POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

Cicero Reloaded: A Mixed Reality Tool for Enhancing Memorization with the Ciceronian Loci Method

Tesi di Laurea Magistrale in Computer Science and Engineering - Ingegneria Informatica

Matteo Baggetto, 10697104

Advisor:
Prof. Nicoletta Di Blas

Co-advisor:
Matteo Valoriani

Academic year:
2024-2025

Abstract: The Ciceronian Loci Method, a powerful mnemonic technique with ancient origins, has seen limited digital adaptation and has yet to fully exploit the potential of immersive technologies. While some rudimentary applications exist in VR and AR, none make use of Mixed Reality (MR). This thesis addresses these gaps by presenting the design, development, and validation of a novel MR application that revitalizes the memory palace concept for contemporary educational use. MR offers a key advantage over VR and AR: it preserves the real world (in keeping with the original spirit of the technique) while allowing digital content to merge with real-world elements.
Our approach empowers users with deep personalization through custom content uploading, ensures persistence of memory palaces via spatial anchors, enables users to freely move and physically interact with virtual content within their space. To enhance engagement, the application integrates gamification and sharing features, allowing for collaborative and competitive learning scenarios.
The application's usability and perceived usefulness were evaluated through a two-round validation process with 40 participants. The results show very positive reception, with an excellent ease-of-use score (4.45/5) and unanimous approval from the testers. Notably, despite the vast majority being unfamiliar with the technique, with 60% having never heard of it and only 7.5% having ever used it, there was a good willingness to adopt the tool (3.88/5). Qualitative feedback from a focus group further confirmed the application's perceived value.
This thesis aims to demonstrate that MR is a potent medium for the practical application and diffusion of the Method of Loci, re-imagining this ancient technique through the lens of immersive technology for today's learners.

Key-words: Mixed Reality, Ciceronian Loci Method, Memory Palace, Educational Technology, Spatial Memory, Mnemonic Techniques

1. Introduction

The work on which the thesis is based concerns the Ciceronian loci, a powerful memorization technique often known as the memory palace, which we aim to adapt to mixed reality.

1.1. Historical Origins of the Method of Loci

The origin of the Ciceronian loci dates back to ancient Greece, around 500 BC. Cicero described thoroughly in his work De Oratore the event that, according to the myth, gave rise to this technique; for this reason it takes its name from him.

It all began during a banquet, when the poet Simonides of Ceos, invited in his capacity as a poet, had a dispute with the noble organizer because of praise directed at the gods Castor and Pollux instead of him and was subsequently invited to leave the building because he was told he had to meet two young men who were insistently waiting for him. However, the roof immediately collapsed, causing the death of everyone present except Simonides. Later, when relatives sought to identify the bodies for the funerals but found them mutilated, Simonides realized that he could identify each corpse by recalling their positions at the banquet.

Cicero thus states that the identified places serve as anchors for ideas and that the very order of these places can preserve the order of the arguments [2]. In De Oratore, Cicero repeatedly revisits this concept, and it is said that he himself memorized speeches to be presented to the Senate by picturing the road he habitually walked, distributing in that path various concepts.

Cicero was not the only one to describe this technique; Quintilian also made use of it [36]. According to surviving accounts, others before Cicero might have understood and attempted to describe the technique, recognizing that an order of mental images can be memorized easier through associations with places. Unfortunately, not all information is preserved today [42]. Other authors in different times wrote about this technique, such as Tommaso d'Aquino and many others.

To summarize the method: one should select a path, a room, or an environment they know so well that they can mentally walk through it using only their imagination. At this point, one must distribute, following a certain order along the path, some concepts. When these will need to be recalled from memory, the person must imagine walking along the path in which the information is placed, and these will come to mind [45, 35].

1.2. Scientific Validation of the Technique

Ciceronian loci has not only a great history, but it's a technique well studied in recent years and its effectiveness is well documented. This approach is not only approved from scientific research but it is currently used by people who participate in memorization competitions.

It might be thought that these people have special natural talents or particularly developed parts of the brain; however, various studies have been conducted that disprove this. Several researches have analyzed the brain behavior of individuals with superior memory during tasks of memorization and recall, and it was discovered that during the use of spatial learning techniques, such as the Ciceronian loci technique, specific parts of the brain like the hippocampus, which otherwise would not be activated since they are directly related to spatial navigation, are involved. It has also been noted that the activation of these areas does not depend on the performance achieved in the activity, suggesting that it is the use of such techniques that activates specific parts of the brain [19]. In these people, particular connections between different areas of the brain have been found rather than a special development of certain parts.

Ordinary people, without any previous experience in memory techniques, can learn to use mnemonic strategies, particularly the method of loci, and significantly improve their memory; in fact a development of brain connections completely similar to that observed in subjects with special abilities is observed after a period of training [5].

The Ciceronian loci technique can be learned by anyone, young and old alike, and be used to improve performance. However, the use of mnemonic techniques appears to widen the memory differences between the young and the elderly, suggesting that maximum effectiveness is achieved by the young. This is also observed in some parts of the brain that are not activated in older people who did not experience benefits compared to others; however, this might also be due to the fact that older individuals may be reluctant to adopt new techniques and may have memorized the information in the way they deemed most appropriate, not following the given procedures; they may also have found it more difficult to apply the technique, having to imagine images representing parts of a narrative [33].

There are other memory techniques besides the method of loci, but this one seems to be the most effective, particularly in cases where the memorization involves not a random assortment of concepts but an ordered list. An advantage of this technique compared to others, such as the Link Method, is that recall relies on the locations themselves, rather than the strict sequence of the memorized information. This is particularly helpful because forgetting an element associated with a specific place along the imaginary path does not compromise the recall of the information that follows [39].

Tests have also been conducted by some students during study sessions to verify the technique's effectiveness in a practical setting; the results indicated that students who used the loci method in addition to traditional lessons showed a significant improvement in their performance on assessments compared to the group that used self-learning. Most of the students found the loci method useful and capable of improving both comprehension and recall [37].

1.3. Motivations of This Work

This study adopts the loci technique and modernizes it through the use of cutting-edge technologies. Building on previous research, our goal is to develop an application that enhances the method and brings it into the realm of mixed reality.

The technique is particularly well-suited for this transposition, as it enables the materialization of virtual content within the real world, thereby enhancing the sense of immersion. As a support for this statement, preliminary studies show that memorization through the loci technique with a headset in VR leads to better overall results compared to the same experience enjoyed from a display thanks to an immersive factor: some of these studies compared results obtained from the memorization of pictures distributed in different places [13], or tracked how memorization of a list of words could get better in the virtual reality condition [9]. However, these studies offer only preliminary indications and do not provide conclusive evidence. In fact, the authors themselves acknowledge several limitations, such as the use of a fixed and non-personalized virtual environment for all users, the lack of actual movement [13], and the fact that the memory palace in VR was often an empty setting without virtual images, which were not visualized through the headset but only imagined by the participants [9]. These constraints likely had a strong impact on the effectiveness of the method. Still, despite these limitations, the results suggest that the technique tends to perform better in virtual reality than on a standard display, an insight we build upon by extending the approach to mixed reality, thereby opening a new path for more immersive and effective solutions.

We have decided to explore this field because, even if there is some academic research on the topic, the studies rely on custom-made applications developed specifically for the purposes of the experiments and are significantly limited compared to what we have implemented.

In addition, a few commercial applications, such as *Mind Palace*, *Memory Palace*, or *Librarium*, exist and are available on the Meta Store, but they are currently outdated and non-functional; because of this they will not be taken into account. Overall, the commercial landscape for applications based on this technique is essentially empty, especially considering mixed reality.

This research intends to be useful especially in educational field, in fact it is important to note that new technologies, such as AR, could increase students' motivation and willingness to learn; some research paid attention to this factor and found a potential positive outcome, even if further studying is needed [11, 17, 41]. Considering educational field as a potential target, this could make students discover the method of loci in an exciting way.

The thesis is done with the technical collaboration of FifthIngenium, a company focused on mixed reality, who provided necessary equipment and support.

2. State of the Art

In this section, we take a broad perspective and review how the method of loci has been transposed into Extended Reality (XR), that is, any kind of virtual environment, in order to identify the key approaches and the gaps our work aims to fill. Specifically, we present the state of the art of academic studies exploring the method of loci in XR, based primarily on custom-made applications and, in rare cases, on third-party ones. A review of the main scientific literature databases reveals that only around a dozen case studies have been published on this topic, indicating that it is an emerging but still embryonic line of research.

It is important to first clarify the terminology related to extended reality in order to better understand the current work:

- AR (augmented reality) means the possibility to view the real world with an overlay of digital elements
- MR (mixed reality) means the possibility to view the real world with an overlay of digital elements which can interact with real objects
- VR (virtual reality) is a fully immersive digital environment

As will be seen in the following sections, studies on augmented reality and virtual reality will be analyzed separately. This distinction is made because research on the loci technique within VR environments is notably more mature, both in terms of technological development and user experience.

Although some studies reference mixed reality, they do not support interactions between physical and virtual elements and, therefore, cannot be considered true MR applications [10, 43]. To date, no existing research has effectively adopted a true mixed reality approach.

2.1. Current research

Current research focused only on the most traditional and pure version of the method of loci, without exploiting complex functionalities available with new hardware or adding something valuable for the target user of the application. However, currently different approaches are used.
To better understand how the method of loci is being explored in extended reality, we now examine a selection of published studies and two past master's theses on the topic.

2.1.1 VR

- **Objectives**
  The common focus of these studies is the creation of Virtual Memory Palaces (VMPs), aiming to enhance the classic method of loci technique and optimize its efficiency. Specific research has different methods to reach the goal: some studies try to improve memory palace design and increase immersion to enhance memory recall performance [31]; others focus on user attitudes towards using VMP in everyday situations [8]; other studies in VR have proposed the use of virtual reconstructions of real-world environments, obtained through dedicated scanning technologies, based on the premise that the loci technique is most effective when applied in familiar settings, thus, paradoxically, aiming to recreate a mixed reality scenario within a virtual context [40]. Another study explored whether a large-scale walking experience offers advantages over a stationary one for memorization, using a huge place in order to walk physically in VR without the risk of bumping into something [32]. Some studies even focus on specific use cases, such as the memorization of scientific papers [44]. Finally, some research compares the effectiveness of virtual memory palaces with traditional 2D screen-based interfaces [13, 9].
- **User Experience Design**
  Studies present various activities for the loci technique in VR: one of them simply presents a memory palace with already placed images throughout the selected virtual environment [13]; in other cases, the user actively places elements to be memorized within the virtual environment using a controller or hand tracking [31, 8, 40, 32, 44]. Tasks may include selecting the most impactful image among several options to represent a given word [31], leaving some kind of personalization; searching for 3D models from online repositories such as Google Poly [40]; or physically moving content within the space, whether user-selected or pre-supplied, in the latter case 3D objects [32] or even abstracts of scientific papers [44] are used. An alternative approach involves the user following a specific path and revealing, one at a time, mnemonic elements positioned at fixed loci by focusing their gaze and pressing a button on certain icons [8]. Navigation is carried out through tracked physical walking [31, 32] or with a controller [8, 40]. Obviously the main concept of the loci technique is then used by all studies, as said before the user is expected to walk through the environments, associating each location with the content to be memorized. Regarding the personalization of the experience, users interact with predefined content [8, 32, 44, 13], select the chosen content from predefined sets [31], or search for and import external content [40] from Google Poly, which, however, was discontinued prior to the time of writing this thesis.
- **Evaluation and Results**
  Studies about VMP agree that this technique leads to better results compared to the normal one [31, 44], and in particular it can help learning the method in a faster way [31], though further studies are needed to confirm this conclusively. Anyway, the obtained results can vary and the perceived immersion of the experience could be a really important factor, united with the perceived usefulness [8]. The possibility of physically walking around increases the immersion and the concentration, but other studies are needed to confirm that this factor increases the efficiency of the technique [32], the benefits of using familiar environments also need to be further investigated [40].
- **Devices**
  As for VR different hardware was used to conduct the experiments; some utilized specific virtual reality headsets such as Oculus Rift S [31], Meta Quest 2 [32], HTC Vive (2017) [44] or Oculus Quest [40], while others employed smartphone-based solutions [8]. Some studies also used VR to simulate MR [40].

2.1.2 AR

- **Objectives**
  Several studies have explored how to adapt the loci technique to augmented reality, each with a specific objective. One line of research investigated alternatives to printed conceptual maps [38]. A complementary study aimed to spark discussion around the use of AR in education by giving students greater initiative and control [16, 15]. Another design-science study introduced an AR-based system to support vocabulary acquisition in language learning contexts [4]. Finally, some studies explored the integration of collaborative

mechanics into AR-based implementations of the method of loci [10, 43].

- **User Experience Design**
  Some studies enable the merging of the real and virtual worlds by using a graph containing data automatically extracted from academic publications [38], enabling a virtual projection on the ground and allowing users to physically walk on it instead of read from a paper. Others implement the technique through various content, such as short texts that appear near printed markers positioned in the real world [43]. Some of these studies also include gamification elements, featuring a simple game that involves collaboration among multiple users to reconstruct a complete knowledge by combining information distributed among these distinct elements [10]. Other research focuses on the educational context, exploring how AR applications with content generated on the spot by students through three-dimensional drawings can be implemented, especially considering history related topics [16, 15]; this was achieved not through a custom-developed application, but by using Google's "Just a Line" app, which allows the user to draw on the screen while viewing augmented reality with the camera. Specific applications aim to facilitate learning the vocabulary of a second language, replacing traditional methods and allowing the free distribution of certain notes along a chosen route [4].
- **Evaluation and Results**
  Some studies present only the experimental framework without reporting empirical outcomes, instead, they plan future trials to assess spatial recall differences [38], though they already highlight the potential of the approach. Other exploratory pilots gathered qualitative feedback from small groups (without providing a final evaluation), showing that learners were able to build and navigate AR memory palaces, thus indicating the method's feasibility [16]. In another study, participants found the AR interface significantly more engaging than traditional methods. Although some recall-test data were collected, the authors chose not to draw conclusions about its effectiveness, focusing instead on insights related to the user experience [4]. Preliminary trials showed higher effectiveness using a game-like approach [10]. Finally, one study revealed a variety of content placement behaviors, including some that were unexpected [43].
- **Devices**
  Regarding AR, this technology is used on various hardware platforms, either through dedicated AR headsets, such as Microsoft HoloLens/HoloLens 2 [38, 10, 43], which can offer a better experience, or on mobile devices like smartphones or tablets running AR applications thanks to their built-in camera [16, 4, 15].

2.1.3 Development Tools

From a development perspective, many studies used Unity [4, 43, 31, 8, 40, 32]. For augmented reality development, SDKs like ARCore were used [4], or Vuforia for marker recognition [43]. Regarding high-level interaction frameworks, one study used Microsoft's Mixed Reality Toolkit (MRTK) [40], while another employed Unity's XRToolkit library [32].

2.2. Review and Critical Reflections

From the analysis of the literature, it emerges that applications developed for the method of loci in virtual reality seem generally more advanced and elaborate compared to those designed for augmented reality.
VR solutions have indeed explored the construction of virtual memory palaces, with some attempts to simulate mixed reality through the integration of scans of real, familiar environments within the virtual world. This allows the user to move within a virtual space that physically corresponds to the real environment [40]. Moreover, studies in VR tend to include a greater degree of interaction with virtual content and sometimes also offer experience customization, allowing the user to choose between different content options or to search for objects in online catalogs [31, 40].
However, despite their level of sophistication, VR approaches also present intrinsic limitations, especially in attempts to replicate the familiarity and details of the real environment. Scanning environments, for example, is often a complex and laborious process, and the resulting 3D reconstructions rarely achieve the visual quality and detail of the physical real environment. Furthermore, physical movement freedom in VR experiences is often limited: in most cases it is mediated by controllers [8, 40] or constrained to large, obstacle-free spaces, or requires the use of additional sensors [31, 32], making the experience impractical in real-world contexts outside of research experiments.
In contrast, the currently available AR applications of the method of loci are generally simpler, often based on predefined content and with limited interactions. Some solutions allow users to draw virtual elements via smartphone [16, 15], but these are mostly generic applications not specifically designed for the method of loci, and thus limited in terms of functionality. Other applications, which merely project floating graphs on the floor

[38], do not fully adhere to the fundamental principles of the method of loci: they lack real anchoring to objects in the physical environment and do not define a structured, retraceable mental path, an essential element for the method's effectiveness. Furthermore, immersion, which is essential for cognitive engagement, tends to be weak in many current augmented reality experiences, especially if mediated by a mobile screen [16, 4, 15] and if interaction possibilities are limited.

Despite these limitations in current projects, some studies have highlighted how augmented reality can offer significant advantages over virtual reality in terms of spatial learning. In particular, experiments based on simulations of the Corsi block-tapping test, which evaluates visuospatial memory, have shown that interaction through real walking significantly improves the memorization of 3D content distributed in space compared to other modalities, such as grabbing or teleportation. This effect was particularly pronounced in AR, where participants achieved better results than those immersed in virtual environments [7].

Moreover, the method of loci in augmented reality fully leverages the surrounding familiar environment, which is a crucial aspect of the traditional technique.

As for mixed reality, as mentioned earlier, there is a notable absence of a truly systematic exploration of the method of loci in authentic contexts where interactive digital elements meaningfully merge with the user's real environment.

Delving deeper into the analysis, some recurring criticalities and design gaps emerge, both in the VR and AR domains, which limit the effectiveness of current extended reality applications of the method of loci:

- **Experience personalization**
  Although some VR applications offer customization options, this aspect remains underdeveloped. The use of external catalogs [40], such as the discontinued Google Poly, also exposes users to risks related to dependence on third-party services. The method of loci, on the other hand, requires a high level of personalization, including the ability to freely choose and place content at significant points in the real environment [34].
- **Content persistence**
  In AR, managing persistence between sessions represents one of the main technical challenges. Some studies have proposed the use of cloud anchors to fix virtual objects to specific positions in the real world [4], but these solutions are often temporary (for example, with a 24-hour duration in the case of Google technology) and implemented mainly on mobile devices, with no counterpart on dedicated headsets. This also entails a spatial constraint, as no study to date has explored the possibility of managing multiple memory palaces distributed across different environments. In VR, persistence is rarely addressed explicitly, given the entirely virtual nature of the environments.
- **Interaction and physical engagement**
  In both VR, but especially in AR, physical interactions with content are extremely limited. Often, the interaction is reduced to placing objects in space without further possibilities for manipulation or active exploration. However, embodied learning theory emphasizes the importance of bodily engagement in the learning process: learning is more effective when it involves actions, gestures, and physical interactions with the concepts to be assimilated. Recent studies confirm the effectiveness of this approach [18, 47]; however, it is still poorly implemented in existing XR applications, which, while suitable, rarely take full advantage of its benefits.
- **Advanced features (gamification and sharing)**
  Features such as gamification, although initially explored [10], are now scarcely present. Yet it is known that playful elements increase motivation and engagement, leading to better learning outcomes, particularly among younger users [46, 14]. The possibility of sharing memory palaces with other users (e.g., between teacher and students) would also be particularly useful in educational and collaborative contexts, but it is still non-existent.

All these criticalities clearly highlight the need for further research and development to design more advanced and effective XR applications in the domain of the method of loci.

3. Application Design

In this section our developed mixed reality application is presented.

3.1. Vision and Goals

Our work, as previously mentioned, aims to combine the Ciceronian method of loci with mixed reality. In this way, we intend to preserve the effectiveness of a methodology that has been used for centuries, while updating and enhancing it through modern technologies. To date, there is no mixed reality approach specifically dedicated to memory palaces: we aim to fill this gap by creating an innovative solution that addresses this need.

The vision of the project is to enhance the memorization of concepts within the context of formal education, specifically for at-home review of school subjects.
To achieve this vision, we defined the following key goals:

- **Personalization**
  The application should allow users to freely choose the concepts to be memorized from any source, ensuring adaptability across various school subjects and enabling a tailored learning experience. This also allows the users to create unlimited memory palaces.
- **Content persistence**
  The system must ensure content remains available over time, allowing learners to pause and resume their memorization sessions across multiple uses. In this way, memory palaces are continuously saved and it is not necessary to build them from scratch every time.
- **Gamification**
  The application should integrate basic gamification elements to boost user motivation and engagement throughout the memorization process. This is essential to maintain a high level of student concentration.
- **Collaboration and sharing**
  The system should support content sharing and comparison of results, fostering peer interaction and community-based learning. This could be particularly useful in classroom settings, where the teacher can create a shared experience for all students.
- **Ease of use**
  The user interface and overall interaction should be simple and intuitive, ensuring accessibility for all students.
- **Learner autonomy**
  The system must support self-directed learning, allowing students to engage with the content independently, without the need for continuous teacher supervision.

3.2. Functionality Overview

This section provides an in-depth description of all the functionalities offered by the application.
Please note that some functionalities rely on the Tinalp platform developed by FifthIngenium, which serves as the foundation for the application. The focus of this thesis was the development of the application itself, not the underlying platform.

3.2.1 Preparation of the Experience

The experience starts with the choice of the materials, our application supports 3D models, audios, videos and images. The possibility given to the user of deciding with which material he can build his memory palace is extremely important as the personalization factor plays a central role for the loci technique. To guarantee the maximum flexibility and make the application usable even for people with poor technical knowledge the application automatically converts this content, when loaded in the mixed reality, to a standard size (approximately 40cm, keeping their original size ratios) and adjust some technical aspects of 3D models.
After the choice of content, the user has to upload it through FifthIngenium's portal "Tinalp", on which he had to be previously registered. Tinalp revolutionizes education and training by introducing the concept of augmented classes that simulate practical scenarios. It also could be used for real-time teacher-student interactions as well as for a multi-user, shared, personalized and engaging learning experience. Tinalp includes a no-code web editor designed to be accessible to everyone. It allows users to create decks, which are containers of data for a single experience. Within a deck, users can upload their chosen content and define the type of experience they want to create, such as the loci activity. They can also specify for the loci activity the order of uploaded content, that will be needed afterwards. Decks can also be shared among users, enabling multiple people to participate in the same experience.
For example, a professor can prepare the content for a specific activity (a deck on Tinalp), and each student can then build their own memory palace using that shared material (deck), without having to search for it again; the first step described about the uploading of the content is not needed for this case.
Once the content is uploaded and a deck is created in Tinalp, the user can start using the headset.

3.2.2 Experience Startup

Once the program is opened, the user should log in through the Tinalp dialog box that appears (if he has not done yet previously) and then he has to choose one experience among the listed ones. These experiences correspond to the decks created before with the web platform.

The application relies on the spatial setup performed by the headset, so in order to function correctly, the room must have been scanned beforehand. If the environment has not been scanned, the application displays a message informing the user that a scan of the current environment is required. Otherwise, if the setup was done correctly, the application starts correctly and a welcome message appears.
Immediately after the start of the program a table with a button on it spawns on the ground in proximity of the user.

3.2.3 Core of the Experience

The menu opened by pressing the button on the table is the command center for the experience, the user using it can access various functionalities:

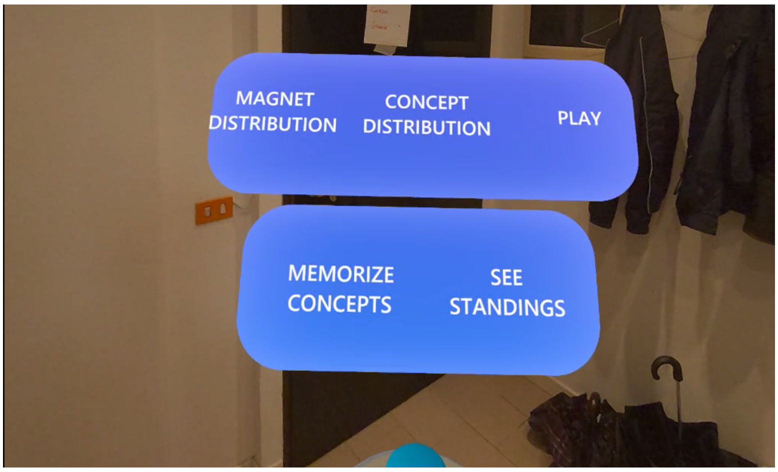


Figure 1: The experience command center

- **Magnet Distribution**
  This is the first phase of the application, when it is selected a virtual magnet spawns floating on the table. The user has to pick up the magnet, with hands or with the controller, and place it in a chosen position of the room. The place where the magnet is positioned represents a locus of the memory palace, so it should be put near a relevant real object that can be recalled from memory when an imaginary walk through the memory palace is done, for instance a television or a window.
  The user should imagine that he will move in a memory path organized with the selected loci.
  After the first magnet is placed sufficiently distant from the table, other magnets, one at a time, will spawn and the user has to complete the same procedure. The number of magnets to be spawned is calculated automatically and it's equal to the number of uploaded content.
  The *Magnet Distribution* phase can be chosen from any other phase in any moment but depending on the current phase the behavior is a little different: if the user is already in the *Magnet Distribution* phase then it will be reset, while if he is in another phase the other content beyond magnets will be deleted and the user will find himself in *Magnet Distribution* phase with magnets already positioned.
  Only in this phase it's possible to move around magnets.

Figure 2: A magnet is being positioned: the door is chosen as a locus of the memory palace

- **Concept Distribution**
  This represents the second phase of the application. In this stage, magnets act as the loci of the memory palace, while concepts are the pieces of information to be memorized.
  These concepts are formed by the contents, which, as previously mentioned, can be of various types and are uploaded by the user beforehand. Videos play when the user is near them and he is turned towards them.
  The concepts appear one at a time on the table, and the user must move each one and attach it to a selected magnet; when a concept is placed it will automatically face the user.
  When a concept spawns on the table, if it is a 3D model, it is flanked by four buttons (up, down, right, left) that allow the user to rotate the object in the corresponding direction. This feature was implemented to ensure that the model faces the user when attached to a magnet. This orientation is also saved for future sessions.
  This process is repeated until all magnets are occupied. Once completed, the memory palace is ready. The user can then walk around the room and observe the concepts and their positions. When it's time to recall the information, the user should mentally retrace the same path taken during the placement phase: by visualizing the real objects positioned in the room, the associated concepts will be triggered in memory. The order in which the content is memorized is also important, so the path should be followed in the same sequence.
  This phase can be accessed from the *Magnet Distribution* phase only once all magnets have been placed. Additionally, if the user selects the *Concept Distribution* phase while already in it, the phase will reset, and they will start again from the first concept. Otherwise, the system will interrupt any ongoing activity and bring the user back to the *Concept Distribution* phase, with magnets and concepts already in their previously assigned positions.

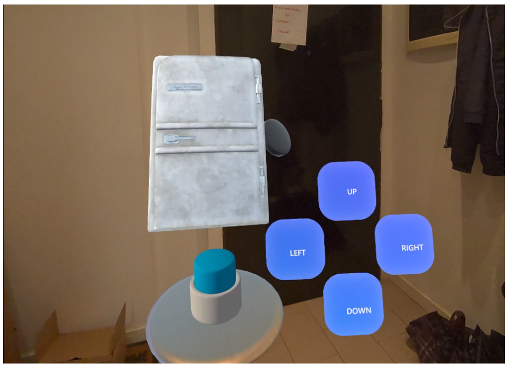


Figure 3: The command center used for rotating 3D concepts

Figure 4: A concept is being positioned near a magnet: when it is released, it is automatically moved and attached to that magnet

- **Play**
  This phase is designed for experiences with a large amount of content. Since the goal is to memorize a path made up of information, we created a mini-game to support this process by repeatedly associating concepts with magnets, which represent specific positions where a concept can be placed (also called "loci").
  The system detaches the concepts from the magnets and places them on the ground near the user. The user must then associate each concept to the correct magnet, the concept once placed will face directly the user. When the association is correct, a positive animation is shown; otherwise, an error animation appears and the concept is repositioned on the ground. Points are awarded for correct matches and deducted for incorrect ones.
  The difficulty level adapts dynamically based on the user's performance. Initially, the system detaches one concept at a time, following the original order selected in Tinalp. As the user improves, the application begins to detach multiple concepts simultaneously. After reaching a certain threshold, it detaches one concept at a time again, but in a random order, and then proceeds, as long as the user is doing well, with multiple concepts out of order. If the user continues making mistakes, the difficulty decreases, and points are lost. As the game progresses, correct associations yield more points. To encourage quicker responses, if a concept remains on the ground for too long, it begins to rotate, signaling that the user is losing points and should act quickly. This is done to prevent leaving some concepts whose position is not remembered and the user aims at not losing points.
  This repetition-based method helps reinforce the memory of the loci: for example, in a room with 20

concepts, memorizing the full path takes time. Having this active approach from the user leads to a better memorization compared to an observation one.
The game as illustrated supports memorization of concepts independently of their order, in fact the strength of the loci technique is that each concept is remembered thanks to its position in space, not because of its sequence, so forgetting a concept doesn't compromise the rest of the memory.
After a certain amount of time, calculated based on the number of magnets and their relative distance, the final phase begins. In this timed mode, the user has a limited window to repeat all associations one last time. Only one magnet is shown at a time, and the user must select the correct concept from the ground and attach it. This last match is done with the order chosen by the user while creating the experience.
When the experience ends, a dialog appears showing the final score and asking the user whether to publish it to the leaderboard.
After the game, the user can return to the table and choose between *Magnet Distribution*, *Concept Distribution*, viewing the leaderboard, or replaying the experience.
The leaderboard is intended to introduce a competitive element. For example, students in the same class can compare their results within the same experience, motivating them to memorize their memory palaces more effectively.

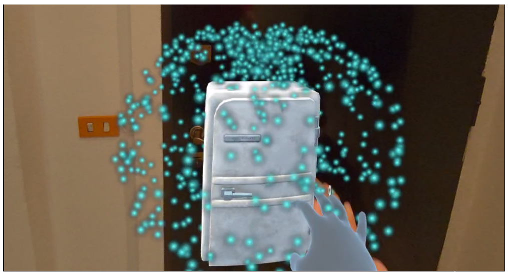


Figure 5: A concept is correctly associated

- **Memorize Concepts**
  This function is only used for more comfort when the user wants to a better immersion, only thing that it does it to hide magnets. This is useful for images or videos, otherwise the magnet can ruin the visual. This can be accessed only from a completed memorization phase.
- **See standings**
  This function can be called at any time, except during the *playing* phase. A message box will appear in front of the user, displaying the leaderboard for the current experience. As previously mentioned, an experience corresponds to a deck created in Tinalp. Since decks can be shared among users, multiple participants can take part in the same experience, and a shared leaderboard is generated and saved on FifthIngenium's server.

3.2.4 Additional information

During all phases of the experience, if the user goes outside from the room in which the memory palace is located, a warning is issued through a message that appears in front of the user.
After selecting a certain phase a message giving welcome and explaining what to do next or telling that it was impossible to select that phase. In addition to this, when a new phase is available to the user he will be notified.
The application features an essential auto-save system. Throughout all phases, except the *playing* phase, the positions of magnets and concepts are continuously saved in space. This allows the user to resume their memory palace without needing to rebuild it each time. When the user starts the application in a specific room and selects an experience though the Tinalp message, if that experience has already been used in that room, the application will automatically restore the previous state, either in the *Magnet Distribution* or *Concept Distribution* phase, depending on the last configuration. The saves are linked to a certain Tinalp account, so if another user utilizes a different account he won't find the memory palace already built from other players. We chose not to save the state of the *playing* phase, as it represents an activity that doesn't benefit from being stored.
It's important to note that experiences (whose content comes from decks created in Tinalp) are tied to the physical room in which they are used. At start, the headset determines its location to load any previously saved content associated with that space. For this reason, the same experience with the same content, can be recreated in different rooms with entirely different configurations of magnets and concepts for the memory palace. For example, if a user previously created the experience "'29 Crisis" in their kitchen, and later wants to use the same content in their bedroom, they will need to start from the *Magnet Distribution* phase to create a new memory palace in that space.

In addition to all cited features, during the execution of the application there is a background sound called binaural beats. Binaural beats are auditory perceptions generated by presenting two slightly different frequencies to each ear. It is hypothesized that they can influence brain activity [12, 1, 6]. Regarding attention, some studies suggest that gamma frequencies (around 40 Hz) may enhance it [1]. As for long-term memory, one study found that binaural beats in the beta range (20 Hz) improved recall and recognition, while those in the theta range (5 Hz) worsened them [6]. Other analyses, which base their work on previous studies, report promising results and suggest a positive effect of binaural beat entrainment on cognitive functions, particularly memory and attention. However, not all studies analyzed agree with these findings, highlighting the need for further investigation [1]. In conclusion, research suggests a potential effect of binaural beats on attention and long-term memory, but results are mixed and depend on various factors, such as the specific frequency used. Therefore, further studies are needed to clarify their effectiveness and the optimal conditions for their application [12, 1, 6].

3.3. Summarizing Functionalities

This section provides a more concise and formal overview of the functionalities of the loci application. The goals and requirements that precisely define the application are presented and serve as a summary of its main functionalities.
Here goals are presented again:

- G1 – **Personalization**
- G2 – **Content persistence**
- G3 – **Gamification**
- G4 – **Collaboration and sharing**
- G5 – **Ease of use**
- G6 – **Learner autonomy**

Table 1: Synthesis of functional requirements and related goals of the application

Code	Requirement	Goals
FR-01	User account management via FifthIngenium server through the Tinalp platform	G2, G6
FR-02	Upload of content (3D, audio, video, images) via Tinalp assigning an order to them	G1
FR-03	Hosting and retrieval of data via Tinalp API	G2
FR-04	Ability to use shared decks on Tinalp for sharing the same experience	G4, G6
FR-05	Automatic conversion of content to a standard size when loaded in the program, approximately 40cm keeping the aspect ratio	G5
FR-06	Login on startup via Tinalp dialog	G2, G6
FR-07	Selection of experience (deck) from the post-login list	G1, G6
FR-08	Verification of room scan by Meta Quest 3/3S; warning and blocking or welcome message depending on the outcome	G5
FR-09	Warning if the user leaves the scanned area during the experience	G5
FR-10	Spawn of the command table on successful startup with a button to open main menu	G5
FR-11	Menu: *Magnet Distribution*, *Concept Distribution*, *Play*, *See standings*, *Memorize Concepts*	G5
FR-12	Selecting *Magnet Distribution* a message notifies the user and then it spawns one magnet at a time on the table	G5
FR-13	Automatic calculation of the number of magnets = number of uploaded contents	G1, G5
FR-14	Picking up and free placement of magnets during *Magnet Distribution*	G1, G6
FR-15	Reset of *Magnet Distribution* if already in that phase; otherwise switch to it preserving magnets	G5
FR-16	Selecting *Concept Distribution* one concept at a time spawns on the table. If the concept is a 3D model, the user should use the rotation buttons (up, down, left, right) to orient it so that it faces them	G1, G5
FR-17	Moving and attaching concepts to magnets (oriented toward the user)	G1, G6
FR-18	If a concept is attached to a non-empy magnet, the already attached concept is moved to the ground near the user	G5
FR-19	Access to *Concept Distribution* only after all magnets have been placed	G5
FR-20	Reset of *Concept Distribution* if already in that phase; otherwise switch to it preservig magnets and concepts	G5
FR-21	Access to *Play* mode only after all concepts have been associated	G3, G5
FR-22	Detachment of concepts and placement on the ground for *Play* mode	G3
FR-23	Concept–magnet association with visual feedback (blue animation for right answer, red for wrong) and scoring in *Play* mode	G3, G5, G6
FR-24	Adaptive difficulty (number, order) in *Play* mode	G3
FR-25	Automatic rotation of inactive concepts to signal timeout during *Play* mode	G3
FR-26	Final timed one-by-one session in the original order at the end of *Play* mode	G3
FR-27	End-of-experience dialog with score and option to publish to leaderboard, warning message if publishing couldn't succeed	G3, G4
FR-28	*Memorize Concepts*: hide magnets for immersion (only after all concepts have been associated)	G5
FR-29	*See standings*: display the leaderboard associated with the Tinalp deck (outside *Play* mode)	G3, G4
FR-30	Continuous auto-save of magnets and concepts via spatial anchors (outside *Play* mode)	G2
FR-31	Automatic restoration of the last phase on startup in a known room and used deck	G2
FR-32	Binaural beats audio playing throughout the experience	G3
FR-33	A message appears in front of the user when he has completed a phase and he can access another one	G5
FR-34	A message appears in front of the user explaining the current phase when accessing it	G5
FR-35	Sounds associated to magnets and concepts	G3, G5

3.4. Use Case Scenarios

Here are presented two use cases of the application:

- **Use Case 1**

Actor: First-year high-school student

Objective: Memorize the Group 1 of the periodic table using the MR memory-palace application

Preconditions: Valid Tinalp account (FR-01)
Deck created in Tinalp with ordered 3D models, images and videos (FR-02, FR-04)
Bedroom scanned by Meta Quest 3/3S (FR-08)

Main Flow:
1. Student dons the Quest headset and launches the memory-palace app
2. Tinalp login dialog appears; student enters credentials and logs in (FR-06)
3. Student selects the “Group 1 Elements” deck from the post-login list (FR-03, FR-07)
4. App verifies the room scan; no warning since scan exists (FR-08)
5. Command table spawns near the student (FR-10)
6. Student presses the table button to open the main menu (FR-10, FR-11)
7. Only *Magnet Distribution* and *See standings* are enabled (FR-11, FR-19, FR-21)
8. Student selects *Magnet Distribution* (FR-12)
9. First magnet spawns on the table; student picks it up and places it by the door (FR-12, FR-14)
10. Additional magnets spawn one at a time; student positions each next to a real-world object until all N magnets are placed. N is th number of uploaded content (FR-13, FR-14)
11. Student attempts to switch to *Concept Distribution* prematurely → app blocks action with a warning (FR-19)
12. After all magnets are placed, the student is notified and then he selects *Concept Distribution* (FR-16, FR-33)
13. Concepts appear one by one on the table with the rotation command if it is a 3D model; student attaches each concept to one magnet (FR-05, FR-16, FR-17)
14. On one occasion, a concept is dropped on an occupied magnet; app swaps them automatically (FR-18)
15. Student selects *Play* after been notified of the *Concept Distribution*'s end (FR-21, FR-33)
16. Concepts detach and lie on the floor; student picks each and re-associates it with its correct magnet, receiving visual feedback and scoring (FR-22, FR-23)
17. System adapts difficulty based on performance (number, order) (FR-24)
18. If a concept remains on the floor too long, it rotates to signal timeout (FR-25)
19. Final timed one-by-one session begins in the original order (FR-26)
20. End-of-experience dialog appears with final score and option to publish to leaderboard; student confirms (FR-27)
21. The user goes back to the table and select *Magnet Distribution* (FR-12)
22. All concepts disappear but magnets remain in their position (FR-15)

Alternative Flows:
A1: Missing Scan If the room has not been scanned, the app blocks startup and displays “Please scan your environment” (FR-08)
A2: Publish score failed The score couldn't be published for network problems (FR-27)

Post-condition:
- The experience can be resumed from the last saved phase (FR-30, FR-31)

- **Use Case 2**

Actor: Last-year high-school student

Objective: Resume a paused memory-palace setup and complete the ‘29 Crisis’ experience using the MR application

Preconditions: Valid Tinalp account and deck created with ordered content by the professor (FR-01, FR-02, FR-04)
Bedroom and kitchen environments scanned by Meta Quest 3/3S (FR-08)
Initial *Magnet Distribution* in bedroom started but incomplete

Main Flow:
1. Student launches the app in the kitchen
2. Tinalp login dialog appears; student logs in (FR-06)
3. Student selects the ‘29 Crisis’ deck (FR-03, FR-07)
4. App verifies kitchen scan; succeeds (FR-08)
5. Command table spawns (FR-10)

6. No magnets appear (no saved state for this room) (FR-31)
7. Student closes the app and moves to the bedroom
8. Student relaunches app in the bedroom and logs in again (FR-03, FR-06, FR-07)
9. App verifies bedroom scan; succeeds (FR-08)
10. Command table spawns and previously placed magnets reappear, including one over the table (FR-10, FR-11, FR-31)
11. Student picks up and places the magnet over the table in the next location (FR-13, FR-14)
12. After placing it, app displays "*Magnet Distribution* complete" (FR-33)
13. Student opens menu and selects *See standings* (FR-29)
14. Student returns to menu and selects *Concept Distribution* (FR-16)
15. Concepts appear one by one on the table with the rotation command if it is a 3D model; student attaches each concept to one magnet (FR-05, FR-16, FR-17)
16. When all concepts are placed, app displays *Concept Distribution* complete (FR-33)
17. Student selects *Memorize Concepts* to hide magnets (FR-28)
18. Student walks around the room, mentally rehearsing each concept's position
19. Satisfied with memorization, student closes the application (FR-30)

Alternative Flows: A1: Missing Scan If the current room isn't scanned, app blocks startup with "Please scan your environment" (FR-08)

Post-conditions:
- The memory palace for the '29 Crisis' deck is fully configured in the bedroom
- All phase data and standings are saved and can be resumed later (FR-30, FR-31)

4. Technical Implementation

The program, developed with Unity and C#, is created specifically for Meta Quest 3/3S: this approach ensures the possibility to use all functionalities thought for the headset. In order to reach desired objectives, the application combines Unity default functionalities with MRTK (Mixed Reality Toolkit) 2 powered by Microsoft, which is cross-platform, and various specific Meta SDKs.
MRTK is mainly used as the tool which has to handle all kinds of spatial interactions with the virtual objects placed in real world through the headset or with some pre-made blocks, such as buttons, sliders, or more sophisticated creations [3].
Meta SDKs enable a lot of possibilities and new functions, such as spatial anchors, scene model and the passthrough, which gives us mixed reality.
Note that also interaction can be handled by one Meta SDK, but in this project it was decided to use MRTK due to FIfthIngenium's standards [21].
In summary, Unity's XR Plug-in Management system acts as the central hub that manages XR backend so the project can communicate with a headset's native drivers and APIs. By selecting the Oculus XR Plug-in as the Plug-in Provider for Meta Quest 3/3S, all requests for tracking data, passthrough video, spatial anchors and scene model information are routed through Meta's native runtime, whereas choosing OpenXR or another vendor's plug-in would expose a different (or more generic and limited) feature set. Built on these data streams, MRTK serves as a high-level framework to manage spatial interactions and the user interface.

4.1. Anchors and Scene

In previous sections a brief introduction about spatial anchors and scene model was done while discussing other functionalities, but here a more in-depth description is done.
In order to use some of the following functionalities the final user should give certain privacy permissions at the first start of the application and the developer should asks rights for these permissions [25, 28].
These are relatively new features, and they are in constant update; for this reason some things could be changed from the period in which this thesis was done.

4.1.1 Spatial anchors

A spatial anchor is a reference that saves virtual objects' position and orientation in the real world, their biggest advantage is given by the fact that these information can be saved across different sessions. One can implement a system where an object is associated with an anchor and it is placed on a table and each time the user starts the application, he finds the object where he has left it [27].

This is extremely important for this work because after an experience is created in a specific room with specific concepts placed in certain positions, the user does not have to do the preparation phase each time (which can be very time consuming if there are a lot of concepts), but the application will load automatically all the contents in respective positions.
Spatial anchors are managed by the OVRSpatialAnchor component, each anchor is identified by a unique UUID generated during its creation.
As mentioned before, one can do various operation with anchors, it's important to note that almost every anchor method is asynchronous, so it's a must to pay a lot of attention using them. Here a syntetic list of possibilities :

- **creation**: it's only needed to add OVRSpatialAnchor component to an object, then it will create a spatial anchor automatically
- **saving**: this function enables the anchor to be persisted across sessions. The UUID of anchor should be saved in a file in order to retrieve it later
- **loading**: This is done in three steps:
  - load unbound anchors with a given UUID, an unbound anchor is an anchor not associated to OVRSpatialAnchor component
  - localize the anchor in order to get its position and rotation in the real world, and use them as a reference for a GameObject position
  - bind the loaded unbound spatial anchor with a new OVRSpatialAnchor component added to the prior GameObject
- **erasing**: it's clearly possible to delete a saved anchor from the memory
- **destroying**: one can destroy anchor's component without erasing it from the memory. It's a different action: deleting the instance or the saving
- **sharing**: anchors can be shared among users in order to let them see same virtual objects, anyway this function is not used in this thesis, so it's only listed here for completeness but no further details will be written

[25].
There are functions for saving and erasing one anchor at a time or a collection of them; furthermore note that there is an hard limit about the maximum number of anchors loaded or erased by one call of their method, this limit is respective 50 and 32; it's developer's duty to bypass the limitation, for instance creating a list of lists with the maximum size and calling in sequence the method.
Meta only offers methods to use these functionalities, it's left to the user creating a data structure for managing references of anchors and the connection to their respective GameObjects. It's suggested to use one anchor per GameObject for best accuracy, or at least to use one anchor for objects placed in a maximum radiuos of 3 meters from the anchor.
Objects with a OVRSpatialAnchor component can't be moved, it's left to developer to bypass this limitation [26].
Note also that anchor methods are asynchronous and require careful attention, especially considering that they only work in the built application running on the headset, which makes debugging more challenging.

4.1.2 Scene model

Interacting with the sorrounding environment can be a game changer experience, so it's really important for the application to know what's happening around us; to achieve this goal Meta give us the MRUK (Mixed Reality Utility Kit).
The scene model is a representation of the physical world that can be queried in order to obtain some information; one for instance can know the position in the room of the sofa, if there is any. This model of the room is saved and managed by the headset, and it's available for every program that request it. Since the scene model is program-independent, it's obvious to understand that it's created by a headset setting, specifically by space setup, which is a procedure that let the user scan the environment.
Meta Quest 3/3S can save up to 15 different rooms/spaces after V62 update released starting from Febraury 2024 [29].
The scene model is loaded in the application thanks to MRUK singleton class. Once it's loaded, the developer can also use various functions based on the room in which the user stands, thanks to MRUKRoom class [22].
MRUK also offers the possibility of uploading some default rooms from prefabs instead requesting it to the headset: this is particularly useful for debugging purposes on the pc.
Until now, we have talked about spatial anchors, which are used for saving positions and rotations in a cross-session way; but there is more. While developers create spatial anchors, scene anchors are created automatically with space setup and they constitute the scene: users can't create them but only query them for obtaining various information. Scene anchors are made with MRUKAnchor class [22]. Scene anchors can have different labels, called semantic classification, representing different objects; here the list:

- FLOOR

- CEILING
- WALL_FACE
- TABLE
- COUCH
- DOOR_FRAME
- WINDOW_FRAME
- OTHER
- STORAGE
- BED
- SCREEN
- MAP
- PLANT
- WALL_ART
- GLOBAL_MESH
- INVISIBLE_WALL_FACE

Meta Quest 3/3S are able to recognize automatically and categorize all of these objects, but it's left to the user the possibility to mark not-recognized items and assign to them one of these labels or to correct some errors present in the reconstructed scene [30].
It's important to note a special kind of scene anchor: scene mesh, which describes the boundary between free and occupied space in a room; but keep in mind that scene mesh is an approximation and should be used knowing its limitations regarding accuracy in delimiting objects [24].

4.1.3 OVRAnchor

Under both scene and spatial anchor there is the OVRAnchor struct, which is a light interface for accessing anchors [23]. Anchors can be seen as entities univocally identified with a UUID that hold various components; these can be activated or not, depending on the type of the anchor. Here the available components:

- Locatable: once enabled, an application can continually query the pose information of the anchor
- RoomLayout: contains references to anchors that make up the walls, the ceiling and the floor. Only room anchor has this component.
- AnchorContainer: contains a reference to a list of child anchors
- Bounded2D: provide information about the dimensions of an anchor
- Bounded3D: provide information about the dimensions of an anchor
- TriangleMesh: provides an indexed triangle mesh for an anchor
- Boundary2D: provides access to the polygon outline of an anchor, commonly found on floor anchors
- SemanticClassification: useful for categorize the anchor
- Storable: available only for spatial anchors
- Sharable: available only for spatial anchors

Given the fact that a developer could want to access anchors' information, it's possible to fetch all anchors with a given UUID or a given component [20].

4.2. How the Code Works

This section provides a technical analysis of the application's code.
To avoid unnecessary detail, a comprehensive description of every method or attribute is not included (for instance, calls to play particle effects, to play or prepare audio, and to enable or disable scripts are omitted. Note that this is only an example, other aspects are not covered). Instead, it offers a clear overview of the main aspects, detailing how different classes interconnect together and explaining key elements, without covering all details. Important functions will only be explained the first time they are mentioned.
Note also that for a better understanding of what is happening the section 3.2 should have be read.
My code is tightly integrated with FifthIngenium's one to leverage its existing functionalities as described previously, however, for privacy reasons, the internal workings of code not developed by myself are not described here in a depth way but they are only mentioned in order to understand better the context. Every time a class was already developed it's explicitly said next to its name.
The following convention is adopted for clarity: class names are presented in bold text, attributes and methods names in italics. Methods will have parenthesis to identify them, without necessarily specifying all parameters. When *ClassName var.method()* appears, it indicates that an instance of the class invokes that method. In the case of a "Manager" class, the controller instantiates it only once. Thus, if *ClassManager.method()* is written, it is understood not as a static call but as a call on that single instance. Consequently, throughout this thesis, references to any "Manager" class implicitly denote its instance and it won't be clarified each time.

4.2.1 Preparation of the Experience

Main class from which the experience starts is **LociActivityController**, this represents the experience's controller (the experience in this context is the activity to be performed), whose role is to instantiate and initialize all necessary scripts and classes.

This controller inherits from **BaseActivityController**, a FifthIngenium abstract class used as the parent for all specific controllers. It is primarily responsible for launching the controller itself once everything is prepared, at which point the actual experience on the headset can begin.

The controller also holds a reference to **FIActivityContent**, a FifthIngenium class containing specific definitions of possible activities: **FILociActivity** is one such definition, inheriting from the class just mentioned. This class is linked to data uploaded to Tinalp; for instance, it contains *List<int> ListOfMediaToDownload* (IDs of media to be downloaded), *Dictionary<string,int> standings* (the leaderboard for a given experience), *List<string> order* (the IDs of concepts in the user's chosen order), and *Dictionary<string,int> concepts* (mapping each concept ID to a media ID).

When the user logs in to Tinalp and selects a deck, the associated experience is set up: the information in **FILociActivity** is bound to that deck and it is available for other functions.

For performance reasons, we have also decided to download from the server all the content before starting the controller and to instantiate all the objects immediately, keeping them disabled until they have to be spawned. When in this text a concept is said to be spawned, it is meant that it's activated and moved to the desired position.

ActivityRoomController is a FifthIngenium class responsible for downloading and instantiating concepts at startup; we have modified it in order to include the data for the loci experience. In particular here *concepts* is iterated and each time a media with the given ID is found, after analyzing the type of the content associated to it (3D model, audio, video or image), the connected file is downloaded on the device from the server using *UnityWebRequest.Get()*. **UnityWebRequest** offers the possibility to access downloaded bytes with *UnityWebRequest var.downloadHandler.data* and saving them in a desired local path with *File.WriteAllBytes()*.

After the download happened, **ObjectManager** has to instantiate the content in the scene and disabling it as explained before. This is done for performance reasons, in this way from now on objects are already prepared before the start of the experience and they only need to be moved and enabled/disabled when necessary. There are four possible directions depending on the type of the content to be instantiated:

- Retrieve an image texture using *UnityWebRequestTexture.GetTexture(url)*, *DownloadHandlerTexture.GetContent()* and then spawn a textured quad via *ObjectManager.InstantiateImage()*
- Reconstruct and instantiate a 3D model from its byte data with **GLTFast**, using *ObjectManager.InstantiateObject3D()*
- Assign a video URL directly to a *VideoPlayer* component by calling *ObjectManager.InstantiateVideo()* and placing the video on a quad
- Retrieve an audio from an URL by *UnityWebRequestMultimedia.GetAudioClip()* and *DownloadHandlerAudioClip.GetContent()* in *ObjectManager.InstantiateAudio()* and then spawn a speaker model that can play the audio

Spawned GameObjects are saved in *Dictionary<string, GameObject> conceptsToMove* of **ObjectManager**, which will be used later when these concepts will have to be enabled and repositioned. These methods from **ObjectManager** class create GameObjects with a standard size while preserving their original scale factor. The *InstantiateObject3D()* method also calculates a BoxCollider based on the full object bounds, taking into account children at all levels. It is also worth noting that in this case, the actual 3D object is set as a child of an empty parent object. This is done to control the rotation of the 3D model during the *ConceptDistribution* phase. The chosen approach involves modifying the *localRotation* of the child (the true 3D model), allowing the empty parent's negative Z axis to face the user. As a result, the object appears correctly oriented towards the user. Additionally, these functions assign **ObjectManipulator** and **NearInteractionGrabbable** (**MRTK** classes for managing interaction with virtual content), **SpatialAnchorsManipulation** and **ConceptController** scripts to the spawned GameObject.

ConceptController saves concepts' ID and manages video/audio reproduction; these assets are only played when the user is nearby and directly viewing them.

SpatialAnchorManipulation, on the other hand, provides various functions for managing objects' movements.

4.2.2 Experience Startup

After this procedure, **LociActivityController** is called. It waits for **MRUK** to load the current scene information by using *MRUK.Instance.RegisterSceneLoadedCallback()*. Once this callback is invoked, the controller calls **RoomManager**, which stores a reference to the previously scanned room where the user is located and calculates its area for later use. If the user isn't in a recognized room a message appears (messages are managed with **Dialog** of **MRTK**) and the experience is basically terminated notifying the user; otherwise, the controller loads a JSON from which past experiences can be reconstructed, if it is present. Savings are stored in *Dictionary<string,ExperienceData> experiences*, where **ExperienceData** is a class consisting of two properties:

- *Dictionary<string, List<float»* used to store the rotation of 3D concepts as chosen by the user through the buttons when the concept is initially spawned. The key represents the concept's ID, while the *List<float>* contains the **Quaternion** data used to reconstruct the object's rotation
- *Dictionary<System.Guid, AnchorType>* used to manage and save spatial anchors; *System.Guid* represents the ID of a specific anchor. **AnchorType** contains an *id* and **ObjectType**, an enum that identifies possible objects in the scene (e.g., magnets, concepts, or the table; **ObjectType** is assigned as a tag to objects). The *id* is present only for concepts and is required to identify that specific concept

The controller also calculates the ID of the current experience using the room ID, the user and the deck ID retrieved from Tinalp. If the current experience is new, it is added to *experiences*. From *experiences* information about the rotations and the anchors are obtained and *anchorDictionary* is created; this dictionary is of type *Dictionary<string, Dictionary<System.Guid, AnchorType»*, where the outer key is the experience key. *anchorDictionary* and *concepts3DRotation*, together with other data, are used for initializing the instance of **SpatialAnchorManager**.

This class is used for managing spatial anchors, in its initialization *LoadAllAnchorsFromCurrentExperience()* is called to restore all virtual content previously placed in that specific experience. The structure on which **SpatialAnchorManager** relies the most is the already introduced *anchorDictionary*. In the cited method, *OVRSpatialAnchor.LoadUnboundAnchorsAsync()* manages to get all unbound anchors with given IDs, these are anchors previously saved by the Meta Quest that are not connected to any object in the current Unity scene; the list of IDs to load is in *anchorDictionary[currentExperienceKey].Keys*. After unbound anchors are loaded, their pose in the real world are found using *OVRSpatialAnchor.UnboundAnchor var.TryGetPose()*. Then, thanks to *anchorDictionary*, the type of object associated with the anchor is identified. If it's a concept, it is moved and enabled (since it was already downloaded and instantiated, as mentioned earlier, and its ID can be retrieved from **AnchorType**) with the correct rotation obtained from *concepts3DRotation*; otherwise, it is instantiated. In this last case, the content is stored in Unity as a prefab. This approach was chosen because instantiation is not computationally intensive, especially considering the average number of objects that will be spawned. In addition to this, when an object is loaded in the scene, its reference is added to *conceptsInScene*, *magnetsInScene* or assigned to *table*. Subsequently, an **OVRSpatialAnchor** component is added to the new GameObject and the unbound loaded anchor is bound to this new one through *OVRSpatialAnchor.UnboundAnchor var.BindTo()*; the anchor is also added to *allAnchorsEnabled*, which keeps track of anchors whose objects are currently rendered. Note that *LoadUnboundAnchorsAsync()* can handle to 50 anchors, so a solution using multiple temporary list is used in case the limit is exceeded.

Once all the content is restored in its previous position (this is known checking when *allAnchorsEnabled* is equal to *anchorDictionary[currentKeyExperience].Keys.Count*), the controller loads the leaderboard associated to the experience downloading it from the server and then initialize **LociManager**, which from now it becomes the manager class for the loci experience.

Init() is the method used to start the instance of **LociManager**, this function first prepares the background audio formed of binaural beats and the particle systems that need to be played during the *Play* mode; for performance reasons only two particle systems are created (one for right answers and one for wrong ones) and the empty GameObject to which they are attached is dynamically moved to the right position when needed.

After this step, *PreparePhase()* is called. This method reconstructs the saved content: first, the numbers of magnets and concepts are obtained from *conceptsInScene* and *magnetsInScene*; then, the missing magnets or concepts are computed by knowing the total elements via *ObjectManager.GetConceptsCount()*, which uses the list of concept IDs passed at startup by Tinalp in the procedure described earlier. Based on the placed content, the current phase is determined between *MagnetDistribution* and *ConceptDistribution*; a list of available phases is also calculated, in which *PlayingMain* may be added if all concepts are already attached to magnets.

Here a clarification is due: until this moment, and particular in 3.2, possible phases are said to be *Magnet Distribution*, *Concept Distribution* and *Play*. This is true from a functional point of view, while from a code perspective the available phases are *MagnetDistribution*, *ConceptDistribution*, *PlayingMain*, *PlayingFinal* and *Ended* (the latter three are associated with the functional *Play* mode), as well as *Memorize* (associated with *Memorize Concepts*).

At this point, two different paths are possible:

- **New Configuration**
  If no magnets or concepts are loaded, a table is instantiated near the user, replacing any existing one if present. Several attempts are made to place it in front of the user within a certain radius; if that fails, it is placed behind, and if that also fails, it is placed at the user's position.
  A button with the **TableButton** script is placed and initialized on the table, its sole purpose is to call *LociManager.TableButtonPressed()*, which positions the buttons menu over the table.
  In addition, before instantiating the table, the position is checked using *MRUKRoom var.IsPositionInRoom()* and *RoomManager.IsBoundInRoom()* to ensure that all vertices of the table are within the room.
  It is also verified that the table's bounds do not intersect with the bounds of any scene anchors, otherwise the table could be positioned inside a real world object, making it impossible to touch.
  In order to save the virtual table in the real world, an anchor has to be created; *SpatialAnchorManager.AddAnchorToObject(GameObject)* is called. This function first checks that the object does not already have an associated anchor; if not, it assigns an **OVRSpatialAnchor** component and then calls *SetupAnchorAsync*. The anchor state is first checked using *OVRSpatialAnchor var.WhenLocalizedAsync()*, and if the state is valid, the anchor is saved on the Meta Quest 3/3S through *OVRSpatialAnchor var.SaveAnchorAsync()*. If the operation is successful, the anchor is added to *allAnchorsEnabled*, and a new entry is created in *anchorDictionary[currentExperienceKey]*. Subsequently, the *Dictionary<string, ExperienceData>* in the controller is updated with the new information and saved as a JSON file in memory. As illustrated, this saving process is essential for retrieving anchor information in the next session.
- **Existing Configuration**
  If there are some magnets or concepts in the scene, *LoadPreviousData* is called. This method first initializes the scripts for magnets, concepts and the table. Then, *RestoreMagnetData()* is called. This function reconstructs *magnetsData*, which is a *Dictionary<GameObject, MagnetData>*. This dictionary is the main data structure used by **LociManager**, as it contains all the essential information. It does not need to be saved because it is reconstructed during setup, as described here. The key of the dictionary is a GameObject instance of a magnet, while the value is a **MagnetData** type. This class is formed by various fields:
  - *bool isOutSideTableSpace* is used for marking when the magnet is far from the table
  - *GameObject associatedConcept* is a reference to the associated concept placed during *ConceptDistribution*
  - *float timeIsFree* indicates for how long during the playing phase the magnet is separated from its concept
  - *bool associatedConceptIsPicked* is used for knowing when the associated concept is picked up
  - *GameObject attachedConcept* is a reference, used in *Play* mode, to the currently attached concept. This differs from *associatedConcept* when the magnet is free or the attached concept is not the right one

  At this point, *RestoreMagnetData()* checks whether only magnets were placed or if concepts were also present. If only magnets were placed, it adds them to *magnetsData* and sets *isOutsideTableSpace* by calculating distances. Otherwise, if there are concepts, it means that the magnets were already moved away from the table to proceed to the next phase. In this case, it calls *ReassociateConceptsToMagnets()* to calculate the nearest concept to each magnet and associate it accordingly, adding these information to *magnetsData*.

After this iteration is finished *Init()* of **LociManager** initializes the menu which will spawn over the table when its button is pressed and the buttons used for rotating a 3D concept: it instantiates the prefabs and it creates a dictionary saving a reference to each button, adding a listener that connects a function to each of them. *Init()* also calls *LociUpdate*, a **Coroutine** that have to check whether the user exits the scanned room.
The setup phase is now complete, and the available functionalities depend on the current phase of the application. The possible phases in which the user can be are *MagnetDistribution*, *ConceptDistribution*, *PlayingMain*, *PlayingFinal*, *Memorize*, and *Ended*; these are defined by the **Phase** enum.

4.2.3 Core of the Experience

As previously mentioned, the user can press the button positioned on the table, which spawns a menu with several buttons: *Magnet Distribution*, *Concept Distribution*, *Play*, *See Standings*, and *Memorize Concepts*.
Each button object, as already illustrated, has a listener with a specific function that is triggered when the button is pressed; this mechanism is used to change the current phase of the application or to activate specific functionalities.

- **Magnet Distribution**
  When this button in the menu is pressed, *MagnetButtonPressed()* is called. This function checks whether the *MagnetDistribution* phase is included in *availablePhase* (which can contain *MagnetDistribution*, *ConceptDistribution* and *PlayingMain*); if the check passes, it then calls *ChangePhase(Phase nextPhase)*.
  This method, given the next phase and the current one, determines the appropriate actions to perform.
  - If the current phase is *ConceptDistribution* or *Memorize*,
    SpatialAnchorManager.EraseAndDestroyAnchorObjectListCurrentExperience(concepts) is called.
    This method first saves a list of the UUIDs of the objects' anchors and removes these anchors from *allAnchorsEnabled*. Since the maximum number of anchors that can be deleted from memory at once is 32, if the total number exceeds this limit, the list is split into smaller sublists, each containing at most 32 elements. Each list (or sublist) is then passed, one at a time, to *OVRSpatialAnchor.EraseAnchorsAsync()*, a method that erases anchors from memory. After the anchors are successfully deleted, they are removed from *anchorDictionary*, and *SaveAnchorDictionary()* is called to update and save the experience file. Finally, each GameObject is disabled and its anchor is destroyed. *conceptsInScene* is cleared and the current phase is set to *MagnetDistribution*, *RestoreMagnetData()* is called and *SpawnMagnetIfPossible()* is invoked: it checks if *leftMagnetToSpawn* is positive and *AreAllMagnetsOutsideTableSpace()* is true, in that case it then spawns a magnet over the table, decreases *leftMagnetToSpawn* and add value to *magnetsData*. In the end, if all magnets are far enough from the table and *leftMangnetToSpawn* is 0, *ConceptDistribution* is added to *availablePhase*.
  - If the current phase is *MagnetDistribution*, *magnetsInScene* is passed to *EraseAndDestroyAnchorObjectListCurrentExperience()* and the list of magnets is cleared. The current phase is then set to *MagnetDistribution*, *availablePhase* is cleared of other phases, and *RestoreMagnetData()* and *SpawnMagnetIfPossible()* are called.
  - If the current phase is *PlayingMain*, *PlayingFinal*, or *Ended*, any existing coroutines used during these phases are stopped. Magnets and the table are destroyed, while concepts are disabled and *conceptsInScene* is cleared. Then, *FromPlayingToMagnetDistribution()* is launched and concepts are erased from memory with *EraseConceptsAnchorFromMemoryCurrentExperience()*, which finds in *anchorDictionary* concepts' anchors and erase them with the already seen procedure (note also that at this time objects in scene don't have an anchor component, it's just in memory); when this method has terminated (its async **Task** has finished),
    SpatialAnchorManager.LoadAllAnchorsFromCurrentExperience is called for restoring all magnets and the table with the anchor component. Once anchors are fully loaded, magnets and the table are prepared, current phase is set and *RestoreMagnetData()* is executed. *ConceptDistribution* is added to the available phases, if *AreAllMagnetsOutsideTableSpace()* is true and *leftMagnetsToSpawn* is zero.

  During *Magnet Distribution*, as already explained, the user has to place magnets in the real environment. *SpatialAnchorManipulation* script is attached to all magnets, this class has two main methods: *ManipulationStarted()* and *ManipulationEnded()*. These methods are called when the user picks up the object to which the script is attached or when they put it down. They are associated with magnets and concepts, and their main purpose is to manage spatial anchors: as mentioned multiple times, objects with spatial anchors cannot be moved. Therefore, to allow movement, in *ManipulationStarted()* it's first saved in *hasAnchor* whether the object has an anchor associated and then if the boolean is true, *SpatialAnchorManager.MovementStarted()* is called, which in turn invokes *EraseAndDestroyAnchorFromCurrentExperience()*. This method removes the anchor from *allAnchorsEnabled*, triggers *EraseAnchorFromCurrentExperience()* and destroy the anchor component attached to the GameObject. *EraseAnchorFromCurrentExperience()* calls *OVRSpatialAnchor var.EraseAnchorAsync()*, which deletes the saved anchor from the headset. If the operation is successful, the anchor is removed from *anchorDictionary*, which is then saved to reflect the new state.
  When the object is released, if it is not a concept, then *OVRSpatialAnchor.SetupAnchorAsync()* is called to save a newly added anchor component.
  Additionally, for magnets, *LociManager.MagnetMovedMagnetDistribution()* is invoked, based on the correct assumption that magnets can only be moved during this phase. In this method first it is determined whether or not the magnet is distant from the table, then, if possible, a new magnet is spawned and in the end *conceptDistribution* is added to *availablePhase* if *AreAllMagnetsOutsideTableSpace()* is true and *leftMagnetToSpawn* is zero; otherwise, that phase is removed if requirements are no longer met.
- **Concept Distribution**
  When this button in the menu is pressed, *ConceptButtonPressed()* is called. This method checks whether the *ConceptDistribution* phase is included in *availablePhase*; if so, it calls *ChangePhase(Phase nextPhase)*.
  This method, given the next phase and knowing the current one, performs different actions accordingly.
  - If the current phase is *MagnetDistribution* or *Memorize*, then the current phase is updated, *Re-*

storeMagnetData() and *SpawnConceptIfPossible()* are called. The latter method checks whether *CountFreeMagnet()* equals *leftConceptToSpawn* and whether *leftConceptToSpawn* is greater than zero; if both conditions are met, *ObjectManager.SpawnNextConcept()* is invoked: after identifying the first concept in the chosen order that is not yet in the scene, it is enabled and positioned. Furthermore, if the concept is a 3D model, the menu for rotating it is enabled facing the user. If *AreAllConceptsAssociated()* returns true and *leftConceptsToSpawn* is zero, then *PlayingMain* is added to *availablePhase*.

- If the current phase is already *ConceptDistribution*, then concepts' reference are cleared and their anchors era erased as already introduced, the new phase is set, *RestoreMagnetData()* and *SpawnConceptIfPossible()* are called, and *availablePhase* is updated accordingly.
- If the current phase is *PlayingMain*, *PlayingFinal* or *Ended* then existing coroutines used during the game phase are stopped, magnets and the table are destroyed while concepts are only disabled. After this, *SpatialAnchorManager.LoadAllAnchorsFromCurrentExperience()* is called in order to restore all contents with the anchor component attached to them. Then, *FromPlayingToConceptDistribution()* is launched: once the anchors are fully loaded, magnets and the table are prepared, *ConceptDistribution* is set as the current phase, *RestoreMagnetData()* is called, and *availablePhase* is updated based on the current state.

During this phase the user has to associate concepts to magnets. **SpatialAnchorManipulation** was already introduced, but in this case there are some differences. When a concept is released, *LociManager.ConceptMovedConceptDistribution()* is called. This method first checks whether the released concept was already associated with a magnet; if so, the *associatedConcept* property of that magnet is set to *null*. Then, the nearest magnet is searched for (within a limited distance), and two possible paths follow:

- If the nearest magnet does not exist, then a spatial anchor is added to the released concept.
- If a nearest magnet exists, the released concept is then associated with it. Moreover, the previously attached concept, if it existed, is moved to the floor using *MoveToRandomPositionOnFloor()*. Finally, the newly associated concept is moved to the magnet's position, and an anchor is attached to it using *SpatialAnchorManager.AddAnchorToObject()*. *SpawnConceptIfPossible()* is called. *MoveToRandomPositionOnFloor()* calls *SpatialAnchorManipulation var.MovementStarted()* and then attempts to find a free position near the user with the strategy illustrated for the positioning of the table. The found position is verified to be free within the room by calling *RoomManager.GetCurrentRoom().IsPositionInRoom()*, checking that it is not in the same position as other concepts or magnets, and that the bounds of the new position object do not interact with scene anchors, the table, or the room bounds. Finally, a spatial anchor is attached to the object.

At the end of the method, *availablePhase* is updated with the new state. *SpatialAnchorManipulation.ManipulationEnded()*, when a 3D concept is moved for the first time, also saves the controller's **Quaternion** by calling *SaveRotationData()*, which updates and stores the JSON file of the experience. The visibility of the rotation buttons is disabled in *ManipulationStarted()*.

- **Play**

When this button is pressed, if *availablePhase* contains *PlayingMain*, then *ChangePhase()* is called. Based on the current phase, different actions are done:

- If the current phase is *Memorize* or *ConceptDistribution*, then *PlayingMain* is set as the current phase. For each content in the scene, *SpatialAnchorManager.DestroyAnchor()* removes the object's anchor from *allAnchorsEnabled* and destroys the anchor component, without deleting the anchor from memory. This is done because, during playing mode, saving the objects' positions is unnecessary. After this step, *StartGame()* is called. This method calculates the time available for this phase, taking into consideration the average distance between magnets and their total number. It also resets the points from previous matches, if any, orders the list of magnets based on the sequence of concepts specified by the user in Tinalp and set *attachedConcept* equal to *associatedConcept* for all magnets. The first concept is moved to the ground with *MoveToRandomPositionOnFloor()*, and then *magnetsData* is updated with information about that magnet: *attachedConcept* is set to null. Two coroutines are then launched: *ToPlayingFinal()* and *ManageTimeFreeMagnets()*.
- If the current phase is *PlayingMain*, *PlayingFinal*, or *Ended*, then, after stopping any existing coroutines and setting the phase to *PlayingMain*, *magnetsData* is reverted. In particular, for each magnet, *associatedConceptIsPicked* is set to false, *timeIsFree* is reset to zero, and each concept is moved back to its corresponding magnet position. Finally, *StartGame()* is called.

ManageTimeFreeMagnets() is used to manage the *timeIsFree* property in *magnetsData*. If *magnetsData[magnet].GetAttachedConcept()* is null or *magnetsData[magnet].GetAttachedConcept()* is different from the *associatedConcept*, it means that the magnet does not have its concept attached, and the timer is increased. After some time, if the magnet is still not correctly associated, its concept begins to rotate (provided it is not picked up or associated elsewhere), and every three seconds, points are lost.

When a concept is moved and released during the playing phases, the **SpatialAnchorManipulation**

script attached to it calls *LociManager.ConceptMovedPlaying()*. This method updates the state of *magnetsData* setting the magnet associated to the concept picked up as free and searches for the nearest free magnet using *FindNearestMagnet()*, iterating through all available magnets. If a nearest magnet exists and is free, then *ConceptMovedPlayingMain()* is called. This method moves the released concept to the position of the nearest magnet, sets it as the attached concept, and takes different actions depending on whether the association is correct or incorrect.
  - If the association is correct, *timeIsFree* is reset. The score and the number of concepts to be detached are calculated as follows: first, the maximum number of magnets (*maxMagnets*) is determined as half the number of elements in *magnetsData*, rounded up. The correct answer streak (*correctStreak*) is then incremented. Next, the streak is adjusted (*adjustedStreak*) and used to compute how many magnets should be freed (*magnetsToBeFree*), limiting this number to the maximum available and setting the minimum to *adjustedStreak* / 4 + 1. A multiplier (*multiplier*) is then assigned: it is doubled if the correct streak divided by four is greater than or equal to the maximum number of magnets. The *score* is updated by adding the product of the magnets to be freed, a fixed value, and the *multiplier*. Finally, for each magnet to be freed, either *ConceptMovedPlayingMainSequential()* or *ConceptMovedPlayingMainRandom()* is called, depending on whether *correctStreak* / 4 is less than *maxMagnets*.
    ConceptMovedPlayingMainSequential() follows the previously described procedure for detaching a concept and placing it on the ground, also updating *magnetsData*. The variable *indexToFree* is used to indicate the next occupied magnet to be freed, based on the order selected by the user at the beginning.
    ConceptMovedPlayingMainRandom(), on the other hand, performs the same procedure but maintains a list of magnets already associated with concepts and randomly selects one to be detached. When the list becomes empty, it is recalculated.
  - If the association is incorrect, *score* and *correctStreak* are decreased, *DetachedMagnet()* is called. This method moves the magnet to the ground using *MoveToRandomPositionOnFloor()* and updates *magnetsData* to reflect the detachment.

When *gameTime* expires, *ToPlayingFinal()* is activated. This method starts another **Coroutine** which, once completed, stops the game and instantiates a dialog box asking the user whether to publish their score on the leaderboard. A listener is added to *Dialog var.OnClosed* to trigger an action based on the user's response.
While the final timer is running, *ChangePhase()* is called. This method iterates over each magnet, calls *MoveToRandomPositionOnFloor()*, updates *magnetsData* to reflect the current state, and disables all magnets except the first one.
When a concept is released during this phase, the method *ManipulationEnded()* triggers a sequence of calls that eventually invoke *ConceptMovedPlayingFinal()*, if a nearby magnet exists. This latter method updates *magnetsData* setting the released concept in *attachedConcept* property of the near magnet, moves the concept to the magnet's position, and determines the action based on the outcome.
  - If the outcome is correct, the next magnet is enabled, and if all concepts are correctly placed, the match ends by prompting the user to publish the score, multiplied by a certain factor.
  - If the outcome is incorrect, *DetachConcept()* is called, which handles the movement of the concept to the ground after some instants, also updating *magnetsData*.

- **Memorize Concepts**
  This button transitions to the *Memorize* phase if *availablePhase* contains both *ConceptDistribution* and *PlayingMain*. This phase is only used for disabling the visual display of magnets.
- **See standings**
  When the user presses this button, if they are not actively playing, a **Coroutine** executing *ManageStandingsMessage()* is launched. The standings data is converted into an ordered *List<KeyValuePair<string,int»* and subsequently into a *List<List<KeyValuePair<string,int»>*. This conversion is performed because each sublist represents a page within dialogs that are displayed to the user and can be navigated to see the leaderboard.

4.2.4 Additional Information

In addition to these main functions, the system includes some simple data structures and alternative execution paths within certain methods, which are activated for debugging purposes. Specifically, using spatial anchors with the Meta Quest simulator on a PC causes errors, which can interrupt the application's execution; thus, it is important to bypass these functions. In this work, the adopted solutions are not presented due to their simplicity and their exclusive use for debugging.
The same applies to another Unity scene that can be activated and allows the user to utilize its features. In this first experiment, various actions are available:

- place a simple object that is immediately saved in memory through an anchor
- deleting displayed objects keeping them in memory
- erasing all objects placed in current room from memory
- erasing all objects from memory
- loading all objects from memory which were placed in the current room
- moving objects with hand tracking

These functionalities utilize the same methods previously discussed for managing spatial anchors and others methods that will not be illustrated due to their strong resemblance to the ones already discussed. However, this scene is particularly important for testing, as it only contains anchor functions that are also present within the loci experience: this is useful in determining whether a problem is caused by anchor methods or other code related to the loci experience itself.

4.3. Charts

This section is dedicated to charts that summarize some aspects of the code analysis. Note that these are intended only as a quick reference and do not represent all the details.

4.3.1 Application Startup

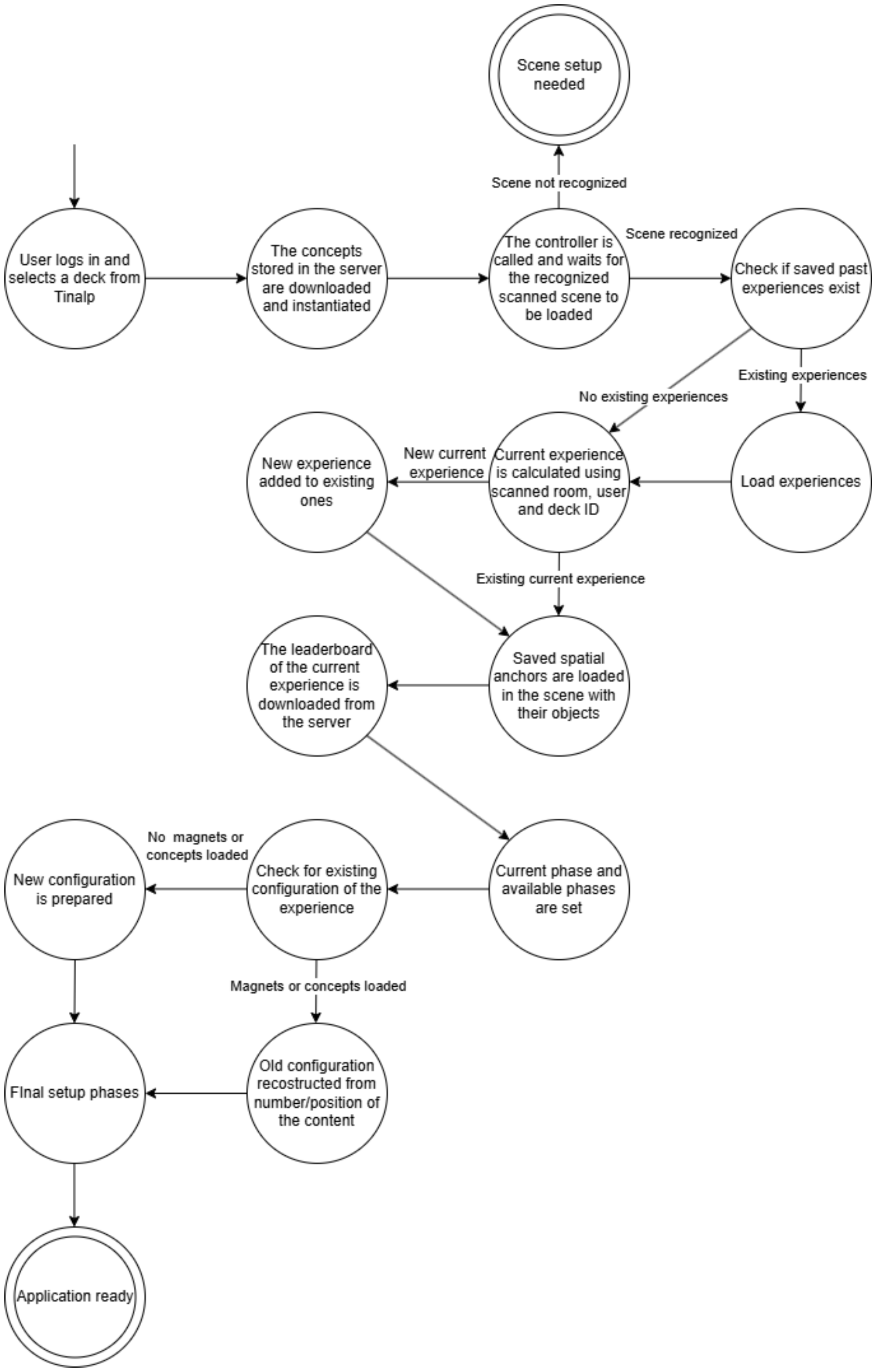


Figure 6: This chart represents the steps required for the application to be ready for use.

4.3.2 Phases' Change

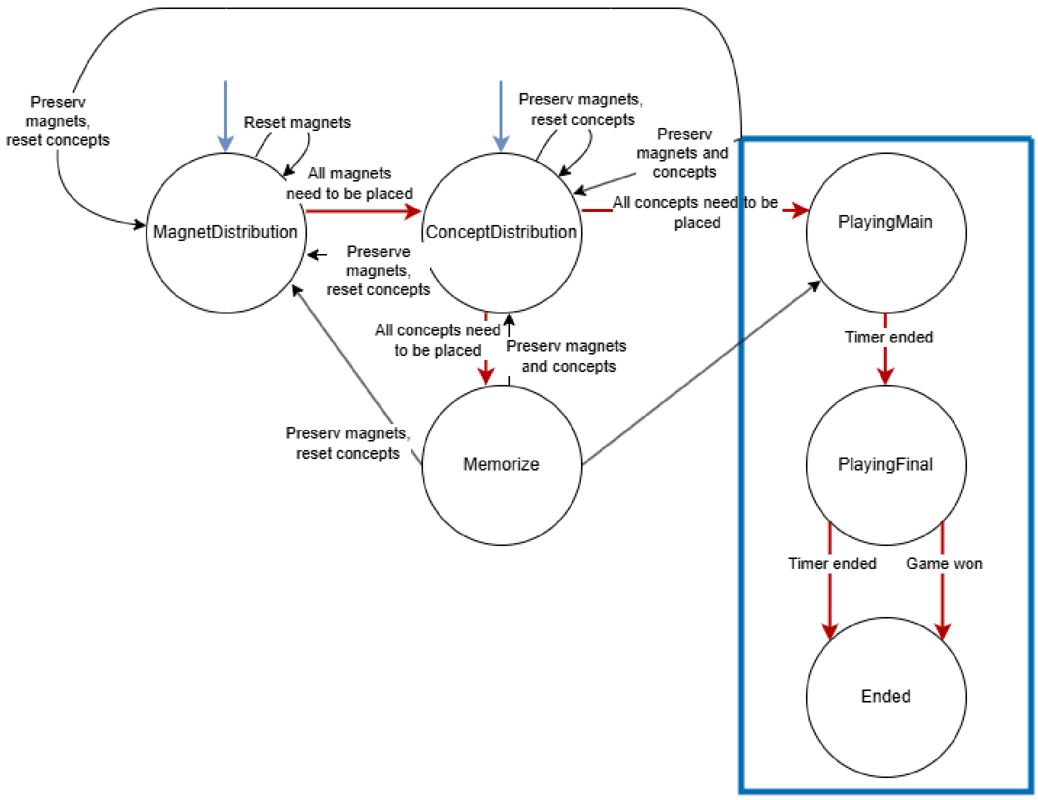


Figure 7: This chart illustrates how the various phases are interconnected. Note that these are code phases, and their relationship with the menu buttons was previously explained in Section 4.2. For completeness, it is also worth to remember that the leaderboard can be accessed in any phase except for the three represented in the blue box. Black arrows indicate actions that are always available between phases, while red arrows denote transitions that require specific conditions to be met. Additionally, the two arrows exiting from the blue box indicate that the corresponding actions are available from all the phases within the box.

5. Validation

This section presents the validation of the application, carried out in two rounds. The first round took place during the Open Day at Politecnico di Milano, while the second was conducted during a class session at the same university, involving a smaller group of students. In both cases, participants were asked to complete the same questionnaire. Additionally, in the second round, a focus group was also conducted. The evaluation focused on two main aspects: the usability of the application and its perceived usefulness.

5.1. Tested Memory Palace

The experience was designed to help memorize pages regarding the 1929 economic crisis from the high school history book *Tempi e Culture 3* by Alberto Mario Banti. The objective was to design a memory palace to help recall the chronological order of key historical events. Below follows an extreme synthetic representation of the content used in this experience and its meaning:

- a 3D fridge, representing the innovation and industrial expansion that took place at the beginning of the 1920s, when market demand for consumer goods was very high due to their novelty
- a photo of people attempting to withdraw money from a bank, representing the sudden crash of the financial markets
- a photo of a growing snowball, symbolizing how the crisis spread across all economic sectors and reached Europe
- a video of a Roosevelt speech, representing the New Deal that succeeded in resolving the crisis

- a 3D radio, symbolizing the widespread use of radio by Roosevelt to communicate his policies and that took him popularity

5.2. Open Day

The Open Day took place on March 29th, 2025, and it was the occasion for holding a testing session with as many participants as possible. In this occasion there was a little time for each user, less than 10 minutes, so we had to organize various aspects: the content of the experience, how to present our project to the public and what was important to ask after the test was done.
The main scope of this experiment was to test people's reactions and impressions in order to understand if an application of this type could potentially become a feasible and concrete method to help studying.
The first thing to do was to explore the possibility of using the Meta Quest 3 outside, which is not an expected use-case scenario; some difficulties were met during space setup as the device was not in a room delimited with walls.
The content for the experience was chosen carefully and described in previous section 5.1. We have decided to use this content keeping in mind that the vast majority of users during the open day would have been students of last two years of high school and a relatively simple topic, which probably they had studied, was needed in order to explain the meaning of our project and how it's possible to use it in a every-day situation. In particular, this was a right choice afterward because I noticed that a lot of participants actually knew the argument and it was faster for me to explain the experience: considering the little time available this was crucial.
To every person I illustrated the technique of loci: how it works and its well known benefit for improving memorization; the fact that most people were students could have been optimal because they could empathize better with the situation. After a first general introduction to the method, I exposed the functionalities available on the application, the chosen content and how that content could help to remember the '29 crisis, which I briefly introduced. This script was basically the same for every participant, 33 in total.
After the practical session I gave them a questionnaire about the overall experience, due to the little time available we decided to ask general questions and to focalize on first impressions.
People during the open day were extremely pleased to try such new technologies and asked various questions regarding the system's functionality and my experience at PoliMi.

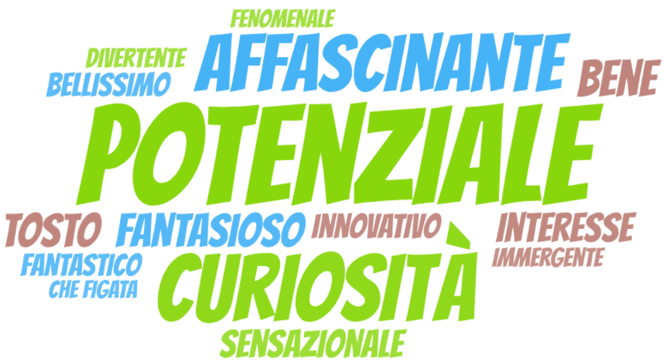


Figure 8: Here there is a list of thoughts written by some participants

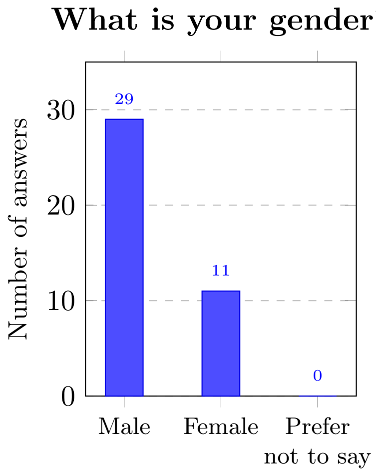
5.3. Class Test

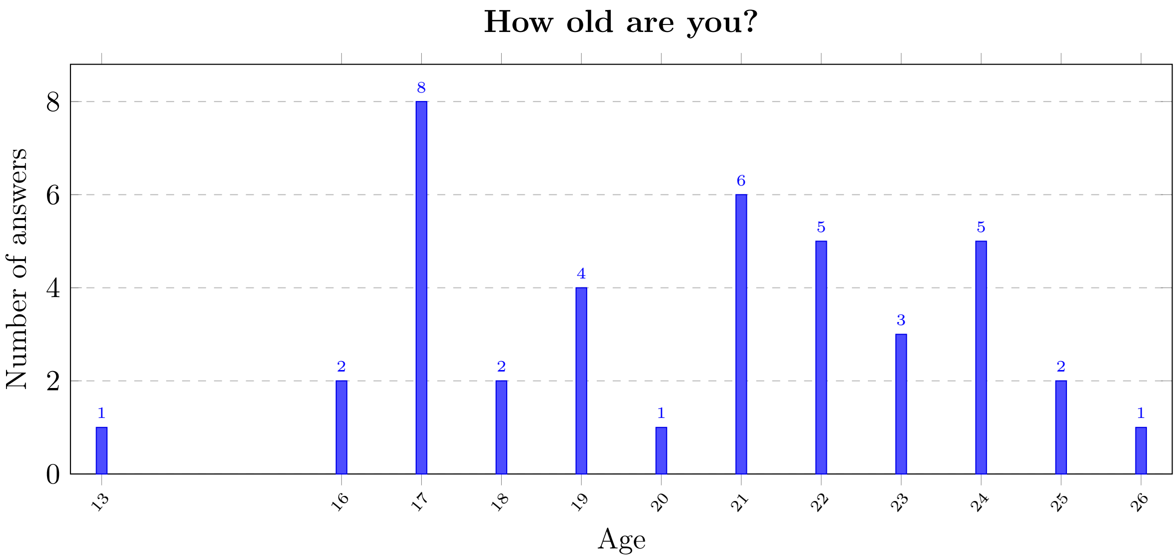
On May 9th, 2025, we had the opportunity to conduct another test session, this time more in-depth, with seven students following my professor's master class. The potential and future applications of mixed reality were explained by Matteo Valoriani, CEO of FifthIngenium. Subsequently, we held the testing session.
For this occasion, I reused the experience and the questionnaire created for the open day, as the time available for each participant was similarly limited.
After compiling the questionnaire, we had some available time, which we decided to utilize for gathering deeper impressions and suggestions regarding the application. This was not possible during the open day. Students expressed that the usability and their first impressions of the application were very positive, as it was easy to understand the application's phases and its functionality. The use of hands contributed to a greater sense of immersion, and the background binaural beats evoked positive sensations.
Students suggested that the table and the menu that spawns above it should adapt to the user's height. They also recommended that magnets should be placed at a greater distance from the table, as the content attached to them could cover the table. Additionally, one student indicated that magnets could benefit from an improved model. However, these aspects are minor points within the application and are easily resolvable.
Students were also asked regarding the overall usefulness of the experience and its practical applications. They responded that this application is suitable for everyday personal use with an educational purpose, and could also be used to compare and review students performance. From this perspective, the purpose of our application was completely understood by the students.

Figure 9: Here there is a list of thoughts written by some participants

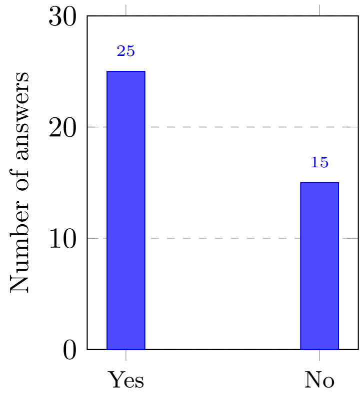
5.4. Overall Results

The questionnaire utilized for both occasions is presented below with the results. To obtain a comprehensive overview, and for brevity reasons, we decided to unify the results, as presenting a survey with only seven answers from the class test is not statistically significant. However, the data distribution was quite similar; the only noticeable difference pertained to age: participants during the open day were predominantly recent high school graduates or recent bachelor's degree holders, whereas during the professor's class, they were primarily in the 22-25 age range.

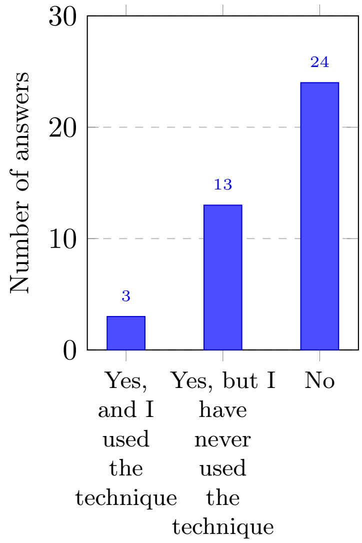




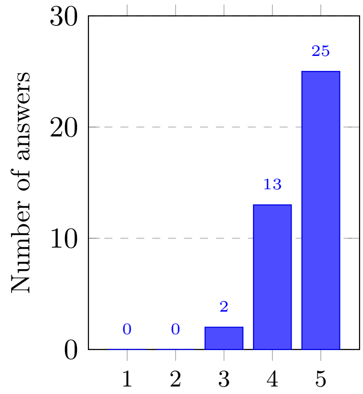
Have you ever used a mixed reality headset?



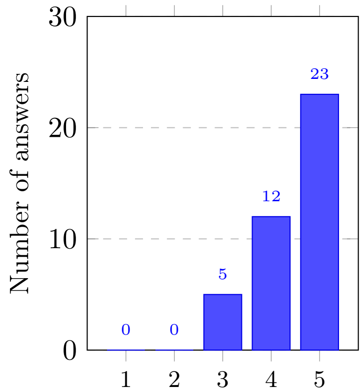
Have you ever heard of Ciceronian Loci?



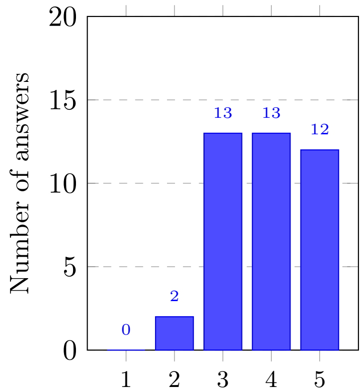
Was the experience comfortable (no nausea, good hand tracking and moving of virtual objects)? 5 represents a comfortable experience



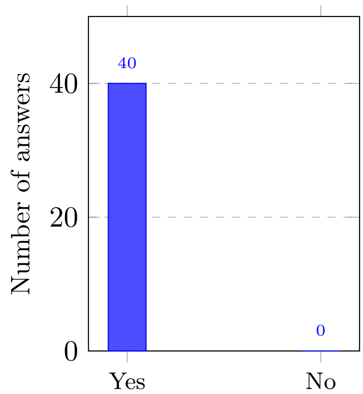
How easy was it for you to use the features of the application? 5 represents great simplicity.



Would you use the application at home for studying, if you had a headset? 5 stars represent a great willingness to use



In conclusion, have you found the experience positive?



Summarizing the results, it's evident that the experience received very good ratings. 63% of participants who tried the experience had also used a headset in the past; this could explain why the experience was perceived as comfortable, scoring 4.58/5. The application was also appreciated for its ease and immediacy of use: it scored 4.45/5, which is a very high score. Even though the willingness to use the application was 3.88/5, which might not seem like an extremely high score, it's important to consider that 60% of participants had never heard of the loci technique, and only 7.5% of the total had previously used the memory palace method to memorize anything. This percentage is extremely low, indicating how few people were already familiar with the technique and its potential. From this perspective, a 3.88/5 rating for the concrete possibility of using our application at home is considerably high: they understood the potential in a short time and considered using their own time at home for it. It's also worth highlighting that all 40 testers found the experience positive and worthy of note, often expressing interest in actually using it.

6. Final Considerations

In this section the final thoughts are presented together with the reflections on the possible work that can follow this research.

6.1. Future Directions

Based on the results obtained, future work should focus on assessing whether our application enhances the performance of traditional loci methods while simultaneously expanding its functionality by incorporating additional features and capabilities.

- **Performance Analysis**
  Future testing should be carried out to rigorously determine whether our application enhances memorization performance, a task that was not feasible within the timeframe and constraints of this thesis due to the need for a large, statistically reliable sample of participants.
  Additionally, given that only a small minority of our participants were already familiar with the loci technique, studies with a statistically significant group of users who regularly use the method of loci would also be informative; however, this too poses a significant challenge for a thesis project.
  Despite these limitations, existing literature consistently demonstrates that the loci method is both effective and widely utilized in practice.
- **Application Improvement**
  The application still has room for improvement, particularly when it comes to sharing memory palaces. Although users can already share the content of a created experience as described in this work, someone wishing to build their own palace from that shared content might also want to see exactly how the original creator arranged items within the real environment. One way to address this would be to add a VR-only functionality that allows users to virtually walk through the creator's memory palace, observing how the content is distributed in the real space. Since it is not currently possible to scan and share that environment directly within the headset, a more laborious workflow involving third-party tools must be used.
  Another aspect worth exploring is a true multiplayer mode, in which a memory palace could be experienced simultaneously by multiple users, along with a gamification approach tailored to a multi-user setting.

6.2. Conclusions

This work relies on mixed reality, which keeps the user in a familiar environment, the real world around him, while adding virtual elements that, in the classic loci method, would be entirely imaginary.
In our application, users can feel fully immersed: they can walk around freely and use their hands to interact with virtual content. This active interaction supports deeper learning compared to using only a simple visual input.
Another key feature is the system that saves and anchors virtual content in the real world. Thanks to this, a user can build one or more experiences across different rooms and then reload them at any time, without having to rebuild the "memory palace" from scratch.
Everything is fully customizable: the user picks and uploads the elements they want, creating an unlimited number of memory palaces for different experiences.
It is also possible to share these experiences with other users.
We added a gamification approach to keep motivation high: users can challenge themselves or compete with others, always trying to improve their memorization through ongoing engagement. This makes the learning process more enjoyable and encourages users to revisit their memory palaces in order to get better results.
The tests we conducted confirmed that our approach was well received and met our goals.
In conclusion, combining the proven effectiveness of the method of loci with the positive feedback from testing sessions and the fact that all objectives set were achieved, we believe our application can serve as a strong foundation for further research and help spread this powerful memorization method more widely.

References

[1] Sandhya Basu and Bidisha Banerjee. "Potential of binaural beats intervention for improving memory and attention: insights from meta-analysis and systematic review". In: *Springer* (2022).

[2] Cicerone. *De Oratore*. 55-54 b.C.

[3] Microsoft Corporation. *MRTK 2*. 2024. URL: `https://learn.microsoft.com/it-it/windows/mixed-reality/mrtk-unity/mrtk2/?view=mrtkunity-2022-05` (visited on 01/21/2025).

[4] Alexia Larchen Costuchen, José María Font Fernández, and Minos Stavroulakis. "AR-Supported Mind Palace for L2 Vocabulary Recall". In: *International Journal of Emerging Technologies in Learning* (2022).

[5] Martin Dresler et al. "Mnemonic Training Reshapes Brain Networks to Support Superior Memory". In: *Cell Press* (2017).

[6] Miguel Garcia-Argibay, Miguel A. Santed, and José M. Reales. "Binaural auditory beats affect long-term memory". In: *Springer* (2017).

[7] Dongyun Han and Isaac Cho. "Evaluating 3D User Interaction Techniques on Spatial Working Memory for 3D Scatter Plot Exploration in Immersive Analytics". In: *Institute of Electrical and Electronics Engineers* (2023).

[8] Jan-Paul Huttner, Kathrin Robbert, and Susanne Robra-Bissantz. "Immersive Ars Memoria: Evaluating the Usefulness of a Virtual Memory Palace". In: *Hawaii International Conference on System Sciences*. 2019.

[9] Jan-Paul Huttner and Susanne Robra-Bissantz. "An Immersive Memory Palace: Supporting the Method of Loci with Virtual Reality". In: *Americas Conference on Information Systems (AMCIS)*. 2017.

[10] Keisuke Irie et al. "Pervasive HoloMoL: A Mobile Pervasive Game with Mixed Reality Enhanced Method of Loci". In: *Association for Computing Machinery* (2017).

[11] Juming Jiang, Luke K, and Fryer. "The effect of virtual reality learning on students' motivation: Ascopingreview". In: *Wiley* (2023).

[12] Jakub Kraus and Michaela Porubanovà. "THE EFFECT OF BINAURAL BEATS ON WORKING MEMORY CAPACITY". In: *Studia Psychologica* (2015).

[13] Eric Krokos, Catherine Plaisant, and Amitabh Varshney. "Virtual memory palaces: immersion aids recall". In: *Springer* (2018).

[14] Richard N Landers et al. "Gamification Science, Its History and Future: Definitions and a Research Agenda". In: *Simulation & Gaming* (2018).

[15] Kenneth Y T Lim et al. "Representation of constructions of memory palaces with Learner-Generated Augmentation". In: *National Institute of Education (NIE)* (2020).

[16] Kenneth Y. T. Lim and Ryan Lim. "Semiotics, memory and augmented reality: History education with learner-generated augmentation". In: *British Journal of Educational Technology* (2020).

[17] Xiaojun Liu, Hui Zhang, and Liang Liu. "The perceived coolness of using virtual reality technology in blended learning performance can improve learning motivation and learning satisfaction". In: *Frontiers* (2024).

[18] Manuela Macedonia. "Embodied Learning: Why at School the Mind Needs the Body". In: *Frontiers in Psychology* (2019).

[19] Eleanor A. Maguire et al. "Routes to remembering: the brains behind superior memory". In: *Nature* (2002).

[20] Inc. Meta Platforms. *Access Scene data with OVRAnchor*. 2024. URL: `https://developers.meta.com/horizon/documentation/unity/unity-scene-ovranchor` (visited on 01/21/2025).

[21] Inc. Meta Platforms. *Meta XR Packages*. 2024. URL: `https://developers.meta.com/horizon/documentation/unity/unity-package-manager/#meta-xr-packages-overview` (visited on 01/21/2025).

[22] Inc. Meta Platforms. *Mixed Reality Utility Kit Features*. 2024. URL: `https://developers.meta.com/horizon/documentation/unity/unity-mr-utility-kit-features/` (visited on 01/22/2025).

[23] Inc. Meta Platforms. *OVRAnchor Struct.* URL: https://developers.meta.com/horizon/reference/unity/v71/struct_o_v_r_anchor (visited on 01/21/2025).

[24] Inc. Meta Platforms. *Scene Best Practices.* URL: https://developers.meta.com/horizon/documentation/unity/scene-best-practices (visited on 01/21/2025).

[25] Inc. Meta Platforms. *Spatial Anchors.* 2024. URL: https://developers.meta.com/horizon/documentation/unity/unity-spatial-anchors-persist-content (visited on 01/21/2025).

[26] Inc. Meta Platforms. *Spatial Anchors Best Practices.* 2024. URL: https://developers.meta.com/horizon/documentation/unity/unity-spatial-anchors-best-practices (visited on 01/21/2025).

[27] Inc. Meta Platforms. *Spatial Anchors Overview.* 2024. URL: https://developers.meta.com/horizon/documentation/unity/unity-spatial-anchors-overview (visited on 01/21/2025).

[28] Inc. Meta Platforms. *Spatial Data Permission.* 2024. URL: https://developers.meta.com/horizon/documentation/unity/unity-spatial-data-perm (visited on 01/21/2025).

[29] Inc. Meta Platforms. *UnitySceneOverview.* 2024. URL: https://developers.meta.com/horizon/documentation/unity/unity-scene-overview (visited on 01/21/2025).

[30] Inc. Meta Platforms. *Using the Scene Model.* URL: https://developers.meta.com/horizon/documentation/unity/unity-scene-build-mixed-reality (visited on 01/21/2025).

[31] Brigham Moll and Ed Sykes. "Optimized virtual reality-based Method of Loci memorization techniques through increased immersion and effective memory palace designs: a feasibility study". In: *Springer* (2020).

[32] Jonas Taule Nordli. "VR Memory Palaces: Can Walking Improve Memorization?" MA thesis. University of Bergen, Norway, 2024.

[33] Lars Nyberg et al. "Neural correlates of training-related memory improvement in adulthood and aging". In: *PNAS* (2003).

[34] Anco Peeters and Miguel Segundo-Ortin. "Misplacing memories? An enactive approach to the virtual memory palace". In: *Elsevier* (2019).

[35] University of Pittsburgh. *Active Memorization.* URL: https://www.asundergrad.pitt.edu/study-lab/study-skills-tools-resources/active-memorization (visited on 03/18/2025).

[36] Quintiliano. *Institutio oratoria.* 90-96 a.C.

[37] Ayisha Qureshi et al. "The method of loci as a mnemonic device to facilitate learning in endocrinology leads to improvement in student performance as measured by assessments". In: *American Physiological Society Journal* (2013).

[38] Rocco Raso et al. "Walkable Graph: An Immersive Augmented Reality Interface for Performing the Memory Palace Method". In: *Emergent Research Forum (ERF)* (2019).

[39] Henry L. Roediger. "The Effectiveness of Four Mnemonics in Ordering Recall". In: *Journal of Experimental Psychology: Learning, Memory, and Cognition* (1980).

[40] Matthew Rossman. "Simulating a Mixed Reality Memory Palace". MA thesis. University of Massachusetts, USA, 2020.

[41] Ángela Di Serio, María Blanca Ibáñez, and Carlos Delgado Kloos. "Impact of an augmented reality system on students' motivation for a visual art course". In: *Elsevier* (2012).

[42] Unknown. *Ad Caium Herennium.* 86-82 a.C.

[43] Yuki Yamada et al. "HoloMoL: Human Memory Augmentation with Mixed-Reality Technologies". In: *Association for Computing Machinery* (2017).

[44] Fumeng Yang et al. "A Virtual Reality Memory Palace Variant Aids Knowledge Retrieval from Scholarly Articles". In: *Institute of Electrical and Electronics Engineers* (2021).

[45] Frances A. Yates. *The Art of Memory.* Routledge, 1966.

[46] Zamzami Zainuddin et al. “The impact of gamification on learning and instruction: A systematic review of empirical evidence”. In: *Elsevier* (2020).

[47] Wenhao Zhang, Zhixia Chen, and Ruibin Zhao. “A Review of Embodied Learning Research and its Implications for Information Teaching Practice”. In: *Institute of Electrical and Electronics Engineers* (2021).

Abstract in lingua italiana

Il Metodo dei Loci Ciceroniani, una potente tecnica mnemonica dalle antiche origini, è stato raramente trasposto in ambito digitale e non ha ancora sfruttato appieno il potenziale delle tecnologie immersive. Sebbene esistano alcune applicazioni rudimentali in realtà virtuale (VR) e in realtà aumentata (AR), nessuna fa uso della realtà mista (MR). Questa tesi affronta tali lacune presentando la progettazione, lo sviluppo e la validazione di una nuova applicazione MR che rinnova il concetto di palazzo della memoria per un utilizzo educativo contemporaneo. La MR offre un vantaggio fondamentale rispetto a VR e AR: conserva il mondo reale (in linea con lo spirito originario della tecnica) permettendo al contempo la fusione di contenuti digitali con elementi reali.
Il nostro approccio consente un'elevata personalizzazione grazie al caricamento di contenuti personalizzati, garantisce la persistenza dei palazzi della memoria attraverso l'uso di ancore spaziali, permette agli utenti di muoversi liberamente nel proprio spazio e di interagire fisicamente con i contenuti virtuali. Per aumentare il coinvolgimento, l'applicazione integra elementi di gamification e funzionalità di condivisione, favorendo scenari di apprendimento collaborativi e competitivi.
L'usabilità e l'utilità percepita dell'applicazione sono state valutate attraverso un processo di validazione in due fasi che ha coinvolto 40 partecipanti. I risultati mostrano un'accoglienza molto positiva, con un eccellente punteggio in termini di facilità d'uso (4,45/5) e l'approvazione unanime da parte dei tester. In particolare, nonostante la grande maggioranza non conoscesse la tecnica, con il 60% dei partecipanti che non ne aveva mai sentito parlare e solo il 7,5% che l'aveva mai utilizzata, è emersa una buona disponibilità ad adottare lo strumento (3,88/5). Il feedback qualitativo raccolto da un focus group ha ulteriormente confermato il valore percepito dell'applicazione.
Questa tesi intende dimostrare che la MR è un potente mezzo per l'applicazione pratica e la diffusione del Metodo dei Loci, reinterpretando questa antica tecnica attraverso le lenti della tecnologia immersiva per gli studenti di oggi.

Parole chiave: Realtà Mista, Metodo dei Loci Ciceroniani, Palazzo della Memoria, Tecnologia Educativa, Memoria Spaziale, Tecniche di Memorizzazione

Acknowledgements

First of all, I would like to thank those who made this experience possible: Professor Nicoletta Di Blas and Matteo Valoriani from FifthIngenium. I found myself in a stimulating and welcoming environment and this experience has been incredibly valuable, as it bridges academic work with a more practical and professional perspective. I would also like to thank Matteo Di Marco, also from FifthIngenium, who supported me throughout the various stages of the project.
My deepest thanks go to my parents, for their love and for always standing by me, even during the most difficult times. If I've come this far, it's undoubtedly thanks to them. The environment in which we grow up shapes us deeply, and having people by my side even during the most stressful moments has played a fundamental role. This is something that should never be taken for granted and deserves to be acknowledged. For this reason, my gratitude extends to the rest of my family as well, for always helping to create a positive atmosphere around me.
I would also like to thank my friends, from those who have been in my life since childhood to those I've met along the way. Thank you for the moments of fun, distraction, and even meaningful conversations. I truly hope our friendship continues for many years to come.
Lastly, I want to thank all those who, in one way or another, have crossed my path and left a mark, whether positive or negative. We are shaped by our experiences, and I am proud of who I am. I cannot say whether I would be the same person if I had lived different moments or met different people, so, for all of it, thank you.