IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 31 July 2025, accepted 22 August 2025, date of publication 2 September 2025, date of current version 11 September 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3604765

RESEARCH ARTICLE

TeachAR–An Interactive AR-Based Learning Platform

MARIA ANCA BĂLUŢOIU, ANCA MORAR, ALIN MOLDOVEANU, ALEXANDRU GRĂDINARU, VICTOR ASAVEI, (Member, IEEE), MAGDALENA ANGHEL, AND CRISTIAN LAMBRU

Faculty of Automatic Control and Computers, National University of Science and Technology POLITEHNICA Bucharest, 060042 Bucharest, Romania

Corresponding author: Maria Anca Băluţoiu (maria_anca.balutoiu@upb.ro)

ABSTRACT Virtual reality and augmented reality technologies have experienced a significant increase in popularity due to their potential to deliver a highly immersive user experience. However, there are still substantial difficulties in creating augmented environments with different purposes that can precisely localize users. In this context, TeachAR is an augmented reality solution designed to provide students with an interactive learning experience through an augmented reality quest system. The system comprises a desktop application and a mobile application, designed for three distinct user roles: building administrators, teachers, and students. Administrators use the desktop application to upload the building's architectural plans, create a 3D replica of it, and define a navigation graph. Then, using the mobile application, the correspondence between the real building and its replica is created. Teachers employ a simple desktop user interface to design interactive quests. Students interact with the system through the mobile application in three different use cases: free exploration of the building, guided navigation to reach a destination, and individual or collaborative quests proposed by teachers. This paper provides a brief overview of the system's main functionalities, implementation details, and evaluation of its technical capabilities. The pilot testing is also described in this paper, revealing the potential of using the TeachAR system in schools in the educational process.

INDEX TERMS Augmented reality, gamification, indoor localization.

I. INTRODUCTION

In recent years, augmented reality (AR) has gained popularity, as evidenced by the increasing number of emerging AR development platforms that offer a seamless integration process for AR content into the real world. In this context, smartphones are a popular choice for augmented reality applications. Other devices (such as smart glasses) are also being used for AR applications; however, the price range is still a significant impediment for these devices in comparison to smartphones.

To develop a seamless AR application, precise localization is crucial for accurately displaying augmented content in the real world. Even though many AR development platforms provide localization functionalities, it is still a challenge to precisely localize a user in a custom coordinate system without complex technical setups using beacons or other hardware infrastructure, especially when multi-floor indoor localization is involved or when the environment contains visually repetitive content (such as schools or office buildings). In this context, there is a clear need for tools designed for non-technical users to enable the creation of spatially aware AR experiences that support meaningful interactions.

To address these challenges, this paper introduces TeachAR, an AR solution designed to facilitate indoor navigation, augmented content exploration, and gamified learning activities inside educational and cultural buildings. It offers a platform that requires no special hardware

The associate editor coordinating the review of this manuscript and approving it for publication was Francisco J. Garcia-Penalvo.

infrastructure, involving only regular desktop and mobile devices, with a simple enough setup workflow targeted at non-technical users. TeachAR comprises two applications: a desktop application used by building administrators and teachers to configure and manage interactive educational content, and a mobile AR application used by administrators and students to interact with location-based augmented content and navigate the building. The system offers three user roles (administrator, teacher, and student), each with a dedicated purpose within the system.

Administrators are guided through a step-by-step, intuitive graphical interface in the desktop application to create a virtual 3D replica of the building, starting from uploaded architectural plans and interactively placing architectural elements (such as walls, windows, doors, stairs, and elevators). Then, they can define points of interest and navigational routes through a connected node graph. For each point of interest, there is the possibility to enrich it with multimedia content (images, 3D models, text descriptions, videos, or audio files). Lastly, administrators can use the mobile application to map the 3D replica to the real-life building by simply navigating and placing AR Foundation cloud anchors at the designated points of interest.

Teachers benefit from a simplified user interface in the desktop application of TeachAR to create quests, which are gamified learning experiences consisting of multiple checkpoints (linked to real locations within the building) that guide students through different areas of the building to complete various tasks, such as solving quizzes, scanning QR codes, taking photos, or conducting experiments.

Students interact with TeachAR solely through the mobile application. They have access to three main functionalities: free roam inside the building, guided navigation towards a specific destination, or quest solving.

The paper is organized as follows. The second section provides a brief overview of recent developments in the augmented reality field, with a focus on indoor localization techniques and gamification elements. The third chapter offers a short description of the TeachAR system and its key functionalities. Next, the architecture and implementation details are described. The fifth chapter evaluates the technical capabilities and presents the results of pilot testing the system. Lastly, the final section presents several conclusions and discusses potential improvements for the proposed solution.

II. RELATED WORK

Augmented reality is a field of computer science that is continually evolving. In recent years, multiple software development kits (SDKs) have emerged, aiming at creating a smooth and seamless workflow in the development process of AR applications [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14]. Even though these platforms offer a wide variety of functionalities, such as tracking, pose estimation, image recognition, and augmentation, as well as multi-user content visualization, location-based AR, anchors, and cloud anchors, there remains a lack of solutions that cover most real-world use cases for augmenting a specific location with virtual content. For instance, there is still no solution that allows non-technical users to easily map an entire building and augment it, enabling multiple users to interact simultaneously with the augmented content. In particular, there remains a gap in educational AR platforms that integrate cloud anchor-based localization, indoor navigation, and a quest-based learning system, while also offering an intuitive tool for content creation for teachers.

However, considerable effort has been put into developing various solutions that allow users to easily reconstruct entire buildings in the virtual augmented space. This section provides a brief overview of existing applications in the augmented reality field.

A recent survey [15] demonstrates the successful integration of AR technology into various applications across different fields, including education, medicine, architecture, and entertainment. It also classifies these applications into five categories based on how AR technology is used: marker-based AR, markerless AR, projection-based AR, superimposition AR, and outlining AR. Another survey [16] analyzes future trends of AR technology. Both these surveys also emphasize a series of challenges and issues AR applications have encountered over the years. One of them is the high cost necessary to produce the required hardware in order to display augmented content over the real world (whether referring to headsets and smart glasses, handheld devices, projection-based systems, AR contact lenses, spatial augmented reality, or head-up displays). One of the most affordable solutions, also chosen for the TeachAR system, is the use of smartphones or tablets. Another common challenge is the unavailability of specific content for AR applications.

In response to these limitations, TeachAR provides a more straightforward solution based on a desktop interface used by teachers and administrators to create augmented content and to define a navigation graph intuitively, and a mobile application to place cloud anchors for building mapping and a gamified quest system to enrich the learning process of students. Unlike other educational AR systems, which either focus solely on multi-user content display or localization, TeachAR offers a solution for a real-world use case that integrates precise indoor localization and guidance, educational content customization, and interactive learning.

A. INDOOR AND OUTDOOR LOCALIZATION IN AUGMENTED REALITY APPLICATIONS

As evidenced by multiple studies that address the technical and environmental complexities of augmented reality systems [17], [18], [19], [20], [21], localization (primarily indoor localization, where GPS technologies do not prove to offer accurate enough measurements [22]) remains a significant challenge. Therefore, this sub-section focuses on the localization problem of existing augmented reality systems.

A comprehensive survey on indoor localization techniques based on computer vision, which also covers AR systems, is provided by Morar et al. [23]. Zafari et al. [24] present different techniques which have been used to estimate the indoor position and orientation (such as received signal strength indicator (RSSI), common system interface (CSI), arithmetic optimization algorithm (AoA), time of flight (ToF), time difference of arrival (TDoA), return time of flight (RToF), proof of authority (PoA), Neural Networks, k-Nearest Neighbors, and Support Vector Machine). Belghit et al. [25] present a comparative study between solving the perspective 3 points (P-3P) problem and acquiring coplanar information from 2D images in order to estimate the pose (position and orientation) of the device. Martin et al. [19] propose a system that uses Vuforia Engine to scan position markers with the smartphone camera in order to position the user inside the building. These markers can be scanned from any angle and position. They can be placed either on the floor or on a wall. They should be placed all over the site, but especially near points of interest in the building. Herbers and König [20] have successfully developed a framework that uses a building information modeling (BIM) and different indoor templates (they assume indoor geometry is not arbitrary; walls are usually straight lines perpendicular to one another) in order to estimate the pose of the user. Using inertial measurement units (IMUs) and simultaneous localization and mapping (SLAM), a point cloud is estimated through multiple frames. Next, the point cloud is compared to the existing templates to match the location in the building. A solution based on the mixture between feature extraction from images, Wi-Fi fingerprints clustering, and a prior reconstructed 3D space is presented by Hou et al. [26]. InLoc [27] is an indoor localization system that uses dense feature extraction and matching in order to estimate the pose of the user inside the building. Friske [28] uses AprilTags [29] to estimate the initial position. Afterwards, the pose is estimated with the help of ARCore (which uses input from IMU and the camera) in combination with SLAM (the collected input is sent to a base station PC, which computes the estimated position and orientation and sends the results back to the device). IndoorGNN [22] is a system that uses Graph Neural Networks to construct a graph representation of the building, where each node is represented by a specific RSSI vector and the weights of the edges are computed based on the similarity between the RSSI vectors. Zafari et al. [30] propose an iBeacon-based [31] solution. Using RSSI, the user's proximity is classified into four categories (immediate, near, far, unknown). When the user receives a universal unique identifier (UUID), a server provides relevant information and available actions for the specific location.

B. GAMIFICATION IN EDUCATION

Besides indoor localization and guidance inside schools, TeachAR is also designed as a solution for creating a fun and engaging learning experience during school classes by introducing quests, which teachers can easily define. Therefore, this subsection focuses on introducing elements of gamification into the educational process. Moreover, since the TeachAR quest system is also based on augmented reality for quest solving, special emphasis is placed on AR gamification techniques in educational activities.

A recent survey [32] analyzes the most popular gamification elements that have been utilized in recent years, with points, badges, leaderboards, levels, feedback, and challenges being prominently used (these elements can also be noticed on learning platforms such as Duolingo [33]). Moreover, the survey also highlights that most of the existing interactive learning systems do not rely on gamification theory or on existing gamification frameworks. Smiderle et al. [34] have analyzed how a basic gamification system (which offers points, badges, and a ranking system) has impacted a classroom of university students enrolled in a programming course. Before starting the experiment, all students were required to take personality tests. Afterwards, they were split into two groups: the ones who took part in a gamified learning process and the ones who followed the traditional learning process. At the end of the research, it was concluded that overall, regardless of their personality, students from the gamified focus group were more engaged in learning. However, depending on the personality type, even though they were more engaged, the efficiency in gaining knowledge varied. Numerous other studies [35], [36], [37] show similar results: gamification determines students to have a more positive attitude towards the subject they are studying; however, differences between students who learn in a gamified way and students who follow the traditional path are insignificant; still, there can be noticed a slight improvement.

Ryan and Deci [38] argue that motivation is determined by the fulfilment of three basic psychological needs: autonomy, competence, and relatedness. By satisfying these needs, people's intrinsic motivation (determined by enjoyment) increases. As a result, higher performance is obtained. Vygotsky [39] considers that the learning process involves interaction and collaboration, while Mayer [40] argues that people learn more effectively when information is conveyed through different channels (for example, visual and auditory). Taking these ideas into consideration, TeachAR was developed to include multimedia quests, both individual and collaborative, that involve solving multiple interactive tasks.

1) AUGMENTED REALITY IN GAMIFICATION

Even though gamification in education can be achieved using various strategies and elements, the current research targets augmented reality. Therefore, this sub-section covers recent findings on augmented reality in gamified learning.

A recent survey [41] emphasizes the benefits of using AR technology during the teaching learning process. The results of the survey show how AR applications can be used to create real-world simulations. Therefore, abstract ideas and concepts become more accessible for students to comprehend.

Other surveys [42], [43] present how AR applications are a good way for students to acquire knowledge in a practical manner, with hands-on experiences. As a result, it was proven that AR applications can reduce boredom in comparison to traditional learning methods. Moreover, it was proven that AR applications are beneficial for students with disabilities, to help them easily gain new knowledge and maintain their interest in learning for a longer period of time [44].

Çelik and Ersanlıhave conducted a study [45], where they have incorporated an AR application in a high school foreign language classroom. Students had to scan QR codes to be directed to different areas of the school, where they could experience an AR learning activity. Afterwards, they had to complete a quiz in a scavenger hunt activity in groups of three. Students who took part in this experiment have reported that the visuals from the AR experience increased their curiosity, and they were eager to attend the next class. Another study [46] (which was conducted on students in computer science) presents a comparative analysis between three different approaches: gaining knowledge using lecture notes, an AR prototype, and an AR prototype mixed with a gamification quiz. The results of this study emphasize the higher motivation reported by students who used the AR prototype. Other developed prototypes [47], [48], [49] aim at analyzing the benefits of using augmented reality in an educational context. The results of these experiments are promising and show great potential in using AR applications in order to stimulate the curiosity and engagement of students no matter the age.

The functionalities of TeachAR, both the quest system and the indoor localization functionality, are derived from the analysis of existing AR solutions. To cover a multitude of use cases, the system requires the administrator to provide the building plan. Afterwards, the administrator can augment the building by adding multimedia content (text descriptions, images, video files, audio files, and even 3D models). After a virtual replica of the building has been obtained, teachers can create classes, where students can be enrolled, and create AR quests for their classes. Lastly, students can use the mobile version of the system to navigate inside the building or to solve quests and gain experience, badges, or real-life prizes. All these use cases are detailed in Section III.

III. CONCEPT

The TeachAR system consists of two different applications: a desktop one and a mobile one. Each application has a different purpose, is designed for different users, and provides different functionalities. This section provides a brief overview of the entire system's workflow, including how the two applications communicate with each other and their specific purposes within the system. TeachAR comprises three distinct user roles: administrator, teacher, and student. This section presents the use cases of the two applications, structured according to the functionalities associated with each user category. The workflow of the entire TeachAR system is illustrated in Fig. 1.

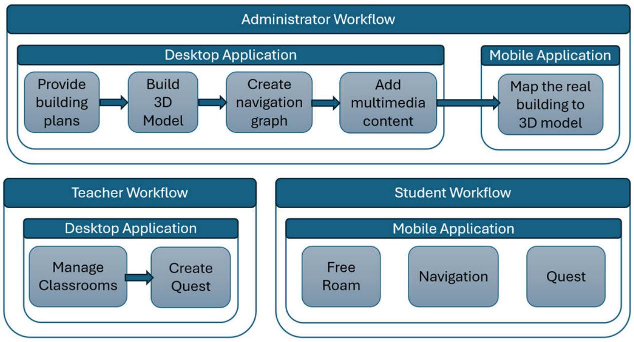


FIGURE 1. Workflows of the TeachAR applications associated with each user category: administrator workflow (top), teacher workflow (bottom left), and student workflow (bottom right).

A. ADMINISTRATOR

The role of the building administrator is to manage existing user accounts, provide architectural plans of the building, create a 3D model of the building, define walking routes inside the building, and supply multimedia content to augment the real-world environment. All these functionalities can be easily performed within the desktop application of TeachAR, requiring only minimal knowledge of using computer graphical interfaces, with the user being guided through a step-by-step process. Further on, to map the virtual floor plans to the real building, after the virtual replica of the building has been created, the administrator must add a series of cloud anchors [50] inside the real-life building using the mobile application. This step is necessary to ensure that indoor localization is computed correctly. This role can be filled by the building administrator, if one exists, or the person responsible for maintenance and configuration of IT systems in the school.

1) DESKTOP APPLICATION

The administrator must first create a virtual replica of the building using the desktop application. To do this, the administrator must upload the architectural plans and specify the dimensions for each floor of the building. Afterwards, the replica creation process of the 3D model of the building can begin. During this step, the administrator uses the floor plans to place walls, doors, windows, and any means of connecting the floors (stairs and elevators). After replicating the building, the next step in the configuration process is to define the navigation graph. At this stage, the stairs and elevators are automatically converted to nodes of this graph. The administrator adds control points (used to define routes) and points of interest (representing specific locations within the building) to the graph. Afterwards, the administrator completes the graph by adding edges to connect all these nodes. The weight of each edge is automatically computed as the distance between the two nodes that the edge connects. This step in the administrator's workflow is complete only when all nodes are connected to each other through a path and there are no isolated nodes. Lastly, the administrator can add multimedia content (text descriptions, video files, audio files, images, and 3D models) inside the building. This content can only be attached to doors (for example, doors of specific classrooms or laboratories) and to previously defined points of interest.

2) MOBILE APPLICATION

After the building and the navigation routes have been completely constructed using the desktop application, the administrator proceeds to the phase of mapping the virtual content to the corresponding physical locations within the real-world building. The AR-based mobile application utilizes Google ARCore's cloud anchor technology. With this application, the administrator begins by scanning the surroundings and continues by selecting the corresponding floor. Afterwards, a top-down view of the replica of the building will be displayed on screen as a minimap. Administrators are now required to set their initial orientation in degrees in the real world using a slider, and determine their starting position by selecting a point on the minimap. Lastly, the administrator can begin placing AR cloud anchors in the real-world building and hosting them on Google servers [51]. The anchors should be placed inside the entire building, but they should not be too close to each other (there should be at least 8 meters between them; if not, the students' mobile application may exhibit reduced responsiveness due to excessive background computations required for anchor solving since image matching algorithms are run for each anchor) or too far away (otherwise, students might find themselves in locations where there are no anchors and the application cannot localize them inside the building). Moreover, for accurate alignment of virtual content across multiple floors, administrators should place cloud anchors at critical transition points, such as the start and end of each staircase or in front of each elevator.

B. TEACHER

The role of the teacher is mainly to manage the gamification elements of TeachAR. To achieve that, teachers only need the desktop application of the TeachAR system. Using this application, teachers can define multiple quests tailored to their pedagogical needs. A quest is defined by the following elements: title, description, maximum number of students who can collaborate for solving the quest (maximum four students), deadline (if necessary), start location (where the students can find the quest inside the school), rewards (experience (XP) points, badges, real-world prizes), and a list of multiple checkpoints. For a collaborative quest, students can send invitations to each other to form teams. Once a team has been created, each student from the team will receive an ID, starting from 1. A checkpoint is an intermediate step in solving the quest. A checkpoint is also defined by the following elements: title, description, location (where the students can find this checkpoint to solve it inside the building), the id within the team of the assigned student for this checkpoint (if the checkpoint is part of a collaborative quest), hint (available only for students with lower grades), a list of the possible checkpoints of the quest which can follow the current checkpoint, and a validation form of the checkpoint (scanning a QR code, taking a photo of an object from the real world, solving a quiz, or conducting an experiment in the real world). Except for experiment validation, all other validation forms for a checkpoint are performed automatically. However, for the experiment, manual grading is required. To choose the starting location for a quest or the location of a checkpoint, teachers do not need to use the AR application to be localized indoors. Instead, the desktop application renders the 3D model of the building previously created by the administrator, allowing them to select the location from the rendering.

After the quests have been created, the teacher can define a new class or update an existing class. To achieve this, the teacher can enroll students in the class and, subsequently, assign previously created quests. As a result, all students enrolled in a class will be able to solve all the quests assigned to that specific class. Students cannot see quests that are not assigned to their class.

C. STUDENT

Users assigned the student role require only the mobile application of the TeachAR system. Within the application, students have access to three primary functionalities: free roam, guided navigation, and quest solving. However, before using any of these application modes, students must perform a calibration step: once the application is running, they must scan their surroundings until a cloud anchor is detected. Once the anchor has been solved, the indoor localization algorithm can accurately compute the user's position within the virtual map previously created by the administrator.

1) FREE ROAM

The free roam use case of the mobile application in the TeachAR system allows students to freely explore the building and discover augmented content. This is the default running mode of the application. In this mode, students are given no on-screen indications. They can freely explore the building and can visualize and interact with all the multimedia content previously attached to doors and interest points by the administrator. Moreover, if they are in proximity to available quests or ongoing checkpoints of quests they are currently working on, students can visualize a 3D model of a quest/checkpoint marker in the augmented space. As detailed in Section V, students find this use case very useful when they are in new locations and want to familiarize themselves with the location.

2) GUIDED NAVIGATION

The guided navigation functionality was designed to help students orient themselves within the school. Instead of getting lost trying to find a specific classroom, laboratory, or teacher's office, students can use this functionality to let the mobile application guide them. All they have to do is select their desired destination. Using the Dijkstra algorithm [52] on the graph previously defined by the building administrator, the shortest path is computed, and the student is guided to their destination with an arrow on the screen, pointing in the direction they should go. In the case where the student needs to go to another floor, the application displays more specific details on the screen (for example, the floor where the student needs to exit the elevator or whether the student should go up or down the stairs). When the students reach their destination, a notification message appears on the screen for a few seconds, and the guidance arrow disappears.

3) QUEST SOLVING

Regarding the quest system, students can visualize, start and solve quests using the mobile application. As previously mentioned, students can see markers in the augmented world indicating available quests or the ongoing checkpoints of quests they have already started. By tapping on such a marker, students can see more details about the quest or the checkpoint. If the student taps on the quest marker, details about the quest (such as the title, the description or the deadline) will be displayed, and if the quest is a single player quest, the student will be able to start it; however, if there is a collaborative quest, the student will be able to send invitations to other students to join the quest. Once all the students have accepted the invitation, the quest is automatically started. On the other hand, if the student taps on the checkpoint marker, details about the checkpoint and the possibility of solving the checkpoint will be displayed. Details about an ongoing checkpoint can be viewed from any location, but the checkpoint can only be solved at its designated location. To solve a checkpoint, the students might be required to do a practical task (in which case they cannot move to the next checkpoint unless they have been manually graded by the teacher), to complete a quiz (which will be displayed inside the mobile application), to scan a QR code or to take a photo of a specific object. If the scanned QR code does not match the expected QR code or if the photo taken by the student does not match the ground truth image from the database, then an error message is displayed on the phone screen, and the student is required to scan a different QR code or take a new photo. When all checkpoints of a quest are successfully completed, the student receives a notification message, the quest is marked as complete, and the student receives the rewards.

IV. PROPOSED SOLUTION

Fig. 2 illustrates the architecture of the TeachAR system. As previously stated, TeachAR consists of two different applications: a desktop one (used by the administrator to configure the building and by the teacher to define the quests) and a mobile one (used by the administrator to map the real-world building to its 3D model and by the students to navigate inside the building and to solve quests). Furthermore, the system requires a content management system (CMS) that stores the multimedia content, all the necessary information to render the building in the virtual world and to map it to the real-life building, and the navigation graph required for route planning for students.

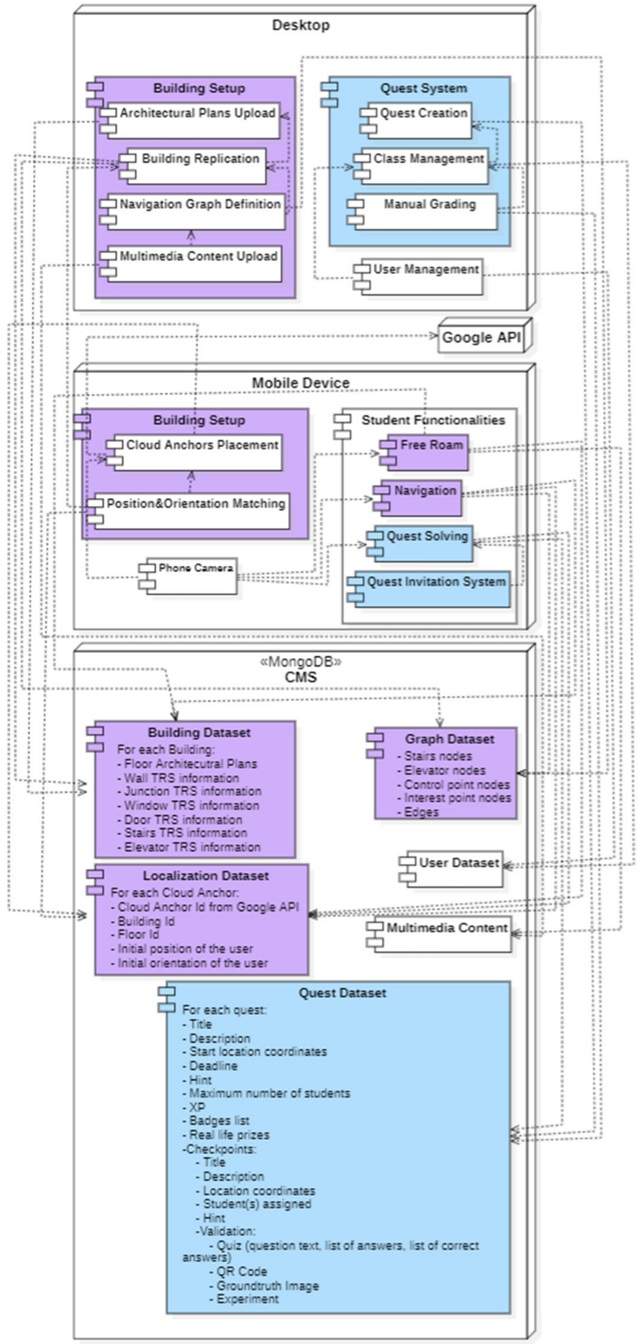


FIGURE 2. Deployment diagram for TeachAR system; localization modules are represented in purple, quest system modules are represented in blue; the application comprises of three subsystems: the desktop (used by administrators to define the building and by teachers to create and manage classes and quests), mobile (used by administrators to map the real-world building to its 3D replica), and the CMS (used to store the necessary configuration and user data).

A. ARCHITECTURE

Fig. 2 illustrates the architecture of the TeachAR system. As previously stated, TeachAR consists of two different applications: a desktop one (used by the administrator to configure the building and by the teacher to define the quests) and a mobile one (used by the administrator to map the real-world building to its 3D model and by the students to navigate inside the building and to solve quests). Furthermore, the system requires a content management system (CMS) that stores the multimedia content, all the necessary information to render the building in the virtual world and to map it to the real-life building, and the navigation graph required for route planning for students.

Before the system can be used by teachers and students, the administrator of the building must firstly configure the virtual representation of the building and then align it with the corresponding physical environment. All the data necessary for this configuration is stored in the CMS. The virtual configuration of the building is divided into four phases: uploading the architectural plans of the building (the Architectural Plans Upload component), building the 3D model of the building (the Building Replication component), providing the navigation graph (the Navigation Graph component), and uploading the multimedia data necessary for the augmented reality enhancement of the building (the Multimedia Content Upload component). The mapping of the virtual world to the real building is done by adding cloud anchors [50] in the real world, storing them using Google API [51] (the Cloud Anchors Placement component), and aligning the 3D model of the building to the real-world building (the Position & Orientation Matching component).

Afterwards, teachers and students can start using the system. Teachers can define classrooms (the Classroom Management component), enroll their students, define quests (the Quest Creation component), and assign them to their classrooms. Students will be able to see and complete only quests assigned to the classrooms they are enrolled in (the Quest Solving component and, in case of collaborative quests, the Quest Invitation System component). To define quests and checkpoints, teachers require data from the CMS regarding the 3D representation of the building to accurately place the starting locations within the building.

Regardless of the operational mode selected by students within the mobile application, all three use cases (Free Roam, Navigation, and Quest Solving) require the data that defines the building from the CMS, along with the localization dataset (which is used for mapping the virtual and the real world by retrieving from Google API the previously defined cloud anchors) and the multimedia content. Moreover, the navigation use case also requires the CMS to provide information about the navigation graph in order to compute the optimal path to the destination.

B. IMPLEMENTATION DETAILS

Regarding the technologies used in implementing the proposed solution, both the mobile and desktop applications have been developed using the Unity game engine [53]. This choice was motivated by several factors, including Unity's relative ease of use compared to other game engines, its robust and easy-to-integrate physics system, as well as the possibility to easily create builds for different platforms (Windows, Android, iOS). Furthermore, as previously stated, the mobile application needs to integrate AR functionalities to localize the user within the building. In this regard, Unity's AR Foundation [1], along with Google ARCore extensions package [54], was used. Regarding the CMS, MongoDB Atlas [55] was selected as the database solution. The primary reason behind this choice was its NoSQL architecture [56], which is particularly advantageous for TeachAR because it does not require rigid schemas, making it suitable for complex nested relationships such as a quest and its checkpoints. The communication between the two applications and the database is done using REST API [57].

1) DESKTOP APPLICATION

The desktop application was designed to provide functionalities tailored only for administrators and teachers. Administrators can define the buildings and their navigation graphs in the virtual environment, along with the augmented content for the mobile application. Teachers can manage their classes, create quests, and manually grade students.

To define the building, after uploading the architectural plan for the current floor, the user must place the walls (by clicking on the ground floor), doors, and windows (by clicking on the walls). Afterwards, the navigation graph can be defined by placing its nodes (stairs, elevators, control points, and interest points) on the floor and connecting them by adding edges between them. Additionally, in this phase, the previously placed doors are also considered nodes in the graph, so they must be connected to the other nodes. The weight of an edge is automatically computed as the distance in meters between the nodes that are connected by that edge. Fig. 3 illustrates a 3D reconstruction of the ground floor from two existing floors of a building, in which the following elements were added: walls, doors, windows, control points, interest points, stairs, and the graph edges.

In the case of multiple floors, when the administrator adds a stairs node to the current floor, the connected floor must be specified. Once this step is done, the application automatically creates a new stairs node on the specified floor at the same position and connects it to the initially added stairs node. The weight for this new edge is automatically computed as the length in meters of the stairs node. On the other hand, when adding a new elevator node to the current floor, the application automatically adds a new elevator node to the same position on all existing floors. Then, edges are added between all these newly created nodes. The weight for such an edge is 0 (since moving from one floor to another using an elevator does not require the user to move). The administrator cannot save the building's status until the navigation graph has become a connected graph.

Lastly, for the augmentation step of the building, the administrator can add multimedia content to any door or pre-

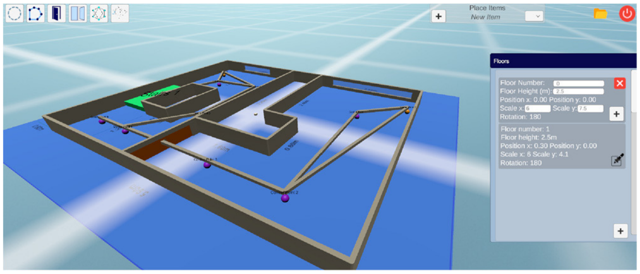


FIGURE 3. **Visualization of a building floor and its corresponding navigation graph; administrators can select and edit floors from the right panel of the screen (upload map, set the translation, rotation and scale of the floor); walls, windows, and doors can be added from the top left menu; interest points, control points, elevators and stairs can be added from the top left menu; graph edges are added from the top right menu.**

FIGURE 4. **Multimedia upload menu for doors and interest points; the administrator can set a title, a description, upload one audio file, multiple images, one video, and multiple 3D models.**

viously defined interest point. The menu for this functionality is shown in Fig. 4.

To create a new quest, the teacher must provide several details, including a title, a description, a starting point (if necessary), a deadline, or a list of checkpoints that must be completed to finish the quest successfully. Fig. 5 illustrates the user interface (UI) for creating a new quest. If teachers want a specific start location for the quest, they need to select it in the virtual model of the building. The coordinates are automatically computed.

The process of adding a new checkpoint to the quest is similar to creating a new quest (including the process of selecting the location of the checkpoint). This time, however, teachers need to choose from the list of available checkpoints the ones that could follow the current checkpoint. Fig. 6 presents the UI for creating checkpoints, while Fig. 7 illustrates three of the four different types of checkpoint validation.

For each validation method, teachers are required to take different additional steps:

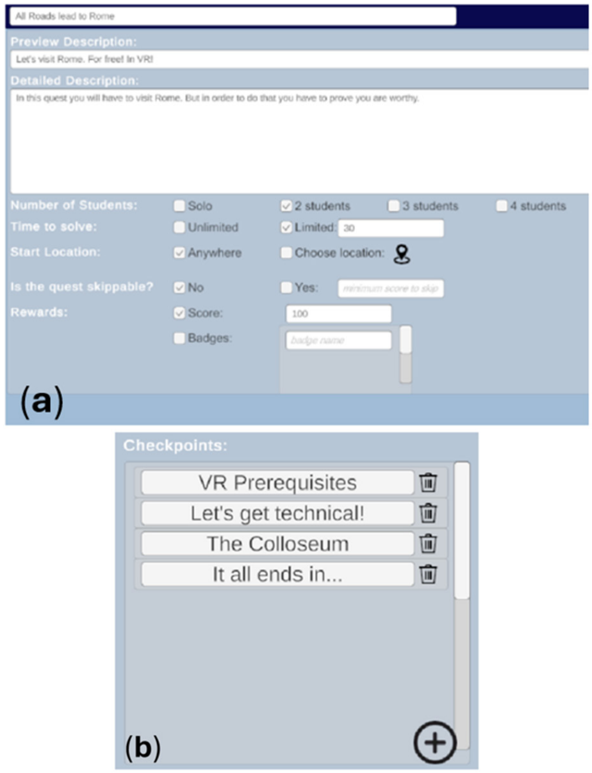


FIGURE 5. **Visualization of the quest edit menu. Main menu – teachers choose a title, write a description, select how many students can collaborate for the quest, set a time limit, choose a starting point (if necessary), select if it is a mandatory quest or not to be solved by the students, and choose the rewards for completing the quest (a); checkpoint list menu for a quest (b).**

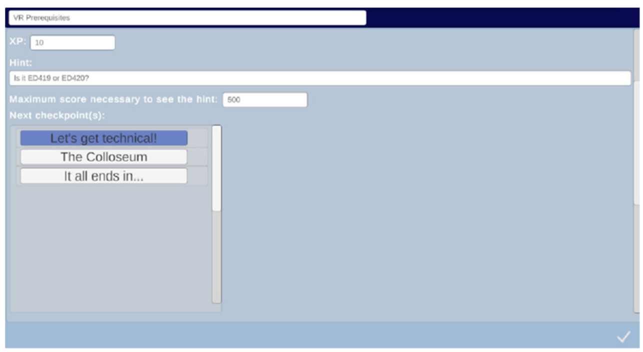


FIGURE 6. **Visualization of the checkpoint edit menu; teachers fill in the necessary information to define a checkpoint and link the current checkpoint to already existing checkpoints to define the checkpoint order for a quest.**

- Practical Task: no additional steps are required when defining the checkpoints, but manual grading will be necessary when the student solves this checkpoint;
- QR Code: teachers are required to download the automatically generated QR code (ZXing library [58] was used for QR generation and scanning);
- Photo: the teacher needs to provide a ground truth image, which will be used by the Augmented Images [59]

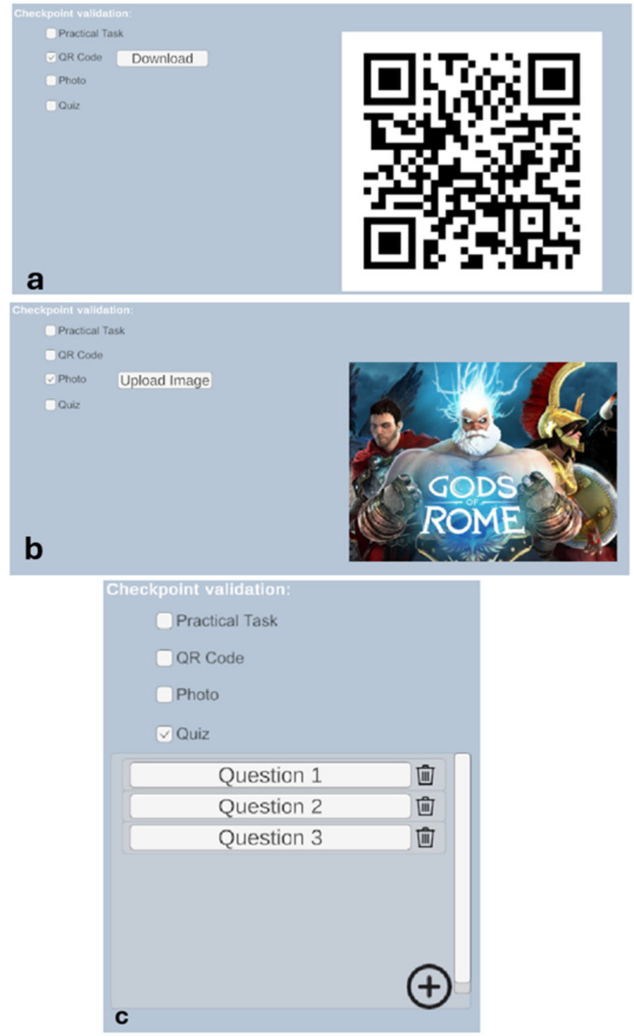


FIGURE 7. Visualization of the process of defining the checkpoint validation form. QR scan (a); photo taking (b); quiz answering (c).

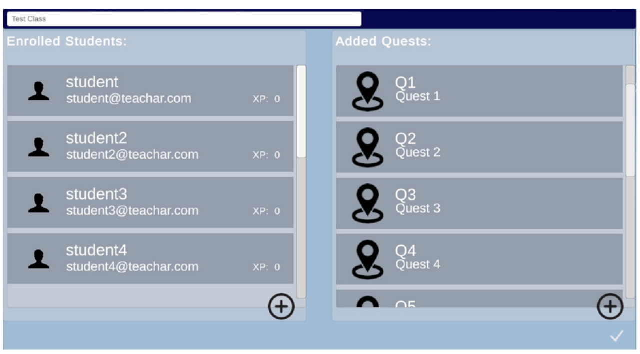


FIGURE 8. Classroom management user interface: on the left side teachers can enroll students to the current classroom, and on the right side teachers can assign quests to the classroom.

functionality of ARCore to automatically evaluate if the student has taken a photo of the required object;
- Quiz: the teacher is required to generate a quiz with multiple-choice questions.

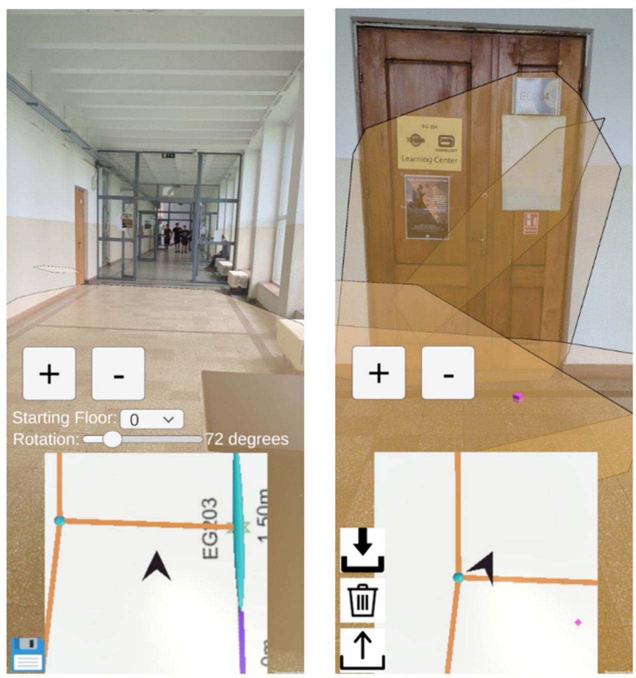
The classroom management functionality allows the teacher to enroll new students (or remove already enrolled students) and add (or remove) quests. Fig. 8 illustrates the interface of the classroom management.

2) MOBILE APPLICATION

The mobile application was designed to provide functionalities for administrators and students. Administrators can complete the final phase of the building definition process (mapping the virtual building to the real-world building), and students can explore the real-life building in one of the 3 possibilities: free roam, guided navigation, and quest solving.

The steps the administrator must follow to map the 3D model of the building to the real-world building (with screenshots from the process in Fig. 9) are summarized below:

1) Environment scanning: the user scans the surrounding environment using the smartphone until most of the planes are detected;
2) Setting the initial position and rotation: the administrator taps on the UI minimap to set the initial position of the phone when the application is turned on. Similarly, the administrator uses a slider from the UI to rotate the minimap to reproduce the initial orientation of the phone;
3) Cloud anchor placement and hosting: after the initial pose of the mobile phone has been set, the administrator can start placing cloud anchors (which will be hosted online using Google API) on the previously detected planes.



(a) (b)

FIGURE 9. Workflow of the mapping process of the real building to the 3D model: planes detection and initial pose estimation (a), cloud anchor (represented as a pink cube) placement and hosting (b).

When students launch the application, they must scan their surroundings until at least one cloud anchor (which was saved by the administrator during the mapping process) is solved and placed in the scene. Then, using the initial pose (position and orientation) of the camera relative to the cloud anchor, the coordinate system of the virtual building is transformed such that the student's position within the virtual environment aligns accurately with their physical location in the real-world building. Afterwards, to ensure that no computation errors accumulate over time, the system computes the closest cloud anchor and repositions the entire virtual building relative to the anchor.

When the localization process is complete, students automatically enter the free roam running mode of the application. In this mode, they can visualize, superimposed on the video frames acquired with the phone camera, the multimedia content previously defined by the administrator. Moreover, if they are in proximity to an available quest or to an ongoing checkpoint, they can also visualize the special marker for quests and checkpoints. They can interact with these markers by tapping them on the phone screen, and the quest can be started or the checkpoint completed.

Students can also use the guided navigation system to help them orient themselves within the building. From the search menu, they can select one of the possible destinations, and afterwards, an arrow will guide them to the chosen place they want to reach. The navigation route is computed using the Dijkstra algorithm. The arrow always indicates the direction in which users should walk to reach the next node of the selected graph route. When the distance between the user and the current node is smaller than a threshold, the next node in the path will be selected. When they arrive at their destination, a message will be displayed on the smartphone screen, and the arrow will disappear. Special indication messages will also be displayed when the user has to climb stairs or take the elevator to a specific floor. Fig. 10 presents various screenshots captured from a mobile device running the application under a student account.

FIGURE 10. Screenshots illustrating the use cases of TeachAR from the perspective of the student as follows: augmented content inside the building and the guidance arrow (a and b), checkpoint marker in the real world (c), message to go up the stairs (d), notification that the destination was reached (e), and message guiding the student to take the elevator to the ground floor (f).

V. EVALUATION

The TeachAR system was evaluated to assess both the accuracy of the indoor localization system and the intuitiveness and feasibility of the overall user experience. To achieve this, the system was piloted with 48 potential users, enabling the collection of both quantitative and qualitative data on localization accuracy and user experience. All selected participants were between 20 and 35 years old and reported regular use of smartphones in daily activities such as social media, gaming, online shopping, or work. Regarding prior AR experience, all participants were familiar with the concept. However, not all of them have used AR applications before this experiment. Approximately three-quarters of the participants had previously engaged with AR-based games. Moreover, to evaluate the utility of the navigation component, 16 additional participants were asked to participate in the pilot testing, which involved navigating in traditional ways (using paper maps and indoor signs from the location) rather than using the application. Before starting the experiments, all participants were notified about data collection, and they could opt out of the piloting if they did not agree. No other approvements were necessary for these experiments.

The application was tested within the Faculty of Automatic Control and Computers at the National University of Science and Technology POLITEHNICA Bucharest [60]. The initial participants in this evaluation were third-year students from the Faculty of Industrial Engineering and Robotics at the same university [61]. They have visited the Faculty of Automatic Control and Computers a few times in the past, but have not spent enough time there to easily find specific locations within the building. In the second phase of piloting TeachAR, 16 individuals who had never been on the university campus were asked to test the application. During the pilot testing, participants completed two different tasks designed to evaluate both the indoor localization component and the gamification functionalities.

To assess the accuracy of the indoor localization functionality, all users started from the same classroom. They were individually asked to find four different classrooms inside the faculty in a specific order using the TeachAR mobile application. The classrooms were deliberately selected to require the participants in the study to navigate across multiple floors using both the elevator and the staircase. This experiment was conducted during both class hours, when most students were attending classes, making the hallways relatively free, and during course breaks, when hundreds of students were traveling from one classroom to another.

The second task participants had to complete was to take part in a gamified quest. A collaborative quest, designed for two participants with a one-hour time limit, was created for this use case. In this quest, participants had to complete four checkpoints, as follows:

1) They had to find a teacher's office from which they could borrow two Meta Quest 3 headsets [62]. To assess the completion of this checkpoint, each participant had to scan a QR code;
2) Only after both participants had scanned the QR code could they move on to the next checkpoint. In this checkpoint, only the first user had to find the virtual reality (VR) laboratory of the faculty. Once there, the user had to answer a quiz with three questions about virtual reality;
3) The next checkpoint required both participants to locate the computer graphics laboratory, where they had to complete a practical task: using the VR headsets to find a hint that would aid them in solving the next checkpoint. This task was manually graded by the teacher. When both users were graded, they could move on to the last checkpoint;
4) This checkpoint required only the second user to find another classroom. Inside the classroom, there were posters with different mobile games. The user had to take a picture of the poster, which was hidden inside the VR application.

A. EVALUATION OF TECHNICAL CAPABILITIES

To evaluate the technical capabilities of the solution in terms of indoor localization accuracy, multiple tests were conducted both before and during the pilot testing. During the development process, accuracy tests were conducted for the ARCore localization technology. The results were presented in [63] and [64].

During the pilot testing of the system, data was constantly collected from the participants. All participants have used the same smartphone, Google Pixel 7 [65]. The first conducted measurement was to determine the time it took participants to reach their destination while using the TeachAR mobile application. This measurement was performed for all participants in the study. Table 1 represents the recorded times for ten participants to reach the four destinations they were asked to find. On average, participants who used the application required approximately 55.73 seconds to navigate between EG204 and EG304. In contrast, participants who used paper maps required 68.25 seconds, while a person who knew the location reached the destination in 35 seconds. To reach the second destination, participants who used TeachAR needed on average 140.46 seconds, while participants who used paper maps took on average 161.81 seconds. A person familiar with the location requires approximately 120 seconds. On average, 40.32 seconds were necessary to reach the third location for users of TeachAR and 52.50 seconds for users of maps and signs from inside the faculty. Meanwhile, a person familiar with the location needed 25 seconds. For the last destination, participants who used the application took approximately 130 seconds to reach it. In contrast, the participants who used maps and indoor signs required, on average, 142.81 seconds, while those who knew the location required 115 seconds.

To assess whether the obtained results were relevant or further testing was required, a t-test was performed, considering the two focus groups: participants who used the application and participants who used paper maps and indoor signs. The test was performed on all four routes that the users had to follow. In all cases, the obtained p-value was under 0.005 (6.19×10^{-5} for the first case, 1.27×10^{-4} for the second case, 2.83×10^{-12} for the third case, and 4.42×10^{-4} in the last case). Thus, the difference between the two focus groups was statistically significant.

Another test that was conducted was to determine how often the participants have felt confused while using the navigation component. In this regard, a "Feeling Confused" button was added to the interface of the application. Participants were asked to press the button each time they felt confused in their navigation. Analyzing the logged data, it was noticed that users who pressed the button felt, in general, confused in the same area of the building. In total, 13 users have pressed the button. This emphasized the limitations of the TeachAR indoor localization system:

- To function properly, cloud anchors should be used in areas with a variety of textures (for instance, a room with only white walls would be problematic for cloud anchors). Moreover, the textures should be varied and unique (repetitive patterns might result in placing the cloud anchors at the wrong position);
- The building administrator should have the exact plans of the building. Moreover, these plans should accurately depict all the angles between the walls of the building (especially if the building does not have only walls perpendicular to one another). Otherwise, when placing the cloud anchors, the administrator will not be able to properly set the initial orientation, resulting in a displacement as illustrated in Fig. 11. If the angle is incorrectly calculated, then as the user moves down the hallway, the positioning error propagates. For the pilot testing of TeachAR, the building plans of the faculty did

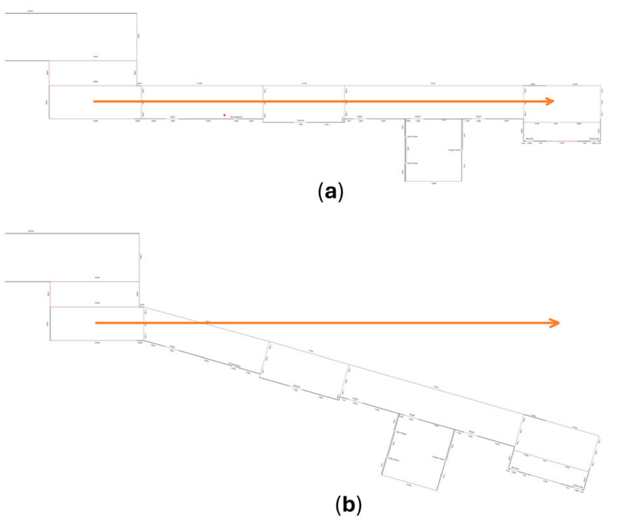


FIGURE 11. Possible navigation path of a user if the angles between the walls are not properly set: ideal navigation path if the angle between the walls is considered 90° (a); actual navigation path and displacement of the user's position if the angle between the walls is considered 90°, even though it actually is 106°.

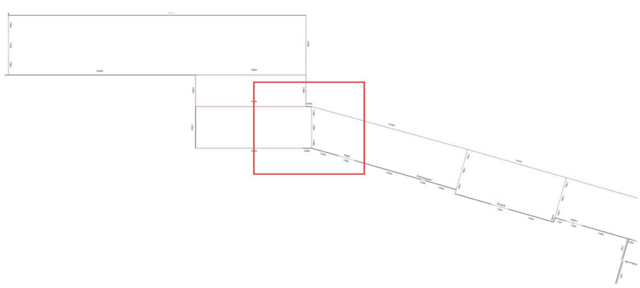


FIGURE 12. Snippet of the faculty of automatic control and computers ground floor plans. The hallway is not perpendicular to the rest of the building.

not specify the angles between the walls, so the angles from Fig. 12 were manually approximated.

As illustrated in Fig. 12, there is a particular case inside the Faculty of Automatic Control and Computers where a hallway is not perpendicular to the others. At the same time, the hallway has repetitive textures therefore, cloud anchors had to be carefully placed. As a result, in this hallway, there are fewer cloud anchors placed, so the algorithm does not recompute the position and orientation of the map as often as it does in other areas of the faculty. The lack of cloud anchors, combined with the fact that the hallway is not perpendicular to the rest of the building, resulted in a displacement of the hallway (pictured in Fig. 13). Some participants felt this displacement and became disoriented, while others experienced no issues with it.

Another aspect evaluated during the pilot testing was to determine the positional accuracy (specifically, the extent to which users adhered to the intended navigation route) in areas where there were no known problems, as previously mentioned. In this regard, the system logged the user's position every few seconds, as well as the closest point in the navigation graph (computed as the projection of the user's position onto the current edge in the graph), as illustrated in Fig. 14. Table 2 presents ten different measurements from participants in the pilot study. On average, the position error is approximately 0.53 meters.

Additionally, the orientation error was assessed through two distinct sets of tests specifically designed for this purpose. In the first test, the angle between the direction in which the user is supposed to go (computed as the vector between the position of the user and the position of the target node in the graph) and the actual orientation is computed and

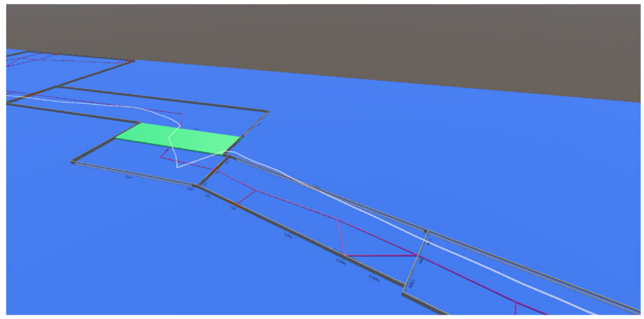


FIGURE 13. Displacement felt by some participants in the hallway with repetitive textures and not perpendicular to the rest of the building; purple lines represent the edges of the navigation graph, while the white lines represent the trajectory of the user.

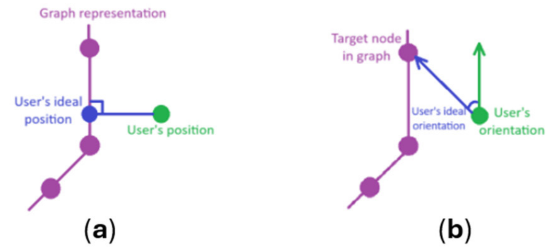


FIGURE 14. Position and orientation error computation: position error computed based on the actual position of the user and the closest point from the navigation graph (a); orientation error computed as the angle between the user's orientation and the actual direction in which the user is supposed to go (b).

TABLE 1. The necessary time to reach the destination using the TeachAR mobile application.

Entry	EG204-EG304 (seconds)	EG304-EC105 (seconds)	EC105-Gatekeeper (seconds)	Gatekeeper-ED419 (seconds)
1	76.66	138.12	52.48	141.47
2	69.40	137.99	41.36	145.22
3	60.45	151.80	45.08	122.13
4	61.31	141.41	42.59	119.06
5	60.01	178.13	56.27	111.27
6	42.01	131.45	44.04	131.65
7	57.60	129.45	32.70	171.26
8	64.64	117.28	45.12	117.69
9	37.34	149.68	50.75	131.66
10	62.72	125.15	47.12	177.58

TABLE 2. Position error computed as the distance between the position of the user and the actual position in the graph.

Entry	User Position	Projection on Edge	Position Error (meters)
1	(-33.53, 0.37)	(-33.59, -0.98)	1.35
2	(-46.36, -1.54)	(-46.25, -1.31)	0.25
3	(-47.06, 5.82)	(-47.35, 5.90)	0.30
4	(-25.63, -8.21)	(-25.41, -8.35)	0.26
5	(0.99, -15.80)	(0.68, -16.43)	0.70
6	(47.97, -24.60)	(47.34, -23.45)	1.31
7	(53.97, 35.85)	(53.18, 35.07)	1.11
8	(50.63, 36.98)	(50.50, 36.98)	0.13
9	(53.51, 36.93)	(53.38, 35.57)	1.36
10	(47.14, 27.29)	(46.44, 27.26)	0.7

TABLE 3. Orientation error between the ideal orientation of the user and the actual orientation.

Entry	Ideal Orientation	User Orientation	Orientation Error (degrees)
1	(−0.63, 0.40, −0.35, −0.54)	(-0.59, 0.38, -0.38, -0.59)	8.46
2	(-0.16, -0.47, -0.12, 0.84)	(-0.11, -0.32, -0.08, 0.93)	15.43
3	(-0.10, 0.0, 0.0, 0.99)	(-0.08, 0.0, 0.0, 0.99)	7.32
4	(-0.16, 0.48, 0.12, 0.83)	(-0.11, 0.32, 0.08, 0.93)	16.35
5	(0.03, -0.37, -0.07, 0.92)	(0.02, -0.19, -0.04, 0.98)	20.73
6	(-0.03, 0.32, 0.06, 0.94)	(-0.02, 0.19, 0.04, 0.98)	13.82
7	(0.16, -0.47, 0.11, 0.84)	(0.11, -0.32, 0.08, 0.93)	21.33
8	(0.05, 0.0, -0.04, 0.99)	(0.11, 0, -0.08, 0.98)	6.52
9	(0.10, 0.43, 0.0, 0.89)	(0.08, 0.32, 0.0, 0.93)	11.77
10	(0.06, -0.42, 0.07, 0.89)	(0.04, -0.27, 0.05, 0.95)	9.38

logged every few seconds. Fig. 14 illustrates how the orientation error was computed. Table 3 presents ten different logged angles. Overall, an average orientation error of 13.11 degrees was obtained.

The second test measured the number of instances in which the user was oriented entirely in the opposite direction of the intended movement. It was considered that the user was oriented in the opposite direction if the angle between the orientation of the user and the direction of the target was approximately 180°+/- 45°. Besides the moment the user had just started the application (when it was possible that the user was facing backwards when selecting the location to navigate to), there were no logged cases where the user was facing backwards in the supposed direction of movement.

B. PILOTING THE SYSTEM IN A REAL BUILDING

After completing both tasks, participants in the pilot testing were asked to complete a form to evaluate the system. Regarding the indoor localization component, the system received generally positive reviews, with 97.5% of the

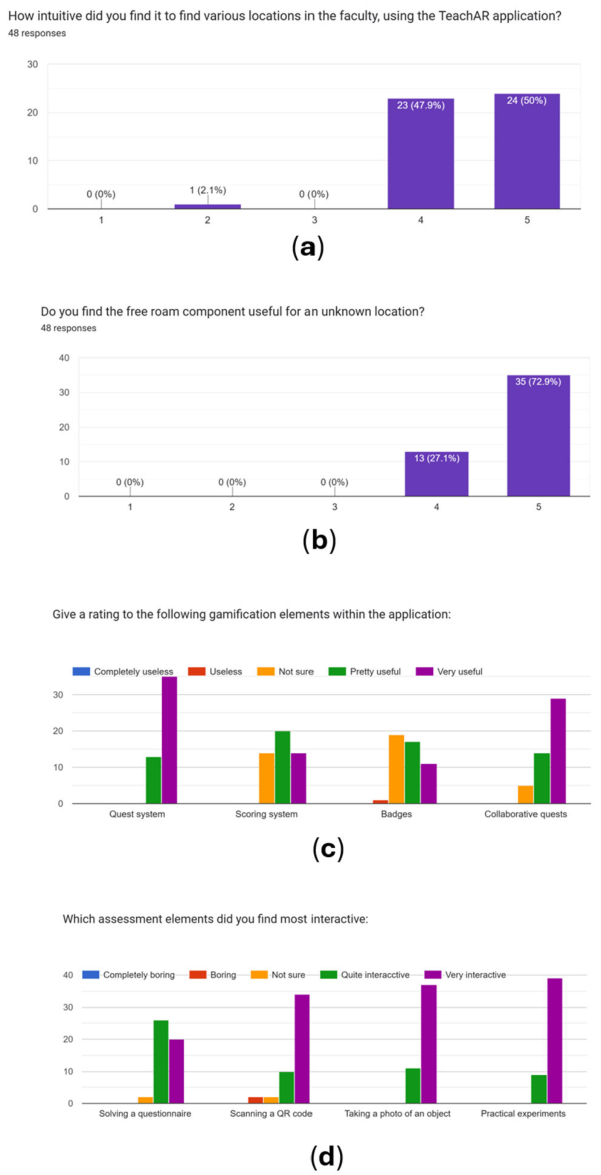


FIGURE 15. Results of the feedback form completed by participants in the pilot study. Navigation component feedback (a); free roam component feedback (b); used gamification component feedback (c); checkpoint validation feedback (d).

respondents finding the navigation component intuitive and very intuitive to use, and 100% of them considering the free roam functionality very useful when having to orient themselves in unknown locations. Regarding the gamification component of the system, 100% of the respondents considered the idea of integrating interactive quests into the learning process to be useful and very useful, while 43 out of 48 users enjoyed the idea of having to solve collaborative quests. The most enjoyed checkpoint validation method was the experiment. The results of the survey are presented in Fig. 15.

The form also included a free-answer question with suggestions for improvement from users. In general, most users

FIGURE 16. Results of the feedback form completed by teachers. Quest creation feedback (a); Classroom management feedback (b); Evaluation components feedback (c).

considered that audio guidance would be extremely useful, in addition to the augmented arrow. One user also suggested adding more text indications on the phone screen, and another suggested adding a map where they can see the entire journey.

During the pilot testing, three teachers were also asked to use the TeachAR system to define a quest. At the end of the process, they were asked to complete a form regarding the ease of quest definition as well as their general opinions about a gamified learning process using quests. The results of the form are shown in Fig. 16. In general, they have found the processes of adding quests and class management to be intuitive and easy. Regarding the preferred evaluation methods for checkpoint validation, teachers have considered quiz solving and practical experiments to be the most valuable methods.

The evaluation of the proposed system highlighted both its strengths and its limitations. Participants in the pilot study considered the application to be accurate, intuitive, and engaging. The gamification elements were particularly well received by the participants. At the same time, a series of technical limitations were noticed regarding the cloud anchor stability in environments with lacking or repetitive textures. Performance differences across devices were also observed. Important feedback was collected from the participants in the pilot study that will enable improvements in future versions of TeachAR. Among their suggestions, the most notable were: audio guidance, textual instructions, and a minimap with details of the building.

VI. CONCLUSION

This paper introduces TeachAR, an AR solution designed to facilitate indoor navigation, augmented content exploration, and gamified learning activities inside educational and cultural buildings. It offers a platform that requires no special hardware infrastructure, involving only regular desktop and mobile devices, with a simple enough setup workflow targeted at non-technical users.

The free-roam navigation, guided navigation, and quest system were individually tested. The pilot testing was successful, with the TeachAR system being considered by the study participants a promising solution for future students. However, a series of limitations were uncovered that should be addressed before releasing the system as a standalone product.

One problem encountered was previously discussed. The building administrator should have very accurate building plans, especially when it comes to the angles between the walls. Moreover, when placing the anchors in the real world, the administrator should pay particular attention when selecting the initial orientation. Otherwise, displacement errors might accumulate, and students may feel confused during navigation. The method used to resolve this issue was to reattach the entire map every few seconds to the closest detected anchor, thereby resetting the environment's pose relative to the anchor. Using this approach, the administrator must pay particular attention to the anchors added to the real world, ensuring they are not too far apart but also not too close to each other.

Another issue observed during the piloting of the system is the textures of the building. To successfully host and solve cloud anchors, the anchors should be surrounded by unique textures with no repetitive content. The lack of textures (i.e., white walls) or repetitive models will result in the impossibility of solving the anchor or placing the anchor in the wrong position.

Another issue encountered was the navigation inside the building, which involves using an elevator. Because, when inside an elevator, the image captured by the smartphone remains mostly unchanged, but the device's sensors detect an elevation change, the system is unable to estimate a proper position and orientation. As a result, all the content might appear randomly displaced when exiting the elevator. To resolve this issue, when leaving the elevator, the user must wait for a few seconds for the application to recalibrate. This is achieved by resetting the AR session. The recalibration process automatically starts when the phone detects the first anchor outside the elevator. As a result, the administrator should place an anchor on each floor in front of the elevator doors.

During the pilot testing, significant variations in application performance were observed across different smartphone models. As previously stated, the pilot testing was conducted using a Google Pixel 7 smartphone. However, initially, the pilot testing was designed to be implemented using two different devices: the Google Pixel 7 and a Xiaomi [66] device. Unfortunately, the detection and solving of cloud anchors were incredibly slow on the Xiaomi smartphone compared to the Google Pixel; thus, the user experience was negatively impacted. As a result, only the Pixel smartphone was ultimately used during the pilot study.

On the other hand, one positive aspect noticed during the pilot testing was the accuracy of the indoor localization algorithm when the environment suffers slight modifications. During the days when pilot testing was being conducted at the Faculty of Automatic Control and Computers, the PM Fair [67] annual event was taking place. This resulted, on the one hand, in rearranging the furniture in the entire main hall of the faculty and, on the other hand, in a large number of students walking through the main hallway. However, this did not affect the accuracy of the indoor localization algorithm, and the testers of the application experienced no particular issues navigating to their selected destinations.

Most of the presented limitations are, in fact, inherent to the AR Foundation technology itself (such as cloud anchor solving when there are repetitive textures or an insufficient amount of distinctive textures in the environment or significant variations in wait times across different devices). Variations in wait times were observed during the initialization process of the AR session. This resulted in increased wait times for the Xiaomi device in two situations: when the application starts running (before the proposed method of indoor localization is computed) and when the user exits the elevator (in this case, to reduce the positional drift cumulated inside the elevator, the AR session has to be reinitialized). One future approach to try to address these problems (for example the long wait times when using less performant devices for initializing and resetting the session, and the lack of textures or repetitive content) is to incorporate some markers or beacons to be detected by the system on the floor or on the wall. This way, an approximate location could be provided by the closest detected marker until the AR session has been completely initialized, or in places where the AR Foundation technology fails to properly solve the anchors. Other future research directions include incorporating the suggestions of participants who took part in the pilot testing, such as developing an audio guidance module, a complete map view of the entire building, route previsualization, or generating multiple navigation routes.

The TeachAR system was piloted within the Faculty of Automatic Control and Computers at the National University of Science and Technology POLITEHNICA Bucharest. During the testing process, both the desktop and the mobile applications were evaluated. Teachers from the faculty staff considered the quest-defining functionality to be intuitive, while students were very excited about the functionalities the mobile application offers them. This result is quite promising, and it demonstrates the efficiency of the TeachAR system.

Future work for TeachAR will also focus on further evaluating the system over an extended period by integrating it into existing university courses. This way, we can assess the effects of using the system in comparison to traditional learning methods. The study will involve two focus groups: an experimental group that will use the application, and a control group that will learn using traditional methods. Evaluation of students from both groups will be performed before, during, and after the experiment.

REFERENCES

[1] Unity Technologies. (2025). *AR Foundation*. Accessed: May 12, 2025. [Online]. Available: https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@5.0/manual/index.html
[2] Google. (2025). *ARCore Developer Guide*. Accessed: May 12, 2025. [Online]. Available: https://developers.google.com/ar/develop
[3] Apple Inc. (2025). *ARKit*. Accessed: May 12, 2025. [Online]. Available: https://developer.apple.com/augmented-reality
[4] Vuforia Developer Portal. (2025). *Vuforia Library*. Accessed: May 12, 2025. [Online]. Available: https://developer.vuforia.com/library
[5] EasyAR. (2025). *EasyAR Documentation*. Accessed: May 12, 2025. [Online]. Available: https://www.easyar.com/view/support.html
[6] MAXST. (2025). *MAXST AR SDK Documentation*. Accessed: May 12, 2025. [Online]. Available: https://developer.maxst.com/MD/doc/6_2_x/intro
[7] DeepAR. (2025). *DeepAR Documentation*. Accessed: May 12, 2025. [Online]. Available: https://docs.deepar.ai
[8] 8th Wall. (2025). *8th Wall Documentation*. Accessed: May 12, 2025. [Online]. Available: https://www.8thwall.com/docs
[9] Kudan. (2025). *Kudan AR SDK*. Accessed: May 12, 2025. [Online]. Available: https://www.xlsoft.com/en/products/kudan/index.html
[10] Microsoft. (2025). *Mixed Reality Toolkit (MRTK) for Unity*. Accessed: May 12, 2025. [Online]. Available: https://learn.microsoft.com/en-us/windows/mixed-reality/mrtk-unity/mrtk2/?view=mrtkunity-2022-05
[11] Snap Inc. (2025). *Lens Studio Documentation*. Accessed: May 12, 2025. [Online]. Available: https://developers.snap.com/lens-studio/4.55.1/references/guides/overview
[12] Meta. (2025). *Spark AR Studio*. Accessed: May 12, 2025. [Online]. Available: https://spark.meta.com/blog/meta-spark-announcement
[13] Niantic. (2025). *Lightship ARDK Documentation*. [Online]. Available: https://lightship.dev/docs/maps
[14] echo3D. (2025). *Echo3D Documentation*. Accessed: May 12, 2025. [Online]. Available: https://docs.echo3d.com
[15] C. E. Mendoza-Ramírez, J. C. Tudon-Martinez, L. C. Félix-Herrán, J. D. J. Lozoya-Santos, and A. Vargas-Martínez, "Augmented reality: Survey," *Appl. Sci.*, vol. 13, no. 18, p. 10491, Sep. 2023, doi: 10.3390/app131810491.
[16] P. Sirohi, A. Agarwal, and P. Maheshwari, "A survey on augmented virtual reality: Applications and future directions," in *Proc. 7th Int. Conf. Inf. Technol. Trends (ITT)*, Nov. 2020, pp. 99–106.
[17] A. S. Shewail, N. A. Elsayed, and H. H. Zayed, "Survey of indoor tracking systems using augmented reality," *IAES Int. J. Artif. Intell.*, vol. 12, no. 1, pp. 402–414, Mar. 2023, doi: 10.11591/ijai.v12.i1.pp402-414.
[18] H. Lynam, S. Dascalu, and E. Folmer, "Augmented reality navigation: A survey," *Int. J. Hum.-Comput. Interact.*, vol. 41, no. 16, pp. 1–17, Aug. 2025, doi: 10.1080/10447318.2024.2431757.
[19] A. Martin, J. Cheriyan, J. Ganesh, J. Sebastian, and V. Jayakrishna, "Indoor navigation using augmented reality," *EAI Endorsed Trans. Creative Technol.*, vol. 8, no. 26, Mar. 2021, Art. no. 168718, doi: 10.4108/eai.17-2-2021.168718.
[20] P. Herbers and M. König, "Indoor localization for augmented reality devices using BIM, point clouds, and template matching," *Appl. Sci.*, vol. 9, no. 20, p. 4260, Oct. 2019, doi: 10.3390/app9204260.

[21] R. S. Naser, M. C. Lam, F. Qamar, and B. B. Zaidan, "Smartphone-based indoor localization systems: A systematic literature review," *Electronics*, vol. 12, no. 8, p. 1814, Apr. 2023, doi: 10.3390/electronics12081814.

[22] R. Vishwakarma, R. B. Joshi, and S. Mishra, "IndoorGNN: A graph neural network based approach for indoor localization using WiFi RSSI," in *Proc. Int. Conf. Big Data Anal.*, Dec. 2023, pp. 150–165.

[23] A. Morar, A. Moldoveanu, I. Mocanu, F. Moldoveanu, I. E. Radoi, V. Asavei, A. Gradinaru, and A. Butean, "A comprehensive survey of indoor localization methods based on computer vision," *Sensors*, vol. 20, no. 9, p. 2641, May 2020, doi: 10.3390/s20092641.

[24] F. Zafari, A. Gkelias, and K. K. Leung, "A survey of indoor localization systems and technologies," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2568–2599, 3rd Quart., 2019.

[25] H. Belghit, A. Bellarbi, N. Zenati, and S. Otmane, "Vision-based pose estimation for augmented reality : A comparison study," 2018, *arXiv:1806.09316*.

[26] S. Hou, F. Lin, Y. Huang, Z. Peng, and B. Xiao, "Mobile augmented reality framework with fusional localization and pose estimation," 2025, *arXiv:2501.03336*.

[27] H. Taira, M. Okutomi, T. Sattler, M. Cimpoi, M. Pollefeys, J. Sivic, T. Pajdla, and A. Torii, "InLoc: Indoor visual localization with dense matching and view synthesis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 7199–7209.

[28] M. D. Friske, "Integration of augmented reality and mobile robot indoor SLAM for enhanced spatial awareness," 2024, *arXiv:2409.01915*.

[29] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *Proc. IEEE Int. Conf. Robot. Autom.*, Shanghai, China, May 2011, pp. 3400–3407.

[30] F. Zafari, I. Papapanagiotou, M. Devetsikiotis, and T. Hacker, "An iBeacon based proximity and indoor localization system," 2017, *arXiv:1703.07876*.

[31] Apple Inc. (2025). *IBeacon*. Accessed: May 18, 2025. [Online]. Available: https://developer.apple.com/ibeacon/

[32] A. Khaldi, R. Bouzidi, and F. Nader, "Gamification of e-learning in higher education: A systematic literature review," *Smart Learn. Environments*, vol. 10, no. 1, p. 10, Jan. 2023, doi: 10.1186/s40561-023-00227-z.

[33] Duolingo. (2025). *Duolingo Language Learning*. Accessed: May 18, 2025. [Online]. Available: https://www.duolingo.com/

[34] R. Smiderle, S. J. Rigo, L. B. Marques, J. A. P. De Miranda Coelho, and P. A. Jaques, "The impact of gamification on students' learning, engagement and behavior based on their personality traits," *Smart Learn. Environments*, vol. 7, no. 1, p. 3, Dec. 2020, doi: 10.1186/s40561-019-0098-x.

[35] M. Khoshnoodifar, A. Ashouri, and M. Taheri, "Effectiveness of gamification in enhancing learning and attitudes: A study of statistics education for health school students," *J. Adv. Med. Educ. Prof.*, vol. 11, no. 4, pp. 230–239, 2023, doi: 10.30476/jamp.2023.98953.1817.

[36] Ç. Öztürk and Ö. Korkmaz, "The effect of gamification activities on students' academic achievements in social studies course, attitudes towards the course and cooperative learning skills," *Participatory Educ. Res.*, vol. 7, no. 1, pp. 1–15, Mar. 2020, doi: 10.17275/per.20.1.7.1.

[37] M. Hodds, "The early impact of the COVID-19 pandemic on mathematical competencies on entry into a U.K. university," *Teaching Math. Appl., Int. J. IMA*, vol. 40, no. 4, pp. 254–262, Dec. 2021, doi: 10.1093/teamat/hrab021.

[38] R. M. Ryan and E. L. Deci, "Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being," *Am. Psychol.*, vol. 55, pp. 68–78, Jan. 2000.

[39] L. S. Vygotsky, *Mind in Society: The Development of Higher Psychological Processes*. Cambridge, MA, USA: Harvard Univ. Press, 1978.

[40] R. E. Mayer, "Cognitive theory of multimedia learning," in *The Cambridge Handbook of Multimedia Learning*, 2nd ed., Cambridge, U.K.: Cambridge Univ. Press, 2005, pp. 31–48.

[41] Y. Koumpouros, "Revealing the true potential and prospects of augmented reality in education," *Smart Learn. Environments*, vol. 11, no. 1, p. 2, Jan. 2024, doi: 10.1186/s40561-023-00288-0.

[42] A. D. Ramadhan, A. E. Permanasari, and S. Wibirama, "Gamification opportunity in augmented reality-based learning media: A review," in *Proc. 2nd Int. Conf. Intell. Cybern. Technol. Appl. (ICICyTA)*, Bandung, Indonesia, Dec. 2022, pp. 117–122, doi: 10.1109/ICICyTA57421.2022.10037922.

[43] G. Lampropoulos, E. Keramopoulos, K. Diamantaras, and G. Evangelidis, "Augmented reality and gamification in education: A systematic literature review of research, applications, and empirical studies," *Appl. Sci.*, vol. 12, no. 13, p. 6809, Jul. 2022, doi: 10.3390/app12136809.

[44] M. Cai, G. Akcayir, and C. D. Epp, "Exploring augmented reality games in accessible learning: A systematic review," *Educ. Sci.*, vol. 14, no. 3, p. 307, 2024, doi: 10.3390/educsci14030307.

[45] F. Çelik and C. Y. Ersanlı, "The use of augmented reality in a gamified CLIL lesson and students' achievements and attitudes: A quasi-experimental study," *Smart Learn. Environments*, vol. 9, no. 1, pp. 12027–12045, Nov. 2022, doi: 10.1186/s40561-022-00211-z.

[46] E. Ivarson, V. Erlandsson, M. Faraon, and S. Khatib, "Augmented reality and gamification in higher education: Designing mobile interaction to enhance students' motivation and learning," *E-Learn. Digit. Media*, pp. 1–20, Mar. 2024, doi: 10.1177/20427530241239981.

[47] M. Weerasinghe, A. Quigley, K. C. Pucihar, A. Toniolo, A. Miguel, and M. Kljun, "Arigato: Effects of adaptive guidance on engagement and performance in augmented reality learning environments," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 11, pp. 3737–3747, Nov. 2022, doi: 10.1109/TVCG.2022.3203088.

[48] C. H. Godoy, "Augmented reality and gamification: A framework for developing supplementary learning tool," *Int. J. Comput. Sci. Res.*, vol. 5, pp. 595–612, Feb. 2021, doi: 10.25147/ijcsr.2017.001.1.63.

[49] A. Gill, D. Irwin, D. Towey, Y. Zhang, B. Li, L. Sun, Z. Wang, W. Yu, R. Zhang, and Y. Zheng, "Effects of augmented reality gamification on students' intrinsic motivation and performance," in *Proc. IEEE Int. Conf. Teaching, Assessment Learn. Eng. (TALE)*, vol. 9, Auckland, New Zealand, Nov. 2023, pp. 1–8, doi: 10.1109/TALE56641.2023.10398240.

[50] Google. (2025). *ARCore Cloud Anchors*. Accessed: May 24, 2025. [Online]. Available: https://developers.google.com/ar/develop/unity-arf/cloud-anchors/developer-guide

[51] Google. (2025). *ARCore API*. Accessed: May 24, 2025. [Online]. Available: https://console.cloud.google.com/apis/library/arcore

[52] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Math.*, vol. 1, no. 1, pp. 269–271, Dec. 1959, doi: 10.1007/bf01386390.

[53] Unity Technologies. (2025). *Unity Game Engine*. Accessed: May 28, 2025. [Online]. Available: https://unity.com/

[54] Google. (2025). *ARCore Extensions for AR Foundation*. Accessed: May 28, 2025. [Online]. Available: https://developers.google.com/ar/develop/unity-arf/getting-started-extensions

[55] MongoDB. (2025). *MongoDB Atlas*. Accessed: May 28, 2025. [Online]. Available: https://www.mongodb.com/atlas

[56] W. Khan, T. Kumar, C. Zhang, K. Raj, A. M. Roy, and B. Luo, "SQL and NoSQL database software architecture performance analysis and assessments—A systematic literature review," *Big Data Cognit. Comput.*, vol. 7, no. 2, p. 97, May 2023.

[57] Amazon Web Services. (2025). *RESTful API*. Accessed: May 29, 2025. [Online]. Available: https://aws.amazon.com/what-is/restful-api/

[58] ZXing.Net. (2025). *ZXing.Net GitHub Repository*. Accessed: May 29, 2025. [Online]. Available: https://github.com/micjahn/ZXing.Net/tree/master

[59] Google. (2025). *ARCore Augmented Images*. Accessed: May 29, 2025. [Online]. Available: https://developers.google.com/ar/develop/augmented-images

[60] Univ. POLITEHNICA Bucharest. (2025). *Faculty of Automatic Control and Computers*. Accessed: May 29, 2025. [Online]. Available: https://acs.pub.ro/

[61] Univ. POLITEHNICA Bucharest. (2025). *Faculty of Industrial Engineering and Robotics*. Accessed: May 29, 2025. [Online]. Available: https://www.fiir.upb.ro/index.php/ro/

[62] Meta. (2025). *Meta Quest 3*. Accessed: May 30, 2025. [Online]. Available: https://www.meta.com/quest/quest-3/

[63] A. Morar, M.-A. Bălutoiu, A. Moldoveanu, F. Moldoveanu, and A. Butean, "CultReal—A rapid development platform for AR cultural spaces, with fused localization," *Sensors*, vol. 21, no. 19, p. 6618, Oct. 2021.

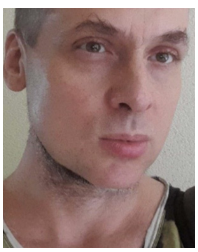
[64] A. Morar, M. A. Balutoiu, A. Moldoveanu, F. Moldoveanu, A. Butean, and V. Asavei, "Evaluation of the ARCore indoor localization technology," in *Proc. 19th RoEduNet Conf., Netw. Educ. Res. (RoEduNet)*, Bucharest, Romania, Dec. 2020, pp. 1–5.

[65] Google. (2025). *Google Pixel Phones*. Accessed: May 30, 2025. [Online]. Available: https://store.google.com/category/phones

[66] Xiaomi. (2025). *Xiaomi Phones*. Accessed: Jun. 2, 2025. [Online]. Available: https://www.mi.com/global/phone/
[67] Facebook. (2025). *PM Fair*. Accessed: Jun. 2025. [Online]. Available: https://www.facebook.com/p/Proiectarea-cu-Microprocesoare-100067756107295/

MARIA ANCA BĂLUŢOIU received the Graduate and master's degrees in graphics, multimedia, and virtual reality from the Faculty of Automatic Control and Computers, National University of Science and Technology POLITEHNICA Bucharest, in 2018 and 2020, respectively, where she is currently pursuing the Ph.D. degree. Since then, she has been a Teaching Assistant and a Researcher of various research projects (Lib2Life, CultReal, SoV Lite, and VIP SpaceNav). She is teaching computer graphics, software engineering, data structures and algorithms, virtual and augmented reality, procedural generation, and e-health systems.

ANCA MORAR received the B.S. degree in computer science from the POLITEHNICA University of Bucharest, in 2009, and the Ph.D. degree in computer science, in 2012, with a focus on medical image analysis and visualization. She is currently a Full Professor with the Computer Science and Engineering Department, Faculty of Automatic Control and Computers, POLITEHNICA University of Bucharest. Her research interests include computer graphics, GPGPU, computer vision, and e-health.



ALIN MOLDOVEANU is currently a Professor and the Vice-Dean of the Faculty of Automatic Control and Computers, POLITEHNICA University of Bucharest, where he completed as a Valedictorian some 25 years ago. He is teaching virtual and augmented reality at the master's and software engineering at bachelor's levels. He is in charge of the 25 master's programs of the faculty. His focus is on applied research using VR and AR (exploring and applying immersion, sensory substitution, and distorted reality), mainly in e-health, e-learning, and e-culture. As a Leader of the 3DUPB Research Team, he was/is the Director or a Team Leader for many national or European research projects in these areas, such as Sound of Vision, TRAVEE, HAI-OPS, and Lib2Life. His works received prestigious prizes, such as the Best "Tech for Society" Horizon 2020 Project awarded by European Commission through Innovation Radar at ICT 2018–received by the Sound of Vision (project conducted as a Technical Coordinator and UPB Team responsible).

ALEXANDRU GRĂDINARU received the Ph.D. degree in computer science from the University of POLITEHNICA Bucharest, in 2019, with a focus on educational virtual environments. Then, he held a postdoctoral fellowship in generative computer graphics. He is currently a Lecturer with the University of POLITEHNICA Bucharest and the University of Padua. He was actively involved in multiple research projects and has hands-on experience in academia, the industrial environment, and entrepreneurship. His research and teaching interests include game development, XR, e-learning, e-health, computer graphics, cybersecurity, software engineering and architecture, DevOps, and cloud-based application development.

VICTOR ASAVEI (Member, IEEE) received the Ph.D. degree in computer science and information technology from the POLITEHNICA University of Bucharest, in 2011. He is currently an Associate Professor with the Computer Science and Engineering Department, Faculty of Automatic Control and Computers, POLITEHNICA University of Bucharest. He has co-authored approximately 90 articles and four books in the fields of computer graphics, distributed computing, software engineering, and medical IT. He has participated in numerous national and international research projects. His current research interests include real-time computer graphics and general-purpose computing on graphics processing units (GPGPU).

MAGDALENA ANGHEL received the Graduate degree from the Faculty of Automation and Computers, National University of Science and Technology POLITEHNICA Bucharest, in 1997, and the Ph.D. degree in systems engineering, in 2012. Currently, she is an Associate Professor with the Automation and Industrial Informatics Department, National University of Science and Technology POLITEHNICA Bucharest. Her research interests include e-health (medical devices and assistive devices), virtual reality, reliability and diagnosis, measurements, and transducers. She is a member of the 3D UPB Research Group (https://3d.upb.ro/team.html).

CRISTIAN LAMBRU received the B.S. degree in informatics from the University of Bucharest, in 2015, and the master's and Ph.D. degrees in computer graphics, multimedia, and virtual reality from the POLITEHNICA University of Bucharest, in 2017 and 2022, respectively. He is currently an Assistant Professor with the Computer Science and Engineering Department, Faculty of Automatic Control and Computers, POLITEHNICA University of Bucharest. His research interests include computer graphics, virtual reality, and real-time global illumination.