ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
UNIVERSITY OF WEST ATTICA

Université de Limoges

Development of interactive gaming web application based on video

Σχεδιασμός και ανάπτυξη διαδικτυακής πλατφόρμας διαδραστικού παιχνιδιού βασισμένο σε βίντεο

Μεταπτυχιακό στην Πληροφορική, Σύνθεση Εικόνας και Σχεδιασμό Γραφικών (ISICG)

Dimitrios Arkolakis - ISICG180336

Aegaleo, July 2021

Contents

Acknowledgments

I would like to thank Professor Athanasios Voulodimos for his guidance and help on this postgraduate thesis and my fellow student Dimitrios Mathes for his help on the video shooting.

I would also like thank my parents for their unwavering support, as well as my brother Costas Arkolakis and my cousin Stefanos Gerardis for various conversations on the topic.

Abstract

The purpose of this essay is the planning and development (creation) of an e-game, along with the development of interactive videos by using modern methods and technologies. The application gives the player the possibility to decide the flow of the game according to his own decisions and actions. In addition, the player can build the story based on the sequence of the videos according to the choices he/she will make on each level of the game.

Περίληψη

Σκοπός της παρούσας διατριβής είναι ο σχεδιασμός και η δημιουργία ενός διαδικτυακού ηλεκτρονικού παιχνιδιού, καθώς και η ανάπτυξη διαδραστικών βίντεο, κάνοντας χρήση σύγχρονων μεθόδων και εξελιγμένων τεχνολογιών. Η εφαρμογή δίνει την δυνατότητα στους παίκτες να αποφασίζουν για την ροή του παιχνιδιού βάσει των δικών τους αποφάσεων και ενεργειών. Επίσης, ο παίκτης μπορεί να χτίσει την ιστορία - βάσει της αλληλουχίας των βίντεο, σύμφωνα με τις επιλογές που θα κάνει σε κάθε στάδιο του παιχνιδιού.

1. Introduction

Video Game Development has been a part of Computer Science for a decades-long, as the first video games were developed back in the 1960's. Nowadays, millions of people around the world are working into the video game development industry such as professional gamers, programmers, testers and designers.

Web Development, on the other hand, is a much newer part of Computer Science as it started from the early 90's but made dramatic progress over the late decades along with the Internet explosion. In the latest years a huge amount of I.T. professionals choose to get involved with Web Development as the large number of websites on the world wide web, web apps, companies' applications etc. create thousands of job opportunities around the world.

For the purpose of this thesis, a website titled "Room of Associations" has been created, which is a real video-based video game hosted on a website as a game platform. In this game/website the player has to solve some quizzes based on clues that he/she finds on real videos in each level, in order to complete the game successfully. In case that the player doesn't succeed in solving these quizzes, having a limit of 10 "lives" in total, it leads to "Game Over".

The purpose of this thesis is to combine the "digital arts" of Game Development & Web Development, along with Visual Content Production, as the website-game contains real videos.

2. A short description of the game

In the web game "Room of Associations" the player -after registration and login- watches a video with a man waking up in a room alone. The man doesn't know where he is at, who he is and why he has been placed inside a locked room. He suddenly read a letter which says that the only way that he can escape is by solving some quizzes that are included in the items of this room! The player (named in the game as his "Guardian Angel") is the only one who can save him by helping him choose the correct answer of each quiz.

The game contains all the fundamental principles of a video game, such as player's account creation, levels, lives, "Save Game" option, session continuation & "Game Over". ROA's (Room of Associations) basic distinctiveness is that it uses real videos instead of the classic gaming computer graphics and quizzes as certain old adventure games did.

Player's purpose is to solve a number of quizzes (currently 7) each quiz in each game's level without losing 10 lives in total (one life is been deducted for each wrong answer). If the player solves all the quizzes in all levels without losing 10 lives in total, then the game's hero will be free out of this room. If the player fails, then the game's hero will "die" or be imprisoned forever. Of course, the user has the ability to save the game anywhere he/she wishes in order to continue from there with his/hers remaining lives.

3. Field Review / Similar Applications

3.1 Introduction

In this chapter we will present, after thorough research, some similar websites or games in Greece and in the rest of the world.

Games, Web Development and Visual Content Production are fields of research and development for several years the result of which are the numerous web games developed and distributed globally nowadays. But, as we mentioned earlier, the basic distinctiveness of ROA is that it uses real videos and quizzes, in order to combine of Game Development & Web Development, along with Visual Content Production. Hence, it is very difficult to find a very similar application anywhere. Nevertheless, there is a plethora of web games out there with similar structure and development architecture.

3.2 Similar Applications in Greece

DOD
https://dod.gr

Dod.gr is an online game and chat platform with more than 20 classic -mostly multiplayer- games, mostly childhood alike, as Scrabble, Word Snake, Digitalized Board Games, Chess, Backgammon, Cards, Quizzes, Painting etc. The site's members can also create accounts & profiles and socializing with each other.

The old version of DOD was built in Flash technology (Adobe Flash Player) but as the new and updated browsers no longer support flash, the modern website is built mainly on JavaScript Libraries.

[...Since Adobe no longer supports Flash Player after 31 December2020 and blocked Flash content from running in Flash Player beginning 12 January2021, Adobe strongly recommends all users immediately uninstall Flash Player to help protect their systems...] (Adobe, 2020)

Image 3.1: dod.gr website

ZOO
https://zoo.gr

Zoo is one of the oldest and most popular web games platforms in Greece. It contains more than 100 games, which are divided in the following categories:

- Multiplayer Games (such as Chess, Backgammon, Cards etc)
- Browser Games (such as Strategy games, Action games etc)
- Singleplayer Games (such as BoxKid, JellyBomb etc)

The old version of ZOO was also built in Flash technology, but as the new and updated browsers no longer support flash, the modern website uses mostly JavaScript Libraries and HTML5 based games. The majority of Browser Games are external affiliate links.

Zoo.gr has also a chat platform and the members can also create accounts & profiles and socializing with each other.

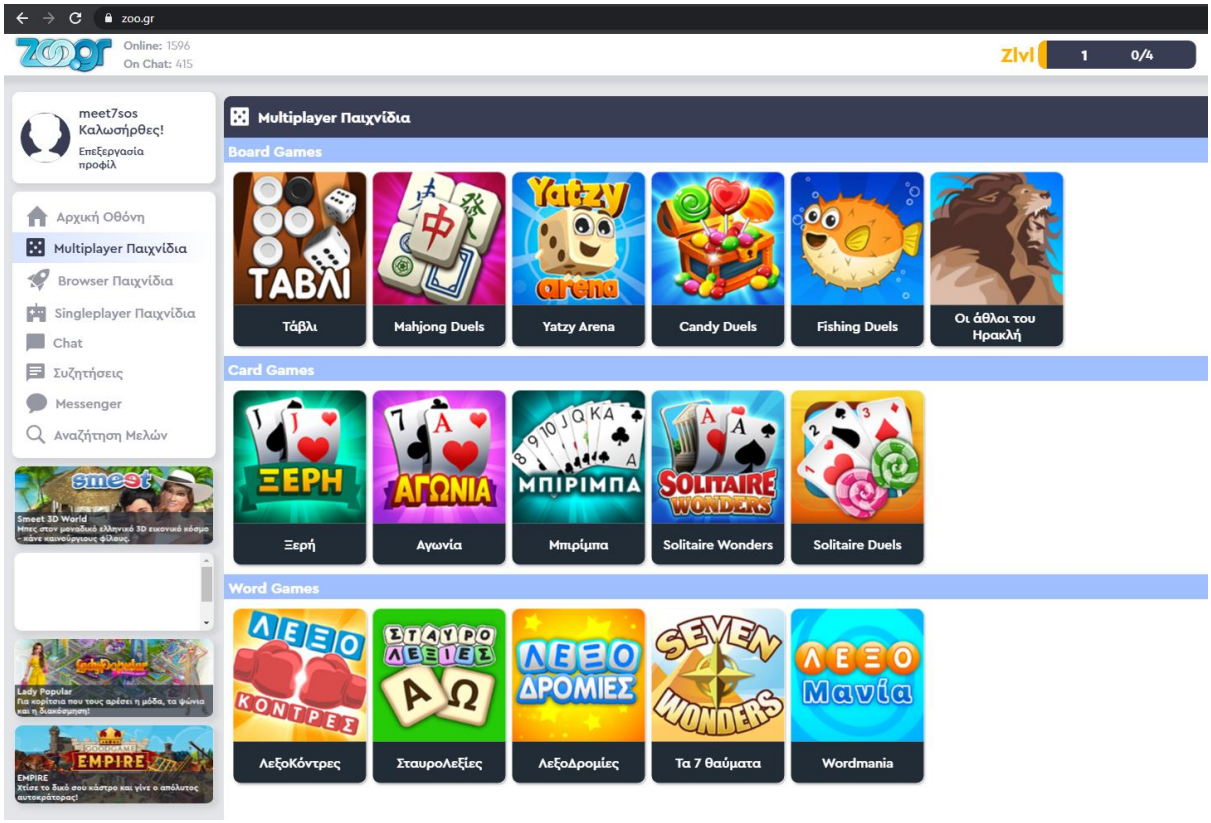

Image 3.2: zoo.gr website

MYKOSMOS
https://mykosmos.gr

MyKosmos.Gr is not a gaming platform, as it has a log of fields of interest – such as news, astrology, weather etc, but one of the site's categories is games. Most of these games are Quizzes and Knowledge Games, such as Geography in Greece, Greek History Quizzes, Map of Greece puzzles etc.

Questions in the geography quizzes are followed by interactive maps, which consists visual content. Though the games are simple quizzes, and not complete games with "save" options, account creations etc.

The whole website is built in PHP/MySQL, HTML, JavaScript.

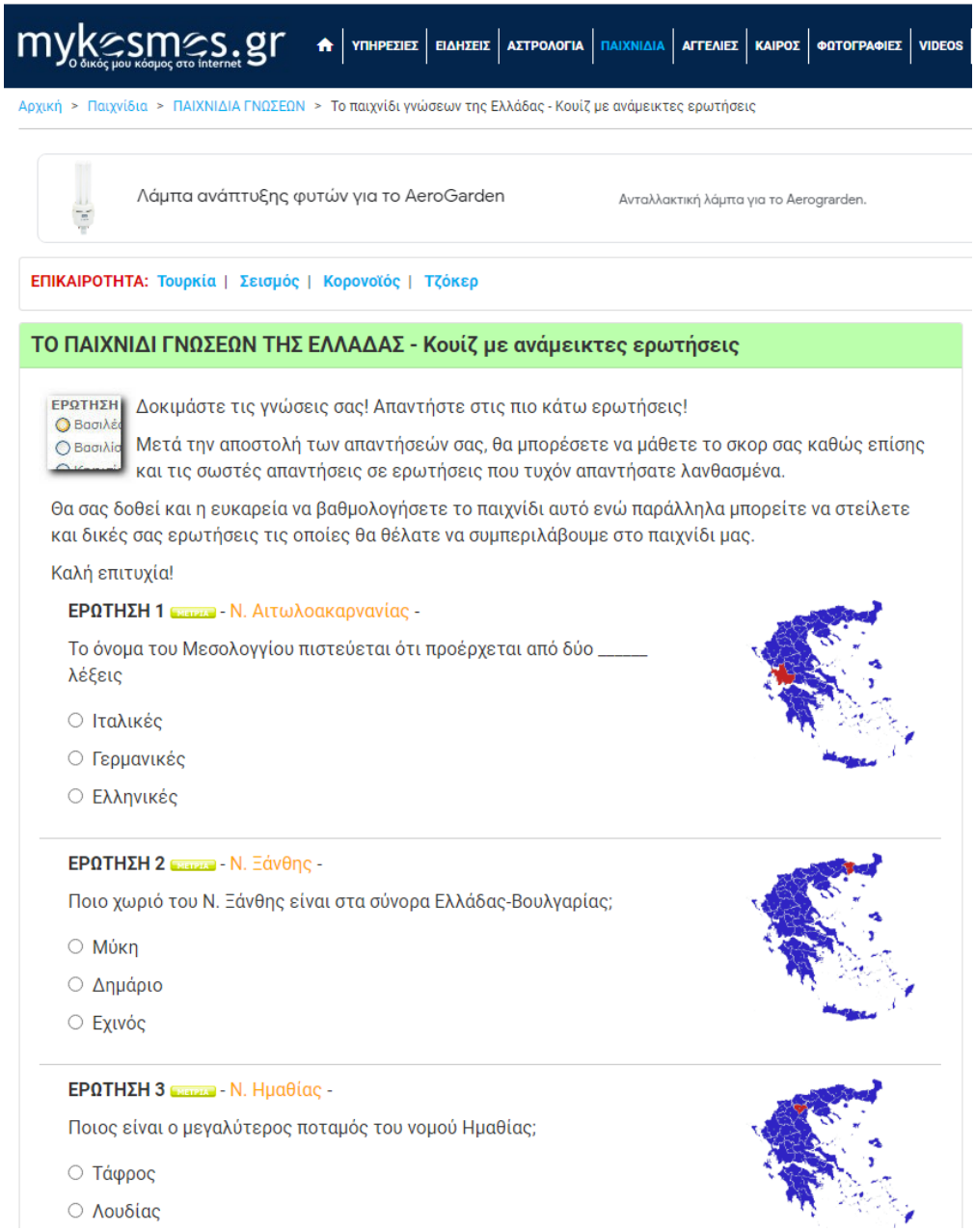

Image 3.3: mykosmos.gr website

123 PLAY GAMES

https://123playgames.gr

123PlayGames.Gr is a Greek online games platform which hosts more than 100 games and belongs to Ant1 Group, which is a media company in Greece and currently the largest Greek media company.

In the website we can find Role Playing, Strategy, Simulation & sports games. Most of these games are hosted in external platforms and the player redirects there through 123PlayGames referral affiliate links.

123PlayGames website also hosts the “Next Player” video blog, which is a Vlog for gamers and computer geeks.

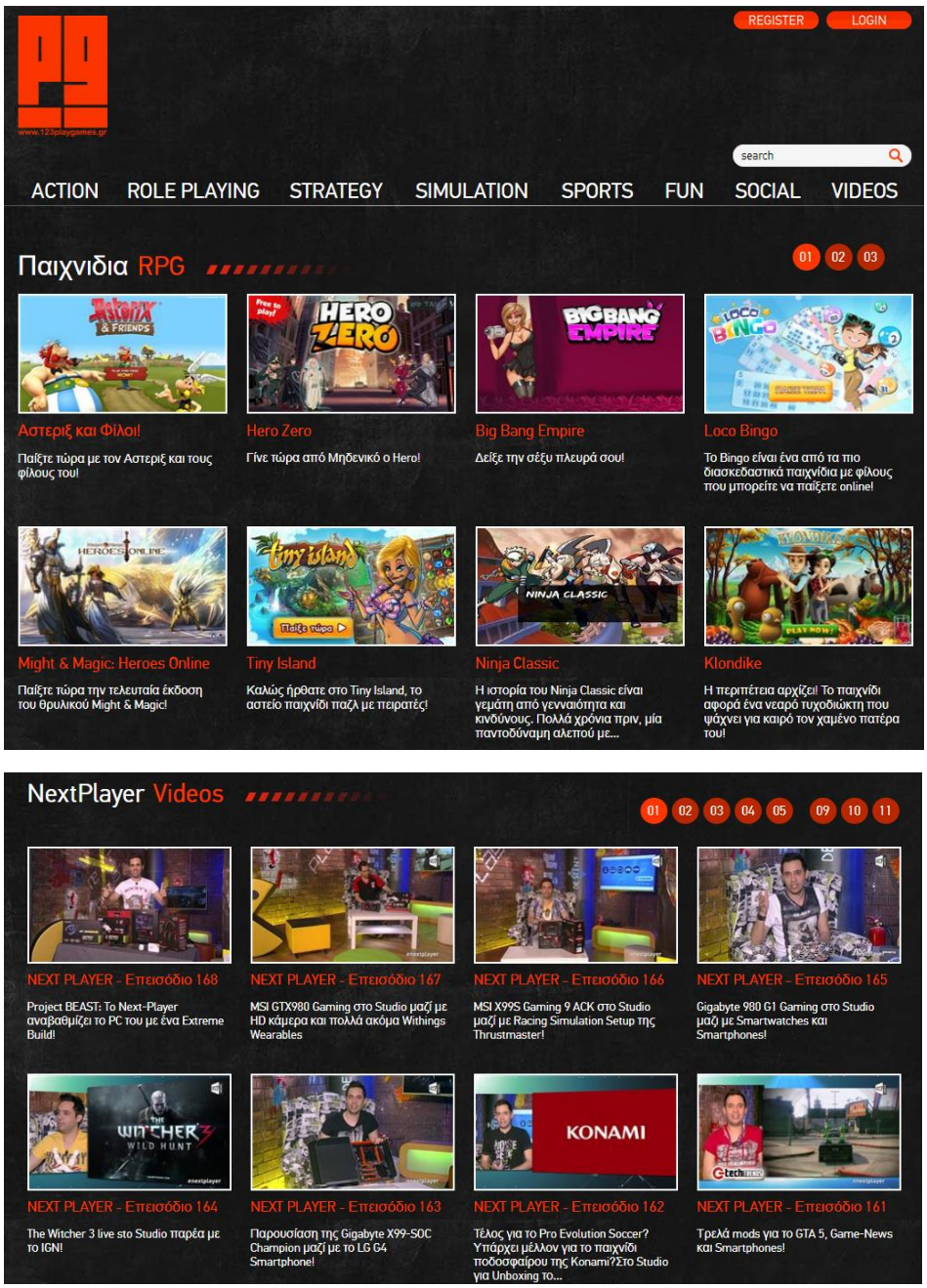

Image 3.4: 123playgames.gr website

3.3 Similar Applications Worldwide

NODIATIS
https://nd1.nodiatis.com/

Nodiatis is an online multiplayer RPG (Role Playing Game) browser game that started back in 2008. Nowadays, Nodiatis has been ported to HTML5. The result is a client that runs on virtually everything: PCs, Macs, cell phones, tablets, any touchscreen device.

In this browser based MMO (multiplayer online game) the players compete in a massive online virtual world, competing other players on a fame and wealth acquisition arena for fame and riches. The Nodiatis skill system is incredibly in-depth and allows player to build a unique character exactly how he/she likes.

The game was developed and published by Glitchless, LLC and was coded in Java programming language.

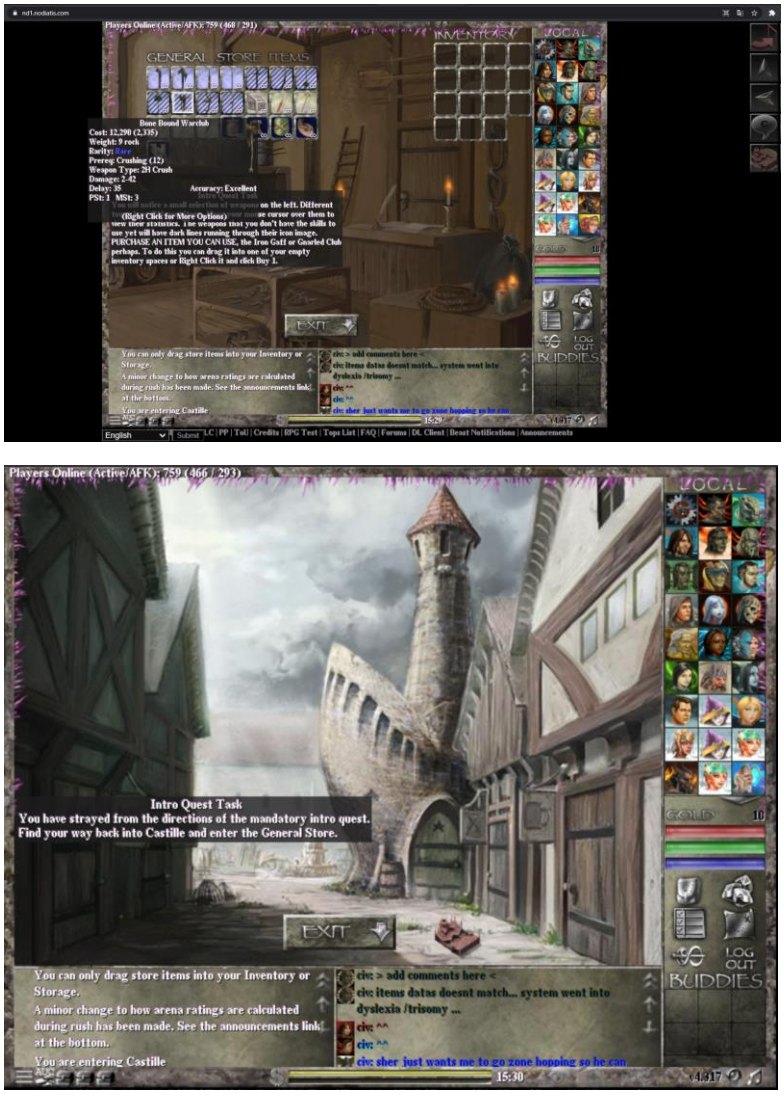

Image 3.5: Nodiatis

Fallen London
https://fallenlondon.com

Fallen London is a browser-based interactive narrative game developed by Failbetter Games and set in an alternative Victorian London with gothic overtones. The game has been running continuously since 2009 and in June 2018 the website received a major graphical update, with a page redesign as well as better scaling across devices and HTTPS integration.

A particularity of this game is than it can be lost, but can't be won! As we can read in the game's info:

[... "A questline to "Seek Mr. Eaten's Name", about destructive obsession, requires the player to damage their character in like manner repeatedly, until its completion leaves the character permanently unplayable. The game requires players to opt into this questline and warns them against playing it."

"Players take the role of new arrivals to the underground down on their luck, and make their way to the cream of the crop of the city's various legal and illegal activities. Players are gentlebeings of leisure, plumbing the vices and secrets of Fallen London. They have no living expenses, and though players may choose a profession for a periodic income, they can publish a newspaper, serve out repeated prison sentences and feed deliverymen to a man-eating plant without harming their job security." ...] (Fallen London, 2009)

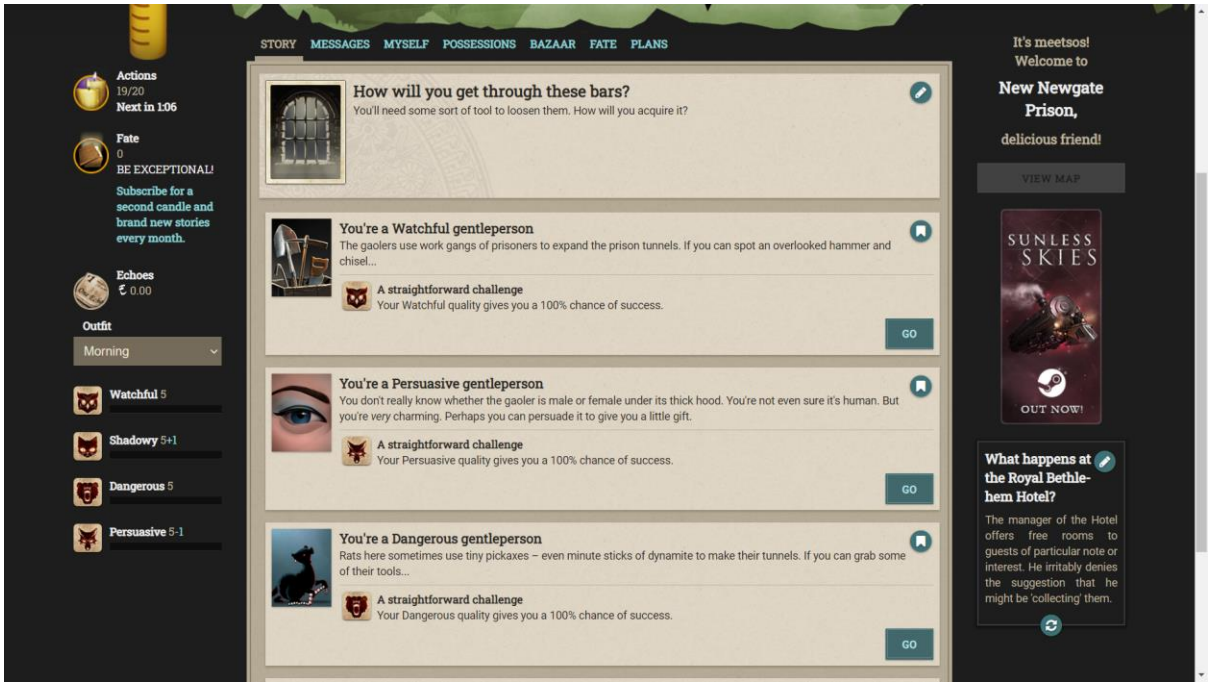

Image 3.6: Fallen London

Kingdom of Loathing
https://kingdomofloathing.com

Kingdom of Loathing (is a browser-based multiplayer role-playing game designed and operated by Asymmetric Publications which was released in 2003.

The interesting and fun part of this game is that it uses hand-drawn stick figure graphics and writing characterized by surreal humor, wordplays, parody and references to mass culture.

In this game a player's character fights monsters to increase the experience of his/her virtual character and search and collect items in order to increase his/her skills and endurance, through a turn-based system. Players can also interact with each other by trading goods and services, organize their characters into clans and speak to each other in many different chat channels.

The game is been financed from donations and the purchase of merchandise. It is now using advertising or subscription fees, as most of other online games do. KoL has an active community than organizes fan meet-ups and also runs an internet radio station (in 2008 the game had between 100,000 and 150,000 regular players, but we don't have an official number as for today).

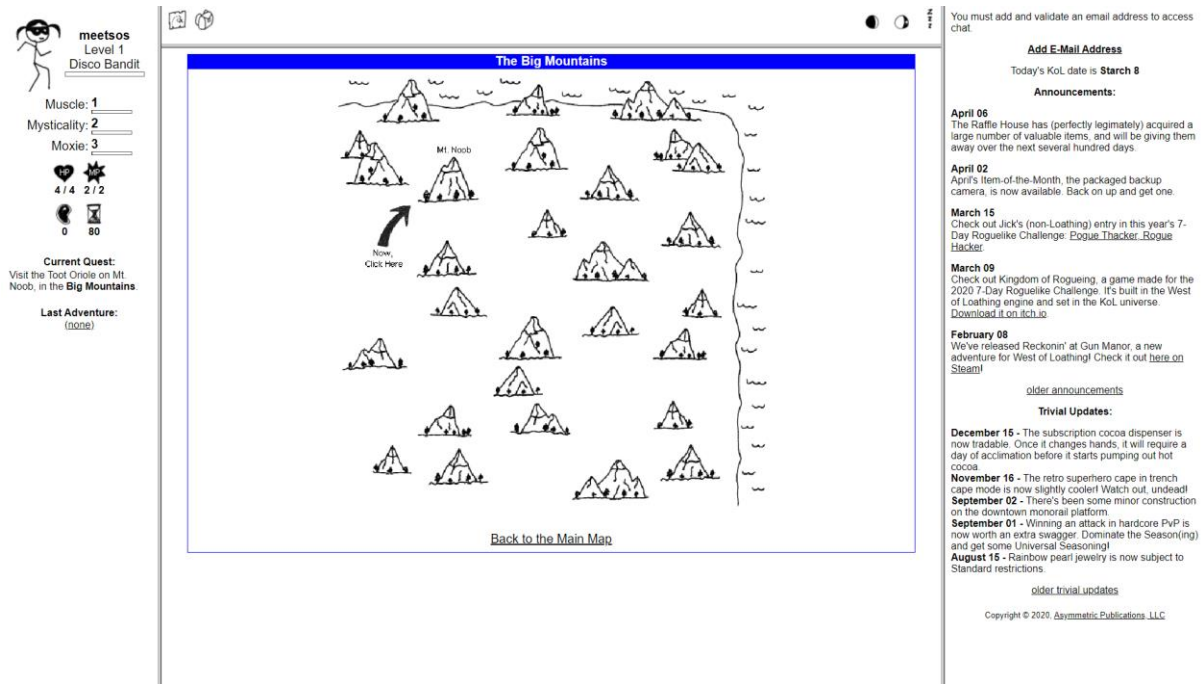

Image 3.7: Kingdom of Loathing

Depression Quest
http://www.depressionquest.com/

Depression Quest is an interactive fiction game where the player can play as a person that lives with depression. The player has to manage everyday life's matters (such as job, studies, relationships etc.) and he/she can experience through the game how difficult are the small everyday day matters for a person with depression. Depression Quest, can show us this way, other sufferers of depression that they are not alone in their feelings. It aims, also, to illustrate to people who may not understand the illness, the impact that depression can have to one's psychology.

The initial purpose of this game was to spread awareness, so a new player doesn't have to pay a minimum amount in order to play the game. However, the visitors have the choice to pay in order to donate the developers and the rest of the team and, also, a portion of the proceeds will be donated to National Suicide Prevention Hotline.

The game is fully developed in HTML & JavaScript by Zone Quinn, Patrick Lindsey and Isaac Schankler and it is based on quizzes (the choices that players does when he/she navigates in everyday life events while managing illness, personal relationships, and job).

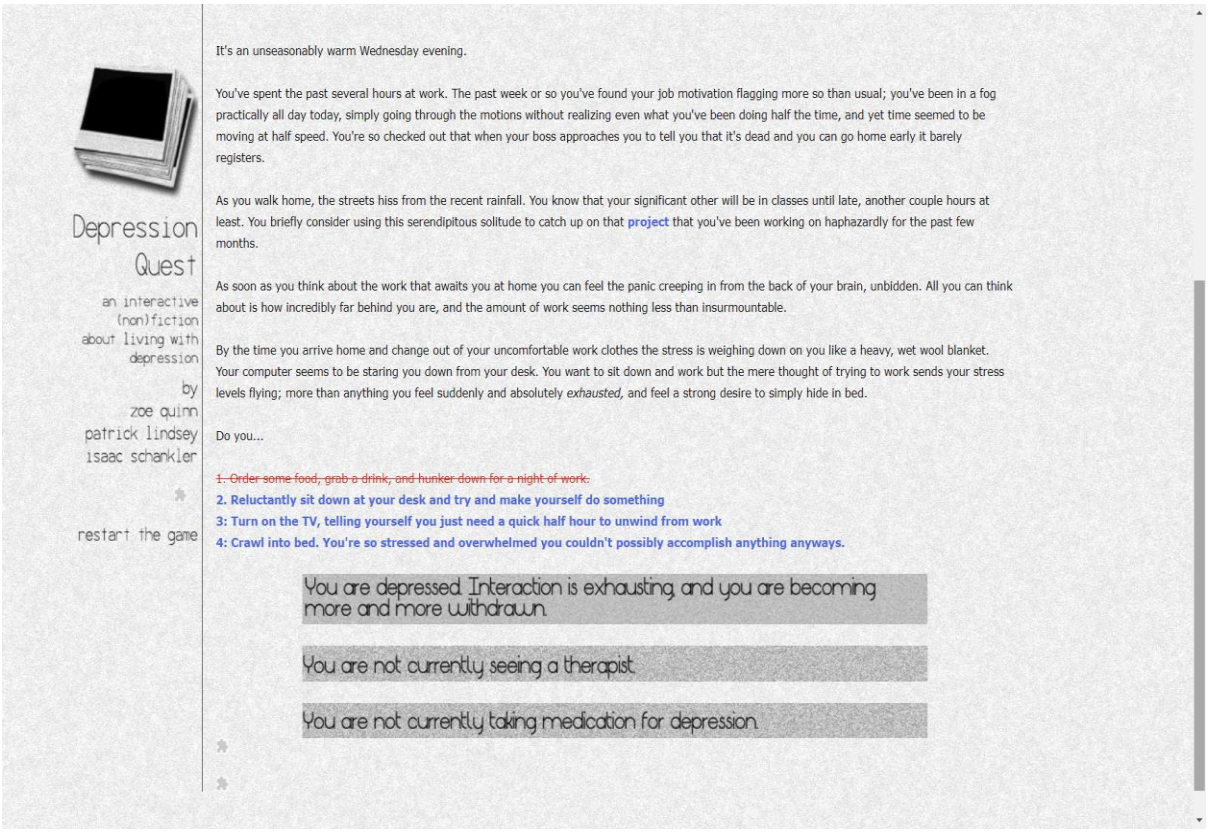

Image 3.8: Depression Quest

A Dark Room

https://adarkroom.doublespeakgames.com/

"A Dark Room" is an open-source text-based adventure/role-playing single-player game that was initially published for web browsers by Doublespeak Games on 2013. The game begins with the player awakening in a cold, dark room after a mysterious event, where he/she can only light and tend a fire in the room ("the fire is dead. the room is cold. awake. head throbbing. vision blurry."). As the game continues, the player is able to collect resources, talk with people, create villages, and explore the world.

"A Dark Room" was created by Michael Townsend, the founder of this Canadian indie studio. It was released as a browser game on June 2013. In July 2013, Townsend released the source code of the game on GitHub under the open-source license MPL 2.0. Later, the game became also available for iOS, Android etc. The browser game code, is basically based on JavaScript and it is available here: https://github.com/doublespeakgames/adarkroom

What makes this game unique is, as The New Yorker mentions, *[... "the game evokes the simplest text-based computer games of the nineteen-seventies while stimulating a very modern impulse to constantly check and recheck one's phone. It's like a puzzle composed of deconstructed to-do lists" ...]* *(New Yorker, 2014)* as the player does everything by entering text & mouse clicks.

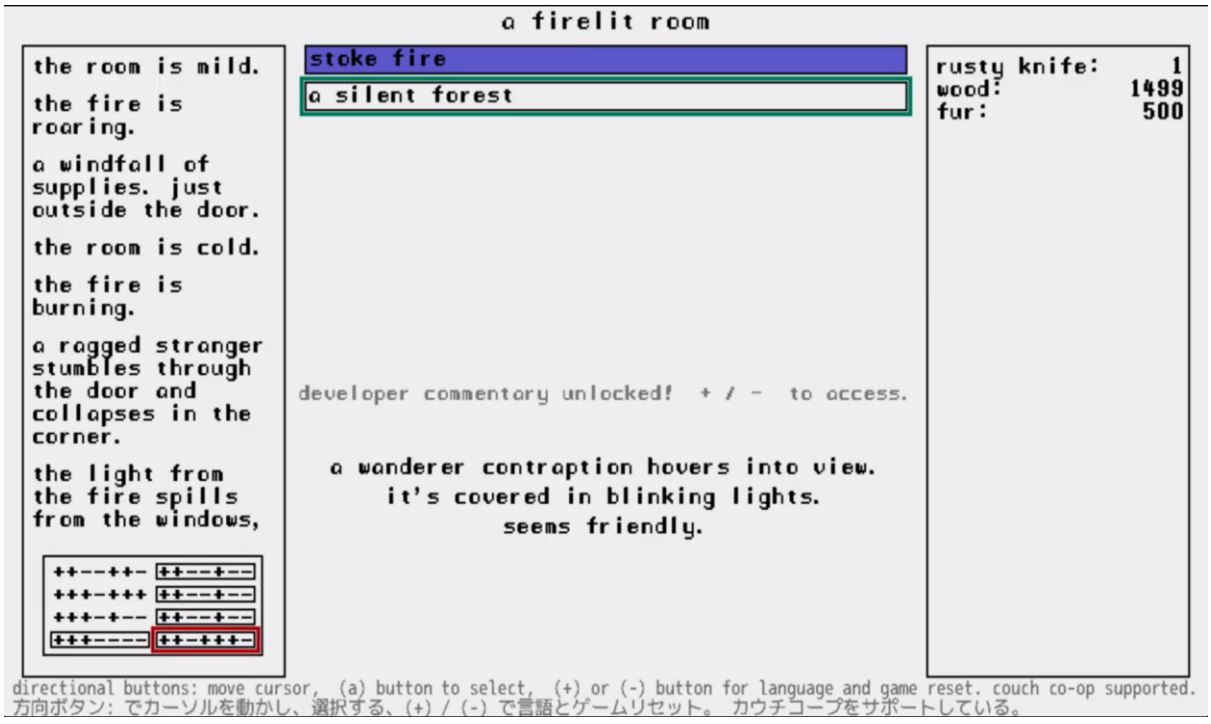

Image 3.9: A Dark Room

City Guesser

https://virtualvacation.us/guess

City Guesser is a browser-based geography game that launched on August 2020. It was created by Paul McBurney Jr. and was inspired by GeoGuessr, with the exception that is real-video based, it has been (and will be) always free and there is no need of an account in order to play. It is just "Click & Play".

When the player joins a game (from a big selection of maps - countries - cities than he/she can select) he/she see an HD video of a city or place somewhere the world in which a camera holder is walking for a couple of minutes. Player's purpose is to collect clues about this place and guess where it is. When the player feels ready to answer, he/she submits the "Guess" button and a pop-up map appears where the player can set a pin about the place, he/she guessed. Then, the system says how close the player's guessing was.

City Guesser exist in both single-player and multi-player versions where the player can create private server and invite friends. Users can submit their own video with a simple way:

[... "post it to YouTube with the title "City Guesser Video:" and the city being showcased. Example: City Guesser Video: Barcelona, Spain. Make sure videos are at least 5 minutes long and filmed at a desktop aspect ratio."...] (Official City Guesser Guidelines) (Virtual Vacation, 2020)

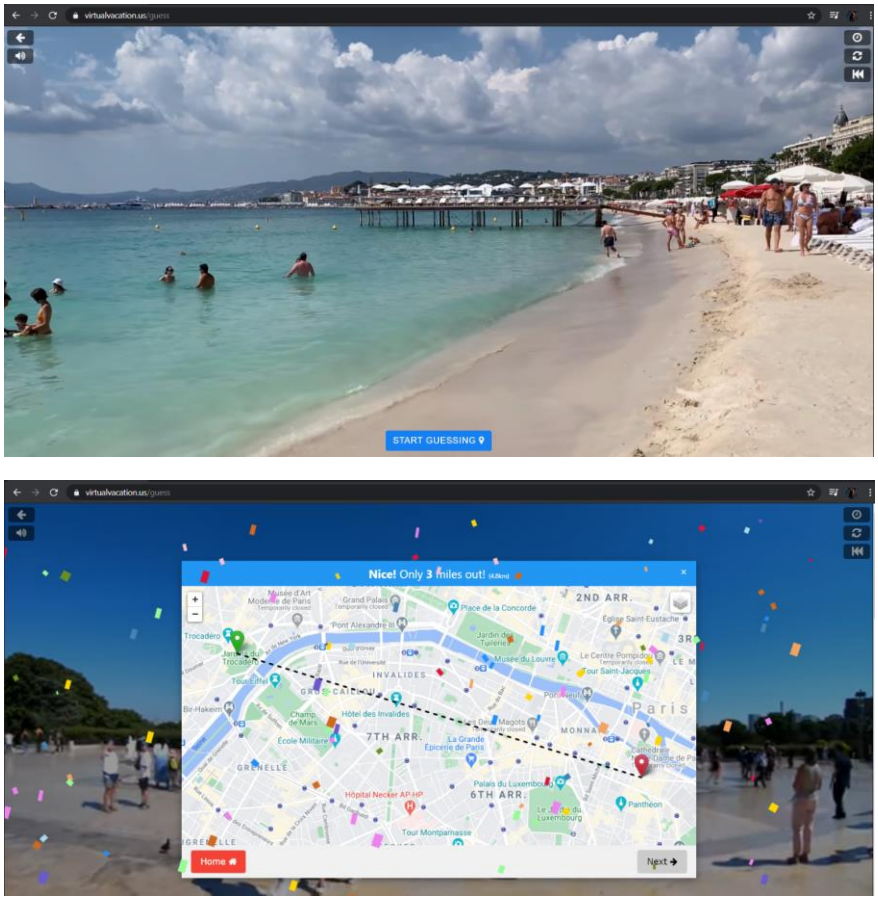

Image 3.10: City Guesser

4. Theory and History behind the Application

Adventure Games are the video games in which the player assumes the role of the game's main hero and usually the player proceeds in the game through riddles and puzzle-solving type quizzes. Most of these games are based on a story -mostly an interactive one- and as the game continues, the story is proceeding. Most of the adventure games are designed for single-player mode, because creating many characters in the same story, is much more complicated in comparison to a single character story. *(Dillon, Teresa, 2005)*

First adventure games were created back in the 70s and were text-based. "Text-Based" means that the player needs to import text or commands from the keyboard, in order to proceed. For example, player needs to type "YES" or "NO", instead from choosing one of the available options from a radio button. Later, with the development of new more powerful computer systems and the evolution of graphic cards, we had the graphic adventure-game format in which the player is able to use point-and-click interfaces also. *(Ammanabrolu, Prithviraj, 2018)*

The first adventure game was Colossal Cave Adventure, that was released in 1976. Other notable classic adventure games are Rogue (1980), Mystery House (1980), Myst (1993), The Secret of Monkey Island (1990), King's Quest (1980) and Leisure Suit Larry in the Land of the Lounge Lizards (1987).

The Adventure Gamed Subgenres are:

- Text adventures and interactive fiction
- Graphic adventure
    - Point-and-click adventure games
    - Escape the room games
    - Puzzle adventure games
    - Narrative adventure games
    - Walking simulators
- Visual novel
- Interactive movie
- Hybrids

4.1 Text Adventures

Text adventure games are defined as the games in which the participant uses a text-based user interface, in which the player can use from a set of encodable characters such as ASCII instead graphics. The player submits simple commands to interact with the game. Adventure Games owes their name to "Colossal Cave Adventure" which is considered to be the first adventure game. Early text adventures used a simple verb-noun parser to implement command instructions.

Later text-based adventure games used more natural language in order to enable more complex player commands, in some cases a full phrase. With the advance of graphic game, the text adventure fell apart. Text based adventures are still popular, but most of the new games are just free games on the internet.

4.2 Graphic Adventures

Graphic adventures use graphics in order to deliver the environment to the player. Some graphic adventure games utilize a first-person perspective, while others are in a third-person perspective in which a camera is focusing on the player's moves.

Graphic Adventures are separated in point-and-click adventure games (where the player typically controls the main-hero through a point-and-click interface, that is a mouse, or in some cases a touch-screen), escape room games (where the player is trying to figure out how to escape a usually small room using the resources within the room and via solving of puzzles & quizzes), puzzle adventure games (that emphasizing on puzzles and each puzzle opens more of the game to proceed and usually more puzzles & quizzes to solve), narrative adventure games (in which choices made by the player are influencing incidents throughout the game's story), walking simulators (that allows the player to experience his/her story in an "explore and discover" situation and also allows the player to wonder around the game's rooms & places in order discover items, clues etc.).

4.3 Visual Novels

Visual Novel Adventure Games are a form of interactive fiction that consist a hybrid form of text-based and graphical games. The main reason that visual novels are different from other games is that the gameplay is mainly based on clicking to keep the text, graphics and sound moving, while player is taking decisions along the way. Moreover, in a lot of visual novels there are multiple storylines and alternative endings. Many visual novels use real voices for the game's characters and most of them feature dialogue-trees for the characters conversations.

Visual novels originated in Japan, where they made up the majority of video games until today, and they weren't that popular in the Western world, until the late 2000s. The first visual novel was Portopia Serial Murder Case, back in 1983, in which the player has to resolve a murder by searching for clues, exploring crime scenes, interacting with other game's characters, and solving puzzles. *(Salazar, Francisco Lepe, Tatsuo Nakajima, Todorka Alexandrova, 2013)*

4.4 Interactive Movies

Although the Interactive Film is not a subgenre of Adventure Games, there are a lot of adventure games that considered to be interactive movies. These type of adventure games usually have full video with real actors on a studio set, or either animated. In Interactive Movies we usually have a scene, where players can interact by keyboard, mouse, joystick etc. and each player's choice cause the game to play a new scene.

[... In the early 1990s, following on the heels of laserdiscs, CD-ROMs were able to store digitized video, and thence became the standard, most widely distributed support for computer data. The interactive movie, which flourished in this technolog ical environment, is not easy to categorize. The first popular game named as such, The 7th Guest (1992), is more a puzzle game with few live-action cut-scenes. ...] (Perron, Bernard, 2013)

The invention of laserdiscs was the occasion that Interactive Movies came out, as it was the first nonlinear video media type. Laserdisc technology allowed the player to jump in any game's chapter instantly, so that "create your own story" game design could be possible. Although interactive films usually follow a main storyline, in most of the occasions, alternative scenes are filmed or animated in order to trigger wrong player decisions and "Game Over" scenes.

4.5 Hybrid Adventures

Hybrids are the adventure games that combine two or more of the above four classifications. For example, Action-Adventure games is a species of Hybrid Adventure Games, as it combines basic elements from both the action game and adventure game genres.

Action-Adventures usually combine reflexes & other gaming skills (gun shooting accuracy, speed etc.) with puzzles, problem-solving and other adventure basic elements, and also have a storyline, dialogues and other typical adventure features. Though this is a hybrid game type, it also has its subgenres, as 1st-person action-adventure, 3rd action-adventure, Survival Games, Stealth Games etc.

5. Game Design

5.1 Puzzle-Solving

In puzzle-solving games, the player is usually called to solve some puzzle in order to unlock access to restricted areas, proceed to the game and reveal parts of the game story. Those puzzles could be simple or hard quizzes, knowledge questions, decoding of encrypted messages, open secret or locked doors, find weapons or items and use them etc. Main types of puzzles are the following:

- Inventory puzzles (player should combine items that he/she collected)
- Environmental puzzles (are those that force the player to interact with the environment and it's elements)
- Dialogue based puzzles (in which the player interacts with non-player characters)

[... Language Games proved to be a successful tool in Language acquisition. Tenses and other grammatical items can be taught more effectively. Students actively take part when Language is taught through such innovative tools ...] *(Avinash, 2016)*

5.2 Gathering & Using items

In several adventure games, the player can pick up items (weapons, food, keys, notes etc.) during the game procedure. These items play a key factor in the plot of the game, as they allow the user to unlock new levels of the story, so the game is usually leading the player to the collection of these tools and the player usually can go back in previous scenes in order to collect something that he/she missed. Plenty of graphic adventure puzzle video games back in the 80s & 90s required from the player to find objects that were very small in the size - just a few pixels - and often hidden in the scenery. This is known as "Pixel Hunt" and is one of the most common artificial difficulties in graphical adventure games.

5.3 Story, Setting & Themes

Adventure Games are often driven by a story and are usually set in a fantasy - imaginary environment or even in a completely different world or dimension in each game's level. Many types of these mysterious background stories use the "Problem of Amnesia" (Ernest W. Adams) where the game's main character starts the game without the player having any clue about his past, experience, history, knowledge etc. So, usually, the player's main target is to reveal all this background mystery and help the main hero to complete the story line.

Also, with the use of VR (Virtual Reality) many adventure stories are set in an Immersive Environment which is a perception of being physically present in a non-physical world. The conception of this immersive world is created by surrounding the user with images & sounds that provide a such feel-like environment. *(Cruz-Neira, Carolina, Fernández, Cristina Portalés, 2018)*

6. Game Manual

6.1 Introduction

In this project we emphasized in the simplicity of user's interaction. The player is supposed to be an outside watcher who is helping the video's hero to choose the right answer for each quiz - level. So, the game's main purpose is to focus on player's research as a lot of time and web search is needed in order to find the correct answer, as it combines elements of the real world (music, books etc). The only thing that a new player has to do in order to start the game, is to register a new account.

6.2 User Registration

Every player needs to register, in order to play the game. The registration is mandatory, as without an account there wouldn't be "Save Game" option, Score, Game Sessions etc. User Registration requires a Username and a Password. No e-mail needed so far, as the game was developed in a local environment (personal computer's localhost), but it will a needed feature in case that the game will be someday published in the World Wide Web.

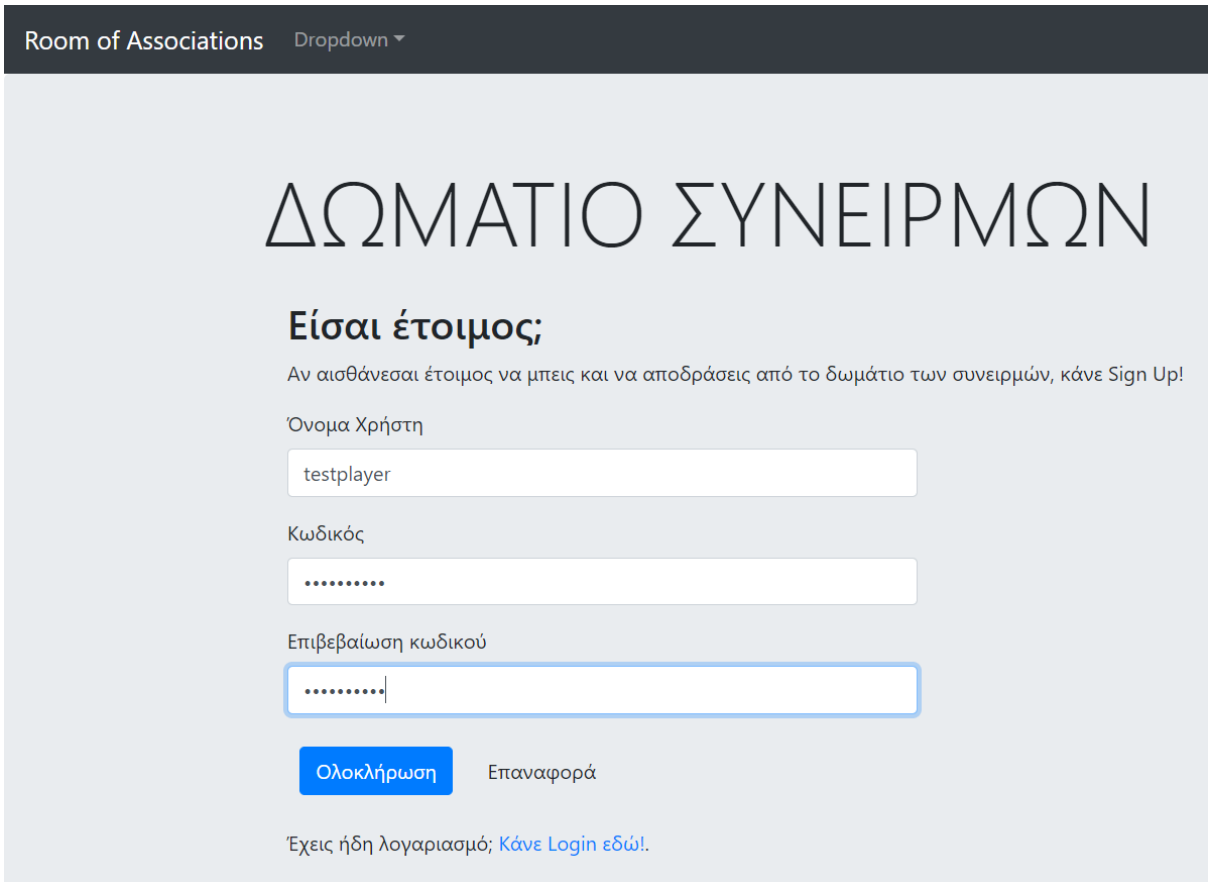

Image 6.1: Register screen

If the player has already an account and just wants to continue game, he/she can login directly.

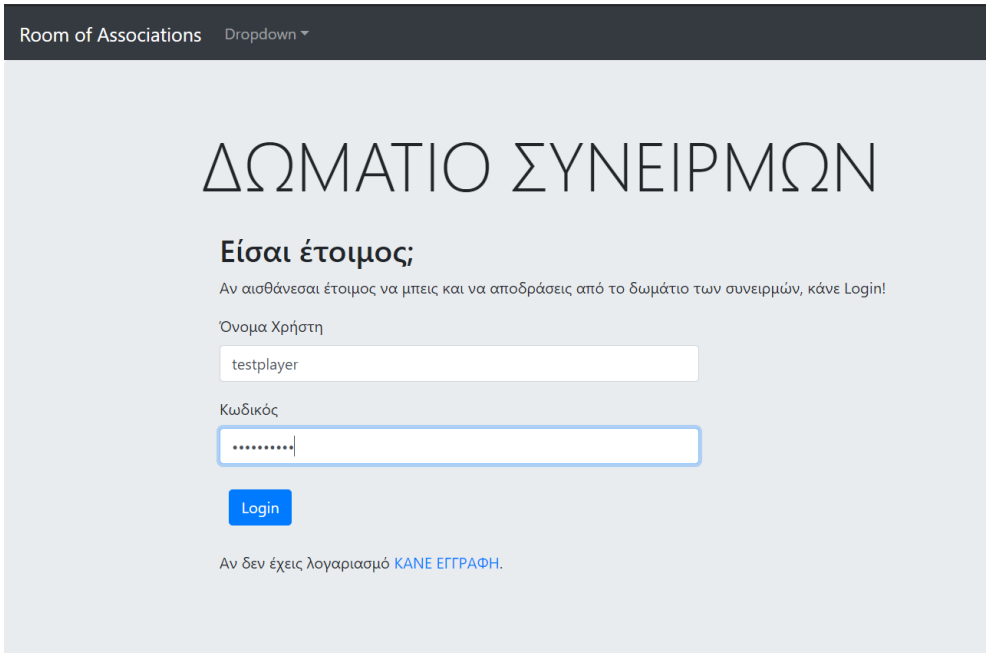

Image 6.2: Login screen

6.3 Game's Intro

As the game begins, after user's registration and login, there is an intro scene where the player can see story and the game rules. It provides the player all the basic information (lives, rules etc.) in order to proceed to the main game and start the adventure.

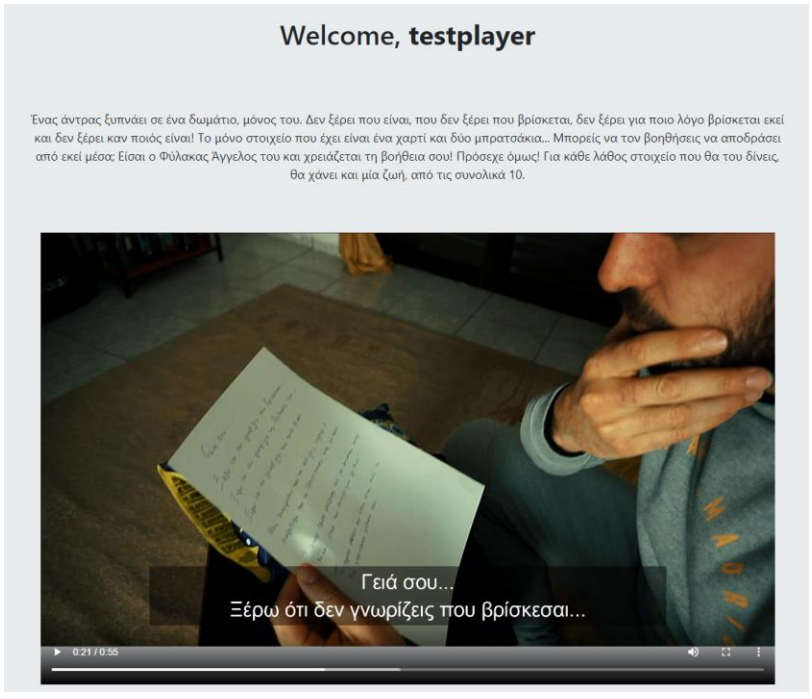

Image 6.3: Welcome video

6.4 Game Levels

After the introduction video, game starts from Level 1 (named: The Photos). Each level, as Level 1, has a similar philosophy which is:

- Watch the video
- Do your research
- Collect clues
- Answer the quiz (help the hero)

For example, in Level 1, the hero is watching some photos on the wall and has to choose the right one based on the only clue provided at him/her at that stage, in order to find another clue to unlock the next level.

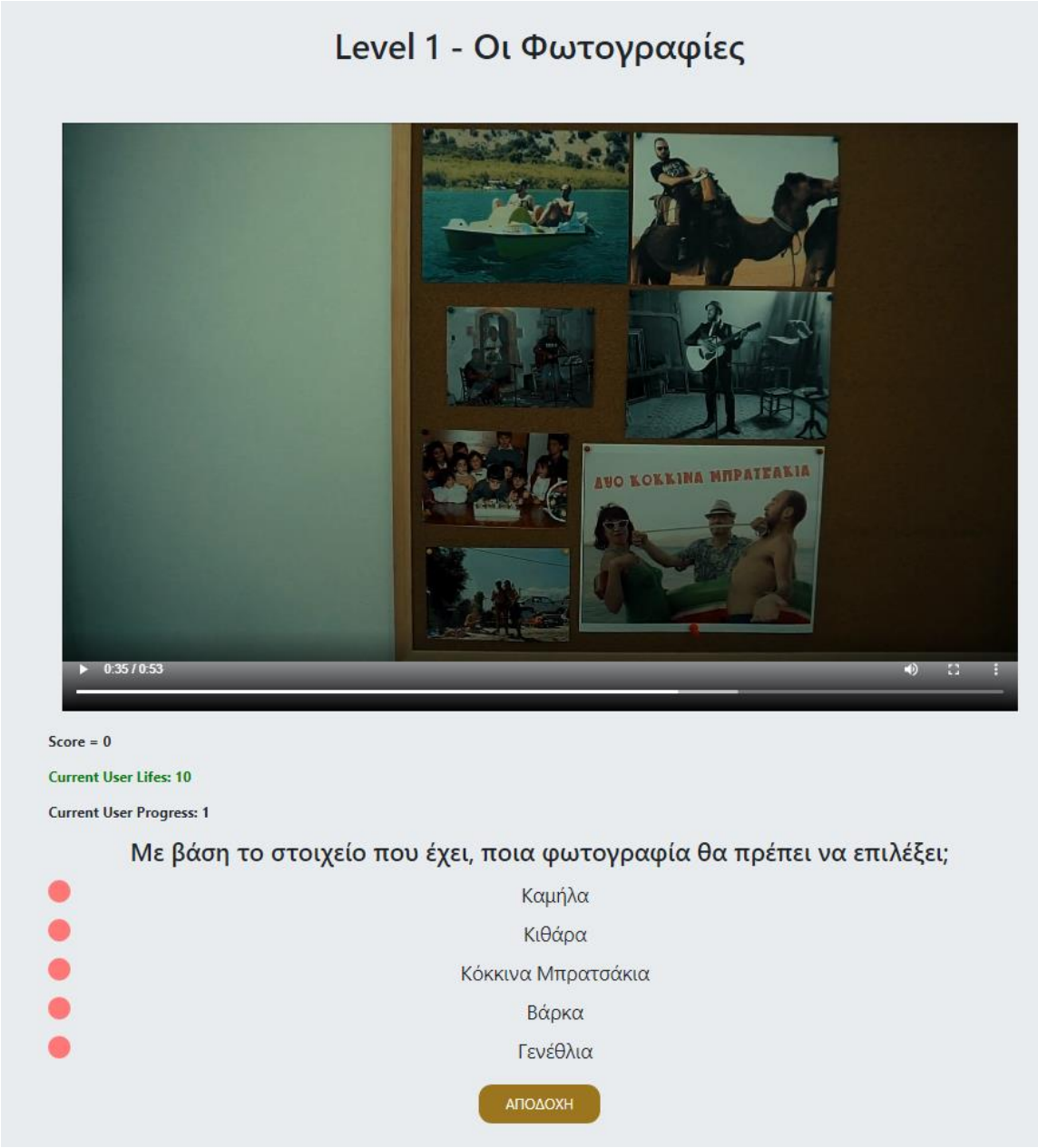

Image 6.4: Level 1

6.5 Lives and Score

In the case that the player chooses the wrong answer, he/she loses one life and 2 points. In the below example, the new player – that started with 0 points and 10 lives, in his/her first try chooses the wrong answer.

The system informs the player about the wrong decision and lives/points loss. As the player continues in the same level after the wrong answer, the previous chosen answer is been vanished in order to avoid unwanted confusion to the player by choosing the same wrong answer the player from choosing the same wrong answer again and lose more lives & points for no reason.

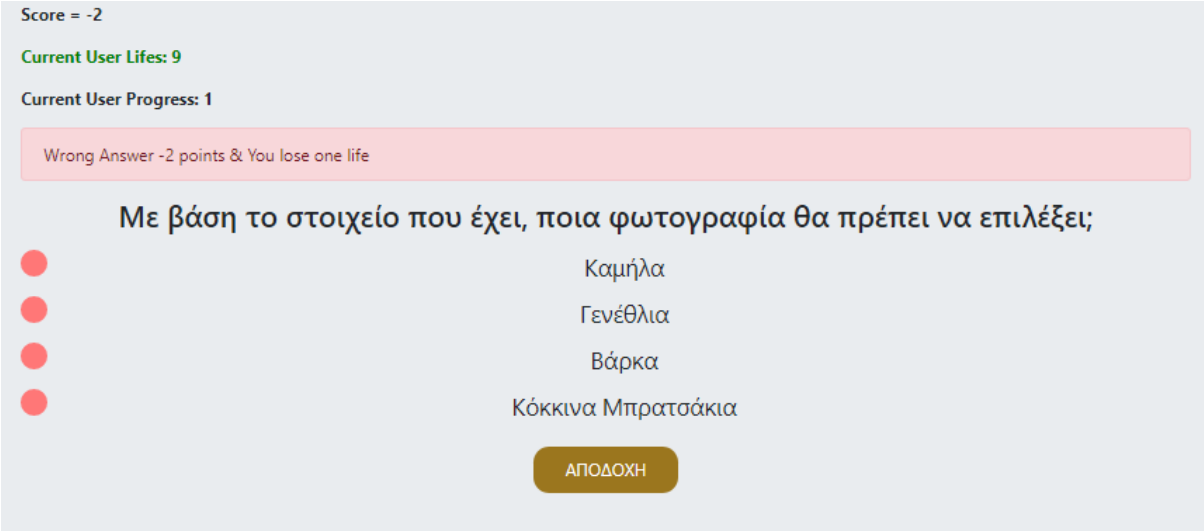

Image 6.5: Lives & Score

In the case that the user has chosen the correct answer, the player gains 10 points and maintain his/her lives. The system informs the player about the current score and the remaining lives. User can now proceed to next level by clicking on "PROCEED TO THE NEXT LEVEL" or wait 8 seconds for the system to redirects automatically to the next level.

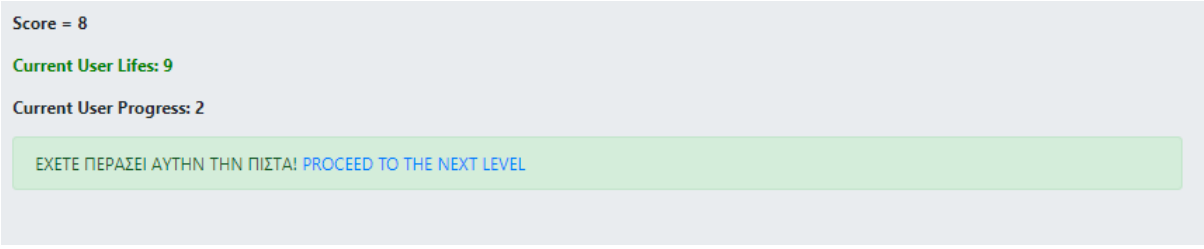

Image 6.6: Proceed to the next level

6.6 Save Game & Sessions

The player can stop playing the game anytime. If the player stops the game by accident (power loss or anything) the next time that he/she logs in again he/she will continue from the level that the game was left with the remaining lives and achieved score. This is an auto-preserved method, the so called “auto session”.

For example (see below image) this player succeeded in Level1 and proceeded to Level2 when he/she exited the game by just closing the browser. In his/her next log in he/she sees a screen that tells him/her that he/she can continue from the spot that it was left (Level 2)

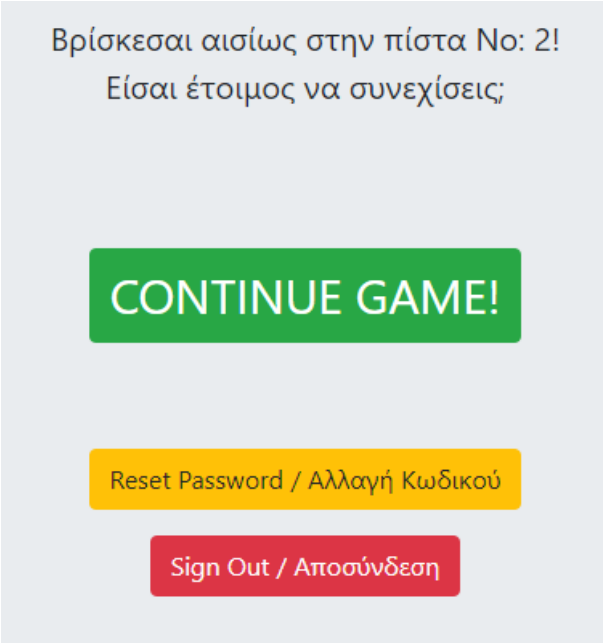

Image 6.7: Continue game

Meanwhile the player has the option to manually save the game, by clicking “SAVE GAME” on the upper right corner, at any time. This option gives the player the opportunity to control the game and manage it as he/she wants, as a part of a strategy in order to complete the whole game successfully.

When the player saves the game, the page reloads and informs the player that the game is successfully saved. Of course, in this case also, the player will continue playing the game with the remaining lives and achieved score.

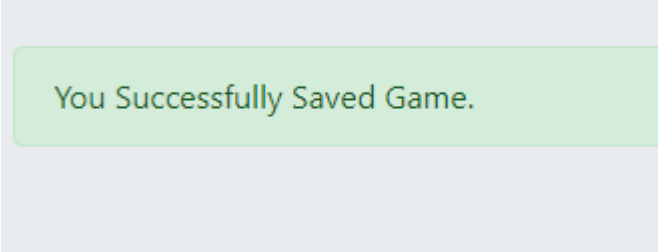

Image 6.8: Game save

When the player logs in again, next time, he/she can see the option on the opening screen that he/she can continue the saved game from the spot that he/she saved it (in the below example is Level3)

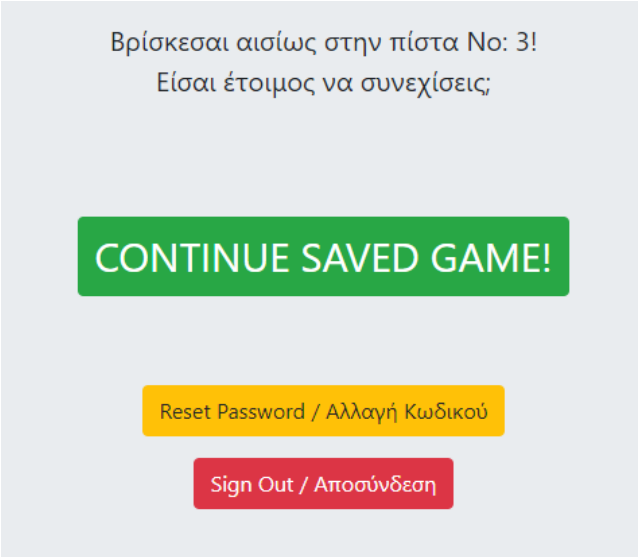

Image 6.9: Continue saved game

In the case that the player saved the game manually, but also logged out in another game's spot without saving the game, the system gives the player the choice to continue the game either from the saved point or from the "auto session" point. These options gives the player more scalability in game control and prevents the case of messing out the game by accident, such as accidentally browser closing or logging out.

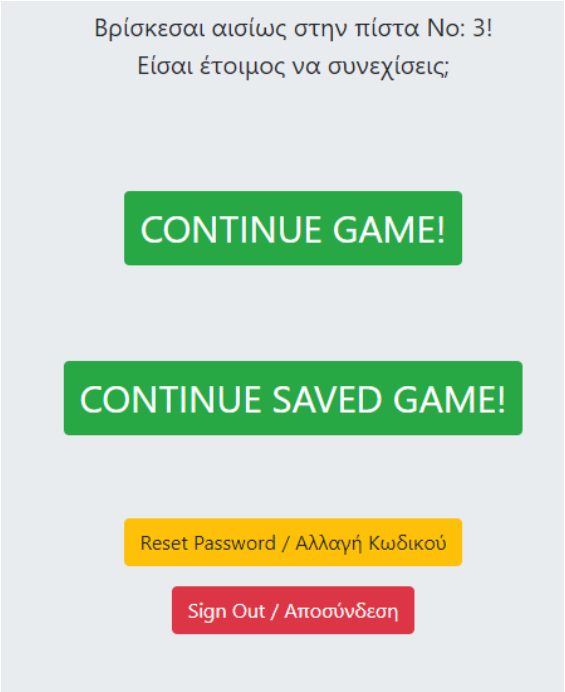

Image 6.10: Continue saved session

6.7 Player's preferences

The player can change his/her password at any time by clicking the "Reset Password" button. The username thought cannot be changed, as it a unique entry in the database. No other information needed from the user after account creation, such as real name, age etc. so no other options given.

Player can view his/her score and remaining lives at any time and can also log out from the game at any time.

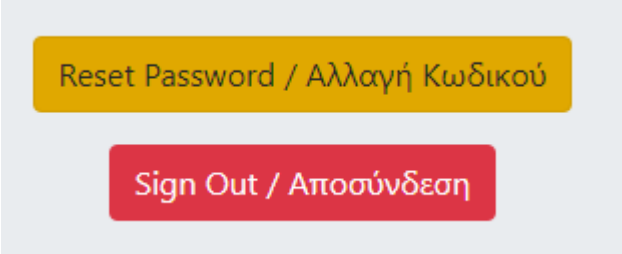

Image 6.11: Player's preferences

In case of Password Change, player needs to insert current existing password, as well as new password twice, for security reasons.

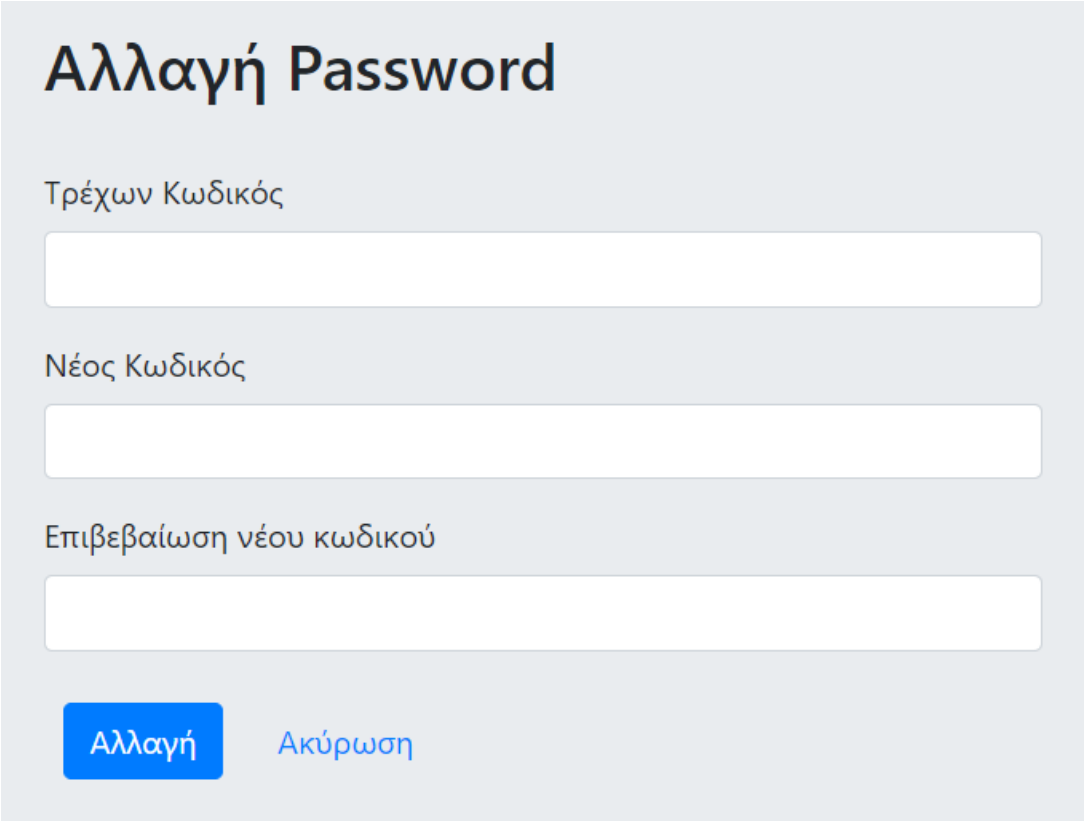

Image 6.12: Password change

6.8 Game Over

In the case that the player lost all his/her initially given lives (that means 10 total wrong answers) the game is over for him/her. He/she can though continue from a saved point if he/she saved the game before.

The other option that the system gives to the player, is to start the game over, from Level 1. That gives the player the choice of strategy: In case he/she remembers the correct answers from the start until the "Game Over" point, he/she can be able to continue (with a little loss of time, of course) the game from the last point with more potential lives and bigger score.

In any case, the system informs the player the end of the game ("game over") and the total score he/she achieved during the game. The score will remain forever in the database.

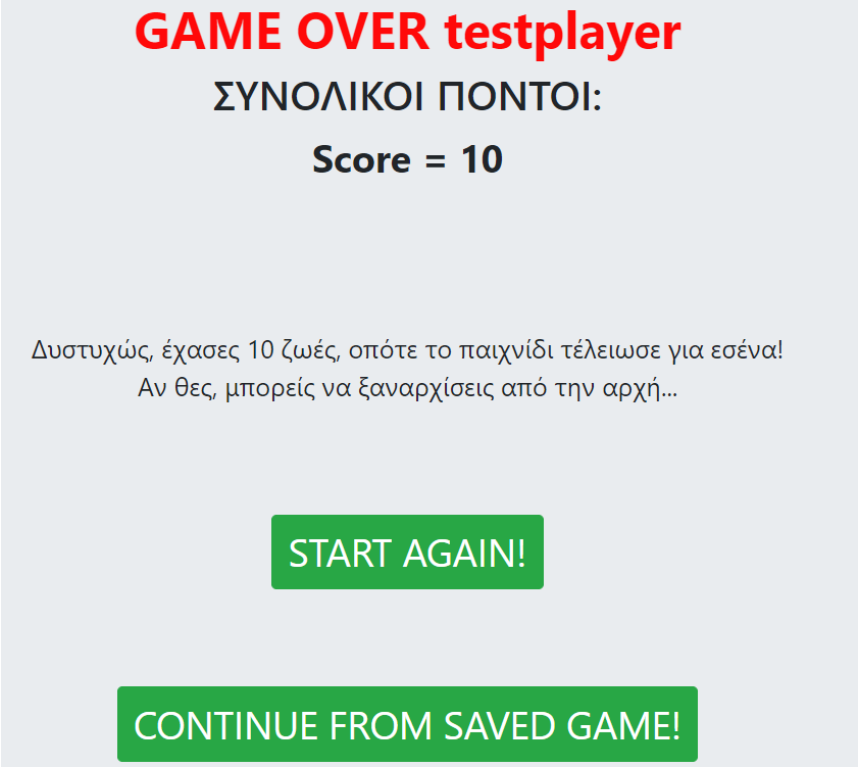

Image 6.13: Game over

6.9 Finish Game

In case that the player answers correctly all the answers until the last level (now 7) without losing 10 lives in total, he/she achieves the goal of resolving all the puzzles and finishes the game.

7. System Architecture

7.1 Introduction

In the current chapter we describe the programming languages and the tools that have been used for the creation of this project. We also present some portion of the code, providing an insight in the programming logic, thus facilitating the understanding of the code.

The "Room of Associations" project has been written in PHP programming language with MySQL database. Part of the application is written in JavaScript programming language and there is been of course plenty of use of HTML and CSS to build the necessary pages. Part of the project used the MVC system architecture (Model-View-Controller) so we could divide the application into 3 main logical components.

In case of software, the only tools that have been used are Notepad++ as code text editor, FileZilla as an FTP client to transfer the files to a web server and of course a web browser (Chrome, Firefox, etc.) to test, implement and play the game.

In short:

- PHP (main programming language)
- MySQL (database script language)
- JavaScript (programming language)
- HTML (markup language)
- CSS (style sheet language)
- MVC (system architecture in a part of the project)
- Notepad++ (code text editor)
- Filezilla (ftp client)
- Web Browser (Chrome, Firefox, Edge)

7.2 PHP

PHP is the most used programming language nowadays for the creation of dynamic websites. It is a server-side language (it runs from the server's side, not the client's). It is widely known for the ease of use, it's efficiency and because it is fast and light. *(Tatroe, Kevin, 2020)*

The way that PHP makes a webpage dynamic, is that a .php type file (for example page.php) runs through a compatible server (Apache, Nginx, Lightppd etc) and creates a real-time content that is been shown on client's browser (in simple HTML format).

The most popular and important CMS (Content Management Systems) are written in PHP. For example:

- WordPress
- Joomla!
- Drupal
- Magento
- Prestashop
- OpenCart
- B2evolution
- Elxis
- Mambo
- MediaWiki
- Moodle
- OsCommerce
- PhpBB
- PHP-Nuke

PHP began back in 1994, when Danish university student Rasmus Lerdorf created with the use of C programming language, a script for personal use. This script helped Rasmus to keep online stats about the visitors who were viewing his CV in his website. Just 3 years later, PHP was hosted in more than 50.000 sites. From 1997 and after Zend company from California, U.S is responsible for PHP language, which is now in version 8. *(Lerdorf, Rasmus, Kevin Tatroe, and Peter MacIntyre, 2006)*

The most revolutionary change in PHP happened back in 2000 when Zend founders (Andi Gutmans και Zeev Suraski) rebuilt the language based on a new technology named "Zend Engine", which brought the following improving changes:

- Multiple servers support
- Better memory usage
- Avoid of memory leak
- Improvement in speed and performance

In the following image, we can see an example of PHP code (source: Wikipedia)

```php
<?php
// single line comment 1
# single line comment 2
/* multi line comment - ?> */
/** doc-style comment - ?> */
function printNumber()
{
    $number = 1234;
    print "The number is $number,\n<?xml version=\'
    for ($i = 0; $i <= $number; $i++) {
        $x++;
        $x--;
        $x += 1.0;
        $z = 2. ; // error
    }
    if (PRINT_SUMMARY_CONSTANT)
        echo <<< custom_HEREDOC
<br />
Summary:<br />
The \$x variable has value: $x
<br />
custom_HEREDOC;
}
?>
```

Image 7.1: PHP code

As we already said, ROOM OF ASSOCIATIONS was based on PHP. Here is an example of how via the use of this programming language the system manages user's score in case of right or wrong answer.

```php
$answer = $_POST['ans'];
if($answer == ''){ // empty answer
   $_SESSION['level1failed'] = "Please Select any Answer and hit Submit.";
} else {
   $_SESSION['level1answers'][] = $answer;
   if ($answer == $correct_answer) { // right anwser
   $_SESSION['user_score'] = $_SESSION['user_score'] + 10;
      $_SESSION['user_progress'] = 1;
```

```
        $sql = "UPDATE users set user_progress = 1, user_score = '".$_SESSION['user_score']."' where user_id = $currentuserid";
        $conn->query($sql);
         $_SESSION['level1success'] = "RIGHT ANSWER - <a href='/level2.php'>PROCEED TO THE NEXT LEVEL</a>";

      } else { // wrong answer

        $_SESSION['user_score'] = $_SESSION['user_score'] - 2;

        $_SESSION['level1failed'] = "Wrong Answer -2 points";

        // user losses life
        $_SESSION['user_lifes'] = $_SESSION['user_lifes'] - 1;

         $sql = "UPDATE users set user_lifes = '".$_SESSION['user_lifes']."', user_score = '".$_SESSION['user_score']."' where user_id = $currentuserid";
        $conn->query($sql);

         $_SESSION['loselife'] = "You lose one life";
      }
    }
```

7.3 HTML 5

HTML5 is the markup language for creating and presenting content on the Web (Lubbers, Peter, 2014). It is the latest HTML version and it was first released in a January 2008, with a major update in October 2014. It is currently maintained by a consortium of the major browser vendors (Apple, Google, Mozilla, and Microsoft) and WHATWG (Web Hypertext Application Technology Working Group).

Goals

Version 5 goal was:

- To improve the language with support for the latest multimedia
- To keep the language both easily readable by humans and and devices
- To remain compatible with older software.

Features

HTML5 came by with a significant number of new elements and form controls:

- HTML new elements were: article, aside, audio, bdi, canvas, command, data, datalist, details, embed, figcaption, figure, footer, header, keygen, mark, meter, nav, output, progress, rp, rt, ruby, section, source, summary, time, track, video
- And new types of form controls were: dates and times, email, URL, search, number, range, tel, color

In this project, to be more specific, the video element has been the most used, as we used it in every Game Level (plus the intro). The video tag is used to embed video content in a document, like a movie clip It contains one or more

source tags with different video sources and the browser will choose the first source it supports. The supported video formats for video tag in HTML are MP4, WebM, and OGG.

All video files in this project are compressed in MP4 filetype. In the image below, we can see how easily we embed a video in our project with its respective controls.

```
<?php $video = "videoz/video2.mp4" ?>
<?php $leveltitle = "Level 2 - Τα Βιβλία"; ?>

<main role="main">

  <div class="jumbotron">
    <div class="container">
        <?php if(isset($_SESSION['savemessage'])){?>
            <div class="alert alert-success"><?php echo $_SESSION['savemessage'];?></div>
        <?php unset($_SESSION['savemessage']);
        } ?>
      <h1 style="text-align: center;"><?php echo $leveltitle ?></h1>

        <!-- start page -->
        <p style="text-align: center;">
        <video width="1080" height="720" controls oncontextmenu="return false;">
          <source src="<?php echo $video ?>" type="video/mp4">
        </video>
        </p>
```

Image 7.2: HTML5 Video feature

It is also important to mention that, for security reasons, the /videoz directory (where all the videos are stored) is encrypted, so no one can directly access them without access (for example by typing a video address directly on a web browser). These videos are accessible only to logged-in authorized players, and of course only if the player has access to the level that follows each video.

7.4 Notepad++

The software tool that we use to write and edit the code is Notepad+++. Notepad++ is a tool that replaces the Default Windows Notepad and is designed specifically for code edit. It is an ideal tool for experienced programmers, as much as the new ones. This tool is multilanguage, but in our case, English language was preferred.

Notepad++ uses tabs, something that allows us to have multiple opened and ready to edit files, in a single window. The software is free and open-source. It supports almost every old and modern programming language and it is extremely popular because it provides syntax highlighting and code folding, stuff that make programmer's life much easier!

Notepad++ General Characteristics:

- ANSI, UTF, UCS encoding support
- Auto-backups of unsaved files

- Search & Replace operators
- Data comparison
- Drag and drop
- Tabs
- Code Folding
- Split Screen
- Text Zoom
- Macro-commands
- Multiline edit
- Plugins

Notepad++ Programming Characteristics:

- Autofill
- Bookmarks
- Regular Expressions
- Spell Check
- Automatic programming language detection
- Manual assign of language in a file
- Brackets Autocomplete
- HTML & XML tags Autocomplete
- Run code based on macro-commands
- Functions list
- Multi-programming languages support

In the image below, we can see an example of how Notepad++ can manage multiple tabs, multiple windows, as well as syntax highlighting and spell check.

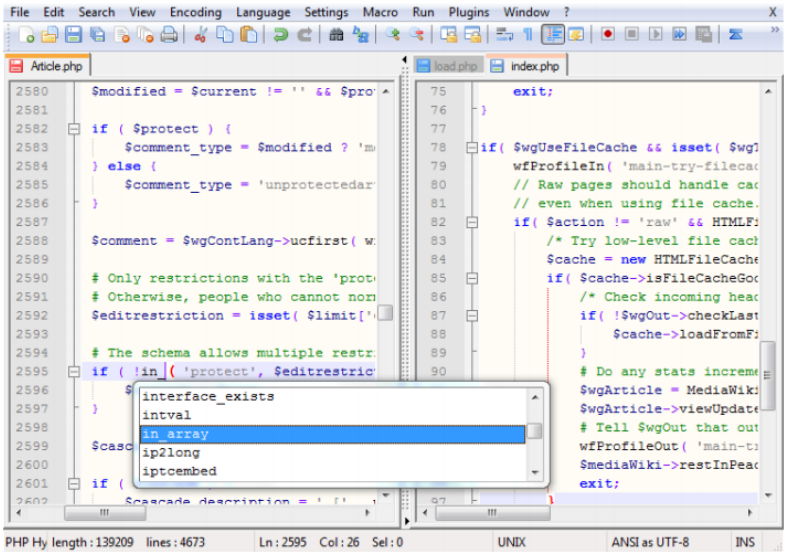

Image 7.3: Notepad++

7.5 MySQL

MySQL is a very popular relational database management system. The software runs in a server, giving access to many users at the same time in a total of relational databases. MySQL is the most popular database system and it is been used in some of the most important websites on the worldwide web, such as YouTube, Wikipedia, Facebook, Twitter, Google etc.).

MySQL is free and Opensource. Due to scientific benchmarks, MySQL is the database type with the highest performance, as it requires minimum processing power and at the same time it can provide maximum compatibility.

When clients need to ask some information from the database, it needs server's interference. In the matter of communication between database and client side, PHP comes to help as it collaborates perfect with MySQL. PHP commands have the ability to integrate inside HTML, so client through the browser can run queries to the database. In the following image we can see an example of this type of communication *(Nixon, Robin, 2014)*.

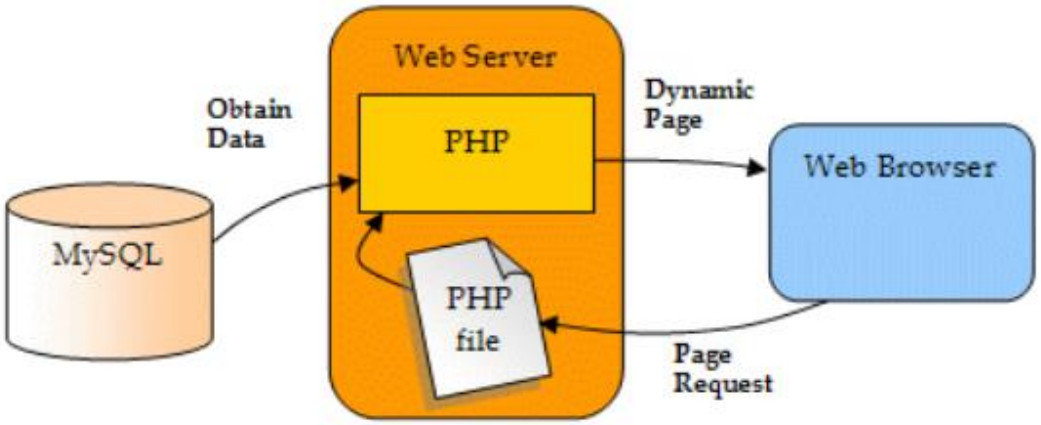

Image 7.4: MySQL

7.6 PHPMyAdmin

For the administration of the game's database, we chose PHPMyAdmin, which is an open-source tool for MySQL databases management and administration. Through this tool, we can process actions as creation, edit, delete of lines, rows and tables in a database. In the following image, we can see the PHPMyAdmin working environment. *(Delisle, 2014)*

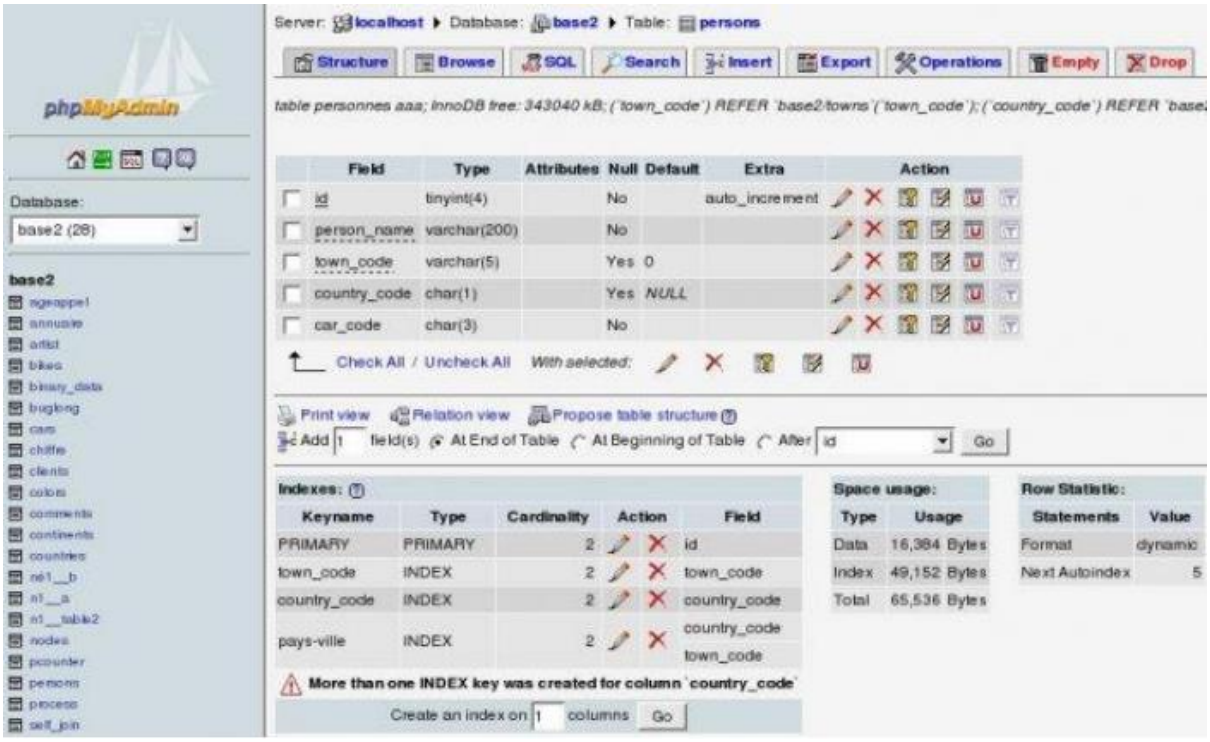

Image 7.5: PHPMyAdmin

PHPMyAdmin platform accepts SQL queries. Besides the ready-to-use functions that are available from the user's menu, through the graphic environment, the user can run any valid SQL command in a database and that is something that makes the platform powerful. In the following image we can see an example of an SQL query procedure, through PHPMyAdmin.

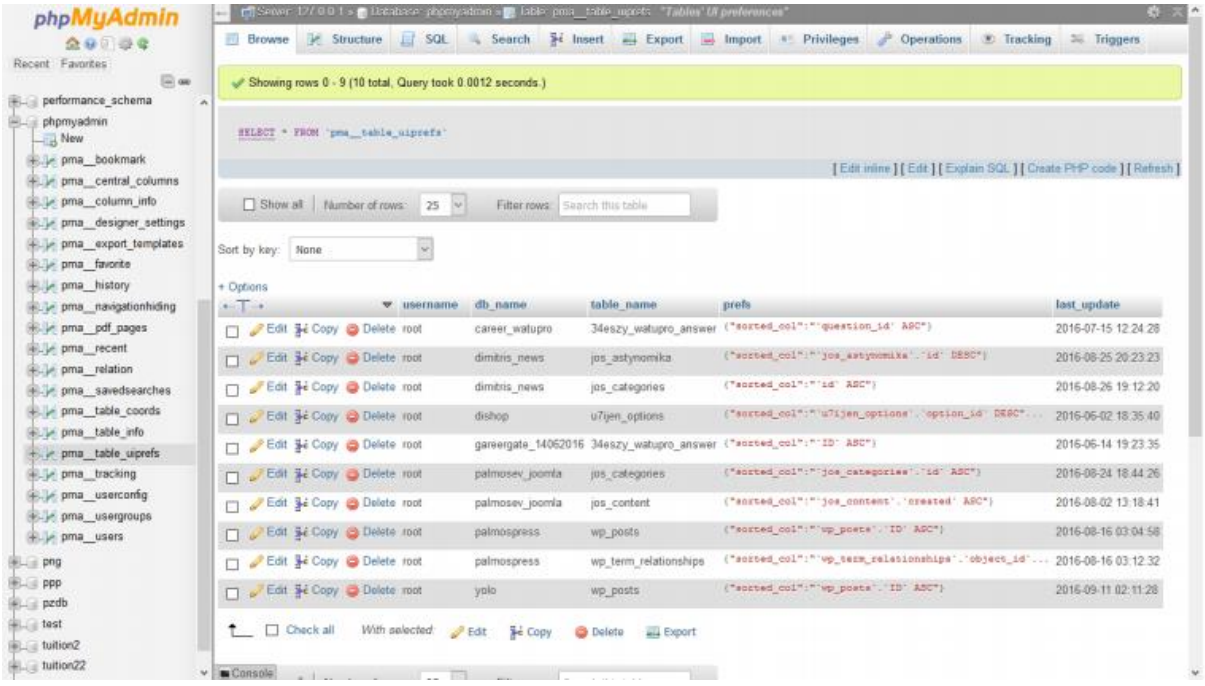

Image 7.5: PHPMyAdmin II

PHPMyAdmin General Characteristics

- Browser based environment
- MySQL & MariaDB database type administration
- CSV data import
- Export data to many supported filetypes (CSV, SQL, XML, PDF etc)
- Parallel multi-server administration
- SQL commands support
- Search & Find in Database
- Compatibility with almost any operating system
- Data conversion (for example: image to text)
- Multi-language support

The fact that PHPMyAdmin accepts MySQL commands and that is also freeware, light and accessible from a simple browser, were the main reasons that this platform was used in our project as a database management solution, instead of any other solutions (such as MySQL Workbench, Navicat). Most of the times, during the development, when we needed to integrate SQL queries into the PHP code, we tested them first on PHPMyAdmin and after we got the results we needed, we passed them into the code.

For example, the following query returns the quiz questions and answers of Level 1

```
SELECT * FROM answers WHERE question_id = 1
```

In the following image we can see how we can run this query in PHPMyAdmin and what results it is returning:

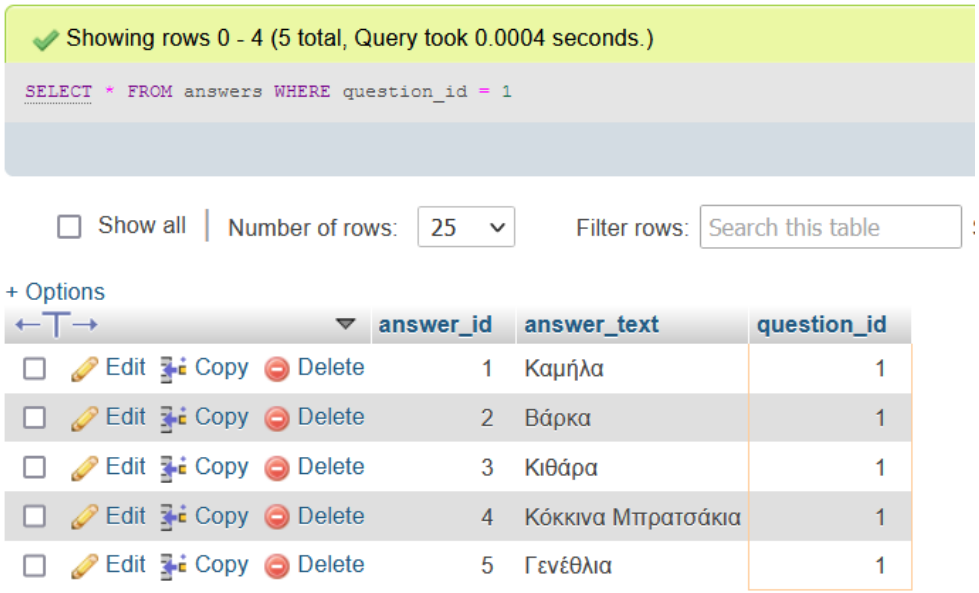

Image 7.6: SQL query

We, then can transfer this query in our code with the use of PHP

```
$getquestion = $conn->query("SELECT * FROM questions WHERE question_id = 1") or die($conn->error); // Fetch Question from the table
$question = $getquestion->fetch_array();
$correct_answer = $question['correct_answer_id'];

$getanswers = $conn->query("SELECT * FROM answers WHERE question_id = 1 order by RAND()") or die($conn->error); // Fetch Answers from the table
```

7.7 MVC

What is MVC architecture?

MVC architecture (Model, View, Controller) is a software architecture model for the creation of user interaction environments. In this type of model, the application is been divided to three, different but interconnected, parts in order to divide the export of the information to the user in the form that it has been saved in the system. *(Deacon, 2009)*

The main characteristic of these 3 parts, is that they are completely distinct and each one has its own role in the application. This way it separates the internal information display from the way that they appear or being accepted from the user. MVC architecture is very popular in web development.

- Model - Manages the data storage and restore
- View - The display of the information to the user
- Controller - Process the data entrance and command export to model & view

Model

Model is an object or a data structure and has the role of the connecting wire between data storage of data and program's logic. It is usually the programming portrayal of a database table.

View

View is the visual representation of the model and it is connected with the model, deriving and updating the information that is necessary for its representation (for example via an SQL query in the database). We can practically say that the graphic part that user sees in his/her interface, it is the view.

Controller

Controller is the link between view and controller, but also between user and system. User, through the inputs that are provided by the controller, can see the information that he/she asks via the "view". In the controller there is the whole

logic of each system, as inside it there are been implemented the procedures that later shown to the user via the “view”. In an ideal MVC application, controller has not direct access to the data, but only to the model, and through the model to the data.

Even if it was initially developed for desktop programming, MVC has been as well adopted as an architecture for applications of the world wide web in the most important programming languages. Many frameworks have been created based on this model, but they often differ in the way that MVC operations are been shared between client and server.

First MVC web systems had a very materialist approach on client, so as a result the whole MVC model to be mounted on the server. In this approach the client sends a request to the controller and then updates the client’s page from the view. This model is completely on the server. In the course of time and the improvement of MVC frameworks, the most modern frameworks have created a logic in which some MVC parts are been executed in client’s side.

MVC Advantages

- Problems Segregation
- Scalability
- Ease of Testing
- Clean URLs

The fact that in an MVC application the three parts (Model – View - Controller) and distinct and co-operative, is the most important advantage and makes the structure and the software’s logic simpler and the programmer’s life easier.

Another main important advantage is the scalability, that is the ability to add in the software extra features in the feature. The ease that problems segregation provides us in three levels, makes the scalability very easy, in contrast to older and different models and systems.

In the testing level it is easy, if a problem is found, for the programmer to understand in which part the problem exists, in order to find it and fix it.

The clean URLs are the capper of MVC advantages. A web address like “website.com/a-sample-page” it is very SEO and Web Search Machines friendly and can be easily created dynamically from an MVC system. A counterexample of a non-SEO and Web Search friendly URL type is for example a URL like this: “website.com/3847=dyscfcih-89sd7f89sdh” (with a lot of random strings etc.)

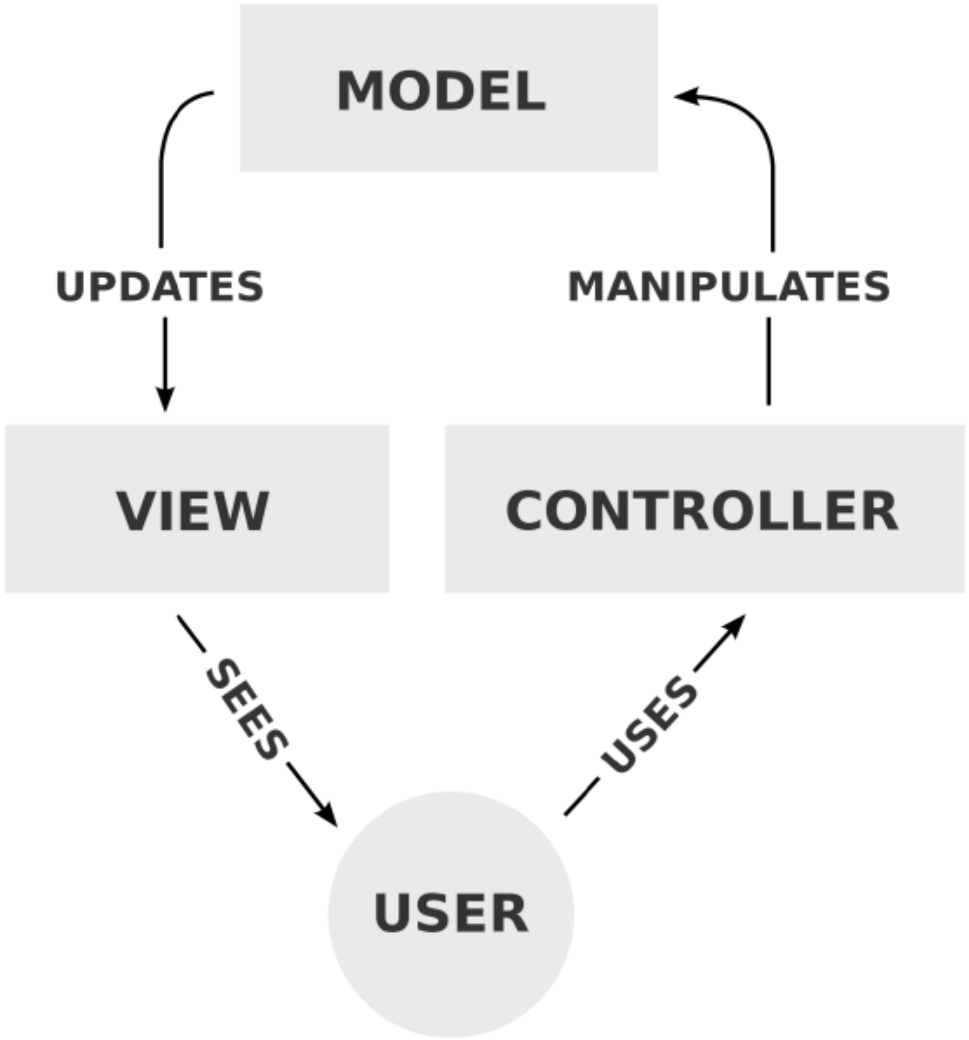

Image 7.6: MVC

8. Design and Development

8.1 Introduction

In this chapter we will take a look on how we designed and developed the project "Room of Associations" game and we will see some code snippets from the project files, as well. The project was built from scratch, with no use of a CMS, but of course we used some libraries to implement it better, such as Bootstrap for the creation of the site's template and jQuery – which is a famous JavaScript library.

8.2 Template

As we said before, for the creation of the site's template we used Bootstrap, which is the most popular HTML, CSS, and JS library. Bootstrap is free and open-source and very focused on responsiveness (mobile friendly etc.). The current version of this library, as we write this document, is 4.5.2 and it is also the version that we used in our project.

The main templates files are located under the /template directory. In the next image we see the files tree on that directory.

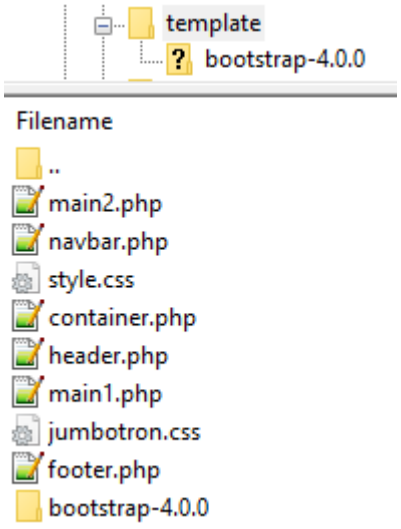

Image 8.1: Files tree

As we can see, the Bootstrap library files are located in a separated subfolder, so we can call them from there. But, we preferred, for mostly speed reasons, to use a CDN in order to call the whole library. This also gives as the ability – apart from the speed – to update the library version with a single line of code on the future updates.

In the following code snippet, we can see the code of */template/header.php* file. As we can see, both Bootstrap and jQuery are being called in the header and from maxcdn URLs, which is a fast and quick solution.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Room of Associations</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1">
  <!-- Custom styles for this template -->
  <link href="template/jumbotron.css" rel="stylesheet">
  <link rel="stylesheet" href="template/style.css">
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.m
in.css">
  <link rel="shortcut icon" href="/favicon.ico" type="image/x-icon">
  <link rel="icon" href="/favicon.ico" type="image/x-icon">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js
"></script>
  <script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popp
er.min.js"></script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min
.js"></script>
</head>
<body>
```

Every template file is been called on each main game's file that is needed (for example in the Level files) so that we avoid the code duplication in each file, save space and make our life easier in general, by updating each template file once, when is needed. For example, in the following code snippet, we can see the code of */template/header.php* file that is been called on every single php file of the project.

```
  <div class="container">
    <div class="row">
      <div class="col-md-4">
        <h2>Σχετικά</h2>
        <p>Το παρόν αποτελεί προϊόν πτυχιακής εργασίας του
μεταπτυχιακού φοιτητή Αρκολάκη Δημητρίου για το Πανεπιστήμιο Δυτικής
Αττικής και ουδεμία σχέση έχει με εμπορική εκμετάλλευση.</p>
      </div>
      <div class="col-md-4">
        <h2>ISICG</h2>
        <p>Κοινό ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ "Πληροφορική,
Σύνθεση Εικόνων & Σχεδιασμός Γραφικών με κατεύθυνση Τεχνολογίες
Διαδικτύου και Πολυμέσων" του Τμήματος Μηχανικών Πληροφορικής &
Υπολογιστών του ΠΑ.Δ.Α σε συνεργασία με το Πανεπιστήμιο της Limoges
στη Γαλλία.</p>
      </div>
      <div class="col-md-4">
        <h2>Επίβλεψη</h2>
        <p>Επιβλέπων Καθηγητής: Βουλόδημος Αθανάσιος, Επίκουρος
Καθηγητής του Τμήματος Μηχανικών Πληροφορικής & Υπολογιστών του
Πανεπιστημίου Δυτικής Αττικής.</p>
      </div>
    </div>
    <hr>
  </div>
```

Another important template file, is */template/style.css* file which contain almost every style edit of our project and is of course based on the main CSS of Bootstrap library. As an example, in this code snippet, we can see three of the rules that we created to set the project's appearance to our needs.

```
/* The container */
.radiocontainer {
  display: block;
  position: relative;
  padding-left: 35px;
  margin-bottom: 12px;
  cursor: pointer;
  font-size: 22px;
  -webkit-user-select: none;
  -moz-user-select: none;
  -ms-user-select: none;
  user-select: none;
}

/* Hide the browser's default radio button */
.radiocontainer input {
  position: absolute;
  opacity: 0;
  cursor: pointer;
}

/* Create a custom radio button */
.checkmark {
  position: absolute;
  top: 0;
  left: 0;
  height: 25px;
  width: 25px;
  background-color: #ff7777;
  border-radius: 50%;
}
```

8.3 User Accounts

Before the development of the main game, the first think we had to take care of, is how to create a users-players system. The method that we used in this project, is to divide the user management files in two parts:

- Main account functions (registration, login, logout, reset)
- Player functions (sessions, save game, continue game, game over)

Main Account Functions

All user details are stored in the database and been fetched through the database. In the following code snippet, we can see how the code handles user's nickname, user's password and password's validation throughout the registration.

```
    // Validate user_nickname
    if(empty(trim($_POST["username"]))){
        $user_nickname_err = "Please enter a username.";
```

```
    } else{
        // Prepare a select statement
        $sql = "SELECT user_id FROM users WHERE user_nickname = ?";

        if($stmt = mysqli_prepare($conn, $sql)){
            // Bind variables to the prepared statement as parameters
            mysqli_stmt_bind_param($stmt, "s", $param_user_nickname);

            // Set parameters
            $param_user_nickname = trim($_POST["username"]);

            // Attempt to execute the prepared statement
            if(mysqli_stmt_execute($stmt)){
                /* store result */
                mysqli_stmt_store_result($stmt);

                if(mysqli_stmt_num_rows($stmt) == 1){
                    $user_nickname_err = "This username is already
taken.";
                } else{
                    $user_nickname = trim($_POST["username"]);
                }
            } else{
                echo "Oops! Something went wrong. Please try again
later.";
            }

            // Close statement
            mysqli_stmt_close($stmt);
        }
    }

    // Validate password
    if(empty(trim($_POST["password"]))){
        $user_password_err = "Please enter a password.";
    } elseif(strlen(trim($_POST["password"])) < 6){
        $user_password_err = "password must have atleast 6
characters.";
    } else{
        $user_password = trim($_POST["password"]);
    }

    // Validate confirm password
    if(empty(trim($_POST["confirm_user_password"]))){
        $confirm_user_password_err = "Please confirm user_password.";
    } else{
        $confirm_user_password =
trim($_POST["confirm_user_password"]);
        if(empty($user_password_err) && ($user_password !=
$confirm_user_password)){
            $confirm_user_password_err = "user_password did not
match.";
        }
    }
```

Player Functions

Player functions are maybe the most important and complex part of user account functions in the project, but not all player functions are stored in the database. That is because we preferred to use PHP sessions for sessions and store the rest of it in the database.

PHP $_SESSION is an associative array containing session variables available to the script, so we preferred this way in case the player closes accidentally the browser or the tab, as this variable store player's current lives, score in the browser's cache. For example, for current user progress, we use it like this:

```
        <p class="userprogress">
            Current User Progress: <?php echo
$_SESSION['user_progress']+1;?>
        </p>
```

Or, another example, is how we change "Remaining Lives" text color depending how many lives left for the player

```
            if ($techi_totali > 7) { // 8, 9, 10
                echo "<span style='color: green;'>Current User Lifes:
".$techi_totali."</span>";
            } elseif ($techi_totali > 3 && $techi_totali <= 7) { //
4, 5, 6, 7
                echo "<span style='color: #73730c;'>Current User
Lifes: ".$techi_totali."</span>";
            } elseif ($techi_totali > 0 && $techi_totali <= 3) { //
1, 2, 3
                echo "<span style='color: #ff0909;'>Current User
Lifes: ".$techi_totali."</span>";
            } else {
                echo "is there a bug?";
            }
```

Of course, we also use SQL to store what $_SESSION gets, when it is necessary. For example, the following code shows us how the player losses one life, after submitting a wrong answer, with the use of PHP $_SESSION and SQL in order to store it as updated data in the database.

```
        } else { // wrong answer
            $_SESSION['user_score'] = $_SESSION['user_score'] - 2;
              $_SESSION['level2failed'] = "Wrong Answer -2 points";

            // user losses life
              $_SESSION['user_lifes'] = $_SESSION['user_lifes'] - 1;
              $sql = "UPDATE users set user_lifes =
'".$_SESSION['user_lifes']."', user_score =
'".$_SESSION['user_score']."' where user_id = $currentuserid";
              $conn->query($sql);
              $_SESSION['loselife'] = "You lose one life";
        }
```

Similar process, as the above, occurs in every level of the game.

8.4 Levels

Each Level of the game has a separate file, but they mostly follow similar logic. We can describe this logic, like this:

1. Initialize the session
2. Check if the user is logged in, if not then redirect him to login page
3. Check if user has lives

If the above is all true

- Fetch Question from the database
- Fetch Answers from the database
- Do stuff on Answer Submit
    - Case of right answer
    - Case of wrong answer
- Fetch the HTML

In the code screenshot below, we can see how the code works on level pages (except the "Fetch the HTML" part).

```php
// Check if the user is logged in, if not then redirect him to login page
if(!isset($_SESSION["loggedin"]) || $_SESSION["loggedin"] !== true){
    header("location: login.php");
    exit;
}

$currentuserid = $_SESSION["user_id"];
require 'config.php';

if($_SESSION['user_progress'] < 1){
      header("Location:level".($_SESSION['user_progress']+1).".php");
    }

// Check if user has lifes
$techo_totalo = $_SESSION['user_lifes'];
if ($techo_totalo < 1) {
    header("location: gameover.php");
    exit;
}

$getquestion = $conn->query("SELECT * FROM questions WHERE question_id = 2") or die($conn->error); // Fetch Question from the table
$question = $getquestion->fetch_array();
$correct_answer = $question['correct_answer_id'];
$getanswers = $conn->query("SELECT * FROM answers WHERE question_id = 2 order by RAND()") or die($conn->error); // Fetch Answers from the table

// On answer submit
if(isset($_POST['submit'])){

    $answer = $_POST['ans'];
    if($answer == ''){ // empty answer
        $_SESSION['level2failed'] = "Please Select any Answer and hit Submit.";
    } else {
        $_SESSION['level2answers'][] = $answer;
        if ($answer == $correct_answer) { // right anwser
        $_SESSION['user_score'] = $_SESSION['user_score'] + 10;
            $_SESSION['user_progress'] = 2;

            $sql = "UPDATE users set user_progress = 2, user_score = '".$_SESSION['user_score']."' where user_id = $currentuserid";
            $conn->query($sql);

              $_SESSION['level2success'] = "RIGHT ANSWER - <a href='/level3.php'>PROCEED TO THE NEXT LEVEL</a>";

        } else { // wrong answer
            $_SESSION['user_score'] = $_SESSION['user_score'] - 2;
              $_SESSION['level2failed'] = "Wrong Answer -2 points";

            // user losses life
              $_SESSION['user_lifes'] = $_SESSION['user_lifes'] - 1;
              $sql = "UPDATE users set user_lifes = '".$_SESSION['user_lifes']."', user_score = '".$_SESSION['user_score']."' where user_id = $currentuserid";
              $conn->query($sql);
              $_SESSION['loselife'] = "You lose one life";
        }
    }
    header("Location:level2.php");
    exit();
}?>
```

Image 8.2: Level file

This way offers security and speed to our project. In every web project, even it is a simple static website – or a complex web game, security is very important and that way we try to avoid SQL injections or anything that could be vulnerable to the project.

8.5 Layers

Layers are the dynamic parts that we can embed anywhere on the website, such as score (on the header and the content), user lives & user progress (in the content, or anywhere). They consist of small code parts that can be called on any other file, such as level files or user account files.

Below, we can see the code of /layers/score.php file

```
        <p class="score">
        <?php
            $tech_total = $_SESSION['user_score'];
            if ($tech_total != 0) {
                echo "Score = ".$tech_total;
            } else {
                echo "Score = 0";
            }
        ?>
        </p>
```

This file returns the current player's score at any time. So, the only thing that we have to do, every time we need to show the player's score, is to call this layer inside another file. PHP "include" command can do this job for us. We have only to include this file, inside any other file we need. In the following code snippet, we can see how the "score layer" is being called in the "GAME OVER" screen that is being produced by the /gameover.php file.

```
        <div class="page-header" style="padding: 80px;">
            <h1 style="color: #ff0909"><b>GAME OVER <?php echo
htmlspecialchars($_SESSION["user_nickname"]); ?></b></h1>
            <h2>ΣΥΝΟΛΙΚΟΙ ΠΟΝΤΟΙ:</h2>
            <?php include "layers/score.php"; ?>
        </div>
```

8.6 Other Hacks

A lot of other small hacks are been used in the project in order to give the players a better user experience. These so-called "hacked" are just small parts of code, but very important for the game's image and UX. Some examples are:

Auto redirect after a correct answer

When a user submits an answer and this answer is correct, he/she auto-redirects to the next level after some seconds (currently set on 8). This is done with the use of the code below:

```
<?php if(isset($given_correct_answer) && $given_correct_answer ==
1){?>
    <script>
        var timer = setTimeout(function() {
            window.location='/level4.php'
        }, 8000);
    </script>
<?php } ?>
```

Randomize questions

For security reasons, in order to avoid spoil between the players, the order of possible answers in every question is random. This is done with the help of SQL, as below:

```
$getanswers = $conn->query("SELECT * FROM answers WHERE question_id =
3 order by RAND()") or die($conn->error);
```

8.7 Database

Total dynamic information of the project is stored in the database, as an organized collection of data:

- The questions and the answers of game's quizzes
- User details
- Users points and lives
- Saved games information

	Πίνακας	Εγγραφές	Τύπος	Σύνθεση
☐	answers	48	MyISAM	utf8_general_ci
☐	lifes	97	MyISAM	utf8_general_ci
☐	points	265	MyISAM	utf8_general_ci
☐	questions	6	MyISAM	utf8_general_ci
☐	saved_games	14	MyISAM	utf8_general_ci
☐	users	52	MyISAM	utf8_general_ci
	6 πίνακες	482	MyISAM	utf8_general_ci

Image 8.2: Database tables

List of database tables

All database tables use MyISAM MySQL storage engine and "utf_general_ci" encoding type, which is the best optimal solution for projects that use a lot of different alphabets and special characters. "Room of Associations" project is using both Latin & Greek alphabets as well. This Unicode encoding type manages to display any needed character in the website, without any problems.

Tables and structure

The list of the database tables is the following:

- answers
- lifes
- points
- questions

- saved_games
- users

The data structure for each table:

- answers
  - answer_id (INT)
  - answer_text (VARCHAR)
  - question_id (INT)

Each answer is connected to a question through the “question_id”. For example, in the following image we can see how answers 1-5 are connected to question 1 and answers 6-9 are connected to question 2.

answer_id	answer_text	question_id
1	Καμήλα	1
2	Βάρκα	1
3	Κιθάρα	1
4	Κόκκινα Μπρατσάκια	1
5	Γενέθλια	1
6	Νέο Αναστασιματάριο	2
7	Soloup Αϊβαλί	2
8	Οι Θρησκείες του Κόσμου	2
9	Frank Miller - Για Κόλαση Αλερετούρ	2

Image 8.3: Answers table

- lifes
  - life_id (INT, AUTO_INCREMENT)
  - life_lifes (INT)
  - life_answer_id (INT)
  - life_user_id (INT)

Each row represents a lost life from a false answer and is connected to a user through the “life_user_id” field. The total sum of the lost lives for each player is been subtracted from the total of 10 lives.

In the following image we can see the lost lives for the players with ID 12 & 13:

life_id	life_lifes	life_answer_id	life_user_id ▲ 1
11	-1	1	12
20	-1	1	12
19	-1	1	12
18	-1	1	12
17	-1	1	12
16	-1	1	12
12	-1	1	12
15	-1	1	13
14	-1	1	13
13	-1	1	13
39	-1	1	13

Image 8.3: Lives table

- points
  - point_id (INT)
  - point_points (DECIMAL)
  - point_answer_id (INT)
  - point_user_id (INT)

This table represents the gained and lost points for each player. point_answer_id field is connected with the respective answer and point_user_id with the respective player.

point_id	point_points	point_answer_id	point_user_id
269	-2	1	26
270	-2	1	26
271	-2	1	26
274	-2	1	27
275	10	1	27
276	-2	1	27
277	-2	1	28
278	-2	1	28

Image 8.4: Points table

- questions
  - question_id (INT)
  - question_text (VARCHAR)
  - correct_answer_id (INT)

This table represents the data for each question. The field question_text contains the text of each question and the field correct_answer_id the correct answer for each question.

In the image below we can see the data for the questions of the first three (3) levels:

question_id	question_text	correct_answer_id
1	Με βάση το στοιχείο που έχει, ποια φωτογραφία θα π...	4
2	Ποιο βιβλίο πρέπει να επιλέξει; Υπάρχει κάποιο που...	10
3	Ποιον δίσκο πρέπει να επιλέξει;	20

Image 8.4: Questions table

- saved games
  - save_game_id (INT)
  - save_user_id (INT)
  - user_progress (INT)
  - user_score (INT)
  - user_lifes (INT)
  - date_added (TIMESTAMP)

This table manages the saved games of the players. It is important to mention that in addons of the date_added field, which is a timestamp of datetime, “ON UPDATE CURRENT_TIMESTAMP” is been used. This means that in any update without an explicit timestamp results in an update to the current timestamp value.

save_game_id	save_user_id	user_progress	user_score	user_lifes	date_added
1	2	1	6	2	2021-03-11 11:56:41
2	27	0	0	8	2021-03-11 10:42:09
3	28	1	0	4	2021-03-11 10:47:43
4	29	1	4	7	2021-03-11 11:13:03
5	30	1	4	7	2021-03-11 11:17:33
6	31	0	-4	8	2021-03-11 11:26:08
7	32	0	-4	8	2021-03-11 11:58:56
8	23	1	4	7	2021-03-11 12:14:52
9	33	0	-6	7	2021-03-11 12:37:39
10	36	1	8	9	2021-03-18 18:03:40

Image 8.5: Saved games table

- users
  - user_id (INT)
  - user_nickname (VARCHAR)
  - user_score (INT)
  - user_progress (INT)
  - user_lifes (INT)
  - user_password (VARCHAR)
  - user_created (DATETIME)

In “users” table, all the users account data are stored. The fields user_score, user_progress & user_lifes are being updated and depended on other database tables, such as lifes, points.

The field “user_password” is using a hash encryption for security reasons.

47	stef2	58	6	9	$2y$10$yt0XyiCl3xuL/kn9mqpu0OUqZ2jQmcVLI2OdazIDVnr...	2021-03-24 22:59:35
48	aris	60	6	10	$2y$10$d2ZKSoSSj3wmwrE4FttQReujD4BqYF0pZraTPQ5EOJD...	2021-03-26 22:47:26
49	Arko	24	3	7	$2y$10$2JcX6CwQJQVslKhfm6GTqOvOn72QumZ7aHwQ1xl/a1w...	2021-04-15 13:56:33
50	testplayer	10	3	0	$2y$10$4ylK6i.cNt8NQWCYVh8.x.rSi27yWPGgTS0Ee.KBgSL...	2021-05-13 00:42:41
51	mantzios	26	3	8	$2y$10$yFfH6ZS5G0Ft7TO0lJyse.VK3Ga0lGU6DPpwW5e.DVF...	2021-06-24 18:22:24

Image 8.6: Users table

9. Future Improvements

9.1 Introduction

Every game is updating itself version by version. These updates can be graphics or sound improvement, change of gameplay, new levels, new heroes etc. In this chapter we will see can what can be improved in future versions of "Room of Association" game and some new features and options will be proposed as well.

9.2 E-mail integration

In game's current situation, a new user can register without an email. That is because the game was developed in a local environment (personal computer's localhost), so it was hard to implement locally a mail server, for the emails to be delivered. The benefits of implementing an e-mail integration system on the project would be many, such as:

User Confirmation: The majority of world-wide web's website always ask the new user to confirm that the e-mail he/she entered during the registration is real, usually by clicking on a unique confirmation URL (that new user can see on the confirmation e-mail) or by entering manually a security code (usually 4 to 8 digits) that the user can find on the registration e-mail.

Avoid spam: Spammers and spam bots are all over the internet, so without an e-mail confirmation, it would be hard for them to register tons of non-existing users.

Notifications & Updates: With the existence of an e-mail integration, users can always stay updated about game's news, platform's announcements, as well as notifications about their accounts (password change, account deletion etc.)

There is a lot of options in order to integrate the project with a mail system. The most popular is PHPMailer. PHPMailer is a PHP library for safe and easy email sending through a web server. It is related to SMTP protocol and it is the most popular solution for email sending via PHP and was created back in 2001 from Brent R. Matzelle.

PHPMailer Main characteristics:

- Plain Text
- HTML Support
- Files Attachment
- SSL & TLS
- SMTP & POP3 Support
- PHP sendmail method support
- IDN
- DKIM

With the use of SMTP we can make sure that the emails are been sending safely and that we avoid spam, as much as we can.

9.3 Multilanguage Support

Currently the game's language is English (except from the videos that are in Greek – as they contain books and LPs with Greek covers and Greek subtitles, as well). All the menus, buttons etc. are in English language. A multilanguage support would help visitors from all over the world to join and play the game. A full translation of the game could be divided in the following parts:

- Functionality Translation (such as buttons, menus, options, user's menu)
- Content Translation (such as questions, answers and text)
- Video Subtitles

There are three ways of translating a web project:

- Language switch in any part of the webpage, with complete connection of all the different parts between them.
- Different subdomain for each language (that consists of a completely independent website for each language)
- Translation "on the fly" with a service like Google Translate (not really accurate, but doesn't need a lot of extra work)

Of-course, the first way, in much more dynamic and sufficient.

An idea of how the user could switch languages in our game, would be a list of flags (that would be the available languages) on the top menu.

9.4 Hall of fame

A lot of games worldwide, especially these who are on the web, have a list of the top scorers, best players by achievements etc. These lists are known as "Halls of fame".

Even though "Room of Associations" keeps all player's scores in the database, it doesn't have yet a "Hall of fame". This could be a nice addition in one of the next updates, with a "Hall of fame" of top scorers among all the player, who finished the game successfully with the most possible points in the less possible time.

9.5 Admin System

Even though the game – and the project's website in general – is dynamic, currently there isn't any administration panel where the administrators can change something in the game, as the content is even hardcoded or had been manually entered in the database through the development.

It would be easier and more sufficient for game administrators to have an admin system, where they could easily update content (such as texts, descriptions etc.), update/add/remove questions and answers, or even manage levels and videos as well.

An Admin System would require an Admin Role. The game currently has only the player role, as any changes or admin access are directly from the developer, through code. So, as future improvements, a lot of new roles in the website could be created, such as:

- Admins
- Game managers
- Moderators

10. Conclusion

The main subject of this postgraduate thesis was the combination of Game Development & Web Development, along with Visual Content Production. All the technologies and developments methods, that could be possibly used, and the most efficient of them had been chosen.

The purpose of the web application was the creation of a modern and user-friendly web browser-based game. The development of this web application was divided in several parts. Each part of it was developed separately from the others, by strictly following the initial plan and design. The development started after the stages of research and design were completed.

Due to fact that the game developer wanted to create a game with the latest available tools, the use of modern programming languages and tools for the completion of this project was of great importance. Also, the programming and web security is a sensitive matter and of great significance for the protection of user's personal data, thus special concern it was given in this area.

This postgraduate thesis has been completed with the reference in future improvements and additions of the game.

11. Bibliography

Books & Publications

Dillon, Teresa. "Adventure games for learning and storytelling." *UK, Futurelab Prototype Context Paper, Adventure Author* (2005).

Perron, Bernard. "From gamers to players and gameplayers: The example of interactive movies." *The video game theory reader*. Routledge, 2013. 259-280.

Tatroe, Kevin, and Peter MacIntyre. *Programming PHP: Creating Dynamic Web Pages*. O'Reilly Media, 2020.

Lubbers, Peter, et al. *Pro HTML5 programming*. New York, NY, USA:: Apress, 2011.

Nixon, Robin. *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5*. " O'Reilly Media, Inc.", 2014.

Delisle, Marc. *Mastering phpMyAdmin 3.4 for effective MySQL management*. Packt Publishing Ltd, 2012.

Deacon, John. "Model-view-controller (mvc) architecture." *Online http://www. jdl. co. uk/briefings/MVC. pdf* (2009).

Ammanabrolu, Prithviraj, and Mark O. Riedl. "Playing text-adventure games with graph-based deep reinforcement learning." *arXiv preprint arXiv:1812.01628* (2018).

Salazar, Francisco Lepe, Tatsuo Nakajima, and Todorka Alexandrova. "Visual novels: An methodology guideline for pervasive educational games that favors discernment." *International Conference on Grid and Pervasive Computing*. Springer, Berlin, Heidelberg, 2013.

Cruz-Neira, Carolina, Marcos Fernández, and Cristina Portalés. "Virtual reality and games." (2018): 8.

Lerdorf, Rasmus, Kevin Tatroe, and Peter MacIntyre. *Programming Php*. " O'Reilly Media, Inc.", 2006.

Ullman, Larry. *PHP for the Web: Visual QuickStart Guide*. Peachpit Press, 2009.

Welling, Luke, and Laura Thomson. *PHP and MySQL Web development*. Sams Publishing, 2003.

Karl Bunyan. *Build an HTML5 Game: A Developer's Guide with CSS and JavaScript.* Paperback, 2015

Ullman, Larry. *PHP Advanced and Object-oriented Programming: Visual Quickpro Guide*. Peachpit Press, 2012.

Allen, Michael W. *Michael Allen's guide to e-learning: Building interactive, fun, and effective learning programs for any company*. John Wiley & Sons, 2016.

Vara, Clara Fernández. *The tribulations of adventure games: integrating story into simulation through performance*. Georgia Institute of Technology, 2009.

Websites

WikiPedia

https://wikipedia.org/

W3Schools

http://www.w3schools.com

StackOverflow

http://stackoverflow.com/

PHP Net

https://secure.php.net

Stack Exchange

http://stackexchange.com