "Hour of Code" has shown that it successfully sparks interest in coding and computational thinking, with many participants reporting an increased desire to learn more about technology and problem-solving [15]. Building on this foundation, this paper develops a blueprint for a VW experience that can be used as an interactive, engaging "Hour of Code" activity.

Game Development Life Cycle (GDLC)

The Game Development Life Cycle (GDLC) is a structured model that outlines the streamlined engineering principles necessary to build a robust software architecture for games across all platforms. A universal GDLC process typically consists of six key steps: idea, game design, technical requirements, development, testing, and deployment [6]. The idea phase establishes the purpose and objectives behind the game. The game design phase serves as the blueprint for the game, encompassing essential elements such as the rules of play, game story, user interface, environment theme, avatars and characters, missions and tasks, and audio. In the technical requirements stage, all frameworks and rules of play are translated into a requirements document, which guides the development process. The final steps involve thorough testing and the deployment of the game into a real-world environment [6]. The GDLC framework guided the development stages of the VW created for this research, ensuring a structured and efficient approach to game creation.

Game Studies

Salen et al.'s rules of play outlined essential game elements—objective, attributes, procedure, environment, and interactions—that are key to transforming lessons into engaging games. Understanding these elements is crucial for designing effective educational experiences in VWs.

(1) Objective: In VW lessons, the game objectives align with educational goals. Completing game tasks such as quests or missions helps students achieve lesson objectives. For example, building a farm to learn about animals integrates game mechanics with educational content, allowing students to focus on both gameplay and learning goals.
(2) Attributes: These include the characteristics of the game that control its operation (i.e., rules and boundaries). For example, a rule like "touching fire kills a player" creates consequences that challenge players to react quickly, supporting cognitive development.
(3) Procedure: This refers to the sequence of actions needed to progress in the game. In VW lessons, the procedure should be well-defined, guiding students through tasks and ensuring they meet educational objectives. The game's theme plays an important role in structuring the lesson and engaging students.
(4) Environment: The environment in a VW includes both physical and mental obstacles. Physical obstacles might require actions such as pressing buttons or moving objects; mental obstacles challenge players to solve puzzles or make decisions, enhancing cognitive skills.
(5) Interactions: This involves how players interact with the game and with each other [7]. Roblox, for example, allows players to collaborate or compete in both private and public servers, which fosters social interaction and collaboration during gameplay.

The principles of game studies informed the development of the VW in this research, leveraging Roblox's sandbox environment to design an interactive and engaging educational experience.

Method

Design

The GDLC [6] and game studies theories [7] guided the development stages of the VW while considering Roblox as sandbox technology. Consequently, the game design and technical requirements were combined in this stage of the development process.

Four children participated in a preliminary pilot study to evaluate the design and gameplay. At this stage, the core blueprints for the design and missions have been established, though many of the missions are still under development. The coding section has been fully implemented and is operational, showcasing the potential of the VW and providing a foundation for further enhancements.

Figure 6. Computational Thinking Virtual World Development Cycle

Project Idea - Game objective

The main goal of this project is to harness the popularity of Roblox, a platform familiar to millions of children, to create an engaging and educational "Hour of Code" experience for those age 6 and up. Instead of simply using Roblox for entertainment, we will guide young users toward learning-focused VWs, where they can explore core concepts of computational thinking such as engineering, data science, coding, and AI and ARISE (Adaptive and Resilient Infrastructure driven by Social Equity) missions [17] in a fun and engaging way.

Framework and Requirements

The development of the VW and its missions was structured and informed by the resources and principles outlined in the background section. These foundational resources provided a clear framework, ensuring that the design, functionality, and educational objectives aligned with best practices and met the intended learning goals.

Game Design - Story

The project employs a narrative-driven approach to engage children aged 6–12 in learning CT and foundational computer science concepts. Through the lens of an adventurous and magical tale, the game inspires curiosity and problem-solving while immersing players in a world of challenges and discovery.

The General Story: The Hidden Island of Computational Thinking. "Once upon a time, a mysterious island appeared, hidden from the world, visible for only one hour each year. Known as the Hidden Island of Computational Thinking, it was a place of wonder, home to three extraordinary families: Mr. Elephant's family, the Flamingo family, and Granny Squirrely's family. Each family faced unique challenges that only the brightest young adventurers could solve. The families cast a powerful spell, summoning courageous children to their island. One day, you—an exceptional child—received an invitation to join them. But the island held a challenge: You must help the families solve puzzles, uncover secrets, and tackle tasks that require computational thinking. Are you ready to help? Can you promise to try your best and give your all? If so, an unforgettable adventure awaits you!"

Coding Story: Baby Flamingo's Care. The Flamingo family faced a dilemma. Mommy and Daddy Flamingo needed to visit their sick grandmother, but their baby flamingo was too small to fly and needed care at home. This little flamingo could not speak, but it could perform five actions: move forward, turn left, turn right, move backward, and drink milk—especially important because it needed milk every three minutes. Trusting you to care for their baby, the Flamingo family asks you to guide the baby flamingo to its feeding bottle. Can you use your problem-solving skills to make sure the baby stays happy and healthy while Mommy and Daddy Flamingo are away?

Engineering Story: Protecting Mr. Elephant. Mr. Elephant is a kind but sensitive animal who becomes frozen with fear whenever a natural disaster strikes. Tornadoes, floods, fires, and heavy rains constantly strike his home, leaving him vulnerable. He needs your help to build protective structures that will keep him safe. Using blocks of various shapes and sizes, you will design shelters and barriers to shield Mr. Elephant from harm. Will you be the engineer who gives him the courage to stand tall?

Data Science Story: Granny Squirrely's Escape Room. Granny Squirrely has always loved puzzles and riddles, locking her house with codes that only the cleverest could solve. But as she ages, she has started forgetting the codes herself. Now she is locked out of her house and needs your help to crack the riddles and unlock the doors. In return, she will reward you with access to her legendary slide—an incredible ride no one else is allowed to use.

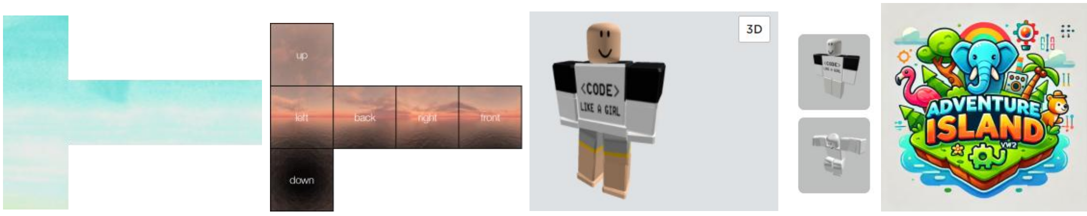
AI Story: Fixing the Recycling Machine. The Twin Monkeys have a mission: to clean up the trash that has been mysteriously teleported to the island. They have invested all their savings into a recycling machine to process the waste, but there is one problem—the machine has no brain! Instead of sorting the trash, it spits it back out. The monkeys need your help to design and build the machine's brain. Using AI principles, you will teach the machine to sort trash into the correct bins. As a reward, every piece of trash you recycle will earn you coins, and if you use the machine, they will double your reward. Can you save the island and make recycling a success?

Technical Development

The VW was built on the Roblox Studio platform, utilizing its templates as the foundation for creating interactive and engaging elements. Templates were customized to include 2D and 3D assets tailored to the game's needs. Photoshop and Illustrator were used to design 2D graphical elements; Cinema 4D was employed for the creation of 3D models. The interactions between the game elements, players, and the VW environment were programmed using Lua scripting and the drag-and-drop tools provided by Roblox Studio. Roblox's sandbox environment also includes built-in security policies and operational features, which the VW inherits by default. Also leveraged were Roblox's default avatar designs. While these avatars come with standardized shapes, developers may customize body parts and create unique clothing designs to match project requirements, therefore personalizing the game. The development process—creation, testing, and publishing—was carried out within the Roblox Studio. During testing, the focus was on identifying and fixing bugs to enhance gameplay functionality. Although the VW is a prototype and not all features are fully implemented, the project has been successfully published on Roblox. Players can access and play the game across various devices—including laptops, tablets, and smartphones—by creating a Roblox account and following the game's link.

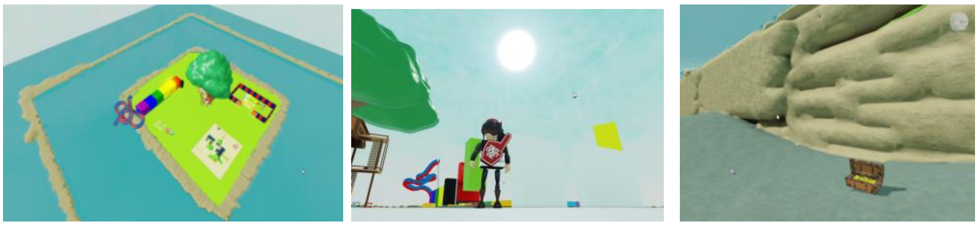
Game Design - Environment

The VW environment is carefully designed to not only support the educational goals of the project but also to create an immersive, interactive experience for young children. Set on a vibrant island, the environment uses bright, appealing colors and a cohesive, inviting, and engaging theme. The island's architecture, sky, and climate are harmonized with the color scheme to maintain a visually pleasing atmosphere. The island is populated with moving creatures, plants, and interactive elements, making it feel alive and dynamic. The solid land ensures that children are always safe from falling into holes or becoming stuck, while the surrounding water offers opportunities for exploration and learning, even extending to the bottom of the sea. To keep players from venturing too far from the island, the coastline features transparent glass barriers, creating a natural boundary that keeps players from getting lost or trapped. The world has written signs and audio that the players can trigger to describe the activities.



a. Sky Template **b. Avatar** **c. VW logo**

Figure 7. 2D and 3D VW Designs



a. Island **b. Sky** **c. Under the water**

Figure 8. Environment

The environment is organized into four main sections, each dedicated to a core area of computer science: coding, engineering, data science, and AI. These sections are thoughtfully designed with age-appropriate challenges, providing children with a clear, engaging path for learning.

a. Coding **b. Engineering** **c. Data Science** **d. Artificial Intelligent**

Figure 9. Environment: Main Sections

In addition to these main areas, the environment also includes four secondary sections designed for play and relaxation: the treehouse, playground, store, and great slides. These areas offer children a space to unwind, interact with other players, and enjoy the rewards they earn throughout their educational journey.

a. Playground **b. Store** **c. great slides** **d. Tree house**

Figure 10. Environment: Secondary Sections

Game Design – Nonplayer Characters

The nonplayer character (NPC) is not a real player but exists to guide the players. The VW should have an NPC at every activity spot to guide the children, provide help, or assign missions to players. NPCs can also engage with players by facilitating the purchase or sale of items. There are seven main NPCs: Mr. Elephant, the Flamingo family (Mr. Flamingo, Mrs. Flamingo, and Baby Flamingo), Granny Squirrely, and the mischievous twin monkeys. NPCs play a vital role in guiding players through the VW and enhancing their learning experience. While NPCs are not controlled by players, they act as integral components within the environment. Each key activity area within the VW is assigned an NPC to ensure players receive guidance, help, or instructions. Additionally, NPCs make the world feel dynamic and engaging by facilitating transactions, offering rewards, or providing hints for more challenging tasks. They help drive the narrative forward and also immerse players in the story.

a. Flamingo family **b. Twin monkeys** **c. Mr. Elephant** **d. Granny Squirrely**

Figure 11. Nonplayer Character

Game Design - Audio

The VW will feature background music with a happy, island-themed beat. Each mission also will have its own unique soundtrack to enhance the experience. The Roblox sandbox provides built-in sounds such as beach waves, jumping, and footsteps to further immerse players in the environment.

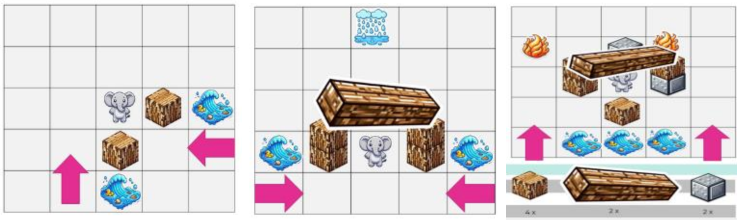
Game Design - The Mission

Missions in the game are essential tasks that align with both gameplay and educational objectives. These missions engage children with CT concepts, offering interactive problem-solving opportunities that teach valuable computer science principles. Designed to be immersive and educational, each mission encourages children to apply their knowledge to help the island's families. The missions are divided into four categories: engineering, coding, data analysis (dataying), and AI.

Engineering missions focus on applying logic and spatial reasoning, challenging children to design and build structures to protect Mr. Elephant from natural disasters. Using a limited set of resources (blocks of various shapes and sizes), children must construct functional solutions such as shelters or dams to withstand disasters—fire, rain, tornadoes, and floods. As missions progress, the disasters become more dynamic, requiring children to anticipate changes in direction or intensity and to select the appropriate materials for protection. These constraints encourage children to apply core engineering principles such as stability, balance, and resource management. When structures fail, feedback is provided, helping children identify mistakes and refine their designs.

The mission is divided into levels of difficulty:

- Easy: Basic shapes and more blocks, encouraging experimentation
- Medium: Limited resources and specific shape constraints to increase the challenge
- Hard: Complex designs that require precise spatial planning under time pressure



a. Easy b. Medium c. Hard

Figure 12. Engineering Missions

Coding missions introduce children to programming concepts by guiding a baby flamingo across a 5x5 grid to reach its feeding bottle. Children write step-by-step instructions using four basic commands: forward, backward, rotate left, and rotate right. Once the instructions are programmed, they can execute them by pressing the play button to move the flamingo. As missions progress, obstacles are added to the grid. Some obstacles require children to adjust their paths, while others demand creative solutions such as jumping or taking additional steps. The challenge lies in identifying the most efficient path, which is not always obvious. These tasks develop critical coding skills, including sequencing, debugging, and problem-solving, while demonstrating how programming controls digital objects.

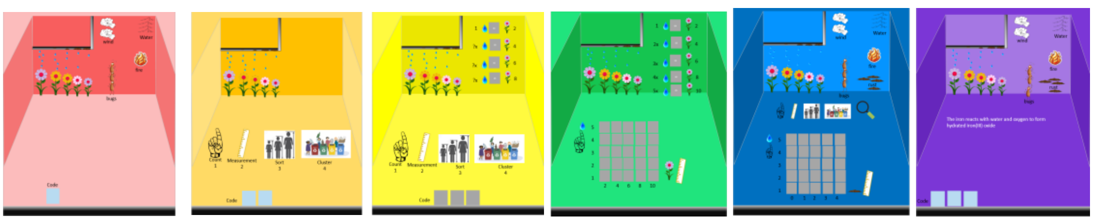
a. Easy b. Medium c. Hard

Figure 13. Coding Missions

Dataying missions introduce children to the basics of data science through an engaging, escape-room-style challenge. Players solve riddles and analyze clues to find a passcode and unlock virtual doors. Each mission is designed to represent a step in the data science process, progressively solving a larger problem. In the game, children help a farmer determine why flowers are growing at different heights.

The game consists of multiple escape rooms, each corresponding to a step in the scientific investigation process:

- Room 1: Identify potential causes of the problem
- Room 2: Choose the appropriate tools to collect data
- Room 3: Complete a graph of the collected data
- Room 4: Re-collect data and refine the graph for better insights
- Room 5: Analyze data trends to draw conclusions.
- Room 6: Solve the overarching mystery (e.g., “The rust in a faulty watering system caused unequal water distribution, affecting flower growth.”)



a. Room 1 b. Room 2 c. Room 3 d. Room 4 e. Room5 f. Room 6

Figure 14. Dataying - Scarpe Rooms

AI missions teach children how machines can "think" like humans. Specifically, children learn about the decision tree algorithm, one of the fundamental methods machines use to make decisions. In the game, children explore the island to collect trash and exchange it for coins at the store. However, the recycling machine's brain is broken, and the child must repair it to sort the trash into the correct bins. The machine consists of five bins (plastic, paper, metal, glass, and remains) connected to an input section via a series of pipes. The goal is to repair the machine so that trash dropped into the input section is correctly sorted into the appropriate bin. Mission levels are designed to progressively teach children how to understand and apply decision tree logic. In levels 1–5, children begin by dragging key words like "plastic" or "glass" into the appropriate Boolean conditions on a decision tree graph to complete the sorting logic. This helps them understand how decisions are made step-by-step. In levels 6–10, the same graph from level 1 appears, but half of the paths are removed. The child must complete the missing paths by logically identifying the correct conditions and outcomes. In levels 11 and above, all elements of the graph are removed, leaving only pipes and resources. The child must build the entire decision tree from scratch by connecting the pipes and resources based on their understanding of how the machine should function. Finally, in Level 4, the child tests their constructed machine by placing collected trash into the input section. The trash flows through the machine and into the bins. If the trash reaches the correct bin, the child earns coins. However, if some trash does not sort correctly, the child can go back, troubleshoot, fix the machine, and retry with the remaining trash until all items are properly sorted. This iterative process encourages problem-solving and reinforces learning.

a. levels 1-5 **b. levels 6-10** **c. levels 11**

Figure 15. Artificial Intelligent

Game Design - Activities

Activities in the game are optional tasks that complement the missions and enhance the overall gameplay experience. While not mandatory, these activities encourage children to explore, create, and socialize, supporting the game's pedagogical goals of tinkering, making, and socializing. Examples include adopting and raising a pet, shopping, and engaging in creative or social play, either alone or with a group.

- Playground: The playground is a space for unstructured play, social interaction, and creative expression. Children can engage in traditional playground activities such as seesaws, slides, and swings, as well as interact with their pets.
- Pet raising: At the heart of the island is a treehouse, which acts as a hub for pet adoption and care. Children can adopt a virtual pet, which follows them on their adventures, providing encouragement and companionship. Pets require care, including feeding and playtime, fostering a sense of companionship, responsibility, and nurturing.
- Buying and selling at the store: The game features a store where children can spend earned points to customize their avatars with costumes, hats, and accessories, reinforcing the reward system and enhancing motivation to play. Additionally,

children can sell unwanted items for points, introducing them to basic economic principles such as trade and resource management.
- Exploring the island: The island is designed to inspire curiosity and creativity. Children can tinker by interacting with various objects, some of which trigger sounds or reveal hidden items. These objects can also be combined to create unique, imaginative items, encouraging experimentation and creative thinking.

Game Design - Rules

The game rules establish how players interact with the game world, complete tasks, and progress through challenges while maintaining engagement and motivation.

The player will start in a tree house in the middle of the island. Animated tales describe the situation. The children have a choice to start with any mission (Elephant Circus Tent, Flamingo Yard, or Granny Squirrely Escape Rooms). After finishing all missions, players return to the tree house to finish the game. To complete the VW game, players must fulfill designated missions. This includes playing at least one game from each STEM section or engaging in gameplay for a minimum of one hour. Some missions include time constraints, requiring players to complete tasks within a set period. Time limits, which vary depending on the difficulty level, encourage quick thinking and decision-making. Each task is designed to teach a specific CT concept, whether through missions or optional activities. Completing these tasks allows players to earn rewards while progressively advancing through increasingly challenging problem-solving scenarios. Players assume specialized roles during gameplay, with each role corresponding to a specific STEM-based task or challenge:

Engineering rules: Players are tasked with building structures using blocks while adhering to constraints such as limited resources or specific design requirements.

Coding rules: Players use sequencing skills to guide an object, such as a character or robot, from one point to another, requiring logical planning and execution.

Data science rules: Players solve riddles and analyze clues to unlock doors, demanding logical reasoning and careful attention to detail.

AI rules: Players apply algorithmic thinking to sort items into the correct categories, focusing on accuracy and completing the task within time constraints.

The game features a dual-point system of regular and digital citizenship points. Players earn regular points by completing missions and activities across the island. These points reflect player performance in tasks such as coding, engineering, data science, AI sorting, and environmental cleanup. Regular points can be spent in the store to enhance gameplay and avatar customization. Digital citizenship points reward children for making ethical and responsible choices in the game. Positive behaviors such as helping others or maintaining the island's environment are rewarded with digital citizenship points. Collaboration is encouraged in group activities, but players must follow rules that ensure equal participation and fairness. Players are penalized for engaging in behaviors that disrupt progress. Exploiting glitches or skipping steps in missions is discouraged and may result in reduced rewards or penalties.

Players can personalize their avatars with costumes, hats, and accessories purchased from the in-game store. Hats provide role-specific bonuses (e.g., extra points for coding or engineering tasks) and must be equipped before beginning related tasks to gain their benefits.

Players carry inventory that they collect or purchase. There are permanent inventory and limited inventory. The permanent inventory stays with the player through the games unless dropped. The limited inventory is related to a game; for example, the engineering games have a screen of resources to play with but not to take.

Players receive real-time feedback during tasks. For example, incorrect coding sequences or unstable engineering designs are flagged with prompts for revision. Players are encouraged to retry tasks and refine their solutions until they achieve success, promoting a growth mindset and resilience.

Limitations

As a prototype, not all features are fully developed; only the coding section is operational. The limited pilot testing, conducted with only four children, provides insufficient data to evaluate the broader usability, engagement, or educational effectiveness of the VW. Additionally, the project relies heavily on Roblox Studio, which imposes constraints on customization, functionality, and avatar design because of its predefined sandbox environment. While the VW is accessible across devices, the need for a Roblox account and device compatibility may limit accessibility for some users. Finally, achieving a balance between educational content and engaging gameplay remains a challenge requiring further refinement of interactive elements and reward systems to sustain player interest.

Conclusion

This work highlights the need for structured educational content within digital platforms that young children frequently engage with, such as Roblox. By developing an interactive "Hour of Code" experience tailored for early learners, this project leverages the immersive potential of VWs to introduce computational thinking concepts in an engaging and age-appropriate manner. Guided by established pedagogical frameworks and game development principles, the VW integrates STEM-based missions into narrative-driven gameplay, fostering problem-solving and critical thinking skills. Preliminary pilot testing underscores the promise of this approach, demonstrating its potential as an educational tool. However, further refinement and evaluation are necessary to maximize its effectiveness and scalability. Ultimately, this project serves as a bridge between play and learning, paving the way for innovative, technology-integrated approaches to early childhood education in computational thinking, AI, and data science. The game can be access using: https://www.roblox.com/games/6791204917/CT-Island.

Future Work

Future work will focus on completing the STEM missions—particularly engineering, data science, and AI—to provide a comprehensive learning experience. Larger and more diverse user testing will evaluate the VW's educational impact, usability, and engagement.

Acknowledgment

This work was funded by the National Science Foundation (NSF) with Grant No DRL GEGI008182. However, the authors alone are responsible for the opinions expressed in this work and do not reflect the views of the NSF.

References

[1] M. McCrindle, *Generation Alpha*. Hachette UK, 2021.

[2] J. Clement. "Distribution of Roblox audiences worldwide as of December 2024, by age group." https://www.statista.com/statistics/1190869/roblox-games-users-global-distribution-age/ (accessed 2025).

[3] Statista. "Roblox Games Users Global Distribution Age." https://www.statista.com/statistics/1190869/roblox-games-users-global-distribution-age/ (accessed 2022).

[4] S. Malallah, L. Shamir, W. H. Hsu, J. L. Weese, and S. Alfailakawi, "Computational Thinking Pedagogical+ Framework for Early Childhood Education," in *2023 ASEE Annual Conference & Exposition*, 2023.

[5] S. Malallah, J. Weese, and K. Alsalmi, "The "besTech" Technology Practice Framework for Early Childhood Education," presented at the ASEE Annual Conference, 2023.
[6] S. Malallah, "The besTech Framework II for Early Childhood Education," ed, 2025.
[7] S. Jain and T. Tkach, "Game development life cycle," *LinkedIn.: LinkedIn*, 2017.
[8] K. Salen, K. S. Tekinbaş, and E. Zimmerman, *Rules of play: Game design fundamentals*. MIT press, 2004.
[9] ENISA. "Children on virtual worlds—What parents should know." European Network and Information Security Agency. https://www.enisa.europa.eu/publications/archive/children-on-virtual-worlds/at_download/fullReport (accessed 2022).
[10] J. Clement. "Roblox games users distribution worldwide September 2020, by age." Statista. https://www.statista.com/statistics/1190869/roblox-games-users-global-distribution-age/ (accessed 2022).
[11] NAEYC. "What You Need to Know About Tinkering, Making, and Engineering." https://www.naeyc.org/sites/default/files/globally-shared/downloads/PDFs/resources/pubs/sample_what_you_need_to_know_about_tinkering_making_and_engineering.pdf (accessed 2022).
[12] M. U. Bers, *Coding as a playground: Programming and computational thinking in the early childhood classroom*. Routledge, 2017.
[13] S. Malallah, J. Weese, L. Shamer, and W. Hsu, "Data Science (Dataying) for Early Childhood," presented at the ASEE Annual Conference, 2023.
[14] S. A. Malallah, "Developing computational thinking best practices for early childhood education in Kuwait and United States," Kansas State University, 2022.
[15] J. Su, D. T. K. Ng, and S. K. W. Chu, "Artificial intelligence (AI) literacy in early childhood education: The challenges and opportunities," *Computers and Education: Artificial Intelligence*, vol. 4, p. 100124, 2023.
[16] J. Yauney, S. R. Bartholomew, and P. Rich, "A systematic review of "Hour of Code" research," *Computer Science Education*, vol. 33, no. 4, pp. 512-544, 2023.
[17] ARISE. **https://nsfepscor.ku.edu/track-1-arise** (accessed 2025).