Technical implementation of a visual novel as a tool for gamification of learning in accordance with cyber hygiene requirements

Oleksandr Bobarchuk[1,*,†], Andrii Tolbatov[1,†], Olena Matviichuk-Yudina[1,†] and Natalia Boltivets[1,†]

[1]*State University "Kyiv Aviation Institute", Liubomyra Huzara Ave., 1, Kyiv, 03058, Ukraine*

Abstract

A visual novel, as a relatively new and innovative approach to gamifying the educational process, requires adherence to cybersecurity measures during its creation. This study explores technical, pedagogical, and information security aspects, with a focus on cyber hygiene – a set of knowledge, skills, and technologies aimed at ensuring personal safety in the digital space. The work highlights modern technologies and approaches for developing interactive educational visual novels while meeting cyber hygiene standards: protection of personal data, tracking limitations, secure login mechanisms, avoidance of manipulative content, and fostering a critical perception of information. Special attention is given to modeling an interactive digital environment that not only motivates learning but also cultivates a conscious approach to cybersecurity among students. The technical implementation includes software solutions and project architecture using the example of Milorad Pavić's story The Glass Snail, studied by Ukrainian school students. The paper details the use of tools such as Ren'Py, Easy Paint Tool SAI, Godot, and Unity, along with algorithms for embedding cyber hygiene tools during development. It also substantiates the need to integrate digital ethics, secure programming, and privacy principles at all stages of content creation. The findings are valuable for developers of educational applications, digital discipline educators, cybersecurity professionals, and multimedia specialists working at the intersection of education, technology, and security. Theoretical and practical recommendations support the implementation of gamified solutions that align with modern cyber hygiene standards.

Keywords

visual novel, gamified learning, cyber hygiene, data protection, Ren'Py, cybersecurity awareness

1. Introduction

Education in the modern information society is undergoing transformations under the influence of digital technologies. Traditional methods of presenting educational material are increasingly losing their effectiveness among young people who are accustomed to an interactive and visually rich environment. Against this background, the popularity of gamification is growing as an approach to increasing motivation, engagement and the formation of practical skills in students. A special place among gamification tools is occupied by a visual novel - an interactive digital product that combines an artistic narrative, visual graphics, music, animation and a choice of user actions, ensuring immersion in educational content.

However, the growth of the role of digital tools in education is also accompanied by an increase in cyber threats. Unregulated use of game applications, vulnerabilities in program code, collection of personal data without consent - all this threatens the information security of users, especially minors. Therefore, there is a need for technical implementation of educational games not only with an emphasis

on pedagogical expediency, but also in accordance with the requirements of cyber hygiene - a system of knowledge, skills and technical measures that ensure safe interaction with the digital environment.

The purpose of the study is to analyze technological solutions that allow creating a safe, accessible and effective visual novel as a means of gamified learning. The choice of software, organization of the novel structure, integration of security elements and interface solutions that take into account the peculiarities of user behavior in the digital space are considered. Pedagogical aspects are also highlighted: how visual novels can contribute not only to the acquisition of knowledge, but also to the formation of digital culture, critical thinking and a responsible attitude to online communications.

In the process of technical implementation of a visual novel as an educational product, special attention is paid not only to the convenience and functionality of the software, but also to compliance with the requirements of cyber hygiene. This means adhering to the principles of safe use of digital tools, protecting users' personal data, reducing the load on hardware, and forming an informed attitude towards digital risks. Thus, the combination of technical implementation, gamification approach, and adherence to the principles of cyber hygiene opens up new horizons in the development of educational digital products focused on the needs of the modern generation.

2. Analysis of modern scientific works

In the context of growing threats in the digital environment, the scientific community in the field of information security and education is investigating the effectiveness of using gaming technologies, in particular visual storytelling, to increase the level of digital literacy and conscious behavior of users. Analysis of modern scientific works indicates a clear trend of integrating gamification and elements of cyber hygiene into the educational process.

Researchers Thompson, Melendez, Hempson Jones and Salvi (2022) [1] proposed the conceptual model RAD SIM (Reflection–Action–Design–Simulation), which combines behavioral mechanisms of gamification with a systemic approach to the development of educational simulations. This model is relevant in the context of creating a visual novel, as it allows for the implementation of a multi-level plot structure with adaptive scenarios for responding to potential threats.

Another group of researchers – Williams, Anthi, Cherdantseva and Javed (2024) [2] – investigated story-based CTF (Capture The Flag) as a tool for increasing students' motivation to learn the basics of cybersecurity. The authors point to the effectiveness of immersion in the plot through characters, dialogues, choices and consequences – elements inherent in a visual novel. This experience is relevant when creating interactive educational scenarios with variational logic, which are able to form critical thinking and attentiveness to digital risks. In the study of Jaffray, Finn and Nurse (2021) [3], the game SherLOCKED – a detective 2D novel that teaches the basics of cyber hygiene through a plot investigation. The authors emphasize the importance of emotional perception in building long-term skills of safe behavior. This approach confirms the feasibility of implementing educational games in the format of a visual novel, where the plot acts as the main didactic element.

Scholefield and Shepherd (2019) [4]present a mobile RPG game that introduces users to basic digital security concepts through interactive missions and a reward system. A gamified password survey has been shown to be effective in changing adolescent behavior. Even simple story structures implemented in an interactive format are effective in raising awareness of cyber security issues.

Of particular value to the topic of this paper is the project of a story-driven educational game about USB attacks, built on the Twine platform (2023) [5]. The game is a full-fledged visual novel with a non-linear narrative that simulates realistic threat scenarios and requires the player to make decisions that affect the outcome. This format allows you to recreate real-life situations in a safe environment, developing skills in responding to cyber risks.

It is also important to mention the review studies (2024) [6, 7, 8, 9] that systematize gamification methods in digital security education among non-IT professionals. They emphasize the importance of multimedia presentation, a versatile plot, and a minimal technical barrier, which coincides with the requirements for visual novels in an educational environment.
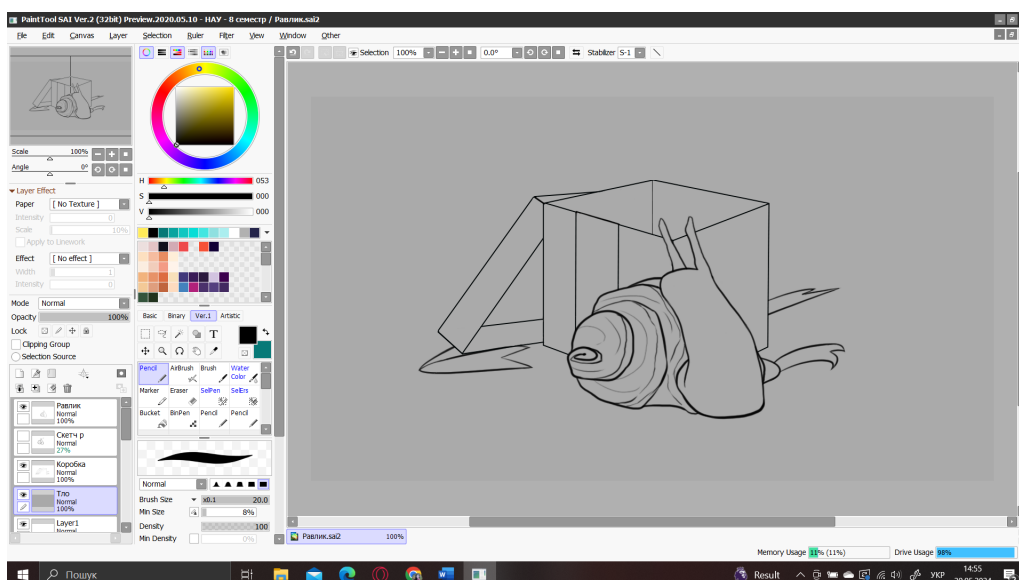
Figure 1: Easy Paint Tool SAI work panel (screenshot from the screen).

3. Graphic software for designing a visual novel narrative for studying an academic discipline

In the process of technical implementation of the visual novel educational resource, special attention is paid not only to the convenience and functionality of the software, but also to compliance with the requirements of cyber hygiene [10]. The fundamental goal of which is to adhere to the principles of safe use of digital tools, protect users' personal data, reduce the load on hardware, and form an informed attitude to digital risks.

A visual novel as a form of interactive storytelling should combine aesthetics, technical efficiency, and compliance with the principles of cyber hygiene. One of the key aspects is the creation of graphic content — illustrations, sprites, backgrounds — using proven and safe software Figure 1.

The user interface is designed with an emphasis on the nonlinearity of the plot, which is implemented through an interactive choice structure: the player independently chooses which section to start from, how to move between scenes, and what ending to get. This approach supports the principles of cyber hygiene — in particular, voluntary choice, no forced passage, and adaptability to the user's individual pace.

Graphics were created in the Easy Paint Tool SAI program, which meets the criteria of a safe, autonomous environment: it does not require an Internet connection, does not store metadata in the cloud, and provides local image processing. The toolkit allows you to create unified character sprites, backgrounds, and illustrations taking into account anime-style aesthetics and the principles of psychological comfort. Additional elements (facial expressions, transition animations, character color coding) help reduce cognitive load while reading.

Developed by the Japanese company SYSTEMAX Easy Paint Tool SAI, it is one of the most effective tools for artists working with raster graphics. Creating visual novels in this environment is explained by the combination of:

- a simple interface that lowers the threshold of entry for beginners and minimizes the need for online connection to cloud services, where user data is often stored;
- low system load, which is important in the context of ensuring digital inclusion and access to educational content on medium-power computers;
- offline functionality that eliminates the need for an Internet connection, which is an additional safeguard against phishing attacks or data leakage.
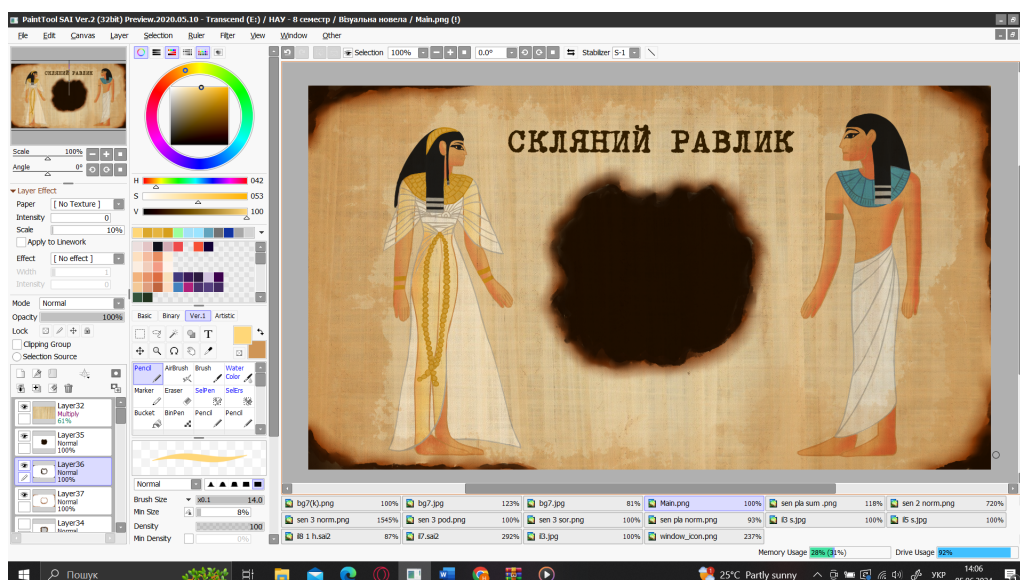
Figure 2: Creating an image for the main menu with the title Glass Snail Creating an image for the main menu with the title Glass Snail (screenshot from the screen).
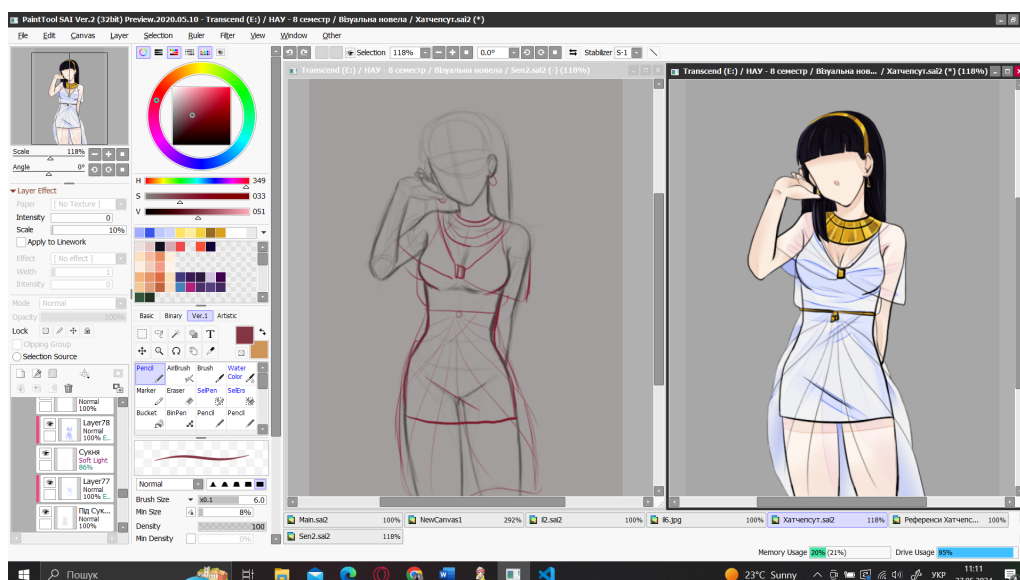

Figure 3: Creating a single character sprite (screenshot from the screen).

The program supports key graphic formats (.psd, .png, .bmp), which allows you to integrate its results into any engine without losing quality or layer structure of the image. Custom brushes, color mixing settings, and support for RGB and HSV models allow you to accurately and safely convey the intended visual style without violating the principles of digital ethics [11]. All illustrations — from character sprites to background scenes — are made in the anime style, which is well received by a youth audience and allows you to create an emotional connection with the characters Figure 2.

Several emotional states have been developed for each character, which dynamically change depending on the situation in the text. This contributes to the user's deeper immersion in the plot, allowing for a better understanding of the motives and mood of the characters. The illustrations are also supplemented with color clues - a background of the appropriate color, signaling whose point of view is currently presented Figure 3.

Thus, SAI is not only an artistic tool, but also a means of digital hygiene that ensures minimal interference in the user's environment and complete control over their own data.
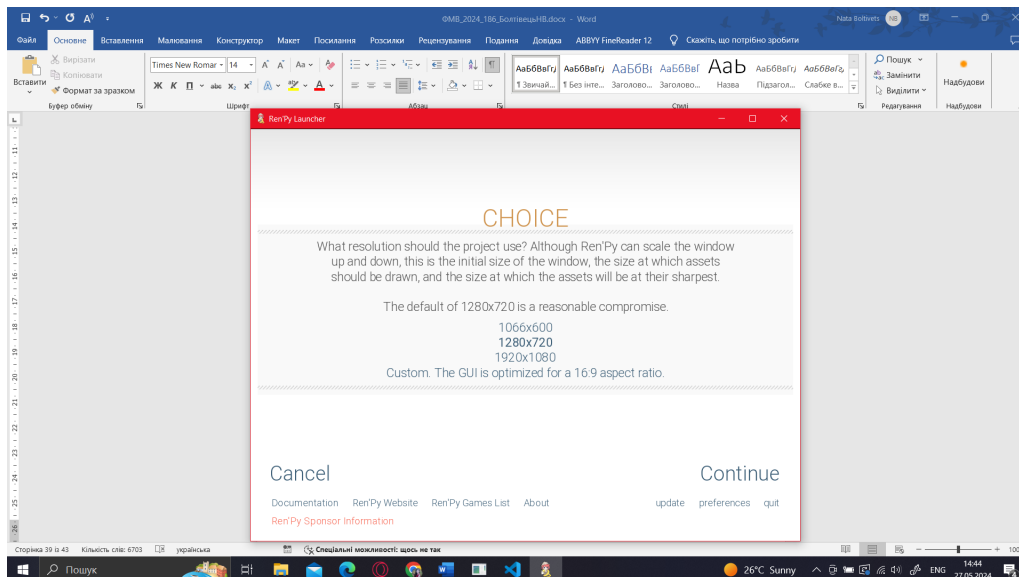
Figure 4: Practical implementation of the project in Ren'Py.

3.1. Visual novel builders: safe interactivity

The visual novel is assembled in the Ren'Py environment, an open-source engine that allows you to create a structured, secure educational product. An important advantage of Ren'Py is the lack of the need for online license verification or an account. All resources are localized, which reduces the risk of data leakage. Each section of the novel has its own label, and transitions between them are implemented using an intuitive selection menu that allows the user to choose a reading trajectory. This implements a non-linear structure that allows each reader to build their own trajectory of passage Figure 4.

In order to begin the process of writing and editing script code in the Ren'Py environment, the developer must first familiarize himself with the basic syntactic principles of this language. One of the key elements of code structuring is the use of indentation, which is of fundamental importance for the correct interpretation of instructions by the engine. Indentation allows you to combine commands into logical blocks that serve as the basic unit of program logic. As in Python, on which Ren'Py is based, violating indentation rules can lead to critical logical or syntactic errors in the program.

The Ren'Py script structure involves the use of the label operator, which defines a specific point in the code that the program can access. For example, the label start: construct means the beginning of the main script, which will be executed after the user clicks the "Start" button in the main menu.

Also among the basic instructions is the operator "say", which can be used in two formats:

1. In the form of narration - text enclosed in quotes, which conveys internal thoughts or descriptions of events.

2. In the form of dialogue - an expression, which indicates the name of the speaking character and his replica.

It is important that all code elements belonging to the same logical block have the same level of indentation - only in this case will the engine be able to correctly interpret the structure of the script.

To display visual content, the operator's scene (for background images) and show (for displaying character sprites) are used, where the corresponding image files are specified. The basis of a visual novel is the text of the story, which is entered using a speed-controlled output tag, for example: n "{cps=20}…{/cps}" The cps attribute (characters per second) determines the frequency of the gradual appearance of characters on the screen, which simulates the effect of close to "live" reading. This is how the entire text narrative of the novel is structured Figure 5.

Ren'Py software was used as the main engine due to its open architecture, Python support, local data processing and extensive script personalization capabilities. The choice of engine for implementing a visual novel is key to maintaining a balance between gamification, accessibility and cyber hygiene.

```
scene bg2 with fade
show hach 3 pal norm with dissolve
n "(cps=20)Працювала в другу змін, до пізньої
ночі.(/cps)".
n "(cps=20)Цього дня біля газетного кіоску,
що на розі, панна помітила вишуканого незнайм-
ця в зимовому пальті кольору черного лаку...
а другою з панновної лівої кишені пощупала річ,
яку там намацала.(/cps).
n "(cps=20)Продавець дав їй і панна зникла.
(/cps)"
```

Figure 5: Example of part of the code for the visual novel story Glass Snail (screenshot from the screen).

Among the available tools, such as Unity or TyranoBuilder, Ren'Py is the most optimal in this context, combining:

- open source and the absence of telemetry, which eliminates the risk of hidden data transmission;
- offline mode of operation, in which the user does not connect to the developer's servers during development, which is critical for maintaining privacy;
- accessibility with the presence of a multilingual interface and a large amount of documentation, which ensures equal access to the tool for all categories of users;
- Python support allows you to create not only simple dialogues, but also mini-games, which is ideal for educational modeling and practices of safe behavior in the digital environment.

Ren'Py also allows for the creation of complex scenarios, which allows for the simulation of situations related to online behavior choices, digital security, and critical thinking. This approach not only promotes gamification, but also builds practical cyber hygiene skills in an interactive way.

### 3.2. Code editor as a tool for secure development

The use of programming elements such as labels, animations, selection branches, and transition logic was implemented through the Visual Studio Code environment, which provides secure storage, functionality expansion through plugins, and integration with the engine without the need to connect to external servers.

Visual Studio Code, an open-source multifunctional editor, was appropriately chosen for editing the visual novel script. Among the advantages are:

- automatic saving and file backups, which prevents data loss;
- Python support with syntax highlighting and error checking, which minimizes code vulnerabilities;
- the ability to expand functionality by installing verified plugins from the official marketplace;
- secure local work without the need for registration or synchronization with the Microsoft cloud.

Thus, VS Code provides a stable and safe environment for creating and testing game logic, including educational scenarios related to Internet safety, online communication ethics, and information hygiene Figure 6.

Visual novels also use scene transition animation to make the transition between scenes less abrupt and more enjoyable for the reader, creating a smooth and natural flow of events. Also, properly selected animations can enhance the emotional perception of a scene. For example, a slow fade can add drama, and a fast transition can emphasize the tension of the moment. Rationally designed scene transitions increase the aesthetic quality of a visual novel and contribute to the impression of professionalism of the development.
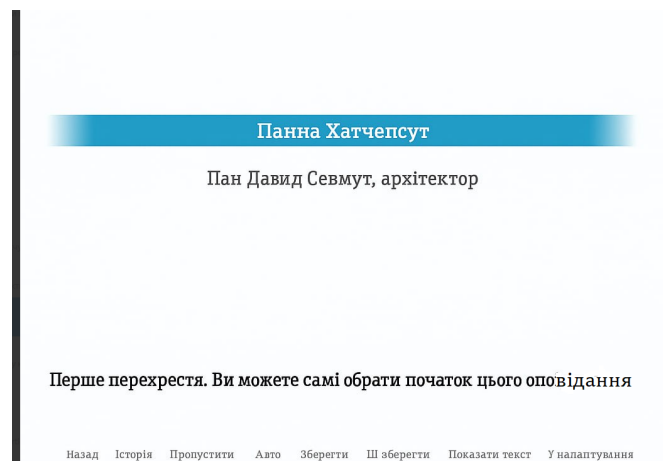
Figure 6: Appearance of selection buttons in a visual novel (screenshot from the screen).

To implement the animation of the gradual disappearance of a character sprite, a construction with the with operator is used, in which a specific type of animation effect is specified. The syntax is as follows:

$$hide\ hach\ with\ dissolve.$$

Since the novella scenario involves an interactive structure, it is advisable to implement a selection mechanism through interactive buttons. This approach gives the user the opportunity to make a choice that directly affects the further development of the plot, which attracts the reader and makes him an active participant in the story. Thanks to the ability to make different choices, the reader can play the game several times, exploring different scenarios and endings. Each reader can read the story and end it with the ending that he likes best, making decisions according to his preferences and values, which creates a unique experience for each player and increases the level of enjoyment of the game.

To create interactive buttons that will transfer the reader to a certain section, you need to write the keyword "menu" under it and write a question so that the reader understands what he needs to choose. Then write what will be written on the button and write a "jump" function on a new line, which will transfer the reader to the desired section.

It is at the stage of concept development that the overall style of the project, its structure, usability, interface, and compliance with modern digital security standards are determined. The entire process of designing the visual novella of the story "The Glass Snail" for the study of the academic discipline of foreign literature - from concept to implementation - was based on the principles of digital hygiene: a safe creation environment, ethical display of content, inclusivity for the target audience, user control over interaction, and adherence to usability standards. This allows you to create a product that not only interests with aesthetics, but also cultivates a responsible attitude towards digital content [12].

At the final stage of creating a visual novel, a distribution is assembled from locally stored files, which prevents unauthorized access to the source code or graphic resources. This corresponds to the principles of cyber hygiene: data control, software environment autonomy, minimization of the user's digital footprint Figure 7.

Thus, the concept of design within the framework of creating a visual novel is considered as a multidimensional category that combines aesthetic design, technical implementation, interface logic and digital security.

This approach contributes not only to effective learning, but also to the formation of digital responsibility in students. In order to prevent malicious actions, it was necessary to implement the following procedural measures when developing a visual novel:

- to prevent modifications of the novel code by third parties, file integrity checks were integrated,
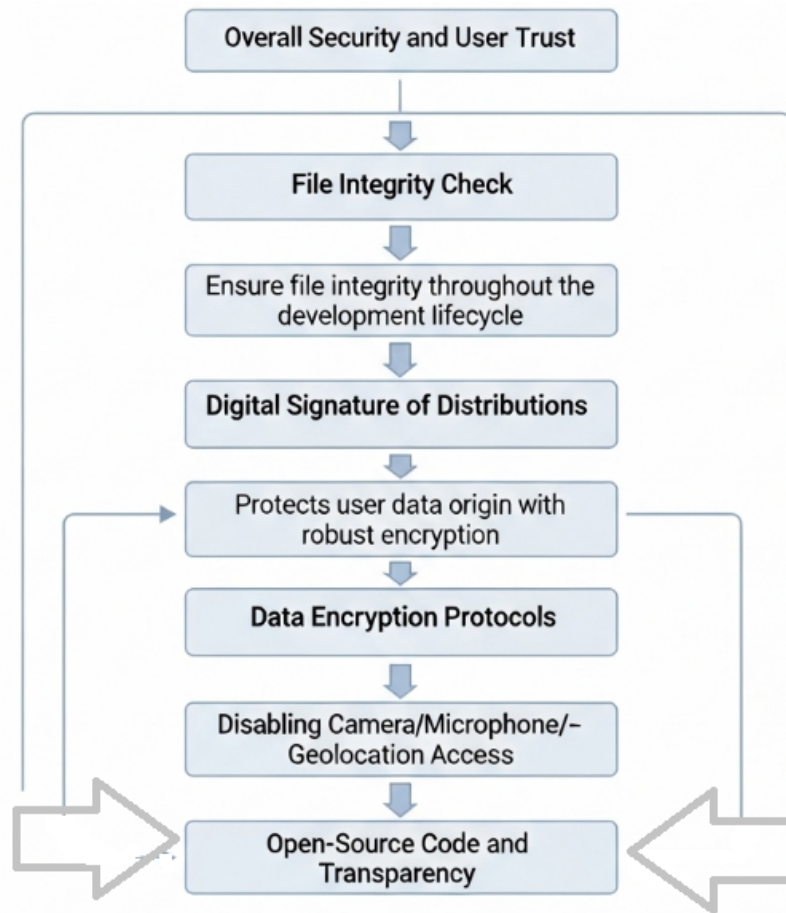
Figure 7: A multidimensional category, combining aesthetic design, technical implementation, interface logic, and digital security. Scheme of the proposed categorical system.

and distributions were digitally signed, which will help users make sure that the novel has not been modified after creation [12];

- encryption protocols were used to ensure that data is stored or processed in any way, even locally;
- the graphics engine was configured in such a way as not to require access to the camera, microphone, or geolocation;
- to increase user trust and the possibility of independent auditing [13], code transparency software and open-source code publication were selected.

Also, to avoid unwanted data collection, analytics and third-party SDKs were not used. In addition, regular software updates are a mandatory software requirement to eliminate potential vulnerabilities. Let's take a closer look at the software tools that were used in the design of the visual novel, which is considered a multidimensional category that combines aesthetic design, technical implementation, interface logic and digital security in countering threats from hackers Table 1.

In the development of the visual novel, an algorithm developed by the authors was implemented, which combines the generation of a pseudonymized unique user identifier, symmetric encryption of interaction files, as well as local storage of encrypted information without the use of external servers Figure 8.

In this way, it is possible to adhere to the principles of privacy-by-design and data minimization recommended by international information security standards (in particular, ISO/IEC 27001, OWASP Secure Coding Practices) [14, 15, 16].

Based on the requirements and framework of standardized concepts and processes reflected in these standards. Description of the algorithm for protecting user personal data in a visual novel includes following functions:

Table 1

Technical Concept in Multidimensional Categories of Information Security of the Visual Novel

Technology name	Function	Action
Obfuscation Anti-tamper Denuvo	protection of game code and resources	obfuscation of the code, making it difficult to read and reverse engineer game scripts make it difficult to make unauthorized changes to the game
Microsoft SignTool OpenSSL	digital signature and file validation	checks content integrity at startup
OAuth, JWT, OpenID Connect Steamworks API, Epic Online Services SDK	authentication and authorization	used for secure player login platforms with built-in security features for authorization
Easy Anti-Cheat, BattleEye, VAC (Steam) Detours, Hook Detection	fraud prevention	a program was used to detect third-party interference, modifications, or bots a tool has been implemented to monitor the connection of unauthorized DLLs or changes in memory
TLS/SSL		data transmitted between the client and the server is encrypted
VPN for testing	encryption and network protection	used to encrypt data transmitted between the client and server for a secure connection during development
Packet validation	protection of game assets (graphics, sound, text)	the correctness of the data on the server has been checked (protection against packet injection)
Ren'Py .rpa CRC Sandboxing	file verification safe game update	implemented control of changes in game files signed patches and hashing of update files (SHA-256) launching new components in a restricted environment

1. Initialization - starting the game:

- Verifying the game's digital signature;
- Loading local settings without network access;
- Displaying a privacy policy message to the user (if the game saves progress).

2. User login - optional:

- If you need to save your progress, create an anonymized profile (generate a random identifier) UUID);
- Setting local encryption for save files (e.g.AES-256).

3. During user interaction:

- Save changes locally without transmitting any information over the network;
- No access to microphone, camera, geolocation, contacts or other sensitive information API;
- Prohibition of inserting third-party scripts.

4. During dialogues, section selection, and interactive:

- Each action is recorded in a local log file, which is encrypted;
- When updating the status (for example, completing a section), the program checks the integrity of the data (hash sum).
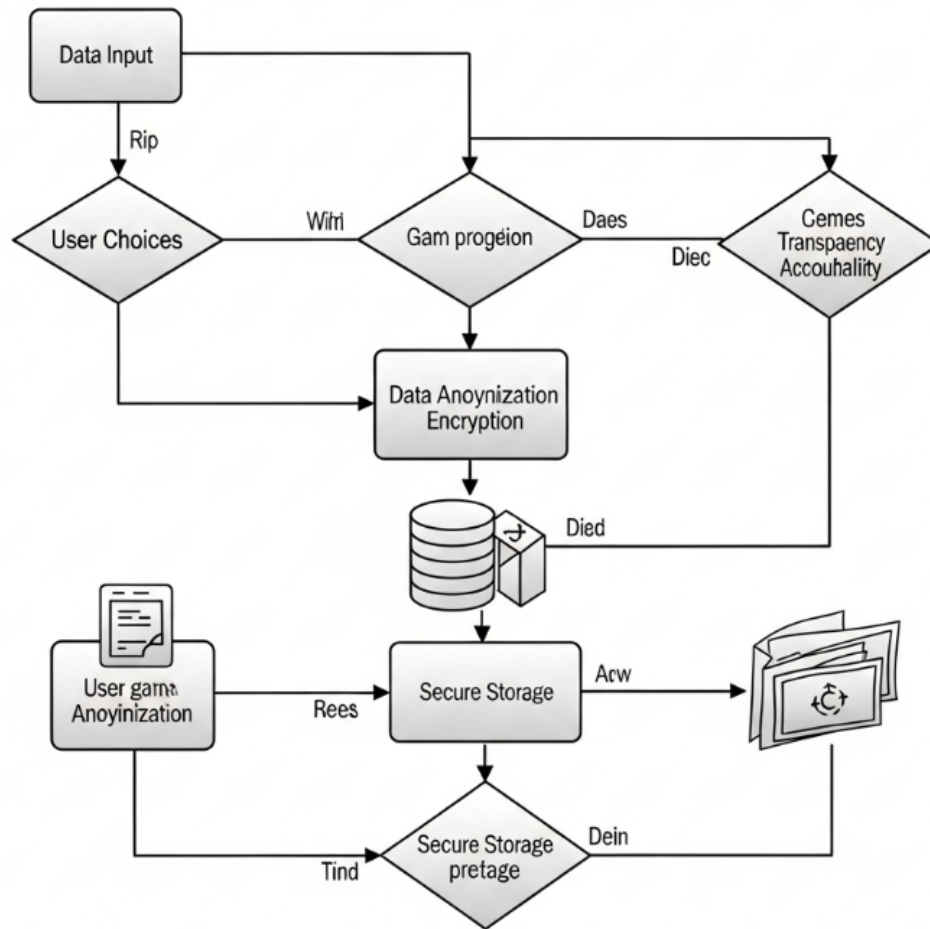
Figure 8: Algorithm of the architecture of the system for processing users' personal data.

5. Saving progress:

- Automatic data storage in encrypted form;
- The data does not contain name, email, IP or personal information.

6. Ending the game or exiting:

- Offer to save/delete profile;
- Automatic cleaning of temporary files, cache, logs.

7. Additional measures:

- Restricting access to game files from outside (read-only attributes);
- Built-in modification check (code/file checksum comparison);
- Support for the "Guest Mode" feature - without any data recording.

Thus, this algorithm for the technical implementation of personal data protection in a visual novel, taking into account the requirements of cyber hygiene, ensures the security and protection of the user's personal data, which are critically important aspects in the development of educational visual novels, especially in cases where the target audience is minor users. To achieve the goals, a technical model of local data processing with encryption is used Figure 9.
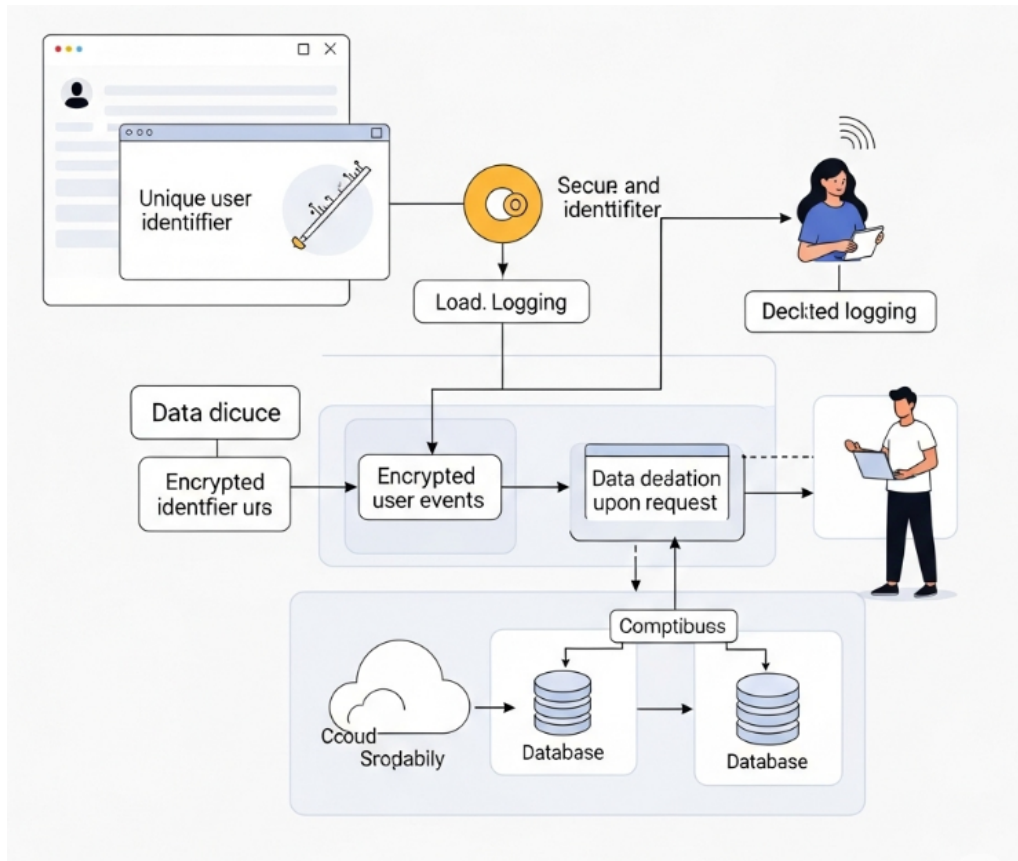
Figure 9: Our own model of a local data processing system with encryption.

This model is implemented using the capabilities of the Ren'Py engine in combination with modules of the Python programming language. Table 2 provides a detailed analysis of the model for protecting personal data and ensuring ethical interaction between a digital product and a user.

Thus, it can be stated that this model demonstrates that even in the conditions of simple development tools in the Ren'Py environment it is possible to implement reliable protection of personal data and ensure ethical interaction between the digital product and the user. In turn, it contributes to the formation of trust in educational games and the development of a conscious attitude to security in the information environment.

3.3. Testing the reliability and information security of a visual novel

In today's digital environment, the effectiveness of educational software is determined not only by its functionality and interface attractiveness, but also by the level of information security. This is especially true for programs that interact with personal or behavioral data of users, in particular in the educational context, where the target audience is often minors. According to international standards (ISO/IEC 27001:2022) and recommendations for secure software development (OWASP Secure Coding Practices, NIST SP 800 53), data protection should be taken into account at the architectural design stage and verified through comprehensive testing.

The visual novel, implemented as a means of gamified learning, integrates local mechanisms for storing user data using symmetric encryption (AES CFB algorithm), generating a unique identifier based on UUID, and logging interactions without using network interfaces. Thus, the system implements the basic principles of secure-by-design and privacy-by-default, recommended by European and international cyber regulation.

The purpose of the testing is to verify the compliance of the implemented functionality with data protection requirements, prevent unauthorized access, preserve information integrity, and guarantee

Table 2

Description of the Model for Protecting Personal Data and Ensuring Ethical Interaction between Digital Product and User

Operation	Description
Generation and encryption of a unique user ID	After the first launch, the program checks for the presence of a local file containing an encrypted user ID (user_id.sec). If such a file does not exist, the system generates a unique identifier in UUIDv4 format, after which the data is encrypted using the symmetric AES (Advanced Encryption Standard) algorithm in CFB (Cipher Feedback) mode. Base64 encoding is used to save encrypted data in text format
Downloading and using the identifier	On subsequent launches of the game, if the user_id.sec file exists, the program decrypts it and reads the identifier, storing it in the user_id variable. This identifier can be used for internal user accounting or personalization of the interaction experience without collecting personal information
Encrypted logging of user events	Each event in the game that requires registration (for example, player selection, passing a certain plot branch) is recorded in the form of a text message. This message is also encrypted using the same AES-CFB method and stored in the event_log.sec file. This approach allows for secure logging of user behavior without transmitting data to external servers
Deletion of data at the user's request	The data cleaning mechanism is implemented as the clear_user_data function, which allows to delete both the user ID file (user_id.sec) and the event log (event_log.sec). This element complies with the principle of digital autonomy and the user's right to delete information about themselves in accordance with ethical and legal standards
Technological basis and compatibility	The model implementation is based on the built-in Python environment Ren'Py. To work with encryption, it is necessary to connect an external library pycryptodome, which supports the necessary cryptographic standards and is compatible with the engine. All data processing is carried out exclusively locally, which minimizes the risks of information leakage and meets the key principles of cyber hygiene in education.

fault tolerance and scenario consistency of all components. Among the aspects to be checked:

- correct implementation of cryptographic transformations, including encryption and decryption;
- resistance of UUID generation to repetitions and collisions;
- consistency of event logging with data minimization requirements;
- protection against local unauthorized access to encrypted files;
- effectiveness of implementation of the personal data cleaning mechanism (right to be forgotten);
- stability of execution during long game sessions and data reloading.

Thus, testing was carried out not only to identify technical errors, but also to take into account ethical and legal obligations regarding digital hygiene and a safe educational environment. Table 3 shows the compliance of the visual novel components with the principles of secure development (Secure Coding Practices) based on the recommendations of OWASP, NIST SP 800-53 and ISO/IEC 27001.

In the context of the implementation of digital educational products, the issue of ensuring stable and secure operation of software is of particular relevance. In particular, we can emphasize the use of interactive systems that operate in a sensitive environment - for example, when interacting with minor users or within educational institutions. Among the key challenges are the protection of personal data, prevention of information loss, avoidance of unauthorized access and failures in operation.

Taking into account the above, at the stage after the completion of the technical implementation of the visual novel, testing was conducted aimed at checking its robustness - that is, resistance to failures, the ability to maintain data integrity, meet confidentiality requirements and demonstrate proper functional security. This approach corresponds to modern practices for developing secure software, in particular, the requirements for educational digital solutions that implement the principles of cyber hygiene.

Table 3
Table of Compliance of Visual Novel Components with Secure Coding Requirements

Game component	Secure coding practice	Compliance/Implementation
User UUID generation	Avoid predictable identifiers (OWASP)	Applied UUID4() – cryptographically secure random ID generation
Saving user_id.sec	Data at rest encryption (ISO 27001, NIST SC-12)	The file is encrypted using AES-CFB, saving in Base64
Encrypting/decrypting event logs	Sensitive data protection (OWASP A2, ISO A.10.1)	All interactions are recorded in event_log.sec in encrypted form
Local data storage	No external storage without consent	All processing happens locally, without third-party APIs or remote servers
Data cleaning	Right to be forgotten (GDPR, NIST AR-4)	Function implemented clear_user_data() complete deletion of sensitive files
Checking access to resources	Limit attack surface (OWASP, ISO A.9.1.2)	No access to microphone, camera, contacts, geolocation, or accounts
File integrity check	File integrity verification (NIST SI-7)	Not implemented, it is recommended to add check hashes at startup in the future
Data minimization	Data minimization (ISO 27001, NIST AP-1)	No personal user data; only pseudonymized data is stored UUID
Ease of code auditing	Readable and auditable code (OWASP SDLC)	The code is structured, all actions are performed through openly defined functions
Exception/errors	Fail securely (OWASP)	Unhandled exceptions are potentially dangerous - it is recommended to add try-except blocks for safe fallback

The testing covered functional and non-functional aspects, in particular: the performance of cryptographic modules, the correctness of the generation and processing of a unique user identifier, the stability of the novella scripts and the storage of interaction logs in encrypted form. The testing results are presented below.

In order to verify the correctness of the functioning, resistance to failures and compliance with information security requirements, a comprehensive testing of the visual novel for studying the story "The Glass Snail" of the academic discipline of foreign literature was carried out, implemented using the Ren'Py engine and supplemented with a data encryption module based on the library PyCryptodome (AES-CFB). The main attention was paid to the evaluation of mechanisms for storing and processing user data in a local environment without the use of network services. To verify the integrity of the encryption and decryption process, a series of module tests was carried out, aimed at verifying the preservation of the authenticity of text data after its encryption and subsequent decryption.

Testing the reliability and information security of a visual novel includes:

1. Encryption and decryption testing. Checking whether data encrypted at rest can be recovered without loss:

- # Test: Does the original text match the decrypted text?;
- original = Test message;
- encrypted = encrypt(original);
- decrypted = decrypt(encrypted);
- assert original == decrypted, "- Error encryption: × the decrypted text does not match the original";
- print(" v Encryption/decryption works correctly").

2. UUID generation test. Checking that a valid unique identifier is generated:

- user_id = str(uuid.uuid4());
- assert len(user_id) == 36 and user_id.count("-") == 4, "× Invalid format UUID";
- print("v UUID created successfully").

3. File saving test. Checking how a file is created, writes data, and read.

- # File path;
- path = "test_data.sec";
- test_text = " Test recording ";
- # Save;
- with open(path, "wb") as f:;
- f.write(encrypt(test_text));
- # Read with open(path, "rb") as f:;
- read_text = decrypt(f.read());
- assert test_text == read_text, "× Data not read correctly ";
- print("v Data is written and read without loss").

4. Data cleaning test. Checking how it works clear_user_data() completely deletes sensitive files.

- # Create test files;
- with open("user_id.sec", "wb") as f:;
- f.write(b"test") with open("event_log.sec", "wb") as f:;
- f.write(b"test");
- # Clear;
- clear_user_data();
- # Verify;
- assert not os.path.exists("user_id.sec") and not os.path.exists("event_log.sec"), "× Data not deleted ";
- print("v Files cleaned successfully ").

5. Visual novel stability test. Verifying those multiple iterations of launch, selection, and exit pass without errors in the logs

- Run - Select - Save - Exit - Rerun;
- Check that: ID is the same; events are logged; no errors Traceback;
- Automation tools (optional);
- pytest - if you move the test logic to a separate -.py file;
- Ren'Py console - using Shift+O to test functions while playing;
- Ren'Py logs (log.txt) - for testing calls.

The results confirmed that the use of the symmetric AES algorithm in CFB mode provides accurate content recovery, provided that the key and initialization vector are preserved Table 4.

Validation of the unique user identifier was implemented in testing the UUID generation process, which proved the compliance of the generated value with the RFC 4122 standard. The identifier was created at the first launch and stored in encrypted form, ensuring user anonymity and the impossibility of personal identification.

The presence, creation and correct reading of encrypted files from the local file system was checked, which was reflected in the file input/output testing. No cases of data loss or corruption were recorded. A separate cleaning test was performed - the function to delete confidential files (user_id.sec, event_log.sec) was successfully executed with each call, which indicates compliance with the principles of digital hygiene.

Table 4

Description of the Result of Validation of the Unique User Identifier Implemented in the Generation Process Testing UUID

Indicator	Description	Expected result	Verification method and Comments
Uniqueness UUID	No duplicates among generated UUID	+ 100% uniqueness of all generated UUID in the test sample	Generate a significant number of UUIDs (e.g. 1 million+) and check for duplicates.
Format UUID	Conformity of generated UUIDs to the standard UUID	+ All generated UUIDs +comply with the specification	Using regular expressions or built-in validation functions UUID
Randomness Unpredictability UUID	The difficulty of predicting the next generated UUID	+ Lack of discernible patterns or predictable sequences	Statistical analysis of the distribution of generated UUIDs, randomness tests
Generation productivity	The time required to generate a certain amount UUID	+ Generation occurs in an acceptable time that meets the system requirements	N UUID generation time measurement and boot analysis CPU/RAM
Validation reliability	The ability of the system to correctly identify valid and invalid UUID	+ The system correctly validates all valid UUID. The system rejects all invalid (incorrectly formatted) entries	Testing the validator on a set of real and intentionally distorted UUID
Uniqueness UUID	No duplicates among generated UUID	+ 100% uniqueness of all generated UUIDs in the test sample	Generate a significant number of UUIDs (e.g. 1 million+) and check for duplicates
Format UUID	Conformity of generated UUIDs to the standard UUID (RFC 4122)	+ All generated UUIDs comply with the specification	Using regular expressions or built-in validation functions UUID

Scenario stability testing was performed by emulating a typical user scenario: multiple transitions between sections of a visual novel, repeated launches, saving progress, checking the display of dynamic content. All operations were performed stably, without critical errors. Information about key user actions was correctly stored in encrypted form.

As a result of testing, it can be summarized that the implemented mechanisms of encryption, event logging, generation of a unique identifier and data cleaning meet the requirements for reliable information systems that operate in an environment with limited access to the network. Such a model of technical implementation is relevant for use in the field of education, in particular in the context of gamified learning, where the priority is the protection of personal data of education seekers and ensuring a safe digital space.

4. Conclusions

As a result of a comprehensive analysis of the possibilities of the visual novel as an innovative means of gamification of the educational process, it was proven that this format of the educational direction contributes not only to increasing the motivation of students, but also allows integrating the components of pedagogy, information security and digital ethics into a single interactive model. The creation of a visual novel is considered as a complex process that covers technical, didactic and cyber hygiene aspects - from software modeling to ensuring the protection of user data.

Particular attention is paid to an in-depth analysis of the role of cyber hygiene as a critically important component of a secure digital environment. The work emphasizes the need to adhere to such principles as: protection of personal data, avoidance of surveillance, implementation of secure login mechanisms,

rejection of manipulative content and the formation of a culture of critical perception of information. The technical implementation used Ren'Py, Easy Paint Tool SAI, Unity and Visual Studio Code tools, and also implemented encryption mechanisms (AES-CFB), UUID generation, logging of user actions and cleaning of traces of digital activity.

The testing confirmed the stability and reliability of the implemented solution, its compliance with secure coding recommendations (OWASP, NIST, ISO/IEC 27001), as well as the stability of functioning during multiple user interactions. The constructed table of compliance of game components with the principles of secure programming illustrates the practical application of theoretical guidelines of cyber hygiene in educational content.

A visual novel based on the work "The Glass Snail" demonstrates an example of modeling a digital environment that not only visualizes educational material, but also develops a conscious attitude to security in the information space. The applied methods can be adapted in educational practice by teachers of digital disciplines, cybersecurity specialists, educational software developers and other participants in the digital transformation of education.

Thus, the results of the study are a relevant contribution to the development of new generation gamified educational technologies based on a combination of interactivity, content and digital security.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] L. A. Thompson, N. Melendez, J. Hempson Jones, F. Salvi, Gamification in cybersecurity education: The RAD SIM framework for effective learning, https://cisse.info, 2022.

[2] L. Williams, E. Anthi, Y. Cherdantseva, A. Javed, Leveraging gamification and game-based learning in cybersecurity education, https://cisse.info, 2024.

[3] A. Jaffray, C. Finn, J. R. C. Nurse, SherLOCKED: A detective themed serious game for cyber security education, Journal of Information Security Applications (2021).

[4] S. Scholefield, L. A. Shepherd, Gamification techniques for raising cyber security awareness, in: Lecture Notes in Computer Science (LNCS), 2019.

[5] K. Hilliard, X. Yuan, et al., Using gamification to enhance mastery of network security concepts, https://dl.acm.org, 2024.

[6] M. Coenraad, et al., Experiencing cybersecurity one game at a time, Journal of Information Security Education (2020).

[7] V. Rikkers, D. K. Sarmah, A story driven gamified education on USB based attack, 2023.

[8] A. K. Gwenhure, F. S. Rahayu, Gamification of cybersecurity awareness for non-IT professionals, International Journal of Serious Games (2024).

[9] D. Pramod, Gamification in cybersecurity education: A state of the art review, Emerald Insight (2024).

[10] O. V. Matviichuk-Yudina, The concept of forming professional competencies of specialists in information technologies and cybersecurity, Naukoiemni tekhnolohii [Science-intensive Technologies] (2019) 330–342.

[11] O. Bobarchuk, O. Briahin, S. Halchenko, R. Berdibayev, et al., Protection of primary tangible media that can be used as evidence against cyberattacks, in: CHCMiGIN'24: Third International Conference on Cyber Hygiene and Conflict Management in Global Information Networks, Kyiv, Ukraine, 2024, pp. 215–228. URL: https://ceur-ws.org/Vol-3925/paper18.pdf.

[12] O. V. Matviichuk-Yudina, S. M. Loboda, Project method in forming infographic competence in future bachelors of cybersecurity, Problemy inshenerno-pedahohichnoi osvity [Problems of Engineering and Pedagogical Education] (2017) 269–277.

[13] O. A. Bobarchuk, S. M. Halchenko, S. O. Hnidenko, I. P. Zavadetskyi, Theoretical and practical aspects of using artificial intelligence technologies in the field of sound design, in: Proceedings of the 7th International Workshop on Augmented Reality in Education (AREdu 2024), Kryvyi Rih, Ukraine, 2024, pp. 319–328. URL: https://ceur-ws.org/Vol-3918/paper350.pdf.

[14] OWASP Foundation, OWASP secure coding practices – Quick reference guide, https://owasp.org, 2023.

[15] National Institute of Standards and Technology (NIST), Security and privacy controls for information systems and organizations, Technical Report NIST SP 800-53 Rev. 5, NIST, 2020. URL: https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/final.

[16] International Organization for Standardization, ISO/IEC 27001:2022 – Information security, cybersecurity and privacy protection – Information security management systems, Geneva: ISO, 2022.