4 CollabAR – Architectural Design

The following chapter will discuss the architectural design of CollabAR, the educational software developed to explore the research topic of this thesis and to gather data related to the research questions formulated. This and the following chapters are dedicated to the technical identity of this research and will discuss the development process followed, the technical analysis done to accommodate the educational context of the development, the technical issues found through the design and implementation stages, the results obtained and the path forward for future implementations.

This chapter will focus on the description and analysis of the conception and design stages of CollabAR, a prototype implementation and proof of concept of an AR technology for education. Specifically, the intention of this chapter is to discuss three main points:

* The architectural design of the software, the challenges that were present in the context of the research questions and the different solutions proposed to tackle them.
* The technology integrated into the proposal and used for the development process.
* The changes in approach and solutions that had to be made in response to the learning design context, the challenges posed by the selected technology and the data gathered during the testing stages.

As stated in the first chapter of this document, the educational pillar of the research has never been far from its software development counterpart. This is especially important in the conception-design-implementation cycle used for the development of CollabAR. It was essential to stablish the proper motivation behind the software development in terms of educational research, its broader position in the field of Technology Enhanced Learning and its more particular purpose in the context of the case study selected for this project.

Chapter 3 offered a detailed explanation of the research questions that motivate the construction of a specialized software. To summarize that information, this thesis pretends to identify what elements, unique to AR technologies, can be used to enhance the classroom experience by offering benefits that are different or comparable to other technologies. Through an extensive review of the literature, which was detailed in section 2.5, four main factors that distinguish AR from other technologies emerged:

* Visualizing and interacting with abstract or difficult to come by concepts and objects.
* The ability to keep and incorporate the immediate context as part of the experience.
* A mobile-first deployment philosophy that eases development to an extent and proves to be more accessible to the end user.
* Easy integration with other tools in the mobile environment like cameras, movement sensors, GPS and ubiquitous connectivity.

The literature review showed that there is an increasing tendency to develop AR applications that leverage the capabilities of a mobile hardware. Access to diverse sensors detect the context and environment of the user and powerful image recognition tools are key to the AR experience. Nonetheless, most educational apps are single-user experiences, which undermines the equally powerful connectivity tools provided by the mobile ecosystem.

Multi-user AR experiences (educational or not) excel the most at co-located, in-person activities. The use of GPS-based augmentation also gives strength to this tendency, building connectivity on top of a different framework in the mobile network infrastructure and prioritizing the sense of sharing a common space rather than sharing an abstract connection with other users.

With this information it is possible to stablish one of the main motivations for the development of CollabAR: to create an augmented reality application made for a multi-user interaction that explores the technology’s capabilities for digital visualization, context awareness, seamless connectivity and a strong link to the physical space shared by the users.

Shifting now to the educational perspective of the project, section 2.5 was used to provide an analysis of the TEL field. We identified two problems that proved to be the most relevant for this project.

First, there is a clear trade-off when adapting any technology in the classroom between the benefits obtained and the issues that are related to the implementation, usage and management of the technology. In this relationship between technological development and educational design it is important to consider all the instances in which one aspect informs the other, and it is even more important to be aware of all the issues that arise when there is a mismatch between both approaches.

The literature review showed a lot of instances in which a technological development was poorly received. Many studies pointed the dangers and inefficiencies of a particular technology in learning scenarios, and it is possible to identify a push towards reconsidering the role of information technologies in the classroom (Boonmoh et al., 2021, Bećirović, 2023, Ifinedo & Kankaanranta, 2021).

The second area of motivation for CollabAR is to explore the synergy between the developer and the educator. Hopefully, the documentation of the design and development process followed for CollabAR provide valuable insight of the challenges of creating educative technology, to the point of becoming useful for future projects in similar scenarios. A good step forward for this project is for the positive experiences to get polished and the negatives to get analysed and discussed in future work.

These are the research goals that guide the development of CollabAR: A multi-user AR interactive experience, an educational tool that leverages the advantages inherent to AR while mitigating its disadvantages and a design and implementation process that is educational in nature, meant to aid in the achievement of learning goals by providing a tool for students and educators.

Section 4.1 of this chapter will provide a general description of the context in which CollabAR is developed and the broad requirements that were identified. Section 4.2 will show a more detailed description of the use cases and scenarios designed for the tool and their relation to the overall research goals of this project. Sections 4.3 is dedicated to the architectural design of the software, focused on the high-level decisions taken to accommodate the learning goals present in the development context and in the research objectives. Finally, section 4.4 offers a broad summary of the results obtained and the conclusions reached from the analysis process made throughout the chapter.

# 4.1 Project Description

CollabAR is born form the desire of experiment with AR technologies in the field of education. Section 2.2 in the literature review chapter explores in detail the process of identifying and describing how AR has been used in the past as an interesting medium for students to interact with educational material. It was discussed that there is value in offering experiences built around immersion, the exploration of visually engaging content and the possibility to augment the context of the learning environment of a student.

From exploring the literature, it was possible to extract the idea that AR development was not taking advantage of the connectivity tools available in the mobile-based deployment environment that is commonly used, and that experiences build around such connectivity showed very positive results and interesting pathways for future research. With this context in consideration is that a software development using AR interactions in a multi-user environment is proposed as an avenue to gather data and build a case study that explores the use of technology that supports collaboration as a teaching tool.

By setting the development in the proper scenario, it is possible to analyse how students use AR technologies as an aid for the collaborative aspects of their projects. It would be of interest to identify what problems are the students solving or finding, how the technology shapes the structure of their work, the relationships between peers or the development of their learning process. By comparing these observations with the behaviours of other students in the same scenario, it would be possible to identify key behavioural changes promoted by the implemented tool, opening the possibilities for further analysis and pathways for future development.

In general, the development of CollabAR aims to fulfil three main objectives, which are independent of the learning goals proposed by the case study context. These objectives will be labelled as Observation Objectives (as explained in chapter 3).

**OO1. Visualizing my team**. Sections 2.1 and 2.5 of the literature review exposed the challenges of teaching collaboration skills, and described the process in terms of the time, guidance an exposure to proper tools needed to succeed. CollabAR is meant to use AR as a way to graphically convey the information needed to manage my group, for stablishing a discussion or for organizing a conversation. The idea is to offer tools that convey a good grasp of who are my companions and what are they doing.

**OO2. Visualizing decisions.** CollabAR should provide the ability to show digital representations of the learning material and allow the user to hold a discussion around that. The discussion can derive in solidifying knowledge or building learning, or could help in managing decisions, debates with peers or giving a visual representation to the workflow followed by the group. It should be possible to determine from the observations if interacting as a group with a digital representation of the learning material promotes any change on how the group is learning, how the material itself is being used or if the augmentation helps at managing the learning activity or not.

**OO3. Augmenting the workspace.** CollabAR is initially envisioned as a tool for presential collaboration with a physical shared space where the students are working. The tool should be able to use that space as an anchor for the visualizations and use the capabilities of AR to give extra meaning to that physical context, either by identifying and using elements of the environment in the augmentation or by adding digital elements anchored to the space that can help the user to create such meaning. In terms of research goals, we want to observe if the ability to augment, visualize and customize digitally the physical environment of a group helps with aspects related to team identity, productivity or team management.

So far, these objectives are devoid of context, their aim is to support the research questions proposed for this project, specially SQ1 to SQ3. The objectives describe activities common in any collaborative situation, and the insights we pretend to obtain through the observation of the student’s work can provide answers to questions related to the value that AR can offer to education and to the particularities of the case study. The main hypothesis does state that AR offer advantages using clever visualizations and quick interactions with the context, and CollabAR is proposed to offer such scenarios.

## 4.1.1 Context: Brief Description of Built Environments Industry Project Course

Industry Project (ENBU 79X/89X in the university’s enrolment system) is the final course offered for a variety of programs under the Built Environments department at Auckland University of Technology (AUT). This department includes the undergraduate programs of Architecture, Construction and Architectural Engineering, Civil Construction and Spatial Design.

ENBU 79X/89X is described as a multi-disciplinary course focused on providing an industry project experience that bridges the academical studies of the program and the professional career of the students. In educational terms, this is a project-based course, providing the students a year-long case study that they must work out as a team to provide a solution design, a construction plan and an estimation of costs. The course does not provide a conventional curriculum or study plan, as it is mostly devoid of content. Instead, the course acts more as a final opportunity to assess the skills and knowledge gained in previous courses, akin to a research project or thesis. Nonetheless, the course has very clear learning goals:

* Propose, organize, design, execute and reflect on a multi-disciplinary project.
* Apply existing knowledge and skills.
* Demonstrate initiative and independence in decision making scenarios.
* Deliver comprehensive client and progress reports.

The course has an overall focus on giving students opportunities to develop crucial soft skills related to teamwork, communication, planning and management. There is also a substantial emphasis in in developing the relationship between client and team in terms of communication, progress and stakeholders. All of the assessments of the course move around the idea of reporting progress and decisions to the client. All the elements of the project involve a two-way communication to understand views and requirements about relevant policies, possible designs and bits of information needed for the proposals.

The course also promotes a strong multi-disciplinary approach. Four different roles are represented from the different bachelors that are part of the Built Environments department: Architectural Engineers, Construction Engineers, Construction Managers and Quantity Surveyors. Each role represents an important aspect of the project, and every team is composed of at least one representative of each role. Every role had a particular set of goals inside the project and works towards specific learning outcomes. The project is built around constant communication withing the group, distribution of tasks and effective collaboration to succeed.

In summary, ENBU 79X/89X is a course that focuses on developing crucial soft skills on students in the context of project development and management, emulating in different ways common processes and scenarios found on the construction industry. In general, the learning goals can be enunciated as:

1. Develop, document and communicate the design, plan and cost estimation of a construction project, considering the expectations of the client and the limitations exposed by the context, relevant policies and local constraints.
2. Participate in and contribute to the project as part of a team, considering the expectations of your role as well as the processes and tools agreed upon by the group.
3. Create and use the proper tools to communicate decisions to the client, report progress, receive feedback and ask for information.

The course provided other set of particularities and complexities that made it a very interesting case study to explore in detail, all that information can be found on Chapter 5. For this section, this is enough context to position the development of CollabAR. Needless to say, the course provided an ideal scenario to test goals associated with teamwork and collaborative learning, as well as to provide the opportunity to have evidence of this process in all the stages of development of both a collaborative project and a course, not only in an individual assessment or lesson.

With this context at hand, we can position CollabAR as an educational technology development meant to offer support to the students of the Industry Project course in the development of their coursework and the management of their team. It is possible then to elaborate on the educational and development goals of CollabAR in relation to the research questions of this thesis, which is the objective of the following sections.

## 4.1.2 Educational Goals

The expected learning outcomes for the Industry Project course highlight teamwork and communication, and as such, the intervention designed around CollabAR should focus on identifying the tools that AR technologies can provide to facilitate or boost that collaborative process. The aim should be in facilitating group management, communication among members and aiding in the production of knowledge and materials for the project.

The guiding prompt of CollabAR is to provide students with a digital environment that helps in visualizing the activities performed by the team and to highlight possible tools at their disposal for proper team management, discussion of ideas and effective decision making.

By aligning these goals with those of course, CollabAR can provide a tangible feel of abstract concepts used in team management like participation, ideas and task allocation. It can also reinforce in the students the need to develop a strategy to deal with issues related to groupwork, it can prompt ideas about self-reflection, evaluation of performance and in the development of each individual role inside the group.

The main goal can also be aligned to research questions SQ1 through SQ3. The case study around ENBU 79X/89x provides an ideal observation environment to compare different approaches to collaboration, not only with those using AR or not, but also behaviours related to the use of different types of technologies and to the circumstances fostered by the design of the course.

Since the case study is complex and strongly based on observations, three scenarios have been selected to focus the development of CollabAR and to simplify and direct observations, comparisons and analysis, making the data gathering process manageable for the timeframe of the research.

These scenarios represent the general framework used to provide context to more specific and goal driven use cases in the future, which will be exposed in section 4.3. The scenarios are born from the learning goals of the course but not necessarily from the particularities of its development. They represent common instances of collaborative work and common roadblocks in the development of effective teamwork (check section 2). A deeper analysis of the implementation of the tool is described in chapter 5, and a description of the changes and adaptations needed to properly incorporate CollabAR as part of the observed cohort of ENBU 79X/89X can be found in chapter 6.

### 4.1.2.1 Scenario 1: Team Management

The first scenario is about tools that can be used to manage the composition and dynamics of a group and to help in answering questions like: Who is my team? Which are their roles? Which is my role? (Bannister et al., 2014, Delgado Piña et al., 2008).

The proposal is to use visual cues and digital constructs to represent information about your teammates and their work, offer the means to interact with such information and to share it with others. This *digital profile* can help students to start a conversation around the structure of the group and the evolution in the dynamics of its members. The same reasoning can be applied to give a visual anchor to the tasks allocated and roles assigned.

A specific manifestation of this scenario is the first contact the members of the group have. CollabAR can help in that first process of identifying who is who, to create the proper environment dialog and to stablish the first rules of communication. It can also be extended to scenarios such as a new member trying to integrate into the group or as a management tool to experiment and consolidate the group’s structure.

CollabAR can also provide a digital presence for the team itself, and represent information about the work being done and who is doing it. This *digital presence* can be defined in terms of the physical space where the group is meeting. This will serve as the positional anchor for all the digital constructs that can be used as part of the team’s activity. A digital room can be created for the team members to *enter* or *connect* which would signify the disposition to start working. The room will be constrained or superimposed to the physical space where the group is gathered and can be composed by the digital constructs the team is using to work and all the physical elements the application could recognize.

This is a good moment to explain the first design constraint taken for this project. CollabAR will provide support only to face-to-face collaboration. Online or asynchronous collaboration won’t be considered.

In section 2.1 and 2.3 of the literature review it was exposed the importance of network technologies in the analysis of collaborative learning and the expansive field of online collaboration. The opportunities of exploring the connectivity capabilities of mobile devices as part of AR application was also part of the discussion, and a big focus of this development is to explore such possibilities and to leverage the ubiquitous connectivity offered by mobile devices to power seamless connectivity between users.

It is also true that this research wants to understand the unique capabilities provided by AR technologies in terms of visualization and augmentation of the physical space, a scenario that adapts easily to face-to-face interaction and connectivity with users in proximity. In no way this means that such elements cannot be together with online interactions, but they do add another degree of complexity to both the research and the development of the software, risking the possibility of taking the spotlight form the main objective of understanding collaborative learning and the influence of AR in the process.

To conclude, Scenario 1 proposes a solution in which the team and each of the team members have a digital profile that can be visualized and interacted with. The information in the profile can be used as the main anchor for the software to provide the visualization and augmentation tools that aid in the process of team management, communication, roles, task allocation and team structure.

A visual representation of all these concepts can be checked in Figure 1, where the room represent both the physical space used by the group to work and the current instance or session the group is participating in. The room will hold all the digital objects used during the session as well as framing all the other activities the team can perform.

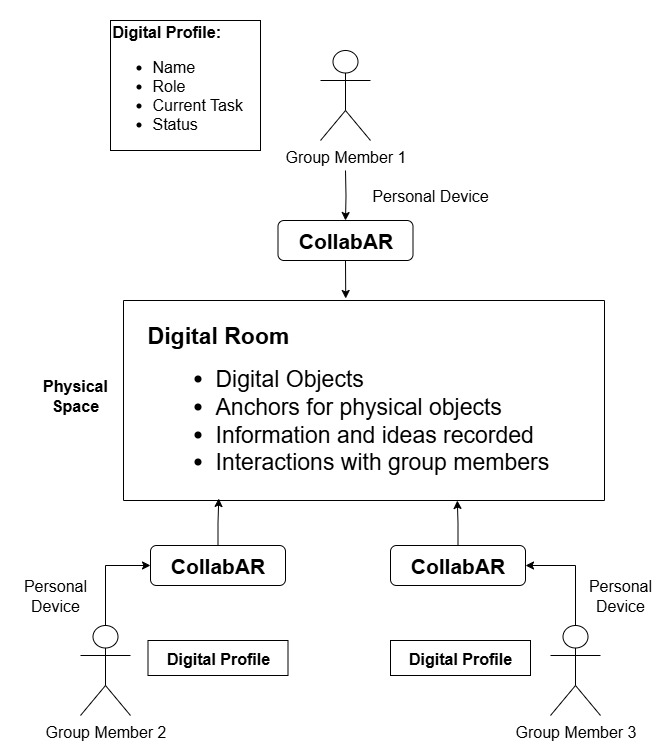


Figure 1 Team Management Scenario

### 4.1.2.2 Scenario 2: Work Session Management

For this scenario we can define a work session as an activity the group decides to execute for a given time with a set of goals in mind. It can also be described as the group members participating in a lesson or a learning activity. The session would be composed by participants, the space the session is taken place and the goals proposed.

For this purpose, it is envisioned that CollabAR can provide tools to identify and manage aspects like:

* The participants in the work session.
* Status of the session.
* Current objectives and the status of those objectives.
* The resources at hand for the session.

Scenario 1 is about giving the room a digital representation that can be interacted with. Scenario 2 uses those elements to construct a digital representation of the work session, as shown in Figure 2. The participants will connect to an active room in which a session is taking place. The room will be associated with a set of goals, active resources and a physical space to which is anchored.

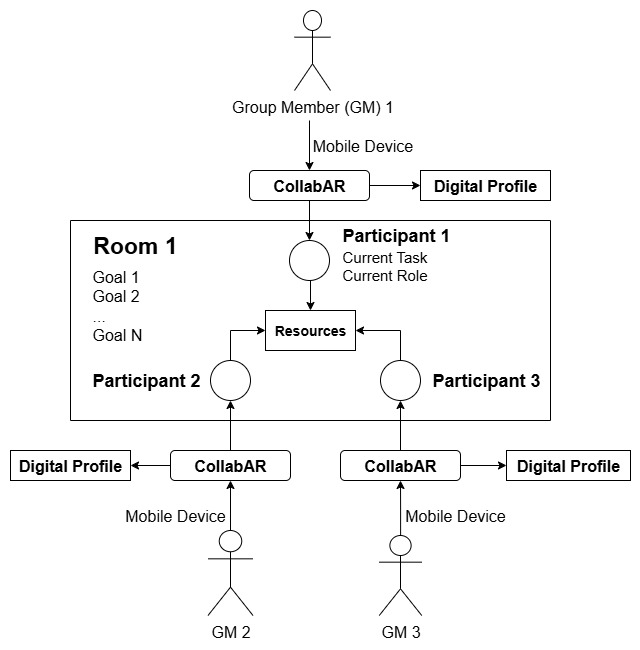


Figure 2 Work Session Management

The second constraint in development that can be analysed at this stage is related to resources. Ideally, CollabAR should be able to provide or interact with all the resources the group needs to perform their work. If we just analyse the context of ENBU 79X/89X, the list of available resources could comprise elements such as:

* Tools for creating documents and presentations.
* Tool for architectural design and analysis.
* Spreadsheets for calculations and project management.
* Visualization of documents and design proposals.
* Tools for digital modelling and design.

It is not feasible for the scope of the development to consider all the possibilities. Seamless interaction between tools is a valuable quality functionality but would not add significant insight into the research goals at hand. In contrast, observing how the students interact and integrate all the technological tools at their disposal, and how they evaluate their usefulness and functionalities, does aligns with the observation objectives of the research, providing insights for questions SQ4 and SQ5, where we are trying to identify the specific value of AR in learning scenarios, especially in comparison to other tools.

CollabAR will be limited to two resources that can be used as part of a working session:

1. Visualization and interaction of simple digital models for planification and discussion.
2. Anchoring data or information to a digital object or physical space.

These two resources offer general help for the most common activities of the course: proposing solutions to the spatial design of the project and organizing discussions around such proposals. The purpose is to analyse how the students change (or not) their approach to collaborative work when they have access to more *tangible* representations of abstract concepts as ideas, proposals and discussion, and how, if any, the discussion is affected by the ideas being more visually anchored to the real world.

### 4.1.2.3 Scenario 3: Discussion and Debate Management

Scenario 3 relates to the activities that can be performed during a work session but that are not related to direct management of the group. CollabAR aims to offer tools for discussions and debates around the work being done by the group and the decisions being taken. The tool can give visualization and structure to elements like:

* The object of debate/discussion.
* The structure and flow of the discussion.
* The responses and ideas generated.
* The active participation of different team members.

The proposal is that AR can offer a visual anchor or structure to the process, helping students in understanding it and managing its flow.

To summarize, the last three scenarios illustrated the main learning objectives guiding the design of CollabAR. The objective is to provide a tool that helps the students to identify and structure the most relevant aspects of their collaborative work such as team management, work management, communication and decision making.

## 4.1.3 Development goals

It is also possible to identify a set of points related to the technological and software development of CollabAR that are worth analysing and that can offer insights on how the technology influence the achievement of the learning goals. Some constraints in the development process can also be discussed related to issues highlighted in this section.

The first development goal is concerned with the multi-user AR experience. This requirement implies some form of network infrastructure to support several users interacting at the same time, sharing resources and visualizations. Thanks to the information provided on the descriptions of scenarios 1 through 3 in the previous section, it is possible to summarize the main extend of the networking development into the following objectives:

* Provide a lobby-like architecture for connectivity, allowing users to connect in and out of a work session in a seamless way.
* Share and synchronize information about the digital profiles of the team, the room and the resources being used.
* Share a synchronous interaction with the digital objects used and the augmentation created for the work session.

The main proposal of the research is not the software solution developed, rather the interaction design engineered to fulfil the networking requirements and the process followed to integrate the solution in the flow of a work session, considering some important requirements imposed by the context of the case study:

* Offering a seamless connection with minimal or high-level configuration.
* Disrupting as little as possible the work of the students, considering they are all still working in a graded project that is part of their professional education.

The second development goal is to provide a platform-independent solution. One of the main ethos guiding this development is a bring-your-own-device (BYOD) approach to the implementation of technology in the classroom (Disterer & Kleiner, 2013). This considers that, in theory, the main platform for deployment of AR solutions are mobile technologies, an ecosystem that is ideal for BYOD initiatives (Imazeki, 2014, Väljataga et al., 2016) due to accessibility and ubiquity. Mobile devices also offer hardware solutions to several of the networking objectives proposed previously.

Additionally, a solution that is platform-independent can also aim to accommodate AR devices that are not based specifically on smartphones, such as specialized headsets like the Meta Quest, Magic Leap or the HoloLens. This would create a challenge in design to adapt the control and interactions of the experience specifications that each participant has access in a given session. Figure 3 shows a high-level architecture of a possible solution for this scenario, which is similar to other systems that offer cross-platform functionalities, commonly across a web deployment (Medvidovic, 2002).

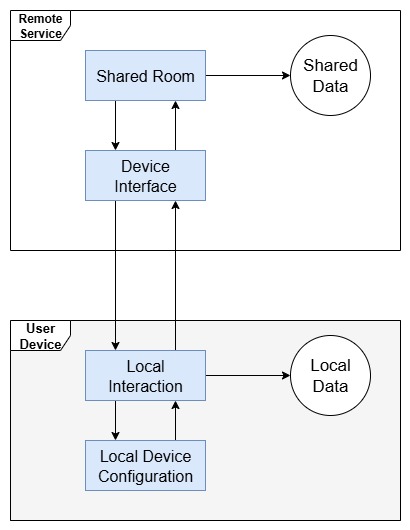


Figure 3 Concept of a Multi-Platform Architecture

Similar to the way in which systems like Netflix balance access to content depending on the device used for access (System Design Netflix | A Complete Architecture, 2023), a general architectural proposal for CollabAR would consider a Device Interface between each local participant and the shared interaction in the room. The Device Interface can manage input interactions, data types and visualization types before handling control to the shared room, and vice versa, handle responses and transform them to data that can be properly handled by every device connected to the session, akin to a system like Andrés & Pérez, 2017.

With that analysis in mind, chapter 5 will describe in more detail the reasoning behind not pursuing a full cross-platform architecture. In summary, the research value did not align properly with the value gained by this type of development. The BYOD approach was instead tested by focusing on mobile devices which, paired with other technologies selected for the development, already covered any mobile architecture using android or iOS, offering a wide spectrum of devices available to the users.

The one element left behind by not pursuing a full cross-platform architecture was testing the integration of different means of interaction in the same work session. It was decided that, for the scope of the research, it was worth to focus first on other aspects of the development and probably relegate this aspect for future work. Nonetheless, the question of cross-platform will resurge in the analysis of the case study and during the user-centred design described in section chapter 5.

In conclusion, the development of CollabAR proposes architectural and implementation solutions for a multiuser AR application deployed over cross-platform mobile technologies. These objectives provide opportunities to understand the technological challenges of implementing multiuser AR applications and how they affect and are affected by the learning goals proposed previously.

# 4.2 Use Cases

The scenarios described in the previous section represent the general goals for which CollabAR will be used as an intervention in a learning setting. To start the software design process, the first step is to identify and describe the specific functional requirements to be developed. The following section will describe the most relevant requirements identified for the research intervention of ENBU 79X/89X. This information is going to be described in the form of use cases, or user stories, to emphasise three aspects:

* The general interactions performed by the users, not the particularities that have to be developed.
* The development goal being fulfilled by the use case, or at least any technological challenge that could be of interest for the analysis process.
* The learning goal being fulfilled and how it is related to the educational objectives of the research.

Before exposing the use cases, it is necessary to discuss some of the vocabulary that is going to be use on the descriptions. These are terms coined to common elements of the interaction with the app, the activities enabled by it and the resources used to create the experience.

A **user** is any student willing to use the app. Every user has a personal profile that is shared between sessions. Although in the context of ENBU 79X/89X all students belong to a particular group for the development of the course’s project, the app is not going to enforce that organization, and a user can become a participant in any session available.

A **participant** is a user that has joined a room and now is part of a work session with other users. As a participant, a user has a role and a set of assigned tasks, which are only valid inside a particular session.

A **room** is the physical space used by the participants to hold the session. The room doesn’t need to be a closed space. The app creates a workspace for the room and associate the session information to a room anchor, a digital object situated in the space selected.

A **work session** is the work done by the participants that enter a room. The session has a general goal associated, which can be modified and updated.

A **workspace** is a digital object representing a space in which the participants can position an interact with digital objects. In conjunction with the room anchor, it also creates a visual representation of the room in which the work session is taking place.

## 4.2.1 Create a shared room

**Description:** A user must be able to initiate a work session by creating a new room for the group. The room is identified by a unique name and described by a set of goals for the session. The user can configure all these elements before launching the session. Upon opening the room, any other user can search the name of the room and join it, adding themselves as participants to the work session.

A room anchor will be added to the space which will present information about the room and the work session upon interaction.

**Development goal:** The room represents the main point for the multi-user experience. All the participants in the room will share the same visualizations and will be able to interact simultaneously with the same digital objects. Ideally, the connection to the room is seamless, requiring only the initial configuration for set-up and run-up.

**Educational goal:** The room will be linked to the concept of a work session. When a room is active it means that the group is working, and all the participants share a current goal. The room provides the students the opportunity to have a structured work session, and the digital markers for the room provide a tangible and interactable representation of the goal and the room itself.

## 4.2.2 Create a shared workspace

**Description:** Upon creating a room, a user needs to define a space that will be used as the shared workspace for the session. CollabAR will identify suitable horizontal spaces like tables, desks or the floor, and overlay a digital plane over them. The user can select one of the detected spaces and modify it to convenience to suit the preferences of the group or the needs of the current session.

**Development goal:** The workspace will be the base augmentation provided by CollabAR to give a digital representation to the room. The room anchor gives basic information about the session, while the workspace facilitates the interactions with other digital objects, constraining the movement and location of all the playing models and providing a reference point for annotations and other constructs.

**Educational goal:** The workspace gives a location for the shared resources of the group, those elements that are public to everyone and could be used as inputs in each task. The workspace anchors all the digital objects to a common point and the group itself, creating a visual representation of the digital room while keeping the constrains of the physical space used. These constraints are more malleable than in a normal room, and the group can use the digital elements of the app to explore or expand what is provided by the physical space.

## 4.2.3 Create a profile

**Description:** A user can create a digital profile to use in a room. The profile is linked to the instance of CollabAR installed in their device. Outside of a room the profile consists of a preferred name and a colour that represent the user in the room. In a room a user can also be described by a role and a set of current tasks.

**Development goal:** The profile is a digital representation of the users in the room. It also serves as an anchor for interactions with other participants and provides a form of digital avatar that carries the information of the profile.

**Educational goal:** The participant’s anchor gives visual structure and prompts the discussion around different topics like team management, assigned roles and allocation of tasks.

## 4.2.4 Point to something or someone in the room

**Description:** This case is related to provide basic communication tools to the students, specially to convey information about the digital resources available and the people participating in the room. Users should be able to specify the person or object (physical or digital) they are referring to in a conversation or instruction. This clarification in the communication process is going to be referred as “pointing”.

**Development goal:** This interaction is meant mostly for digital objects and has an important role in the multiplayer experience. It is a simple tool to facilitate communication and helps mitigating any ambiguity or confusion. It is also an interesting technical challenge to account for both digital objects and the physical context of the room. CollabAR will not deal with image or object recognition procedures, but even a simple interaction with the environment like pointing at an object could be a valuable source of insight.

**Educational goal:** This is a simple tool to promote and facilitate communication. It is one of many ideas to explore the concept of “augmented collaboration” by providing digital tools to visualize the communication process and analyse its impact in the behaviour of the students regarding learning, group dynamics and team management.

## 4.2.5 Check the status of my teammates

**Description:** A user should be able to access information about the participants in a room. This information is related to their profile and their status, described by their role, assigned tasks and participation state in the activity.

**Development goal:** The technical challenge of this use case is related to the way the information present in the profile can be transmitted and synchronized on the network and the proper interaction methods to access it. These interactions can also be extended to accessing the information and current state of the room.

**Educational goal:** This is an opportunity to understand the role and influence of technology as part of the collaboration process. Ideally, the tool provides quick access to information about the current state of the session and the team members, but it could also become an unnecessary barrier for something that should be verbally communicated. It should be possible to observe how students use this functionality, how it affects different communication processes inside the group and how CollabAR compares to other IT tools that mediate in communication.

## 4.2.6 Share and pin an idea

**Description:** A user should be able to type an idea and pin the text representing it somewhere in the room, making it public and accessible to all the other participants. Similarly, a user should be able to see an interact with all the ideas pined by them and other users.

**Development goal:** This use case also belongs in the “augment collaboration” category and offers the opportunity to record and visualize ideas an information in the same way a whiteboard would do it. This is a way experiment with the idea of offering a relationship between the digital representation of the information and the physical space (Zünd et al., 2015, Wang, 2017). The objective is to prompt discussion, associations and structure to a conversation while sharing and recording ideas between teammates.

**Educational goal:** In the same spirit, it is another tool to visualize and interact with information, making communication visible. The idea is that, by adding a relationship between the space of the conversation and the ideas proposed by the group it will open the possibilities to get deeper, different or more structured types of discussions.

## 4.2.7 Add and interact with a model in the room

Description: The third type of interaction proposed by CollabAR id related to simple digital models either build in external tools and imported to CollabAR for visualization, or built in the tool using simple construction blocks. Since the main objective of ENBU 79X/89X is related to architectural design, this interaction with digital models is meant to facilitated communication and discussions related to this field. This communication can be centred around collaborative creation of a design or discussing ideas, feedback or changes to a previous proposal. This use case considers all the interactions needed to add the models to the app, share them in the room and perform simple interactions, like moving, selecting and transforming models.

Development goal: More than a technical challenge, I want to focus in the interaction design needed to propose a solution for this use case. On one front, there is an interesting opportunity to analyse the design and usability of the interactions needed for this task, considering that the app will use a mobile phone deployment and the conventions expected for both the selected hardware and the AR nature of the tool. Second, this interaction also has to consider the multi-user scenario of the project and the necessity of coordinating interactions of shared objects over the network. Although the technical implementations of these requirements are interesting, I am more interested in the insights that the general design process can generate for the research goal, specially for the behaviours prompted to the users that directly affect the collaborative process.

Educational goal: The interaction between users is again the main focus rather than the activity itself. Section 4.4 and several accounts in Chapter 5 will give more detail of why this was chosen as the amin interaction regarding the learning goals of the course. In brief, since the main objective of the course focus on the design and construction planning, it was considered interesting to provide a flexible enough interaction that promotes discussions around different elements of the design, the space and the structures proposed by the students. The building blocks metaphor was chosen early in the design process to promote a visual and physical anchor for the discussion process and because it promoted the idea of experimentation and playfulness that wanted to be encouraged in the work session of the course.

The explicit goal is to organize and structure their work around digital and analogical tools that boost the idea-discussion-decision loop with AR and communication technologies.

4.2.8 Participate in a debate or a vote

Description: A user must be able to propose or participate in a group vote or debate. Both cases will be defined as instances in the work session in which the input of the participants is needed to take a decision. A vote is a decision to be taken among a possible group of options, while a debate has an open input nature.

Development goal: This is an interaction design that allows the users to visualize in the augmented space all the elements taking place in the decision-making process. I want to explore possible implementation for the tool to manage a debate in the group and to storage, compile and show the results of such a debate.

Educational goal: The idea is to reinforce the most prominent activities in which communication skills are developed thanks to the classroom groupwork. I do not expect the tool to be sue as the sole manager for the communication and planning needs of the group, as other tools probably already fulfil those roles. I want to offer a different approach to the communication interactions that happen when taking decisions, specifically in face-to-face scenarios, where discussions are prone to be more informal, unstructured and with no archive of the activity itself. Once more, the aim is to offer a different type of solution based on visualization and context augmentation.

4.2.9 Pause my participation in a room

Description: The user should be able to signal the group that she is not actively participating in the session and that progress is in pause.

Development goal: The reasoning behind this use case is to create a distinction between the physical presence of the user in the room and their active participation in the group, giving more depth to the status of a user, helping in visualise and symbolize some social cues that are hard to verbalize. In a more technical sense, it is also a function that helps in pausing the use of the device without finishing the session nor a future interaction with the hardware.

Educational goal: This option is the best example of tools for the group management that has more significance than simple actions like checking a list of the participants or managing access to the room. This functionality aims at giving visual presence to the work of the user, to stablish boundaries, proper communication habits and prompting discussions around time and space management.

4.2.10 Conclusions

The exposed use cases represent the most important functionalities that CollabAR should be able to fulfil to provide insights for the research goals of the project/ The scenarios present the most common instances in which the students can develop collaborative skills on in which their collaborative work is the key factor in their development f the course project. These actions include team management, task planning, effective communication and decision making.

My proposal is to first identify actionable units in each of those interactions. Things like ideas, options, debates, tasks, status and responses. Through AR is possible to create a visual representation of those ideas and design simple interactions that create a relationship between the ideas and the collaborative skills to be promoted.

The plan also relies on prompting or offering the proper environment to think or discuss issues around the structure of the group’s work and planning, rather than force a specific way of working. This approach can create the opportunity for students to reformulate their current way of working and find ways to incorporate new information into their processes, but it can also cause the students to complete ignore the tool.

This is an acceptable risk scenario. In first instance, it offers a better teaching approach, giving the students a technology, they can freely use to enhance their work, and not a forced experience they could resent or even inhibit their work. It was decided that a free exploration of the tool offered a better and healthier experience.

In terms of research, a scenario in which a group did not interacted with the tool also offers interesting insights. Paired with the case study, a failure engagement with CollabAR can elicit questions regarding the tools that the group is using for communication and planning, how the group is structuring their work and managing their time, among other possibilities. Answers to these questions can help in constructing an interesting case study.

4.3 Architectural Design

CollabAR is an AR software for mobile platforms that provides support to the collaborative work of students in the Industry Project course at the Built Environment programme at the Auckland University of Technology. During the course the students work in a design project of a medium scale construction development and integrate various discipline in the construction field such as architectural engineering, construction management and quantity surveillance. CollabAR provides tools for team management, communication and decision making in face-to-face collaboration using digital visualization to augment the workspace via digital models, info annotation and simple emotes that use the contextual environment of the group to create and augment a collaborative work session.

The following section will discuss the main ideas guiding the architectural design of CollabAR and the reasoning behind the decision taken. The whole process will be described using the most important quality requirements chosen for the development of the tool, the most important functional components identified, and the available frameworks and technologies selected to help in the construction of the tool. Also for this purpose several architectural and structural diagrams were created to explain and visualize the ideas related to design decision and software architecture discussed throughout this section. A common visual language was used that loosely follows the structure and rules proposed by UML, but it was not strictly followed nor any of the diagrams in this section are UML-compliant. The main objective was to offer a quick visual representation of software components and behaviour, while keeping the diagrams simple and easy to read. The main rule was to keep consistency between diagrams, especially with colour coded information as follows:

Gray elements represent hardware and devices, physical instances where software is deployed.

Yellow elements represent software components, entities or a set of entities that have a common purpose and functionality which could be “zoomed-in” for more detail, but that is not necessary to understand the idea conveyed by the diagram overall.

Blue elements represent more abstract entities in the software, either because they represent big complex functionalities which serve several purposes but are a representative element in a diagram or process, or because they represent a state or logical concept in the execution in the software that is actually achieved by the interaction of several other components, but such detail is not important or necessary for the purposes of the information being showed in the current diagram.

Green element are specific instances of an object (a class or a game object in unity) used to show the composition of an important scene or the flow of data in a use case.

White elements are used for data or information being used, created or transmitted between other software components.

Arrows are used to signify relationships and associations between components, or messages sent from one component to another, depending on the context of the diagram.

4.3.1 Quality Requirements

The guiding goal in the development of CollabAR is to create a prototype the helps in promoting proper work habits in the students for communication and collaboration. Independent of the specific functionalities to be built, the software needs to achieve these objectives through the architectural design proposed:

1. Multi-user interaction: CollabAR is a cooperative experience for the synchronous interaction of small groups of 2 to 6 students. All the participants in the interaction will be present on the same physical space and will share the same digital augmentation of the environment, but with different points of view and instances of interaction. Digital objects in the space will also be shared and interactions with them must be synchronized and coordinated between users. The participants will have a digital presence in the shared environment that can also be interacted with, and that interaction also needs to be synchronized and coordinated.

2. Multi-platform deployment: Of the many ideas supporting the educational background of CollabAR is the BYOD approach. To support this idea, it is necessary to approach the solution with a multi-platform deployment of the tool. As the minimum as most essential approach to this requirement, CollabAR must work independent of the mobile platform and aim for the most common AR capabilities of devices currently in the market. Ideally, the tool should also support any other AR capable device, specifically, AR headsets available in the market like HoloLens, Magic Leap or Meta Quest 3. This implies that the tool must consider all the diverse inputs methods available, display varieties and processing capabilities, and coordinate them within the shared experience.

This idea can be also expanded to a multi-role interaction, in which the activities taking place in a room could consider the different roles that any participant can take, and tailor how the user sees information and participates in the experience in accordance to that role, also opening the possibility to interact within the room through mediums that are not necessarily AR. This idea appeared much later in the life cycle of the implementation and considered outside the scope of development. Nonetheless, it was an idea interesting enough to tentatively explore in the architectural analysis and propose conceptual-level solutions for its implementation.

3. Spatial interactivity: Due to the multi-platform approach taken and the irregular nature of the AR capabilities found in current available devices, it was necessary to take decisions about the minimal capabilities needed for the AR experience to function across different devices. It was decided that the core experience was in the augmentation of the communication process and the relationship build between the group and the workspace. It was necessary to identify minimum features common in most devices and decide which aspects of the experience could rely on the available technology and tool, and which needed a solution through a more complex development.

The previous architectural goals establish CollabAR as a multi-user, multi-platform AR system aimed at mobile devices, but with the possibility to expand to other AR-capable platforms and integrate or complement different interactions, roles and activities.

Spatial recognition is important to support the intended interactions of CollabAR, and the functionality planned focused on offering option to “augment” aspects of the communication process between the students, their workflow and the physical spaces being used.

Some other quality attributes have been identified which are not directly related to the outcomes needed for the research goals but are equally important to support a good development flow or to accommodate the particularities found in the execution of the case study.

4. Flexibility: Given the interactive nature of the development proposed and the exploratory approach to the needs and particularities, the general component architecture of the tool needs to be as flexible and decoupled as possible. Particularly, it is important for the input system and interaction methods and processes to be easily swappable without affecting the core functionality of the software. These elements are the most fluid elements of the design, and from the very beginning they are meant to be part of the main experimentation with the software. A flexible approach is necessary to facilitate testing and different ideas and approaches.

Solutions related to the network communication and the creatin of a shared space also needs to be flexible. Ideally the main utility of the tool is independent of the networking architecture. In practice, the tools used to build the networking aspects dictated a lot of the design decisions, especially for simplicity of the implementation. It will be possible to show and ideal composition of the tool at the architectural level while the actual implementation section will discuss the simplifications made to speed up the production of the software.

5. Extendibility: Independent of the differences between the design and the implementation, component development will aim for cohesion and extendibility. The purpose is to support small and incremental development that is easy to extend into full functionality over time. This will also support the user-centred approach proposed, where the final users can experiment with a simpler version of a functionality, provide feedback and incorporate that information in the extension and refinement of the functionality.

6. Easy-to-learn: One of the core issues identified in previous literature related to the negative outcomes of an AR implementation in the classroom was the amount of downtime needed for students to learn to properly use the tool and to solve issues related to the technology, mishandling of the tool or simple management of the activity that had to be done in parallel to the lesson.

Solving this issue related to cognitive overload is a complicated task, but focusing on a design that leverages common knowledge possessed by the students related to their phone and the common interactions with them can help in identifying solutions that emerge from natural use of the technology. This can be achieved by exposing the users to small samples of the software and incorporate a systematic analysis of their feedback in the design process. This is also an objective that belongs in the implementation phase rather than in the architectural design.

It was important to follow a formal approach to the analysis and development of the easy-to-learn design, taking decisions based on principles found in the literature and approaches proven in similar successful scenarios. The most relevant guiding principles are based on the work of Wigdor & Wixon (2011), who state that a design based on natural interaction is characterized by:

The feel of natural extension of the body to expert users.

Offering the same natural experience for novice and for expert users.

An experience tailored for the medium (mobile devices), not a mimicry of the real world.

The proper use of context, previous knowledge, metaphors and input/output devices.

Using paradigms when needed but not being afraid to ignore them.

Several other ideas were summarized from works like Macaranas et al., (2015), Strijbos et al., (2004) and Good et al., (1984) stated some general good approaches like:

Test and use different metaphoric mappings depending on the content of the interaction.

High usability is achieved through careful analysis of feedback

Manage and consider user’s expectations and knowledge

How the use context informs and affects the expectations of the participants and the tools available for interaction design

The importance of integrating the learning outcomes of an educational design in the interaction design process

4.3.2 Technological Frameworks

For the implementation of CollabAR a set of tools were selected to implement current available technologies and frameworks to simplify the constructions of the basic functionalities of the application, support the solutions for more complex problems and provide a pipeline for testing and deployment. In more specific terms, the tools selected help in the implementation of basic AR functions and work as stepping stones for more complicated interactions, the networking development and the multi-platform deployment.

The Unity engine (*Unity Real-Time Development Platform | 3D, 2D, VR & AR Engine*, n.d.) was selected as the main building platform. Similar to other tools and environments like Unreal and Godot, the initial use case of Unity is in the development of videogames, but the tool can also be adapted to be the engine behind any type of multimedia interaction software (Hussain et al., 2020, CodeMonkey, 2024). Unity was selected due to convenience and to leverage my own expertise with the tool.

The main functionalities of Unity were used to handle the implementation of digital content and the interactions with it. Unity also helped in the management of diverse input methods and the multi-platform deployment of the software. Some of the issues related to networking were also solved using the Netcode (*Networking & Netcode Software Solution*, n.d.) utility of Unity. This package allows for basic network configuration and provides a framework to communicate information or coordinate states of the objects created in Unity.

Finally, one of the biggest advantages of using Unity (and other similar tools) is that it provides an implementation of the OpenXR framework (*OpenXR - High-Performance Access to AR and VR —Collectively Known as XR— Platforms and Devices*, 2016). This framework is and open-source implementation that abstracts into a common language all the specifics implementations of different VR, AR and MR devices in the market. OpenXR provides a platform agnostic implementation of different features like tracking management, display, environment tracking, input handling, among other things. This is a useful tool that helps in maintaining a single code base for the application and easily deploy in different platforms at the same time, so that the project can focus on solving specific problems without having to deal with problems at a hardware level.

Additional to the tools provided by the Unity engine, two other frameworks were integrated to the development to help with functionalities related to AR and the shared interactions in the network.

4.3.2.1 ARFoundation

ARFoundation is an implementation by Unity that provides a similar level of abstraction as OpenXR but specifically tailored to the AR environment and the mobile devices ecosystem. This is an important tool that helps in dealing with the most important frameworks in the market for AR development: AR Core and AR kit for Android and iOS devices. These to frameworks cover most of the mobile devices available in the market, while more specialized technology uses APIs or frameworks that are either derivation of the two main ones or can be tackled with the OpenXR framework. This is a powerful tool to solve the multi-platform requirements of the project.

ARFoundation strongly influences the architecture of CollabAR. ARFoundation provides a set of manager utilities that act as a front-end configuration for several common functionalities of AR such as image processing, feature detection, point-of-view tracking or image composition. The managers communicate with a series of subsystems which objective is to abstract specific hardware implementations of the ARCore, AR Kit or other similar toolkits. This means that all the functionalities of CollabAR are built by coordinating the managers provided by ARFoundation with the proprietary code developed for the application. Additionally, any implementation communicates with the hardware found in deployment via the subsystems available, allowing the creation of a single code base for every platform. Figure 4 shows the component architecture of CollabAR when deployed in a generic mobile device, differentiating between the code found in the core Unity implementation vs the local toolkit used to provide AR functionalities in the device.

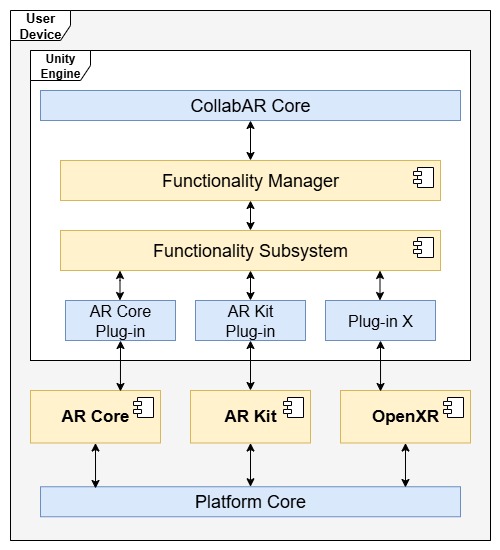


Figure 4 Integration with ARFoundation

With this solution is possible to aim at a deployment that is, at least in theory, independent of any platform. In practice, it is not that simple. It is possible to support any mobile device that implements AR Core or AR Kit, and there is also the possibility to deploy in any other AR-capable device as long there is a subsystem in the ARFoundation toolkit that supports it or if the device supports the OpenXR framework. The problem is that all these implementations do not communicate smoothly, and to support a proper platform-agnostic single code base it is necessary to identify all the intended platforms that are going to be used and coordinate all the low-level communication elements of the code to properly function. Other elements to consider is that if used as-is, ARFoundation only offers functionalities exposed by the managers, which can obscure or not show at all more updated versions of the lower-level systems or not show them at all.

4.3.2.1 Lightship

This is a component part of the Niantic Spatial SDK suit (Niantic Spatial, n.d.) and the set of solutions present in the Shared AR product (Shared AR, n.d.). Niantic has pioneered the commercial use of AR in mobile environment through the successful implementation in entertainment, educational and industrial contexts (Lee, 2022, BowTiedRobin, 2021). The SDK offered through Niantic Spatial is a set of solutions compatible with most development environments to create and manage complex AR interactions like life meshing, semantic detection, complex image recognition and networking.

For the development of CollabAR, the component integrated to the architecture is Lightship, a tool for the creation of shared AR experiences without the need of a complex network infrastructure that has to be built and maintained. This tool is meant for small scale connectivity, supporting up to 10 concurrent users, and scale ideal for the context and scope of the research and the prototype. Lightship eases the development of the multi-user requirements and offers a scalable and flexible architecture that can be expanded and build upon in future work.

In isolation, Lightship works by using a simple client-server architecture. Any instance of Lightship can be connected to another one through an internet connection if they are using the same application ID. The architecture allows for the implementation of a dedicated server that controls the experience or for a distributed connection with one of the client instances acting as a host. At the network level it is possible to use any tool at the developer’s disposal, abstracting as much as possible the connection with the transport layer and seemly integrating with netcode, for example, which acts as the particular implementation of Unity for network communications.

At the application layer, Lightship provides three important concepts to manage the shared interactions:

Room: Parallel to the concept of room provided in the use case definitions, Lightship defines a room as a digital abstraction for the main connection point for the users that are going to share an interaction. A room is identified by a unique ID within the rooms created for a unique application ID. Any client can see and connect to all the rooms available for that application. The rooms are created and managed by the SDK itself, they are hosted in an infrastructure provided by Niantic and can only be configured via the functionalities exposed in the SDK.

Shared Origin: To create the functionality of a shared AR space, a host client first needs to create a Room by providing an ID, configuration data and a point in space relative to the client that is going to be use as the origin point of the experience. This point can be obtained in different ways, like image recognition, spatial features recognition, the relative position of the device or geospatial localization. Upon connection with other clients, each instance of the shared space is coordinated through the network using the shared origin. If a new object is added to the interaction, its position is always relative to the shared origin and coordinated through the network.

Anchors: As optional additions to the shared origin, the host or the clients can create (if the deployment platform has the capabilities) detected features in the real space as anchor points. An anchor can be used to provide visual reference points in the space to maintain a smooth AR experience, especially when a lot of movement and points of view changes are expected, or when the network connection is not stable. An anchor is also shared through the network and helps in stabilizing the room and the different points of view.

It is easy to integrate the main architectural points of Lightship into the architecture of CollabAR, mirroring the use of rooms to signify the connection of users to the group, the detection and analysis of features in the physical space and the coordination of data between interactions. The architecture proposed by Lightship also provides a simple and transparent networking component that allows a seamless connectivity on the side of the user with an easy process and minimal configuration, while also allowing the developer some control over the process. Figure 5 shows the architecture of the shared space as integrated with the core elements of CollabAR.

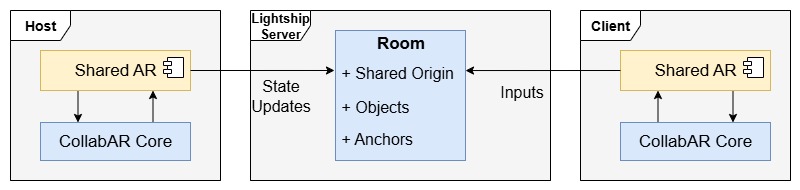


Figure 5 Integration with Lightship

In terms of the architectural considerations impose by the Unity engine, it is important to consider some design constrains and technological implementations that are imposed on every software build using this environment. There is no necessity to go in deep detail of the general architecture of Unity, it is an extensive topic outside the scope of this document that offer little value to the research. It is enough to say that Unity encourages a modular component approach based around a core engine in charge of the coordination of resources and the communication with the local device. More information can be found in Messaoudi et al., (2015).

As for ARFoundation and Lightship, Figure 6 shows a relational model between their most important components, the core development of CollabAR and the external elements that interact in the execution of the experience.

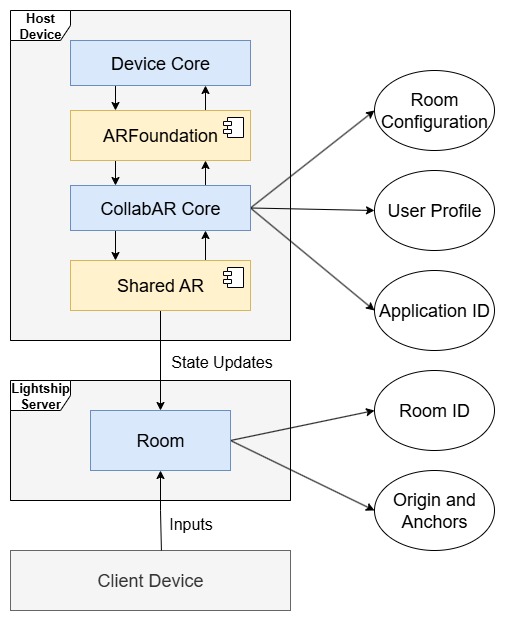


Figure 6 Relational Model with External Frameworks

4.3.3 Component Architecture

CollabAR uses two major components in its architecture: the Session Manager, in charge of the flow of activities in a single session for one participant, and the Tracking Manager, in charge of managing the AR interactions. Additional components are added to manage profile data and UI interactions. CollabAR relies on the unity engine framework for deployment in mobile devices, on the ARFoundation package to communicate with the local implementation of AR and uses Lightship in coordination with some proprietary netcode development to create the shared room and coordinate interactions.

CollabAR follows the recommended architecture for general Unity projects, in which the experience follows the execution of one or more scenes which in turn are composed by a set of game objects. Each object has a specific behaviour determined by its interaction with the engine, other objects and the actions scripted by its modular components.

Following this architecture, CollabAR creates a work session for the users and manages the tracking of all the AR elements through three main objects in a single scene:

The user anchor acts as the visual representation of each user in the room and the main control point of each individual actions through the Session Manager component.

The AR Session is part of the ARFoundation implementation and provides all the AR functionalities to integrate the scene with the camera image captured in the deployment device as well as the detection and management of detected features.

The Network Manager helps with the implementation of Shared AR and manages all the netcode used to create the shared experience.

Other objects are part to the scene to facilitate the visualization of UI elements and manage local data. Figure 7 shows a relational model of the previously described elements in the scene.

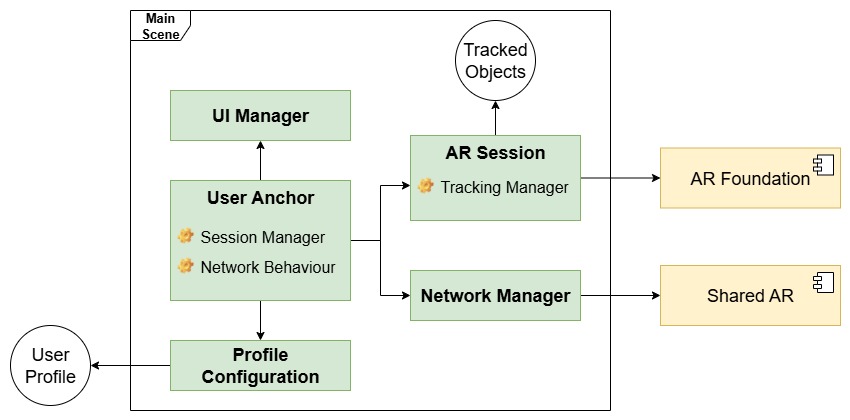


Figure 7 Unity Scene Relational Model

Session Manager

The architecture of CollabAR divides its functionality in two fronts, the collaborative session and the AR tracking. The session manager is in charge of coordinating three elements of the experience: access to the local profile of the user, the configuration of the shared room and the coordination of interactions with the room and with other participants through the network. Other less prominent components are used to provide networking functionalities not managed by Shared AR as well as the interactions with the UI. The component functionality can be overall described by the following process:

Get the local user profile

Configure the connection to a room

Coordinate network behaviours to fulfil the different interactions

Coordinate interactions calls through the network

Coordinate local interactions with the necessary data to show in the UI

The Session Manager is also associated with the user anchor. The anchor is the visual and logical representation of the user in the room, and provides different behaviours to show visual representation of different elements of the user profile and social interaction cues such as emotes and the user’s gaze direction, all meant to be visual elements for the augmented collaboration design. The user anchor is described in more detail in section 4.4. For the purpose of the architectural description, the session manager coordinates the creation of user anchors in the network, bridges local data with different elements on the network and controls the flow of the session. Figure 8 shows a relational model of the most relevant components involved with the Session Manager.

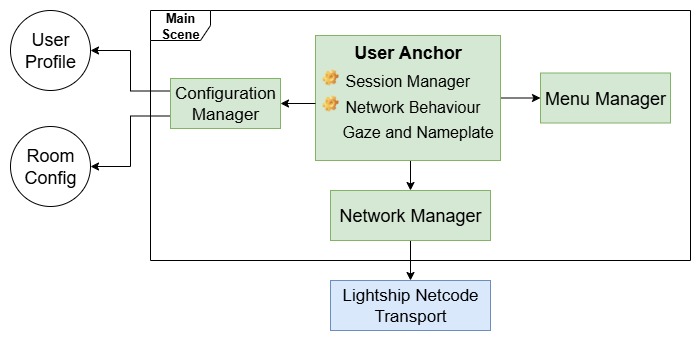


Figure 8 Session Manager Relational Model

4.3.3.2 Tracking Manager

The Tracking Manager is a component part of the AR Session game object in charge of controlling the visualization of digital objects, their behaviour and life cycle. For purposes of the current prototype of CollabAR, the available digital objects are play models and annotations. Each object has a unique behaviour that will interact with the Session Manager or the Tracking Manager as needed.

The tracking manager is the bridge between CollabAR and the ARFoundation framework. It connects all the components needed to create an AR scene and configure the components that provide AR functionalities like image composition and pane detection. The manager is also associated with the main camera of the scene and to the XR Origin object, a component needed in the OpenXR framework to connect and abstract any type of XR device in which the app is going to be deployed, and which AR Foundation uses to communicate with the functionalities of a mobile device. In summary, the tracking manager is in charge of:

Initializing the AR device

Manage the behaviour of the main camera

Manage the digital objects on the scene

Configure the AR functionalities provided by ARFoundation

From the ARFoundation framework, the Tracking Manager specifically pulls the functionality related to plane detection and the creation of visual anchors. The manager configures these components and keeps track of the game objects used for the visual composition of the scene with the information of the camera. A similar coordination is done with the Session Manager, which keeps track of the life-cycle of the objects in the network and the data that needs to be coordinated between participants. Figure 9 shows a relational model of all the components related to the Tracking Manager.

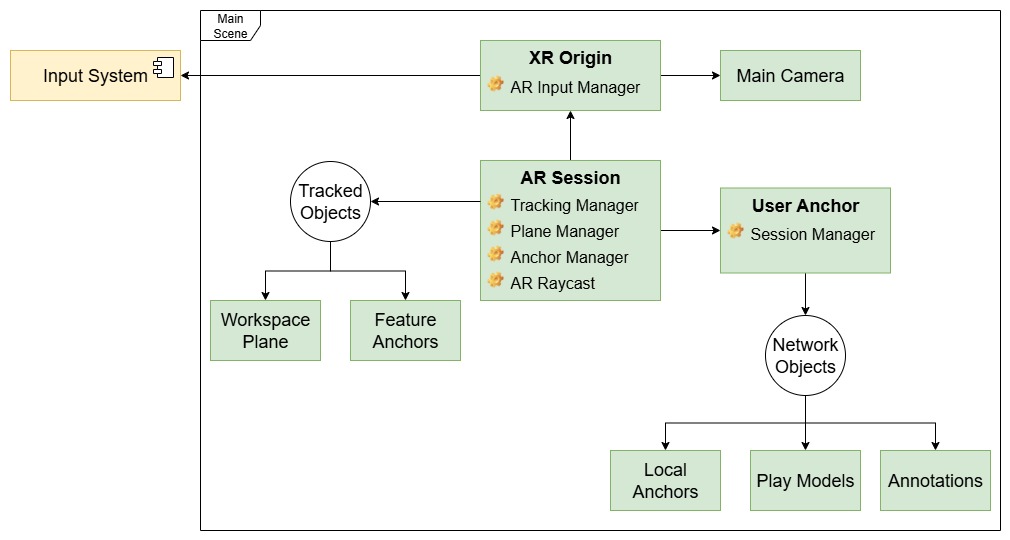


Figure 9 Tracking Manager Relational Model

4.3.3.3 Interaction Pipeline

For purpose of interaction with the host device and between users, CollabAR follows both the structures enforced by Unity and the OpenXR framework, plus the architecture proposed in the system itself to coordinate the AR visualization with the network behaviour. Figure 10 shows the interaction flow between components in case of a local interaction with a digital object, like a play model or an annotation.

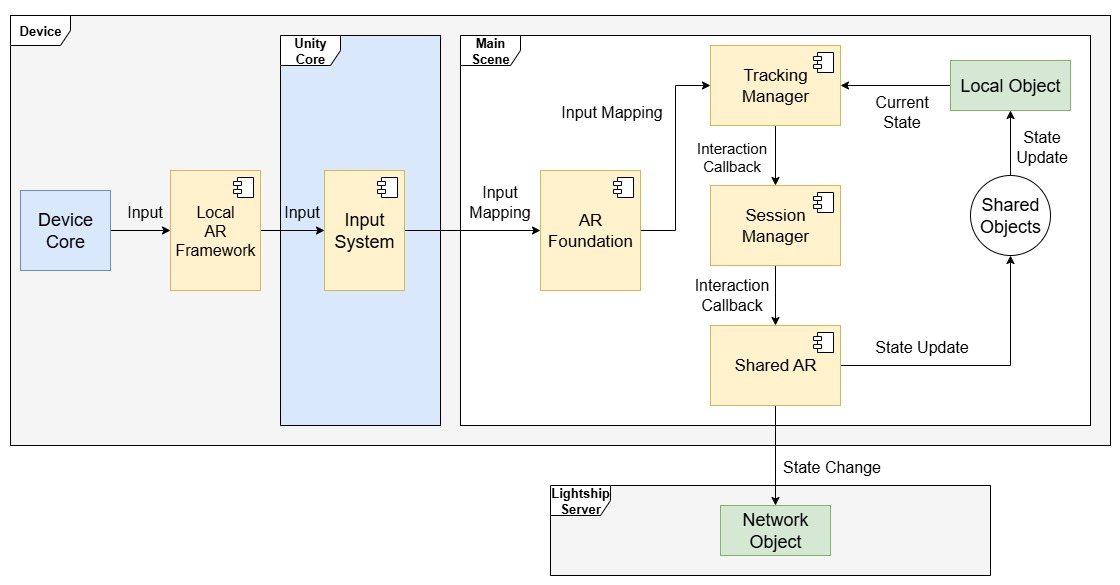


Figure 10 Device Interaction Flow

The local device is in charge of detecting interactions at a hardware level, most commonly the input at a touch screen, the relative position of the device (accelerometer, gyroscope, etc.) and the image feed of the camera. All the input information is then related to the context of the AR application thanks to the local framework present in the device (AR Core or AR Kit i.e.) which transports the raw date of the input the Unity core engine.

In Unity, the AR Foundation framework uses the Input System provided by the engine to read the raw information of the device and maps it to designated functions implemented by the Tracking Manager, which coordinates all the components, either from AR Foundation or from CollabAR, need to respond to the input. Independent of the components involved in the response, first it is necessary to determine if the interaction is local or if it changes the state of a shared object.

Any change to a shared object is handled by the Session Manager, who coordinate the remote network procedures that propagate the relevant information to all the users, and reacts locally when a procedure call is received. The Session Manager is in charge of updating the state of all the shared objects, while the Tracking manager only read the information for visualization purposes and to manage interactions.

Shared AR supports both server-authoritative and client-authoritative network architectures, as well as the possibility to have dedicated servers or host/client dynamics. For CollabAR, a combination of configurations was selected depending on the qualitative requirements being prioritized and to ease the development of the prototype. As it stands right now, CollabAR follows a host/client architecture with no dedicated server controlling the experience. The users gathering for a work session must decide who is going to act as the host. The selected user needs to configure the room and open it, while the other users connect as clients. If the initial host leaves the room, it is easy to pass the host authority to any other connected user.

In terms of authority architecture, the host holds server authority with everything related to the configuration of the room, state of the room, the detected physical plane used as the shared workspace, and the physical features detected as visual anchors to stabilise the room. Meanwhile, the user anchors, the play models and the annotations are client-authoritative, with ownership given to the user that initially spawned the object in the network.

A final consideration is that, despite being designed as a serverless architecture, CollabAR needs to follow the architecture enforced by Shared AR, which works as a central server that hosts the shared room and manage the connections of the client to it. This element is outside the architecture of CollabAR, and for the end user, they just need to worry about the connection with the host, and do not have to think about any external server. Figure 11 shows a diagram that represents the main components in the network architecture, the objects over which each component holds authority and the most important procedures being transmitted over the network.

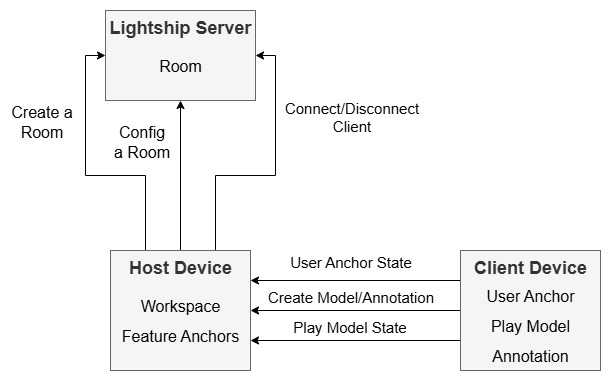


Figure 11 Network Authority Architecture

In case interaction that do not involve digital objects are involve, the pipeline changes a little but maintains the same architecture. CollabAR relies on the services provided by ARFoundation for the detection of features using the camera present in the mobile device, as it is shown in Figure 4. The specific subsystems used by the prototype are the plane detector, the anchor manager and the AR raycast functionalities.

The plane detector is used to identify flat surfaces in the physical space, one of the steps in the configuration of a room. Similarly, the position of a digital object can be used to create an anchor in the digital space, which helps improve the quality and stability of the AR visualization. A new anchor is created in the room every time a new user connects, and these anchors are managed by the host’s session manager. The AR raycaster is needed to interact with digital objects in the AR space or with features in the AR space created by other subsystems. This is a local operation, so it does not need to be propagated in the network. Figure 12 shows the interactions of the previously described components in the network architecture of CollabAR.

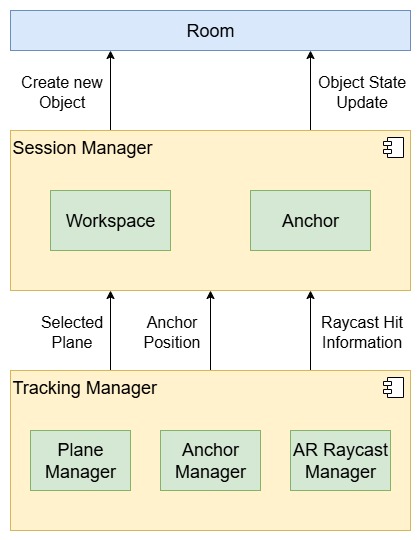


Figure 12 Interaction Pipeline with ARFoundation Subsystems

4.3.4 Data Architecture

For the implementation of CollabAR’s prototype, each use session is meant to be a single contained experience with little to no persistence between activities. In terms of the intended use f the application, it makes sense to have continuity of the work done, track progress and log the activities done for future consultation. This poses a non-trivial design that needs to consider the intended use of the software and how it lines-up with the learning goals of the classroom.

In the simplest scenario, it would be ideal to keep the state of the room in every work session performed by the group. In this scenario, the session would have to persist the following elements:

The users participating in the session

Roles and tasks assigned to each participant

Goals for the session

Digital objects (models and annotations) present in the room and their current state

In technical terms the only challenges lay in managing the ownership of digital objects, and to keep the layout of the room even if the original owner is not present in a new session. This can be solved by stablishing a protocol to deal with “orphaned” objects while keeping them in the room.

There are important architectural considerations in this scenario. First is the responsibility of storage the necessary information to rebuild the room in the future. The previous section explained the virtual serverless architecture of CollabAR, where the central server provided by the Lightship service is in charge of hosting the shared room, but it is not in the capability of managing all the process necessary for running CollabAR, opting better for a distributed host/client approach to manage the networking process of the application. To keep this approach, it is necessary to decide where the data would be storage and who has the responsibility to reconstruct the room in each session.

If the data is responsibility of the selected host device, the reconstruction process is simple, but the role of the host users acquires some “permanency” and there is the possibility of a scenario in which the group cannot continue their work because the assigned host is not present. This puts too much responsibility in a single user.

If each client is responsible of his own data the architecture is then respected and the host does not have unnecessary responsibility, but the room reconstruction becomes more complex and has the risk of not being complete if a participant is missing.

The proper decision would consider the needs of the educational design. In the case of this research project, it is important to give the students the ability to have a flexible group, either because groups are fluid during the classroom’s activities, or because different work session of the same group do not necessarily happen with the same participants. The key is to provide the users with the ability to quickly jump into the tool and start working in the session goals, with the app itself and its configuration being as transparent as possible.

In this case, providing the ability to boot up the room with minimal configuration and previous planning or coordination is the best approach, even if it risks losing information. Figure 13 shows a data flow architecture for this scenario.

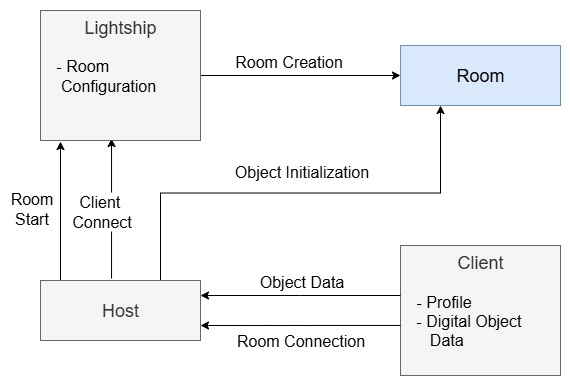


Figure 13 Data Persistence and Ownership

An alternative solution is to drop the serverless approach and add a central server which main objective would be to save and coordinate the data needed to reconstruct a room. Since the Lightship server is outside the architecture of CollabAR, it only provides services to control the creation of a room and the connection of clients to it, and a second server would be needed to provide both the persistence of data and the networking services to coordinate the shared interactions.

Section 4.5 will provide more details about the design changes need to accommodate the role of the educator in the experience and why in that scenario a solution with a centralized server would be a better option. For now, a centralized server could take the responsibility of storing the state of each room and would coordinate its reconstruction in further sessions without involving the host or running the risk of losing information. Figure 14 shows the same data flow architecture if a central server is included.

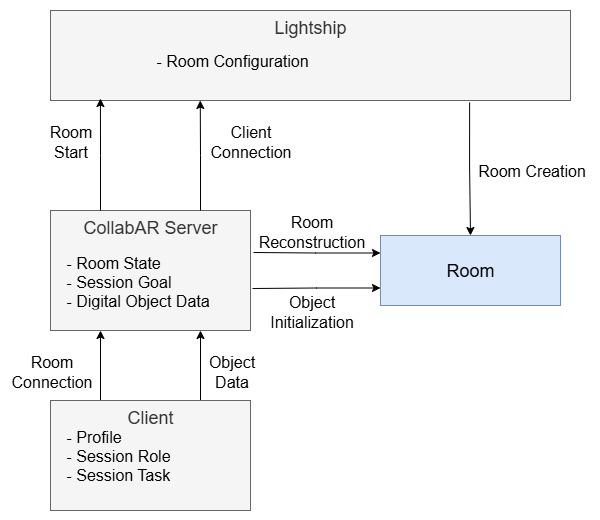


Figure 14 Data Persistence and Ownership with a Centralized Server

For the final implementation no actual persistence of each individual session was implemented, but the general architecture stablished leaned into the solution proposed in Figure 13. The only persistent information of the session was the configuration of the room, which will be explored in the next section.

4.3.4.1 Room Configuration

To simplify the development process and to accommodate the group dynamics observed in the execution of the Industry Project course, it was decided that the information of each individual work session would not be persisted over time. In this way the room creation process would be quicker and the students would not need to organize and sort out information and activities that happened several sessions in the past when the last observation activity took place, a situation that became common due to the data gathering process undertaken.

In this simplified scenario, the room is described by its ID, by the image used to determine the shared origin point and by the max number of users allowed to connect. This information is required by the Shared AR framework and is managed by the Lightship server, CollabAR just needs to manage the configuration process and signal the creation of the room, the start of a session, its end, and the eventual deletion of a room.

On the subject of the configuration process, the user designated as host also needs to provide a list of session goals to share with the group. This information is not mandatory, it can be provided and changed during the execution of the session itself or completely ignored by the group. This information is stored and managed by the host, and in the end, it would be the current host decision to use information locally stored to reconstruct a session or start anew. This acts as a middle ground solution between forcing the host to always fill that role in the group and forcing to star the session form zero every time. Figure 15 shows a simple data flow of this scenario.

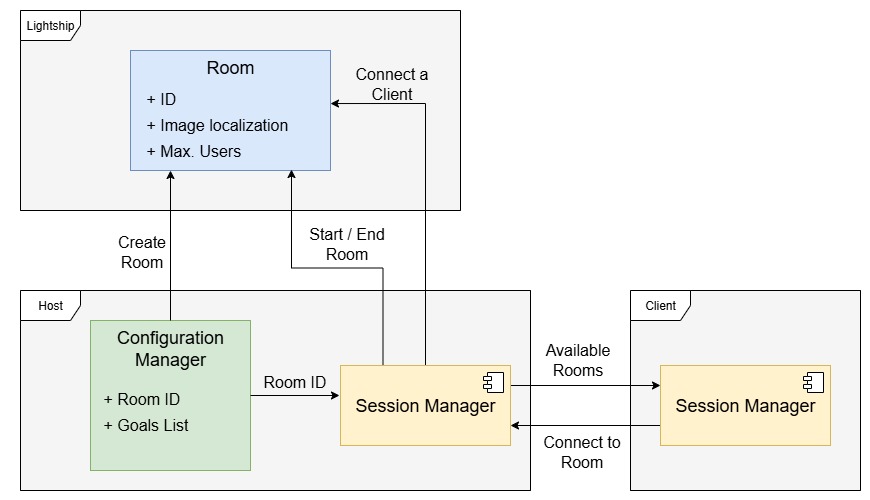


Figure 15 Room Configuration Data Flow

The image colocalization process is a system used by Shared AR based on the image recognition subsystem of ARFoundation. The process requires a base image to be recognized by each client connecting to the room which will be used as the physical anchor for the shared origin of the room. This means that no positioning information is being stored, and instead the localization of the origin point is calculated every time a room is booted. The only information stored is the target image in a raw binary data format. This encourages physical and synchronous gatherings, but if information about the room ID and the target image is shared, remote collaboration is also possible, at least in terms of the visualization of the digital objects being added to the room, but without much of the local context of the experience. That particular scenario was not tested during the data gathering process, as it did not provided value for the research goals.

A client device would only need the room ID to initiate the connection and access to the target mage to initiate the visualization. It is possible to type the ID of the room if the information is externally shared by the users, or it is also possible to ask the system to provide a list of all the available rooms, those associated to the particular application ID of CollabAR. This configuration allows easy hop in and out of session between different groups, a behaviour identified in the observation stage of the Industry Project course. More information about the room creation process can be found in section 4.4, while chapter 5 will discuss in more detail the reasoning behind the group hopping behaviour.

4.3.4.2 User Profile

The second element that is persistent between sessions is the individual profile of each user. For purposes of the prototype, each user is identified by a nickname and a preferred colour. Both elements are used to create a visual representation of the user in the room that is shared and coordinated in the network. This information only acts as a aesthetical choice, and does not uniquely identify the user in the session. It is also the only data locally stored in the user’s device.

When connected to a room each user is identified by a unique ID linked to the current session. The ID is assigned and managed by the host on connection. Once in session, a user can define a current task, also linked to a single session. This is not a mandatory process and is the sole responsibility of each user to manage this information after being discussed by the group. With this setting, CollabAR emphasizes the need to discuss and consider issues related to task allocation, but does not force a particular implementation, and puts the responsibility of discussing the matter to the physical communication of the group.

Finally, each user anchor offers two visual cues to all the users in the room, a nameplate with the current nickname of the user and a set of visual markets representing the gaze of the user, that is, the direction the camera on the user’s device is pointing at. These elements represent information of the user in and out of the room, the first one about the participant and the second one about the current state and actions in the room. Both of them are also optional, and each user can op to turn off or on the visualization of these elements for each of their teammates if needed.

Figure 16 shows a simple data structure of the user profile, while a more detailed discussion about the interaction design behind these elements and their purpose in the educational design.

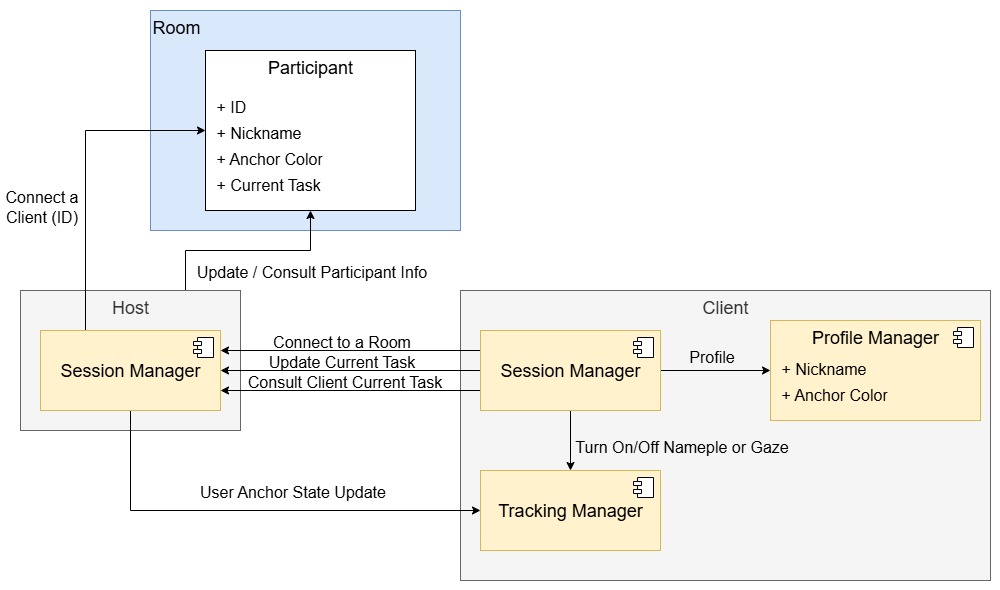


Figure 16 User Profile Data Flow

4.3.5 Use Case Diagrams

The following sub-section will detail the intended flow of the most prominent use case scenarios for CollabAR, giving emphasis to the components involved and the data transmitted and coordinated through the network.

The main objective is to understand the macro functions of CollabAR in terms of the components needed to start the development process and quickly identify the most crucial interactions for the interest of quick testing with users and most valuable for the research goals. This analysis process was also a useful tool for the interaction design process and to initial generation of ideas about how students could be using the software for collaboration and the behaviours that could be promoted by the experience.

4.3.5.1 Room creation and room join

The most basic interaction between users is to create a room and join it to start a session. As for every meant to be mediated by CollabAR, the idea is to distribute responsibilities between what is provided by the tool and what the group has to physically coordinate and discuss. Taking this philosophy in consideration, the process of creating a room and joining it to start the session can be described in the following steps:

1. Decide the overall objective of the work session.

2. Determine who is going to act as a host and any other roles needed or relevant for the session.

At this point CollabAR is not involved yet, but it architecture enforces the user to discuss and plan the session in terms of objectives and roles. The first software related steps involve the configuration of the room, meant to be as detached from the host as possible in order to not force unnecessary responsibilities on the role:

3. The host creates a room by providing an ID, the tracking image and optionally a session goal.

4. The host communicates the room ID to the rest of the group.

5. Each client fiends the room using the correct ID and joins it, becoming a participant in the session.

6. Each client (including the host) scans the tracking image to stablish the room shared origin, initiating the visualization of the room.

7. Each participant specifies a role and a current task if needed or desired.

Each user is responsible of providing the information needed for the session, depending on how much the students have structured their work around CollabAR. This information can be changed and updated at will. It is also possible for each user to have access to as much information as desire from their teammates, opting for turning on and off visual elements loke nameplates and gaze representation, and accessing information about goals and tasks only upon direct interaction.

8. If the host leaves, the host responsibility is transferred to any other connected client.

9. The session finishes and the room stops if all clients have disconnected.

CollabAR provides the opportunity to visualize elements about the organization and planning of the work session but is the responsibility of the users to properly discuss and determine that information and its evolution across the session. Figure 17 shows a use case diagram detailing the most important components involved in the process and the data flowing between them.

Diagrama

El contenido generado por IA puede ser incorrecto.

Figure 17 Use Case Diagram - Create and Join a Room

Thanks to the diagram it is possible to realize that the bulk of the communication and coordination of information is done through the host and client devices, specifically through the coordination of remote process in the Session Manager component, Communication with the Lightship server itself is only done the host device, and is mostly use too initialize the connection of new clients and the initial configuration of the shared origin and all the objects already in the session.

4.3.5.2 Add and interact with a model

To give support to the activities in the core of the course educational goals it was decided that CollabAR was best suited to give support in the initial steps of the architectural, engineering and management process by communicating ideas among the different roles taken by the students in a group. The interaction with digital play models is a simple implementation that facilitates ideation and exploration activities by providing simple and “tangible” visualizations of the discussion about space, the relationship between different physical entities, positioning of elements and sketching ideas. These interactions are meant to be used when:

Architectural and construction engineers want to discuss initial ideas of a design in a high level of detail, using the play models as stand-ins for spaces and constructions.

Construction managers or quantity surveyors need to understand any aspect of the proposed design and need a simple-to-use visual tool to communicate a concept, question or idea.

The group wants to have a shared visualization or discussion of a custom model that has been imported to CollabAR.

The most straightforward version of this scenario involves a participant adding a set of play models to the room in a particular position and configuration to convey the desired idea, which can then be interacted with by the other participants using direct transformation, pointers or annotations.

The participant selects the desired play model form the Play Models menu.

The participant uses the camera pointer to position the model in the shared workspace.

The participant can tweak the position, orientation and scale of the model. The model is now shared by the group, and its state is being coordinated over the network.

Once the model is shared over the network the matter of ownership and shared interaction becomes important. Determining the ownership of a model is a technical issue and can be boiled down to decide if the host holds the ownership and the authority of the state of every model in the room or if the responsibility lays in every client and the host only communicates the changes in state. Considered the size of the groups expected for the data gathering process, the difference between both approaches is negligible.

For the implementation it was decided that the client should have the ownership over every object spawned and that its state should be client-authoritative, a configuration that would ease the processing load on the host device, but that would require the construction of a protocol to manage the ownership of an object in different scenarios, as discussed previously.

The most important architectural decision related to this scenario is related to the coordination of interactions between participant and a single model. The interaction needs to consider who is transforming a play model in any given time to avoid conflict of the state of the object over the network. This requires a proper protocol to “request” the permission of transforming a model and “blocking” it while a user is actively manipulating it (citation of other examples). This protocol is implemented via software, needs a visual signal in the UI and can be enforced by a parallel protocol in the physical space implemented by the users themselves. Figure 18 shows the use case diagram of this scenario.

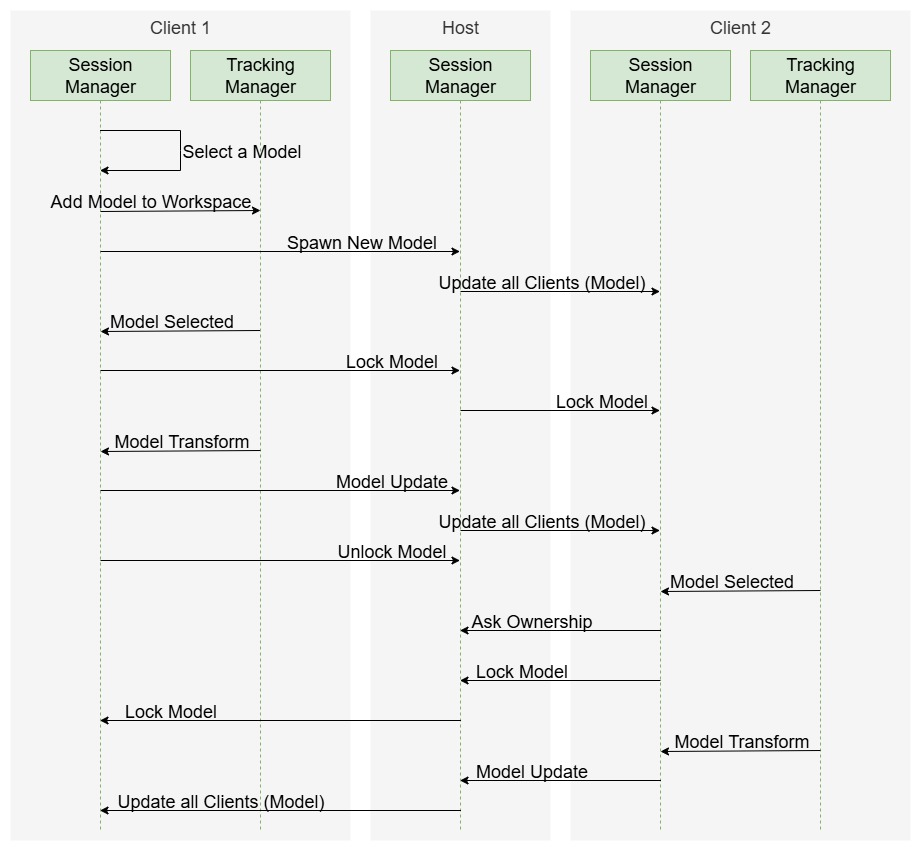


Figure 18 Use Case Diagram - Add and Interact with a Play Model

4.3.5.3 Add an annotation

The concept of an annotation is used to communicate ideas between the participants of a session and have them “attached” to the room, acting as a whiteboard of concepts or linking the position of the annotation with some element of the physical space to convey meaning. More details of this design can be found in section 4.4.

The general process in this scenario is the same as with the play models, the main difference in the degree of interaction any user can have with the annotations. As with the play models, any participant can create an annotation using mainly text, although the systema can easily be expanded to also use images. The annotation is placed in the room using the Raytracing Manager and the best approximation of the position detected in relation to the shared origin of the room. This process is important because the position of an annotation cannot be relative to the workspace like with the play models, but relative to the space or real object detected in the room. This final position is approximate calculated by the Raytracing Manager and the relative position of the shared origin in each device.

To create a consistent view of the room to all participants the shared room has to be consistent in terms of position and orientation. Shared AR offers the option to use image recognition or geolocalization to provide the shared origin initialization. Geolocalization is precise to a degree but cumbersome, and requires a lot of previous set-up. It would also limit the execution of a group session to the same physical space every time. CollabAR uses image recognition to simplify the colocalization process of the shared origin by using a simple image marker that needs to be scanned. This provides flexibility at the cost of a less precise visualization. Section 4.4 will detail the interaction design implications of this implementation and the different ideas tested to facilitate a smooth experience for the users. Figure 19 shows the use case diagram for the creation of and interaction with annotations

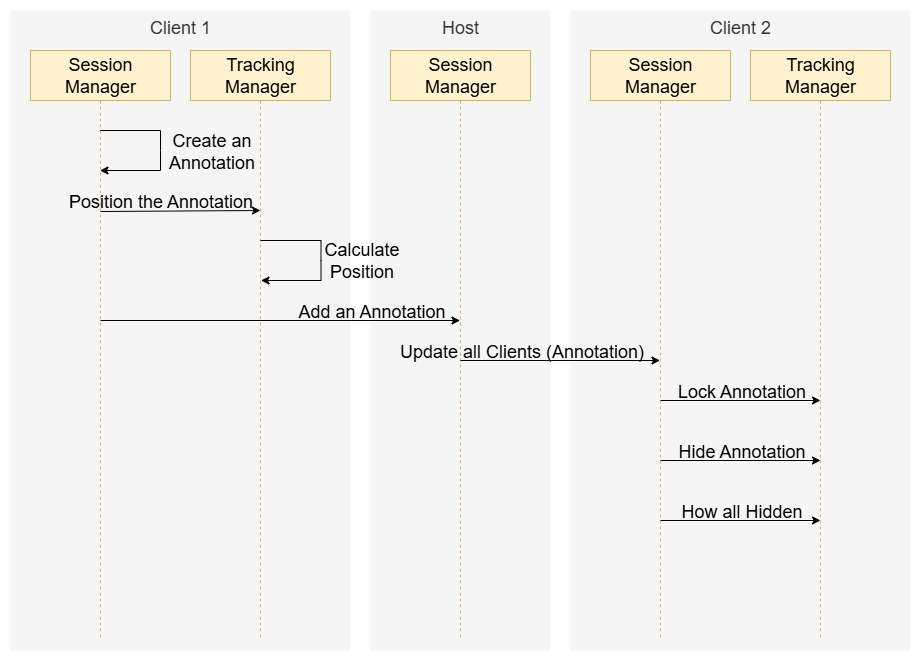


Figure 19 Add and Interact with an Annotation

It is important to notice that, unlike the play models, each local tracking manager needs to calculate the position of the annotation before transmitting it to the host, which creates discrepancies when updating the information in each client. Additionally, all interactions with the annotation are local because they only concern each user visualization of them, unlike the play models, that need to be updated in each client.

4.3.5.4 Propose and participate in a debate

The final big interaction with CollabAR s to support a debate or vote in the group, meant to mediate the discussion and to visualize the data being used in the conversation. A good portion of section 4.4 discusses the extend in which CollabAR intervenes in the process and the reasoning behind the decisions taken. For the purposes of describing the architecture it is enough to comment that the role of the software itself is minimal, letting the participant handle the flow of the process themselves. The application limits itself to show visual indication of the objective of the discussion, the current state of the process and the responses proposed by the participants. CollabAR uses simple and quick to read emotes for this purpose and proposes this three-step process for the whole activity:

Prompt the debate: One of the participants creates the debate by providing a question and possible options. All participants receive the debate prompt and have access to the questions and the options.

Propose an answer: Each participant can select an option and the selection is shown to all other participants.

Discuss: When every participant has answered the prompt there is an opportunity to discuss the answers and change emotes if so desired. This process is coordinated by the participants and not the app.

At this point the participants can opt to continue with the debate or conversation outside the control of CollabAR, maybe referencing the information displayed by the app. On the other hand, the group can structure the activity around the software and keep editing the debate or creating new ones to keep the conversation flowing. Either approach is valid and an interesting analysis point in the future.

The decision to create the emote system, or at least a simple prototype of it, derives from the feedback gathered from the users that proposed it as a form to have a more interesting and active way of handling the responses of the debate, as well as a tool outside this one scenario. In the end the whole debate process is simple and hints the users at a structure they can use but without forcing them into it, which is the main philosophy for the entire design of CollabAR. Figure 20 shows the use case diagram for this scenario.

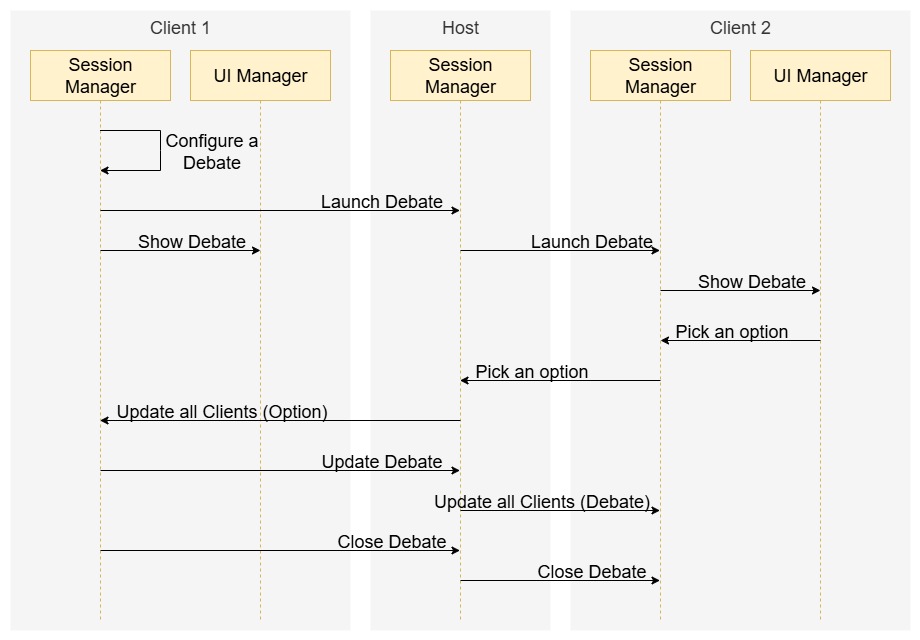


Figure 20 Propose and participate in a debate

4.4 Final Thoughts

The exercise in architectural design described through this chapter had the main objective of stablishing the guiding principles behind the development of CollabAR and the way they aligned with the goals stablished for the research and the educational design, as well as informing the decision taken to facilitated development and reach a satisfactory set of conclusions for the research.

From the architectural design process, it was possible to identify key elements of analysis related to the research questions, specifically SQ5 and SQ6, those related to identify the elements in the technological intervention that influence the most the learning design and the results obtained. First, the quality goals proposed highlighted the most flexibility and adaptability, which in turn guided the decision-making process into a constant analysis of the effects on the long run, and how could it be possible to facilitated experimentation and constant change. The criteria for picking technologies or possible solution also weighted heavily the foreseeable effects such implementations would have on the learning design, how well they aligned with the learning goals or with the intended behaviours the tool was meant to promote.

One of the most relevant examples presented itself during the analysis of the debate/voting use case. Initially, the proposed approach considered information found in the literature about collaborative work by stablishing a process that the users could lean into if the communication in the group was poor or lacking purpose (citation?). The process that the users had to follow was initially in the likeness of:

Launch the debate activity

Wait for the participants to confirm

Show AR representation of the debate.

Prompt for introductory question

Wait for votes

Show AR representation of the results

Prompt for discussion

Close debate activity

This process was also modified for different types of debates, like those based around yes/no questions, multiple options or open discussions. At one point it was also important to distinguish between a vote using the UI or using models in the AR space. It is possible to see remnants of this approach in the Task design of the gestures analysis described in section 5.1, as well in the general structure of the process shown in this chapter, but overall, the approach to implementation changed substantially.

The main source of change came from the data obtained during the architectural analysis paired with the previously mentioned research. The result was a simpler process more focused in flexibility and options rather than in structure. In that way, the final users could adapt the tool into the more unstructured flow or the classroom and to the other way around, trying to give structure to a more flexible approach to work.

Several other instances of redesign and adjustments to the original ideas behind CollabAR were a direct result of investing time into the architectural design, and of testing and revising such design. Early interaction with users and revisiting the architectural plan during the iterative implementation of the software were key activities to obtain these results.

Some other aspects of the functionality of CollabAR that were refined by comparing the architectural proposal with the needs of the learning design were:

The responsibilities of the student taking the role of host, which ended acting as the leader of the session or the group. This clashed with the more open structure of the course and the way the students were required to work.

The structure imposed on the ownership of objects, which, in contrast to the previous point, more structure was imposed into the activity to facilitate development and give more clarity to the interaction using the application.

The lack of persistence between sessions, that, in a different tangent form the previous points, prompted the students to improvise on tools and process to keep track of the information between sessions.

Overall, the architectural design process was flexible and was evaluated in each iteration. Akin to the role that software architecture takes in more agile-focused developments, the architecture of CollabAR focused more on creating a eagle-vie plan for production, aligning that plan with the objectives of the development and documenting the process with just the necessary elements to facilitate communication and evaluation (Babar, 2009). An effort was made to create enough documentation to reflect the development process of CollabAR without overfocussing in formality or in an exhaustive coverage of all the details of the final product.

Hopefully, this chapter provided enough insight in the development process followed for CollabAR, on the reasoning followed for the decisions taken in matters of technology and the plan proposed to tackle the implementation stage. The architectural design also reflects the changes and adjustments that took place thanks to the data obtained from testing and the initial usage of the end users and other interested stakeholders. Finally, it is possible to draw some preliminary conclusions about the technical facet of the overall research, with the objective to build a set of answers to the research questions:

1. As with any exercise within the TEL context, the learning design process, and more specifically, the execution itself, will be heavily influenced by elements of the technical development. It is very difficult to create generic solutions in learning scenarios (citation), and both the learning context and the technology used must be aligned and properly developed. For this project, in which AR was the core technology involved, it was possible to identify a big effort being made in the current market to create a common language for design, interaction and technology, all in hopes of facilitating and encouraging development.

This also implies that it is important to follow a rigorous design process that considers what the market is converging to and the expectations that the users are starting to develop. AR technology is entering little by little into commercial maturity, creating a set of design tropes and common language that must be considered, studied, and obviously, worked upon and challenged.

2. It is possible, at the design stage to identify how the users will possibly approach the usage of a tool. If the design process is paired with early testing and a substantial analysis of the involved users and stakeholders, it is easier to gather data and start drawing conclusions as soon as possible. Specifically in the educational context, literature shows that the mayor pitfall for any TEL development is a misalignment between the learning goals and the actual deployment of the software. It is important to consider how a technology is achieving a particular objective, if at all, and how it is integrated into the bigger context of the classroom, the activities that is supposed to support and all the other goals, behaviours and tangential learning that is promoting. Teaching technology deals with a very complex context that is both easy to misunderstand and underestimate (citation?). As shown in this chapter, it is possible a good deal of these hurdles by following the proper process and creating the right mindset in the development team.

With these partial conclusions in mind, it is possible to continue with the technical implementation of this research in the next chapter, which will focus on the final implementation and testing processes followed in the development of CollabAR.