

**NATIONAL UNIVERSITY OF SCIENCE AND TECHNOLOGY
POLITEHNICA BUCHAREST
FACULTY OF ENGINEERING IN FOREIGN LANGUAGES
COMPUTERS AND INFORMATION TECHNOLOGY -INFORMATION ENGINEERING**

DIPLOMA PROJECT

**Project coordinator
S.I. dr. ing. Iulia-Cristina STĂNICĂ**

**Student:
Mihai-Alexandru ANDREI**

**Bucharest
2025**



**NATIONAL UNIVERSITY OF SCIENCE AND TECHNOLOGY
POLITEHNICA BUCHAREST
FACULTY OF ENGINEERING IN FOREIGN LANGUAGES
COMPUTERS AND INFORMATION TECHNOLOGY -
INFORMATION ENGINEERING**



Gamified Museum Experience for Learning History

Project coordinator
S.I. dr. ing. Iulia-Cristina STĂNICĂ

Student:
Mihai-Alexandru ANDREI

**Bucharest
2025**



Universitatea Națională de Știință și Tehnologie
POLITEHNICA București
Facultatea de Inginerie în Limbi Străine
National University of Science and Technology
POLITEHNICA Bucharest
Faculty of Engineering in Foreign Languages



Comisie pentru specializare
Specialization commission

Decan
Dean

Prof. dr. DASCĂLU Maria-Iuliana

TEMĂ PROIECT DE DIPLOMĂ DIPLOMA PROJECT THEME

Subsemnatul, voi pregăti lucrarea de diplomă cu titlul:
Experiență Muzeală Gamificată pentru Învățarea Istoriei

domeniul Calculatoare și tehnologia informației
specializarea Ingineria informației
pentru sesiunea 2025 Iulie

I, the undersigned, will prepare the diploma thesis with the title:

Gamified Museum Experience for Learning History

field Computers and information technology
specialization Information engineering
for the 2025 July session.

**Profesor îndrumător
Project coordinator**

S.L. STĂNICĂ Iulia

**Student
Student**

ANDREI Mihai-Alexandru

Academic Honesty Statement

I, Mihai-Alexandru ANDREI, hereby declare that the work with the title Gamified Museum Experience for Learning History, to be openly defended in front of the diploma theses examination commission at the Faculty of Engineering in Foreign Languages, National University of Science and Technology "Politehnica" of Bucharest, as partial requirement for obtaining the title of Engineer is the result of my own work, based on my work.

The thesis, simulations, experiments and measurements that are presented are made entirely by me under the guidance of the scientific adviser, without the implication of persons that are not cited by name and contribution in the Acknowledgements part.

The thesis has never been presented to a higher education institution or research board in the country or abroad.

All the information used, including the Internet, is obtained from sources that were cited and indicated in the notes and in the bibliography, according to ethical standards. I understand that plagiarism is an offense and is punishable under law.

The results from the simulations, experiments and measurements are genuine. I understand that the falsification of data and results constitutes fraud and is punished according to regulations.

Mihai-Alexandru ANDREI

Table of Contents

1.	Introduction.....	1
1.1	Introduction.....	1
1.2	Motivation and Objectives.....	1
1.3	Structure of the paper.....	2
2.	Presentation of Domain.....	4
2.1	Background.....	4
2.1.1	The evolution of educational games.....	4
2.1.2	The role of museums in education.....	5
2.2	Importance of Learning History.....	5
2.3	Theoretical Foundations of Game-Based Learning.....	6
2.4	Engaging with History Through Gamification.....	9
2.5	Similar Applications.....	10
2.5.1	Similar applications in turn-based RPGs.....	10
2.5.2	Similar applications in historical games.....	12
2.5.3	Similar applications in museum-based games.....	13
3.	Research Methodology and User Survey.....	14
3.1	Research Methodology.....	14
3.2	User Survey.....	14
3.2.1	Survey – general information.....	14
3.2.2	Survey – learning preferences.....	16
3.2.3	Survey – museum visits.....	19
3.2.4	Survey – virtual museums and gamified learning.....	20
3.2.5	Survey – feedback on gamified learning.....	22
4.	Gameplay and Player Flow.....	25
5.	Game Design and Development.....	30
5.1	Overworld Side-Scroller.....	30
5.2	Turn-Based Combat.....	31
5.3	Assets and Animations.....	33
5.4	Sound.....	35
5.4	Dynamic Difficulty.....	35
6.	Technologies and Tools.....	37
6.1	Unity Engine.....	37
6.2	Visual Studio.....	39

6.3 Unity Hub and Plastic SCM Version Control.....	39
6.4 Other Game-Making Platforms.....	41
7. Conclusions.....	43
Bibliography	44
Appendix 1	45

List of Tables and Figures

Figure 2.1 – Bird’s eye view of a room in The VR Museum of Fine Art.....	5
Figure 2.2 – Answering question during a schoolday in Persona 5.....	10
Figure 2.3 – Positive party relationships in Darkest Dungeon 2..	11
Figure 2.4 – Rural life in 15 th century Europe as seen in Kingdom Come Deliverance 2.....	12
Figure 2.5 – Building exhibits in Mondo Museum.....	13
Figure 3.1 – Survey question: “What is your age?”.....	15
Figure 3.2 - Survey question: “What is your gender?”.....	15
Figure 3.3 – Survey question: “How familiar are you with History?”.	16
Figure 3.4 – Survey question: “How do you prefer to learn?”..	16
Figure 3.5 – Survey question: “How do you feel about using technology to learn?”.....	17
Figure 3.6 – Survey question: “Would you consider using a video game as a way to learn about history?”.....	17
Figure 3.9 – Survey question: “How much do you enjoy learning through video games or gamified experiences?”.....	18
Figure 3.9 – Survey question: “What type of game mechanics do you enjoy the most in educational games?”.	18
Figure 3.9 – Survey question: “What type of museums do you enjoy visiting the most?”.	19
Figure 3.10 – Survey question: “Do you typically enjoy reading exhibit labels or plaques when visiting a museum?”.....	19
Figure 3.11 – Survey question: “Would you consider visiting a museum more often if it had interactive digital exhibits (e.g. virtual tours, AR experiences)?”.....	20
Figure 3.12 – Survey question: “How likely would you be to try a virtual museum experience from home (e.g. interactive exhibits, virtual tours)?”.....	20
Figure 3.13 – Survey question: “Would you prefer a physical museum experience over a virtual one?”.....	21
Figure 3.14 – Survey question: “How interested are you in the idea of using a game-based approach to explore historical events (e.g. a turn-based RPG set in a historical museum)?”..	21
Figure 3.15 – Survey question: “Would you be more likely to engage with historical content in a game that incorporates elements like quests, rewards, and interactive narratives?”.	22
Figure 3.16 – Survey question: “What do you think are the most important aspects of a gamified learning experience?”..	23
Figure 3.17 – Survey question: “What would make you more likely to engage in a virtual gamified historical experience (e.g. a museum RPG)?”.....	23
Figure 3.18 – Survey question: “What art styles do you prefer for educational games?”.....	24
Figure 4.1 – Screenshot of main menu showcasing all buttons.....	25
Figure 4.2 – Screenshot of the first quiz gate the player encounters.....	27
Figure 4.3 – Screenshot of the player attacking the enemy knight in the turn-based combat.	28

Figure 5.1 – Question gate sequence diagram..	31
Figure 5.2 – Turn-based combat sequence diagram.	33
Figure 5.3 – Screenshot of Animator setup for player character..	35
Figure 6.1 – A screenshot of the Unity UI in the project.....	38
Figure 6.2 – A screenshot of the Unity Hub UI and the current projects on my device.....	40

List of Abbreviations

Abbreviation	Definition
SotA	State of the Art
RPG	Role Playing Game
GBL	Game-Based Learning
VR	Virtual Reality
AR	Augmented Reality
HP	Health Points
AAA (Triple-A)	a buzzword used to classify video games produced or distributed by mid-sized or major publishers, which typically have higher development and marketing budgets than other tiers of games
UI	User Interface
MP3	MPEG-1 Audio Layer III
FPS	Frames Per Second
SFX	Sound Effects
KCD2	Kindgom Come: Deliverance 2

1. Introduction

1.1. Introduction

EDUCATION is constantly evolving, and technology is playing a big part in making the learning process more fun and accessible. Conventional methods of teaching history do not tend to stimulate the interest of the students, and it is hard for them to become involved with historical events and ideas. The aim of this project is to bring together the past and the future, history and modern learning to make the whole process more interactive and engaging. Through its gamified design, this project aims to make the learning process of history an immersive one where players would experience history firsthand just like in a museum tour, but from the comfort of their homes.

1.2. Motivation and Objectives

Ever since childhood, I have had a fascination with both video games and history, be it recent or ancient. With this project I found a chance to combine these two passions of mine into an experience I wish I could have had as a child. I have always believed the way history is handled in schools is boring and uninteresting, driving away students from what is a diverse subject full of wonders and discoveries. I could easily understand how even someone keen on the subject might be driven away from it if their only activity is memorizing dates and writing paragraphs of names. My experience was similar, however in my own free time I pursued learning in a more practical way, be it books full of illustrations, charts and examples, documentaries exploring certain cultures in depth and, my personal favorite, museum visits.

Museums are, by design, interactive experiences where customers can immerse themselves in their subject of choice. Unlike textbooks, which only provide descriptions, museums allow visitors to explore exhibits and see real, authentic items up close. Historical Museums, most of all, offer such an experience allowing people to view different artifacts from many time periods. Some are in pristine shape, others show the age they have endured, but nonetheless they all show people what life was like in the past. By exploring the halls of a museum, you become part of the time period presented, immersing yourself in times long gone. Unfortunately, not everyone has access to such an enjoyable way to learn, be it from financial problems, huge distances or even health problems, but nonetheless I believe they deserve to experience it, even if they can't physically reach one.

Here is where my passion for games come in: by taking the innovative idea of a museum, with the interactive environment and informational plaques, we can create a similar environment, but digital. I find that games are quite helpful in the learning process. The instant feedback and reward systems make the whole experience engaging in a way that is rarely replicated in education. By keeping student focused and awarding them palpable progress for each action they make it helps them more easily absorb information and stops the mind from wondering off to other

subjects. By holding the attention of the player, it makes most of what is learned be remembered for longer periods of time by associating it with certain events or challenges in the game.

My objective is to create a gamified historical learning experience by using museums as a template. Gamification will further amplify user engagement by implementing game mechanics such as achievements, quests, progress tracking and reward systems. I hope to make the whole process of learning more accessible and intuitive by designing a virtual museum complete with everything you would find in a real museum, from exhibits, artifacts. By virtue of it being a game, I can also implement things that would otherwise be impossible in real life, like certain creatures or humans from history that come to life or actually usable ancient weaponry. The player will have to not only learn, but use the knowledge they gain to defeat certain enemies or solve interactive puzzles. My hope is that such an experience is unique and memorable enough to help people find a passion for this amazing subject, while also allowing them to retain information for longer periods of time through game-like experiences.

1.3. Structure of the Paper

This paper is organized into 7 main chapters, each presenting a specific aspect of the project: *Presentation of Domain, Research Methodology and User Survey, Project Planning, Game Design and Development, Technologies and Tools, and the Conclusions.*

Chapter 2, “Presentation of Domain”, establishes the theoretical and practical context for the gamified museum experience. First, it reviews the evolution of educational games and the role of museums in education. Next, it discusses why learning history matters and examines game-based learning (GBL) theories like behaviorism, constructivism and cognitivism and finally it discusses State-of-the-Art applications in turn-based RPGs, historical games, and museum-based experiences, highlighting gaps that our project aims to fill.

Chapter 3, “Research Methodology and User Survey,” describes the methods used to validate our approach. It begins by outlining the overall research methodology, then presents the design, distribution, and analysis of an 18-question survey. The survey results are divided into five parts: general information, learning preferences, museum visits, virtual museums and gamified learning, and feedback on gamified learning, each analyzed to inform design decisions.

Chapter 4, “Project Planning,” focuses on how the project’s objectives and scope were translated into an actionable schedule. It describes the initial brainstorming sessions, literature review, and discussions with the coordinator that shaped the high-level goals. Next, it lays out four major milestones: Conceptualization & Design, Core Mechanics Implementation, Combat Polishing, and Beta Testing explaining the key tasks for each phase (such as drafting wireframes, coding scripts like Movement.cs and GateScript.cs, integrating audio, and running internal playtests). Finally, it presents a compact Gantt chart–style timeline showing how those phases overlap and build on one another to ensure timely completion.

Chapter 5, “Game Design and Development,” dives into the creative and technical work itself. After outlining the core concept: a museum-inspired side-scroller where quizzes gate combat encounters it presents sequence diagrams for quiz-gate interactions and turn-based battles. The section then details the overworld mechanics (infinite scrolling backgrounds, Movement.cs, CameraFollow.cs, GateScript.cs), explains the combat system (buttons for Fight, Block, Inspect, enemy patrols via EnemyPatrol.cs, HP management via HPManager2.cs, and scene-specific audio

in CombatMusicManager.cs), and describes asset and animation workflows (importing free itch.io art, building Unity Animator controllers for player and enemy actions) [9]. A brief Sound subsection covers how free MP3 tracks were assigned per scene, and the chapter closes by listing realistic functional requirements (movement, infinite tiling, quiz integration, combat logic, audio toggles) and key non-functional constraints (60 FPS performance, responsive UI, modular scripts, high-contrast UI, smooth audio fades) for a one-person bachelor's project.

Chapter 6, "Technologies and Tools," presents a detailed overview of the software and platforms employed. We explain why Unity Engine was selected, describe how Visual Studio and C# scripting were used for implementation, and discuss version control via Unity Hub and Plastic SCM. This chapter also briefly compares alternative engines, Unreal Engine and Godot, outlining why they were ultimately not chosen for this project.

Chapter 7, "Conclusions," reflects on the project's accomplishments and personal growth. It explains how the game meets its dual aim of being both entertaining and educational, quiz performance directly influencing combat difficulty, and highlights the satisfaction of building everything solo in Unity. The chapter then outlines future possibilities: adding new themed levels (Ancient Egypt, Jurassic era), recruitable companions for battles, and eventually a full VR museum experience. It closes by reaffirming that this solo effort successfully demonstrates how a small, tightly scoped educational game can engage players and encourage learning.

2. Presentation of Domain

2.1. Background

IN this part of the project I will be outlining out-lining the key concepts, as well as the connections and overlaps of educational games, museums and historical learning as a whole. The main goal is to provide information on the subject in order to be able to create more engaging learning experiences in the future.

2.1.1. The evolution of educational games

Educational games have existed almost as long video games as a whole have. Initially, video games were only made for the purposes of entertainment, but as the industry grew it attracted educators and developers alike have seen their potential as teaching tools, especially for young children who have a smaller attention span and a harder time focusing when they find tasks uninteresting. The first significant games came out back in 1980s, with games such as Oregon Trail that were intended to make the process of learning fun and combine educational content with gameplay. Back then, these games were mostly text-based, similar to online tests, but with animations and instant feedback to questions to keep students engaged, making for quite an interactive experience.

In the 1990s and 2000s, as technology advanced and the gaming industry became larger and larger, it allowed educational games to evolve too, incorporating better graphics as well as more complex mechanics. Games such as Civilization allowed players to not only develop skills such as strategic thinking and decision-making, but also to learn about concepts such as city-planning, economics, and even consequences of real historical events.

In the 2010s game-based learning (GBL) was accepted as a formal pedagogical strategy. GBL, still used today, is a strategy that uses game-like mechanics and concepts in learning in order to improve student engagement and improve academic performance. Slowly but surely, teachers began using commercial video games in classrooms as part of the didactic process across many subjects. Games such as Duolingo, a language-learning app and Kahoot, a quiz-based learning platform are today widely used in schools and informal learning environments, as well as allowing students to learn on their own in private with ease.

These days, technology such as virtual reality (VR) and augmented reality (AR) are advancing the industry of GBL even further, allowing for immersive educational experiences that were impossible to implement before. VR, for example can allow users to experience historical events first hand, as seen in The VR Museum of Fine Art or Titanic VR, where can users can interact with historical environments in a 3D space.



Figure 2.1 – Bird's eye view of a room in The VR Museum of Fine Art [1]

2.1.2. The role of museums in education

Museums, historical museums in particular, are crucial in understanding the world around us. Unlike, textbooks and PowerPoint presentations, the actually offer tangible, interactive experiences that are, in my opinion, crucial for learning about certain subjects. When visiting a museum, you are able to engage with history directly, by viewing historical artifacts and reading informative plaques, the exhibits often designed in a way that explains quite complex topics in easy-to-understand terms, making the whole experience accessible and engaging for all ages.

Museums have also evolved over the past few decades, embracing technology in order to enhance their visitors' experience. Virtual tours, as well as interactive exhibits and tablets have become quite common in most museums around the world. Many museums also have online collections and educational resources in order to cater to global audience, this expands their market to people who cannot physically visit due to geographical, financial or medical reasons, though there are still many museums who do not have such innovations.

Museums are not just a mere exhibition, they offer a space for immersion, discovery and critical-thinking. As the closest thing we have to bringing history to life, they help the current generations understand what came before.

2.2. *Importance of Learning History*

Understanding history is important in order to form a well-rounded view of how the world works today. History isn't just a narrative of past events, it allows people to understand how past actions influence the present and the future, how society has evolved over the years and the major events that led to the state of the world as we know it. By studying history, people can gain insight into a variety of subjects including culture, politics, economics and social structures, among many others.

The problem is traditional methods of teaching history are often bland and dry. Mechanical memorization of dates, names and events is completely devoid of any agency and it doesn't allow

for any critical-thinking or curiosity. This makes the use of modern methods of learning, including gamified experiences crucial in making history a more attractive and engaging subject.

History isn't only important in an educational context, but also helps in social situations. In the globalized world we live in, understanding the different histories and backgrounds of other cultures and countries allows us properly navigate social situations with people from other parts of the world.

By presenting history in a more interactive and fun way, this project aims to make history a more accessible topic for people of all ages, allowing for an engaging twist on a virtual museum combined with game mechanics.

2.3. Theoretical Foundations of Game-Based Learning

Game-based learning, from now on referred to as GBL, is an educational approach that aims to boost engagement and help reach educational goals, by implement mechanics and elements of games. There are several theories that support the use of games as tools for learning including behaviorism, constructivism, cognitivism, and experiential learning, all of which influence the design of educational games and have a direct impact on the learning outcomes of students.

Behaviorism is a theory of learning that emphasizes the role of environmental factors in shaping the behavior of a person [2]. According to behaviorist theorists such as B.F. Skinner and John Watson, all behaviors are learned through the interaction between the person and the environment around them. They also mention that the process of learning can be enhanced using reinforcement techniques such as rewards and punishment. This theory is clearly visible in gamified learning in applications such as the following:

Instant feedback to actions. In games feedback is instant, every action and click, immediately yields a response, be it positive or negative. When a player takes an action in game, they immediately receive feedback to whether that action was correct or not, which is an important concept in behaviorism. This makes it easy for players to understand and assimilate the desired behaviors they need to follow in order to reach the desired outcomes. Obviously GBL functions the same way. For example, in my game, if the player wishes to open a gate, they have to solve a question. The feedback to their answer is practically instant, the player being informed if they answered correctly, opening the gate in order to progress further. This helps players improve at a faster pace, rewarding positive behavior while correcting mistakes immediately, improving the learning process.

Progress indicators. A key element of video games are things such as progress bars, levels and achievement systems. These allow players to track their progress and feel accomplished when reaching certain milestones. This allows them see a visual representation of how much they have improved. In my game, when fighting any of the enemies in the turn-based combat system, the player always knows exactly the amount of HP the enemy has, and can see their progress through the fight as the health bar becomes smaller and smaller. In this way players stay engaged, seeing exactly how much they have accomplished and it keeps them motivated, showing how close they are to a victory.

Hints and Tips. It is common in games for players to receive automatic hints when they are stuck or encounter difficulty in order to help them progress. This ensures players don't become frustrated or give up, and ensures they can continue slowly progressing through the tasks. This is

a behaviorist approach that focuses on providing reinforcement through assistance. It is even better if the hints don't outright solve the problem, but allow the player to piece together the solution themselves. In my game, players are supposed to answer historical questions in order to progress. Hidden inside the exhibit plaques of my virtual museum are the answers to these questions. If the player reads them carefully, they will be able to easily progress, even if they did not possess the knowledge earlier. This approach increases player retention, keeping progression steady, while minimizing frustration.

Reflective feedback. After completing tasks, games often offer what is known as reflective feedback, where they will go more in depth into the topic that the player just handled, broadening their understanding and allowing them to more easily synthesize the information. This type of feedback can also be used to give context to the actions made by the player and mentioning in what other contexts it may be applied. For example, in my game, after the player answers questions to the armor worn by medieval knights, the player will quickly find that the information they just read will be useful in defeating the upcoming enemy. Reflective feedback helps deepen the player's understanding of certain topics by showing broader uses for a single concept.

By incorporating the principles of behaviorism mentioned above, GBL promotes continuous engagement, using elements of games such as instant feedback and clear progress indicators. This turns the otherwise monotone activity of learning, into a dynamic, iterative process where players can build upon their successes, while understanding and correcting their mistakes.

Constructivism, promoted by theorists such as Jean Piaget and Ley Vygotsky, says that when learning, you gain knowledge through active engagement with the environment, as opposed to passively receiving information [3]. In short, every person constructs their own understanding of concepts based on their own experiences and their prior knowledge. This concept is easily seen in gamified learning in applications such as:

Iterative Tasks. A key feature of constructivism is that the students are encouraged to engage in the content actively through iterative tasks. Games, and by extension gamified learning, are perfect examples of this as players are immersed in environments where they have to make constant decision while interacting with multiple mechanics at a time. For example, in my game, players must apply their knowledge directly when progressing by answering tests in order to pass gates as well as learn weakness of enemies in order to more easily defeat them in combat scenarios. By making the player actually use the knowledge they acquired after they have read it in order to complete tasks and pass challenges, it helps the information cement, making learning memorable.

Simulation and Role-playing. Constructivism also emphasizes the importance of simulation and role-playing. Games provide the perfect environment for players to engage with realistic immersive scenarios. They allow players to make choices that actually affect outcomes, engage with events and storylines and interact with complex game mechanics and systems. In my game, the player takes the role of a night guard at a museum. The exhibits come to life and ravage the museum, and only by learning and using historical knowledge are you able to bring order back to the museum before the next morning, when the visitor come to see the exhibition. This helps form a connection between the events, characters, and player, thus better cementing the information through first-hand engagement, rather than passive reading and memorization.

Collaborative Learning. Many gamified learning experiences promote collaborative learning. By the players working together in order to solve problems and progress, they learn skills

such as teamwork, communication and problem-solving. Many games have a major focus on working together in order to overcome challenges that are otherwise impossible to be done alone. While my game does not have any multiplayer elements to it, many RPGs have strong communities. Players interact outside of the game exchanging information, asking questions and revealing secrets. Such a community could exist for my game as well if it were ever released on the market.

Challenging Puzzles and Problems. Constructivist theory pushes for problem-solving and hard challenges that actually force the participants to engage with the material. Games often have high-difficulty scenarios where the players are forced to apply their knowledge and skills in creative ways in order to pass challenges and clear higher levels. While my game isn't necessarily difficult, without paying attention and using the mechanics at your disposal, the enemies can easily defeat you. Also, based on the performance of the player answering the historical questions along the game, the combat may be easier or more difficult. Theoretically you can use this as a difficulty slider, for players who wish for a more challenging experience.

Iterative Learning and Reflection. Iterative learning is important in constructivism. Students are encouraged to experiment, make choices and adjust strategies and ways of thinking depending on the received feedback. This process works perfectly with GBL, as games already allow for multiple iterations of the same activity and gradual improvement through levels and stages. For example, in my game, players might reattempt the gate puzzles, finding that new questions appear instead of the old ones, being chosen at random from a larger pool. Also, they might retry combat encounters trying different strategies each time. Iterative learning helps the learners become more resilient and it improves critical thinking as they learn from mistakes and continue improving.

Cognitivism is the last concept I will be tackling here and it focuses on the internal process of learning focusing on areas such as memory, problem-solving and critical thinking. It pushes for the idea that learning is a process that takes information you receive and builds structures with it inside your mind, creating connections and building upon already existing knowledge. It is clearly seen in GBL applications such as:

Information Processing. As players progress through games, they are often introduced to new information that they must quickly assimilate and integrate into their already existing routines and strategies. GBL works similarly, breaking the information into smaller digestible chunks that are more manageable. For example, in my game, players are slowly introduced to the mechanics of the overworld part of the game such as movement and interacting with objects, then, only later do they have contact with the turn-based combat system, with its own set of mechanics, thus not overwhelming the player and allowing time for each information set to properly cement in their minds before adding another.

Memory Reinforcement. Games are really good at reinforcing memory, players having to recall information and mechanics constantly in order to form new strategies as they progress. This process of constantly retrieving information from their mind makes their memory vastly better and keeps the knowledge for longer periods of time, without it being forgotten. In my game players can learn historical knowledge from museum plaques that they use later to solve puzzles, thus needing to remember it after they have read it. They can also memorize patterns in the first combat encounters in order to recognize the later enemy's moves more easily and predict what it might

do. By providing repeated opportunities for the player to use the information they have gained it helps them cement it better in their minds, thus improving their memory.

In conclusion, many scientific theories apply to GBL, proving its effectiveness in being used as a learning tool. Behaviorism pushes for reinforcing positive behavior through rewards and progression, constructivism focuses on hands-on approaches to learning, making the experience more engaging and interactive, and cognitivism highlights the importance of a good memory gained by taking part in interactive and iterative activities widely present in games. I also wish to incorporate many of these elements into my own project, creating an immersive, fun and effective learning experience for the people playing it.

2.4. Engaging with History Through Gamification

While traditional history education often relies on passive consumption like reading textbooks, listening to lectures and memorizing dates and names, gamification turns historical content into an interactive and engaging experience. In my project, I use gamified mechanics in order to turn the stale process of learning history that most people are used to into a dynamic environment. By embedding quiz questions, puzzles and other challenges into my game, players can do more than observe history, they can interact with it and apply it in fun ways. My goal was twofold: to create a fun, enticing game experience while offering a way to learn history, a subject I quite enjoy, that is both engaging and enjoyable.

At its core, gamification harnesses elements such as immediate feedback, progressive challenge and meaningful rewards to keep learners engaged. Players get immediate feedback and tangible progression each time they answer a question correctly, boosting retention and encourages correct approaches. Beyond simple Q&A mechanics, I introduced layered challenges that require synthesizing more complex pieces of historical knowledge in order to pass harder challenges like my turn-based combat system. For example, in the medieval level of the museum, the player learns about how iron ore smelting worked as well as how forges created armor sets. The player can then apply this knowledge in order to more easily defeat the knight in the upcoming combat. By offering players a tangible way to use their knowledge it will boost their engagement and make the information cement better over long periods of time.

Narrative context also plays an important role in sustaining learner engagement. Rather than presenting history as decontextualized facts, my virtual museum has a cohesive storyline: players assume the role of a museum night guard whose exhibits have suddenly come to life. I tap into narrative transportation theory, which posits that stories can heighten attention and emotional investment. When players feel they are “part of the story,” they become more motivated to learn and remember the underlying material.

By combining all of these game features like immediate feedback, narrative framing and progressive challenges among others, the gamification approach turns historical content into an active learning environment where exploration is encouraged. The players who participate are not merely memorizing facts, dates and names, but they’re constantly applying their knowledge in context, seeing the tangible benefits of their growing understanding. In this way it harmoniously combines the thrill of gameplay with the prospect of learning.

2.5. Similar Applications

For this project I will be outlining 3 key areas of Sota relevant to the topic: turn-based RPGs, historical games and museum-related games specifically. By doing so I will highlight what this project does different from the industry standards of today.

2.5.1. Similar applications in turn-based RPGs

Role-playing games or RPGs for short, are a genre of video games that allow players to assume the role of characters inside the game setting. They are typically focused on character progression and narrative-driven experiences, concepts taken from table-top RPGs, games in a similar style played with pen and paper ever since the launch of Dungeons & Dragons in 1974.

Turn-based RPGs are then a subgenre of RPGs where combat is structured in a sequential manner, where players take turns to perform actions such as attacking, defending or using items. These types of games often have a focus on strategy and resource management, letting the players think over each one of their moves carefully on their turn, without the need for fast reactions, making them a more tactical experience. Out of this genre I have chosen 2 modern games which I have played and greatly enjoyed: Persona 5 and Darkest Dungeon 2. Critically acclaimed and widely available, these games have revolutionized the genre they are part of.

Persona 5 is an eastern singleplayer turn-based RPG developed by ALTUS and originally released in September 2016 in Japan and worldwide in April 2017 on consoles. Its massive success drove the developers to create Persona 5 Royal in October 2022, a more polished version, featuring new content, further building on the game's formula and finally bringing it to a wider audience by also releasing on PC. While also bringing innovation to the combat formula itself by rewarding players with extra turns if they match the enemies' weaknesses, the most notable improvement they added was the seamless blending of the genre with social simulation elements. While other games in the genre before it had engaging stories, they often felt scripted and allowed for minimal choices from the player at key moments. The way the systems in Persona 5 are structured the player must balance schoolwork, character relations and other extracurricular activities at all times, all of these choices having an impact on both story progression and combat abilities. The simulation of daily life makes the world feel more alive and immersive.

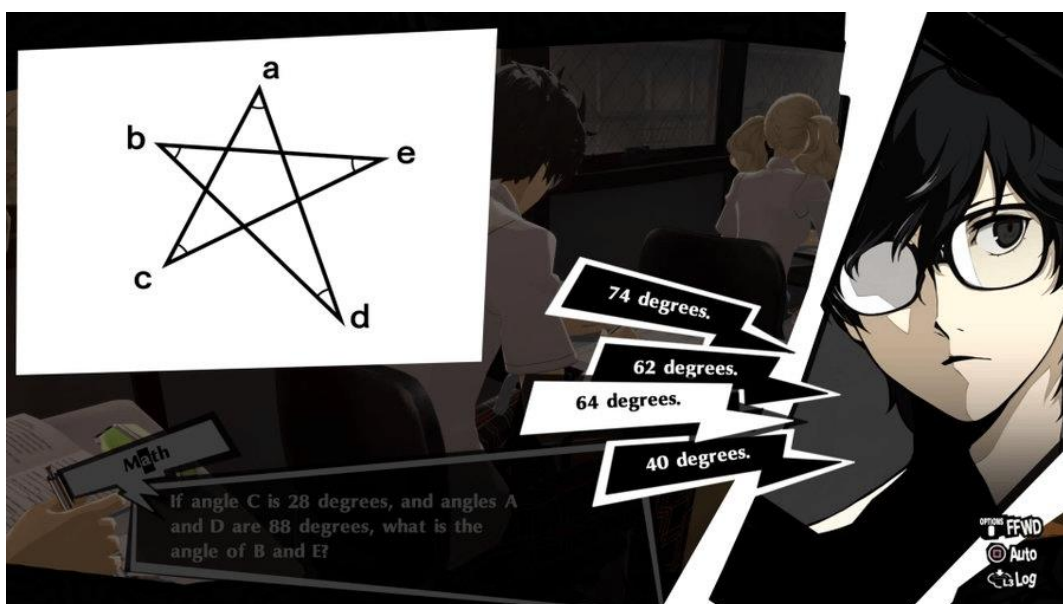


Figure 2.2 – Answering questions during a school day in Persona 5 [4]

Darkest Dungeon 2 is yet another popular turn-based RPG which I have had the pleasure of playing. Released in May 2023 by Red Hook Studios, this game's main draw is its Lovecraftian-inspired cosmic horror aesthetic and instantly recognizable dark and gothic art style. Darkest Dungeon 2, the sequel to Darkest Dungeon, refines the original's stress-based formula. In this game you do not only have to worry about your character's health, but also manage their psychological stress, a simple concept, but one that makes the experience infinitely more complex and engaging. Without proper care taken of your characters' mental needs they might fall victim to paranoia or irrational behavior, so it is always in your best interest to keep them content. Also new to the second game in the series are the relationships between members of your party, each one has an opinion on the others and each choice made may break or consolidates the relationships between them. All of this makes managing your party's mental health and interpersonal relations just as important as the combat itself, offering players rewards for having a calm party that works well together, and punishing them for high stress levels and hate between members.



Figure 2.3 – Positive party relationships in Darkest Dungeon 2 [5]

From these examples it is easy to understand that a good turn-based RPG isn't made by the combat system alone, but by fusing it with other outside concepts to create captivating and unforgettable experiences. This project will also integrate puzzles and exploration in a virtual museum, rewarding players for their knowledge of history, knowledge they will be able to gather in the game of course. By exploring and participating in extra activities players will be awarded with useful information, not only facilitating the learning process, but also making future encounters easier.

2.5.2. Similar applications in historical games

Historical video games are designed to immerse the player in settings based on real-world history. They usually feature accurate representations of historical events, cultures and technologies. Such games are based on thorough research, trying to reproduce certain time periods as accurately as possible in order to create authentic experiences. While many historical games do take certain creative liberties in order to make for more enjoyable gameplay, most do try to stick to realism and educate the player about the past.

Such a game is Kingdom Come: Deliverance 2, or KCD2 for short. A more recent title, launched February of this year by Warhorse Studios, KCD2 seeks to recreate medieval life from the 15th century Europe as accurately as possible. The combat system is complex, simulating physics and weapon weight emphasizing timing and positioning. The day-to-day aspect also has a focus on realism, having to eat, drink, sleep and even take care of your hygiene in order to stay healthy. You also have to respect the hierarchy you are part of, listening to guard's orders and obeying the law. The quest system accurately depicts the political complexity of the time, by having no clear morally good choices. The game is also completely grounded, straying away from fantasy elements in order to deliver a true historical experience.



Figure 2.4 – Rural life in 15th century Europe as seen in Kingdom Come Deliverance 2 [6]

Most historical games follow a similar trend of trying to simulate a part of the historical world for the player to experience. I opted for a more direct and familiar approach, by having the learning experience be at the center of my game and incorporating museum elements, I can offer a more direct learning experience, while also incorporating non-realistic elements such as exhibits coming to life without compromising the integrity of the world.

2.5.3. Similar applications in museum-based games

The museum-based game genre is quite a niche one and the one I have personally had the least experience with. As the name suggests these types of games take place inside virtual museums, however from the research most of these games have a focus on building and running the museum itself. While yes, these games do still contain much information about the exhibits and attractions you place down which can be used for learning, it is not the main focus of the games and as such it is comparatively lackluster. The games usually have the managerial and logistical aspects of running the institution itself, managing layouts and finances and ensuring visitor satisfaction. The fact that you are building the whole building from scratch, room by room also doesn't allow for the type of exploration that I wish to implement in the project. As far as I am aware, there is no game with similar ideas to mine on the market, transforming a museum into an interactive adventure.



Figure 2.5 – building exhibits in Mondo Museum [7]

In conclusion, I believe this project carves out a unique space on the video game market, blending turn-based RPG mechanics with a museum setting and fantastical storytelling. By researching into the SotA, I believe that my game offers a fresh take across the 3 genres mentioned above, creating a unique experience that merges historical learning, strategy and story-based adventures.

3. Research Methodology and User Survey

3.1. Research methodology

TO ensure that my project is both engaging for players and is able to achieve the desired educational goals. Other than researching the current applications and technologies present on the market currently, I set up an 18-question survey in order to gauge the interest of people in history as whole, museums, learning through games, and finally features of the game specifically.

The survey was distributed via social media, by me, friends and colleagues, as well as my family members in order to reach as many people as possible. At the moment of writing this it has 70 responses that have a decent spread among the different available age groups. For a project of these scale, I have surmised that this number of users is enough and I started analyzing the data gained from the questionnaire.

Before being posted, I got the help of 2 colleagues who have already completed their bachelor's projects in previous years, as well as my coordinator in order to make sure the questions are easily understood and they cover enough of the subjects I needed information on.

I then used the google form export feature in order to easily visualize the data using pie charts as well as line graphs depending on the type of question and I analyzed every single one with regards to the application and how successful it would be if it actually were to appear on the market.

3.2. User Survey

In this section, I will be examining and commenting on the survey data I have gathered for this project in order to have a better understanding of general opinions regarding history, learning in general, museums and the use of games in education. I will also discuss how the results affect my game. The questions are split into 5 distinct parts: General Information, Learning Preferences, Museum Visits, Virtual Museums and Gamified Learning and Feedback on Gamified Learning.

3.2.1. Survey – general information

What is your age?
70 responses

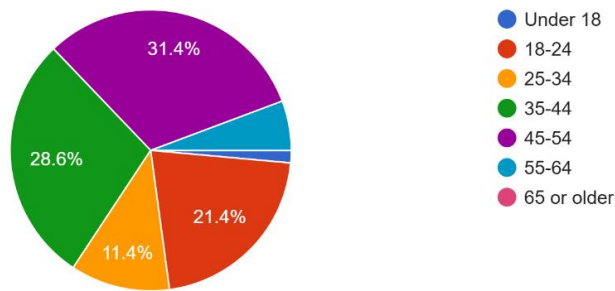


Figure 3.1 – Survey question: “What is your age?”

Results: I have tried my best to get responses from as many age groups as possible in order to have accurate unbiased data, even so around half of the answers ended up falling in the 35-54 range, but 30% were also people between 18-25, an age group more comfortable with technology, digital learning and video games.

What is your gender?
70 responses

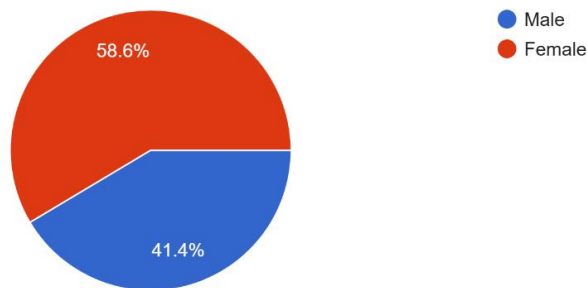


Figure 3.2 – Survey question: “What is your gender?”

Results: I have also tried to be as close as possible to the 50/50 mark for the gender distribution, however there are more women that have completed the survey than man. Even so I consider it a decent ratio and wish to see how the data is affected, as video game audiences, especially RPGs tend to be predominantly male.

How Familiar are you with History?

70 responses

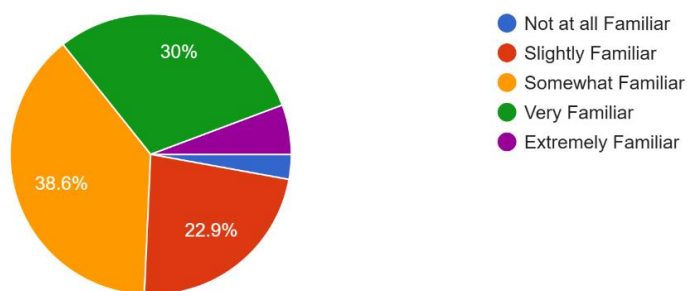


Figure 3.3 – Survey question: “How familiar are you with History?”

Results: In this section I wanted to analyze the level of understanding in history that potential players might have. The data is quite conclusive in this regard, the vast majority of people, around 90%, have some sort of prior knowledge in history, favoring the average, while just a few are experts or complete novices. This is understandable, as most people have contact with history during their lives one way or another, and people who have mastered the subject, or have no interest in it whatsoever are not the target audience of my game anyway, as they would have nothing to gain from it from an educational point of view.

3.2.2. Survey – learning preferences

How do you prefer to learn? (Select all that apply)

70 responses

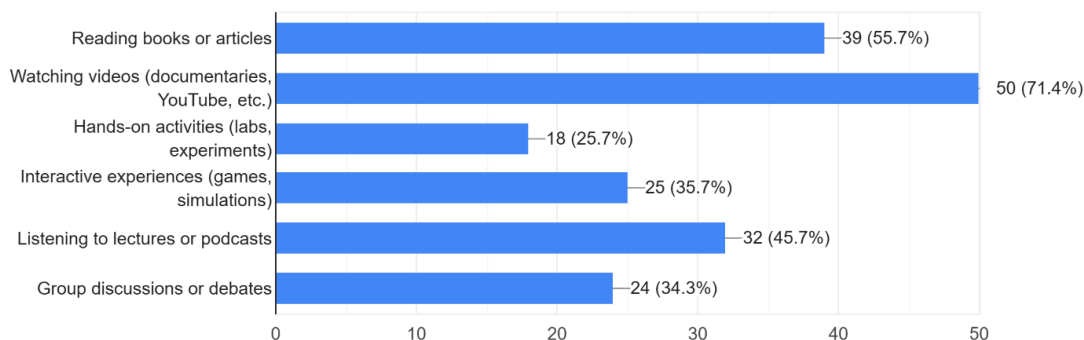


Figure 3.4 – Survey question: “How do you prefer to learn?”

Results: In this question I wanted to highlight people’s preferred method of learning. An overwhelming majority prefers learning using videos and documentaries. Visual and auditory storytelling at the same time is clearly well received, which is good news for my game. Reading books takes over half of the votes as well, clearly people still value the in-depth written content they provide. Only a third of the people voted for games and other interactive experiences, no doubt because of the age range predominant in my survey. Games, as a tool for learning, are clearly not as well-known and sought-after as videos, books and audio podcasts, however I hope this may change in the future.

How do you feel about using technology to learn (e.g., educational apps, games, virtual environments)?

70 responses

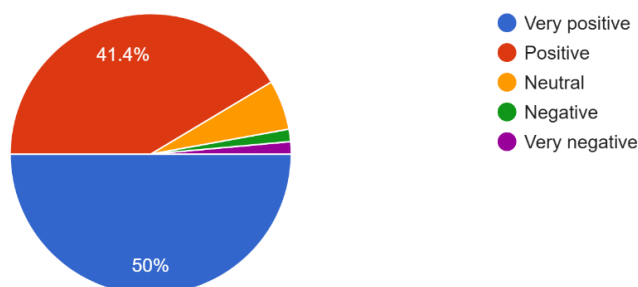


Figure 3.5 – Survey question: “How do you feel about using technology to learn?”

Results: It seems that the vast majority (over 90%) have are positive or very positive about using technology in the learning process. Only a slight minority, 5.7% are neutral on the subject, while only a mere 2.8% have negative or worse opinions on the matter. This shows me that there is an overwhelming interest in this day and age in using technology like apps and virtual experiences as a tool for learning. I have no doubts that, had a younger demographic been asked the same question the answer would skew even more in favor of very positive due to how they have a deep connection to technology from a very young age.

Would you consider using a video game as a way to learn about history?

70 responses

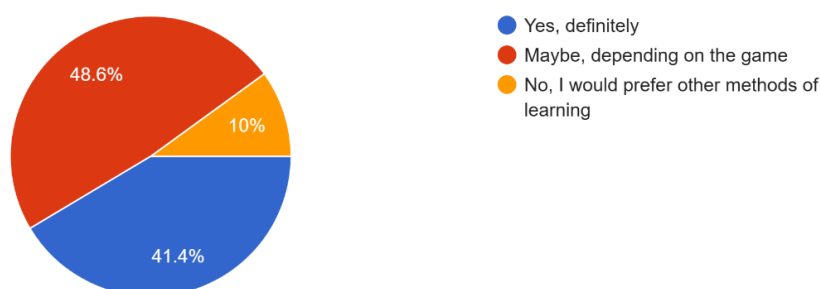


Figure 3.6 – Survey question: “Would you consider using a video game as a way to learn about history?”

Results: I wanted to see whether the positivity regarding technology also spreads to video games as well, or is only limited to educational apps and similar products. There is clearly a strong openness towards GBL, as a whopping 90% of all votes were either on board with it, or open to it depending on the level of quality and the content delivered, while only 10% of people would prefer only using other methods of learning. I am sure that my project would be well received as well, as even with half of the “maybe” group, I would still have way over 60% interest from users.

How much do you enjoy learning through video games or gamified experiences?

70 responses

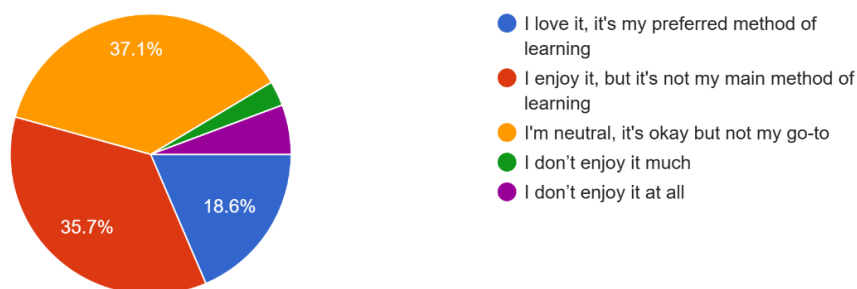


Figure 3.7 – Survey question: “How much do you enjoy learning through video games or gamified experiences?”

Results: I wished to gauge the level of engagement people have with using video games for learning. While more than half of all people do enjoy it, indicating a solid base of fans, only 1 in 5 has it as a preferred method, many only using it as a supplement for other for the more popular ones discussed above, rather than a replacement. Similarly to the last question, many are neutral, meaning that they would be open to experiencing GBL, but it needs to meet certain standards of quality and content in order to capture their interest fully. Under 10% of the votes have a negative opinion on the subject, clearly preferring other learning methods.

What type of game mechanics do you enjoy the most in educational games? (Select all that apply)

70 responses

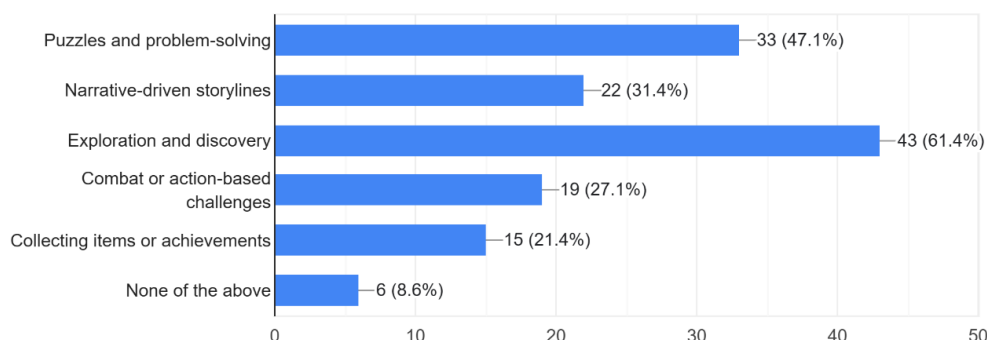


Figure 3.8 – Survey question: “What type of game mechanics do you enjoy the most in educational games?”

Results: I wanted to see exactly what part of games people would be most excited to experience in an educational game. There is an overwhelming majority, above 60%, with an interest in exploration and discovery. It is clear that the fact that the learning process is non-linear is greatly valued by most of the users. On second and third place there are puzzles and problem-solving as well as narrative-driven storylines. I find that people are looking for engaging prospects to enhance their learning experience. A few appreciated combat and collecting items achievements

and only a small 8.6% did not wish for any of these mechanics to be present. This shows me that most players are interested in at least one, if not more of the game elements present in GBL.

3.2.3. Survey – museum visits

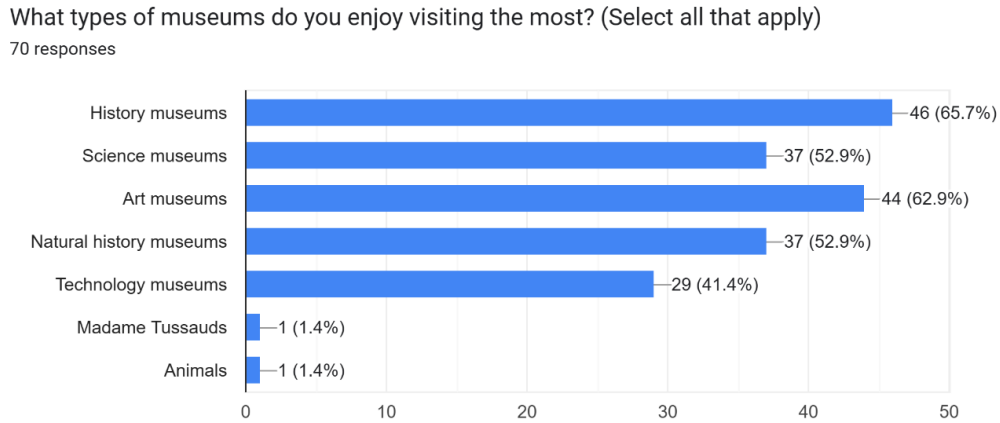


Figure 3.9 – Survey question: “What type of museums do you enjoy visiting the most?”

Results: I wished to gauge the interest of people in different types of museums. History Museums lead with 65.7%, which is a good sign for my game. It seems this is the most popular type overall, with Art Museums coming in a close second spot with 62.9, also a good sign since it indicates an appreciation for aesthetics and visual stimuli which are a major part of my game. Science and Natural History museums had an average amount of votes, this clearly shows how users appreciate factual content presented in museums. There were minimal “other” answers, only 2 votes, so it is safe to focus only on the mainstream museum types already present in the poll for the rest of the way.

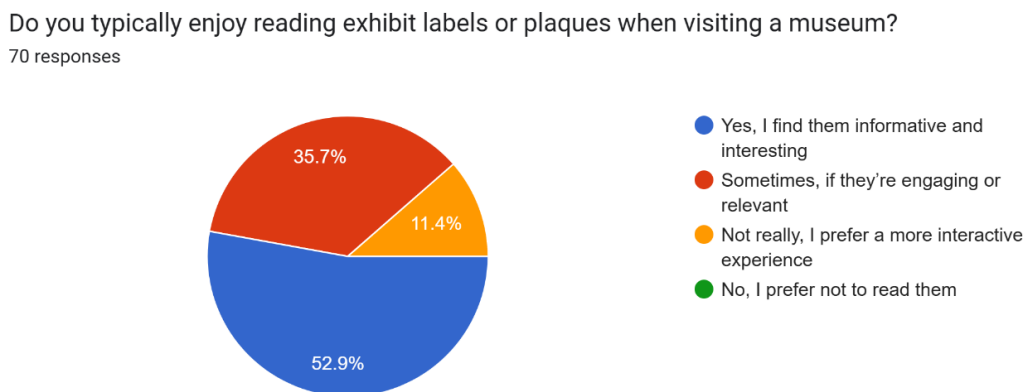


Figure 3.10 – Survey question: “Do you typically enjoy reading exhibit labels or plaques when visiting a museum?”

Results: Since plaques are a major part of my game, I wished to see exactly how interested people are in reading these when they visit and actual physical museum. Over half of all users said they enjoy reading the labels, while 35.7% stated they read the plaques for the exhibits that they find engaging and relevant to their tastes. This is a good sign for my game, as only 11.4% stated

they prefer other forms of interacting with the exhibits and not a single person said they never read them when they visit museums.

Would you consider visiting a museum more often if it had interactive digital exhibits (e.g., virtual tours, AR experiences)?

70 responses

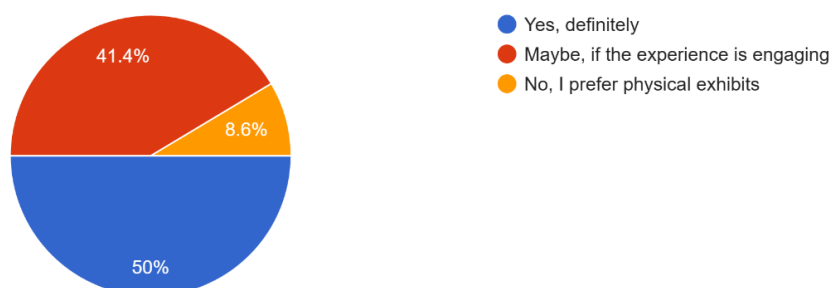


Figure 3.11 – Survey question: “Would you consider visiting a museum more often if it had interactive digital exhibits (e.g. virtual tours, AR experiences)?”

Results: I also wished to know whether there would be more interest in visiting museums if there would be more digital and interactive exhibits such as virtual tours or AR experiences. The results were overwhelmingly positive, over 91% of all people are certain they would enjoy the experience more, or at least are open to the idea, while only 8.6% said they would prefer the classic museum experience without any technology involved. This shows me that there is a strong market for combining history and modern technology, the museum RPG having a good amount of people who would be open to trying it.

3.2.4. Survey – virtual museums and gamified learning

How likely would you be to try a virtual museum experience from home (e.g., interactive exhibits, virtual tours)?

70 responses

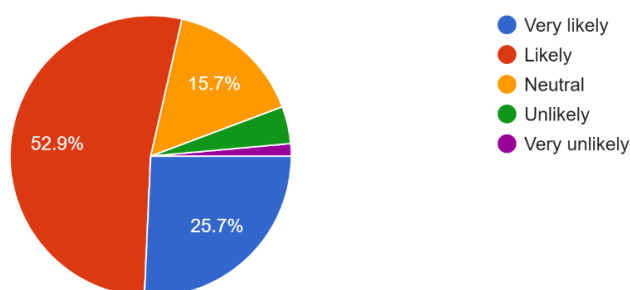


Figure 3.12 – Survey question: “How likely would you be to try a virtual museum experience from home (e.g. interactive exhibits, virtual tours)?”

Results: In order to gauge interest for the museum RPG game, I first wanted to see whether people would be open to experiencing a virtual museum experience, a similar product to my game

that is more common on the market already. Around 78% of people showed high or very high interest in the experience, proving that offering such an experience from the confines of their home is quite sought after by most users. Around 1 in 6 people were undecided and would probably be convinced based on the quality and specific content tackled, while only around 6% of the users had negative thoughts towards it indicating that most people would be open to this sort of virtual museum experience if only, they knew about them and had the proper access.

Would you prefer a physical museum experience or a virtual one?

70 responses

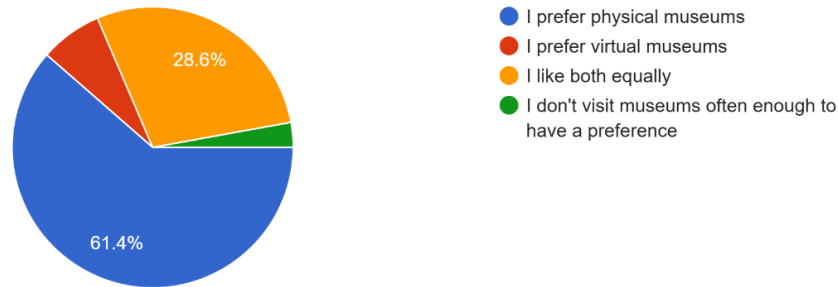


Figure 3.13 – Survey question: “Would you prefer a physical museum experience over a virtual one?”

Results: Now that the virtual museums were so well received, I wanted to see if people would prefer such an experience over the traditional physical museum. Users held on to their preference, probably because they value things like the social aspect, the atmosphere and the physical interaction and proximity to exhibits. Around 28% were neutral, not preferring one over the other, and 7% of users were already into virtual museums. I believe that most of these people have never experienced a virtual museum and have no way to gauge their preferences properly because of how niche and undocumented the subject is. I would hope the votes would shift greatly, had this technology be more widely spread and information on it more readily available.

How interested are you in the idea of using a game-based approach to explore historical events (e.g., a turn-based RPG set in a historical museum)?

70 responses

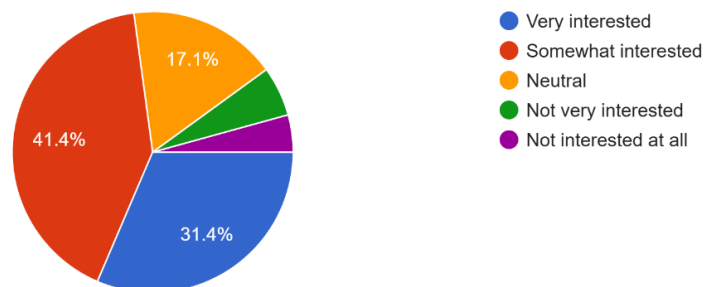


Figure 3.14 – Survey question: “How interested are you in the idea of using a game-based approach to explore historical events (e.g. a turn-based RPG set in a historical museum)?”

Results: Now that I had the general information out of the way, it was time to check the overall enthusiasm for a game made for historical learning. A whopping 72.8% responded positively, with 31.4% showing very high interest. Another 17.1% were neutral showing they might be able to be swayed by playing a demo or watching a presentation. Only 10% had negative takes on the subject, showing that the vast majority of people would be interested in trying this sort of experience for their learning. There is clearly a good market for my game to release in, with many potential players waiting for such a product.

Would you be more likely to engage with historical content in a game that incorporates elements like quests, rewards, and interactive narratives?

70 responses

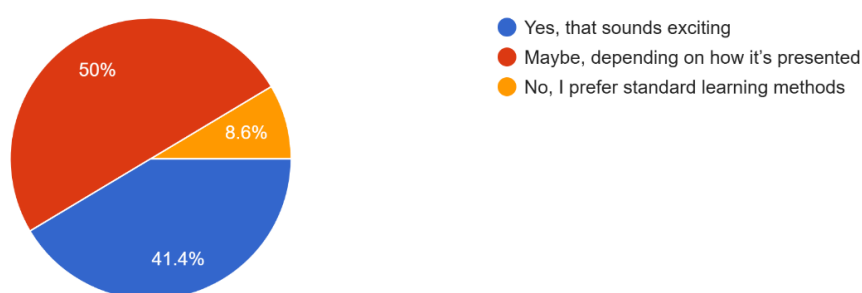


Figure 3.15 – Survey question: “Would you be more likely to engage with historical content in a game that incorporates elements like quests, rewards, and interactive narratives?”

Results: In this question, I wanted to see if people would find that incorporating elements of games such as quests, rewards and interactive narratives would make them engage more with historical content. A strong majority, 41.4%, said that they would find such a concept exciting, while 50% were interested, but needed more concrete information before being convinced. This shows that 91.4% of all users have an open mind about it and would give it a try, if it was presented properly. Only 8.6% said they would stick to traditional learning methods, showing that most people find this sort of approach to learning more desirable and engaging. It is clear that a majority of people would be interested in the use of game mechanics present in GBL in order to enhance their learning process.

3.2.5. Survey – feedback on gamified learning

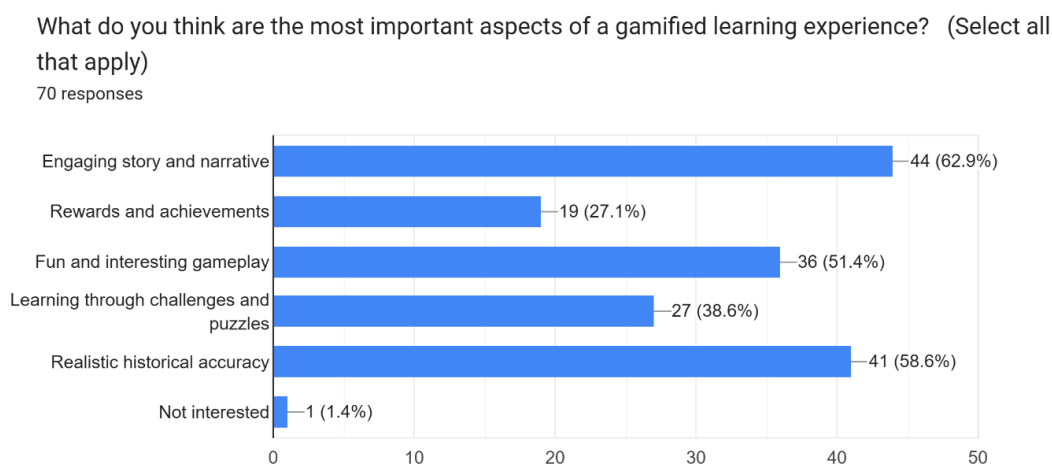


Figure 3.16 – Survey question: “What do you think are the most important aspects of a gamified learning experience?”

Results: It is clear that there is widespread interest in using games for learning, so I wanted to see exactly which parts of GBL people find most engaging. A vast majority of 62.9% stated that the engaging story and the narrative elements in a game would be the main aspects they would play an educational game for. In a close second place with around 58.6% of votes, people look for realistic historical accuracy, looking for an authentic experience where the knowledge gained is accurate and true. Over half of all people are excited about the concept of fun and interesting gameplay along with their educational process. A few enjoy puzzles and achievements, and only 1 person has no interest in any of the mechanics making up a gamified learning experience. This shows me that the many elements that make up GBL cater to a wide and differed audience having quite a wide-spread appeal for most people in one way or another.

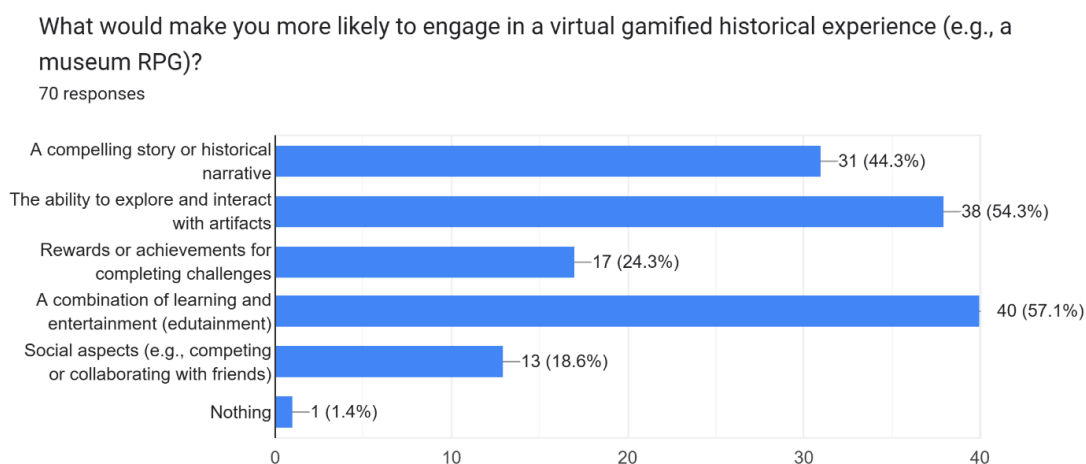


Figure 3.17 – Survey question: “What would make you more likely to engage in a virtual gamified historical experience (e.g. a museum RPG)?“

Results: Here is where I wanted to see exactly which features of my game would make people more eager to play it. The top answer was “edutainment” with 57.1% of people clearly

wishing for an engaging and fun way to tackle learning history. The ability to explore and interact with the environment was also very important for people getting 54.3% votes. Players want an interactive experience where they can explore at their own pace and make contact directly with historical artifacts and events at their leisure. With a decent 44.3% of votes, people also wish for a compelling story with a historical narrative as part of their educational activities to keep things engaging and make them more invested in what is happening on screen. There was also a about 1 in 5 people who enjoy rewards and achievements as well as the social aspect of games, while only 1 person said none of these are appealing. Since my game has a large focus on simulating an actual museum, blending gameplay with learning seamlessly, with a 2d world to explore and a fun story happening in the background, I suppose it matches with most people's preferences, making it quite a desirable product.

What art styles do you prefer for educational games?

70 responses

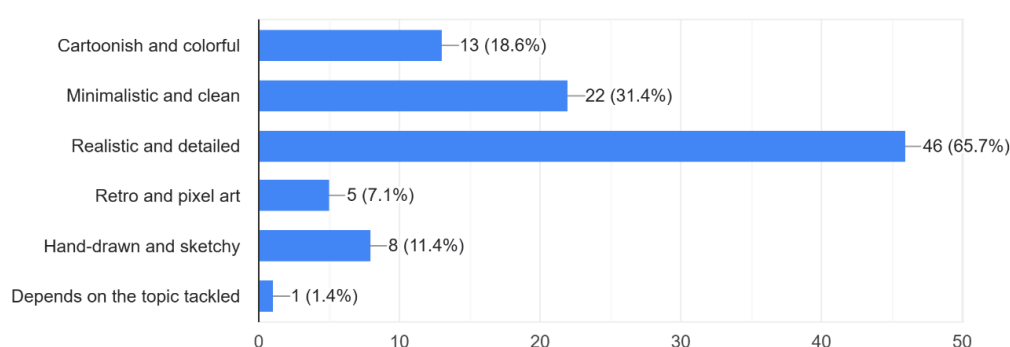


Figure 3.18 – Survey question: “What art styles do you prefer for educational games?”

Results: Lastly, I was curious what art style people would prefer in such a game. A whopping 65.7% of people would prefer a realistic and detailed art style, most likely because they prefer historical accuracy and that would be the best medium in which to convey it. A decent 31.4% also like the idea of a minimalistic game for learning history, without much clutter so they can focus on the educational part most likely. The rest of the votes are split among the other types, with 1 person saying that the art style should vary depending on the topic that is tackled in the game.

4. Gameplay and Player Flow

IN this chapter, I walk through the player's journey from the main menu into each level of the game, describing how menus, controls, and in-game environments come together. I explain what the player sees and experiences at every step, from clicking "New Game" to facing off against the final boss.

When the game launches, the player is greeted with a dimly lit foyer that resembles the front of a museum at night. Moonlight filters through tall, arched windows and casts soft patterns across marble floors and stone columns. Centered on a dark wall is the game's title logo, rendered in an elegant serif font with a metallic sheen that calls to mind brass plaques in a real museum. Below this logo are five rectangular buttons arranged vertically, each encased in a thin golden border that gives off a soft glow when hovered over.

The top button reads "New Game," and selecting it immediately starts a fresh playthrough, loading the first overworld scene outside the museum. Directly beneath is "Load Game," which remains greyed out until the player has reached Level 2 in a previous session; once unlocked, clicking it places the player back at the castle entrance. The third button, labeled "Logs," opens a scrollable list showing every exhibit plaque the player has read so far, each entry displays a small icon, an exhibit title, and a preview of the plaque text so players can review any historical facts they might have forgotten. Below that is the "Mute/Unmute" button, which toggles all background music and sound effects; when muted, the button icon changes to a crossed-out speaker. Finally, the bottom button, "Exit Game," closes the application and returns the player to their desktop. Even this menu screen hints at the museum setting with background audio softly playing a distant echo of footsteps and hushed voices, suggesting other visitors strolling through empty galleries.



Fig 4.1 – Screenshot of main menu showcasing all buttons

After the player clicks “New Game,” the screen fades into Overworld1, showing a simple exterior of a grand museum building at night. The sky is a deep indigo, and soft lamplight from wall-mounted sconces casts warm pools of light on the cobblestone walkway. Here, the player learns the basic controls in a stress-free environment. Pressing D or the right arrow key causes the character to walk to the right, while A or the left arrow key moves them to the left. Holding Shift increases movement speed, allowing the character to sprint. Most importantly, pressing W triggers any nearby interactable object, allowing him to enter the building and progress.

Once inside Museum1, the player finds themselves in a long, rectangular hall lined with marble floors, stone pillars, and display cases. The lighting here is soft and ambient, provided by overhead chandeliers and wall sconces, which gives each artifact a gentle spotlight. Along the left and right walls, several exhibit pedestals hold glass cases containing artifacts ancient pottery shards, small bronze statues, and faded scrolls. Each glass case is flanked by a polished wooden plaque. When the player presses W near a plaque, a UI overlay shows up in the center of the screen containing a detailed description of the artifact and its historical context. Hidden within that text is the clue to the next quiz gate’s correct answer. For example, one plaque describes Roman aqueducts and explains how engineers relied on gravity to move water across valleys. Reading that plaque informs the player that “gravity” is the answer to the question they will encounter shortly.

Spread throughout this hall are three quiz gates. As the player approaches a gate they are prompted to interact with it and a UI panel slides up from the bottom of the screen. The panel presents a multiple-choice question, such as “Which force carried water across a Roman aqueduct?” along with three answer buttons. If the player clicks the correct button, the gate’s metal doors animate open, letting the player progress further through the level. If the player selects an incorrect answer, a brief red “Try Again” message displays near the top of the screen, and the wrong-answer counter in DifficultyManager increments by one. This counter persists via DontDestroyOnLoad so it carries over wherever the player goes next. After successfully passing the first gate, the player walks down a short corridor to the next exhibit plaque. That plaque focuses on Greek pottery and includes the word “clay” in its text. When the player re-encounters the second quiz gate, they must choose “clay” from among three presented options. After the second gate opens, the player reaches the final plaque. That last plaque discusses medieval manuscripts and mentions that scribes wrote in Latin. At the third gate, the player types “Latin” into a small input field and presses Enter. The player is then able to continue to the next level, entering the medieval wing.



Fig 4.2 – Screenshot of the first quiz gate the player encounters

Descending the staircase, the player enters Castle1, a dimly lit, torch-lit corridor that feels like stepping back into medieval times. The stone walls bear faded tapestries depicting knights in jousting tournaments, and suits of armor stand at intervals on pedestals. The floor is uneven cobblestone, and flickering torchlight casts dancing shadows across the hallway. Exhibits here focus on medieval weapons, armor types, and everyday life in a castle. Each exhibit comes with a wooden plaque that the player activates by pressing W. These plaques present longer descriptions than those in Museum1 and hide clues to the next gates. For example, one plaque about blacksmithing explains how a smith heated iron in a forge and hammered it to create a blade. The next quiz gate asks, “What did a smith heat iron in to create a blade?” Instead of multiple choice, the gate presents a blank text field where the player must type “forge.” If they mistype or guess incorrectly, a message appears saying “Incorrect, try again,” and the wrong-answer count increases.

Near the end of Castle1, the player notices a knight patrolling in heavy plate armor. This knight moves along a set path defined by EnemyPatrol.cs. As the player approaches a certain range around the knight, the knight’s head turns, as it detects the player’s presence. It then starts running towards the player until they intersect. Immediately, the screen fades to black, and the scene loads Combat1. This patrol-trigger ensures that the player cannot skip directly to the final corridor without answering the earlier gates, and also teaches the player to be mindful of their surroundings before entering combat.

Combat1 takes place in a small courtyard just beyond Castle1’s gates. The courtyard is enclosed by high stone walls, and torches flicker at regular intervals, casting an orange glow onto a cracked flagstone floor. In this arena, TurnBasedCombat.cs handles battlefield setup. The player’s character appears on the left side of the screen. On the right stands the knight enemy, fully armored with sword and shield. Above each character is a health bar: the player’s HP is displayed in the upper left corner as “HP 10/10,” while the enemy’s HP is shown in the upper right once the enemy is spawned. Below the combat area, the CombatUI panel is visible, containing the three main buttons labeled “Attack,” “Block,” and “Inspect.”

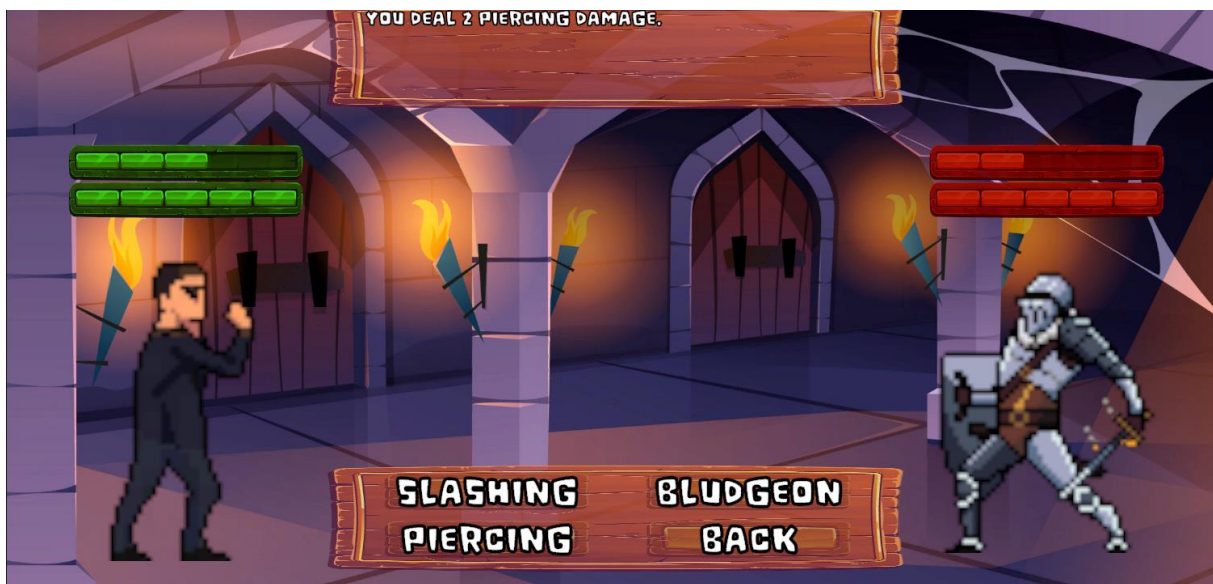


Fig 4.3 – Screenshot of the player attacking the enemy knight in the turn-based combat

Selecting “Attack” slides up a submenu showing three icons, each representing a different attack type: Slashing, Piercing, and Bludgeoning. Clicking Slashing triggers an animation of the player performing a swift diagonal sword slash (controlled by `animator.SetTrigger("Attack")`), if the knight’s weakness is slashing it will take more damage. Once an attack lands, the knight’s health bar updates in real time, and the knight’s sprite briefly flashes red.

If the player clicks “Block,” their character raises their blocking stance with a “Blocking” animation and `isPlayerBlocking` is set to true. This flag ensures that the next incoming enemy attack deals half damage. Clicking “Inspect” brings up a small tooltip in the center of the screen that says, “This knight wears plate, bludgeoning attacks are most effective,” but uses up the player’s turn entirely. After the player’s action, the knight’s AI chooses randomly between `WeakAttack` and `StrongAttack`. `WeakAttack` simply triggers the knight’s “Attack_Weak” animation and deals base damage (for example, 1 HP) to the player. `StrongAttack` telegraphs by turning the enemy red, then plays the “Attack_Strong” animation and deals `baseDamage + 3`. If the player had blocked, the damage is halved; otherwise, the player’s HP is reduced by the full amount. Once the player’s HP reaches zero, `GameOverMusic` plays, and the screen displays a “Defeat” banner. Pressing the Space bar reloads `Combat1` so the player can try again. If the knight’s HP drops to zero, `VictoryMusic` plays and a “Victory!” banner appears; pressing Space returns the player to `Castle1`, where the final gate is now removed.

`Combat2` uses the same turn-based framework as `Combat1` but introduces greater complexity. The cavalier prefab has higher baseHP (20) and baseDamage (2), and is equipped with a Heal move that only becomes available when its HP falls below 30 percent. As soon as `Combat2` starts, `TurnBasedCombat.cs` calls `DifficultyManager.Instance.CalculateScale()` to adjust the cavalier’s stats based on how many quiz gates the player missed in both `Museum1` and `Castle1`. A scale of 1 adds 5 HP and 2 damage to the cavalier; a scale of 2 adds 10 HP and 4 damage.

Thus, at the highest difficulty, the cavalier’s stats might read HP 40 and damage 6. Immediately following the stat adjustment, a brief message appears: “Enemy is battle-hardened” if `scale = 1` or “Enemy is enraged!” if `scale = 2`, warning the player that the fight will be especially challenging.

The CombatUI remains identical: “Attack,” “Block,” and “Inspect” buttons at the bottom. However, now when the player chooses “Attack,” the submenu’s slashing, piercing, and bludgeoning icons must be weighed carefully, and the boss also has an extra move that allows him to heal, with a similar warning to the strong attack, but this time turning blue instead of red before the move. Heal move restores 2 HP when his health dips below a certain threshold, the player must time an attack to suppress it. Hitting “Inspect” reveals a tooltip: “The cavalier’s horse is weak to piercing damage.” Since the cavalier’s

After the player’s turn, the cavalier’s AI chooses among three actions: WeakAttack (base damage), StrongAttack (baseDamage + 4), or Heal (restores 2 HP). This healing process uses up the cavalier’s turn, leaving him open to a retaliatory strike on the next round. If the player’s HP reaches zero at any point, GameOverMusic plays and the “Defeat” panel appears, allowing the player to press Space to return to the main menu. If the player depletes the cavalier’s HP to zero, a unique cinematic animation plays: the cavalier staggers, drops his sword, and kneels. VictoryMusic_Boss.mp3 swells while “Victory!” fades onto the screen. Pressing Space then returns to the main menu for closing credits and a final epilogue screen, where the player can choose to watch short cutscenes showing the museum’s exhibits in daytime light.

Back on the main menu, selecting “Logs” brings up a scrollable panel listing every exhibit plaque the player has opened during the current session. Each entry features a small icon of the artifact, the exhibit title in a bold serif font, and the first two lines of the plaque text in a slightly smaller font. When the player clicks any entry, the panel expands to fill the screen with the full exhibit description. This feature allows players to review historical details they may have missed or forgotten, especially useful before reattempting a quiz gate. A “Back” button at the bottom of the panel returns the player to the main menu.

The overall player flow begins at the main menu, where “New Game” leads into Overworld1. Here, the player practices walking, sprinting, and interacting until they discover the side gate to Museum1. Inside Museum1, the player reads plaques and answers multiple-choice quiz gates that unlock corridors and lead deeper into the museum. After clearing Museum1’s three gates, they descend to Castle1, where interactive plaques provide single-word answers to more advanced quiz gates. A roaming knight guards Gate 3 in Castle1; if the player wanders too close, they trigger Combat1, an introductory turn-based fight featuring basic attacks, blocks, and inspections. Victory in Combat1 opens the path to the Throne Room, where Combat2 awaits the player as a boss fight against a scaled cavalier, complete with a Heal move. If the player fails any combat encounter, they can press Space to retry that fight. Upon defeating the cavalier, VictoryMusic_Boss plays and the player is treated to a final epilogue before returning to the main menu.

Throughout all these scenes, the wrong-answer count maintained by DifficultyManager influences how tough each combat becomes, encouraging the player to pay close attention to exhibit text. The “Logs” screen on the main menu offers a convenient way to review historical facts, and the “Mute/Unmute” button lets players toggle sound at any time. By weaving together movement, interactive exhibits, quiz gates, turn-based combat mechanics, and dynamic difficulty scaling, this chapter demonstrates how the game creates an immersive, educational, museum-themed experience.

5. Game Design and Development

THIS chapter describes in detail how the game was constructed, from the overworld side-scroller and quiz gates to the turn-based combat system. The two sequence diagrams illustrate the flow of events when the player interacts with a quiz gate and when the player engages in combat. We begin by examining the overworld environment and its mechanics, then move on to combat, assets and animations, sound, and finally functional and non-functional requirements.

5.1. Overworld Side-Scroller

The overworld portion of the game is built as a 2D side-scroller with three distinct maps:

- **Tutorial Map** (a simplified introductory museum wing)
- **Main Museum Map** (the default museum space with historical exhibits)
- **Castle Map** (the medieval section where knight encounters occur, tied thematically to a special exhibit)

Each overworld map uses a pair of “infinite” scripts: `InfiniteBackground.cs` and `InfiniteFloor.cs` in the museum maps, and `InfiniteBackgroundCastle.cs` and `InfiniteFloorCastle.cs` in the castle map, to give the illusion of a much larger world than what fits on screen. As the player’s X-coordinate changes, these scripts dynamically instantiate background or floor tile prefabs ahead of the player’s direction of movement and destroy those that have moved off-screen behind. This approach allowed me to use freely available background art from itch.io without worrying about seam lines or loading multiple discrete scenes [9].

The `Movement.cs` script controls horizontal movement and sprinting. Pressing the A or D keys moves the player at a configurable walk speed, while holding Shift increases the speed to a sprint value. Pressing W triggers any object in front of the player tagged “Interactable,” such as quiz gates or collectible exhibits. The script uses `Rigidbody2D` physics for smooth acceleration and deceleration. Meanwhile, `CameraFollow.cs` smoothly interpolates the camera’s X-position toward the player’s position in `LateUpdate`, creating a clean, lag-free side-scrolling feel. By adjusting the damping value in the Inspector, I achieved a camera follow that neither snaps abruptly nor lags too far behind.

Quiz gates appear as stylized doorways in each map. When the player collides with a trigger zone, `GateScript.cs` (attached to an invisible trigger collider) calls `TriggerGateEnter(gateID)` on `GameEngine`. The engine then calls `FetchQuestion(gateID)` on the `GateSystem`, three answer options, and metadata (hints, penalty values). Upon receiving the question data via `ReturnQuestion(text, options)`, `GameEngine` displays the quiz dialog UI, pausing the museum or castle ambience (played by `OverworldSound.cs` or `MuseumMusicStart.cs`) and dimming the background.

When the player submits an answer, GameEngine calls `ValidateAnswer(answer)` on the GateSystem. If the result is “Pass,” GameEngine invokes `OpenGate(gateID)`, which triggers Door.cs to play an opening animation and disable its collider so the player can walk through. Simultaneously, ExhibitLog records the clearing by calling `MarkGateCleared(gateID)` for analytics. If the player fails, GameEngine displays “Try Again” for one second, and DifficultyManager.cs increments its internal `WrongAnswerCount` by one. This wrong-answer count will later feed into combat scaling. The sequence of events is shown in Figure 5.1, where dashed return arrows indicate asynchronous UI updates.

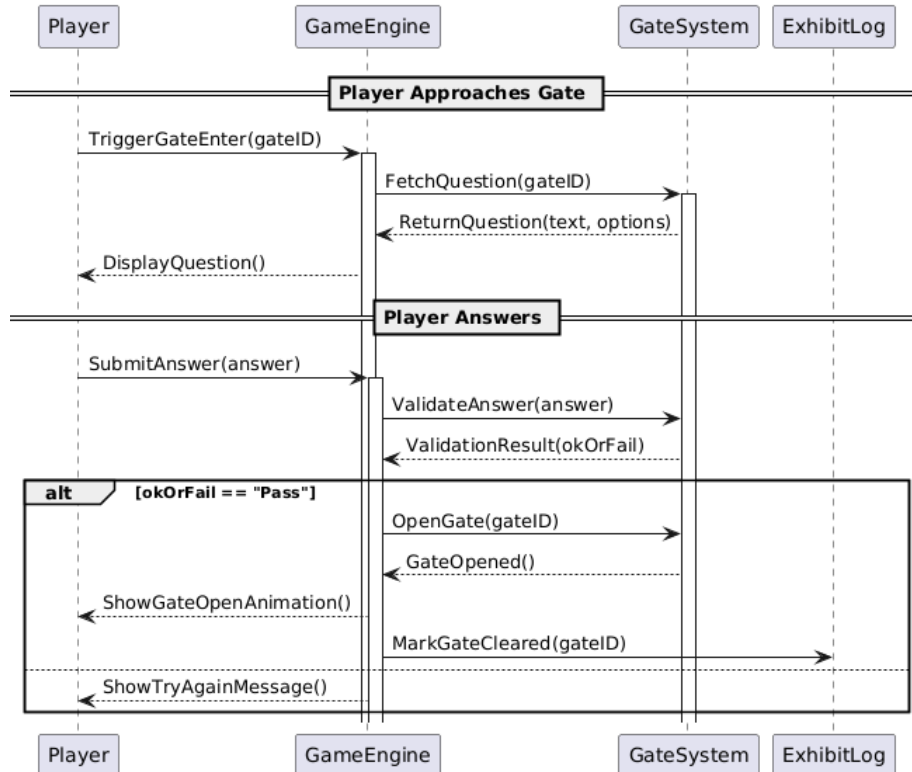


Fig 5.1 – Question gate sequence diagram

5.2. Turn-Based Combat

Once the player is transported to a combat scene (Combat1 or Combat2), the **TurnBasedCombat.cs** script governs all combat logic. There are two fights in total:

- **Tutorial Combat** – a roaming knight on the castle map (weak enemy prefab)
- **Boss Combat** – a cavalier miniboss at the end (stronger enemy prefab)

When the player enters a combat scene, Player calls `EnterCombat(sceneID)` on GameEngine. GameEngine responds by spawning the appropriate enemy prefab using `SpawnEnemy(enemyType)`. Once the enemy’s collider and health bar are in place, Enemy sends an `EnemyReady()` message back to GameEngine. At that point, GameEngine calls `ShowCombatUI()`, which displays the combat interface defined in `CombatUI.cs`. This UI includes the following buttons:

- **Fight** (opens a submenu with options Slashing, Piercing, Bludgeoning)
- **Block**

- **Inspect**
- **Back** (closes the fight submenu, returning to the main combat options)

Simultaneously, `CombatMusicManager.cs` ensures any still-playing overworld or castle audio stops, then begins to play either `Combat1Sound.cs` (tutorial fight) or `Combat2Sound.cs` (boss fight) on a loop. All audio transitions fade over 0.5 seconds to avoid abrupt cuts. The combat camera is then locked to a static position where the player and enemy face each other.

The combat loop continues while `Enemy HP > 0` && `Player HP > 0`. On the player's turn, they select an action:

- **Inspect**: calls `RequestWeaknessInfo()` on **Enemy**, prompting it to send back a weakness description via `ReturnWeaknessLore()`. **GameEngine** then calls `DisplayWeaknessLore()` so the player can plan a stronger attack.
- **Fight**: opens a submenu (`CombatUI`) where the player picks one of three damage types:
 - **Slashing** – strong vs lightly armored foes
 - **Piercing** – strong vs animals and opponents with shields
 - **Bludgeoning** – strong vs highly armored foes

When a damage type is chosen, **Player** calls `ProvideDamageType(dmgType)` on **GameEngine**, which calls `ApplyPlayerDamage(dmgType)` on **Enemy**. **Enemy** calculates `NewEnemyHP = currentHP – damageAmount` (applying any defense multipliers) and returns that value. **GameEngine** then calls `DisplayEnemyHP()` to update the enemy's health bar.

- **Block**: sets a boolean `isBlocking = true` in **GameEngine**, so when the enemy's next attack occurs, **GameEngine** calls `ApplyReducedDamage()` (halving the damage). An animation for raising a shield plays via `animator.SetTrigger("Block")`.
- **Back**: simply closes the fight submenu and returns to the main combat menu.

All player animations are controlled via `Animator` parameters in `TurnBasedCombat.cs`. For example, `animator.SetTrigger("Attack")` triggers the swing animation, and the state automatically returns to "Idle" once the animation finishes.

If the enemy survives, `GameEngine` calls `ChooseEnemyMove()`, and the AI randomly selects between two moves:

- **WeakAttack**: enemy sprite triggers `animator.SetTrigger("Attack_Weak")`, and the script calls `ApplyDamage(weakDamage)` on **GameEngine**, which deducts `weakDamage` from `playerHP`.
- **StrongAttack**: enemy sprite triggers `animator.SetTrigger("Attack_Strong")`, then **GameEngine** displays a brief warning UI (`SignalStrongAttackWarning()`) for 0.5 seconds, allowing the player a chance to have chosen `Block`. After the warning, **Enemy** calls `ApplyDamage(strongDamage)` on **GameEngine**.

If `isBlocking == true`, `GameEngine` calls `ApplyReducedDamage()`, otherwise `ApplyFullDamage()`. `GameEngine` then updates the player's health bar with `DisplayPlayerHP()`.

Next, `isBlocking` resets to `false`. At this point, the loop returns to the player's turn if `Player HP > 0`.

Once either side's HP reaches zero:

- If `Enemy HP ≤ 0`, `GameEngine` calls `VictoryScreen()` and plays `VictoryMusic.cs`. The victory screen waits for a Space key press to return to the museum overworld on the same map.
- If `Player HP ≤ 0`, `GameEngine` calls `DefeatScreen()` and plays `GameOverMusic.cs`. The defeat screen also waits for Space to either reload the previous checkpoint or exit to the main menu.

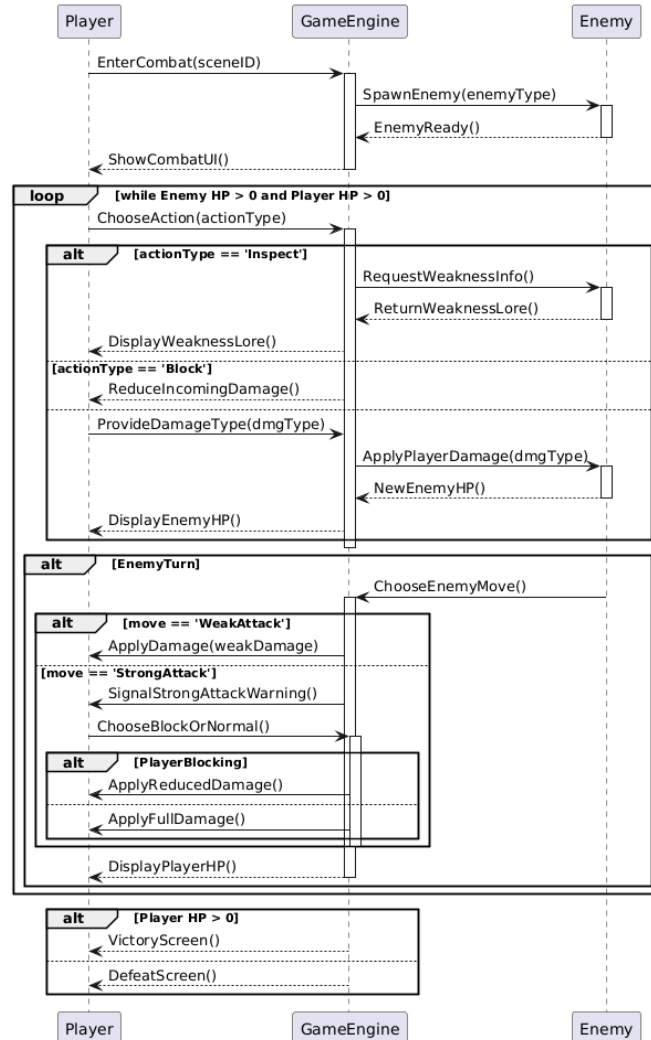


Fig 5.2 – Turn-based combat sequence diagram

5.3. Assets and Animations

All sprites, background art, character sheets, UI elements (buttons, health bars), and audio clips were imported from free, royalty-free collections on itch.io (Category: Free → Tag: Castle) [9]. These assets include:

- Backgrounds for museum and castle scenes (subfolders: `Assets/Sprites/Backgrounds`)
- Character sprites for player frames and enemy frames (`Assets/Sprites/Characters`)
- UI elements (buttons, health bar frames, choice panels, victory/defeat icons) in `Assets/Sprites/UI`

- Audio clips (MP3s for music and SFX) in Assets/Audio

I organized them into the following folders under Assets/:

- **Sprites/** (subfolders: Characters, Doors)
- **Audio/** (subfolders: Music, SFX)
- **Prefabs/** (player prefab, enemy prefabs, door prefabs, UI panels)
- **UI/** (UI, Backgrounds)

By dropping each file into the Unity Project window, Unity automatically created the associated .meta file. I then assigned the imported sprite sheets to the correct Sprite Mode (multiple for character sheets, single for UI icons) and sliced them accordingly.

For both the player character and enemy prefabs, I used Unity's Animator to build state machines:

Player Animator Controller:

- **Idle** (6-frame loop)
- **Walk** (10-frame loop)
- **Sprint** (10-frame loop)
- **Attack** (4-frame animation, then return to Idle)
- **Hit** (3-frame animation, then return to Idle)
- **Defeat** (5-frame collapse animation)

Transitions are controlled by triggers: e.g., `animator.SetTrigger("Attack")` and float parameter "Speed" (Idle ↔ Walk ↔ Sprint logic: if Speed > 0.1, go to Walk, if Speed > 1.5, go to Sprint). Exits from Attack states automatically return to Idle via an Animation Event at the last frame.

Enemy Animator Controller:

- **Idle** (4-frame loop)
- **Patrol** (4-frame walking loop)
- **Attack** (4-frame swing)
- **Hit** (2-frame flinch animation)
- **Death** (5-frame collapse and fade)

Transitions occur when `EnemyPatrol.cs` or `TurnBasedCombat.cs` calls `animator.SetTrigger("Attack_Weak")` or `animator.SetTrigger("Attack_Strong")`. After Death completes, the enemy's collider is disabled and the `GameObject` is destroyed.

For the quiz gates, **DoorAnimator** handles two states: **DoorClosed** and **DoorOpening**. When **Door.cs** calls `animator.SetTrigger("Open")`, the door's sprite plays an "opening" animation (3 frames) and then sets a Bool "IsOpen" to true to keep the final frame displayed.

All animation clips are set to "Loop Time" when they should repeat (walking, patrol, block stance) and to "Once" when they are one-off actions (attack swings, door opening, victory, death). This Animator architecture ensures fluid, believable transitions between player and enemy behaviors throughout the game.

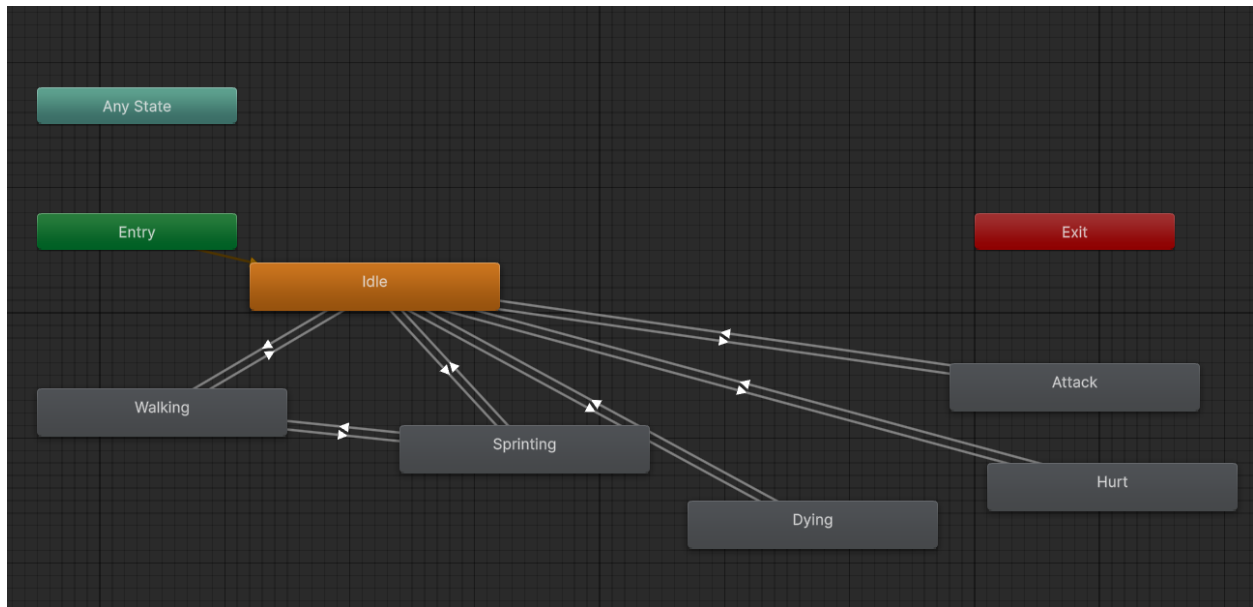


Fig 5.3 – Screenshot of Animator setup for player character

5.4. Sound

All background music and sound effects were imported as MP3 files from free online sources like pixabay.com [10]. Each scene uses a dedicated audio track:

- **MuseumMusicStart.cs** – plays a looping ambient museum track whenever a museum or tutorial map loads.
- **OverworldSound.cs** – crossfades to a subtle nighttime ambience in the museum overworld area.
- **CastleMusic.cs** – begins if the player crosses into the castle entrance trigger on the museum map.
- **CombatMusicManager.cs** – stops the current audio and plays either **Combat1Sound.cs** for the tutorial fight or **Combat2Sound.cs** for the boss fight, both looping until the battle ends.
- **VictoryMusic.cs** and **GameOverMusic.cs** – each play a short jingle once the fight is over, then fade out to black.

I created a singleton AudioManager GameObject with multiple AudioSource components (one for each track). By calling `audioSources[i].Play()` or `audioSources[i].Stop()`, each script pauses or starts the appropriate music. A 0.5 second crossfade routine smoothly adjusts volume on two AudioSources whenever a transition is needed. Lastly, **AudioToggle.cs** sets an `AudioMixer` parameter to mute or restore audio, allowing the player to disable sound at any time.

5.5. Dynamic Difficulty

To make combat encounters feel meaningful and directly tied to the player's learning, I implemented a dynamic difficulty system that adjusts enemy strength based on how many quiz questions the player answered incorrectly. From the very first quiz gate, every wrong answer increments a counter in **DifficultyManager.cs**. I placed the **DifficultyManager** script on a dedicated GameObject called **LoadScript**, then used `DontDestroyOnLoad(LoadScript)` so that this object and its wrong-answer count persists across scene changes. That way, once the player moves from the museum overworld into a combat scene, the difficulty information is still available without needing to reinitialize or pass data manually.

Here's how it works in practice: when the player submits an answer to a gate question, **GateScript.cs** calls `DifficultyManager.Instance.IncrementWrongCount()` if the answer is incorrect. The counter simply tallies every mistake until the next combat. Then, as soon as a combat scene loads, **TurnBasedCombat.cs** calls `DifficultyManager.Instance.CalculateScale()` to determine a scale value, zero if the player answered all questions correctly, one if they had one or two wrong answers, or two if they had three or more. That integer "scale" is then added to the enemy's base stats.

For example, if the player has answered three questions incorrectly, `CalculateScale()` returns 2, so a knight with base HP 5 and base damage 1 spawns with HP 20 and damage 4. In-game, a small "Difficulty Up" icon appears in the top-right corner whenever `scale >= 1`, and a brief message: "Enemy is battle-hardened" displays for two seconds before combat begins. This immediate feedback reinforces the connection between quiz performance and battle challenge: players quickly learn that studying the exhibit plaques carefully not only unlocks gates more easily but also keeps future fights manageable.

Because **LoadScript** uses `DontDestroyOnLoad`, I didn't need to write custom save/load code between scenes: the same **DifficultyManager** instance simply carries forward until the player either wins or loses a fight. After combat ends (regardless of victory or defeat), the counter remains intact for any subsequent battles, meaning early mistakes can snowball into tougher miniboss encounters unless the player opts to start a new game, in which case **NewGame.cs** calls `DifficultyManager.Instance.Reset()` to zero out the wrong-answer count. This approach keeps the implementation simple, avoids redundant data passing, and ensures that every quiz moment directly influences how the rest of the session plays out.

In summary, by anchoring my dynamic difficulty logic to a persistent **DifficultyManager** and linking it to quiz results, I created a system that feels organic: players who breeze through the museum face easier foes, while those who struggle with the content encounter stiffer resistance. This mechanic ties the educational and gameplay loops together, making history quizzes truly matter in the heat of battle.

6. Technologies and Tools

DURING the development of the project I have used several tools and technologies in order to achieve my goals. I have chosen Unity Engine as my game engine of choice, probably the most well-known one currently on the market. All of the code I have written for the game scripts was handled in Visual Studio as Unity has Visual Studio integration. I have also used Unity Hub and Plastic SCM Version Control in order to be able to test out features, without fear of breaking the current build of the game. In the last section I will also talk about other very popular game engines besides Unity on the market that I have considered using before starting the development of the project.

6.1. Unity Engine

The main technology used to create the game was my chosen game engine: Unity Engine version 6000.0.46f1, the latest version that came out this year [11]. While Unity is known for its strengths in the 3D space, I already had experience with 3D game development and wished to use this chance to learn 2D as well. I started with the Universal 2D Unity Template, that uses and already set-up 2D renderer from the Universal Renderer Pipeline that Unity is so well-known for.

Unity has quite an easy to use and intuitive UI that has greatly helped in the development process as I was slowly learning how the 2D environment behaves, as well as remembering game development basics from the last time I used Unity. The default Unity UI is neatly spread between 4 panels. The main panel in the center is the Scene panel in which you will do most of your work and preview exactly what the game looks like during each step of development. This is where you will move around and resize each asset, as well as where you will preview your game's functionalities as with a simple click of the "play" button, the game appears in this window, in its play-ready state for testing and previews. On the right side of the screen is the Inspector window, here you can easily preview any element of the game, add new components to it and modify their parameters in an easy to understand, code-light way. If you ever need to make changes to any element in your scenes, chances are you need to do it from here. On the top left side of the screen is the Hierarchy window. In this window you can see the entire structure of your project, neatly arranged into the different scenes you have created so elements don't overlap. Whenever you create a new element to add to your scene you will do it from here, as well as create hierarchies and children-elements that take the features of their parent element for easy grouping and making changes on the fly. By keeping all the elements properly named and organized it is quite easy to find them subsequently and make any changes or debugging on them as needed. The final Window on the bottom is the Project window. In here you will have easy access to all of your files and folder, as you can easily preview the entire structure of your project. Any Scenes, Animations and Scripts will also appear here, giving you an easy way to access any file you need, as well as create new ones during the development process.

During the development of the game, I created new elements constantly, as well as important free assets and music mp3 files from the internet. By simply dragging these outside

elements into the project folders, Unity automatically recognizes them and allows you to use them with ease as if they were already its own assets. While my project was not large enough in order to cause problems in loading, Unity also has a feature called Addressable Asset System, where it automatically keeps any assets that are not on-screen on a low memory usage, as well as identifies any large or otherwise high-resolution ones and pre-loads them in order to make load times between scenes as low as possible.

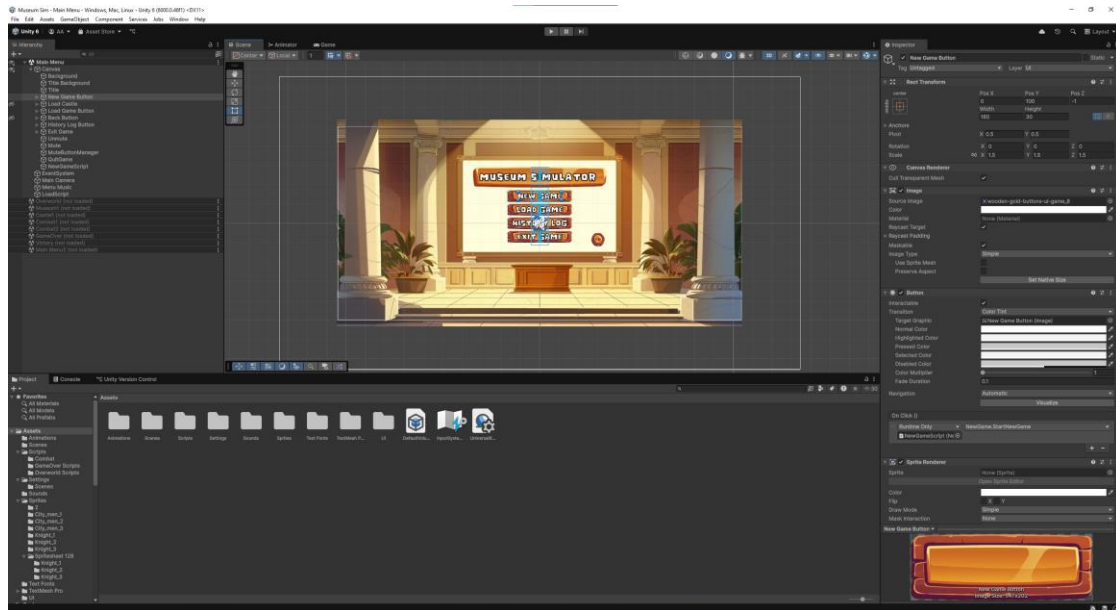


Figure 6.1 – A screenshot of the Unity UI in the project

A core part of the Unity development process is the scripts. Instead of making a huge monolithic script, each one is easily implemented as what is known as a MonoBehaviour, each outlining a certain feature of the implementation. For example, there might be a MonoBehaviour that only handles the sound or one that handles the player movement, each functioning completely independently of each other. All of the scripts are written in C# in Visual Studio which I will discuss below.

Unity also has easy-to-learn features for implementing sound and animations. You can easily import and introduce any commonly used sound file type through Unity's Audio Manager that allows an easy-to-understand way to customize it such as changing volume, reverberation, pitch etc. as well as attach it to an object for more realistic spatial sound and even have it automatically start at the start of the scene or when an action take place. Through the Unity Animator you can very easily create frame-by-frame 2D animations, spacing each frame at exact intervals to make cohesive movement. You are also able to create intuitive schemas with transitions between animation states where you can set exactly the condition in which they swap, for example, the animation goes from "moving" to "sprinting" if the "shift" key is pressed. I had no prior experience with such as system, but I found it quite easy to implement and also attach to the written scripts.

In conclusion, Unity has proven to be quite a useful tool, and I definitely understand why it is one of the most used game engines on the planet and I would also recommend it to any aspiring developers as it has many features to make it more approachable and intuitive, making the learning process that much easier. I had a great time using it, making the development process feel infinitely more engaging than if I was to only be writing code in an open terminal.

6.2. *Visual Studio*

All of the scripts for the project were written in C# and I have chosen Visual Studio 2022 as my IDE [12]. The decision was easy to make as Visual Studio has a very tight integration with Unity. It was also a familiar application that I have used many times before for game development as well as other tasks. It also presents a plethora of quality-of-life features that I used many times during the writing of the scripts.

IntelliSense is the auto-completion tool present inside Visual Studio, it helps suggest class names, method signatures and values in order to make the code writing take less time and also suggested built-in unity components that I might need for certain parts of the script. Overall, this made the experience quite a bit better, making it easier for me to adapt to C#, a language I'm not particularly familiar with, as well as make the whole process quite a bit faster as I didn't have to write the same variables hundreds of times by hand. It also comes with integrated refactoring tools, making it easy to modify pre-existing code as needed, using the "Rename Symbol" feature. When I changed the "HealthManager" script into "CombatManager" for example, I was easily able to update all the references, not only in every single script in the project, but also wherever it was mentioned and used in Unity as whole. This allows me to easily make changes without fear of forgetting a reference and making the process free of risk. The CodeLens feature also showed me exactly where every single method was used in the process, so I didn't accidentally break any gameplay logic when making changes or updating older methods.

Visual Studio also made it quite easy to debug code, since it is linked directly with Unity. By simply using the "Debug.log" statement, the debug log appearing under the main gameplay window as you test the game, and continuously updating itself in order to find exactly where the issue is. For example, it helped me understand exactly when the roaming enemy is supposed to spot and charge the player by allowing me to see exactly the distance between the player and the enemy at any given moment as well as give a warning when the player is spotted, allowing me to find exactly why it wasn't working and fix it.

In conclusion, Visual Studio is an easy-to-use application with many built-in quality-of-life elements, that I am quite familiar with and it has many integrations with Unity Engine, a few of which I mentioned here that make it invaluable in creating a Unity project. No other IDE comes even close to it when talking about developing games in Unity and I would recommend it to any newbie that wants to try out Unity Engine for the first time.

6.3. *Unity Hub and Plastic SCM Version Control*

In order to easily manage multiple versions of Unity projects I used a combination of Unity Hub and Plastic SCM Version Control [13]. Both applications were very important in creating, transferring and managing different project branches as well as for the simple utility of being able to work on the project on multiple devices, both on my main PC at home, as well as my university laptop when I'm on the go.

Unity Hub 3.12 is the current latest version of Unity Hub that I use on Windows 10 computers. It is a general hub where you can manage existing projects, create new ones, view premade Unity templates, access files and folders, and update Unity Engine versions as needed during the development process. From Unity Hub you can easily launch any project with a simple click. All of the project-specific overrides like custom Editor paths and license selections are

automatically handled by it, preventing any meta-file conflicts and ensuring that the project works properly on every single device it is accessed on from the get-go. It also has an intuitive interface where you can easily view all of your downloaded projects, make modifications to them, create new projects or import others from other devices with a simple click, check all currently installed Unity version on you device and even check community features such as the Unity Asset Store for easy access to official and community assets as well as community discussions in order to ask questions or find tips to solve certain problems.

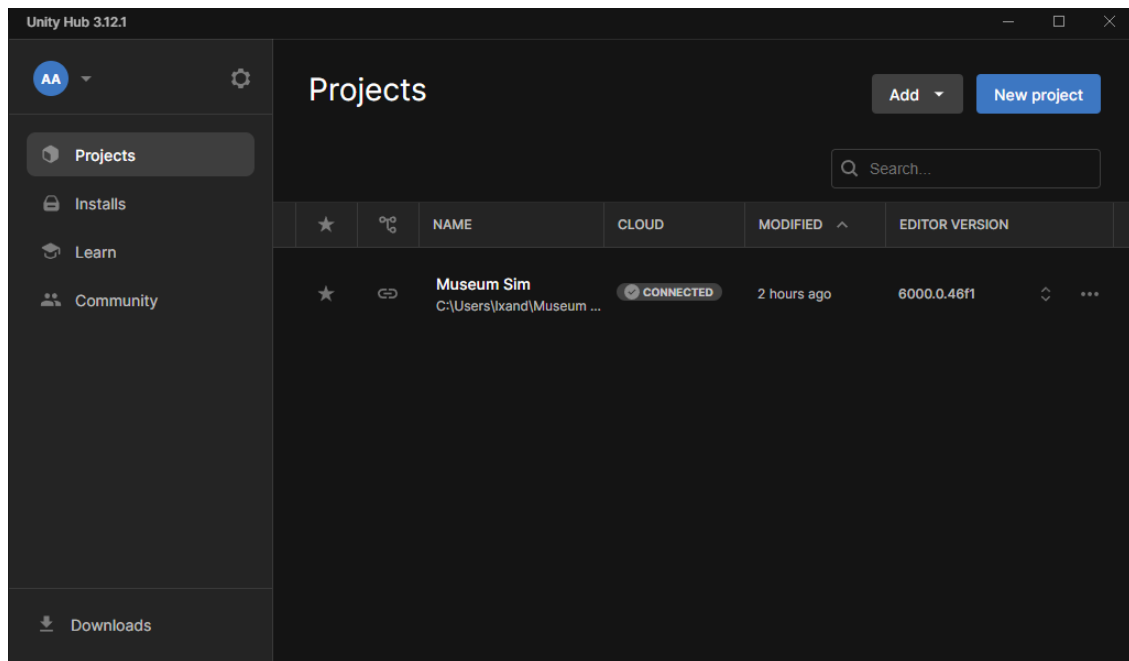


Figure 6.2 – A screenshot of the Unity Hub UI and the current projects on my device

For version control, I used Unity Version Control, more well-known as Plastic SCM, Unity’s integrated version control system. I chose this specifically because it handles textures, audio clips and tiled maps more gracefully than git and it is also aware of all the Unity-specific file types like “.unity” and “.asset”. It also has integration in the Unity UI where you can easily change between branches, merge into the main branch and see exactly which files are being kept updated in the current branch without ever leaving the project. It also automatically ignores transient folders like “Library/” or “Temp/” and tracks all the essential folders like “Assets/” and “Plugins/”, removing the need for the user to manually ignore certain files. When renaming or moving assets in the Unity Editor, Plastic SCM updates both the assets and its meta file, preserving references in order to prevent meta-file mismatch.

I implemented a branching strategy during the development process, with a main branch for all the stable versions of the game and smaller branches for developing each feature. Each of these feature branches allowed me to build and test each new mechanic for the game without fear of destabilizing the main branch or causing any lasting damage. While this project was solo, if working in a team, it makes it really easy for multiple people to be working on different features at the same time without inconveniencing the others in any way. Once a feature branch is fully completed it is easy to merge it back into the main branch with a simple button press and it also generates a merge changeset allowing you to see every single change that has been made.

Plastic SCM also has a visual merge tool that displays a tree-based diagram in order to show exactly where all the branches were started and merged back in order to keep a tidy history

of every single branch made and when it was finished. Plastic SCM also has a feature called delta compression that is meant to keep the remote repository size manageable, another advantage it possessed over Git. It also has graphs that allow for viewing and accessing every single element of the project history from changes to branches and merges. Together Unity Hub and Plastic SCM created a cohesive environment for me to continue the development of the project, helping me easily make updates, changes and transfer the project among devices, making for quite the streamlined development pipeline.

6.4. Other Game-Making Platforms

Although, I ultimately chose Unity as my engine of choice, I did evaluate the options checking other widely-known engines on the market. The main contenders for the project besides Unity Engine were Unreal Engine as well as Godot. In each case I assessed their 2D game making capabilities, the complexity of their scripts and community resources available. While I ultimately did not choose to go with these in the end, they are great tools for game development and they deserve a mention here.

While Unreal Engine is one the most highly used game engines in the world, it had glaring problems [14]. While yes, Unreal engine does support 2D through its Paper2D in order to make games such as side-scrollers and top-down ultimately it is an engine with a major focus on 3D and it shows in the way the performance is handled when making a 2D project and the lack of specific features present in Unity for 2D game development. Also, while yes, the powerful C++ API used by Unreal Engine is a top choice for creating major AAA titles, I have found that it is not good for small games such as mine. Firstly, even small features such as a side-scrolling background or camera mechanics made the project size quickly balloon over hundreds of MB, and made the performance a serious problem even for a game such as mine. In conclusion, I found that for a project of this scale, Unreal Engine is not optimal due to a lack of 2D development features and a huge amount of space and performance requirements in relation to the amount of content created.

On the opposite side of the spectrum, Godot Engine is an open-source project that has been growing rapidly in popularity in the recent years [15]. It has a lightweight runtime, a Python-like script and quite an intuitive scene structure. The main problem with it is it lagged behind Unity's robust support and massive community. While at the beginning it seems comparable, instability quickly grows as the game becomes larger and more complex scripts as added. Also, the many features I have grown to love from a mainstream engine such as Unity are completely absent in Godot, making for a more arduous development experience with a larger amount of hands-on work. Godot is also a lesser-known engine so it is harder to go on the internet for tips or to try to find help when you run into problems during development as not many people use it comparatively. It also requires a bunch of third-party software in order to be able to implement many features, making the experience quite cumbersome since you have to install and get familiar with so many different applications. All of these being said, I came to the conclusion that, for a long-term choice Unity is a more reliable choice, as it has more quality-of-life features as well as more community support and performance for integrating more complicated features.

In summary, I wanted to outline in this chapter all of the technologies and tools that I researched and used in order to complete the museum RPG project, from the engine I used, to the IDE of choice for writing the code in the scripts, to the version control system I chose in order to keep the project from having any fatal errors that may break it completely. I also wanted to mention

other engines that I considered, as they all have their own strengths, but ultimately, for my needs I chose to use Unity, and I urge any other person who wants to realize a similar project to do the same.

7. Conclusion

REFLECTING on the past several months, I'm proud to say that I've succeeded in creating an educational game that's both engaging and effective for learning. From the very start, I envisioned a museum-inspired side-scroller where players explore, answer quiz questions, and then see the results of their learning in turn-based combat. Watching those pieces come together in Unity was incredibly rewarding. When I see a player click on a gate, answer a history question, and then face off against a knight whose strength is influenced by how well they did, I know I hit my goal: combining fun gameplay with genuine learning.

Working in Unity became a daily joy. I spent countless hours importing free assets from itch.io, placing backgrounds and floor tiles to create an infinite scrolling world, and then configuring the Animator to bring both the player and enemies to life. Tweaking sprite colliders, making sure camera follow felt smooth, and optimizing performance so the game stayed at 60 FPS taught me a lot about the nuts and bolts of game development. Even debugging small audio crossfade issues making sure the museum music faded out just as combat music kicked in, became a lesson in attention to detail. Each little fix and improvement reminded me how many skills are needed to build even a relatively small game.

Beyond the technical side, this project helped me grow in terms of planning and iteration. Setting clear milestones, checking in regularly with my coordinator, and running internal alpha tests taught me the importance of feedback loops. Seeing players stumble over a quiz question or miss a combat button prompted me to adjust UI contrast or update the "Inspect" tooltip so it's more obvious. Those moments of user testing felt like small victories every time I resolved an issue and made the experience smoother.

Looking ahead, there's so much I'd love to add. First, I plan to expand this museum with extra themed wings: an Ancient Egyptian section full of hieroglyphic puzzles, a Jurassic exhibit where players dodge dinosaur animatronics while answering fossils trivia, and more. I'd also like to introduce recruitable companions: NPCs who, once freed by answering certain quiz questions, join the player's party in battle. Their own stats could scale with the player, creating interesting strategies for tougher fights. And in the farther future, I dream of a full VR version, imagine walking through a virtual museum, reaching out to interact with exhibits and then stepping into a 3D combat arena. A VR "museum of history" could be a truly immersive learning environment.

All in all, this project showed me that a lone developer can still create something meaningful and polished. I met my goal of blending education with fun, and along the way I grew as both a designer and a coder. I enjoyed every challenge and celebrated each milestone, from conceptual sketches to the final beta build. In the end, I built not just a game, but a proof of concept: with creativity and dedication, an educational title can be both entertaining and instructive. For anyone who plays it, I hope it's both an enjoyable adventure and a spark for deeper curiosity about history.

Bibliography

- [1] The VR Museum of Fine Art. Available online: https://store.steampowered.com/app/515020/The_VR_Museum_of_Fine_Art/ (accessed on 03 May 2025).
- [2] Study.com. Behaviorism in the Classroom: Gamification. Available online: <https://study.com/academy/lesson/behaviorism-in-classroom-gamification.html> (accessed on 15 Mar 2025).
- [3] University at Buffalo Center for Teaching and Learning. Constructivism. Available online: <https://www.buffalo.edu/catt/teach/develop/theory/constructivism.html> (accessed on 17 Mar 2025).
- [4] Eurogamer. Persona 5 Royal Test Answers & Exam Quiz. Available online: <https://www.eurogamer.net/persona-5-royal-test-answers-exam-quiz-7022> (accessed on 10 Apr 2025).
- [5] PC Gamer. Darkest Dungeon 2 Tips. Available online: <https://www.pcgamer.com/darkest-dungeon-2-tips/> (accessed on 15 Apr 2025).
- [6] Steam. Kingdom Come: Deliverance II. Available online: https://store.steampowered.com/app/1771300/Kingdom_Come_Deliverance_II/ (accessed on 20 Apr 2025).
- [7] JGeekStudies. Mondo Museum. Available online: <https://jgeekstudies.org/2019/10/30/mondo-museum/> (accessed on 27 Apr 2025).
- [8] Online Gantt. Available online: <https://www.onlinegantt.com> (accessed on 01 May 2025).
- [9] itch.io. Free Castle Game Assets. Available online: <https://itch.io/game-assets/free/tag-castle> (accessed on 02 May 2025).
- [10] Pixabay. Free Music Tracks. Available online: <https://pixabay.com> (accessed on 05 Mar 2025).
- [11] Unity Technologies. Unity Documentation. Available online: <https://docs.unity3d.com> (accessed on 05 Mar 2025).
- [12] Microsoft. Visual Studio Documentation. Available online: <https://docs.microsoft.com/visualstudio> (accessed on 06 Mar 2025).
- [13] Unity Technologies. Unity Hub Documentation. Available online: <https://docs.unity3d.com/hub> (accessed on 07 Mar 2025).
- [14] Unreal Engine. Available online: <https://www.unrealengine.com/en-US> (accessed on 10 Feb 2025).
- [15] Godot Engine. Available online: <https://godotengine.org> (accessed on 10 Feb 2025).

Appendix 1

Survey questions:

What is your age group? *

- ☐ Under 18
- ☐ 18-24
- ☐ 25-34
- ☐ 35-44
- ☐ 45-54
- ☐ 55-64
- ☐ 65 or older

What is your gender? *

- ☐ Male
- ☐ Female

How Familiar are you with History? *

- ☐ Not at all Familiar
- ☐ Slightly Familiar
- ☐ Somewhat Familiar
- ☐ Very Familiar
- ☐ Extremely Familiar

How do you prefer to learn? (Select all that apply) *

- ☐ Reading books or articles
- ☐ Watching videos (documentaries, YouTube, etc.)
- ☐ Hands-on activities (labs, experiments)
- ☐ Interactive experiences (games, simulations)
- ☐ Listening to lectures or podcasts
- ☐ Group discussions or debates

How do you feel about using technology to learn (e.g., educational apps, games, virtual environments)? *

- ☐ Very positive
- ☐ Positive
- ☐ Neutral
- ☐ Negative
- ☐ Very negative

Would you consider using a video game as a way to learn about history? *

- ☐ Yes, definitely
- ☐ Maybe, depending on the game
- ☐ No, I would prefer other methods of learning

How much do you enjoy learning through video games or gamified experiences? *

- ☐ I love it, it's my preferred method of learning
- ☐ I enjoy it, but it's not my main method of learning
- ☐ I'm neutral, it's okay but not my go-to
- ☐ I don't enjoy it much
- ☐ I don't enjoy it at all

What type of game mechanics do you enjoy the most in educational games? (Select all that apply) *

- ☐ Puzzles and problem-solving
 - ☐ Narrative-driven storylines
 - ☐ Exploration and discovery
 - ☐ Combat or action-based challenges
 - ☐ Collecting items or achievements
 - ☐ None of the above
-

What types of museums do you enjoy visiting the most? (Select all that apply) *

- ☐ History museums
- ☐ Science museums
- ☐ Art museums
- ☐ Natural history museums
- ☐ Technology museums
- ☐ Other...

Do you typically enjoy reading exhibit labels or plaques when visiting a museum? *

- ☐ Yes, I find them informative and interesting
- ☐ Sometimes, if they're engaging or relevant
- ☐ Not really, I prefer a more interactive experience
- ☐ No, I prefer not to read them

Would you consider visiting a museum more often if it had interactive digital exhibits (e.g., virtual tours, AR experiences)? *

- ☐ Yes, definitely
- ☐ Maybe, if the experience is engaging
- ☐ No, I prefer physical exhibits

How likely would you be to try a virtual museum experience from home (e.g., interactive exhibits, virtual tours)? *

- ☐ Very likely
- ☐ Likely
- ☐ Neutral
- ☐ Unlikely
- ☐ Very unlikely

Would you prefer a physical museum experience or a virtual one? *

- ☐ I prefer physical museums
- ☐ I prefer virtual museums
- ☐ I like both equally
- ☐ I don't visit museums often enough to have a preference

How interested are you in the idea of using a game-based approach to explore historical events (e.g., a turn-based RPG set in a historical museum)? *

- ☐ Very interested
- ☐ Somewhat interested
- ☐ Neutral
- ☐ Not very interested
- ☐ Not interested at all

Would you be more likely to engage with historical content in a game that incorporates elements like quests, rewards, and interactive narratives? *

- ☐ Yes, that sounds exciting
- ☐ Maybe, depending on how it's presented
- ☐ No, I prefer standard learning methods

What do you think are the most important aspects of a gamified learning experience? (Select all that apply) *

- ☐ Engaging story and narrative
- ☐ Rewards and achievements
- ☐ Fun and interesting gameplay
- ☐ Learning through challenges and puzzles
- ☐ Realistic historical accuracy
- ☐ Other...

What would make you more likely to engage in a virtual gamified historical experience (e.g., a museum RPG)? *

- ☐ A compelling story or historical narrative
- ☐ The ability to explore and interact with artifacts
- ☐ Rewards or achievements for completing challenges
- ☐ A combination of learning and entertainment (edutainment)
- ☐ Social aspects (e.g., competing or collaborating with friends)
- ☐ Other...

What art styles do you prefer for educational games? *

- ☐ Cartoonish and colorful
- ☐ Minimalistic and clean
- ☐ Realistic and detailed
- ☐ Retro and pixel art
- ☐ Hand-drawn and sketchy
- ☐ Other...