2026 ASEE ANNUAL CONFERENCE June 21–24, 2026 | Charlotte Convention Center, NC
Host Sponsor UNC Charlotte
ASEE AMERICAN SOCIETY FOR ENGINEERING EDUCATION
Paper ID #51702

HackAdemyX: Game-based Learning Environment for Cybersecurity Education

VIKTOR NIKOLOV, Fairfield University
Kristopher Marte, Fairfield University
Dr. Akshay Mathur, Fairfield University

Dr. Akshay Mathur is an Assistant Professor of Computer Science at Fairfield University's School of Engineering and Computing, specializing in smartphone cybersecurity, malware detection, and adversarial AI resilience. He graduated with a Ph.D. in Engineering from the University of Toledo in 2022, majoring in Computer Science. His research combines technical rigor with an educational mission, focusing on practical tools for identifying malware threats, promoting user safety, and developing resilient AI-based security models. Dr. Mathur mentors undergraduate and graduate students through hands-on research and development projects and is committed to advancing cybersecurity awareness. He teaches a variety of undergraduate and graduate CS courses on cybersecurity, programming, software development, and data analytics, and serves as the academic advisor for Fairfield's chapter of UPE, a CS honor society.

WIP: *HackAdemyX:* Gamification and Game-Based Learning for Foundational Cyber Education

1 Introduction

Today's rapidly changing digital landscape has seen an unprecedented rise in cyber threats, yet many teenagers and young adults do not fully comprehend the drastic impact of cyberattacks. For students starting university education, the threat is acute, since most degree programs today rely heavily on online platforms and smart devices to deliver content [1, 2]. Yet, basic cybersecurity education remains underrepresented in high school and undergraduate non-engineering degree programs [3, 4]. Although some high school graduates exhibit a general understanding and need for online security mechanisms, most are unaware of existing threat-mitigating security controls. Current cyber educational programs focus on general security practices using traditional pedagogical methods, such as lectures, labs, and assignments, which are effective for students interested in computer science, but does not apply to students in non-STEM (Science, Technology, Engineering, and Math) programs.

Gamification has seen a significant acceptance across STEM and Computer Science (CS) education. It involves acquiring knowledge from serious games integrated with content-related game mechanics to deliver content and captivate individuals to enhance learning and inspire action [5]. Game-based Learning (GBL), where knowledge is gained by developing a game has been effective as well. GBL-based coding platforms, such as TryHackMe, CodeCombat, and GameDevDojo, have gained massive popularity and demonstrated learning effectiveness among STEM and non-STEM students. While GBL environments are available and effective in many domains, cybersecurity education needs further exploration [6]. Therefore, we believe education about cyber threats and corresponding controls, with engaging activities and challenges on real-life scenarios, could enhance awareness towards digital threats. [7].

To this effect, we propose *HackAdemyX*, a 2D story-based GBL environment in which cybersecurity concepts are embedded directly within the gameplay experience. This is being developed by undergraduate CS students, where they learned the concepts of game development through an incremental approach for designing, developing, and testing the game. The contributions of this work are:

1. Designing and developing a story-based game focusing on teaching foundational cybersecurity concepts and necessary precautions for threat mitigation at each level in a

role-playing, Capture The Flag (CTF) format.

2. Developing interactive challenges, puzzles, and infomercials integrated with secure programming practices for game-based learning education and testing the player's and developer's understanding of the concepts simultaneously.

2 Related Work

In STEM, GBL has proven effective in subjects like science, climate education, and health. Fante et al. [8] conducted a decade-long review of GBL in secondary STEM education. Their study showed a shift in research focus, from basic game mechanics to learner engagement and emotional design. Although broad in scope, the study focused only on abstracts and titles rather than real-world implementations, limiting insights into classroom effectiveness.

Another group of researchers created a 2D educational quiz game to raise COVID-19 awareness in Indonesia. Built with Construct 3 and validated through black-box testing and user surveys, the game applied gamification effectively [9]. However, its design was limited to multiple-choice questions and did not include adaptive difficulty, interactive mechanics, or progress tracking. In the area of science education, RumbleBlocks used Unity to help young children learn balance and stability concepts through physics simulations [10].

For CS education, a study conducted by a research team[11] revealed that CodeCombat[12] considerably enhanced primary students' computational thinking, especially in creativity, algorithmic thinking, and problem-solving skills. Over a two-week period, the research involved 49 first-grade students and assessed their abilities using the Computational Thinking Scale. The findings endorsed CodeCombat as a valuable resource for introducing programming to young learners. One main study by Comber et al. introduced GameDevDojo [13], a Unity-based platform designed to teach game development fundamentals in CS classrooms. The system provided guided steps and challenges that encouraged hands-on, project-based learning [14]. Another notable project, Bunny Code [15], presents a web-based game that teaches programming basics using drag-and-drop coding, animations, and storytelling. CyberGuardian [16] is one interactive educational game that introduces students to basic concepts in cybersecurity using scenario-based learning, such as cryptographic primitives and adversarial thinking.

While GameDevDojo promotes hands-on learning through guided challenges in a project-based learning environment, it lacks measurable data on learning outcomes, making it difficult to assess its educational impact. On the other hand, Bunny Code offers a visually engaging introduction to coding for children through drag-and-drop mechanics and storytelling, but lacks a feature for progress-tracking or teaching about safe coding practices. Cyberguardian, however, talks about GBL for novice users, but dives into complex encryption techniques, which would be useful for students interested in the field of cybersecurity. Moreover, these games focus on individual capture-the-flag activities, which leaves a disconnect between the various cybersecurity concepts they utilize to complete their activities.

3 *HackAdemyX* Methodology

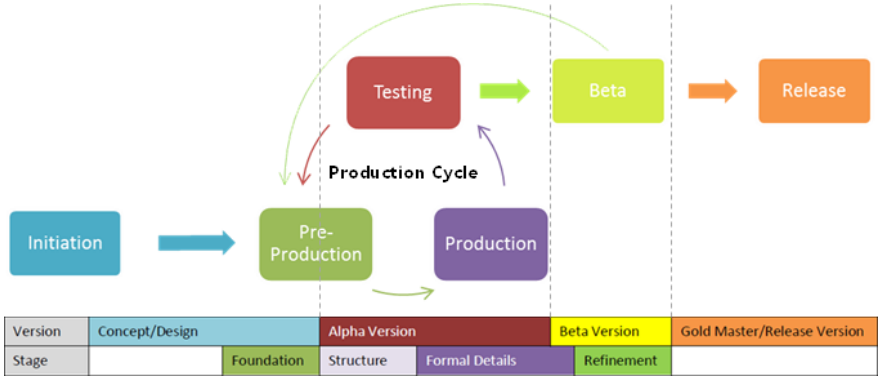


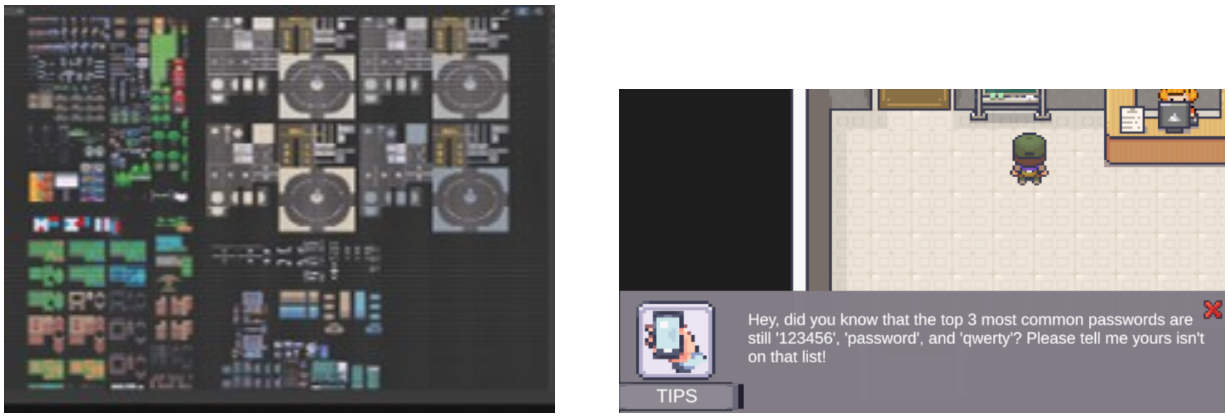
Figure 1: Game Development Life-cycle[17]

HackAdemyX is a single-player 2D learning environment that used gamification in cybersecurity foundations for players and the application of a secure programming paradigm for undergraduate students using the traditional iterative game development life cycle as seen in Figure 1. Gamification starts at the initiation phase to identify cybersecurity concepts and bind them to game design concepts, tools, and outcomes for the game-based learning.

i. **Cybersecurity Topics:** During this phase, we identified the basic applicable cybersecurity concepts that may help novice users everyday. Adolescent social media habits make them targets since they disclose personal information, which malicious threat actors exploit to launch spear phishing attacks and scams on social media platforms[18]. Hence, learning about identity access management, Multi-factor Authentication (MFA), and phishing/smishing is necessary for responsible online behavior. We modeled each of these concepts in the game environment, with each level providing key cybersecurity fundamentals through carefully curated challenges, informational dialogues, and interactive quizzes.

ii. **Tools and GBL Techniques:** The game was developed in Unity [19] to provide three junior undergraduate computer science students with experience to improve their secure programming concepts. We prioritized collaboration among the student team and their advisor through daily sprint meetings, prototype development, and exposure to industry-standard practices. All core systems, like player movement, NPC dialogue, puzzle logic, scene transitions, animations, and so on, were implemented in C#. To build this world efficiently and maintain visual consistency, we used Unity's Tile Palette (Fig. 2a) to design maps and environments with a 32x32 grid precision.

iii. **Game Design:** The game follows a role-playing story-based progression to complete the game's objective. To win, the player must cross the game levels by understanding, reasoning, and implementing the cybersecurity concepts learned and overcoming the challenges modeled as different levels in the game. We modeled the concepts in the game environment, with each level providing key cybersecurity fundamentals through carefully curated challenges, informational dialogues, and interactive quizzes. The content was



(a) Unity's Tile Pallet Grid Tool

(b) NPC Dialog Design

Figure 2: Unity components in the game

delivered using various scaffolds such as tool-tips, non-playable characters (NPCs) interactions, hints, and storyline. These challenges range from solving a small mathematical problem to intimidating an in-game (NPC) adversary's efforts to breach a system. These experience points also serve as a metric to determine the player's mastery of the concept. Some NPCs introduce the game's core mechanics (movement, inventory, interaction), while others gradually transition into more complex dialogue, as seen in Figure 2b, to allow information capture and help understand content.

iv. **Performance Evaluation Metrics:** To keep the players engaged and collect progression data, we implemented positive reinforcement using the eXperience Points (XP) model. These were based on the level of mastery shown in completing tasks, which was calculated using the number of correct decisions, time taken, number of tries, and items collected, among others. To understand the performance of the game, we implemented a logging system to keep track of the game and the players' progress.

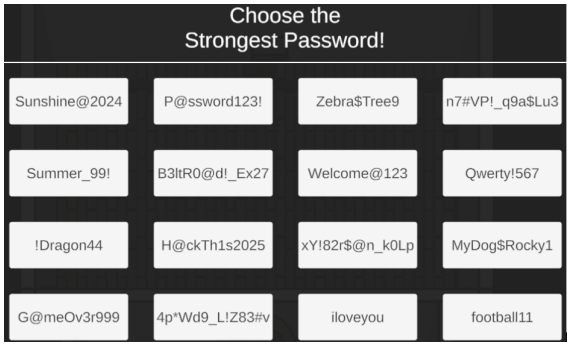
v. **Alpha Version:** Starting with the pre-production phase, we prioritized modular system design, planned puzzle mechanics, outlined dialogue scripting, and tested early prototypes of the user interface to speed up level design and maintain consistent visual flow across scenes. Unity's Play Mode allows for real-time testing and debugging, aiding in the early identification and addressing of issues. We categorized bugs into three tiers – minor, medium, and major. Minor bugs typically involve visual glitches that don't interfere with gameplay logic and are resolved promptly during periodic testing cycles. Medium bugs affect user experience but don't completely break functionality. Major bugs significantly disrupt progress or cause the game to crash.

4 Development Results and Outcomes

Once the design details were completed, we started prototype-driven agile development. For brevity, below are the core mechanics and systems we implemented in Level 1:

(a) Game Map

(b) In-Game Menu UI and Inventory



(c) A Challenge in Level 1

(d) NPC Adversary Challenge of Level 1

Figure 3: Game Features and Challenges in Level 1

Participant	Key Insights / Learning Outcomes
n1	*"Designing interactive components and implementing gameplay mechanics ... was an amazing learning experience."*
n2	*"Design and development in a team environment strengthened both my technical and collaboration skills while working across gameplay systems and scripting."; "... realize the importance of building technology that educates others, especially by making cybersecurity concepts accessible to non-technical users."*
n3	*".. its engaging and easy to follow, especially through the story-based format. I have a much clearer understanding of password security and how weak credentials can be exploited..."*
n4	*"The interactive challenges made it easier to recognize phishing attempts. Learned to identify suspicious links and think more critically before interacting with unknown messages. The final boss fight was challenging."*
n5	*"The inventory and clue system reinforced learning as I had to remember and apply information. I am more accepting of multi-factor authentication now and see it's importance currently."*
m1	*"NPC interactions are helpful for breaking down complex cybersecurity topics into simple explanations. Incorporating the clue system at the cost of XP points could strengthen serious game strategies. A short pretext at the start of the game would be good to provide a guided approach to the game in the initial levels."*
m2	*"The game has the potential to introduce cybersecurity concepts to non-CS audiences in an engaging way. Noted that the combination of challenges and feedback supports active engagement and knowledge gathering, but reinforcement in the following levels was lacking. The XP-points and rewards at level change would help with keeping the game interesting."*

Table 1: α-testing results with n students (n=5) and m faculty evaluators (m=2)

i. **Story Line and Objectives:** The overarching objective of the game is for the player to build themselves up as a cybersecurity engineer and defeat an NPC adversary who plans to steal sensitive information of the people at a university. The player starts off as a first-year undergraduate student in *"Guardians University"* with limited technical knowledge, and through various challenges and activities, becomes an expert to save the university from a vicious cyber attack. As shown in Figure 3a, the map is modeled as Guardians University's main campus with different areas ready for exploration. This was done deliberately to create a sense of familiarity and relevance for student users. Players are more likely to engage with a cybersecurity learning tool if it reflects environments they actually encounter in real life.

ii. **GamePlay:** The player is exposed to several challenges throughout the game, as the one shown in Figure 3c, as checkpoints at each level for testing user retention. At the end, the player engages in a turn-based battle against a "hacker boss" NPC (see Figure 3d). Once the player defeats the hacker, they receive calculated XP points and are prompted to continue their journey. A visual progress tracker displays challenges the player has completed out of the total required to unlock the final challenge ("5/5 NPCs completed"). This adds a layer of goal orientation and clarity, especially for younger players who benefit from visible markers of achievement. We adopted a dialogue-driven format to immerse the player in the world while subtly delivering key cybersecurity lessons, as shown in Figure 2b. Each NPC introduces a concept, hint, or challenge related to password protection or digital safety. To encourage full exploration and learning, and prevent speed runs, we structured the game so players must interact with all relevant NPCs and objects before the final puzzle or battle becomes available.

iii. **Inventory and Game Settings:** Players can pick up, inspect, and manage items found throughout the level (Figure 3b). Some items contain clues, such as access codes that are required to progress. We used this mechanic to reinforce information retention through visual repetition and interactivity. Scenarios where players locate, remember, and apply information in context, mirror how secure behavior works in the real world. Players can also manually save the game and access basic player stats. Map and Player tabs features are in development and will be ready in future versions of the game.

iv. **System and Alpha Testing:** Initial system testing included testing the integrated components using the tools and iterative testing mechanisms. To mitigate moderate and major bugs, additional safety checks and conditional triggers were introduced to ensure that required dialogues and puzzle conditions activated reliably even if players bypassed optional interactions. This iterative debugging process allowed the team to stabilize the gameplay systems before external testing. Initial α-testing results and feedback is shown in Table 1.

5 Conclusion and Future Work

The main goal of *HackAdemyX* is to increase cybersecurity awareness among teens and young adults by turning complex concepts into engaging experiences tied in a role-playing format.

Instead of static materials, the game employs immersive storytelling and gameplay to teach identity protection, access management, and phishing attempts. Gamified elements like adversary challenges and technical puzzles are adopted to mimic real-world decisions. Based on feedback from alpha testing, *HackAdemyX*'s potential for use in classrooms, after-school workshops, and home-based learning is yet to be evaluated. Future work would focus on testing the educational impact in a formal classroom environment and developing more levels on additional topics such as malware, ransomware, and secure browsing. The collected and aggregated gameplay data could also drive student improvement and Serious Game Design Assessment.

References

[1] ZDNet, "Phishing via sms: A rising threat to mobile security," *ZDNet*, 2024. [Online]. Available: https://www.zdnet.com/article/phishing-via-sms-a-rising-threat-to-mobile-security/

[2] F. T. Council, "Ransomware on mobile devices: What's new in 2024?" 2024. [Online]. Available: https://www.forbes.com/sites/forbestechcouncil/2024/09/12/ransomware-on-mobile-devices-whats-new-in-2024/

[3] A. Sultan, "Improving cybersecurity awareness in underserved populations," Center for Long-Term Cybersecurity, UC Berkeley, Whitepaper, 2020, whitepaper on cybersecurity in underserved populations. [Online]. Available: https://cltc.berkeley.edu/publications/improving-cybersecurity-awareness-in-underserved-populations

[4] B. Slavin, "A guide to navigating the cyber threat landscape for teenagers," 2024, https://www.duocircle.com/data-privacy/guide-navigating-cyber-threat-landscape-for-teenagers.

[5] F. Dahalan, N. Alias, and M. S. N. Shaharom, "Gamification and game based learning for vocational education and training: A systematic literature review," *Education and information technologies*, vol. 29, no. 2, pp. 1279–1317, 2024.

[6] S. Karagiannis, T. Papaioannou, E. Magkos, and A. Tsohou, "Game-based information security/privacy education and awareness: theory and practice," in *European, Mediterranean, and Middle Eastern Conference on Information Systems*. Springer, 2020, pp. 509–525.

[7] S. Purnama, M. Ulfah, I. Machali, A. Wibowo, and B. Narmaditya, "Does digital literacy influence students' online risk? evidence from covid-19," *Heliyon*, vol. 7, p. e07406, 2021.

[8] C. Fante, F. Ravicchio, and F. Manganello, "Navigating the evolution of game-based educational approaches in secondary stem education: a decade of innovations and challenges," *Education Sciences*, vol. 14, no. 6, p. 662, 2024.

[9] E. Sudarmilah and A. M. Mulyana, "The 2d educational game prevention of covid-19 in indonesia," *Journal of Education Technology*, vol. 6, no. 2, pp. 288–298, 2022.

[10] M. G. Christel, S. M. Stevens, B. S. Maher, S. Brice, M. Champer, L. Jayapalan, Q. Chen, J. Jin, D. Hausmann, N. Bastida *et al.*, "Rumbleblocks: Teaching science concepts to young children through a unity game," in *2012 17th International Conference on Computer Games (CGAMES)*. IEEE, 2012, pp. 162–166.

[11] K. Y. Choi and K. W. Choi, "The influence of codecombat on computational thinking in python learning at primary school," in *Proceedings of the 2024 8th International Conference on Education and Distance Learning (ICEDL)*. ACM, 2024, pp. 94–99. [Online]. Available: https://doi.org/10.1145/3632731.3632750

[12] CodeCombat Inc., "Codecombat," https://codecombat.com, 2013, educational game developed to teach programming through interactive gameplay.

[13] M. Holly, L. Habich, and J. Pirker, "Gamedevdojo-an educational game for teaching game development concepts," in *Proceedings of the 19th International Conference on the Foundations of Digital Games*, 2024, pp. 1–9.

[14] O. Comber, R. Motschnig, H. Mayer, and D. Haselberger, "Engaging students in computer science education through game development with unity," in *2019 ieee global engineering education conference (educon)*. IEEE, 2019, pp. 199–205.

[15] M. Y. B. Noor Rahimi, M. Z. B. Mohamad Rosdi, and A. M. Zeki, "Bunny code: Interactive educational coding games for kids." *Journal of Science & Technology (1607-2073)*, vol. 29, no. 1, 2024.

[16] S. Huang, G. L. Herman, and A. T. Sherman, "Cyberguardian: A role-playing educational game for learning cryptographic primitives in authentic cybersecurity scenarios," in *Proceedings of the 30th ACM Conference on Innovation and Technology in Computer Science Education V. 2*, 2025, pp. 745–746.

[17] Personanonymous, "A guidelines of game development life cycle (v2.0)," https://personanonymous.wordpress.com/2013/07/17/a-guidelines-of-game-development-life-cycle-v2-0/, July 2013, accessed: 2025-07-28.

[18] H. Collier and C. Morton, "Teenagers: a social media threat vector," in *19th International Conference on Cyber Warfare and Security: ICCWS 2024*. Academic Conferences and publishing limited, 2024.

[19] Unity Technologies, *Unity Game Engine*, https://unity.com/, 2025, version 2023.x. Available at: https://unity.com/.