Technological Educational Institute
of Crete

Department of Informatics Engineering

Bachelor Thesis

Development of a Serious Educational Game for Special Education

Anastasios Kristofer Barianos (A.M. 3936)

Supervising Professor: Dr. Vidakis Nikolaos

Acknowledgements

I would like to express my gratitude towards my supervisor Dr. Nikolaos Vidakis, for his guidance and trust. I would also like to warmly thank Georgios Xanthopoulos for providing the 3D Models for the game, Stavros Charitakis for the cooperation for IOLAOS and the team at NiLE Lab for all their help. Additionally, I thank my father and grandfather for their moral support. Finally, I am grateful to Areti Manolaki, for providing perspective and constant encouragement.

Abstract

Ancient theater is one of the first forms of art humans partook in and as such has been a significant contribution in cultural heritage. Through the years it has evolved, starting from religious ceremonies and reaching to entertainment and education. On this path it has influenced politics and societies, while also being a force for development in multiple fields, including architecture, writing, and engineering. Despite the tremendous significance, most of the legacy is lost. Only a fraction of dramas from the past is saved to this day, builds are not in their former glory and the intricate machines are only known through references. Most of our knowledge doesn’t come from artifacts, but from extensive studying.

In education, ancient theater is taught mainly in a literal way, studying the dramas with traditional means. However, these means are challenging and irrelevant for learners that grow up as digital natives. This is especially the case for students with special educational needs that struggle with traditional learning methods. Inclusion of interactivity and playfulness in the learning process is essential to trigger these students and to create environments for experiential learning, achieving dedication to the educational process.

This thesis has utilized 3D artifacts that represent components of an ancient theater based on bibliographic references, creating a three-dimensional experience in a virtual depiction of an ancient theater. Towards this goal analysis, design and pilot development of the whole area of an ancient theater was also conducted. First person mechanics have been implemented and full interaction between played and 3d artifacts was developed. More over a set of information was chosen about these artifacts and multiple quizzes were created to ensure the learning processes.

Σύνοψη

Το αρχαίο θέατρο αποτελεί μια από της πρώτες μορφές τέχνης που ανέπτυξε ο άνθρωπος. Για αυτό και αποτελεί σημαντικό κεφάλαιο της πολιτισμικής κληρονομίας. Στο διάβα του χρόνου εξελίχθηκε, ξεκινώντας από θρησκευτικές τελετές και καταλήγοντας σε μέσο αναψυχής και εκπαίδευσης. Σε αυτήν την μακρά πορεία επηρέασε πολιτικές και κοινωνίες ενώ αποτέλεσε ερέθισμα για ανάπτυξη σε πολλαπλά πεδία, όπως η αρχιτεκτονική, η συγγραφή και η μηχανική. Παρά την σημαντικότητα της, το μεγαλύτερο μέρος της κληρονομίας έχει χαθεί. Ελάχιστα κείμενα της αρχαιότητας σώζονται μέχρι σήμερα, τα κτίρια δεν σώζονται στην καλύτερη κατάσταση και οι περίπλοκες μηχανές μας είναι γνωστές αποκλειστικά από αναφορές. Οι περισσότερες γνώσεις μας δεν προέρχονται από αντικείμενα αλλά από εκτεταμένη έρευνα.

Στην εκπαίδευση, το αρχαίο θέατρο διδάσκετε κυρίως στη λογοτεχνική του διάσταση. Τα αρχαία δράματα μελετώνται με παραδοσιακές μεθόδους. Ωστόσο, τέτοιες μέθοδοι δυσκολεύουν τους νέους και απέχουν από την πραγματικότητα των μαθητευόμενων που ανήκουν στην ψηφιακή γενιά. Ακόμα περισσότερο σε περιπτώσεις μαθητευόμενων με ειδικές εκπαιδευτικές ανάγκες οι παραδοσιακές μέθοδοι ενέχουν ακόμα μεγαλύτερη δυσκολία. Η ενσωμάτωση της δια δραστικότητας και του παιχνιδιού στην εκπαιδευτική διαδικασία είναι απαραίτητη ως έναυσμα στους μαθητές και για την δημιουργία ενός περιβάλλοντος βιωματικής μάθησης, πετυχαίνοντας συμμέτοχή και αφοσίωση στην εκπαιδευτική διαδικασία.

Η παρούσα εργασία έχει χρησιμοποιήσει τρισδιάστατα αντικείμενα για την απεικόνιση τεχνουργημάτων και δομικών στοιχείων του αρχαίου θεάτρου βασισμένα σε βιβλιογραφικές αναφορές. Έτσι δημιουργείτε μια τρισδιάστατη εμπειρία εντός μιας εικονικής αναπαράστασης ενός αρχαίου θεάτρου. Προς αυτήν την κατεύθυνση διεξήχθη επίσης ανάλυση, σχεδιασμός και πιλοτική ανάπτυξη ολόκληρης της περιοχής ενός αρχαίου θεάτρου. Εφαρμόστηκαν μηχανικές πρώτου προσώπου και αναπτύχθηκε πλήρης αλληλεπίδραση μεταξύ παικτών και τρισδιάστατων αντικειμένων. Επιπλέων, επιλέχθηκε ένα σύνολο πληροφοριών σχετικό με τα τεχνουργήματα και πολλαπλές ερωταπαντήσεις δημιουργήθηκαν για να εξασφαλίσουν την εκπαιδευτική διαδικασία.

Table of Contents

List of Figures

List of Tables

List of Acronyms

2D	Two Dimensions
3D	Three Dimensions
API	Application Programming Interface
AR	Augmented Reality
CDS	Common Data Space
GUI	Graphical User Interface
HUD	Heads Up Display
HTTP	HyperText Transfer Protocol
JSON	JavaScript Object Notation
RGB	Red Green Blue
VR	Virtual Reality
xAPI	Experience Application Programming Interface
XML	eXtensible Markup Language

1 Introduction

Serious games are not clearly defined, even though the term was first stated in 1968 by Clark Abt.[1] [2]. According to him, such games have a clear educational direction while entertainment is irrelevant, but can be present[2]. Building on that definition, Sorensen & Meyer[3] expresses the need for educational design while Zyda[4] expands the definition to include “government or corporate training, education, health, public policy, and strategic communication objectives”. However, there is an agreement, that serious games are games that consist of an objective that serves learning, in any sense[5].

In recent years video games have been evolving rapidly, maturing in content and growing in popularity. What once was a hobby of few people, has grown into one world-wide industry. A shining example would be the video game industry of Sweden, a traditionally industrial country, which has spectacular growth in the last few years, having increased the revenue from 2010 to 2015 as much as 1000%. [6]. In this context, the market for serious games is also rapidly rising with an expected growth of about 16% annually between 2015 and 2020[7]. Under these circumstances negative prejudice against games have faded and their value as educational tools is shining brighter than ever before. Serious games, but also games designed solely for entertainment, have found their way into classrooms, as teachers are exploring innovative ways to pass on knowledge, give students a chance to express their creativity, learn to work in teams and acquire skills much needed in the new age of technology.

Children today familiarize themselves with digital and mobile technologies from a very young age and grow to be natives both in the cyberspace and the physical world. In this setting, serious games are a crucial part of education, as they are offering substantial advantages for the learning process but also as they utilize more of the students’ skills.

The inclusion of serious games actively in the educational framework facilitates the learning and teaching process. Video games have indisputable importance within learning for all academic levels [8]. They offer speed, easy memorization due to gamification and experiential learning, the ability to process the offered knowledge and to compose new. They enable students of all levels to improve their skills and learn amid playing. Students may be assisted to understand topics and broaden their horizons without being restricted by conventional teaching resources. At the same time new horizons are opened for people with special needs, as they are provided with equal learning opportunities while also improving daily life, independence levels and self-advocacy[9][10].

The use of serious games on the subject of special education is especially interesting with several advantages and great prospects. Initially, special education is a very large branch with many different cases, of which each one needs individual treatment. It includes a wide range of difficulties that cause problems during the learning process[9]. Serious games greatly assist in acquiring and developing practical, organizational and problem solving skills[8] , all of which are often challenging to students with special abilities. Furthermore, the environment of games created with special ability students in mind is controlled and predictable, allowing learners to concentrate on the activity without distraction of attention[11]. Jordan & Powell [12] suggests that computers are the most appropriate means of teaching people with autism since for various reasons including delineated conditions, limited sensory stimuli, predicable behavior and lack of punishment for wrong answers. Similarly, learners with dyslexia benefit from the optical and auditory nature of games, those with dyscalculia can understand relations between number and get familiar with basic numerical operations through simulation games.[13]. It becomes clear that the nature of serious games can benefit all learners, regardless of any special conditions. In this context, we are trying to provide educational tools to teachers that will assist them in the teaching process as well as students in learning.

Ancient Theater, as a subject of study, is valuable to children, as cultural, historical and philosophical stimulus. However, as part of schoolwork within the context of general education, knowledge of ancient theater is offered only through conventional and outdated media. Consequently, students' comprehension of the topic is limited by the grasp and viewpoints of authors, teachers, and the medium's limitations. By offering knowledge in ludic ways, contact with the ancient theater as a structure in a self-containing digital world and autonomy within that world, enables children to compose new knowledge and develop their skills, through experiential learning. This way students come into contact with cultural elements and the society of that time and extend them with the guidance of the educator to the modern reality. In the context of special education, it is possible to optimize children's skills such as synchronization, eye-hand coordination, fine motor skills and acquire knowledge tailored to their own needs through this game.

1.1 Main Goal

The goal of this thesis was to develop a serious educational game for primary and secondary school students, including students with special educational needs, by exploring, utilizing and extending the capabilities of the IOLAOS platform[14], in order to customize the game to suit each player. The game would assist students to extend their knowledge about ancient theater and provide an opportunity for experiential learning through playfulness.

Therefore, a three-dimensional world was crucial for the experience and a collection of tools needed to develop the game. Unity3D was chosen as a game engine, all 3D models where created with Blender by the NiLE lab team, IOLAOS platform was utilized for profiling, personalization of the game and its educational capabilities.

Consequently, the resulting game is a depiction of an ancient theater in three dimensions, in which the user is free to roam and interact with objects, through which he/she gathers information and is presented with a set of questions to answer. By utilizing the IOLAOS platform, crucial settings are set to values that suits each user, while live sessions can be arranged by a teacher for a class to play at the same time, while the teacher can supervise at real time or gather information after the session is complete.

1.2 Motivation

Ancient tragedies and comedies are taught in schools as literature. However, the art of ancient theater reaches much further than literary merit. Architecture and tools were essential to the narrative of theatrical plays, nevertheless their significance in creating the experience is overlooked. Moreover, few ancient theaters exist to this day, and visiting them can be challenging for many schools in Greece, and impossible for many schools world-wide. As a result, students do not perceive ancient plays as one unified experience and art-form, but simply as a literary work, and therefore lack knowledge and comprehension of the means that were used in ancient times. Furthermore, for students with special educational needs repetition and hands on experience are vital to learning and understanding anything, but the classroom is not always able to facilitate such repetition.

The advancements in gaming technologies have made game creation approachable, untying projects from heavy financial dependencies. As a result, it is sustainable to create educational games that enhance learning through experiences.

Therefore, we aimed at digitally creating an ancient theater and giving students the opportunity to freely walk around, explore and interact, thus getting familiar with the architecture and the tools and gaining experiences that are otherwise not available to them. This level of freedom, paired with repeatability, assist students with special need to develop initiative in a ludic way that will not frighten them while also ensuring proper conditions for every learner to explore and understand at their own pace.

1.3 Objective

The main purpose of this thesis was to create a serious game for students with special educational needs that would use the IOLAOS platform in order to supply each user with the best possible conditions for experiential learning. The theme of Ancient Greek Theater was chosen as the topic is very theoretical, lacking actual relation to children's understanding and therefore could benefit greatly by an approach to complement schoolwork. Additionally, a 3D world was deemed crucial, as exploring such a world assists in ludic play, makes the experience more relevant and enables children to develop and expand skills such as hand-eye coordination, fine motor skills and many more which are proven to be difficult for children with special educational needs.

1.4 Outline

Chapter 1 "Introduction": This chapter includes a brief introduction as well as a summary, motivation and objectives for this thesis, and an outline of the chapters included.

Chapter 2 "Cultural Heritage Learning Environments and Games": In this chapter learning in cultural heritage assisted by new technologies is studied, as well as the state of the art of virtual tours and games, and the technologies used within the field. The technologies used to develop ThimelEdu are briefly mentioned and finally a thesis plan to achieve the goals is also laid out.

Chapter 3 "Educational Game ThimelEdu": In this chapter game and supporting software analysis and design are conducted. The data acquisition process for ThimelEdu is described and the extracted requirements presented. Furthermore, a detailed analysis of the game and supporting software design are portrayed.

Chapter 4 "Pilot Implementation": In this chapter the implementation of each modality is described. Preferences and settings for the game are explained and examples are provided. Finally, the resulting game is presented.

Chapter 5 "Results": The final chapter includes our conclusion from our research and the efforts to develop ThimelEdu, as well as our future goals for the project.

2 Cultural Heritage Learning Environments and Games

The evolution of smart mobile devices and multimedia have been a crucial factor for development of applications for museums and cultural heritage in general [15]. Digitization and creation of virtual worlds have been utilized to bring cultural heritage and history to broader audiences, offering immersive experiences from the past [16]. Digital museums and virtual tours have appeared to offer the opportunity to explore specific sites and findings by audiences that cannot visit the physical sites and museums. Digital representations have offered the opportunity to interact with objects without any risks while also offering new insights for archaeologists. Finally, some museums have developed applications to be used by visitors within the museum to gather more information, be presented with reconstructions etc.

Ancient theater interests' individuals and experts for various background reasons, including, but not limited to, history, culture and philosophy. As an art form it has evolved and provided the inspiration for a variety of art forms and pieces of art. As a topic with such interest build around it, numerous studies have been conducted while the principals and some plays are being taught to students around the world. The interest might not be as wide as for modern theater, but it is still taught in higher levels of education worldwide, and in Greece it is part of elementary school as well.

However, besides the large interest, not much have been done to bring theater closer to children and to detach it from written media. The advancements within digital museums and applications related with museums have not been brought into ancient theater yet. We were only able to find some educational games regarding cultural, societal and historical details about ancient theater as an event, made by the Greek ministry of culture and sports.

2.1 Learning in Cultural Heritance

Virtual tours, digital museums and representations do offer new opportunities, but they offer information to the audience in a passive way. For better learning an active state must be achieved, and for that reason serious games are utilized as they can engage the audience with entertaining, fascinating and educational content.[17]. Quite a few serious games and other digital media have been created to support learning of various topics within cultural heritage and for usage within a multitude of settings, supported either by schools, museums, archaeological sites, or none. However, we will focus on some of them that are closest to the subject of ancient theater.

2.1.1 Games & Virtual Tours

Web Games from Greek Ministry of Culture

The Greek ministry of culture and sports has a website[18] devoted to ancient theater, with resources for students and teachers in various forms. Included in those resources, are four educational games about varied topics, all sharing ties to ancient theater and its god, Dionysus.

- **What is hidden under the city?**
  Concerning the excavation and restoration of the Ancient Theater in Larissa.
- **The celebration of the great Dionysia**
  A tour of the buildings of ancient Athens, with a theme of the great Dionysia.
- **A modern troupe in the ancient theaters of the world**
  Travel through Greece and the World to find theaters that are still used today
- **Dionysus and his friends**
  Regarding the art on vases and the followers of Dionysus.

All games share a simple design philosophy, tailored for children, with ludic aesthetics in a 2D setting. Each game provides information regarding its subject in the form of text, on which then the player is expected to act in order to solve puzzles and learn more about the subject.

eShadow

Traditional shadow theater has been part of numerus cultures and stays popular to this day. As a story telling device, shadow theater is valuable in education and is actively used in Greek Education. Inspired by the educational value, as well as the communicational opportunities and the traditional element of shadow theater, a team at the Technical University of Crete created a collaborative platform, eShadow, with a set of tools for creating, sharing and watching shadow theater plays. Through this platform the creators hope to preserve, modernize and popularize traditional art, assist the learning procedure both inside and outside the school context, as well as to encourage collaborative learning and creativity expression, for children and adults.[19] With use of the platform, plays that the users enjoy can be recreated, scenarios from other arts can be transformed to a shadow theater narrative, or the users can create their own scenarios.

eShadow consists of a 2D interface, where the user can either create or view a play. To create a play, the user can choose between importing puppets or using the existing. Physics are applied to the puppets, in order to create a realistic movement, as the play is recorded. However, the exported product is not complete and must be further edited with other software to include sound. Collaboration tools are available locally or through the internet and input devices such as the Wiimote are supported in order to incite creative expression.

MUBIL: The 3D Lab Experience

Starting in 2011, MUBIL was a project aiming to propagate knowledge that could be found within the ancient collection of the library of the Norwegian Science and Technology University. The result of the endeavor is an interactive application in Virtual Reality, structured like an adventure game. The player takes the role of an ancient alchemy apprentice, with a mission to search in books for the process of distilling a certain medicine. To find it, he has to search in a 3D Virtual depiction of manuscripts and books that can be read while a set of tools and resources are available, namely multimedia elements, 3d animations, translations, transcriptions and annotations) [20].

La Dama Boba Game

Researchers in Spain, worried about the lack of interest of younger audiences for classic theater, decided to research the reasons of disinterest for classic plays. Considering the results of the study, a serious game was designed, aiming to address the reasons of the disinterest among high schoolers. “La Dama Boba”, the designed game, is a 2D adventure game, where the player assumes the role of an actor and through solving a set of puzzles and participating in conversations, gradually re-creates the plot of the play “La Dama Boba, El juego”, by Lope de Vega.[21]

Acropolis virtual tour (acropolis-virtualtour.gr)

Acropolis virtual tour[22] is a web application, also available as mobile application for iOS devices, through which the user can virtually explore the archaeological site of the Acropolis in Athens. To create this application, high-resolution gigapixel images was used along with a orthophoto map for user navigation between viewpoints[22]. This way the user has access to a set of photographs that are connected to recreate the archaeological site and allow navigation from point to point.

Table 1 – Cultural Heritage Learning Environments and Games

Games **Game constituents & fundamentals**	Web Games from Greek Ministry of Culture								
	What is hidden under the city	The celebration of the great Dionysia	A modern troupe in the ancient theatres of the world	Dionysus and his friends	eShadow	MUBIL	La Dama Boba	Acropolis virtual tour	ThimelEdu
2D	✓	✓	✓	✓	✓	✗	✓	✓	✗
3D	✗	✗	✗	✗	✗	✓	✗	✗	✓
Adventure Game	✓	✓	✓	✓	✗	✓	✓	✗	✓
Virtual Reality	✗	✗	✗	✗	✗	✓	✗	✗	✗
Interactive Game	✓	✓	✓	✓	✗	✓	✓	✗	✓
Accessibility	✗	✗	✗	✗	✓	✗	✗	✗	✓
Logs/Analitics	✗	✗	✗	✗	✗	✗	✗	✗	✓
Dynamic Content	✗	✗	✗	✗	✓	✗	✗	✗	✓
Web	✓	✓	✓	✓	✗	✗	✗	✓	✗
Native	✗	✗	✗	✗	✓	✓	✓	✓	✓
Recreate Archaeological Site	✓	✓	✗	✗	✗	✗	✗	✓	✓
Point to Point Navigation	✗	✗	✗	✗	✗	✓	✗	✗	✓
Support roles for players	✗	✗	✗	✗	✗	✗	✗	✗	✓
Solving Puzzles	✓	✓	✓	✓	✗	✓	✓	✗	✗
Participating in Conversations	✗	✗	✗	✗	✗	✗	✓	✗	✗
Recreate the Plot of a Play	✗	✗	✗	✗	✓	✗	✓	✗	✗
Propagate Knowledge	✓	✓	✓	✓	✗	✓	✓	✗	✓
Support Player's Mission in Doing Something	✗	✗	✗	✗	✗	✗	✗	✗	✗
Multimedia Elements	✗	✗	✗	✗	✓	✓	✗	✓	✓
3D Animations, Translations, Transcriptions and Annotations	✗	✗	✗	✗	✗	✓	✗	✗	✗

Input Devices Support	Mouse	Mouse	Mouse	Mouse	Wiimote	Mouse & Keyboard	Touch	Touch/ Mouse	Mouse & Keyboard / Touch
Collaboration	✗	✗	✗	✗	✓	✗	✗	✗	✗
Puts Students in Creative Role	✗	✗	✗	✗	✓	✗	✗	✗	✗
Sharing Available	✗	✗	✗	✗	✓	✗	✗	✗	✗
Ludic Environment	✓	✓	✓	✓	✗	✓	✓	✗	✓

2.2 Technologies for edugames in Cultural Heritance

Development of interactive apps, digital museums, virtual worlds, games, etc. is a complex procedure that includes many tools and technologies. The amalgamation of technologies such as programming languages, 3D design, physics engines and animation lead to sophisticated pieces of software that bring such environments to life.

2.2.1 Game Engines

A game engine is a system designed for game development, which consists of numerus tools and features, the most important of which is that the game engine acts as an intercessor with the operating system. Therefore game engines are considered as a framework, a tool in the hands of developers, allowing them to concentrate on the parts of the game that users experience while concurrently it takes care of the underlying levels such as hardware and operating systems. [23] Most game engines offer modules that handle input, rendering, sounds and physics.[23] [24]

Furthermore, game engines are designed in a manner that supports modules, extensibility and connectivity with other tools. As a result, developers can use other tools alongside a game engine, expand the engine with their own modules and use modules created by others. Consequently, game engines have become an inseparable part of game development and the main tool that developers use.

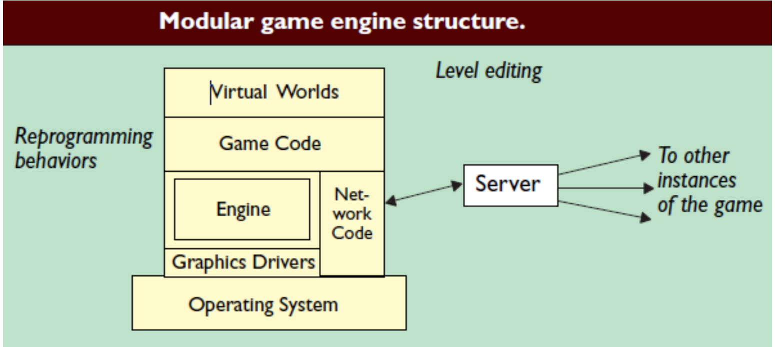


Figure 1 Modular game engine structure [24]

Popular Game Engines

Unity3D

Unity3D is a popular game engine, developed by Unity Technologies. One of its most prominent features is its cross-platform capabilities, reaching 27 platforms at the moment, from the most common, like windows PCs, to cutting edge technologies like Google ARCore. However, the main reason for its popularity lies with its very wide range of capabilities, as it supports 3D as well as 2D games, and is capable of building any genre of game. Its interface is welcoming for beginner developers and at the same time is suitable for professional developers that reach deeper into the engine and its capabilities. This is supported by a vast number of tutorials, extending from simple guidance of the interface to complex procedures regarding programming, animations, etc. Another popular point, is the fact that there is a free version with which anyone, simple user or company, can learn to use the engine, experiment, and even publish games, as long as revenue is within certain limits. This contributes to the engines popularity among small studios and hobbyists. However, some premium features are not available in the free version but are available through payed plans.

Additionally, Unity Asset Store is the official place where assets for the engine can be found, both for free and with a price, making many utilities, tools, sprites, 3D models, etc. readily available. Therefore, the process of building and publishing games needs less effort, time and resources.

Unity 3D was first released in 2005, and there have been 6 major releases since. Initially, 3 scripting languages were supported, C#, Boo and UnityScript, however in the last major release (Unity3D 2017) only C# is supported. Boo was deprecated in 2013 and UnityScript followed in 2017. Essential, out of a multitude of reasons, for the deprecations is the architecture of the engine. Using Mono, an open source system to run .NET applications, C# is part of the core, while the other two languages are not.

Unreal Engine 4

Unreal Engine 4 is another popular game engine. It is developed by Epic Games and was introduced in 1998 as an engine mainly for first person shooters, even though games of other genres have been developed using this engine. It supports a variety of desktop, console, mobile and VR platforms, but not as many as Unity3D. It only supports C++ language for scripting, but it also has a visual scripting tool, called Blueprints. It has immense capabilities in graphics and programming, at the cost of high resource requirements.

In comparison with Unity3D, it has a steep learning curve and is not beginner friendly, but is more capable when it comes to graphics. Additionally, while both engines are free to use, their plans for publishers differ, making Unity3D more suited for small independent studios and UE4 better for studios with bigger budgets.

CryEngine

CryEngine is a game engine initially released in 2002 and is now up to its fifth major release. Designed for development of PC and console games, it is a powerful engine, with similar capabilities in graphics with Unreal Engine 4, but has a steeper learning curve.

A unique feature of the CryEngine is the Live Create, a tool for synchronous editing of multi-platform game environments, which offers the ability to view the results for every platform at once. Yet, the engine’s most prominent features lie with water physics and graphics and tools that offer very realistic terrains.

In contrary with Unity3D and Unreal Engine 4 it does not charge a royalty fee, but requires a subscription for usage.

Torque 3D

Torque3D, available also as Torque2D for 2D development, is an open-source game engine released in 2012. As an open source engine, there are no limitations to its capabilities, however, there is limited support. Therefore, developers rely on their own skills, or on the community, for some features that are supported by other engines, but are free to reach further than they could on proprietary engines.

Blender

Blender is an open source graphics creation toolset that has integrated game engine. Its main features are tools for graphics such as 3D modeling, texturing, UV unwrapping, rendering, rigging, but it also includes tools for video editing and animation. It is a very popular for its graphics tools, but not as much for its game engine. Even though it is a powerful engine its major drawbacks are lack in support of platforms and the usage of Python as a scripting language. In a gaming industry where mobile and console platforms are very important, developers overlook an engine that does not support such platforms. Also, when most developers are used to work with C, C++, C# and Lua, Python is not an enticing option, even though it does have its strengths.

Table 2 - Game Engines

	Unity3D	Unreal Engine 4	Cry Engine	Torque 3D	Blender
2D Support	✓	✗	✗	✗	✓
3D Support	✓	✓	✓	✓	✓
Programming Language	*C#*	*C++*	*C++/Lua*	*C++/TorqueScript*	*Python*
Visual Programming	✗	✓	✗	✗	✗
Cross Platform Support	*27 platforms*	*16 platforms*	*4 platforms*	*4 platforms*	*3 platforms*
Synchronous cross-platform development	✗	✗	✓	✗	✗
Usage Cost	*Free*	*Free*	*Subscription*	*Free*	*Free*
Publishing Cost	*Free (conditions apply)*	*5% of revenue*	*Free*	*Free*	*Free*

2.2.2 Frameworks

IOLAOS

IOLAOS is a platform for serious games in education. At its core, IOLAOS is a platform for learning, based on the contribution and collaboration of various fields of expertise. As a result of this collaboration learning styles are codified into game preferences, which are harnessed from game developers through a framework, creating games that are adjustable to each individual user. Teachers can use these games in their virtual classes that are also part of the platform, and monitor the sessions, while students learn at optimal conditions. [14]

MONO

Mono is an open source implementation of the .NET Framework that brings the Common Language Runtime to a wider spectrum of platforms. The .NET Framework is proprietary and owned by Microsoft, however the Mono project is sponsored by Microsoft and created under the safety of the Microsoft Open Specification Promise, and thus there have been no legal conflicts.

Mono is able to run on various architectures, including x86-64, x86, ARM and MIPS and supports the most popular operating systems, including Windows, Android, macOS, many Linux distributions, BSD and even the operating systems for three of the best-selling consoles, namely Xbox, PlayStation and Wii.

Mono is also an essential part of Unity3D's core and a fundamental factor for the cross-platform support of the engine.

SCORM

SCORM is comprised of specifications that are used in e-learning software. It acts as a guideline for developers to make different e-learning software compatible with each other. It acts as a standard for communication between Learning Management Systems and online learning content.[25]

Experience API (xAPI)

The Experience API is a framework that facilitates data capturing with the use of a simple, yet flexible, specification. It is used in learning technologies as a middleware for communication between software and Learning Record Store (LRS). This way xAPI enables very different systems to have a secure communication while capturing and sharing data of the experience a user has.[26]

2.2.3 Programming & Markup Languages

C#

C# is a multi-paradigm programming language, meaning that both object-oriented and imperative paradigms are supported, that was created by Microsoft for the .NET Framework. It has also been approved as a standard by ECMA and ISO. C# was designed for software robustness and portability. Portability is ensured to a large extent by C# being a part of the Common Language Infrastructure, a specification that allows different platforms to use different programming languages without code needing to be rewritten for each platform.

C++

C++ is a programming language that can be used in imperative, object oriented and generic programming architectures. It was designed for resource-constrained systems aiming for efficiency and flexibility. These two values make it popular for game creation but also for servers and performance-critical applications. It is very widely used and there are compilers for most architectures available.

Lua

Lua is a free open source and lightweight scripting language known for its performance. It has simple procedural syntax and is cross-platform as its interpreter in written in ANCI C, making it able to run on most platforms. Even though classes are not included as a concept, object-oriented programming is achieved with the use of other features of the language. Lua is very popular for game development and many game engines use Lua.

JSON

JavaScript Object Notation is a way to encode Javascript objects into human readable text. It is used for data exchange, commonly in browser-server communication, as a lightweight solution, with minimal overhead and native Javascript data. However, despite the Javascript syntax, JSON can be used with an any programming language, as it consists of text and has therefore spread to a wide range of applications, especially when networking is involved.

XML

Extensible Markup Language, as suggested by the name, is a markup language used to encode documents with a specific format that can be understood by humans but also parsed by machines. It is widely used for representation, storage and exchange of data structures. Due to its popularity most programming languages include parsing and writing for XML files, but even if there is none, the simplicity of the format makes it easy to create in any language.

2.2.4 ThimelEdu Technologies used

Unity3D

All game engines mentioned earlier have similar functions, however they focus and specialize on different features, therefore producing comparable but distinct outcome.

Focused on our desired result and the requirements for the game development procedure, Unity3D was considered the most appropriate solution for our case. Specifically, the decision was influenced by the following reasons:

- The free plan available suits the technical needs and gives us the opportunity to publish the game without financial costs or legal issues.
- The resulting game has low hardware requirements, therefore being suitable for most systems available at schools today.
- Cross-platform compatibility gives us the opportunity to offer the game to wider audiences.
- A huge community that exists behind the engine ensures simpler problem solving.
- The engines lightning and general aesthetics were considered tempting for children.

IOLAOS Platform

Assuming the “Game Developer” role of IOLAOS, a powerful set of educational tools are available to us, without the need for extensive research on learning theories and pedagogical methods and implementation from our part. Much of the needed research is already codified and offered to us in a simple API, along with a set of educational tools that are available to teachers that will be using the game through IOLAOS. By utilizing the Framework offered by IOLAOS, we ensure that ThimelEdu will always be in ideal state for each user, in both educational and entertainment perspective.

Another value of IOLAOS for game developers, is the fact that there is no need for creating and maintaining databases, since IOLAOS maintains a Common Data Space that is accessed through the API, while at the same time there is no restriction on usage of databases outside the CDS, for any reasons that might be required. This does not only mean that there is less expense in man hours to design, build and maintain databases, but also that the data, through being shared by a number of discrete applications, creates information of much greater value for education than what a single application could produce.

Lastly, by being endorsed and approved by IOLAOS, a game is immediately accessed by a community of learners and teachers. Alongside with the CDS of IOLAOS, this community can provide excellent feedback to any developers who wish to keep improving their games. Rating and comments on the side give a clear vision of the users’ thoughts, while data stored in Learning Records can be extracted and analyzed in order to spot design holes not realized earlier.

C# & Mono

Since Unity3D was chosen for the development of ThimelEdu, C# and Mono was also required. UnityScript or Boo could have been used as they are compiled by the engine, but since official support has been dropped they were not even considered as options.

Furthermore, C# has more thorough documentation and a vast number of libraries as is it a language used in a much wider spectrum of industries.

Blender

Blender game engine was not suited for ThimelEdu, but there still is a need for 3D models, textures and sprites that blender is capable of producing. Blender was the obvious choice due to the fact that it is free and open source as well as some prior experience with the toolset.

JSON

IOLAOS implements RESTful web services that encode data in JSON format, therefore usage of JSON for ThimelEdu is mandatory. Therefore, JSON will also be used for data storage not related to IOLAOS, instead of XML that is usually the choice for Unity3D and Mono projects, for uniformity reasons.

Experience API

The second IOLAOS's maturity level requires the implementation of xAPI LRS to store game created data. xAPI offers data analytics techniques and tools that enable involved experts to acquire insight of the produced data. This is very helpful for teachers as they can have access to data analytics for their students. Furthermore, this data can be harnessed by educational experts through the IOLAOS's platform in many ways and is therefore one of the main future engagements of ThimelEdu to implement the required mechanisms.

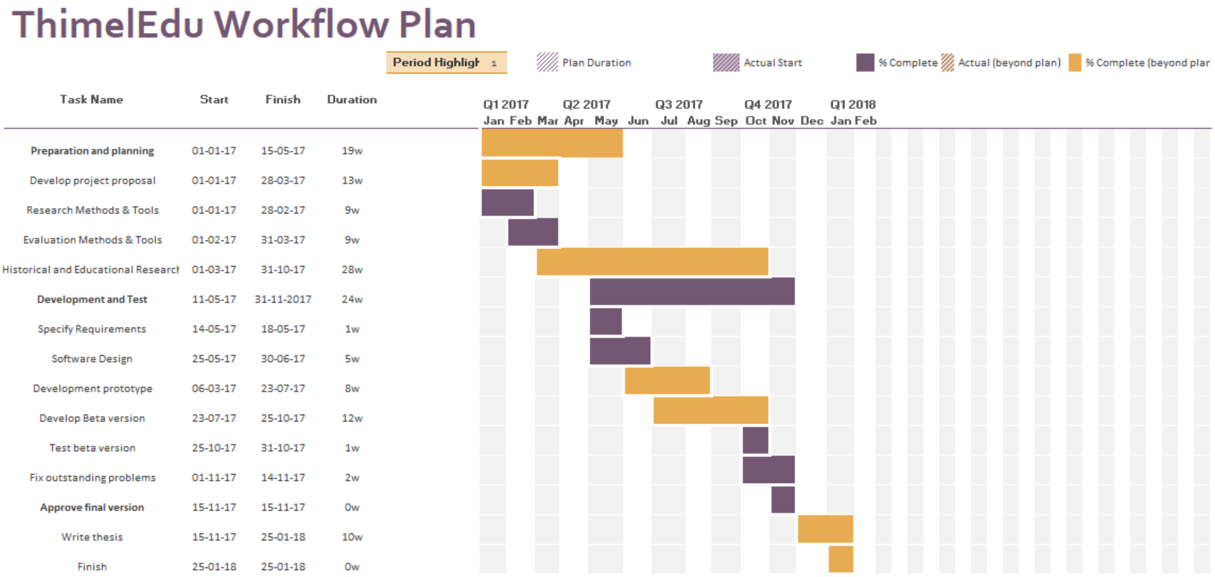
Table 3 - ThimelEdu Technologies

Technologies **Game constituents & fundamentals**	IOLAOS Platfomr	Unity3D	C#	Mono	JSON	Blender	xAPI
Data Repository	✓	✗	✗	✗	✓	✗	✓
3D Avatars	✗	✓	✗	✗	✗	✓	✗
3D Artifacts	✗	✓	✗	✗	✗	✓	✗
2D Sprites	✗	✓	✗	✗	✗	✓	✗
Textures	✗	✓	✗	✗	✗	✓	✗
Accessibility	✓	✓	✓	✓	✓	✗	✗
Automatic Personalization	✓	✓	✓	✓	✓	✗	✗
Personalization at will	✗	✓	✓	✗	✗	✗	✗
Localization	✓	✓	✓	✗	✓	✗	✗
Logs/Analytics	✓	✗	✗	✗	✓	✗	✓
Dynamic Content	✓	✓	✓	✗	✓	✗	✗
Multimedia Elements	✗	✓	✓	✗	✓	✗	✗
Historic Information	✗	✓	✗	✗	✓	✗	✗
Recreate Archaeological Site	✗	✓	✗	✗	✗	✓	✗
Point to Point Navigation	✗	✓	✗	✗	✗	✓	✗
Support roles for players	✓	✓	✗	✗	✗	✓	✗
Propagate Knowledge	✓	✓	✗	✗	✗	✗	✗

Support Player's Mission in Doing Something	✓	✓	✓	✗	✗	✗	✗
Ludic Environment	✓	✓	✓	✗	✗	✓	✗
Cross Platform	✗	✓	✓	✓	✓	✗	✗
Low Hardware Requirements	✗	✓	✓	✓	✗	✓	✗

2.3 Thesis Plan

In order to reach the goals for this thesis, planning and organization was required. For that reason, a plan was developed that would fit the needs of research, software engineering and development and of course writing the thesis. This plan can be seen in the following Gant Chart that illustrates the task and the time frames in which they were completed.



Task Name	Start	Finish	Duration
Preparation and planning	01-01-17	15-05-17	19w
Develop project proposal	01-01-17	28-03-17	13w
Research Methods & Tools	01-01-17	28-02-17	9w
Evaluation Methods & Tools	01-02-17	31-03-17	9w
Historical and Educational Researct	01-03-17	31-10-17	28w
Development and Test	11-05-17	31-11-2017	24w
Specify Requirements	14-05-17	18-05-17	1w
Software Design	25-05-17	30-06-17	5w
Development prototype	06-03-17	23-07-17	8w
Develop Beta version	23-07-17	25-10-17	12w
Test beta version	25-10-17	31-10-17	1w
Fix outstanding problems	01-11-17	14-11-17	2w
Approve final version	15-11-17	15-11-17	0w
Write thesis	15-11-17	25-01-18	10w
Finish	25-01-18	25-01-18	0w

Figure 2 - ThimelEdu Gant Chart

3 The Educational Game ThimelEdu

In modern education ancient drama is used to teach Ancient Theater. However, studies focus on the value of the literature, overlooking the significance of other elements that facilitate the experience of theater as an artform. Such elements are the sites where the plays took place that had specific geo location, architecture for symbolic and practical reasons and various tools, used during the plays to create special effects to symbolize events and many more. As a result of the current education method styles and processes, students understand the basics of the narrative, but lack a comprehension of theater as a complete artform.

Triggering students' interest in arts, drama, architecture and techniques that are already centuries old is challenging for every teacher. Even more so, when teaching is restricted to a literary study due to the nature of historical topics. Inability to provide access to archaeological sites, the opportunity for recreating situations and environments for experiential learning create a monotonous learning process that often disinterests students.[27]

Additionally, for students with special educational needs, literary study can be even more challenging. Traditional mediums and methods are not suitable for these students, while video games use different means that can have lots of advantages in many cases. Furthermore, games act as an effective mechanism for abilities improvement which is very useful within special education, as use of specific intelligences can enhance other, weaker ones. This happens within a setting where problem solving utilizes abilities of students surpassing any difficulties they face [27] such as a learner with highly developed musical intelligence utilizes song-making to learn about a topic in history. [28]

With technologies readily available it is easier than ever to create virtual worlds and environments for users to explore. Recreating historical environments gives an opportunity for educational adventures that elevate the learning process in favor of the students' comprehension on a topic.

On these grounds, this thesis aims to design and develop an educational game that can be incorporated in the classroom environment, offering a complementary alternative to traditional teaching methods, activities and tools that are attractive to students and provide educators with valuable information about the learning process. Furthermore, our game will be designed with special education in mind, in order not only to include learners with such conditions, but mainly to assist them to overcome any difficulties and develop skills they lack.

3.1 ThimelEdu: Game & supporting Software Analysis

In the process of game creation, game and supporting software analysis and design are two parallel tasks that often get mixed up. However, they are two distinct concepts, carried out by professionals with different expertise, with strict collaboration between the two.

Game flow, story, aesthetics, scenario, direction, educational goals, etc. are defined during the game design process. Game design, as a general term, focuses on the experience a game offers. Jesse Schell [29] summarizes game design as "the act of deciding what a game should be". Software design on the other hand revolves around the technical elements that can actualize the game design and produce the game.

The IOLAOS framework acts as a causal factor for ThimelEdu, considering that many crucial elements of the game are implemented following IOLAOS's API such as personalization and accessibility. Furthermore, acquiring IOLAOS's maturity level is a goal for ThimelEdu that affects both the game's design and the supporting software implementation. The various possibilities outlined within IOLAOS's API acts as a guideline for the design and implementation of educational games and in the same way for the design and implementation of ThimelEdu. However, not all protocol services specified by IOLAOS are implemented within ThimelEdu. IOLAOS includes many details to cover all game genres-styles and as a result some of the options are irrelevant to some games.

3.1.1 Data acquisition

To acquire data and valuable information to decide upon matters of game design and content numerous methods were utilized. Bibliographical research, expert interviewing and questionnaires were part of a process through which conclusions were reached. Through this process all of the educational requirements listed below were defined. Historical material that will be used within the game as content are codified into tables Table 4 and Table 5 that follow.

Table 4 - Informational Content Questionnaire

Τύπος Πληροφορίας: Κείμενο 1 Στο κέντρο της ορχήστρας βρισκόταν ο βωμός του Διονύσου, η θυμέλη. Πίσω από τη θυμέλη έπαιρναν θέση ο αυλητής και ο υποβολέας. **Πηγή: Σχολικό Βιβλίο Αντιγόνης**	
Τύπος Πληροφορίας: Κείμενο 2 ΑΜΕΑ **Στο κέντρο της ορχήστρας βρισκόταν ο βωμός του Διονύσου, η θυμέλη.**	
Τύπος Πληροφορίας: Εικόνα 1 **Πηγή:** Θυμέλη αρχαίου Θεάτρου Διονύσου, 2ου αιώνα μ.Χ.	

The above table, both its structure and content, is the result of the collaboration between education and history experts regarding ancient theaters. In respect to the content of this data acquisition structured form, the exemplar in Table 4 includes information about “thimele”, the sacrificing place of the ancient theaters. In respect to the structure of this data acquisition structured form, the first row consists of an article suited for mainstream schools, the second includes the special school counterpart and the third row present a picture that can be used as media. With the use of the above structured forms, all kinds of data, concerning the ancient theater have been acquired, stored and presented to the learner during game play.

Considering the information collected and continuing the collaboration between experts, quizzes have been designed, stored and presented during play aiming to assess the learner’s knowledge acquired during the game play session. Tables like the one following were filled in and organized according to the artifact they relate to. Within the table the target audience for the question is specified (Special education or not), the question along with the possible answers offered to the learners and some extra information on the type of question and its specification. In the following example table, the question is about “thimele”. It is a closed-type question with three choices, designed for special students. During data acquisition close and open type questions were created, however only closed type questions are currently utilized within ThimelEdu, as evaluation of open type questions have not been implemented yet.

Table 5 - Quiz Content Survey

Ερώτηση 1: Τύπος Πληροφορίας Ερώτησης: Κείμενο Η θυμέλη βρισκόταν: (ΑΜΕΑ) 1. στο κοίλο **2.** στη σκηνή

3. στην ορχήστρα

Μορφή / Τύπος Ερώτησης:
Κλειστού Τύπου Ερωτήσεις: *Τα Είδη Κλειστού-Τύπου Ερωτήσεις έχουν μία προ-αποφασισμένη ομάδα επιλογών απαντήσεων.:*

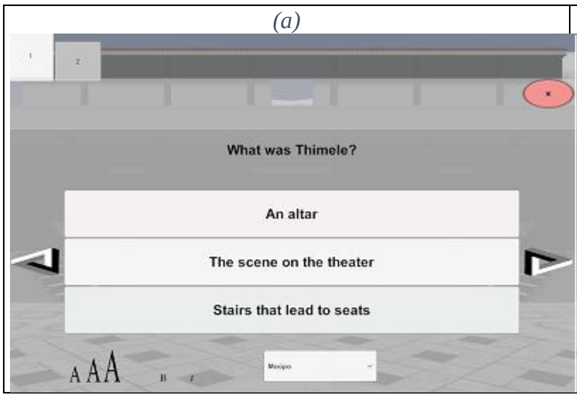
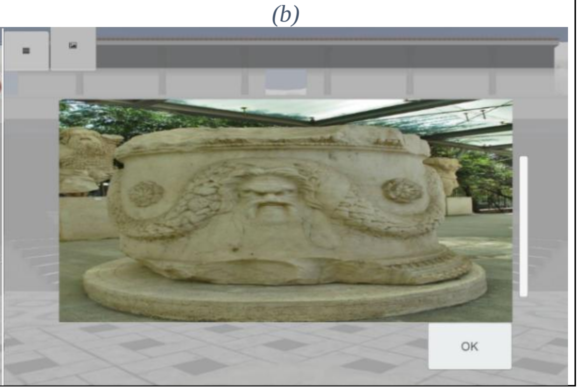
The information and quizzes presented above are accessed in the game through interaction with the “thimele” artifact. The in-game representation of the information can be seen in Table 6 below. Similarly, other artifacts will present information and questions about the corresponding topics.

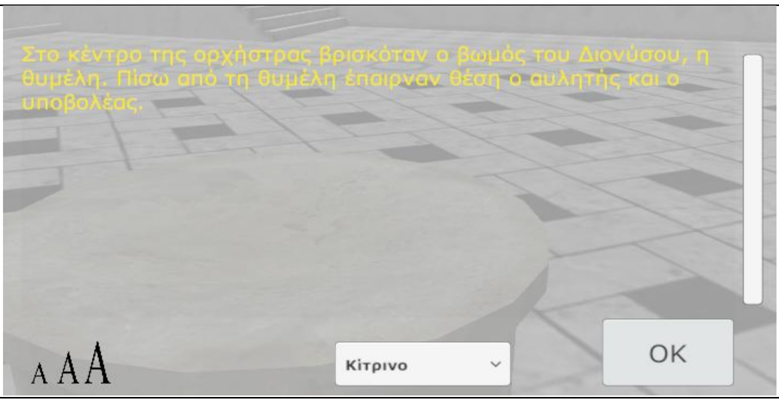
In (a) we see a screenshot of a question posed within the game. The question is displayed and bellow it 3 buttons represent the choices. This way the user can click on the choice he/she feels is right. On the bottom of the dialogue a row of buttons is placed to allow the user to change the font settings for readability. The three “A” of various sizes are used to change the font size, between small, medium and large. The “B” and “I” buttons are used as triggers for bold and italics and the drop box includes various colors for the font. To ensure that the player can leave in case the question is difficult or stressful, there are several options given. The arrow buttons located to the left and right of the choices, allow the player to navigate to the previous or next question. Similarly, the tabs located at the top right can be used to navigate to any question within the artifact. Finally, if the player wishes to stop the quiz and resume exploring an exit button is located in the top right.

In (b) we see a screenshot of an information dialogue displaying a picture. Apart from the picture, the dialogue is quite simple, the tabs at the top right can be used to view an article of the same topic or a picture if the article would be showing. The “OK” button will navigate players away from the information dialogue and will present them with the question dialogue presented in (a).

In (c) we see a screenshot of an information dialogue presenting an article. The layout is similar to when an image is presented, however it is enriched with buttons for font settings, as in (a) and a scroller for moving within long articles.

Table 6 - Example Game Content

(a)	*(b)*
 What was Thimele? An altar The scene on the theater Stairs that lead to seats	 OK



Στο κέντρο της ορχήστρας βρισκόταν ο βωμός του Διονύσου, η θυμέλη. Πίσω από τη θυμέλη έπαιρναν θέση ο αυλητής και ο υποβολέας.

(c)

3.1.2 Requirements List

According to the data gathered from all data acquisition techniques as they are mentioned in the previous section and by applying textual analysis methods we have composed requirement list presented in Table 7. The requirements are divided into System Requirements marked with capital **S**, Educational Requirements marked with capital **E** and Accessibility Requirements marked with capital **A**.

Table 7- Requirements List

ID	**Title**	**Description**
S1	IOLAOS Approved	Crucial requirement is the integration of the IOLAOS API and the approval by IOLAOS platform.
S2	User Authentication	User authentication must be authorized by IOLAOS through the API. Users not registered with IOLAOS will be able to play the game as “Guest User”.
S3	Guest User	Users not registered with IOLAOS must be able to play the game without the benefits of IOLAOS.
E1	Educational Methodologies	The game must consider learning styles as defined by IOLAOS and adapt the gameplay to meet the criteria.
A1	Auto Personalization	In order for the game to be IOLAOS approved, game settings must be automatically changed in accordance with IOLAOS guidelines for each user.
A2	Accessibility	The game must be accessible to students with special educational needs.
S4	Analytics Storage	During gameplay logs must be kept that will be saved for future study and analysis by educators and educational experts.
A3	Real Time Settings Manipulation	Children with special education needs can be unpredictable at times. Therefore, even though IOLAOS has experts working with game preference profiles, providing the best fit for each individual, the automatic game customization can at times be experimented with by teachers for the benefit of students. Additionally, by supporting unregistered users, there is no automatic customization. Therefore, real time settings manipulation that users can control is deemed essential for the game.
S5	Hardware Requirements	Since schools often have a lack in budget for technology it is essential that the game doesn’t have excessive hardware requirements. Furthermore, future ports to other platforms, such as mobile, should be a viable option. Therefore, hardware requirements should be kept at a minimum without sacrificing ludic gameplay.

E2	Appropriate Content	As the game aims to educate regardless of age, educational level, abilities or needs, the content must be flexible in order to be appropriate and attractive to each user.
E3	Dynamic Content	The content of the game must be able to change depending on many factors, including age, educational level, special abilities, educational needs, etc
A4	Localization	The game should be able to translate to other languages
S6	Offline Functionality	The game must be able to be played offline
E4	Informative Dialogue	The game must include a way of presenting information
E5	Quiz Dialogue and Questions	The game must include a way to pose educational questions
E6	Rewarding	The game must include a rewarding system.
S7	3D Game World	The game world must be in 3D
E7	First Person Perspective	The game must use a first-person perspective in order to make the players identify the avatar as themselves
S8	Game Map	The game must include maps for navigational purposes.
S9	Free Roam	The game should not restrict player movement within the world. Movement shall only be restricted when needed for gameplay and to disallow the player from exceeding the world limits.
S10	Player Movement	Player should be able to move towards all direction, jump and run.
E8	Support Roles	Inside the game ways to help the player should exist, in the form of supporting characters.
E9	Multimedia	The game must support and include multimedia
S11	User Interface	The game must have a user interface or HUD that presents the user with crucial info, without being overloading.
E10	Historic Content	All content must be historically correct.
S12	Cross Platform	The game should be able to be ported for different platforms
S13	Low Hardware Requirements	Hardware requirements must be kept at a minimum.
S14	Input	Input devices shall be whatever is native for each platform. Keyboard and mouse for computers, touch for mobile.
E12	Classroom environment	The game should be playable within a classroom environment
S15	Server Communications	The software must be able to communicate with servers through RESTful APIs, including IOLAOS
S16	JSON	The underlying software should be able to parse, understand and create JSON files.
S17	Local Data Storage	All data that needs to be locally stored will be saved in JSON format.

3.1.3 Requirements Diagram

From the above requirement list we have drawn the following UML requirements diagram that outlines not only the system and user requirements but also the relationships between them. The most central requirement, “IOLAOS Approved”, needs a lot of other requirements to be satisfied since it is a complex but meaningful requirement. Others, like “Offline Functionality” for example, are simple requirements that don’t need something else to be satisfied, nor act as a prerequisite themselves.

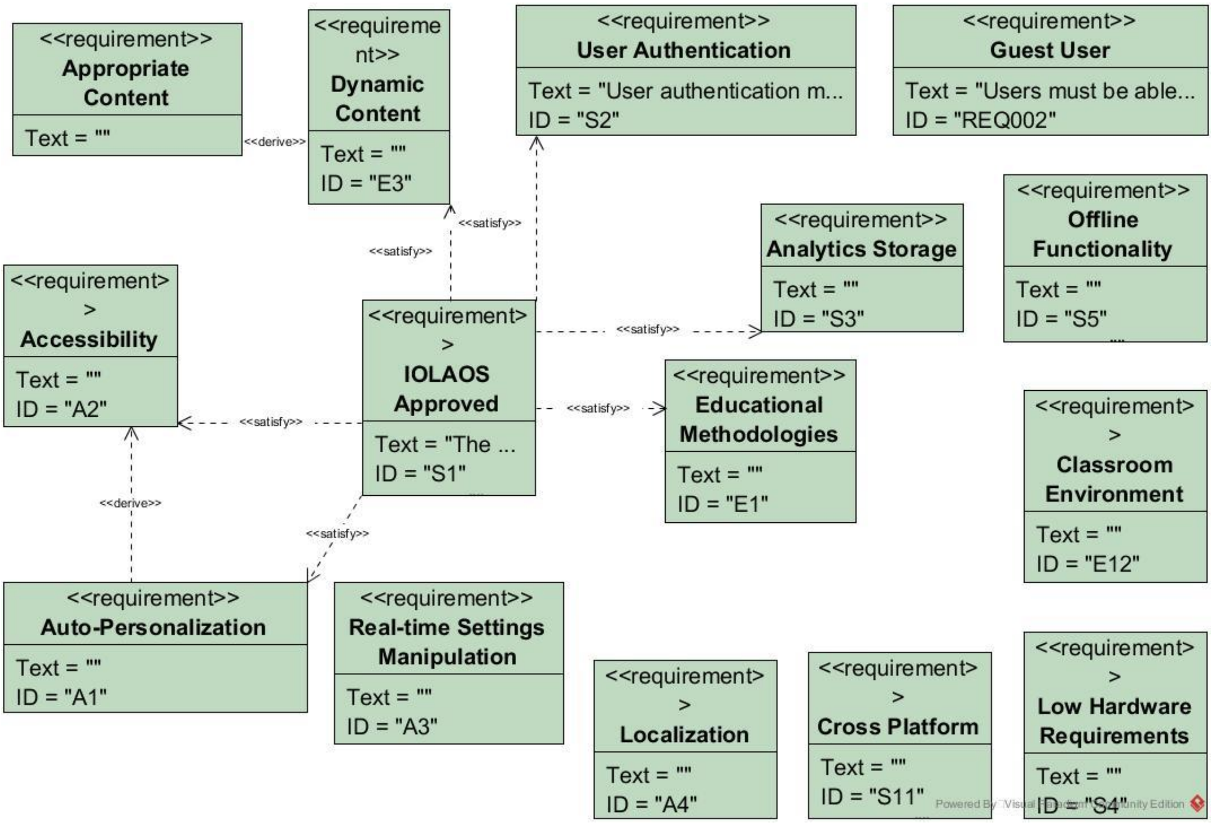

<<requirement>> Appropriate Content
Text = ""
<<derive>>
<<requirement>> Dynamic Content
Text = ""
ID = "E3"
<<requirement>> User Authentication
Text = "User authentication m...
ID = "S2"
<<requirement>> Guest User
Text = "Users must be able...
ID = "REQ002"
<<satisfy>>
<<satisfy>>
<<requirement>> Analytics Storage
Text = ""
ID = "S3"
<<requirement>> Offline Functionality
Text = ""
ID = "S5"
<<requirement>> Accessibility
Text = ""
ID = "A2"
<<requirement>> IOLAOS Approved
Text = "The ...
ID = "S1"
<<satisfy>>
<<satisfy>>
<<satisfy>>
<<requirement>> Educational Methodologies
Text = ""
ID = "E1"
<<requirement>> Classroom Environment
Text = ""
ID = "E12"
<<derive>>
<<satisfy>>
<<requirement>> Auto-Personalization
Text = ""
ID = "A1"
<<requirement>> Real-time Settings Manipulation
Text = ""
ID = "A3"
<<requirement>> Localization
Text = ""
ID = "A4"
<<requirement>> Cross Platform
Text = ""
ID = "S11"
<<requirement>> Low Hardware Requirements
Text = ""
ID = "S4"
Powered By Visual Paradigm Community Edition

Figure 3 - Requirements Diagram

3.2 ThimelEdu: The game Design

An overview of ThimelEdu, as experienced by the users, is presented in the following use case diagram in Figure 4. As illustrated in the diagram, the players will login using their IOLAOS credentials, or proceed to play the game as guests. The choice to exit or logout will be present at all times. During gameplay, the player explores and interacts with his surrounding, ultimately receiving information and answering quizzes.

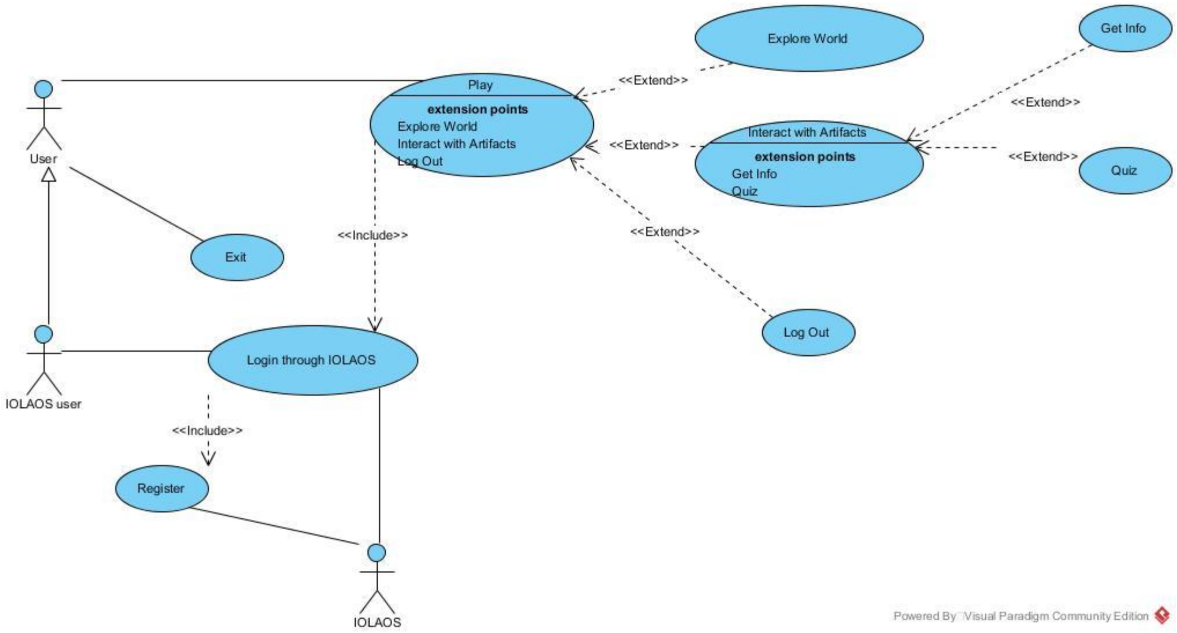


Figure 4 - Overview Use Case Diagram

The game design aims to fulfill all requirements listed above. It also aims to make the game attractive to children without excluding adults and to make it usable in a classroom setting. A final goal within the design is to make the game as simple as possible, avoiding complex gameplay and thus making it suitable for the youngest students and students with special educational needs.

ThimelEdu is a single player educational game consisting of a 3D world where the player can roam freely. Players are called to explore an ancient theater, discover information and answer questions. The target audience consists of school students of all levels, aiming to become an assisting teaching method within the classroom. However, others are not excluded. Therefore, the game content is adaptable, utilizing user profiles from IOLAOS and game personalization techniques to make it suitable for every possible user.

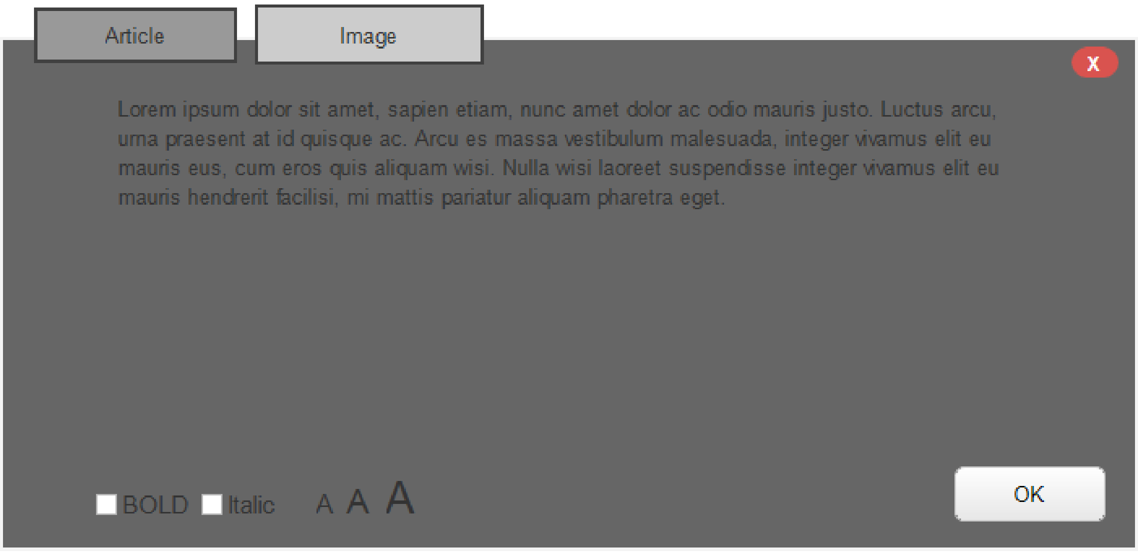
For simplicity reasons the game lacks any scenario and is experienced in a first-person view. This way the player can identify the avatar as his virtual self. By default, full movement control is enabled (jump, run, forward and backward walking) but this can be modified by the player or through IOLAOS. During gameplay, the goal is to locate several ancient theater artifacts, positioned in the scene, and interact with them in pursuance of information about them. Moreover, quizzes are presented to the player during interaction with these artifacts.

Help – Assistant

An owl is available, when dictated by the IOLAOS profiles, as a helper for the player. She can offer insightful information about the game and helps players when baffled. Additionally, she makes comments to encourage and reward the player.

Educational Material Presentation

The information is presented within a dialogue box that offers articles, pictures and other media about the artifact interacted with. The dialogue will be split into tabs and under each tab one type of media will be presented. For each type of media, one object will be randomly selected to be shown. Tabs can be switched between as many times as the user wishes, and unlimited time is also provided for reviewing the content. Also, buttons to modify text size and style are provided within the dialogue for immediate access and text is updated in real time.



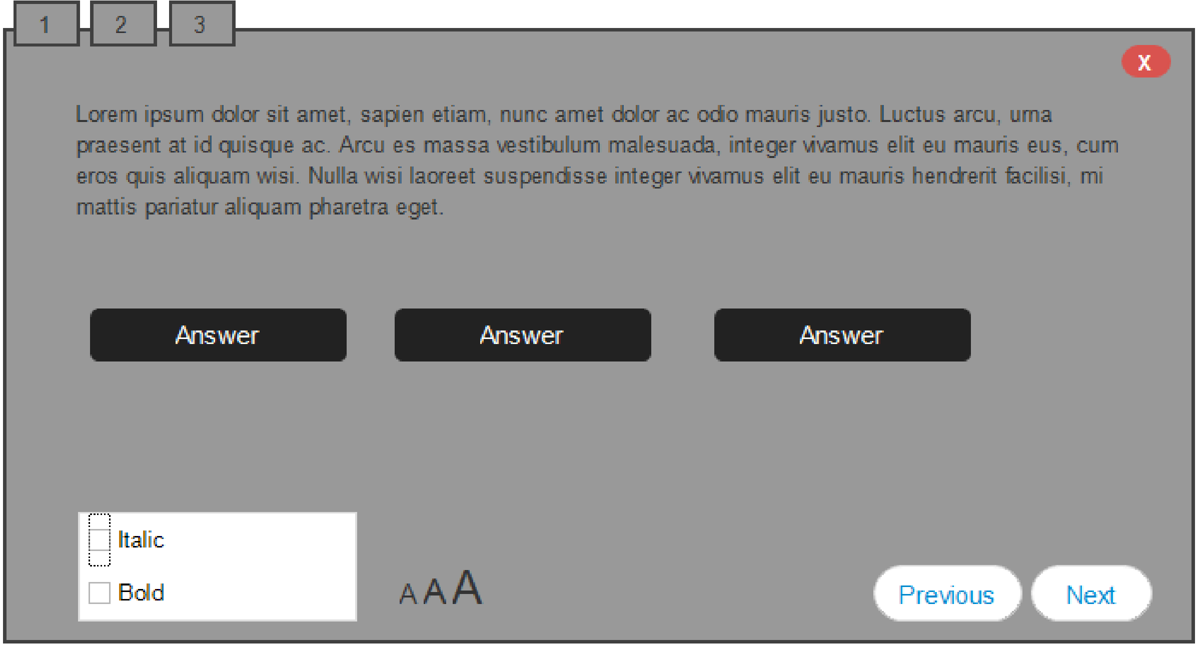
Lorem ipsum dolor sit amet, sapien etiam, nunc amet dolor ac odio mauris justo. Luctus arcu, urna praesent at id quisque ac. Arcu es massa vestibulum malesuada, integer vivamus elit eu mauris eus, cum eros quis aliquam wisi. Nulla wisi laoreet suspendisse integer vivamus elit eu mauris hendrerit facilisi, mi mattis pariatur aliquam pharetra eget.

Figure 5 - Informative Dialogue Mockup

Learning Evaluation

After the user has had the time to review the information, he can continue to the quiz dialogue. In this dialogue, unlike the information dialogue, all questions related to the artifact are available each one under a separate tab. Tabs automatically switch after an answer and jump to the first unanswered when an artifact is revisited. However, switching between questions is in the hands of the player and there is no limit in time or switches. An exit button allows the player to choose to leave the quiz and return later.

Question type is always close-ended, as analyzing open-ended questions would be unnecessary challenging if not impossible. Therefore, all questions are either of "True/False" nature, or multi-choice, with only one answer being the correct one and the possible answers always being three. The number of possible answers for each question was decided so that there are enough choices while they are not overwhelming. Consequently, the quiz is challenging, and boredom is avoided. Additionally, visual and audio feedback are provided upon answering to signify success or failure.



Lorem ipsum dolor sit amet, sapien etiam, nunc amet dolor ac odio mauris justo. Luctus arcu, urna praesent at id quisque ac. Arcu es massa vestibulum malesuada, integer vivamus elit eu mauris eus, cum eros quis aliquam wisi. Nulla wisi laoreet suspendisse integer vivamus elit eu mauris hendrerit facilisi, mi mattis pariatur aliquam pharetra eget.

□ Italic

□ Bold

Figure 6 - Quiz Interface Design

Scoring for those questions is based on a simple incremental model. Each question is worth one score point that is awarder when correctly answered. Additionally, a special reward is provided when all questions of an artifact are successfully answered on the first try. An important detail is that there are no penalties. When the score is equal to the number of questions in game, the game is finished, and the special rewards serve as a recognition of the player's knowledge.

3D world

The game world experienced by the player consists of a single 3D map that includes an ancient theater. Hills act as a natural outline for the map. The player is restricted to not leave the map but has full access within it. Artifacts design is in accordance with historical evidence to the extent possible. 3D models representing building and artifacts are as low-poly as possible while being realistic representations. Non-historical 3D models (trees, etc.) are as low-poly as possible. During gameplay, music is played to create an atmosphere relevant to the historic context of ancient theater.

To help players navigate the word there are two maps available. One is part of the Heads Up Display (HUD) and the other must be manually opened. The map available on the HUD is a mini-map covering a few meters around the player, while the manual map is a map of the whole world. In both, the location of the player can be seen, as well as landmarks and the interest points. Interest points are highlighted with a color-code so that their state of completion is obvious. Blue is unvisited, red is incomplete, and green is completed.

ThimelEdu Expected UI

ThimelEdu's interface aims to be as informative as possible without overloading the user with information. One corner is reserved for the mini-map and related information, another corner is used to display score related information and a third corner displays the helper. Additionally, in the remaining corner the username of the current player will be displayed. Lastly, there is a menu screen which displays information about current settings and buttons, sliders and checkboxes are in place to change settings.

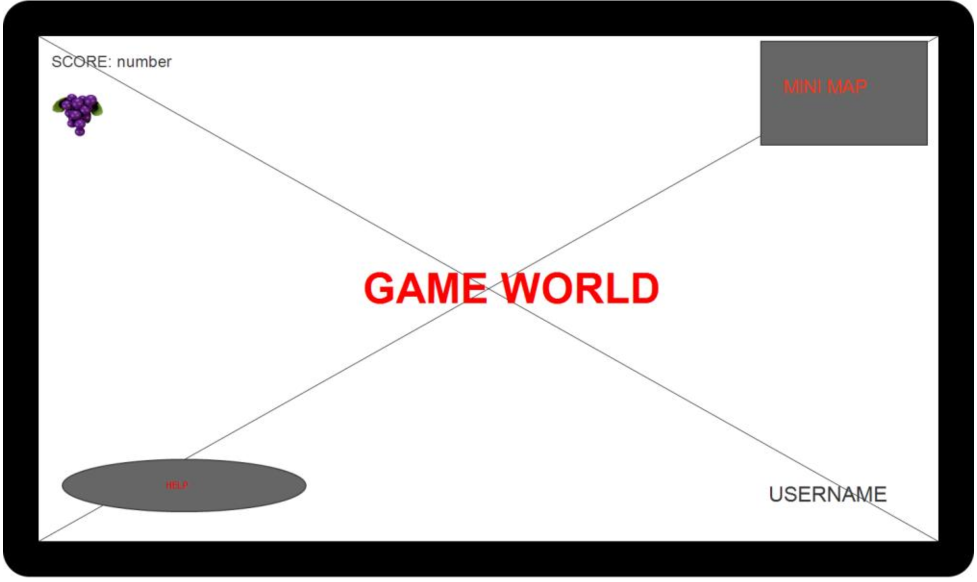


Figure 7 - Expected User Interface

3.3 ThimelEdu: The supporting software Design

One substantial benefit of game engines is that significant pieces of the architecture are handled by the engine, minimizing workload both in design and development. Therefore, developers need to design and implement only modular functionalities that are not included in the game engine and the necessities to bring the game design to a working game.

In our case, modules are needed for communication with IOLAOS, to handle game settings and content and for localization. Communication with IOLAOS is achieved through a solid protocol utilizing HTTP methods, POST and GET, that act as a request, and JSON, which is the reply from IOLAOS and contains the requested information. Thus, the JSON string must be processed for the extraction of needed information which will be acted upon and saved. Since IOLAOS maintains databases with users and offers user profiles and game preferences, assigned by experts, that greatly benefit ThimelEdu's customization and minimizes maintenance on our part, there is no reason for us to create databases for users or save game settings and preferences. Thus, it is decided that ThimelEdu will not implement any databases or save data locally, but instead use the sources offered by IOLAOS. Consequently, communication with IOLAOS is a core necessity that must be thoroughly designed and attentively implemented.

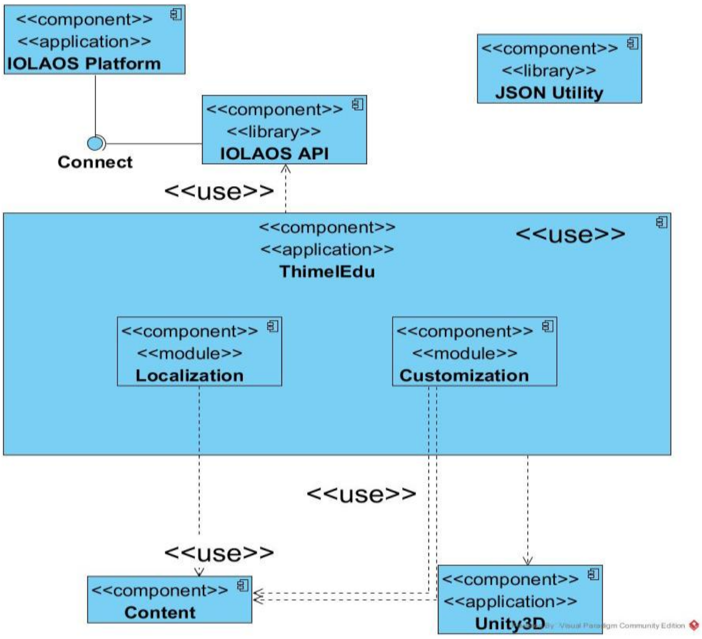


Figure 8 - ThimelEdu Component Diagram

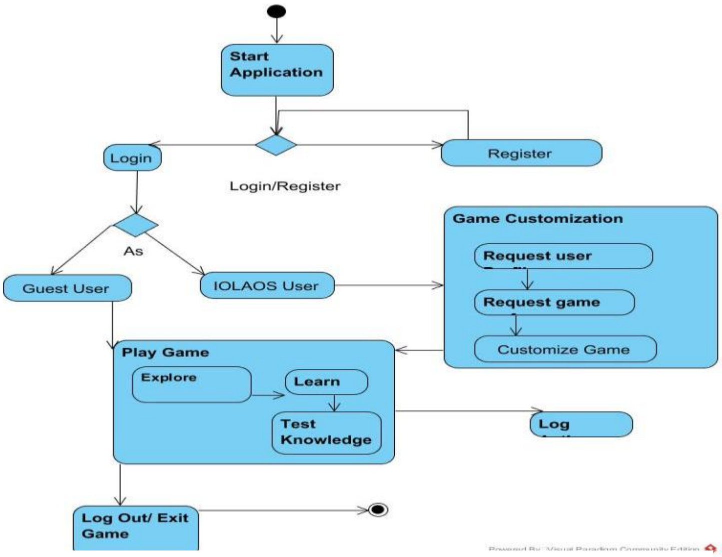


Figure 9 - ThimelEdu Activity Diagram

Equally important is the design and implementation of the modality that will handle the information received, transform it to variables understood by the engine and change game settings in order to realize what the information dictates. Additionally, this modality must handle user dictated changes to settings, changing the game at runtime according to user input.

An overview of all components for ThimelEdu and the relationships between them is summarized in the component diagram illustrated in Figure 8. Additionally, in Figure 9 the activity diagram illustrates the flow of events for a complete session of the game. As highlighted by both diagrams, communication with IOLAOS is essential for customizations within the game. As such the two components are entwined and crucial for the normal function of ThimelEdu.

Communication with IOLAOS:

User data and logs are stored at the CDS of IOLAOS. Each certified developer has indirect access to the CDS through API services. User authentication, user profile and game preference retrieval are the three services that will support ThimelEdu

Since ThimelEdu doesn't maintain servers and all data is stored on the CDS of IOLAOS, the game heavily relies on successful communication through the API. Fortunately, the framework functions in a simple manner, with uniformity along services. Every communication with the API must contain a game token that is given to the developer by IOLAOS, for security reasons, to identify that the game is approved and has access to the data on the CDS.

To authenticate a user, a HTTP post method is sent containing email address, password and game token. IOLAOS replies with a JSON string containing the data for the user, or a message stating why the authentication failed. In this regard it must be stated that two services are completed with one method call, further simplifying procedures.

After a successful verification a HTTP get method can be sent, containing the user id and the game token, to which IOLAOS responds with a JSON string containing either a set of game preferences or a false Boolean, indicating that no game preferences are set to the user.

Hence, a handler for the communications must be in place. Unity3D has the libraries needed for HTTP communication, but unfortunately lacks in support for JSON objects. Simple objects are supported but arrays, a very important aspect, are not. Therefore, a JSON handler class must be implemented to acquire the information enclosed in the JSON objects. This handler will be accepting a string (the JSON object) as input parameter and will create a C# object equivalent to the JSON. The C# objects will be predefined according to the structure of the JSON objects IOLAOS uses as replies.

Game Customization:

The C# objects created by the above procedure contain valuable information that must be employed. However, not all of it is applicable in ThimelEdu. To harness the information necessary and make it available for the needs of the game, a simple solution is designed where all data is stored in one class for reference at any point, while a second class is handling the contents, the settings and does all the procedures needed, according to the data in the storage class.

The handler class extracts the useful information from the C# objects and stores them in the storage class, as variables, and immediately makes any necessary changes to content and settings so that the game conforms to user profile and preferences.

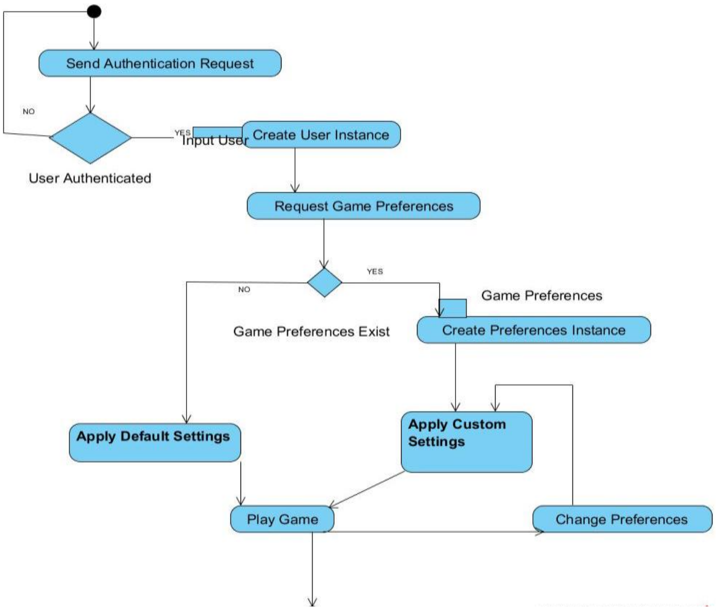


Figure 10 - Detailed Customization Activity Diagram

Useful information for the content of the game include preferred language, school type and learning criteria such as feedback and rewarding. Preferred language is considered for translation, with English being the default. School type is used for selection of articles, audio, video, images and questions, so that appropriate level is ensured without the need of specifically creating game levels for different schools. Learning criteria is used in a true-false fashion, where the represented elements either are included in or excluded from the game. Feedback refers to an in-game helper providing information. Rewarding enables or disables all rewarding methods, which, in ThimelEdu, are the simple score system and the bunch of grapes reward. The diagram showed earlier in **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.** can be modified to show the synergy between the communication and customization modalities in more detail, now that they are defined. The resulting diagram is presented in Figure 10.

Content Management

Most games consist of static, predesigned, content. This is usually what game designers and developers prefer, since that way they can have creative control over their game and the experience of the user. However, ThimelEdu targets different audiences than most games, and with learning being the main goal, educational expertise is much more valuable than game design expertise, even though the collaboration of the two factors can offer the best from two worlds. Therefore, in the context of IOLAOS and the game customization module, a system to manage the content of the game was required. Content for the game is divided into numerous types, thus the content manager must be agile in order to successfully handle the differing forms of data. To do so, each element, regardless of type, should include values, as defined by IOLAOS, that will signify for what audiences it is created. Thus, the content manager can use algorithms to dynamically decide what content to use at runtime.

Localization

Multiple tools are available for localization, including a Localization Manager built-in to Unity and various tools offered in the Unity Asset Store and other places. Most localization solution are based on a key-value set where multiple dictionaries include the values for different languages. However, despite the availability of solutions, creating our own modality, specific for ThimeEdu, was considered appropriate, considering the data structures used for the content and the gameplay specifics that are not covered by those solutions.

Most localization solutions for unity use dictionaries in form of xml files because of the inherent support within unity. However, for uniformity reasons JSON will be used for ThimelEdu dictionaries since the essentials for JSON functionality will be created. Each form of content will be stored in different JSON file, for manageability, and localization will happen through a localization class for static content that will simply need translation, or through other classes that handle more functionality when it comes to content that is affected by personalization and IOLAOS. As such, the localization class will use the dictionaries to translate everything static within the game to the language desired, while any dynamic content will be selected at runtime through an algorithm, as mentioned above, that will consider language among other things.

Synergy of the modalities

It is clear that the aforementioned modalities are not only critical for the game, but also to one another. it is obvious, for example, that without input from IOLAOS, game customization is pointless, and without game customization there is no reason for communication with IOLAOS. Even though they do function autonomously, the essence of their existence is in collaboration in pursuit of the realization of the vision of IOLAOS and ThimelEdu.

4 Pilot Implementation

During part of the development phase of ThimelEdu, IOLAOS was also under development. Consequently, the workflow for development of ThimelEdu was, to a large extent, dependent on IOLAOS's progress. Under these circumstances, the customization module was initially developed to handle settings applied by the user and was later expanded to automatically apply settings based on the information retrieved through IOLAOS. The communication modality was also built gradually, employing each service when introduced. Authentication was the first service to be finalized, followed by the user profile and game preferences services. Similar was the relationship between programing and 3D models' creation, as they were done concurrently.

Hence, ThimelEdu's development was comprised of a workflow entailing collaboration between fields with different responsibilities. Within this environment a continual conversation was ongoing, triggering inspiration, redesign and improvements across all elements. As such, ThimelEdu's workflow is less official and restricting than those commonly used within industries, but still more intricate and collaborative than the work of a sole individual.

Leveraging this workflow, we managed to reach the goals defined by the design as analyzed earlier. Thus, creating elements and modules that would ultimately construct a game with the expected characteristics.

Communication Modality

Communication between ThimelEdu and IOLAOS is of central importance and is therefore treated as such. Utilizing and expanding upon the functionalities offered by Unity, a simple system was built, through which, communication with IOLAOS happens.

IOLAOS's communication is built using Restful APIs, that in turn are based on two methods of data exchange. HTTP methods and JSON strings. Therefore, ThimelEdu had to have the ability to create and send HTTP requests and handle JSON strings as input. Unity features a class that has all the functionalities needed to send POST and GET request in a very simple fashion, as illustrated by the following code snippets for a POST and a GET request.

Table 8 - HTTP Request Code

```
        /*POST request, email, password word and game token are
posted*/
        WWWForm form = new WWWForm();
        form.AddField("email", mail);
        form.AddField("password", pass);
        form.AddField("game_token", data.getGameToken());
        UnityWebRequest www = UnityWebRequest.Post(url, form);
        yield return www.Send();
```

```
        /*GET request */
        UnityWebRequest.GET(url);
        yield return www.Send();
```

POST methods are used to send data. GET methods, on the other hand, are used when data is expected as a response. Within IOLAOS' API however, POST is sometimes used when data is expected as a response. But this does not cause any problems within the framework as IOLAOS handles the requests and responds accordingly. Regardless of the method used, the UnityWebRequest object, seen in the table above, gets populated with a JSON file handled as a string assuming no errors in the connection.

JSON Objects are not supported in C#. Therefore, to incorporate them, Serializable C# classes were made to correspond to each JSON response expected. The JSON is parsed as a string and broken into parts according to the syntax of each JSON. The C# class is then populated with the values contained in this string.

This modality is currently responsible for user authentication as well as data retrieval. Within this modality, the methods for JSON parsing are implemented, therefore it is used for local and remote data retrieval, as even local data is stored in JSON form. Local data retrieval includes the content of the game and dictionaries utilized by the localization modality. Remote data refers to data retrieved from IOLAOS, currently user profile and game preferences, utilized by the customization modality and localization modality.

Customization Modality

ThimelEdu's personalization relies on the user and preferences classes that are described above. However, it is designed to function even if there is no communication with IOLAOS, for unexpected reasons or because the user has decided to play as guest. Assuming the player has logged in with IOLAOS credentials, the user class will be instantiated and populated with his user profile. Otherwise it will be instantiated as a null object. The same process happens with game preferences. However, in this case, IOLAOS might not have set game preferences for some users. In these cases, as when the communication is not possible, the game preferences class in instantiated as a null object.

The core of the customization modality is a class that stores all valuable data. We call this class model and it behaves like a database for the game. It has no functionalities other than storing variables, changing them and distributing their value for use inside other classes.

Around the core, all the functionalities of this modality are executed by another class, called Customization Handler. This class is responsible for transforming values into valid settings codification and storing them in model, applying the values stored in model and updating settings, content and the values in model whenever something triggers such a change.

When the game starts, this modality automatically customizes the game depending on who the player is. At this stage, there are three possible levels of customization

- No customization

  When the user class in a null object, there are problems in communication with IOLAOS or the player has logged in as guest. Regardless of the cause, a default set of settings will populate the model. The game will then be regulated according to these values.
- User with no preferences

  In some cases, IOLAOS will not assign preferences to some users. In that case, the user profile is populated, but the game preferences are null. Therefore, preferred language, school type and username are utilized within the game, but for the settings deriving from the game preference class, the default set is again used.
- Full customization

  In this case IOLAOS provides ThimelEdu with a complete set of data for the user and the game is customized accordingly. This is the most unique gameplay, the full personalization aimed for, and what is most desired.

However, customization doesn't end with the automatic assignment of values, settings and content in the beginning of the game. A menu for settings is always available, from where players can change settings concerning fonts, visual and audio elements, movement and language, whenever they wish. When the user changes a setting, the Customization Handler changes the value in model and applies the new setting to the game. A very simple and common procedure.

Localization Modality

Localization is widely used today. However, for a game with heavy customization that affects the information presented to the user, localization does require some more attention. The architecture of the game is such, that two types of texts can be found, static and dynamic texts. Static texts are those that are always present, like texts on buttons and the text creating the GUI. Dynamic texts, on the other hand, are the articles and question that the user is presented with. These texts must be selected at the right time, depending on the user and the circumstances, a process described below, therefore they can’t be handled in the same fashion as the static texts.

The most common solution for localization is the usage of dictionaries, where all texts needed are included in all the languages supported. Thus, translating any text or all texts, is a matter of changing the text to the corresponding text in another language. This is the basis on which ThimelEdu’s localization was done. For the static texts, a JSON file including all the texts in all supported languages is the dictionary. When loading the game, this JSON file is turned into a C# class, with the help of a part of the communication modality for handling JSON, that acts as the dictionary at runtime. Along with this procedure, a reference to each object containing each text is created. From that point on, translating the game happens with a simple method that sets the text of each object to the text of the language selected.

In the case of dynamic texts, a different, but quite similar, approach was chosen. This solution will be presented bellow along with other procedures concerning the selection of dynamic texts, as it is part of these procedures.

Content Management

One very important aspect of the educational value of the game, is the information and questions offered through the interaction with the artifacts. These articles must be chosen at the time of interaction, to fit the artifact, the progress already done regarding this artifact and the customization particular to each user. To achieve this, all articles and all questions are stored in JSON files that are parsed when the application starts. Each article and each question is accompanied by some properties, such as id, school type, language etc. Thus, when an interaction occurs, the Customization Handler defines the requirements for an article or question, and a search through the content returns all applicable information, that is then presented to the user. When there is more content meeting the criteria than can be shown to the user, if for example there are two articles when the game only presents one article at each encounter, the one to display will be randomly chosen. This way, all things that must be considered for the content are handled in a central process instead of having multiple lists and dictionaries that would have to be accessed with various and numerous methods.

Personalization Process

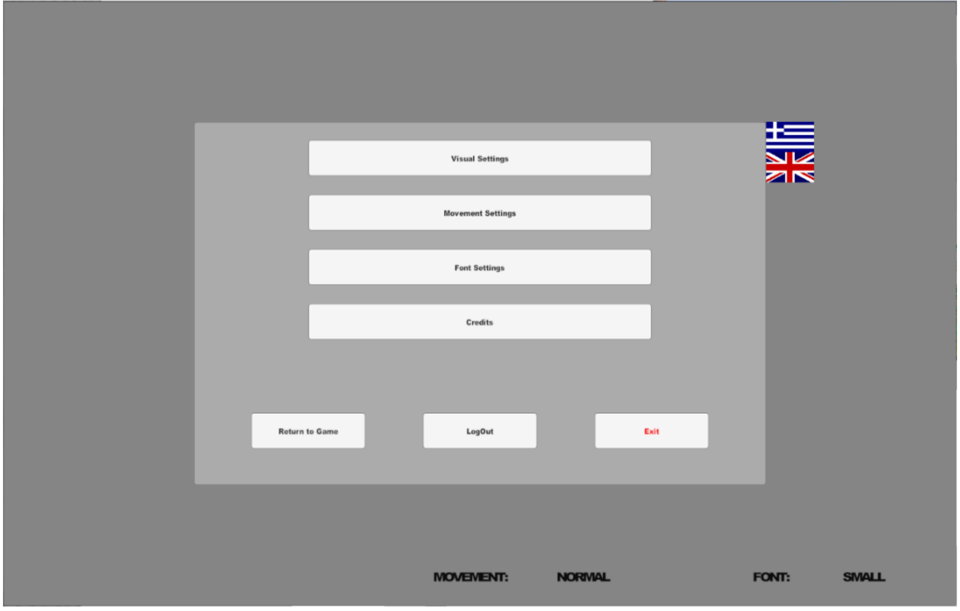
The most important part of the game is the personalization process. As described above, a module was created to handle different settings and another to handle the display language. On the initial login these two modules will personalize the game for the current player. Any player that logs in with IOLAOS credentials will experience the game according to preferences set by an educational expert, specific for this user. In the following table the preferences that are manipulated by the system are presented, along with their default values. These are a subset of the standardized IOLAOS preferences, as other preferences would be irrelevant for ThimelEdu.

Table 9 - Default Preferences with Values

Default Values	
Visual Preferences	
Shader	Full Color Spectrum

Field of View	60
Shadows	On
Brightness	1.0
Movement Preferences	
Walking Speed	3
Running Speed	5
Jumping Speed	5
Mouse Sensitivity	2.0
Sound Preferences	
Volume	1.0
Mute	False
Sound Effects	True
Reverb	True
Stereo Pan	0.0
Font Preferences	
Font Size	22
Font Style	Normal
Font Color	Black
Font	Arial
Learning Preferences	
Vibration	True
Feedback	True
Intervene	True
Rewarding	True
Language	
English	

Those default values are used for players that are not assigned game preferences, namely guest users and IOLAOS users that educational experts have decided should experience each game as initially designed. On the other hand, when a set of game preferences is provided, these values will vary, within each one's scope.



MOVEMENT: NORMAL FONT: SMALL

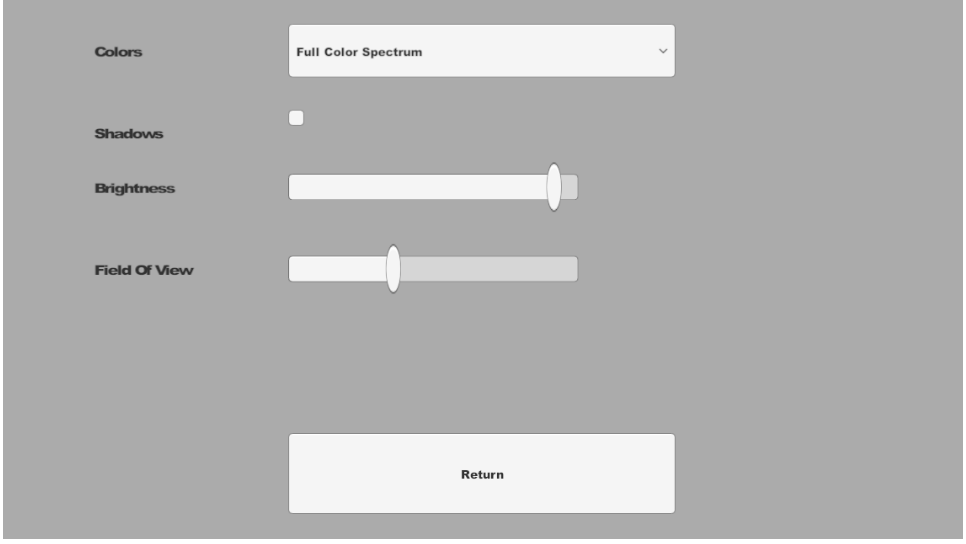
Figure 11 - Main Menu

However, regardless of the preferences assigned, the ability to change some of them is always available to the player through a menu. The main page of the menu can be seen in Figure 11. The main reason for offering this option is that educators or educational experts might need to experiment with the settings they offer τo learners. Also, since automatic customization does not apply to all users, the ability should be open to all users and not only a subset of them.

Game Preferences Explanation

The preferences, as presented above, are grouped by IOLAOS and we choose to follow this grouping too, as it is convenient in many aspects.

Under visual preferences, the menu shown in Figure 12 , there are a set of values that affect the visual representation of the game world. Shader is a piece of software that handles lights, colors and shadows within a scene. By changing the shader, the objects within the scene will be rendered with a different algorithm, resulting in a different view of the same scene. This way the game can be made suitable for people with any type of color vision deficiency. Currently ThimelEdu supports a “normal” shader that is a default by Unity and supports the full color spectrum, and one for greyscale representation of the world. Shadows and brightness are values within the shader, however they are given as an extra option, as they are a critical setting, that can quickly and easily change the visuals without the need for thousands of shaders. As their names states, they represent how bright the object within the world should be and how hard the shadows should be. Both have a double value that is multiplied with the default setting.



Colors: Full Color Spectrum

Shadows: □

Figure 12 - Visual Preferences Menu

Field of view is the angle at which the camera renders, and therefore how wide of a view the player will have, corresponding in a way to human visual field. However, values for the two concepts differ a lot. While a human visual field is typically 210 degrees, a field of view that is considered normal is usually about 60 degrees. Therefore, the default for ThimelEdu is 60 degrees, and any integer between 1 and 360 falls within scope..

Under movement preferences, shown in Figure 13, we have settings that change the way the player avatar can move within the world. Walking, running and jumping speed obviously represent the speed at which the player moves, while mouse sensitivity is a number that handles the speed at which mouse movement is realized in game. IOLAOS does not specify a running or jumping speed, however running is specified as a Boolean while jumping is left to the game designers. Jumping and running were designed into the game and deemed essential. However, within the implementation of the player avatar having actual speed for running instead of a Boolean value made more sense. Therefore, if IOLAOS wants the player to be able to run, a formula is used to calculate running speed. Otherwise running speed is equal to walking speed, therefore creating the same effect as if running was disabled. Similarly, depending on the walking speed a formula is used to calculate jumping speed.

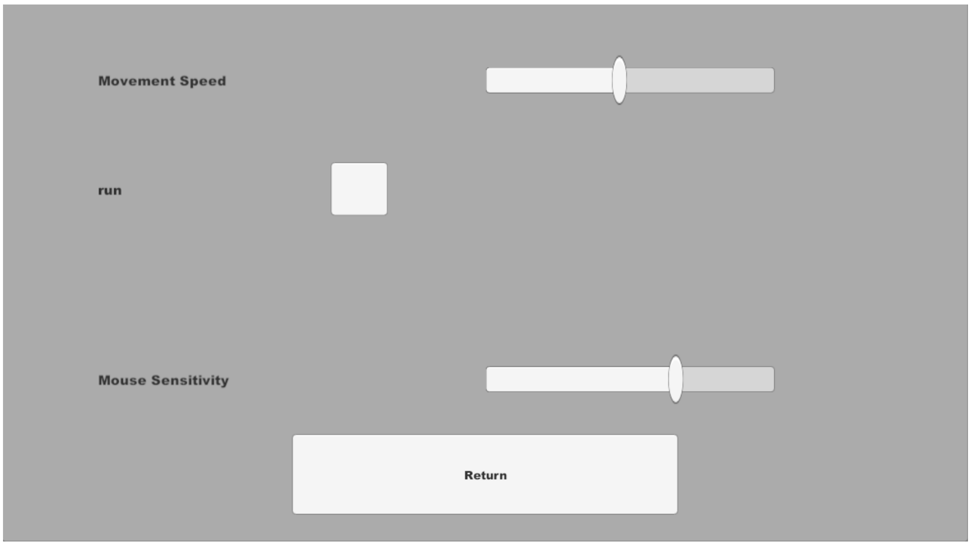


Figure 13 - Movement Settings Menu

Speeds are always relative to the game world. To solve this problem, IOLAOS uses a simple codification in human language, “slow”, “medium” and “fast”. Therefore, the values retrieved from IOLAOS are strings, but need to be integers when applied to the avatar. A simple value replacement based on the table below makes this happen.

The values for these settings come as a string from IOLAOS but need to be integers when applied to the avatar. Furthermore, speeds are always relative to the game world. Therefore, a simple table for correlating values was created and utilized to make IOLAOS values into game specific integers.

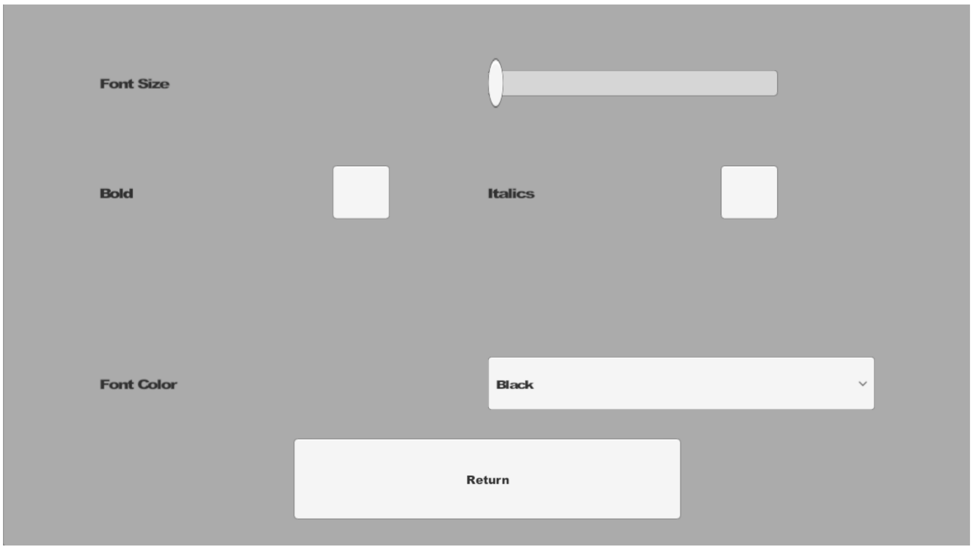
Table 10 - Movement Values Correlation

IOLAOS Value	Walking Speed Value	Mouse Sensitivity Value
DEFAULT	3	2.0
SLOW	2	1.0
MEDIUM	4	2.0
FAST	7	2.5

Font preferences, shown in Figure 14, apply to texts and are crucial since age and reading ability vary largely within the audience of the game. Font size dictates how large, or small, the font will be. As with movement settings, those numbers are relative and IOLAOS encodes this with a string. Therefore, another table, Table 11, can represent the correlation used for fonts. For font style there are 4 options, “normal”, “bold”, “italic” and “bold and italic”. Font color is an RGB value, a color represented in values of red green and blue that blended will create the color that represents a specific color that will be used for the font. As for fonts themselves, IOLAOS offers a wide selection, but ThimelEdu only support the most prominent as there would be no value in using more fonts. Namely, these are Arial, Arial Black, Helvetica and Verdana.

Table 11 - Font Sizes Correlation

IOLAOS Value	Font Size Value
DEFAULT	22
SMALL	16
MEDIUM	20
LARGE	28



Font Color Black

Figure 14 - Font Preferences Menu

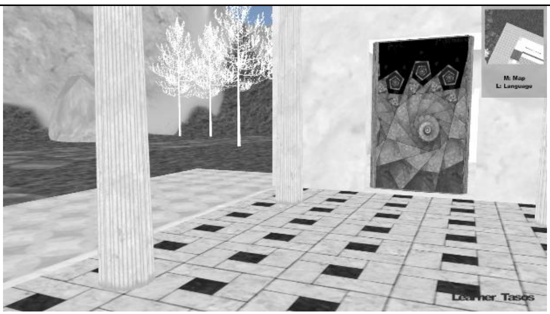
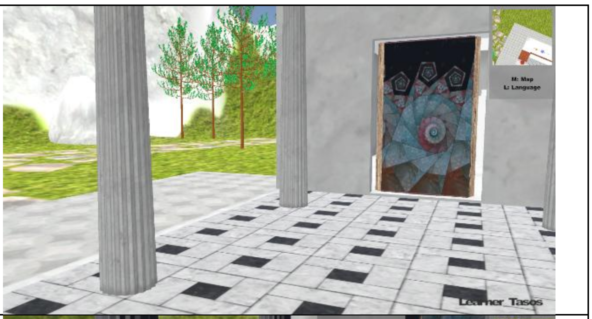
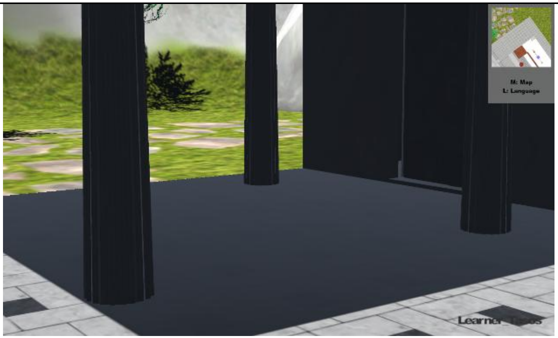
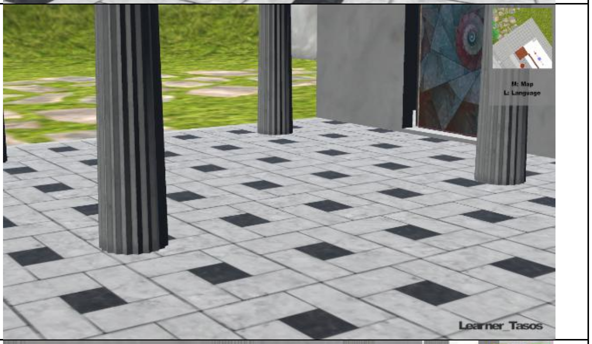
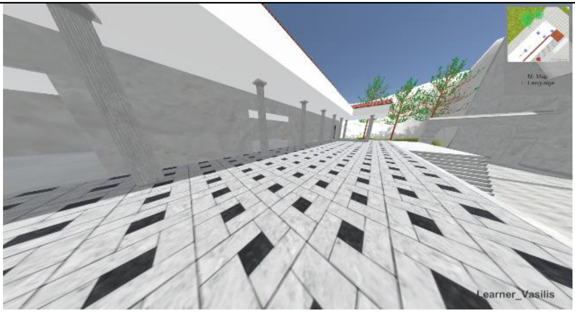
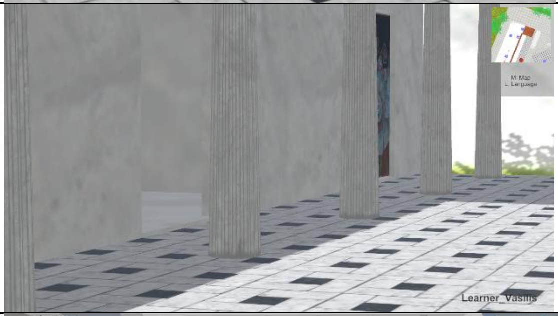
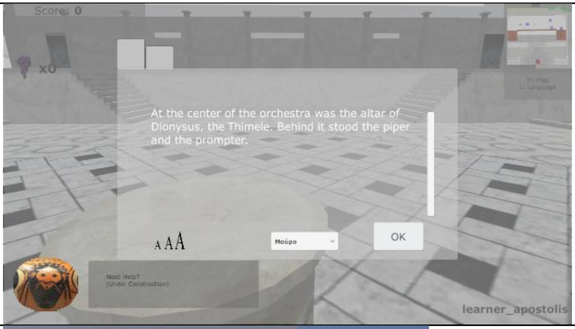
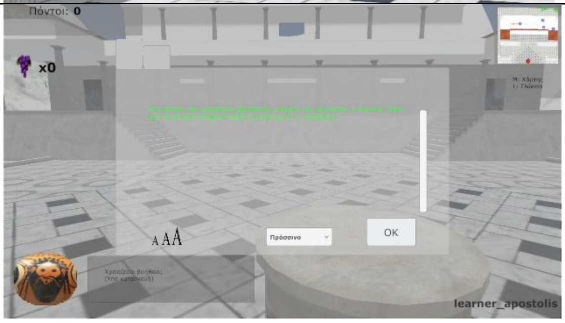
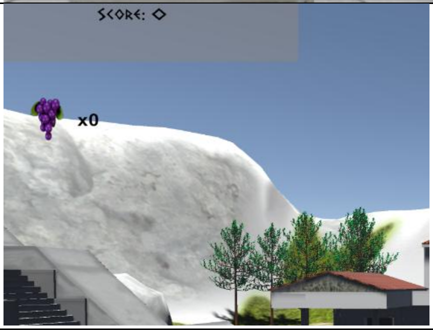
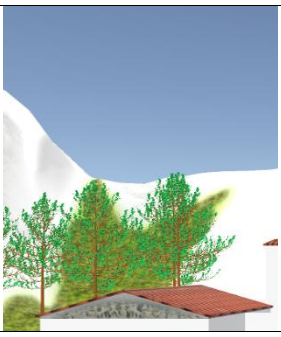
Sound Preferences, not yet implemented in a menu and therefore only adjustable through IOLAOS, refer to any audio settings. Volume is used as a measure for loudness and is a float value representing percentage of full volume. Mute is a Boolean that will either enable or disable all sounds of the game. Sound effects, also a Boolean, regulates whether audio effects (such as spatial blend, pitch, etc.) will be used on the audio samples used in game. Reverb, even though a sound effect, is a part of the IOLAOS preferences, and as such ThimelEdu supports it independently of the other sound effects. Stereo pan refers to a bias towards one of the speakers in a stereo setting. As such, it is a float value between -1 and 1, where audio output is biased left with negative values and right with positive, while 0 has both speakers functioning at full capacity.

Lastly, learning preferences, only adjustable through IOLAOS, are settings that change the game to correspond to various learning theories. All of them are of Boolean type that either enables or disables parts of the game to create a learning environment as suggested by learning theories. In this sense, rewarding is implemented to enable or disable the scoring system and all related elements. For the rest of the learning preferences, consulting from educational experts is needed, therefore it is left for future work.

Personalization Examples

Some of the personalization preferences can be seen in Table 12. The first two images show the difference between two shaders, first is the one that uses greyscale representation and second is a full color spectrum. Then we have two images that show the difference of having shadows or not. Third is the illustration of the difference between a very large and a very small value for field of view. After that we can see some differentiation in text. The user in both images is the same, however, in the first picture he has the default values for him, a large text in white color and English language while in the second he has changed his preference language to Greek and the text to a small in green color. Finally, we can see the difference between a user with rewarding and one without.

Table 12 - Personalization Examples

Shader		
Shadows		
FOV		
Text	At the center of the orchestra was the altar of Dionysus, the Thimele. Behind it stood the piper and the prompter. Score: 0 x0 M: Map L: Language A A A Μαύρο OK Need Help? (Under Construction) learner_apostolis	 Πόντοι: 0 x0 M: Χάρτης L: Γλώσσα A A A Πράσινο OK Χρειάζεσαι βοήθεια; (Υπό κατασκευή) learner_apostolis
Score		

4.1 Presentation of the game

The game in its current state consists of 2 scenes. The first scene is a login screen, designed to be as minimal as possible while still communicating its purpose. In the second scene all the gameplay is implemented. They each carry a different load of background tasks, as will be analyzed in detail bellow.

Login Scene

In the login screen a simple GUI is presented where the options are to register with IOLAOS, login, continue as guest, change the language or exit, as shown and explained bellow.

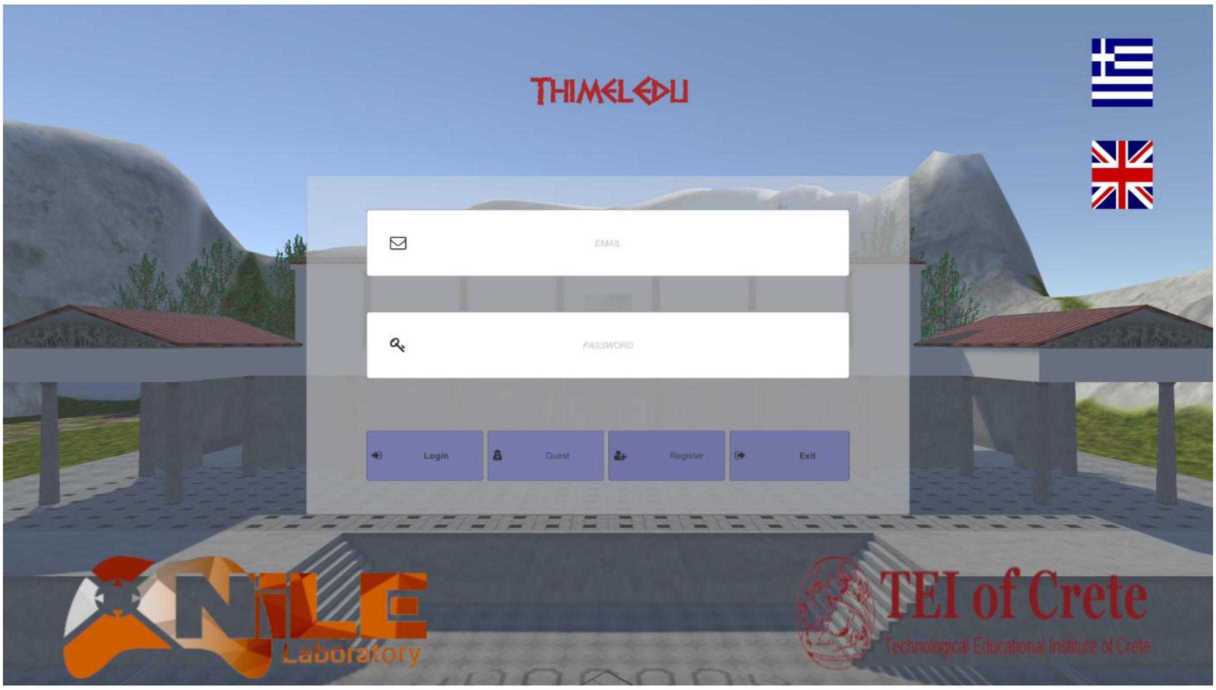


Figure 15 - Login Scene

<u>Login</u>

To successfully login, users must provide their IOLAOS credentials, namely email and password. When the "Login" button is pressed, the information along with a game token, unique to ThimelEdu, is sent to IOLAOS through the API. IOLAOS immediately responds with a JSON string signifying the success or failure of the process, along with some user details if succeeded or the reason for failure if not. This procedure is handled by the modality for communication with IOLAOS, as already presented.

Depending on the reply from IOLAOS, a failure message explaining the issue is presented or the game moves on to the gameplay scene, while setting up the personalization.

<u>Register</u>

For registering, the game will be minimized, the systems default browser will be opened and the page for registering to the IOLAOS platform will be loaded. This method was chosen instead of utilizing the IOLAOS register service, because the user should be able to explore IOLAOS website before deciding on registering. Furthermore, even though it would be possible to open a web page inside the game without minimizing it, opening a browser was evaluated as a better alternative to signify to the user that a third party is involved.

<u>Continue as guest</u>

In case the user doesn't want to use IOLAOS for any reason, he is presented with a choice to circumvent login and continue to the game. This way, all IOLAOS functionality is circumvented, such as the automatic customization. However, the user still has the ability to manipulate some of the game settings through a menu.

Change language

When the application starts, the language displayed is English, as it is the second most spoken language in the world and the third with the most native speakers [30]. Additionally, since some users might not understand English, all buttons and fields include symbols to make the function clear. Furthermore, two buttons can change the language between Greek, since the main audience for ThimelEdu are Greek students, and English. Those buttons are represented by the English and the Greek flag to make their functionality apparent. When one of those buttons is pressed, the current scene is translated into the corresponding language, by the localization modality, as described above.

Gameplay scene

Contrary to the login scene which is simply a necessary step, this scene includes the purpose of ThimelEdu. All the gaming, gamification and educational value is harnessed here.

Background Tasks

When the player logs in, the communication and customization modalities contacts IOLAOS to authenticate and retrieve game preference data and then personalize the game to create the best possible experience, in the way customization was described earlier. In the case that the user continues as guest, there is no communication with IOLAOS and the customization follows the default values, as defined by us instead.

After the customization is complete, the player is spawned in the kerkis, where the audience would sit, and in front of him he can see the orchestra and the skene. A mini map, the username of the player and information about scoring, how to change language and view a bigger map is presented on the HUD. The first impression the user experiences is the image bellow.

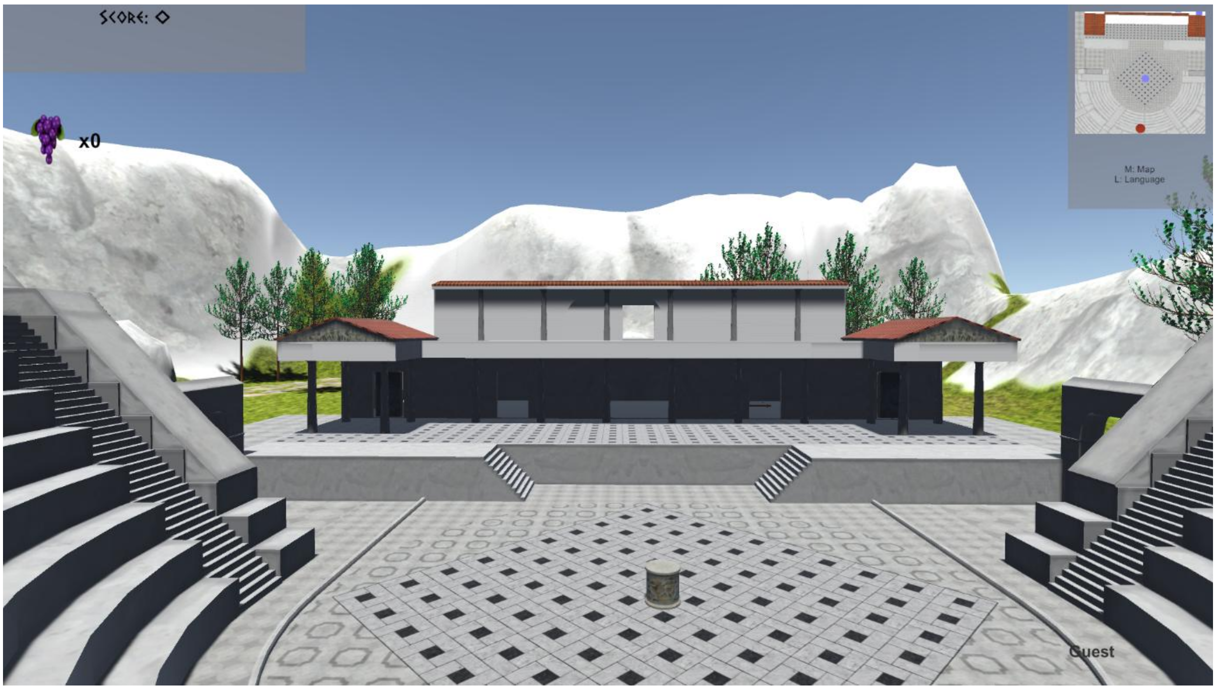


Figure 16-Player Spawns

After spawning, the player has full freedom of actions. Movement is free within the limits of the map and interaction with the artifacts is not limited to any specific order or number of times.

Mini-map & large map

Most maps in games are implemented in one of two ways, either a pre-rendered image is used, or a camera is rendering the world in real time. Both techniques have advantages and are preferred depending on the usage scenario. For the purposes of ThimelEdu, the camera solution was chosen, due to practicality, as it is very simple to be made in Unity without extra graphics work, and the accurate representation. Additionally, with the use of game objects, rather than icons, a real-time depiction of locations of interest is achieved in a very simple manner, using layers provided by unity and without writing any code.

In more detail, two maps that are an actual, real-time representation of the game world were made. One of them, the mini-map, is always displayed on the HUD for assistance in navigation and providing information on the states of the interest points around the player. As a navigational tool, the range of view for the mini-map is quite small. On the contrary, the second map, called large-map, is displayed on explicit command of the user, and has full range over the map. When the large map is active, a useful map legend is displayed, and player movement is disabled until the large-map is closed. The visual of the map can be seen in Figure 17 bellow.

Both maps were created in the same way, with small variation in camera properties and settings, resulting in slight differences. To capture the image to be shown on the map, an orthographic camera is used for each map. Those cameras are located above the scene facing downward to get a map perspective of the world. The feed from each camera is passed to a raw image, that is used as source by an image placed on the GUI layer.

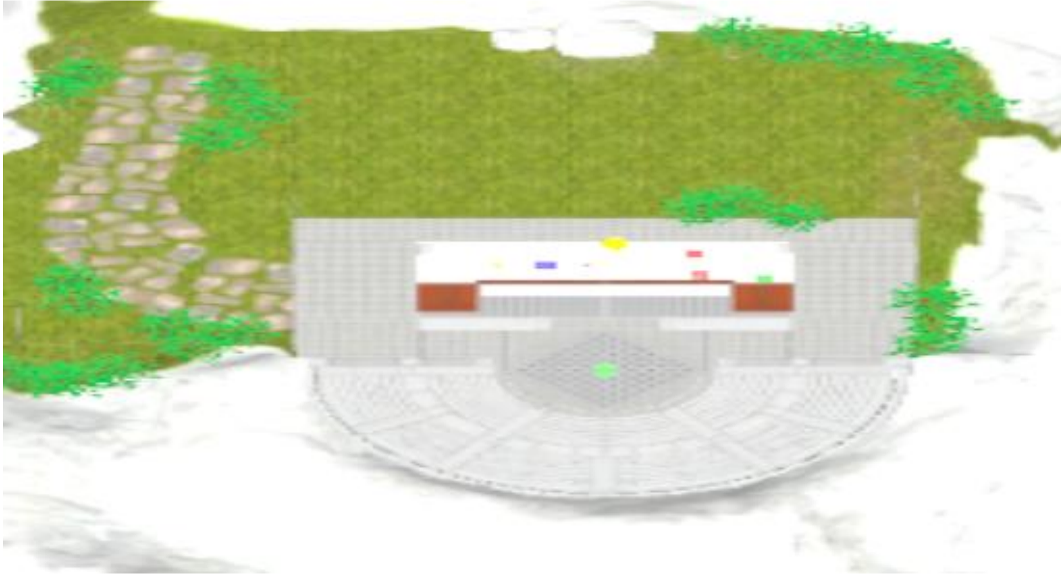


Figure 17 - Game Map

The mini-map will be displayed while the player is moving around. Therefore, the right context, namely perspective and range, are crucial within real-time. Utilizing Unity's hierarchical architecture, the camera, as a game object, can be defined as a child object to the player within the Unity Editor, thus changing location along with the player, while maintaining their relative distance.

To complete the maps, icons are needed to signify where the player and the artifacts are located. Artifacts have static locations, but states may change, while the avatar of the player constantly changes position but never the state. However, to achieve uniformity, modularity and expendability, both cases are treated in the same way. Again, using only Unity editor tools, child objects are created for each game object, that need to have presence on the map and those objects are placed on a different layer, that is viewed only by the two cameras used for the maps. The different states of the artifacts are then signified by change in the color of the objects representing each artifact. These colors change through the scoring process that will be explained later.

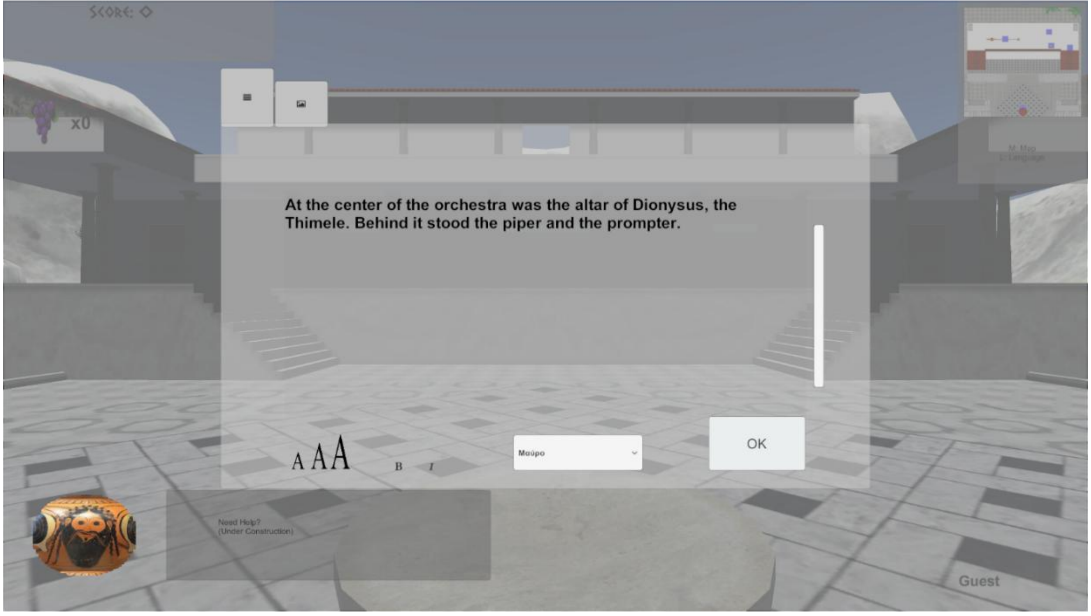
Scoring

The scoring logic as described during design is uncomplicated, therefore the implementations is equally plain. A handler class controls all related functionality, by initializing values at the start of the game and including methods for incrementing the values and updating the GUI elements with new values. A separate class handles the question and decides whether an increment must happen, and the score handler only applies the change.

Informative Dialogue

Upon the interaction with an artifact, which has been detected by utilizing collision mechanics, a dialogue window supplies information about the artifact.

Within the window there are multiple tabs to deliver different forms of information. In the current state, only articles and images are used, however audio and video will soon be integrated, and the system is built to be extendable with easy integration of any other media not foreseen during the initial development.



At the center of the orchestra was the altar of Dionysus, the Thimele. Behind it stood the piper and the prompter.

Figure 18 - Article presented within the Informative Dialogue

Articles consist of text only and the tab that presents this text is designed accordingly. A scroller is in place to facilitate reading of large articles, and buttons that modify the font are readily available for a better reading experience. The image tab is similar, but with less buttons, since no modifications are needed. In both cases there is a single exit point, the “OK” button, which forwards to the Quiz dialogue. Details on the layout are given in 3.1.1, article layout are illustrated in Figure 18 and image layout is presented in Figure 19.

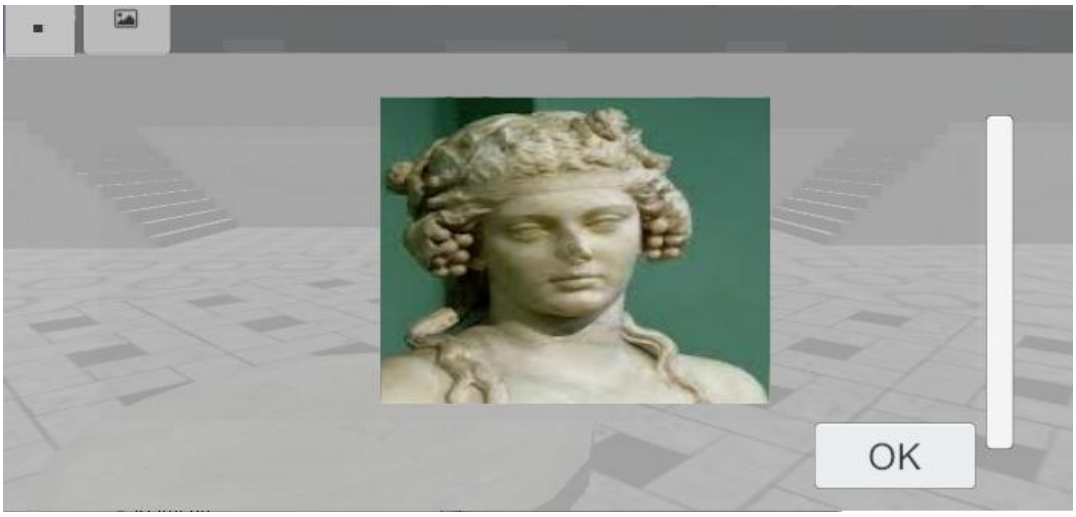
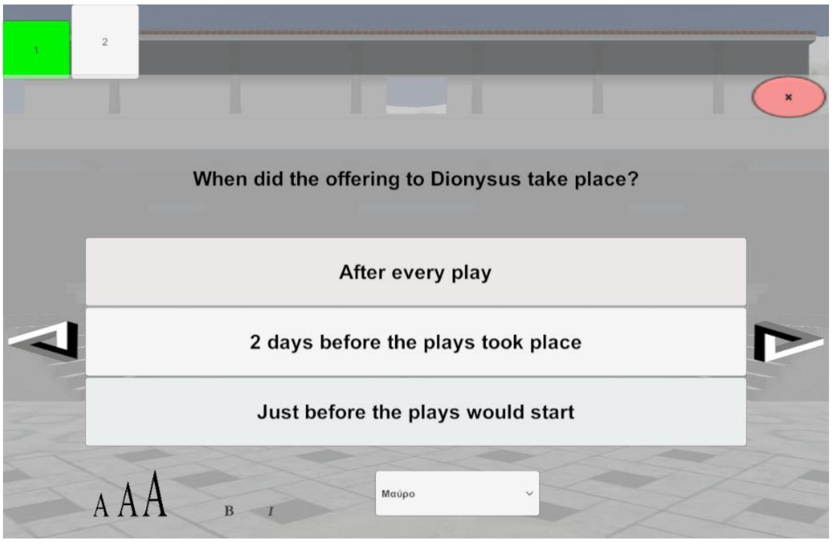


Figure 19 - Image presented within the Informative Dialogue

Before the window is presented to the player, the selection process explained earlier, chooses articles and images related to the artifact. When the window is instantiated any number of articles and images may be suiting, therefore one of each will be randomly chosen. Upon a second interaction, the same process will be followed, therefore the information provided may vary between interactions. The case that one, or more, forms of information will not be available is a possibility, especially after additions and modifications. Therefore, to avoid any unexpected and unwanted behavior, each tab within the window is created after the corresponding information is detected. Therefore, only tabs that include information are included in the window and if no information is shown, the dialogue is not created, and the player is immediately presented with the quiz dialogue, or if no question exist, the game continues.

Quiz Dialogue

After the informative dialogue closes, the quiz dialogue is instantiated. Here, the player is called to answer questions about the artifact. This window has many similarities with the informative window. They share a common selection process, similar design and tab logic. However, here there is only one form of content, as question conform to the same format, and there is no random selection. All the questions related to the artifact will be shown, filed under tabs that are instantiated in a similar logic with the tabs of the Informative Dialogue. One tab will be created for each question found. If no questions are found, the artifact will be marked as completed and the game resumed.



When did the offering to Dionysus take place?

1
2
×
After every play
2 days before the plays took place
Just before the plays would start
A A A
B I
Μαύρο

Figure 20 - Quiz Dialogue

As seen in the Figure 20, navigation through the questions is available in two ways, directly through the tabs or with previous and next buttons. Additionally, there are buttons to modify the font and an exit button. Besides the exit button, there is also a logical exit point. When all questions are answered the dialogue will close and the game will resume. More details on the layout objects can be read in 3.1.1.

A color-code is utilized on the tabs to signify the state of the question within them. The unanswered have a grey color, the currently open is white, the incorrectly answered are red and the completed are green. Additionally, the current tab will be larger to stand out from the rest.

When answering a question, correctness is signified by visual and auditory messages. A green tick sign along with a happy tune signifies that the answer given is correct, while a red “X” along with a sad tune signifies failure. When the answer is correct, the buttons for this question are disabled and the user can no longer press them. However, the option to view the question is still available.

Upon a second instantiation of the dialogue, the tab shown will be the first unanswered (or incorrectly answered). In the case that the artifact is completed but there is instantiation, the first tab will be shown.

5 Results

As analyzed above, it is imperative that innovative methods of teaching be introduced and utilized within the classroom in order to facilitate learning, especially for subjects such as ancient theater. In addition, digital technologies offer great possibilities within the field of special education that have not been sufficiently explored and utilized yet. Therefore, it is critical that special education is included in such actions, so that the gap in knowledge and skills will be reduced, or even eliminated.

ThimelEdu attempts to address this by providing a solution for gamification and experiential learning that will hopefully be integrated in official educational processes. Simultaneously, ThimelEdu along with IOLAOS considers the specificities of each user, creating individual experiences and environments, thus facilitating the learning process.

5.1 Conclusions

In conclusion, this thesis was successful in creating a game that offers innovation of both technical and pedagogical nature. Pedagogically, knowledge of ancient theater is promoted through gamification, experiential learning and a ludic environment, automatically configured to cater to each users' needs. In a technical matter, it is proven that IOLAOS, as a new platform, greatly assists in creating serious games to be more effective and inclusive.

5.2 Future Works

Creating a game is not an easy task. Game creation demands, among other things, a lot of resources and time. Commodities not always available within the timeframe of a thesis. ThimelEdu, as presented in this thesis, is not yet final. Our vision of ThimelEdu is much different. We want to enrich and expand the game as it is presented, but we also want ThimelEdu to evolve into a platform for ancient theater, where the current state of ThimelEdu will be just one out of many games. Assuming time and resources can be found, we hope to bring life to this vision.

As a first step, the goal is to enrich the game to include every element of the design that has not been completely integrated. After that we wish to redesign and expand ThimelEdu. This process will depend on the inclusion of several features, such as the following.

- xAPI implementation. Logging the game is very important for learning sessions and to assist teachers in their work. Unfortunately, there was not enough time to implement a modality for xAPI within the timeframe of this thesis, but it is the first thing to be implemented in the future.
- Black and White. While the game has a setting for greyscale colors, there are a few bugs, including that the HUD never becomes greyscale. This will be issued in the short future.
- Voice acting. The articles and question presented in game will be recorded and the option of listening will be provided by a button. This will be helpful in learning for children of younger ages, auditory students and students with special abilities or special educational needs.
- Age affected content. The content management will be enriched, along with the content, to offer different content according to age group.
- Introduce more content. Content is one difficult matter for the game, as it is the most important element of the game, but it requires historic research, a work that takes a lot of effort, patience, perseverance and time. This research is continuously happening, and the results will be imported into the game at regular intervals.

- Introduce more questions. As with content, questions are difficult. Even with the historic research in place, pedagogy must be taken into consideration for questions to be formed. But, this is also a continuous work that will slowly mature the game.
- Localization. Currently only Greek and English are available in game. However Swedish will be introduced soon. We also hope to be able to bring more localizations in the future.
- Introduce open-ended question. Since ThimelEdu is designed to be a classroom activity, we wish to explore the possibility of providing open-ended questions, to promote creative thinking, initiative and writing skills. The answers to such questions will be evaluated by the teachers, creating a hybrid environment combining the benefits of digital and conventional learning processes.
- Save & Load system. Currently the game doesn’t have any way to save the progress, thus, there is no possible way to continue from a certain state rather than restart after closing the application for any reason. It is not necessary as completion can be achieved rather quickly, within fractions of a school period. However, with the gradual expansion of the game this will become an issue, and therefore we aim to create such a system.

Another step is to include more games and turn ThimelEdu into a platform for ancient theater studies. In that direction we plan to include several features and functionalities.

- Create more levels to the existing game, where different topics concerning ancient theater will be introduced and examined, such as history of the theater, where different environments will be tied to the content. One wooden theater already exists as a 3D model, and we plan to employ it with questions about the early years of theater, once this kind of historical content is available to us.
- Create another game that focuses on theatre studies.
- Create a game that would use map technology and geographical identification of ancient theaters across the world.
- Create a module where users can watch animation of ancient plays, as they would be presented in ancient times.
- Create a module where users can create their own plays, either following historical ones or writing their own, and create the animations mentioned above, to watch and share with others.

6 References

[1] M. Ulicsak, “Games in Education: Serious Games,” *A Futur. Lit. Rev.*, p. 139, 2010.

[2] C. C. Abt, “Serious Games.” 1970.

[3] B. H. Sørensen and B. Meyer, “Serious Games in language learning and teaching – a theoretical perspective,” *Situated Play. Proc. DiGRA 2007 Conf.*, pp. 559–566, 2007.

[4] M. Zyda, “From visual simulation to virtual reality to games,” *Computer (Long. Beach. Calif).*, vol. 38, no. 9, pp. 25–32, 2005.

[5] R. J. Stone, “Human Factors Guidance for Designers of Interactive 3D and Games-Based Training Systems,” vol. 1, 2008.

[6] “Spelutvecklarindex 2017,” 2017.

[7] Marketsandmarkets, “Serious Game Market by Vertical (Education, Corporate, Healthcare, Retail, Media and Advertising), Application (Training, Sales, Human Resource, Marketing), Platform, End-User (Enterprise, Consumer), and Region - Forecast to 2020,” 2015. [Online]. Available: https://www.marketsandmarkets.com/Market-Reports/serious-game-market-67640395.html. [Accessed: 19-Dec-2017].

[8] M. de Aguilera and A. Mendiz, “Video games and education : Education in the Face of a ‘Parallel School,’” *ACM Comput. Entertain.*, vol. 1, no. 1, pp. 1–14, 2003.

[9] A. S. Drigas and R. Ioannidou, “Special Education and ICTs,” vol. 8, no. 2, pp. 41–47, 2013.

[10] R. Gersten, D. Edyburn, C. Okolo, and E. Bouck, “Defining quality indicators for special education technology research,” *J. Spec. Educ. Technol.*, vol. 22, pp. 49–52, 2007.

[11] “Greek Pedagogical Institute,” 2003.

[12] R. Jordan and S. Powell, *The Special Curricular Needs of Autistic Children: Learning and Thinking Skills*. Association of Head Teachers of Autistic Children and Adults, 1990.

[13] A. Makris and P. Markou, “New Technologies in Special Education,” 2015. [Online]. Available: http://www.scientific-journal-articles.org/greek/free-online-journals/education/education-articles/markou-paraskeui/paraskeui-markou-markos-athanasios.htm. [Accessed: 20-Dec-2017].

[14] S. Charitakis, “Design and Development of Effective and Customizable Serious Game Platform Based on Learning Profile,” 2017.

[15] M. Kalochristianakis, H. Vonorta, A. Stamoulias, E. Lakka, K. Kontakis, and A. Malamos, “Experiential educational environments for cultural heritage,” in *2016 International Conference on Telecommunications and Multimedia, TEMU 2016*, 2016, pp. 6–11.

[16] M. Mortara, C. E. Catalano, F. Bellotti, G. Fiucci, M. Houry-Panchetti, and P. Petridis, “Learning cultural heritage by serious games,” *J. Cult. Herit.*, vol. 15, no. 3, pp. 318–325, 2014.

[17] M. Mortara, C. E. Catalano, F. Bellotti, G. Fiucci, M. Houry-Panchetti, and P. Petridis, “Learning cultural heritage by serious games,” *J. Cult. Herit.*, vol. 15, no. 3, pp. 318–325, 2014.

[18] “The Ancient Theater in the cycle of time.” [Online]. Available: http://ancienttheater.culture.gr/. [Accessed: 20-Jul-2011].

[19] M. Christoulakis, A. Pitsiladis, A. Moraiti, N. Moumoutzis, and S. Christodoulakis, “EShadow : A Tool for Digital Storytelling Based on Traditional Greek Shadow Theatre,” *Fdg 2013*, 2013.

[20] C. Lorenzini, M. Carrozzino, and C. Evangelista, “Serious Games for Disseminating the Knowledge of Ancient Manuscripts : a Case Study,” vol. 3, no. 2, pp. 135–142, 2013.

[21] B. M. Iglesias, C. Fernández-Vara, and B. Fernández-Manjón, “E-learning takes the

stage: From La Dama Boba to a serious game," *Rev. Iberoam. Tecnol. del Aprendiz.*, vol. 8, no. 4, pp. 197–204, 2013.

[22] "Acropolis Virtual Tour." [Online]. Available: http://acropolis-virtualtour.gr/index.html.

[23] F. Messaoudi, G. Simon, and A. Ksentini, "Dissecting games engines: The case of Unity3D," *Annu. Work. Netw. Syst. Support Games*, vol. 2016–Janua, 2016.

[24] M. Lewis and J. Jacobson, "Introduction," *Commun. ACM*, vol. 45, no. 1, 2002.

[25] "SCORM." [Online]. Available: https://scorm.com/scorm-explained/.

[26] "Experience API." [Online]. Available: https://experienceapi.com/overview/.

[27] J. L. Gonzalez, M. J. Cabrera, and F. L. Gutierrez, "Using videogames in special education," *Lect. Notes Comput. Sci.*, vol. 4739, pp. 360–367, 2007.

[28] B. Amy, "Multiple Intelligences: Gardner's Theory," *ERIC Clearinghouse on Assessment and Evaluation Washington DC*, 1996. .

[29] Jesse Schell, *The art of game design*, vol. 1. 2008.

[30] G. F. Simons, *Ethnologue 20th Edition*. 2017.