The European Master in Building Information Modelling is a joint initiative of:

Universidade do Minho

UNIVERZA V LJUBLJANI

Practical Augmented Reality for Modular Construction: A BIM Driven Factory to Site Workflow

Eddy Mendoza Jiménez

BIM A+

UMinho | 2025

Universidade do Minho
Escola de Engenharia

Eddy Mendoza Jiménez

Practical Augmented Reality for Modular Construction: A BIM Driven Factory to Site Workflow

BIM A+ European Master in Building Information Modelling

SEPTEMBER 2025

Universidade do Minho
Escola de Engenharia

Eddy Mendoza Jiménez

Practical Augmented Reality for Modular Construction: A BIM Driven Factory to Site Workflow

BIM A+ European Master in Building Information Modelling

Master Dissertation
European Master in Building Information Modelling

Work conducted under supervision of:
José Luís Duarte Granja

September, 2025

AUTHORSHIP RIGHTS AND CONDITIONS OF USE OF THE WORK BY THIRD PARTIES

ACKNOWLEDGEMENTS

This work is the result of months of research, countless iterations, and continuous learning none of which would have been possible without the support, collaboration, and presence of key people along the way.

To my supervisor, thank you for your guidance throughout the process and for allowing the space to explore the direction this thesis eventually took.

To the faculty and administrative staff of the BIM A+ program, thank you for creating an environment that encouraged intellectual curiosity, technical depth, and interdisciplinary thinking.

To Donald for your contribution and support during the creation of the UI framework.

To the classmates, peers, and friends who participated in usability testing and shared feedback during virtual prototyping your observations helped sharpen the outcomes more than you probably realized.

To my family and closest friends your support made all the difference. For the late nights, long talks, and continued encouragement, thank you.

To Andrea and Gabriel, who were my pillars throughout this journey. To H.W., for always being a quiet but steady voice of support and a constant source of inspiration.

To Katherine, Martha Elena, and Xinia, thank you for being there in past years, during harder times and helping me move forward when it wasn't easy.

And finally, to the open source community, whose tools, shared knowledge, and code formed part of the foundation of this research you made building things not only possible, but meaningful.

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

Eddy Alberto Mendoza Jiménez

RESUMO

A integração do Building Information Modelling (BIM) e da Realidade Aumentada (AR) representa uma oportunidade significativa para otimizar os fluxos de trabalho digitais na indústria da Arquitetura, Engenharia, Construção e Operações (AECO). Esta dissertação apresenta a conceção e a implementação parcial de uma framework BIM AR focada no desenvolvimento de aplicações móveis em tempo real baseadas em AR, recorrendo a motores de jogo, em particular o Unity. O trabalho tem como objetivo definir uma metodologia replicável para a incorporação de dados BIM em ambientes de AR, abordando desafios críticos relacionados com interoperabilidade, estrutura de dados e design de interação do utilizador.

Foram desenvolvidas duas frameworks interligadas uma framework de AR e uma framework de Interface de Utilizador (UI), para orientar a implementação. A framework de AR estabelece uma sequência lógica de decisões relativas à seleção do motor de jogo, conversão de formatos de ficheiro, gestão de metadados e estratégias de posicionamento em AR. Em paralelo, a framework de UI aborda a organização, os papéis, as interações e os componentes necessários para garantir uma experiência eficaz de utilização de BIM em AR. Ambas as frameworks foram testadas e aperfeiçoadas de forma iterativa através de prototipagem virtual, storyboards, testes de usabilidade e avaliações baseadas em cenários.

Uma inovação central desta investigação reside na utilização de normas abertas e de ferramentas de desenvolvimento gratuitas. A conversão de ficheiros de IFC para GLTF foi realizada através de scripting em Python, preservando os metadados e garantindo agrupamentos personalizados alinhados com o esquema IFC. Foi ainda implementada uma lógica em tempo de execução que permite o carregamento externo de modelos através de ligações API, aumentando a adaptabilidade entre diferentes plataformas. O protótipo resultante suporta a manipulação de modelos e o acompanhamento da montagem em ambientes de AR, oferecendo perspetivas sobre como os dados BIM podem ser visualizados e aplicados em soluções de AR orientadas para o contexto de obra.

Esta dissertação contribui com uma abordagem estruturada e adaptável para o desenvolvimento de BIM AR, constituindo uma base sólida para futuras melhorias em termos de usabilidade, funções de reporte e interoperabilidade alargada. A metodologia proposta apoia uma adoção mais ampla de ferramentas de AR pela indústria, mantendo a conformidade com os padrões digitais abertos da construção.

Palavras chave: Arquitetura em Tempo de Execução, Integração BIM, Motores de Jogo, Realidade Aumentada, Interface de Utilizador

ABSTRACT

The integration of Building Information Modelling (BIM) and Augmented Reality (AR) represents a significant opportunity to enhance digital workflows within the Architecture, Engineering, Construction, and Operations (AECO) industry. This dissertation presents the design and partial implementation of a BIM AR framework focused on delivering runtime, mobile based AR applications using game engines, particularly Unity. The work aims to define a replicable methodology for incorporating BIM data into AR environments while addressing critical challenges in interoperability, data structure, and user interaction design.

Two interrelated frameworks an AR framework and a User Interface (UI) framework, were developed to guide the implementation. The AR framework establishes a logic based sequence of decisions concerning game engine selection, file format conversion, metadata handling, and AR placement strategies. Simultaneously, the UI framework addresses the layout, roles, interactions, and interface components necessary for an effective BIM AR user experience. Both frameworks were iteratively tested and improved through virtual prototyping, storyboards, usability tests, and scenario based evaluations.

A key innovation in this research lies in its use of open standards and free development tools. File conversion from IFC to GLTF was achieved using Python scripting to preserve metadata, with custom grouping aligned with the IFC schema. Runtime logic was established to allow external model loading through API connections, enhancing adaptability across platforms. The resulting prototype supports model manipulation and assembly tracking within AR environments, offering insights into how BIM data can be visualized and utilized in field oriented AR solutions.

This thesis contributes a structured and adaptable approach for BIM AR development, providing a strong foundation for future enhancements in usability, reporting functions, and extended interoperability. The methodology supports broader industry adoption of AR tools while maintaining alignment with open digital construction standards.

Keywords: Augmented Reality, BIM integration, Game Engines, Runtime Architecture, User Interface

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS AND ABBREVIATIONS

ACC	Autodesk Construction Cloud
AEC	Architecture, Engineering and Construction
AECO	Architecture, Engineering, Construction and Operations
AI	Artificial Intelligence
API	Application Programming Interface
AR	Augmented Reality
BCF	BIM Collaboration Format
BEP	BIM Execution Plans
BIM	Building Information Modeling
BIM AR	Building Information Modelling Augmented Reality
BIM AR UI	Building Information Modelling Augmented Reality User Interface
BREEAM	Building Research Establishment Environmental Assessment Method
CDE	Common Data Environment
CMS	Common Minimum Standards
EIRs	Exchange Information Requirements
ELBIGMAC	Educational Laboratory Big Machine
FBX	Filmbox
GIS	Geographic Information Systems
GUI	Graphical User Interfaces
HAR	Handheld Augmented Reality
HMD	Head Mounted Displays
IDM	Information Delivery Manuals
IDS	Information Delivery Specifications
IFC	Industry Foundation Classes
IoT	Internet of Things
LGG	Level of Geometric Granularity
LiDAR	Light Detection and Ranging
LOD	Level of Detail/Development
LOIN	Level of Information Need
MEP	Mechanical, Electrical, and Plumbing
MX	Mixed Reality
NUIs	Natural User Interfaces
OP	Operating Systems
SDKs	Software Development Kits
T&C	Terms and Conditions
UI	User Interface
UPM	Unity Package Manager
VFX	Visual Effects
VR	Virtual Reality
XR	Extended Reality

1. INTRODUCTION

The Architecture, Engineering, Construction and Operations (AECO) industries are facing an ongoing digital and paradigm transformation from traditional methodologies to digital, virtual and innovative ones, such as Building Information Modeling (BIM), Augmented Reality (AR), Virtual Reality (VR) and others. Construction projects are evolving to be more complex, precise, time sensitive and innovative, to create safer, sustainable, comfortable, efficient and creative spaces during the life cycle of projects. This transformation is not only technological but also cultural, changing how professionals collaborate, plan, visualize, and execute their tasks.

The implementation of Augmented Reality (AR) in AECO has evolved from isolated pilot projects to structured frameworks capable of supporting real life scenarios. However, despite its proven potential to merge digital content with the physical environment, its adoption in practical workflows, especially in modular construction remains limited. The fragmentation between digital design and physical execution continues to challenge project accuracy, coordination, and efficiency. This thesis addresses it by exploring the integration of BIM and AR, focusing on a realistic and replicable solution for the assembly of prefabricated elements.

This research develops and implements a BIM AR software framework, structured through a user interface and supported by a mobile AR engine, to facilitate the assembly of prefabricated elements in modular construction. The framework enables real time integration of BIM models into an AR interface tailored to the AEC sector, realized through a full stack application built with Unity Engine, Python scripting, and runtime API communication. Designed for Android devices, the application follows four stages, initial decisions, format setting, metadata setting, and AR placement aligned with the requirements of the proposed BIM AR framework.

To reach this objective, the following goals were pursued:

1) Evaluate the current landscape of AR and BIM integration, identifying key limitations, usability barriers, technological flexibility, and potential alignment with real-world AECO workflows.
2) Establish a structured BIM and AR integration framework, prioritizing technical viability, usability, and open-source accessibility to support low-cost implementations in real-world contexts.
3) Define a modular user interface structure that supports multidisciplinary interaction across BIM and AR contexts, ensuring intuitive use, functional clarity, and reduced cognitive overload.
4) Demonstrate the practical viability of the proposed system through a partial implementation that integrates BIM, AR, and a user interface, with its performance evaluated through technical testing.

The existing literature offers rich explorations of BIM and AR as independent technologies, but practical examples of their seamless integration, especially using GLTF based runtime approaches, API driven workflows, and modular UI segmentation remain scarce. Prior studies tend to remain in either the experimental or demonstrative phase, without addressing the operational limitations of current tools, such as file format compatibility, mobile deployment constraints, and interoperability losses during export/import processes. In contrast, this research contributes by not only proposing a solution, but by implementing and testing it under realistic constraints.

The research methodology adopted a qualitative and iterative approach, combining technical development with design oriented thinking. The software was created using Unity engine (with C# scripting), Python (for GLTF conversion and metadata structuring), and Docker containers for API access. The process was enriched by virtual prototyping, wireframes and storyboard creation, and architecture modeling (Architecture α and β), enabling structured implementation and evaluation of the system's performance and usability.

This thesis is structured into the following chapters:

- State of the Art: Reviews key concepts such as BIM, AR, game engines, and their intersection, highlighting previous research, tools, and limitations.
- User Interface (UI) Framework: Proposes the general and AR specific UI architectures, detailing design logic, user roles, usability concerns, and visual interfaces.
- BIM AR Framework: Presents the comparison of game engines, definition of the BIM AR framework, and decisions on file formats, metadata handling, and AR placement methods.
- Frameworks Implementation: Describes the full development and execution of the BIM AR tool, including API communication, scene management, GLTF processing, and the final model manipulation features.

In conclusion, this work seeks to contribute a replicable, technically feasible, and user oriented BIM AR solution bridging theoretical frameworks and hands on implementation. The result is a software tool that brings BIM data to the real world through mobile AR, structured interactions, and accessible user experience.

2. STATE OF THE ART

2.1. Introduction

As established in the introductory chapter, the integration of Augmented Reality (AR) in the construction industry has gained significant momentum in recent years particularly when combined with complementary technologies such as Building Information Modeling (BIM), Light Detection and Ranging (LiDAR), Geographic Information Systems (GIS), and the Internet of Things (IoT). This chapter presents a critical analysis of the current literature, technical reports, and recent studies to explore how these technologies are being implemented, with a focus on BIM driven AR applications in modular construction.

2.2. Theoretical review

This literature review reveals three main research streams relevant to this dissertation. The first focuses on the application of AR in construction and its contributions to industry improvement. The second examines the synergy between emerging technologies particularly BIM and AR within both professional practice and educational curricula. The third investigates how AR, when combined with game development platforms, enhances learning and practical implementation in modular construction environments.

2.2.1. Digital environments in construction definitions and implementations.

The Architecture, Engineering, and Construction (AEC) industry is undergoing a shift from manual processes to digital workflows. These transformations rely on digital environments such as Augmented Reality (AR), Virtual Reality (VR), Mixed Reality (MR), and Extended Reality (XR), which facilitate reality capture the process of documenting and analyzing the physical world through sensor equipped devices (Hamledari and Fischer, 2021).

Among these, AR plays a central role in this research. It is broadly defined as the merging of virtual and real world elements to enhance a user's perception of graphic information (Azuma, 1997) . AR helps to bridge knowledge gaps in the construction workforce, especially among younger generations (Sawhney, Riley and Irizarry, 2020), and it facilitates real time progress tracking and visualization of 3D and 4D information (Musarat *et al.*, 2025). This is achieved by overlaying full scale, digital data into real world construction environments (Abboud, 2014).

The advantages of AR include enhanced understanding of geometric complexity, process visualization, and product coordination. It supports quality assurance by superimposing computer-generated models onto actual construction sites(Bademosi, Blinn and A Issa, 2019). Applications of AR in construction include defect detection, site visualization, and integration with Building Information Modelling (BIM) for planning and monitoring. Additional benefits include worker guidance, productivity enhancement, safety risk identification, and sustainability optimization (Grazina, 2013; Bademosi, Blinn and A Issa, 2019). However, challenges remain, including low resolution, computational demand, and precision limitations (Fernandes *et al.*, 2011).

The literature review has identified several key applications of AR in construction:

1) Process guidelines for workers
2) Productivity increases by automation of the information
3) Detection of potential safety risks
4) Increase sustainability and profitability (Bademosi, Blinn and A Issa, 2019).

In contrast, VR creates fully immersive virtual environments, removing the user from the physical world. It is frequently used to simulate construction scenarios, allowing for hazard identification and preemptive design evaluation (Jenuwa *et al.*, 2025). VR also supports skill transfer and reduces cognitive load during complex tasks, such as component assembly and cost estimation (Dinis *et al.*, 2017; Pereira da Silva, Eloy and Resende, 2022; Rashidi *et al.*, 2022).

Contrary, Mixed Reality (MR) blends the capabilities of AR and VR, allowing virtual elements to coexist and interact with the physical environment (Musarat *et al.*, 2025). Extended Reality (XR) encompasses the entire spectrum of immersive technologies AR, VR, and MR, collectively blurring the boundaries between digital and physical realities (Alizadehsalehi, Hadavi and Huang, 2020).

These technologies have significantly contributed to safety management, training, and real time visualization in the construction sector(Li *et al.*, 2018). For instance, AR and MR assist in minimizing safety risks such as falls and collisions, scanning, progress tracking, decision making and hazard reduction (Jenuwa *et al.*, 2025). The integration of Light Detection and Ranging (LiDAR) improves data precision and geometry collection (Puri and Turkan, 2020), while Internet of things (IoT) enables real time monitoring and automation (Li *et al.*, 2018).

The key distinction between AR and VR lies in their relationship with and representation of the physical world. While VR replaces reality with a fully simulated environment, creating virtual data, AR enhances the real world by superimposing digital content onto it, thereby amplifying existing real world data(Wang, 2009). AR reduces physical and mental errors during assembly by offering real time, context specific visualizations, whereas VR supports improvements in budgeting and scheduling accuracy (Davila Delgado *et al.*, 2020; Hajirasouli and Banihashemi, 2022; Makransky and Klingenberg, 2022)

Despite these advancements, the main barriers to widespread adoption of AR, VR, and MR remain the high cost of implementation. This includes expenses related to hardware, software, training, equipment, and workforce adaptation (Musarat *et al.*, 2025).

Furthermore, both AR and VR facilitate the transfer of expert knowledge to newer users, improving comprehension of construction topics and technological tools (Dinis *et al.*, 2017). For instance, AR optimizes the assembly process by reducing mental load, speeding up component selection, and minimizing errors (Rashidi *et al.*, 2022). Likewise, VR improves time management, cost forecasting, and obstacle detection during project execution (Pereira da Silva, Eloy and Resende, 2022).

Another important contrast is that AR supports real time visualizations during on site work, helping to reduce rework, errors, and completion time. VR, on the other hand, is more effective in off-site environments where simulation, issue detection, and planning are the primary goals (Davila Delgado *et al.*, 2020; Hajirasouli and Banihashemi, 2022; Makransky and Klingenberg, 2022).

This research has identified the main distinctions among XR, VR, AR, and MR (Figure 1):

- AR: enables a juxtaposition of digital and real-world elements.
- MR: creates a dynamic blend of physical and digital content with real-time interaction.
- VR: offers a fully immersive experience, isolating users from the physical environment.
- XR: serves as the overarching term encompassing all of these technologies (Verdelho Trindade *et al.*, 2023).

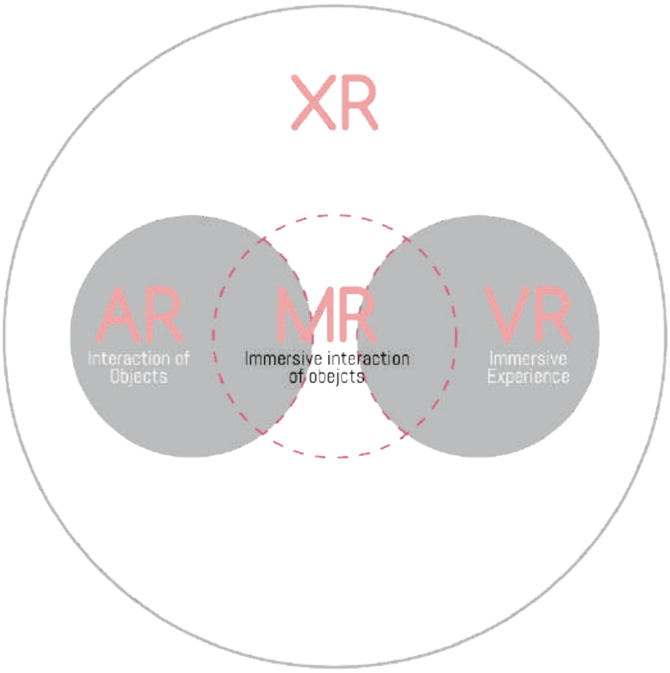

Figure 1 – Extended Reality environments

Indeed, the emergence of AR, VR, MR, and XR reflects ongoing innovation in construction management driven by the need to improve safety, workforce training, and real time project visualization (Li *et al.*, 2018).

Finally, given its proven capabilities in enhancing on site visualization, reducing errors, guiding the workforce, and improving safety and sustainability outcomes, AR should be considered as a practical, high impact tool to advance digital transformation in the AEC sector.

2.2.2. BIM and AR in education and implementations

Once digital environments have been established, it becomes essential to clarify the concept of Building Information Modelling (BIM), as it frequently overlaps with AR in the literature explored. BIM represents a shift in mindset from 2D drawings to 3D, data driven models. It emphasizes site safety and links digital design with simulations of processes, conflict resolution, optimization, and logistical planning of buildings and their assets. Additionally, BIM facilitates graphical engagement in safety management processes by identifying potential hazard zones and material storage areas (Azhar, Hein and Sketo, 2008). Moreover, BIM integrates geometric data with time and cost information, enabling more effective planning and real-time process monitoring (Sabil and Erizal, 2023).

Other studies have defined BIM as a digital methodology for managing and generating building related information throughout a project's lifecycle. The benefits of BIM include improved design accuracy, reduced errors, cost savings, and enhanced time management (Liu and Chen, 2025). Some researchers, however, describe BIM as a tool that offers a static, monocular visualization, meaning that the user receives spatial information through a single visual input. This approach lacks real world positional awareness or kinaesthetic feedback. In contrast, integrating BIM with AR creates a binocular or oculomotor experience, giving users a richer understanding of spatial contexts through interactive, real-time visualizations (Zou, Arruda and Ergan, 2018).

BIM models are commonly categorized into three interaction methods: Graphical User Interfaces (GUIs), voice based operations, and vision/gesture based operations such as VR and AR. Currently, GUIs are the most widespread. These interfaces rely on graphical elements menus, toolbars, and icons interacted with via mouse and keyboard (Liu and Chen, 2025). However, in construction environments where workers may face mobility constraints or operate specialized equipment, voice based interaction can improve accessibility and simplify operations (Liu and Chen, 2025).

Furthermore, combining BIM and AR creates a forward thinking approach to improve project management efficiency. This integration supports real time progress tracking, quality control, decision-making, and schedule management (Surri and Kuncoro, 2025). It also improves the identification, analysis, communication and resolution of process gaps on site by linking 3D model with real time visualization (Machado and Vilela, 2020). Research shows that this integration is particularly effective for project tracking and inspection (Pan and Isnaeni, 2024).

Nevertheless, the human factor presents a significant challenge in the adoption of BIM and AR, particularly in cases where users lack prior education or training expose in these tools (Yang, de Vries and van der Schaft, 2021). Moreover, the absence of standardized guidelines, planning strategies, and interoperability remains a key obstacle to AR implementation in construction (Revolti *et al.*, 2023). Additional limitations include reduced visual accuracy, metadata loss, hardware/software compatibility issues, and human readiness for organizational change (Surri and Kuncoro, 2025).

The literature contains numerous studies exploring the implementation of digital environments particularly AR and BIM in educational settings. For example, at the University of Florida, three random groups of students were assigned different instructional materials to complete construction tasks involving masonry walls, roofing, and steel structures. Conducted in two phases problem solving via questionnaires, and a quantity take off exercise the study aimed to evaluate the effectiveness of AR in teaching construction management (Bademosi, Blinn and A Issa, 2019).

Similarly, Auburn University (Alabama, USA) conducted a simulation involving 36 construction management students who were tasked with identifying plumbing points in a concrete slab using AR technology. The simulation employed Trimble's Connect AR and SiteVision GPS software to assess student adaptability to AR tools (Olsen and Kim, 2025).

Some studies have noted a decline in qualified professionals in the AECO industry due to generational turnover (Rokooei *et al.*, 2023). This trend has contributed to a loss of field expertise and practical knowledge, reinforcing the idea that passive educational methods are insufficient to teach the spatial, temporal, and problem solving skills required in construction (Bademosi, Blinn and A Issa, 2019).

Additionally, most of the literature on BIM and AR emphasizes applications in progress tracking, decision making, and on site visualization. AR has been shown to aid construction projects by capturing site-specific information, planning, execution monitoring, and enhancing collaboration among stakeholders (Chi *et al.*, 2012).

Regarding implementation, AR has proven especially effective in large-scale and complex projects (Haji Rasouli *et al.*, 2022). It reduces misinterpretation of 2D drawings (both physical and digital) by offering hands-on, real-time visualizations of projects (Safira and Hidayat, 2024).

The literature also presents two implementation frameworks. The first outlines various use cases for AR, including:

1) GPS based AR systems offering spatial audio navigation for visually impaired users
2) AR tools for power plant inspections
3) Problem-solving guidance during vehicle assembly at Ford
4) Localization based AR using site photographs to contextualize real time data
5) GPS enabled automated capture of site operation information
6) Integration of BIM and AR for online 3D object mapping (web3D)
7) Mobile AR applications (e.g., InfoSpot) using gyroscopes, accelerometers, and GPS for positioning
8) Real-time construction site simulations supporting 4D and 5D visualization

This framework also identifies critical implementation technologies such as fiducial markers, Geographic Information Systems (GIS), GPS, laser scanning, and photogrammetry. It suggests operational inspections, building maintenance, infrastructure monitoring, and installation assemblies as key AR+BIM application areas (Machado and Vilela, 2020).

The second framework (Musarat *et al.*, 2025), outlines the phases of AR and VR implementation:

1) Understand the organization's needs and identify areas where VR adds value
2) Determine the cost of training, software and hardware
3) Competencies needed for the implementation
4) Integration of the current staff and workflow by creating an integration plan

2.2.3. Implementation of BIM AR on gaming engines

After establishing the definitions and examining existing BIM and AR implementations, it is necessary to explore accessible and effective ways to adopt these methodologies in real world contexts. This is particularly relevant for Augmented Reality (AR), which serves as the central focus of this research. The literature identifies gaming environments as an effective medium for teaching and learning scenario specific tasks or technical content. In this context, AR functions as a tool to simulate the environment and enable virtual visualization of data overlaid onto physical spaces (Ayer, Messner and Anumba, 2016).

In recent years, the use of AR, MR, XR, and VR applications has become more affordable and user friendly across industry and education due to growing public adoption. Much of this popularity can be attributed to the successful integration of AR, VR, and MR in the entertainment sector, which accelerated mainstream familiarity with immersive technologies (Dinis *et al.*, 2017).

For the integration of BIM with AR, MX, XR, or VR, two primary workflows were identified for transferring data from BIM authoring tools into game engines. The first involves exporting BIM files into compatible game engine formats such as OBJ, FBX, or IFC. However, this method may result in the loss of material, textures or metadata, requiring rework within the game engine. The second workflow uses rendering tools before transferring to preserve textures and materials during the export process (Zou, Arruda and Ergan, 2018).

During the integration of BIM AR to game engines, it is essential to consider the quality, quantity, complexity, and granularity of the data involved. This includes the proper application of Level of Detail/Development (LOD), Level of Information Need (LOIN), and Level of Geometric Granularity (LGG), each at their appropriate implementation phase. However, research indicates that for achieving optimal interoperability, minimizing import/export time, and reducing data loss, the most effective approach is the application of LGG. This concept defines the geometric components to be exported from BIM tools, quantified primarily by the polygon count. While LOD includes both geometric and semantic information, LGG focuses specifically on the geometric precision needed for integration (Zou, Arruda and Ergan, 2018).

Additionally, this integration is not linear, as BIM data must be optimized first to reduce polygon and polyline complexity before being imported into a game engine. This optimization aims to decrease import/export times but can introduce interoperability issues specifically the loss of properties, materials, textures, and geometry (Zou, Arruda and Ergan, 2018; Graham *et al.*, 2019).

BIM data regularly includes both geometric and non geometric attributes, such as levels, room names, spaces, and element properties. To preserve this information, Industry Foundation Classes (IFC) file formats are often recommended, as they retain metadata related to cost estimation, scheduling, building storeys, geolocation, classification systems, and more. However, most gaming engines do not natively support IFC. Instead, they accept formats such as FBX (Filmbox, typically exported using Revit's Section Box) or OBJ (Johansson and Roupé, 2024).

Current studies have also examined the use of gaming engines such as Unity and Unreal Engine to merge BIM with AR, VR, and MR technologies. A notable example is the Educational Laboratory Big Machine

(ELBIGMAC) project at the University of Porto in Portugal. This initiative observed the behavior and engagement of pre university students interested in civil engineering (Peixoto and FEUP, 2018).

ELBIGMAC was conceived as a pioneering initiative to transform civil engineering education. The central concept was to treat a house as a "Big Machine," encouraging students to explore its design and systems using a blend of AR, VR, sandboxes, QR codes, and focus groups. BIM, various Levels of Detail (LOD), and gaming engines were integrated to make the technology both accessible and pedagogically effective for students and the general public. As a result, the project introduced a new, collaborative, and open educational model (Dinis *et al.*, 2017).

The project also contributed to defining key terms, such as Natural User Interface (NUI) and the role of Unity 3D. A NUI refers to an interface that allows intuitive interaction between users and digital content. It must be easy to use and designed to accommodate users with little or no prior experience with AR or the AEC (Architecture, Engineering, and Construction) industry (Dinis *et al.*, 2017).

Another critical aspect is the performance and interoperability challenges faced during the conversion of BIM models into game engine compatible formats. A relevant study by Zou (2018) at the University of New York tested the transformation of 36 BIM models into Unity engine. These models varied in discipline (architectural, structural, mechanical, furniture), building size (small, medium, large), and level of detail (coarse, medium, fine). For example, the architectural discipline included three building sizes, each with three detail levels, creating nine unique models. This logic applied across all disciplines to assess the impact of model complexity on AR integration (Zou, Arruda and Ergan, 2018).

Game engines offer immersive, interactive environments that enhance collaboration, coordination, and decision-making. Unity and Unreal Engine have emerged as dominant tools due to their open source development platforms, adaptability, and real time rendering capabilities (Van, Wong and Abbasnejad, 2025).

These engines also support streamlined optimization through features like occlusion culling and mesh/LOD simplification. Occlusion culling refers to the selective rendering of only visible objects within the virtual camera's field of view, improving system performance. Unity 3D enables the creation of virtual environments using toolkits and assets, while supporting input from diverse data sources including BIM software (W.F. Chen and J.Y. Liew, 2002).

Before presenting the possible solutions for game engine integration, (Chapter 3) it is important to highlight the concept of gamification, as it serves as a foundational element for the entire integration process. Gamification refers to the application of game design principles and mechanics into non game contexts such as education, marketing, construction, and business processes to enhance user engagement, motivation, and participation. Within the scope of BIM and AR integration, gamification helps facilitate intuitive navigation, interactive learning, and immersive task performance. It creates engaging user experiences by introducing challenges, feedback mechanisms, and goal oriented structures, which are especially valuable in training, simulation, or instructional applications (Marín-Vega *et al.*, 2023).

2.3. Practical and software review

As part of the review on AR implementations within the AEC industry, a significant gap was identified in practical experience and documentation regarding real world applications. To address this, a preliminary review of eight AR platforms and their integration into AEC workflows was conducted. This section introduces the initial results of that review, while a deeper analysis and discussion of these findings will be presented in Chapter 4. The selected tools were evaluated based on their core functionalities, adaptability, use of AR, and any supporting value found in academic or technical literature.

Firstly, Cupix provides a comprehensive suite of features including 360° video to 3D mapping (Scan to BIM), 4D documentation, spatial collaboration, offsite BIM verification, an all in one spatial hub, smart progress tracking, and enterprise-grade security. Additional features include offline mode, voice control, real-world interaction capabilities, ArcGIS interoperability, AI integration, and progress tracking (Cupix, 2025) (Figure 2).

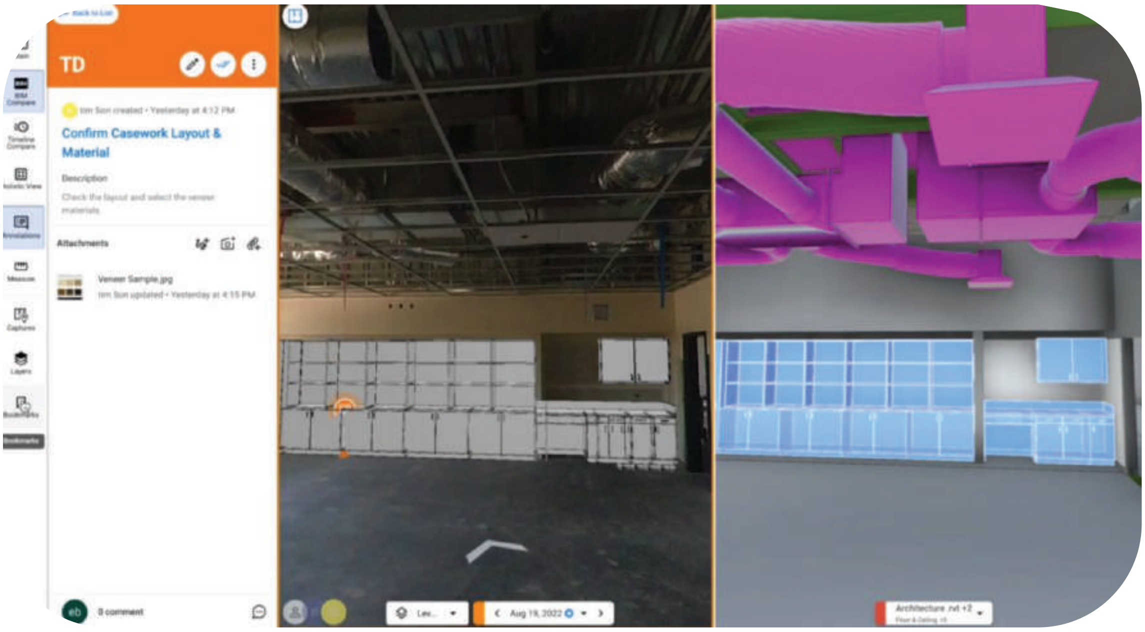

Figure 2 - Cupix Overview, from Cupix website, 2025

Cupix was implemented in a case study at the Tampines Chinese Temple in Singapore, where it was used to create a virtual tour aimed at preserving both tangible and intangible cultural heritage. The platform was selected for its automation capabilities, reduced human labor requirements, geotagging functionality, and interoperability with platforms such as Google Street View, ultimately improving accessibility, usability, and overall value. In this study, Cupix was benchmarked against two other platforms: Walkabout Worlds and Paneek (Mah *et al.*, 2019).

Secondly, Dalux TwinBIM represents the AR module of Dalux software. This tool operates using native IFC models that include room identification tags. However, its core functionality is limited to visualizing rooms, furniture, and installations. It operates on a plane detection system, requiring users to manually define the surface where the 3D model will be displayed necessitating pre alignment for accurate placement (Dalux, 2025) (Figure 3).

Figure 3 - Dalux Overview, from Dalux website, 2025

Dalux was used in a project that integrated Autodesk Forma to assess compliance with the Building Research Establishment Environmental Assessment Method (BREEAM) certification. Dalux served as the Common Data Environment (CDE), facilitating collaborative data exchange such as solar, wind, daylight, and noise analysis, while supporting basic requirements like Level of Detail (LOD) (Chelaru *et al.*, 2025).

CDEs are defined as centralized platforms for managing and integrating project data throughout the project lifecycle (Sacks *et al.*, 2022). A case study involving Swedish construction firms identified Dalux and Trimble Connect as the most widely used CDEs. Interviews with BIM professionals emphasized that the integration of CDEs with structured formats like Information Delivery Manuals (IDM) and Information Delivery Specifications (IDS) led to the most efficient and effective outcomes (Lindholm, Johansson and Yitmen, 2025).

Thirdly, Arki is an app developed by Darfdesign focused on AR visualization for interior design. It leverages LiDAR to create real time floor plans by scanning furniture, windows, and walls (Figure 4). Arki supports interoperability with BIM tools such as SketchUp, Maya, Revit, Blender, Rhinoceros, 3D Studio Max, ArchiCAD, and Cinema 4D, thanks to its compatibility with FBX and GLB file formats. Limitations include a reliance on cloud storage platforms (Google Drive, Dropbox, OneDrive), a file size limit of 100MB, and simplified texture rendering (Darfdesign, 2025).

Figure 4 - Arki Overview, from Darfdesign website, 2025

Fourthly, Gamma AR is deeply integrated with Autodesk Construction Cloud (ACC) through an API called the Gamma BIM Portal (Autodesk, 2025). It supports Revit, IFC, and Navisworks files with capacities up to 45 GB and allows export to formats such as BCF (BIM Collaboration Format), CSV, and PDF. Gamma AR has demonstrated significant performance improvements, with reported reductions of 75% in coordination time, 79% in clash detection effort, and 80% in site inspection tasks (GAMMA AR, 2025) (Figure 5).

Figure 5 - Gamma AR Overview, from Gamma website, 2025

Key benefits include reduced drawing misinterpretation, enhanced real-time communication, time tracking, and support for offline mode, collaborative design, and BCF file integration through BIMcollab (Youngblut Kristina, 2023; MCAD, 2025).

Fifthly, SmartTek is a software development company offering innovative AR and VR tools across various sectors including real estate, healthcare, automotive, and education. Notably, their AResidence platform allows clients to tour real estate projects using AR. Built with Unity Engine and Unity AR Foundation, AResidence offers an intuitive User Interface (UI) with simple interactions such as move, rotate, and scale (SmartTek, 2025) (Figure 6).

Figure 6 - AResidence Overview, from SmartTek website, 2025

Sixthly, Trimble Connect provides functionalities including CDE services, 2D/3D visualization, data management, and file organization. It is widely adopted due to its support for over 45 file formats including IFC, DWG, and various geospatial types (Trimble, 2021, 2025) (Figure 7). While no significant academic studies have been found focusing on Trimble Connect's AR usage, one study compared the Trimble X7 laser scanner with Pro Drone for photogrammetry based site scanning was found (Cunha *et al.*, 2022).

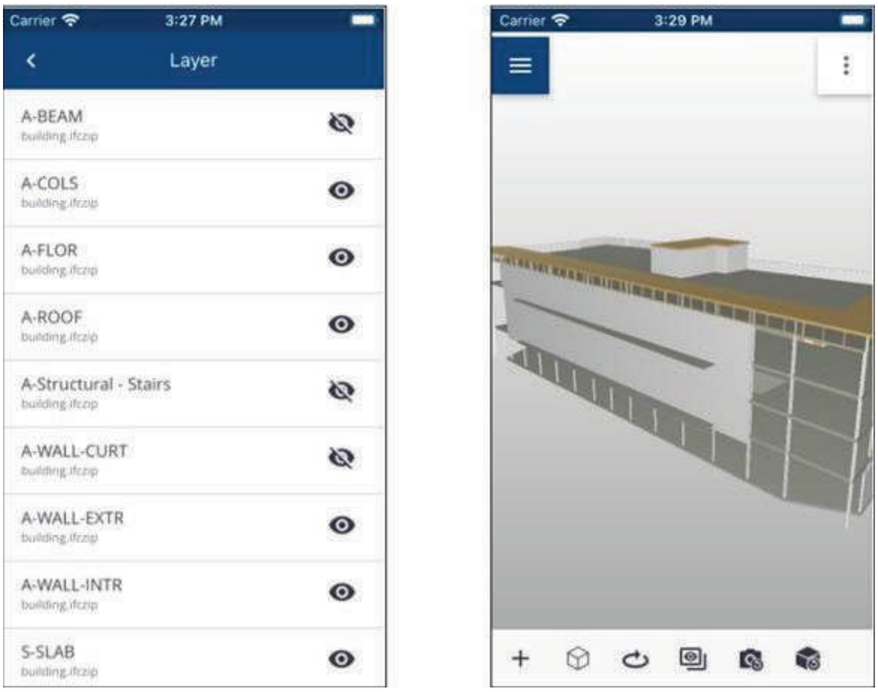

Figure 7 - Trimble Connect Overview, from Trimble website, 2025

Seventhly, VGIS is widely applied in the MEP (Mechanical, Electrical, and Plumbing) domain due to its fast scanning via QR codes or image markers, data collection features, and survey grade accuracy. VGIS integrates point clouds, GIS data, and BIM models to create precise AR overlays. It reports survey cost savings of up to 95% (VSite, 2025). However, its application in core construction workflows remains limited (Figure 8).

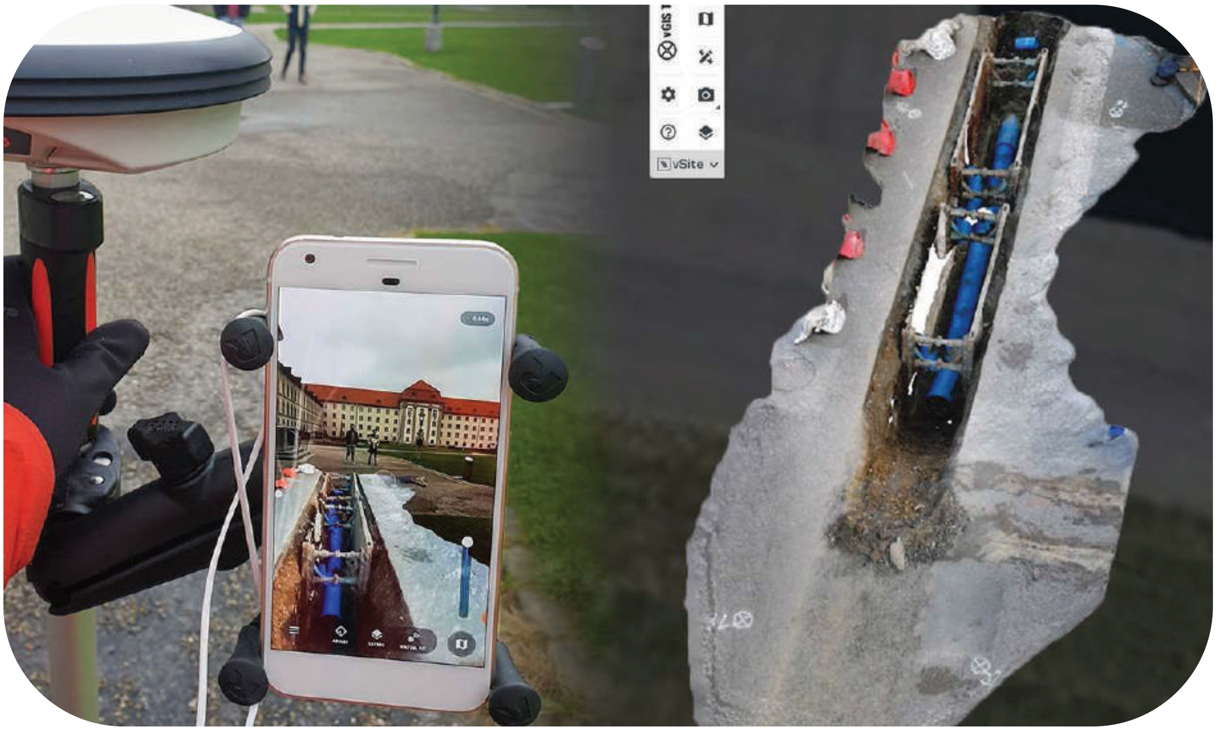

Figure 8 - VGIS Overview, from VGIS website, 2025

VGIS also utilizes beacon based location tracking and initial pose estimation to improve camera pose accuracy similar to the system employed by XYZ Reality (Boan, Jiajun and Bosché, 2025).

Finally, XYZ Reality stands out as one of the most advanced AR solutions tailored for construction. Its proprietary hardware, the ATOM headset. Delivers millimeter level accuracy, along with offline mode, issue tracking, reporting systems, data management, and cost estimation features (Xyzreality, 2025). Despite its capabilities, the primary limitation remains its reliance on specialized equipment, which may limit accessibility and adoption in certain contexts (Figure 9).

Figure 9 - XYZ Reality Overview, from XYZ Reality website, 2025

2.4. Final considerations

The present chapter has outlined the theoretical, educational, and practical dimensions of integrating Augmented Reality (AR) into the Architecture, Engineering, and Construction (AEC) industry, with a particular focus on its connection with Building Information Modelling (BIM) and game engines. From the initial definitions of AR as an overlay of digital information onto the physical environment, this review has demonstrated how these technologies are contributing to a shift from traditional manual methods to integrated digital workflows. The principal benefits were identified including real time tracking, safety improvements, productivity enhancement, process guidelines, and knowledge transfer, while recognizing current limitations such as cost, equipment dependency, and interoperability challenges.

The discussion on BIM provided further insights into how this methodology has evolved from a 2D design tool to a 3D data driven platform for decision making, spatial awareness, and lifecycle management. The literature indicates that when combined with AR, BIM extends its capacity beyond static visualizations into dynamic, on site applications that enable spatial understanding, reduce errors, and optimize logistics. The incorporation of kinaesthetic elements, multiple perspectives (monocular vs. binocular), and user interaction through Natural User Interfaces (NUI) further strengthens the integration between physical and digital spaces. This integration, however, is often restricted by hardware limitations, insufficient educational preparation, and the lack of standardized frameworks and implementation guidelines.

Furthermore, the chapter examined how the synergy between gaming engines (such as Unity and Unreal Engine) and BIM/AR applications has emerged as a practical approach for visualizing and simulating construction workflows. Projects like ELBIGMAC demonstrated how these platforms contribute to knowledge acquisition and pedagogical reform in Civil Engineering education. Technical aspects such as model optimization, data loss during file format conversions (IFC, FBX, OBJ), and mesh simplification were explored to highlight performance bottlenecks and interoperability challenges in real-world scenarios.

Finally, the practical and software review introduced eight relevant AR based tools (Cupix, Dalux TwinBIM, Arki, Gamma AR, SmartTek, Trimble Connect, VGIS, and XYZ Reality), each evaluated according to their capabilities, limitations, interoperability, data management, and suitability for implementation in the AEC context. The review emphasized the growing relevance of tools that integrate LiDAR, GPS, AI, and Common Data Environments (CDEs) while also pointing out concerns such as limited IFC support, reliance on proprietary hardware, and reduced accessibility for small scale or resource limited projects.

The results of this State of the Art provide the foundation for identifying current gaps in theory and practice, while reinforcing the necessity for scalable, adaptable, and cost effective AR applications in construction. The next chapters will focus on defining the methodological approach for this research, expanding the User Interface (UI), User Experience (UX), AR and game engine integration and implementation for construction purposes.

This page is intentionally left blank

3. AUGMENTED REALITY FRAMEWORK

3.1. Introduction

As previously established, game engines have emerged as essential solutions for the integration of BIM and AR technologies. These engines are not mere rendering platforms but rather comprehensive environments combining graphics, networking, physics, scripting, and artificial intelligence (AI) systems to enable real time 3D visualizations (Coronado, Itadera and Ramirez-Alpizar, 2023). Consequently, the selection of a suitable engine becomes crucial to ensure a seamless and efficient integration process.

This chapter aims to investigate various game engines, analyzing their advantages, disadvantages, and overall compatibility with the specific requirements of the assembly focused AR application developed in this research. Furthermore, the chapter will define the structure of a proposed BIM AR framework and its key implementation requirements. Finally, adjustments to this framework will be introduced, tailored specifically to the selected game engine.

3.2. Game Engines

Unity and Unreal Engine have already been introduced as viable tools to facilitate the integration of BIM with AR environments. Nevertheless, the spectrum of platforms and workflows enabling this gamification is far broader. This section outlines several engines that represent mainstream approaches for the development of XR solutions and the gamification of BIM related tasks.

A comparative study explored a selection of engines including Godot, PlayCanvas, Babylon, and Wonderland, contrasting their orientation toward either web based or device based XR development (Marín-Vega *et al.*, 2023). However, the scope of this study was limited by its lack of focus on industry specific applications such as AEC (Architecture, Engineering, and Construction).

Complementary insights were observed during a teaching experiment at Ithaca College, where game development courses were taught across different semesters using Unity and Unreal Engine respectively. The study found that Unity offered a gentler learning curve and reduced mental workload compared to Unreal. Moreover, students reported that Unity provided more accessible online resources, a clearer debugging environment, and a generally more intuitive development experience (Dickson *et al.*, 2017).

The comprehensive investigation conducted identified eight game engines with the potential to support BIM AR integration. These include Unity, Unreal, Godot, Wonderland, and others, each selected based on their proven or potential compatibility with AR and their applicability to AEC industry workflows (Table 1). It should be acknowledged, however, that there are numerous AR platforms not currently oriented toward BIM that may become relevant in the future, including Nuke, Verge3D, Three.js, React XR, and others.

To determine the suitability of these eight engines for the development of the present BIM AR application, a comparative analysis was performed. The evaluation criteria included:

1) Development Focus: Whether the engine prioritizes web-based or mobile-native development. A web-based orientation implies deployment through online browsers, which typically involves internet dependency.
2) Device Compatibility: The range of devices on which the resulting application can operate (e.g., smartphones, PCs, or HMD (Head Mounted Displays)).
3) BIM Integration: The level of interoperability with BIM formats, which may be supported natively or through plugins or Software Development Kits (SDKs).
4) Operating System Support: The operating environments in which the engine is functional and stable.

The results, compiled through online documentation and direct information from the engine developers, are presented in the following table:

Table 1 – Evaluation of Game Engines for BIM AR integration

Engine	**Development focus**		**Devices**			**BIM Integration**	**Operating Systems (OS) Development**		
	Web	**Software (App)**	**Phone**	**PC**	**HMD**		**Windows**	**iOS**	**Linux**
A-Frame	✓		✓	✓	✓	✓	✓	✓	
Babylon	✓		✓	✓	✓	✓	✓	✓	
Cryengine	✓	✓		✓	✓	✓	✓		✓
Godot	✓	✓	✓	✓	✓	✓	✓	✓	✓
PlayCanvas	✓		✓	✓	✓	✓	✓	✓	✓
Unity	✓	✓	✓	✓	✓	✓	✓	✓	✓
Unreal	✓	✓	✓	✓	✓	✓	✓	✓	✓
Wonderland	✓		✓	✓	✓	✓	✓	✓	✓

(Babylon.js, 2023, 2025; Emi, 2024; Godot Engine, 2024, 2025; A-Frame, 2025; Cryengine, 2025b, 2025a; Epicgames, 2025a; PlayCanvas, 2025a, 2025b, 2025c; Unity, 2025a, 2025; Wonderland Engine, 2025c, 2025b, 2025a)

As shown in Table 1, three engines Godot, Unity, and Unreal emerged as viable candidates for the aims of this research. However, Godot currently offers limited and unofficial mobile support (Godot Engine, 2025), which disqualifies it from being a feasible option for the mobile focused AR application proposed in this study. In contrast, Unity and Unreal stand as the most widely adopted engines in both gaming and AR development contexts (Gherghina, Olteanu and Tapus, 2013), with robust ecosystems and proven capabilities in immersive technology workflows.

Accordingly, the next section will present a focused comparison between Unity and Unreal, to evaluate which of the two is best aligned with the technical, usability, and integration needs of this research.

3.2.1. Unity engine VRS Unreal engine

Unity focuses on cross platform development and extensive user support. Projects developed in Unity are composed of one or multiple scenes and GameObjects, which represent individual 2D or 3D elements, or scripts similarly on how levels and assets are structured in a game environment. Additionally, Unity offers a comprehensive resource ecosystem through the Unity Asset Store and Unity Package Manager (UPM), where developers can access both free and paid assets (Coronado, Itadera and Ramirez-Alpizar, 2023).

Unity's asset library is notably robust and includes a wide range of resources, such as 2D and 3D models, audio, tools, visual effects (VFX), templates, artificial intelligence (AI) plugins, add-ons, and industry specific packages (Solmaz *et al.*, 2021; Unity, 2025a).

Contrary, Unreal Engine structures its projects based on assets and employs a system built around "actors" rather than Unity's GameObjects. Unreal is recognized for its advanced photorealistic visuals and strong rendering capabilities. Likely Unity, it features a digital marketplace to manage development assets. However, Unreal requires a more advanced programming expertise, primarily due to its reliance on C++ as its core programming language (Coronado, Itadera and Ramirez-Alpizar, 2023).

Research indicates that Unreal Engine offers capabilities for BIM manipulation, notably due to its strong interoperability, high fidelity real time rendering, and collaborative workflows (Van, Wong and Abbasnejad, 2025). It has demonstrated significant advantages in simulating real construction environments for project planning, risk assessment, and facilities management (Panya, Kim and Choo, 2023). Unity, while slightly less powerful in terms of rendering, excels in supporting interactive design experiences and offers real time visualizations that are more approachable for educational and lightweight industry applications (Van, Wong and Abbasnejad, 2025).

To facilitate the selection process, both engines were evaluated based on several criteria: programming language, ease of learning, cost, availability of online resources, and BIM integration capacity (Table 2). The results are based on official documentation and platform support resources from each engine.

Table 2 – Comparison of Unity and Unreal Game Engines for BIM AR integration

Criteria	Unity	Unreal
Programming Language	C#	C++
	Visual Scripting	Visual Scripting (Blueprints)
Learnability	Easy to Medium	Medium to Hard
	Intuitive UI	Complex engine and UI

	Terminology is easy to understand	
Resources	Free and certified official training programs	Free and official training programs
	Strong and extensive online community	Small and limited online community
BIM Data Metadata included	Asset Transformer (Pixyz)	Datasmith
	Asset Transformer Studio (€2 254)	Free
	Asset Transformer Toolkit included on Industry License	
BIM Data Not Metadata included	Free	
BIM Data Web Runtime	Not IFC supported	
	GLTF supported	
Device Implementation	Google ARCore XR	
	Apple ARKit XR	
AR Implementation	AR foundation	AR Framework
	Vuforia	
Cost	Personal: Free	Revenue $1 million under: Free
	Pro: €185 / monthly	Revenue $1 million over: Seat or royalty based
	Industry: €414 / monthly	Seat: €2 119.29 /yearly
		Royalty: 5% of revenue

(Epicgames, 2025a, 2025c, 2025b; Unity, 2025a, 2025c, 2025b, 2025; Unreal, 2025)

As reflected in Table 2, Unity surpasses Unreal in terms of online documentation and learning resources, largely thanks to its strong global community and official certification programs. Its user friendly interface, C# scripting, and low entry barrier make it ideal for beginners and non specialists. In contrast, Unreal caters more to experienced programmers and professional game developers.

One notable drawback is the cost of full BIM AR integration with Unity. Accessing metadata from IFC files typically requires a paid plugin. On the other hand, Unreal provides such features natively and at no cost. Nonetheless, the AR assembly app developed in this dissertation uses web requests and runtime web functionalities, meaning that BIM data (models, textures, etc.) will be accessed online at runtime. Therefore, no additional cost will be incurred for BIM integration on either engine.

Unreal has more logical and standardized tools than Unity, as the elements are loaded with some basic components, while these components need to be manually added in Unity (Dickson *et al.*, 2017). However, this also makes Unity more optimized specially for BIM AR implementation as several gaming components or characteristics are not applicable for BIM elements, such as gravity, jumping, falling or rolling.

Unreal Engine is known for its structured approach to asset handling. Basic components are automatically associated with new elements, making setup more standardized. Unity, by contrast, requires manual configuration of components (Dickson *et al.*, 2017). However, this manual approach allows for optimization and customization, especially beneficial in BIM AR development, where unnecessary gaming mechanics like gravity, jumping, or rolling can be excluded from the scene logic.

Regarding mobile AR support, two dominant frameworks are currently available: Google ARCore XR and Apple ARKit. ARCore supports both Android and iOS systems, whereas ARKit is exclusive to iOS. Both Unity and Unreal provide compatibility with these technologies. Furthermore, Unity includes Vuforia support, an AR toolkit that leverages computer vision to recognize and track images and 3D objects, enhancing marker based AR experiences (Solmaz *et al.*, 2021).

Ultimately, Unity was selected as the most suitable engine for this dissertation, primarily due to its accessibility, learnability, documentation richness, and broad developer support, making it more adaptable for the intended BIM AR assembly application.

3.3. Unity BIM AR framework, limitations and requirements

The following segment introduces the framework used for integrating BIM AR into Unity, with a specific focus on the development of the AR assembly application proposed in this research. The implementation was deliberately constrained to Android devices, while development and testing in Unity were performed on a Windows operating system (OS). This distinction is crucial, as the development of AR software for iOS entails a distinct approach, primarily due to the complexity and limited interoperability between iOS, Android, and Windows environments.

Moreover, the FBX file format has been recognized in literature as one of the most effective formats for exchanging 3D data, particularly in AR applications with BIM integration. Its precision, efficiency, and compatibility with 3D workflows have been emphasized as central to ensuring the accurate transfer of model geometry into game engines (Balali, Zalavadia and Heydarian, 2020; Van, Wong and Abbasnejad, 2025).

However, the AR application proposed in this dissertation will operate based on a runtime architecture, meaning that BIM models will be accessed via third party API (Application Programming Interface). This decision addresses concerns related to security, data control, and traceability of user interaction. Yet, online resources and platform documentation have confirmed that Unity does not support runtime FBX loading using free and open tools, thus presenting a significant limitation.

In contrast, the Asset Transformer (previously Pixyz), a Software Development Kit (SDK) and Unity plugin, does provide the ability to import various BIM formats such as FBX, IFC, OBJ, and GLTF at runtime while preserving essential (Unity, 2025). However, this solution is exclusive to the Unity Industry license, a paid tier, and therefore falls outside the scope of this dissertation, which relies exclusively on free and accessible solutions.

As a result, the BIM file format became a critical aspect of the framework development. After extensive analysis, the GLTF format emerged as the most appropriate choice. It offers a balance between geometry accuracy, metadata preservation, runtime support, and interoperability with Unity. GLTF has been increasingly adopted in AR pipelines, particularly for web based and runtime driven applications due to its lightweight nature and extensibility.

It is important to clarify that for Unity projects not based on runtime, other formats such as FBX, IFC, OBJ, or GLTF are also viable. However, for real time and remote data loading, GLTF remains the most flexible and reliable solution.

Another major consideration is the placement method of the 3D model. The research identified two primary approaches:

- Image Tracking (or AR Marker based): Relies on visual recognition, typically QR codes or printed markers, to initiate model placement.
- Location Based or AR Plane: Utilizes device sensors, including GPS and accelerometers, to determine physical coordinates and overlay digital content accordingly (Solmaz *et al.*, 2021).

Nonetheless, studies have highlighted limitations with AR markers based, particularly during initial AR camera calibration, which may result in position drift, misalignment, or loss of orientation data. Issues such as scaling errors, incorrect model orientation, and camera angle distortion may occur. Moreover, environmental factors such as lighting, reflections, and shadows can significantly degrade tracking performance and affect user perception of the 3D content (Boan, Jiajun and Bosché, 2025).

Considering all these variables software logic, BIM format, runtime capability, metadata requirements, and AR placement, the BIM AR Framework for Unity was developed (Figure 10). The framework addresses two scenarios:

- Scenario A: Application with embedded BIM models (offline).
- Scenario B: Application retrieving models at runtime via API (online).

Each scenario includes a branching logic depending on the presence or absence of metadata, resulting in two file conversion paths: one optimized for metadata usage, and the other focused purely on geometry. This dual path logic also accommodates performance optimization, where metadata may not be required, such as during early prototyping or visualization only tasks.

Finally, the BIM AR framework was structured into four main stages to guide the integration process:

1) Initial Decisions

- Selection of game engine (Unity)
- Target device platform (Android)
- Operating system for development (Windows)
- Logic of the software (Runtime vs. Embedded BIM models)

2) Format Setting
- Conversion to GLTF if runtime functionality is required
- Other supported formats for non-runtime approaches

3) Metadata Setting
- Inclusion or exclusion of metadata depending on project requirements
- Consideration of performance vs. data depth trade off

4) AR Setting
- Selection between image tracking or location based AR methods
- Consideration of environmental variables for robust tracking

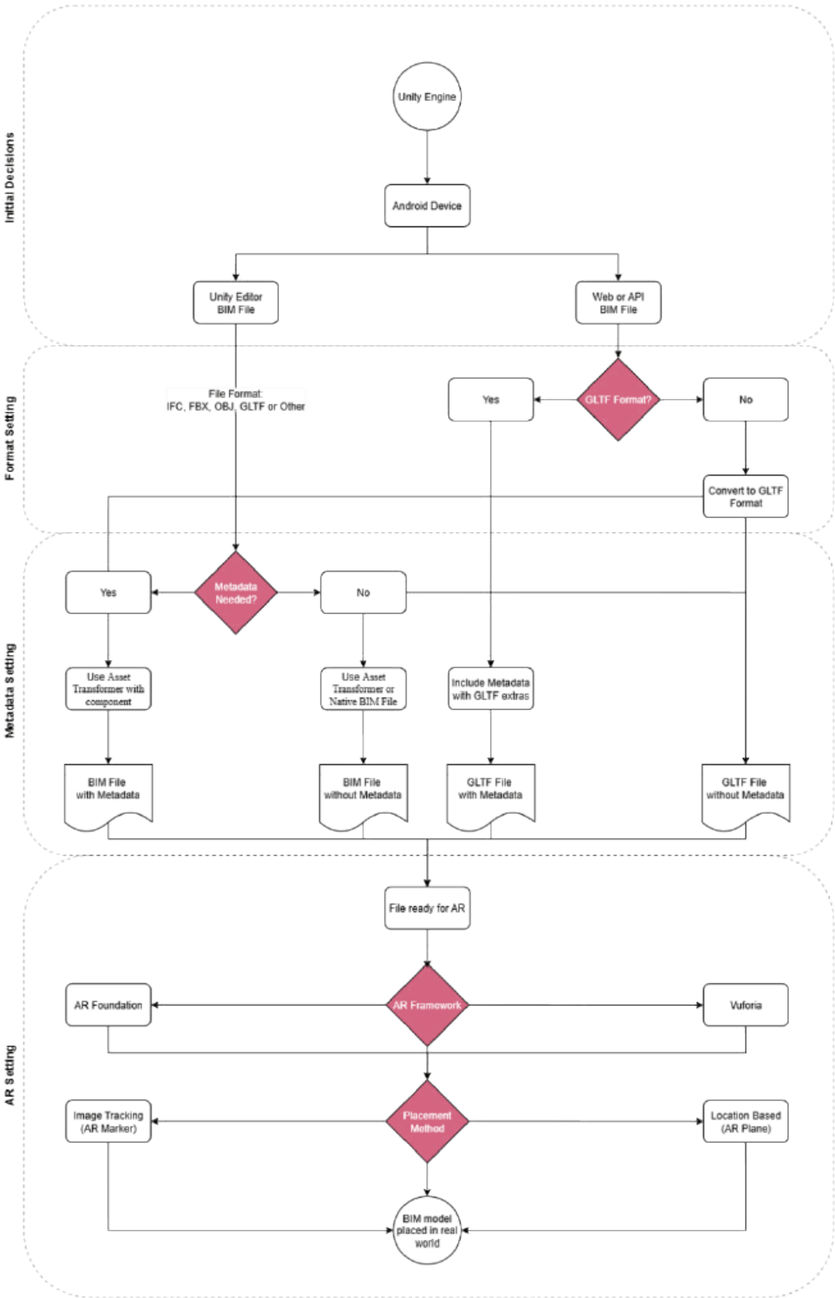

Figure 10 – Proposed BIM AR Framework for Unity Engine

3.4. Final Considerations

This chapter presented the technological foundation and methodological choices for the integration of BIM with AR through game engines, establishing a grounded path to the development of the AR assembly application. The initial exploration of multiple engines confirmed that although various platforms support AR experiences, not all align with the needs of the Architecture, Engineering, and Construction (AEC) industry, particularly regarding BIM interoperability and mobile deployment.

From the comparative analysis, Unity was selected as the most suitable engine due to its flexible architecture, accessible programming environment, wider learning support, and compatibility with Android devices. While Unreal Engine offers advanced visualization and BIM compatibility, its learning curve and professional focus presented a misalignment with the agile and inclusive development approach required by this investigation. These findings support earlier studies that highlight Unity's adaptability and community driven resources as key advantages in mixed reality development (Dickson *et al.*, 2017; Coronado, Itadera and Ramirez-Alpizar, 2023; Van, Wong and Abbasnejad, 2025).

The limitations of runtime file formats emerged as a critical constraint for real time BIM model integration. Although FBX remains a common and widely accepted BIM export format, its lack of runtime support in Unity's free tier excludes it from immediate use in this context. Conversely, the GLTF format offers runtime flexibility, reduced file size, metadata support, and higher compatibility with web based pipelines. This conclusion reinforces the strategic relevance of format selection at starting phases of the development process, a step that significantly impacts performance, visualization fidelity, and development cost.

Another key insight was the importance of AR placement logic. Between image tracking (marker based) and AR plane/location based, the choice affects not only the user experience but also the technical reliability of the application. Environmental conditions, such as lighting and shadows, were identified as critical variables influencing marker recognition and tracking consistency. Therefore, any implementation must be supported by adaptive testing, calibration, and user friendly design logic to ensure usability across real world conditions.

The proposed BIM AR framework for Unity (Figure 10) consolidates all these findings into a four phase structure initial decisions, format setting, metadata setting, and AR setting, providing a logical flow from planning to implementation. This framework also responds to the need for scalability, accommodating both metadata rich BIM workflows and visualization first applications. The inclusion of dual file preparation (with and without metadata) further ensures that the tool can be optimized according to task requirements and performance constraints.

In conclusion, the integration of BIM AR into Unity is not a matter of importing models alone, it is a deliberate process of balancing interoperability, usability, data depth, and technical viability, guided by both research and industry experience. As next chapters will demonstrate, the application of this framework will allow for the creation of a functioning, modular AR solution tailored to prefabricated assembly scenarios in the construction sector.

This page is intentionally left blank

4. USER INTERFACE FRAMEWORK

4.1. Introduction

Having previously established the essential role of engaging and user friendly interfaces (UI) in the integration of BIM, AR, and game engines, this chapter will explore and define the UI framework to be implemented in the mobile AR software to be developed. The analysis will also reference and build upon the previously reviewed eight software applications.

The main theoretical support for this chapter is based on the research and methodology defined by Franklin Hernández Castro in his publication "Usability Cookbook: Metodología para el análisis y diseño de aplicaciones", which outlines a twelve-step framework for software development (Figure 11). However, this chapter will also draw insights from other academic sources.

The Hernández Castro framework was selected due to the investigator's prior experience with its concepts and implementation schema.

4.2. Hernández Castro's framework

As described by Hernández-Castro (2016), the creation and design of software or applications involve twelve phases (Figure 11), each subdivided into smaller subphases. For example, Phase 1, or State of the Art, is subdivided into "Inventory" (for existing products) and "Needs and Requirements" (for new product development). This dissertation will focus on the Needs and Requirements phase, as it addresses a gap in AR mobile applications for assembly-based construction projects.

It is worth noting that Figure 11 includes two main color coded bubbles that encapsulate several stages, illustrating how later phases draw inputs from earlier ones. For instance, the salmon colored bubble with a dotted outline originates from Phase Four (Architecture A), which utilizes information from Phases 1, 2, and 3 to optimize outcomes. Similarly, the black dotted bubble (Architecture B) reflects cumulative inputs from all prior stages.

The following sections will present each of the twelve phases and their subcomponents, identifying which will be included in the development of the AR mobile application.

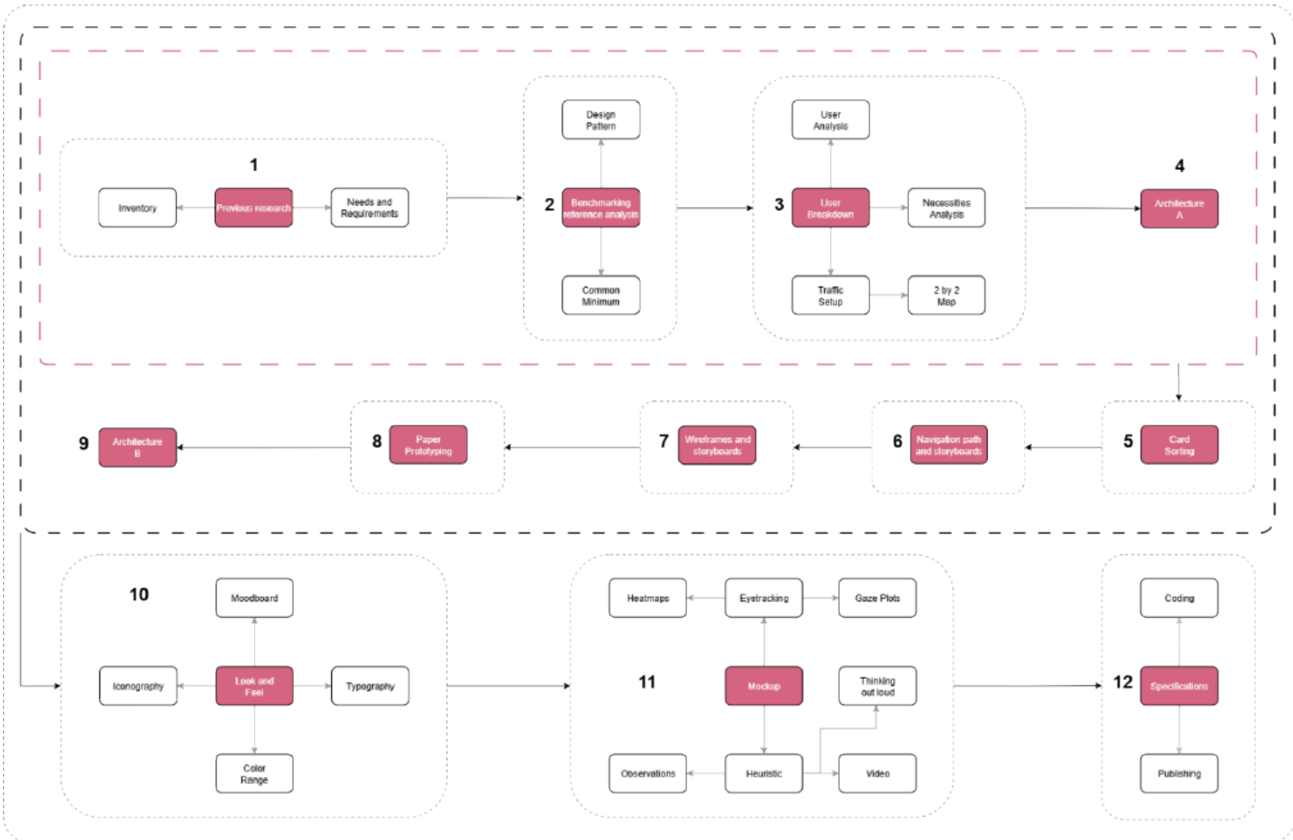

Figure 11 – Interpretation of Twelve Phases for Software Development from Hernández-Castro (2016)

4.2.1. Previous Research

This phase is subdivided into two components: Inventory and Needs and Requirements. The first relates to the study and practical analysis of existing tools with similar focus and functionality, while the second defines the objectives of the new product based on stakeholder inputs.

4.2.1.1. Inventory

This stage consists of summarizing the existing content, functionalities, and structure of comparable software. The analysis can be presented using diagrams, descriptive texts, or process maps, with the aim of understanding each software's architecture and identifying essential requirements (Hernández-Castro, 2016).

The most common method of representation involves node diagrams, where each node reflects a function, screen, or button, and the interconnections are illustrated by relational lines.

4.2.1.2. Needs and requirements

This subphase is used only when developing an innovative product for which no direct reference exists such as the current investigation. Here, hypothetical or preliminary requirements are gathered from stakeholders, designers, and programmers.

The outcome is a list of proposed functionalities and necessities that the tool should satisfy. At this stage, early testers may also contribute by identifying new, previously undetected requirements. The final product of this phase includes an inventory diagram, a synthesized list of requirements, and a preliminary architecture draft (Hernández-Castro, 2016).

4.2.2. Benchmarking o reference analysis

This phase studies existing initiatives to identify potential design patterns, such as color schemes, font types, interaction logic, or interface structure.

4.2.2.1. Design Pattern

This phase analyzes how similar software satisfies design requirements and identifies the advantages and disadvantages of each pattern (Hernández-Castro, 2016). For AR interfaces, emphasis has often been placed on best practices rather than established design patterns due to the lack of standardized structures(Börsting *et al.*, 2022).

However, a notable effort to establish basic design guidelines for Handheld AR (HAR) games was found in Xu et al. (2011) (Xu *et al.*, 2011), who proposed the following design patterns:

1. Device metaphors (to suggest action to user)
2. Control mapping
3. Seamful design
4. World consistency
5. Landmarks
6. Personal presence (representation of the user)
7. Living creatures
8. Body constraints
9. Hidden information

An additional pattern based evaluation was conducted for the ARScribble app, using functionality, usability, accessibility, and other criteria (Börsting *et al.*, 2022).

4.2.2.2. Common minimum

This stage is a compilation of analyzed tools, identifying which requirements are satisfied by each. The results are typically represented in matrix or tabular format (Hernández-Castro, 2016).

4.2.3. User Breakdown

This phase defines the target user groups and includes: 1) User Analysis, 2) Needs Analysis, and 3) Traffic Setup.

4.2.3.1. User Analysis

Users are classified into "model users" or prototype groups with common interests and needs. In this study, users will be segmented into "standard users" and "expert users" instead of traditional sociological categories (Hernández-Castro, 2016).

4.2.3.2. Necessities Analysis

Here, the tool's objectives are defined through hypothetical or real user scenarios. These can be represented as tables or diagrams. If multiple user groups share similar needs, they may be unified under a "frontend" label, and the analysis is then split accordingly (Hernández-Castro, 2016).

4.2.3.3. Traffic Setup

This process calculates the frequency of tool usage per user group or function. The result is an absolute traffic matrix, used to create a Pareto Chart, which helps prioritize functions representing less than 25% of the total needs (Hernández-Castro, 2016).

Additionally, this phase introduces the 2 by 2 Map, where the X-axis reflects usage frequency (frequent to infrequent) and the Y-axis reflects complexity (simple to complex). Each need or function is plotted accordingly.

4.2.4. Architecture α

Derived from phases one, two and three, it can be represented as a diagram, image or flowchart. The architecture must reflect:

1) Clear labels based on user terminology
2) Immediate visibility of high-frequency needs or features (based on the Pareto chart) (Hernández-Castro, 2016)

4.2.5. Card Sorting

The primary objective of this test is to validate the nomenclature and structural organization of the proposed Architecture α. The initial step involves identifying the core concepts to be evaluated and assessing their alignment with predefined user needs, functional requirements, and user group expectations. To maintain objectivity and minimize bias, descriptions or explanatory notes should be avoided, thereby eliminating any potential influence on participants.

Concepts selected for testing should be limited to a manageable number ideally fewer than 50 and no more than 70 and must represent static elements rather than actions. The goal is to assess whether the naming conventions are intuitive and whether their placement within the structure reflects a logical and user friendly hierarchy.

Two types of card sorting methodologies can be employed: open and closed. In open card sorting, users are free to group elements (representing needs or functions) according to their own logic, with all cards presented using a uniform visual style. In contrast, closed card sorting provides predefined categories and differentiates elements through color coding or hierarchical visual cues.

The test should be conducted in small groups of no more than five to six participants; however, each participant must complete the task individually. The outcome of this test is represented by a dendrogram, which visualizes how participants grouped the concepts. This output will serve as a diagnostic tool to identify inconsistencies in categorization, structural misalignments, and potential nomenclature issues, as suggested by Hernández (2016).

4.2.6. Navigation Path and Storyboards

Following the completion of the card sorting process (Step 5), the initial version of Architecture α (Step 4) must be revised and restructured based on the insights and feedback obtained. This iterative process will result in a refined version of the architecture, which will undergo a new round of evaluation to ensure improved usability and alignment with user expectations.

Concurrently, the creation of navigation paths is defined as the user's sequential actions to fulfil a specific need or function. The ultimate objective of this step is the optimization of these navigation paths, aiming to reduce the number of steps, time, or clicks required for users to complete a given task or action efficiently (Hernández-Castro, 2016).

4.2.7. Wireframes and Storyboards

Wireframes are simplified interface diagrams, intentionally devoid of color or detailed graphics, designed to simulate user interaction through predefined scenarios. These wireframes serve as a medium for evaluating two key aspects:

1) Hierarchical organization of the user scenarios
2) Application of consistent and functional design patterns.

The design of these diagrams should be informed by the navigation structure and architectural logic developed in previous phases. Additionally, wireframes must account for the contextual boundaries of the platform on which the interface will operate whether it be a mobile device, web browser, or another medium. This includes considering both external (device screen) and internal (application layout) margins and constraints (Hernández-Castro, 2016).

4.2.8. Paper Prototyping

This step involves manual interface simulation. Two roles are needed: one person acts as the "computer" executing interface logic, while another observes and documents the tester's behavior and interaction issues. These tests often validate specific design hypotheses (Hernández-Castro, 2016).

For continuous improvement of the interface and its interaction model, it is essential to collect iterative feedback from actual users through structured usability testing sessions (Urbancová, Vrabcová and Pacáková, 2024).

4.2.9. Architecture β

The final version of the architecture, developed after incorporating all feedback from previous phases, including card sorting, navigation analysis, wireframing, and paper prototyping

To ensure optimal usability and efficiency, this version reflects all prior feedback and must:

1) Prioritize most used functions
2) Use correct and clear nomenclature
3) Group functions logically
4) Limit visible functions to the most critical (Hernández-Castro, 2016)

4.2.10. Look and Feel

This phase involves the graphic design development of the tool, encompassing four key components:

1) Moodboard,
2) Typography
3) Color range
4) Iconography

The moodboard serves as a visual collage that conveys the anticipated aesthetic direction of the interface. It integrates examples of expected colors, icon styles, typography, and other visual elements that establish the overall tone and identity of the software.

Typography selection should be semantically aligned with the software's domain (e.g., construction, modular workflows), while also prioritizing readability. Font size, weight, and style must be adaptable to the target platforms (web, mobile, or other devices) to ensure usability across contexts.

The color range is typically initiated during the moodboard process. However, the final color palette should be refined using a color matrix, incorporating at least six variations to account for emphasis, contrast, and accessibility. This ensures both visual appeal and functional clarity.

Iconography should be standardized, minimalistic, and tailored to the specific functions defined in previous phases. Whenever a unique or context-specific function cannot be represented using existing icon libraries, custom icons must be designed. The main software icon must also be created in this phase, considering resolution, file format, color palette, and compatibility with target operating systems (e.g., Android, iOS, Windows) (Hernández-Castro, 2016).

A consistent visual identity including color usage, icon styles, transitions, layout logic, and organizational structure is critical. Maintaining this consistency facilitates intuitive navigation and rapid information retrieval, which is particularly essential in high-pressure environments such as construction sites and time-sensitive modular installation projects (Norouzi *et al.*, 2015).

4.2.11. Mockup

This phase focuses on the evaluation of the interface's look and feel, incorporating elements developed in prior stages such as diagramming, wireframes, navigation structure, and user frequency of interaction. The assessment is typically conducted using a combination of heuristic evaluation and eyetracking techniques, both aimed at validating user experience and interface clarity.

4.2.11.1. Eyetracking

Eyetracking involves monitoring and analysing users' visual attention as they interact with the interface. Specialized equipment is required to record eye movements during guided test scenarios, allowing researchers to assess how users respond to specific actions, interface elements, or navigational paths. The two principal outputs of this technique are heatmaps and gaze plots:

Heatmaps visually represent the areas of the interface most frequently viewed by users. These are displayed as color coded overlays ranging from red (most viewed) to green (least viewed) on top of the interface layout. While heatmaps provide valuable insight into visual focus and user attention, they are qualitative in nature and not considered precise data. For reliability, heatmaps should be generated using a sample size of at least 30 participants.

Gaze plots, on the other hand, provide a sequential visualization of users’ eye movements across the interface. They depict the order, timing, and duration of visual fixations, along with entry and exit points. This data is especially useful for understanding how users process information hierarchically. Unlike heatmaps, gaze plots can yield reliable insights with a smaller sample size typically 5 to 6 participants are sufficient for meaningful analysis.

4.2.11.2. Heuristic test

Heuristic testing shares methodological similarities with paper prototyping but is conducted without specialized eyetracking equipment. Instead, it focuses on assessing the interface's visual hierarchy, interaction logic, and the sequence of reading or attention based on established usability principles. This technique relies on expert evaluators identifying usability issues based on a set of predefined heuristics (e.g., consistency, visibility of system status, error prevention).

Both techniques contribute to a comprehensive understanding of the system's user experience, enabling designers to refine visual structure, clarify navigation paths, and enhance overall usability before moving into high-fidelity development.

4.2.12. Specifications

This final stage integrates all previous outputs, wireframes, look and feel, navigation, and architecture β to produce the complete software specification, ready for coding and implementation.

This section reviewed the Hernández Castro framework in detail, covering phases from wireframes and navigation paths to benchmarking and paper prototyping. However, due to time and resource constraints, the present investigation will only apply seven out of the twelve phases, which will be explained in the next section.

4.3. Proposed Framework for BIM AR integration

The Hernández Castro's framework is a proposal aimed at traditional software environments, where the evaluations are primarily focused on design and research techniques. However, its structure reveals limitations in terms of the user application relationship, especially when it comes to stimulating engagement within new applications operating in Extended Reality (XR) environments. In consequence, the researcher proposes a modified version (Figure 12), expanding on the original framework with new phases and additional subdivisions, aiming to enhance engagement during usability testing and to promote a more robust integration of BIM and AR within the user interface (UI).

It is important to acknowledge that the original Hernández Castro framework already integrates several design and research techniques across its phases. For instance, card sorting is used to validate terminology and the structure of "architecture α", followed by navigation paths and storyboards to incorporate card sorting feedback. Paper prototyping then enables a reassessment of this input, leading to the creation of a new "Architecture β." Finally, the mockup phase compiles the accumulated improvements and adjustments across all prior steps.

This methodology, in essence, aligns closely with what is known as the "Wheel model", a cyclic design process that allows continuous iterations of design, analysis, and implementation. Its strength lies in its agility, flexibility, and responsiveness to user feedback (Hartson and Pyla, 2012). In contrast, the "Waterfall model" adheres to a linear sequence, starting from analysis, then design, followed by implementation, integration, and maintenance, where each stage must be completed before the next can begin (Sommerville, 2017). The proposed framework in this research aligns more closely with the Wheel model, as it supports multiple revisions and continuous user engagement.

Considering the above, the proposed framework preserves the twelve-phase structure but categorizes them under three main development stages: "architecture α" (1 to 4), "architecture β" (5 to 9), and the final building phases (10 to 12). The first group compiles and analyses inputs from prior stages, such as previous research, benchmarking analysis, and user breakdown. The second one considers "architecture α", navigation path and storyboards, wireframes and storyboards, and prototyping. Finally, the third category entails the implementation of the "look and feel," creation of the mockup, final specifications, and consolidation of usability tests from all previous stages.

This schema mirrors the original Hernández Castro approach. However, its differentiating factor lies in the subdivision of UI design into two distinct streams: general UI and BIM AR UI. This separation begins in stage four (architecture α) and continue on nine (architecture β). The general UI encompasses all functions unrelated to BIM AR such as menu navigation, user account settings, or project management. In contrast, BIM AR UI specifically addresses functions tied to AR visualization, marker or plane detection, and the interaction with BIM content within an AR environment.

This division is not merely structural. General UI will continue to follow conventional methodologies and design techniques as proposed by Hernández Castro. However, the BIM AR UI stream will emphasize usability testing as its core method for gathering user feedback, due to the inherent limitations of applying traditional techniques such as navigation paths or paper prototyping within extended reality

(XR) environments. Still, techniques like card sorting can remain useful, especially when applied to multidisciplinary testers such as general workers, engineers, architects, managers, or clients.

The first usability test for the BIM AR UI is expected to occur during Stage Seven (Wireframes and Storyboards). At this point, the wireframes produced from "architecture α" and refined via card sorting will form the basis for the initial BIM AR UI mockup. This early mockup will not include design identity elements such as typography or iconography but will instead focus on interaction logic and the BIM AR system's initial structure. The results of this first round of testing will provide the necessary inputs to refine Architecture α and to initiate the virtual prototyping process.

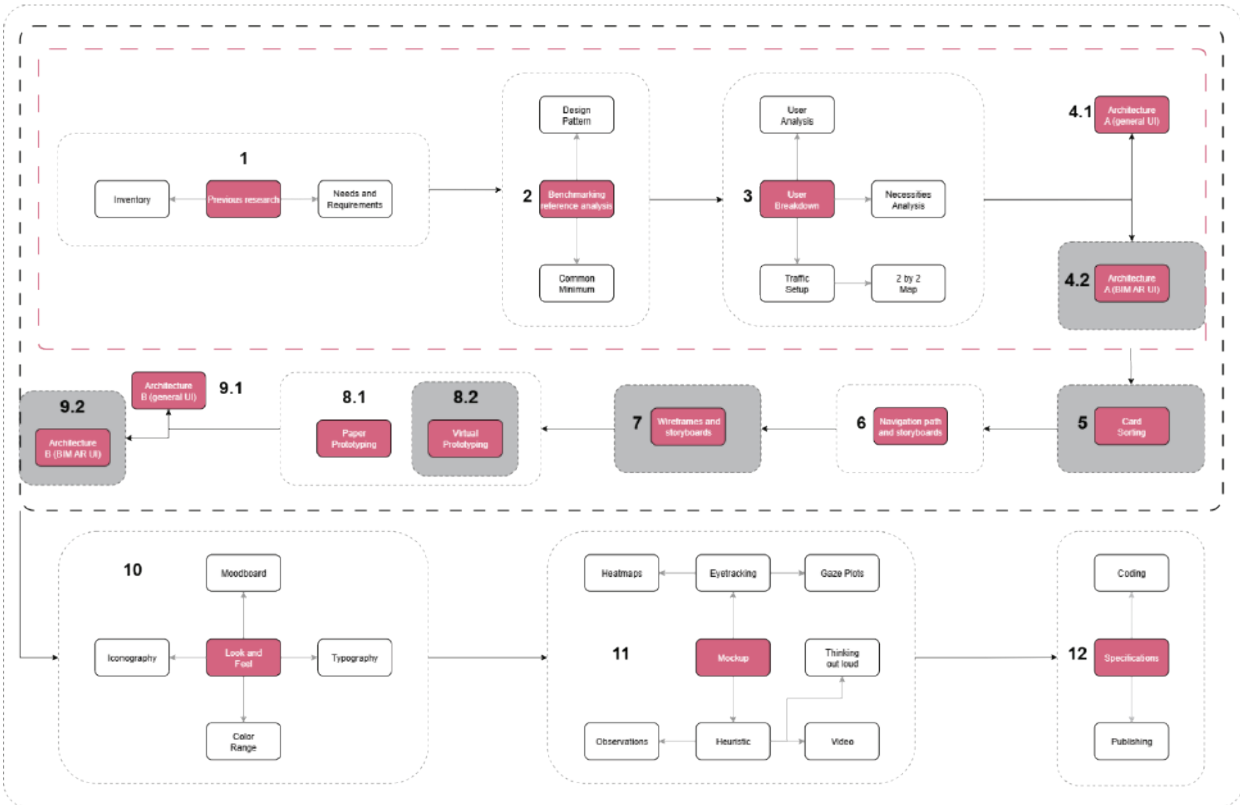

Figure 12 - Proposed XR Environments Framework for BIM AR Integration

The main modifications of the proposed framework are presented in Figure 12 and highlighted in gray. This color indicates elements associated with the BIM AR UI concept, which may also be reflected in the General UI.

To delimit the scope of this investigation, and due to time and resource constraints, the proposed framework was only implemented across seven out of the twelve total phases (Figure 13). Stages such as card sorting, navigation paths and storyboards, paper prototyping, mockup, and final specification were not executed.

Additionally, the implementation will be divided into two distinct sections. The first will address the "architecture α" phase and the wireframes and storyboards (stage seven), as detailed in this chapter. The second section, presented later, will cover the "architecture β" and virtual prototyping stages, reflecting the real world implementation of the proposed structure.

Finally, the following sections define the virtual prototyping phase and the methodology of usability testing for BIM AR UI. Both will be supported by test data, user feedback, recommendations, and evaluation of each development phase.

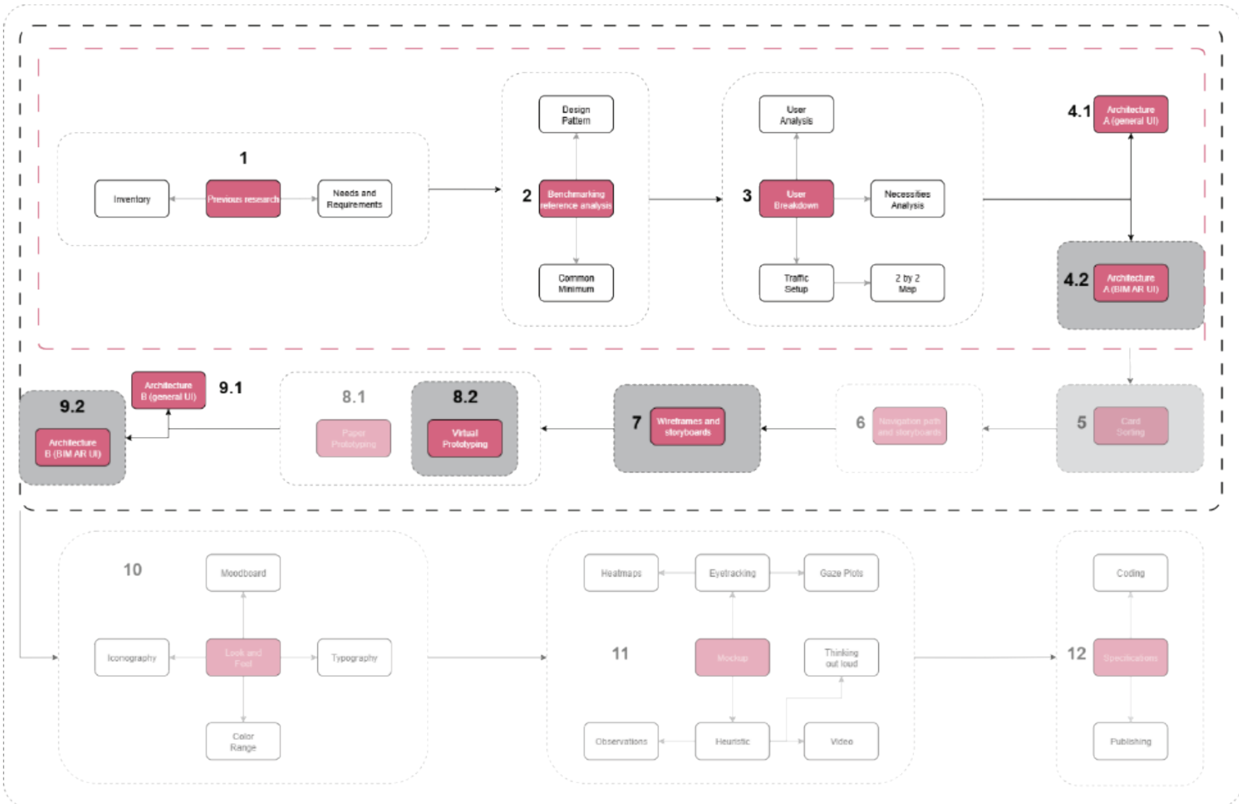

Figure 13 - Implemented XR Framework for BIM AR Integration

4.4. Virtual Prototyping

This phase was introduced as a direct response to the limitations of traditional 2D testing methods such as paper prototyping, when applied to immersive 3D software. AR applications, by their nature, operate in a three dimensional space. Attempting to simulate these experiences in two dimensions risks confusion, disconnection, and a reduction in user engagement during the transition from traditional tools to BIM and AR environments.

Moreover, graphical instructions or interactive animations placed directly on 3D models significantly improve spatial cognition and mental representation. AR reduces the user's cognitive workload by eliminating the need to mentally project or visualize elements, allowing them to interact with data and context directly (Fan *et al.*, 2020).

3D graphics based UI are consistently shown to be more impactful than text based interfaces, particularly when dealing with spatial data. Furthermore, essential information should be persistently accessible not hidden or presented only upon demand (Jeffri and Awang Rambli, 2021). Virtual prototyping, therefore, aims to reduce cognitive load while improving interaction quality during usability testing.

Mental workload in this context refers to the alignment (or misalignment) between the task's demands, the conditions of execution, and the user's available skills. As a result, visual overload must be avoided. Interfaces must be lean, clear, and intuitively navigable especially when applied to AR use cases in construction or on site environments (Jeffri and Awang Rambli, 2021).

Additionally, AR platforms that integrate BIM data must aim to maximize engagement. This can be achieved through real time communication, shared 3D environments, personalized interfaces based on user role, and tools adapted to specific site requirements. In this context, engagement means efficient, easy, and productive interactions between the user and the data (Van, Wong and Abbasnejad, 2025).

The virtual prototyping process involves developers closely observing user behavior, decisions, questions, and emotional responses while interacting with early stage of BIM AR prototypes. These tests are primarily focused on the logic and structure of the AR experience, and can be performed on mobile devices, computers, or dedicated AR headsets, depending on the development platform in use.

4.5. Usability test for BIM AR UI

As seen in Hernández Castro's framework, design and research techniques are essential to gather feedback throughout the software development lifecycle. The proposed framework maintains this approach for general UI development. However, for the BIM AR UI, usability testing must be prioritized to collect accurate, relevant, and real world feedback.

A Usability Test is a structured session where selected testers interact with a product while verbalizing their experience. Meanwhile, an observer records behaviors, struggles, and observations. The objective is to identify usability issues that can compromise the tool's performance. Poor usability often results in frustration, confusion, mistakes, delays, and ultimately, failure to complete tasks (Hertzum, 2020). An optimal usability process must account for mental workload, evaluating how design and interface performance align with user expectations and cognitive limitations (Jeffri and Awang Rambli, 2021).

To develop an inclusive and adaptable BIM AR tool, it is critical to ensure that participants in usability testing are diverse. Diversity guarantees that a broad range of perspectives, disciplines, and working conditions are represented from stakeholders and designers to engineers and site managers (Van, Wong and Abbasnejad, 2025). This variety ensures that the tool can address a spectrum of real world needs (Yu, Jeon and Koo, 2022). Key elements of usability testing include clearly defined tasks, user verbalization, emotional observation, and evaluation of the user system relationship (Hertzum, 2020).

One essential insight from prior studies indicates that untrained workers tend to adapt faster than experts during assembly usability testing, likely because they are less conditioned by prior systems (Funk *et al.*, 2017). However, experts tend to benefit more from AR implementations, as they are already familiar with expected outcomes and data requirements (Jeffri and Awang Rambli, 2021).

In conclusion, usability testing must be applied at multiple stages, including Architecture α (BIM AR UI), card sorting (if applicable), wireframes and storyboards, virtual prototyping, Architecture β, and the final mockup phase. Each of these tests will contribute unique insights to ensure that the final product is optimized for clarity, adaptability, and high performance in construction site environments.

4.6. Final Considerations

A key insight from this chapter is that BIM AR UI design is not static or universal, but rather dynamic and sensitive to context. As highlighted by Chu (2023), the complexity of the task directly influences the AR UI structure, dictating its interactions, instructions, and types of feedback to be provided. In simpler terms, the complexity defines the interface.

In fact, the impact of AR is not uniform across use cases. As demonstrated by Jeffri (2021), AR can produce limited or even negligible results for low complexity tasks, while it tends to deliver significantly higher impact in complex scenarios, where users require spatial understanding, real time guidance, or collaborative visualization.

This correlation reinforces the importance of careful UI planning and modular logic separation between general UI and BIM AR UI. The structure adopted here has allowed the development to isolate traditional interaction components from those that are AR dependent, enabling the integration of extended reality within a clear and scalable design logic.

Furthermore, performance, usability, and mental workload are all influenced directly by the quality and clarity of UI design (Kim, Nussbaum and Gabbard, 2019). The current development path driven by usability tests, feedback cycles, and architecture iterations is expected to improve each of these factors iteratively, supporting a more intuitive and effective AR supported construction tool.

The chapter also introduced a novel way of connecting tool benchmarking with user centred breakdowns, traffic modelling, and design pattern synthesis. This not only strengthens the proposed framework but also ensures that future stages. Particularly the implementation of architecture β and mockups will be grounded on both empirical testing and interface usability logic.

Looking ahead, the next stages will require consolidating feedback from virtual prototyping and usability testing into the BIM AR UI design refinement. More advanced evaluations, including AR logic calibration, user feedback capture, and performance tracking, will be essential to achieve a robust and adaptable solution. These tasks will continue to prioritize flexibility, intuitiveness, and relevance to modular construction workflows especially in the prefabricated kitchen and bathroom use case defined in this research.

This page is intentionally left blank

5. FRAMEWORKS IMPLEMENTATION

5.1. Introduction

This chapter examines the integration and practical implementation of the two frameworks established in the previous sections: the User Interface (UI) Framework and the BIM AR Framework. These frameworks were not conceived in isolation, but rather as foundational layers for the development of a unified application **"ARsembly",** the tool proposed and built during this dissertation.

Due to time and resource constraints, the implementation of the UI Framework was conducted only partially, focusing on its most critical components to ensure usability and interaction consistency. Conversely, the BIM AR Framework reached complete implementation, enabling real time visualization and interaction with BIM models within an augmented reality environment.

The integration of both frameworks under the **"ARsembly"** platform demonstrates their combined potential in supporting efficient data visualization, on site assembly guidance, and enhanced construction communication within the AECO industry. The outcomes of this integration are presented in the following sections, illustrating the developed system architecture, its core functionalities, and its overall contribution to advancing digital workflows through the convergence of BIM and AR technologies.

5.2. Initial Implementation of Framework Proposed

The following section presents the results from the initial implementation phase of the proposed framework, focused on "architecture α" and the development of wireframes and storyboards. This first implementation serves as the foundation and input for the subsequent implementation stage.

5.2.1. Previous Research

5.2.1.1. Inventory

As the tool developed in this dissertation is a new application, not derived from previous models, similar AR centered softwares were examined to establish a baseline of common functionalities. These tools were analyzed based on industry sector, AR implementation, advantages, and limitations. The eight references introduced in Chapter 2 guided this analysis.

Table 3 - Benchmarking AR Applications Relevant to BIM in the AECO Industry

Name	Industry	Advantages	Disadvantages	AR use
Cupix	- Scan to BIM - Digital Twins	- 360 virtual tours - BIM integration - Cloud collaboration	- No interactive AR guidelines - Complex User Interface (UI	- To site inspection - 3D data into real environments
Dalux	- Construction - Facility Management	- On-site 3D models - BIM integration - Easy and friendly UI	- Complex for non-technical users - Limited guidance.	- For visualization of models on-site - Installation verification
DarfDesign (Arki)	- Architecture - Interior Design	- Easy and friendly UI - Conceptual design. - Educational support	- Limited accuracy and scale - Build for non-industrial environments. - No BIM integration	- Easy use of AR technology in the design sector not in construction
Gamma AR	- Construction - Contracting	- AR guidelines - Issue tracking. - Dimensions and measurements validation. - BIM Integration	- Complex (UI) - Detail BIM requirements - No modular block support	- Good AR visualization and tracking - Focus on construction process and verification
SmarTek (House Tour)	- Architecture - Real Estate - Marketing	- Virtual tours - Engaging client functionalities - Fast and easy AR scenes - Mobility of AR models	- Mainly for promotional - No details visualization	- Client engagement
Trimble Connect (Geospatial)	- Construction - Geospatial - Surveying	- High AR precision and accuracy - BIM live integration	- Hardware dependency - Complex (UI)	- Feasible for large-scale project
VGIS	- Geographic Information System (GIS) - Construction - Project Management	- GIS based for AR - Underground AR visualization	- Specialist on MEP infrastructure	- Precision for on-side visualization - For visibility if hidden infrastructure
XYZ Reality	- Construction - Industrial	- AR accuracy - BIM integration - AR guidelines	- Special equipment required - Complex for non-technical	- Measurements system - Ideal for step-by-step installations

Based on the comparative analysis presented in Table 3, Gamma AR, Trimble Connect, and XYZ Reality emerged as the most suitable references due to the following characteristics:

1) High dimensional and measurement precision
2) Advanced BIM integration
3) Presence of embedded AR guidelines
4) Support for tracking systems in construction environments

Nevertheless, key features from all platforms were also noted for consideration including on site inspection capabilities, ease of use, user friendly interfaces, client engagement, and movable AR models for flexibility.

5.2.1.2. Needs and requirements

From this reference analysis, an initial set of necessities and requirements was outlined, to be meet by the proposed application (Table 4).

Table 4 - Defined Needs and Requirements for the Proposed BIM AR Software

#	Need or Requirement	Description	Related Stakeholder
1	Real scale visualization	1:1 scale visualization on site of the prefabricated element with dimensions (height, width and thickness).	- Workers - Clients - Designers
2	Dynamic floorplan location	Display prefabricated elements in the floorplan (2D or 3D), including the connections (structural, mechanical and electrical).	- Installers - Inspectors
3	Categorization system	The elements must be filterable by name, type, size, color, material or any other identifier.	- Designers - Inspectors - Mangers
4	Tracking dashboard	Progress tracking of each prefab in the assembly and installing stage. It should include visual indicators (in progress, completed or pending).	- Managers - Inspectors - Workers - Installers
5	Assembly and construction breakdown	Division of work or task in separate workflows like structural, mechanical, electrical, finishing and other	- Workers - Inspectors - Installers - Managers
6	Comments or specification section	Each element will have their own notes for any necessary clarification or relevant information.	- Managers - Designers - Installers

7	Connection setup	Structural, mechanical, electrical and other type of building connections must be visually highlight and with relevant specifications	- Installers - Inspectors - Designers
8	Quantity takes off	Auto count of components (sufficient or insufficient) per each element and for their assembly, construction and installation process.	- Installers - Inspectors - Designers - Managers - Workers
9	Intuitive UI	Intuitive and easy to use UI for non-technical users. Ideally, each user type will have an independent and customized UI.	- Workers - Installers - Clients - Designers
10	Issues dashboard	Ability to flag errors in the assembly, construction or installation process, including missing part, malfunction or other	- Workers - Installers - Designers - Inspectors
11	BIM Integration	Interoperability with BIM formats and software such as Revit, CAD, ArchiCAD and IFC models.	- Designers - Managers
12	Accessibility and compatibility	The final product must be supported by at least one of the most common AR platforms (iOS, Android, PC or AR glasses)	- Designers - Clients
13	Security control	Data and access protection by allowing a connection with an API.	- Designer - Clients - Manager
14	AR guidelines	Visual and engaging AR guidelines on how to use the tool correctly.	- Designer - Clients - Manager
15	BIM guidelines	Visual and engaging BIM guidelines or requirements to start a project (format or import/export)	- Designer - Clients - Manager

The 15 identified necessities were classified into three core categories (Figure 14):

1) Usability: Core operations needed to drive the application's primary functions
2) Scalability and Accuracy: Dimensions, alignment, and element integrity
3) Collaboration: Communication, annotation, task assignment, and dashboard usage

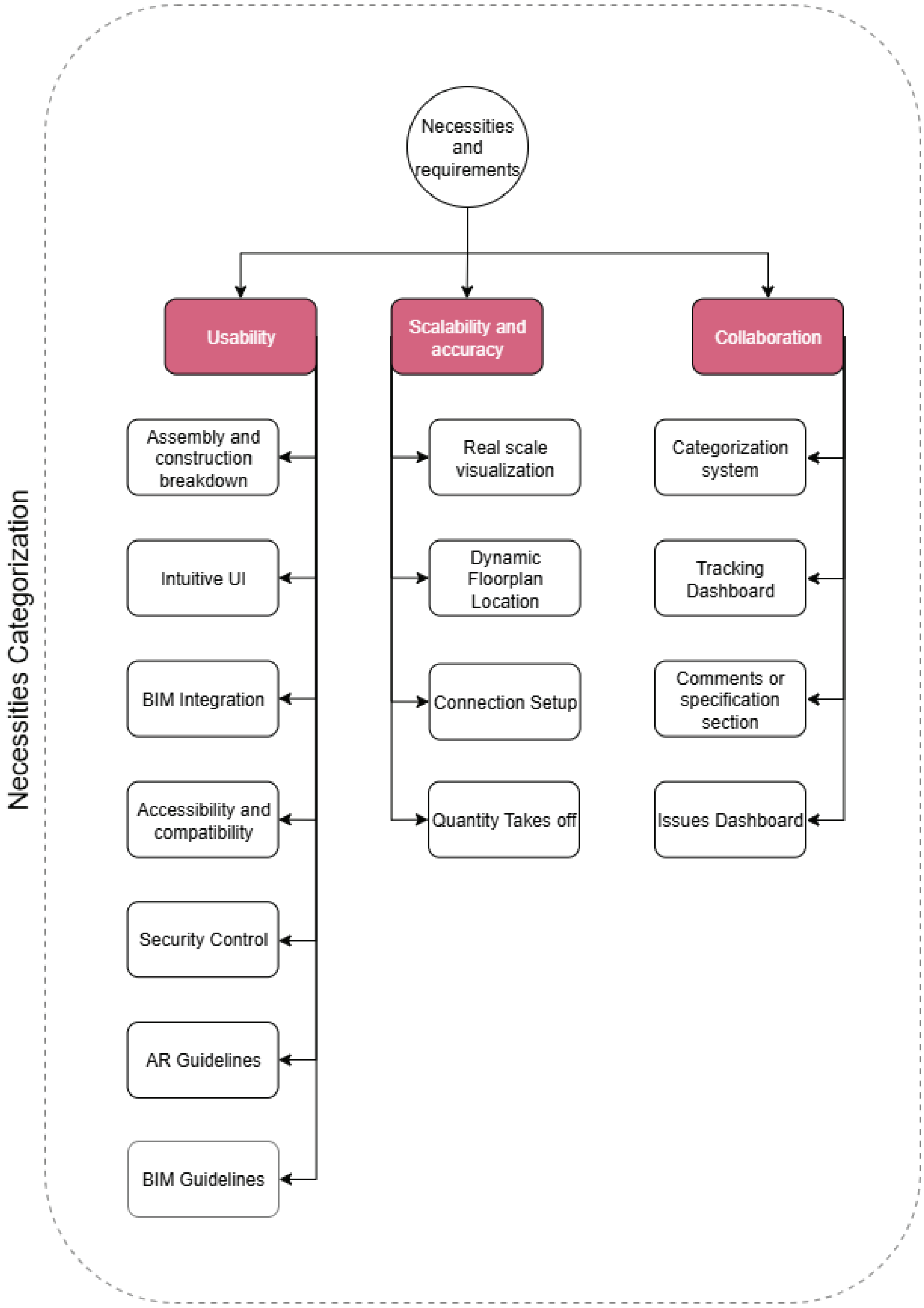

Figure 14 – Categorization of Identified Application Needs

5.2.2. Benchmarking o reference analysis

As defined in Section 5.2.1.1, Gamma AR, Trimble Connect, and XYZ Reality were selected as the key references. The benchmarking phase focused on analyzing these applications to extract useful design patterns and define a “common minimum” for the proposed tool.

5.2.2.1. Design Pattern

Due to hardware limitations, full testing of XYZ Reality and Trimble Connect was not feasible. XYZ Reality requires proprietary hardware, and Trimble Connect requires a Head Mounted Display (HMD) and subscription. Consequently, Gamma AR was chosen for a comprehensive benchmarking analysis due to its full mobile AR compatibility and availability via a 30 days free trial.

However, a limited evaluation of Trimble Connect was still possible:

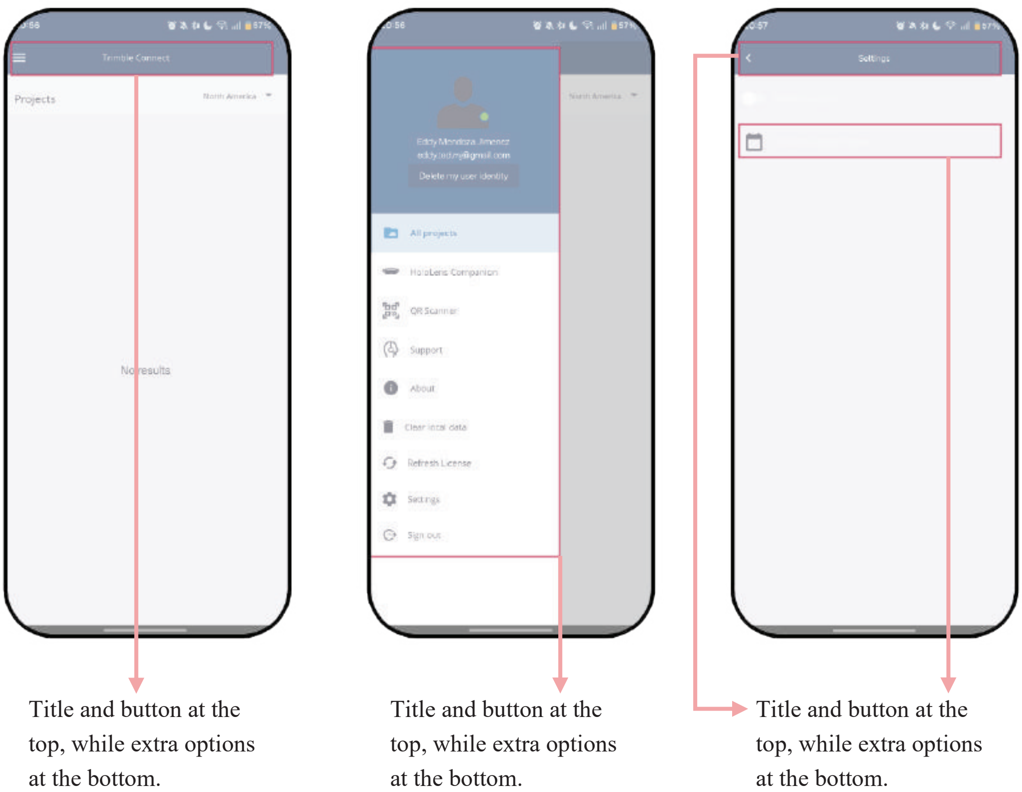

Figure 15 – Trimble Connect Design Pattern

Trimble's layout consistently employs a top bar featuring the app name, screen title, and access to user options via popup menus. Nonetheless, usability and readiness issues were identified for instance, poor contrast between white background and white icons affected readability on the settings screen (Figure 15).

By contrast, Gamma AR offers a more intuitive structure. It includes an authentication screen with access options (email, Autodesk, Procore, BIMCollab), and interface design guided by top, bottom, and sidebars (Figure 16). Pop up panels, screen transitions, and clear titles enhance usability and offer guidance throughout the AR experience (Figure 17).

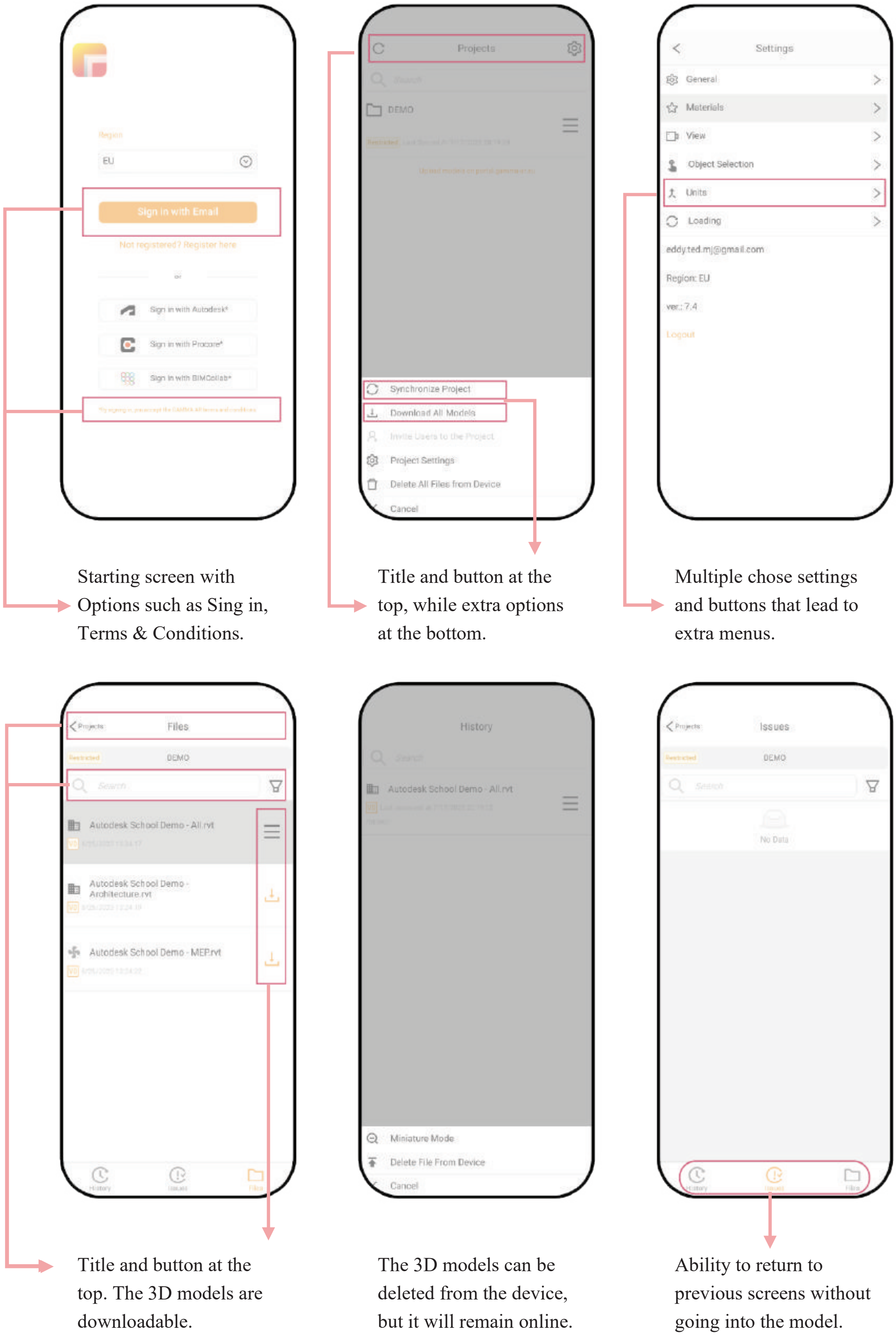

Figure 16 – Gamma AR General Design Pattern

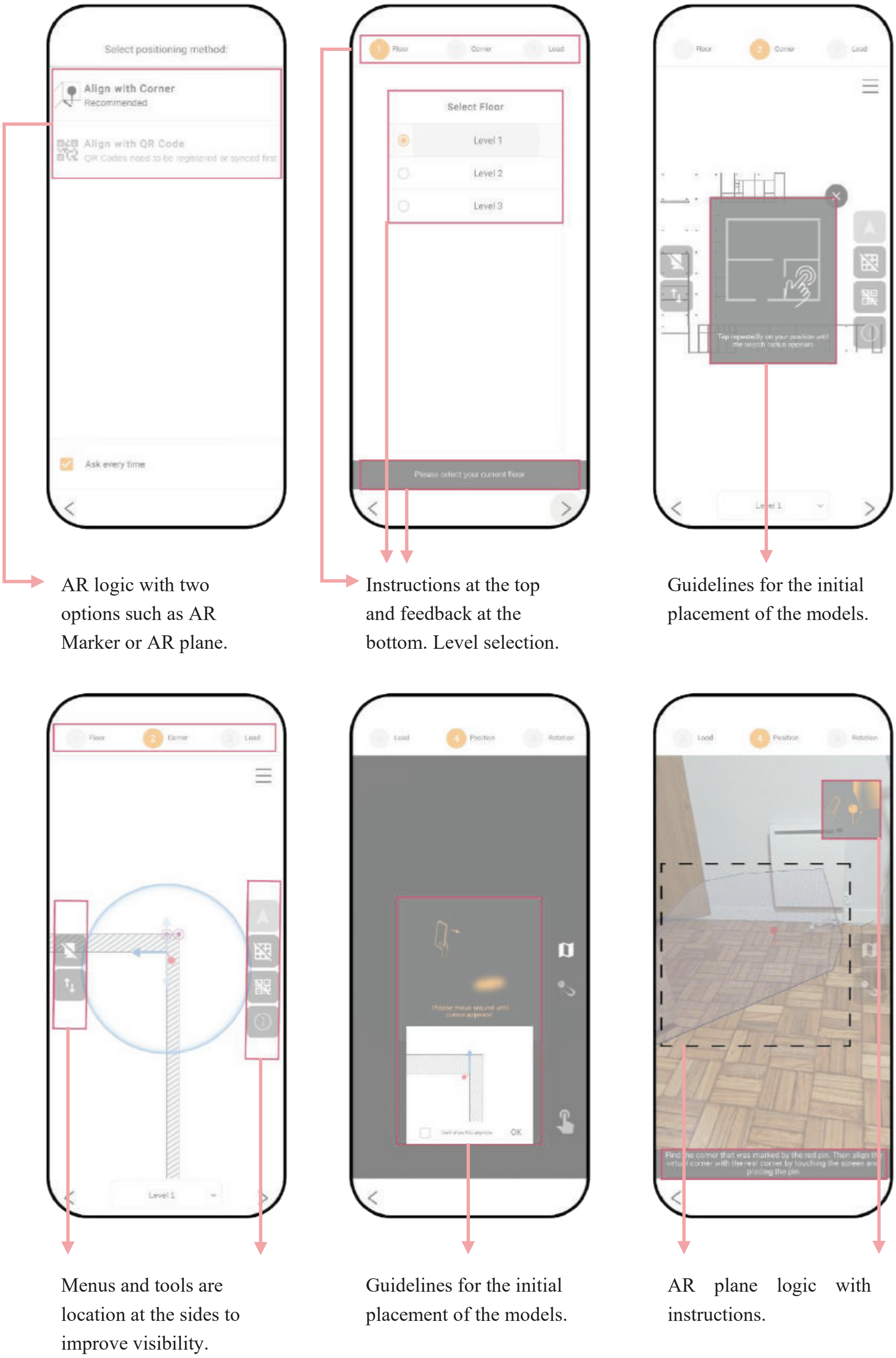

Figure 17 - Gamma AR Placement Design Pattern

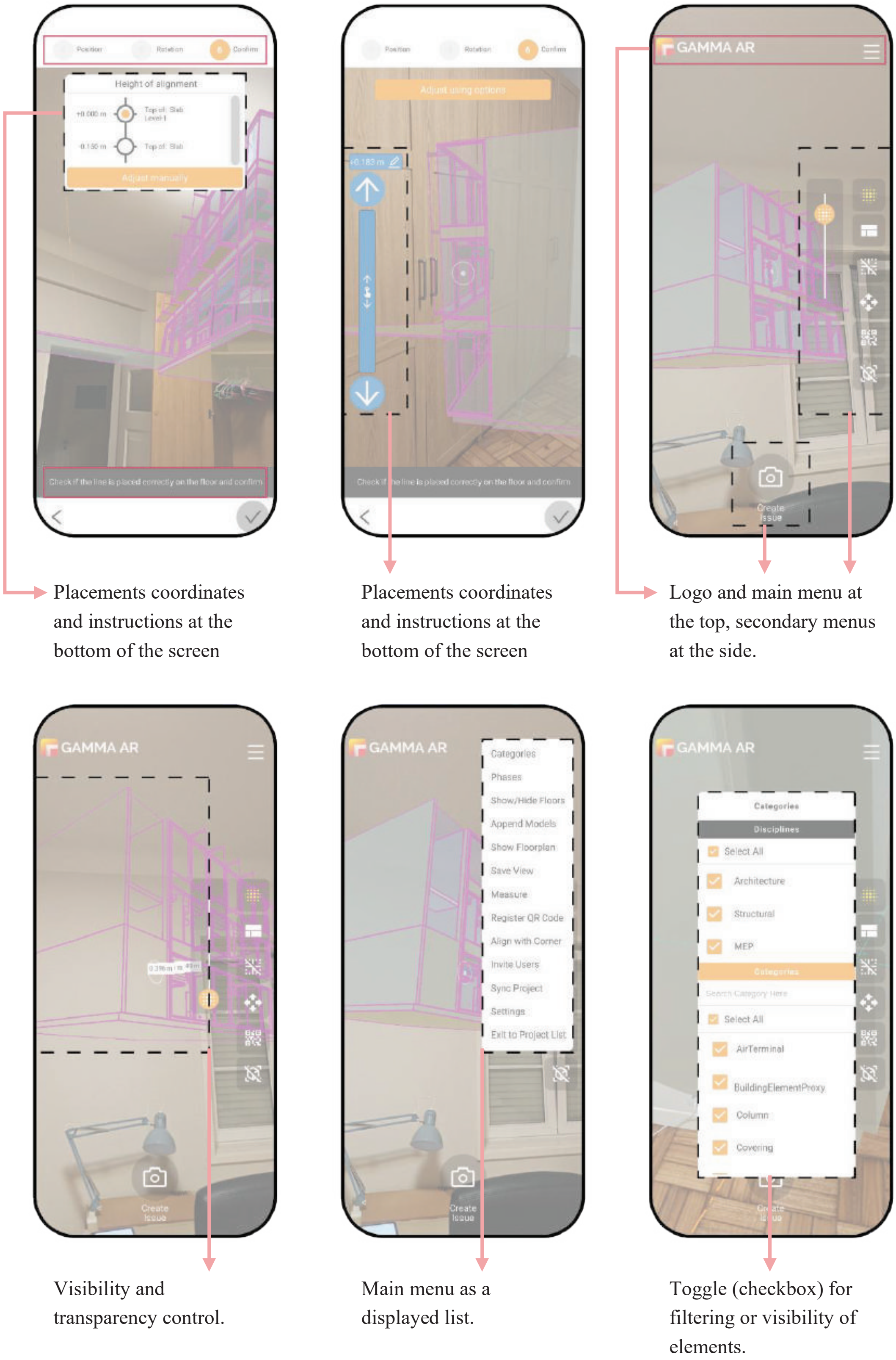

Figure 18 - Gamma AR Model Manipulation Design Pattern

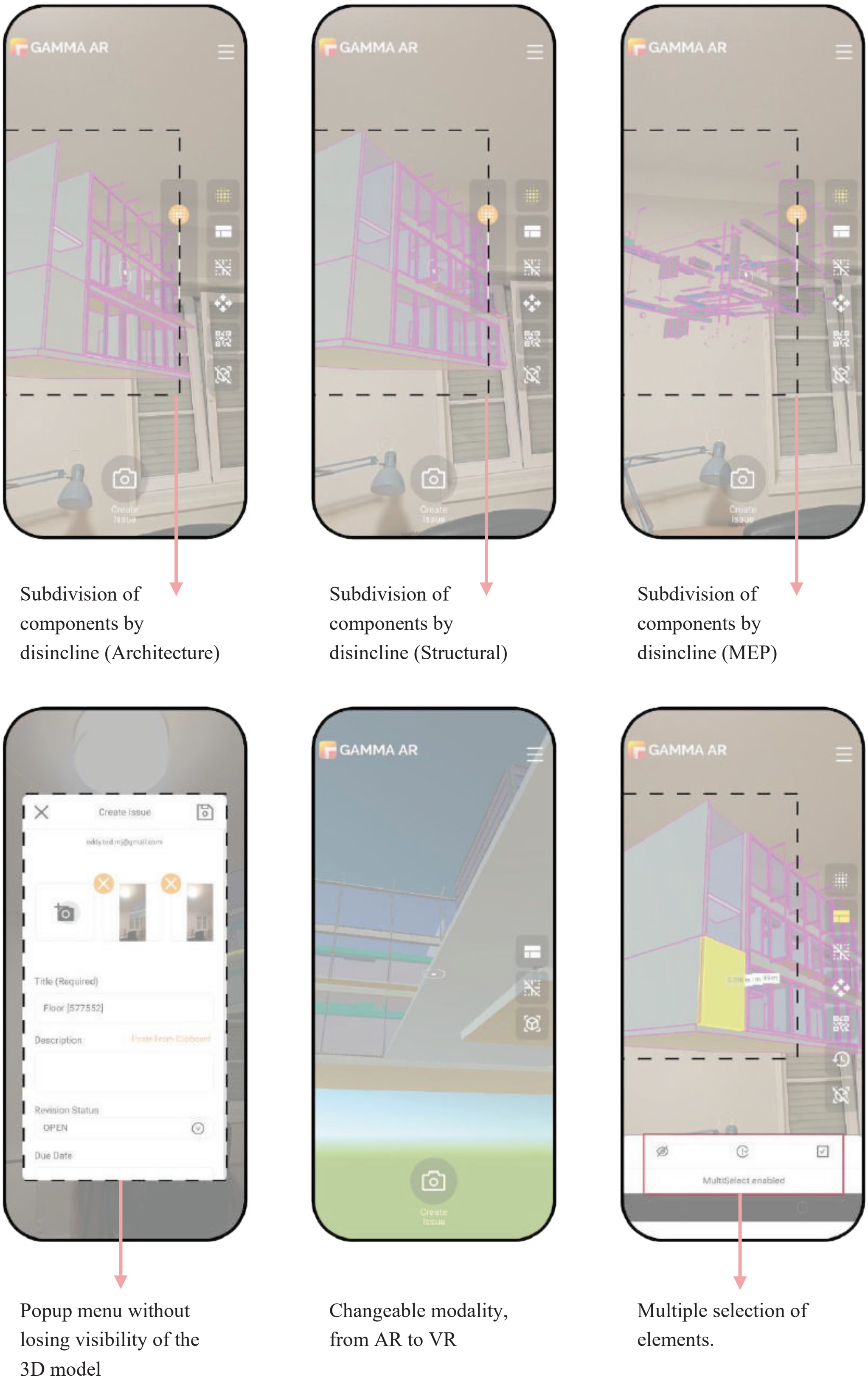

Figure 19 - Gamma AR Selection Design Pattern

In the BIM AR interface, Gamma AR allows 3D model segmentation by discipline (e.g., architectural, mechanical, structural) via toggle menus (Figure 18). It also supports:

1) Transparency controls
2) Element selection
3) Switching between AR and VR modes
4) Issue creation workflows

Despite these features, icon-function association was a noted weakness. For example, the transparency toggle uses an orange circle with no intuitive visual association, and element selection is linked to a generic rectangular icon, which lacks context (Figure 19).

The design patterns identified are summarized in Figure 20:

1) Top bar for screen title, branding, and navigation
2) Bottom bar for guidance (instructions, alerts)
3) Sidebars in 3D mode for tools and element controls
4) Multi-platform authentication screen
5) Clear onboarding and AR usage flow
6) Pop-up screens without hiding the AR view

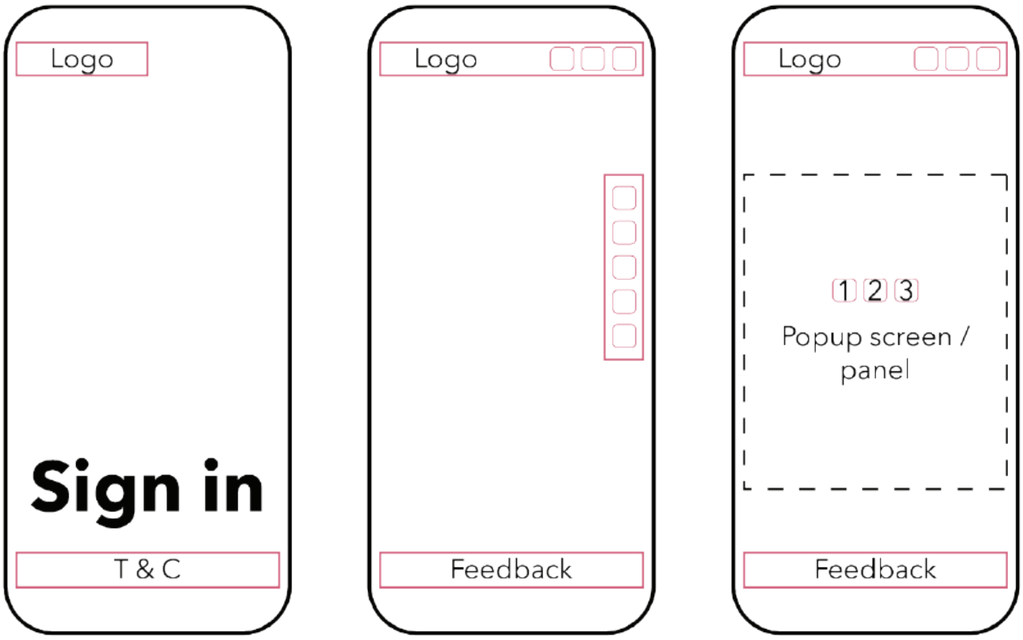

Figure 20 – Summary of Identified Design Pattern

5.2.2.2. Common minimum

The Common Minimum Standards (CMS) were defined by cross referencing the 15 necessities (Section 5.2.1.2) with features across the eight tools (Table 5).

Table 5 – Comparative Analysis of Relevant App's CMS and Necessities

#	Necessity	Cupix	Dalux	DarfDesign	Gamma	SmarTek	Trimble	VGIS	XYZ
1	Real scale visualization	✓	✓		✓		✓	✓	✓
2	Dynamic floorplan location	✓	✓						
3	Categorization system		✓		✓		✓		
4	Tracking dashboard	✓	✓		✓			✓	✓
5	Assembly and construction breakdown				✓				
6	Comments or specification section	✓			✓			✓	
7	Connection setup						✓		✓
8	Quantity takes off								
9	Intuitive UI		✓	✓		✓		✓	
10	Issues dashboard	✓	✓		✓			✓	✓
11	BIM Integration	✓	✓		✓		✓		✓
12	Accessibility and compatibility	✓	✓	✓	✓	✓	✓	✓	✓
13	Security control		✓		✓		✓	✓	✓
14	AR guidelines				✓				✓
15	BIM guidelines								

The results indicate:

1) Gamma AR: 10 out of 15 necessities met
2) Dalux: 9 met
3) XYZ Reality: 8 met
4) Trimble Connect: 7 met

Despite Dalux's strong performance, its complex UI and failure to meet necessity 8 (Connection Setup) led to its exclusion as a primary reference.

5.2.3. User Breakdown

Once the necessities, reference tools, and design patterns were established, the next step involved linking these to realistic end users. The analysis assumed a modular construction scenario specifically, AR supported assembly for prefabricated kitchens and bathrooms.

5.2.3.1. User Analysis

Six archetype users were defined to reflect diverse profiles and roles (Table 6). Each archetype includes a short narrative with strengths and weaknesses.

Table 6 – User Archetypes

Name	Archetype	Narrative	Positive Aspects	Negative Aspects
Tete	Hands-on (On site worker)	Assembly and installation worker, who is in the front line daily. They need clear and simple guidelines with minimum interaction as they wear gloves and safety equipment. They like to complete their work in a faster and efficient way	- Experience with physical work - Innovative in practical executions - Quick learner	- Limited BIM or technical capacity - Resistance to new digitals tools
Eliana	Controller (Inspector)	Overseas multiples tasks, like assembly, installation, tracking and inventory. They will use the tool to validate tasks and recollect data for reports. They prefer visual data instead of text or numeric, this helps for a quick understanding of the data.	- Responsible - Organized - Multitasking - Problem solver - Data oriented	- Overloaded with work - Overloaded with detailed information - Limited experience with assembling work

Ale	Boss (Project Manager)	In charge of the client updates, while monitoring the time, reports, errors alignment and accuracy of the final product.	- Strategic thinker - Compliance oriented - Data driven - Technical skills and BIM understanding	- Limited on-site experience - Possible usability overlook
Andrea	Visionary (Client)	They will use the tool to visualize the completion or progress in the project. They prefer simple, elegant and clear jargon or explanations. They value intuitive design over technical one and are more visual engaged.	- Curios and willing to learn - Innovative - Visual learner	- Limited BIM or technical capacity - Extensive report or issues explanation
Gabo	Stockpile (Logistics or inventory manager)	Manage deliveries, reports, materials and inventory at the project and elements levels. They provide the right number and materials at the right time.	- Materials tracker - Organized - On-site presence - Process oriented	- Manual counts and possible errors - Limited BIM or technical capacity - Limited experience with assembling work
Yenny	Planner and coordinator (BIM Coordinator / Manager)	Ensures the interoperability of all files and make updates. While verifying the accuracy of the model with the real-life elements. They process reporting data and QA/QC.	- BIM expert - Data driven - Detail oriented - Involve in all process	- Limited experience with assembling work - Limited on-site experience - Precision obsessed - Work overload

The analysis revealed two core user categories:

1) BIM Experts: Data-oriented, but limited in on site experience
2) Non BIM Experts: Field oriented, but less data fluent

Each group brings valuable but distinct perspectives to usability design.

5.2.3.2. Necessities Analysis

After establishing the user archetypes, they were mapped to the 15 necessities (Table 4). This helped identify the relevance and impact of each requirement per user type.

Table 7 – Comparative Analysis of Needs and User Archetype

#	Necessity	On-site worker	Inspector	Project Manager	Client	Logistics manager	BIM Manager
1	Real scale visualization	✓	✓	✓	✓	✓	✓
2	Dynamic floorplan location	✓	✓		✓	✓	✓
3	Categorization system		✓	✓		✓	✓
4	Tracking dashboard	✓	✓	✓	✓	✓	✓
5	Assembly and construction breakdown	✓	✓				✓
6	Comments or specification section	✓	✓	✓		✓	✓
7	Connection setup	✓	✓			✓	✓
8	Quantity takes off		✓	✓		✓	✓
9	Intuitive UI	✓		✓	✓	✓	
10	Issues dashboard		✓	✓	✓	✓	✓
11	BIM Integration		✓	✓			✓
12	Accessibility and compatibility	✓	✓	✓	✓	✓	✓
13	Security control		✓	✓	✓	✓	✓
14	AR guidelines	✓	✓	✓	✓	✓	✓
15	BIM guidelines		✓	✓		✓	✓

Unsurprisingly, the BIM Manager and Inspector archetypes scored highest with 14 out of 15 necessities, due to their role in supervision, task assignment, and guideline creation. In contrast, the Client had only 8 necessities, as they do not interact with dashboards or technical BIM features.

5.2.3.3. Traffic Setup

To prioritize necessity visibility within the interface, a Traffic Setup simulation was developed. The model assigned weekly usage percentages to each necessity for each user (100% = full week), reflecting their relative frequency and criticality (Table 8).

The most prominent necessities based on usage were:

1) Tracking dashboard – 95%
2) AR guidelines – 62%
3) Issues dashboard – 61%
4) Quantity take-off – 50%

Table 8 – User Archetype Based Traffic Setup

#	Necessity	On-site worker	Inspector	Project Manager	Client	Logistics manager	BIM Manager	Total Traffic (%)
1	Real scale visualization	20	10	10	5	2	1	48%
2	Dynamic floorplan location	10	10	2	5	2	1	30%
3	Categorization system	1	10	5	0	15	10	41%
4	Tracking dashboard	10	15	20	25	15	10	95%
5	Assembly and construction breakdown	20	5	0	0	0	1	26%
6	Comments or specification section	5	5	10	5	5	10	40%
7	Connection setup	10	10	0	0	2	2	24%
8	Quantity takes off	5	10	5	0	20	10	50%
9	Intuitive UI	5	2	10	10	5	2	34%
10	Issues dashboard	1	15	15	10	10	10	61%
11	BIM Integration	0	2	2	0	0	25	29%
12	Accessibility and compatibility	2	1	1	10	2	2	18%
13	Security control	1	1	5	10	2	1	20%
14	AR guidelines	10	2	10	20	10	10	62%
15	BIM guidelines	0	2	5	0	10	5	22%
Total per user (%)		100%	100%	100%	100%	100%	100%	

5.2.4. Architecture α

As outlined earlier, architecture α was separated into General UI and BIM AR UI segments. Both were developed based on the results from reference analysis, benchmarking, and user breakdown. The transition occurs when the user selects a 3D model, triggering the switch to BIM AR UI logic.

5.2.4.1. Architecture α (General UI)

This portion handles non BIM AR data project setup, tool guides, user profile, settings, and model selection (Figure 21). In addition, it was divided into two phases, starting set up and general set up, setting the foundation for the AR experience.

Notably, 3D models are split between:

1) Assembly: Constructing modular units (e.g., kitchens, bathrooms)
2) Installation: On-site placement and coordination within the building

The section concludes with model selection, which initializes the AR logic and begins spatial placement.

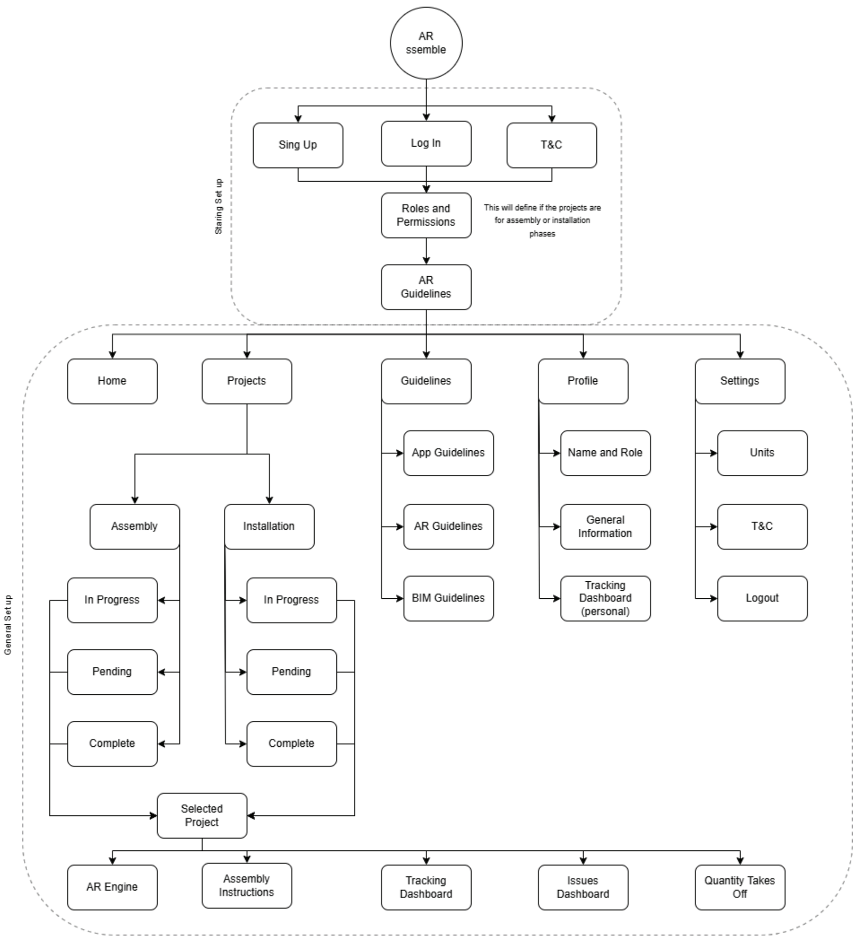

Figure 21 – ARsembly General UI Architecture α

5.2.4.2. Architecture α (BIM AR UI)

Once a model is chosen, the BIM AR engine initiates, offering five core functionalities:

1) AR Engine
2) Assembly Instructions
3) Tracking Dashboard
4) Issues Dashboard
5) Quantity Take off

Given the Traffic Setup results, the current implementation focuses on AR Engine and Assembly Instructions due to the high priority of AR guidelines (62%) (Figures 22, 23 and 24).

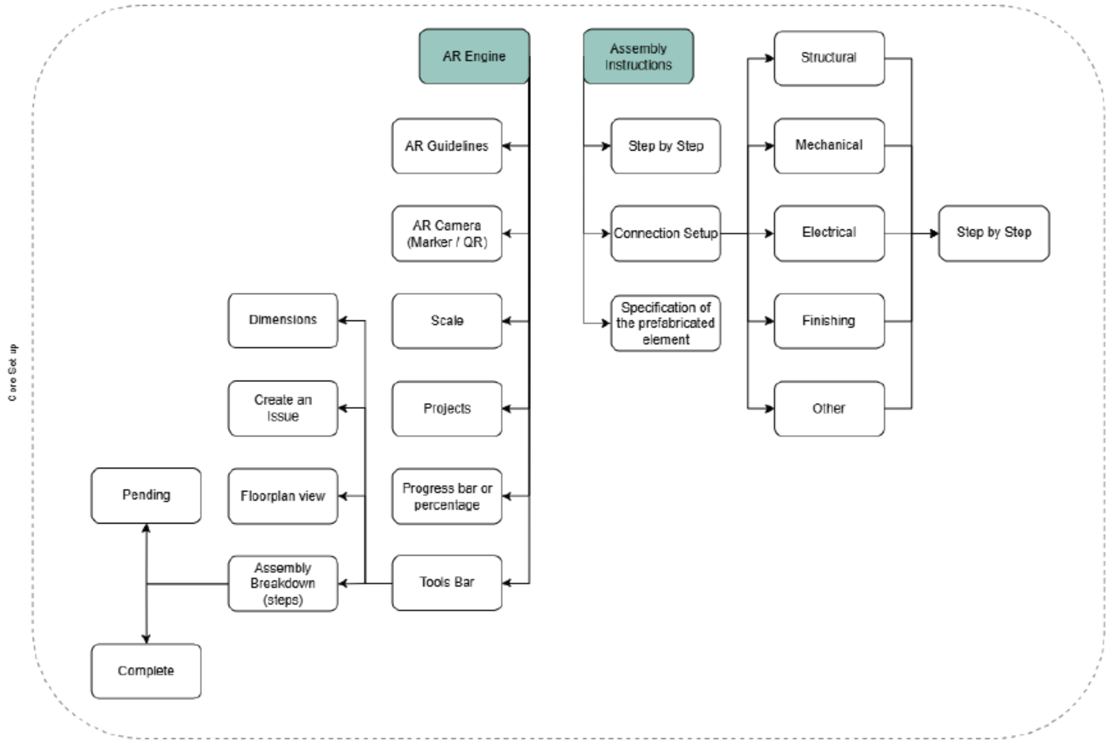

Figure 22 - ARsembly BIM AR Architecture α, Section 1

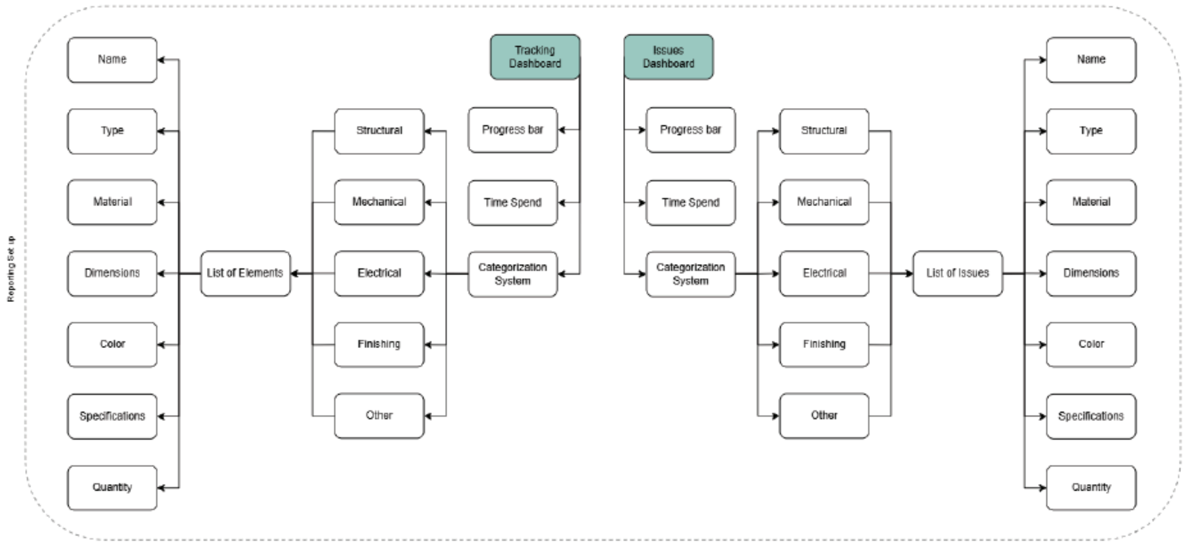

Figure 23 - ARsembly BIM AR Architecture α, Section 2

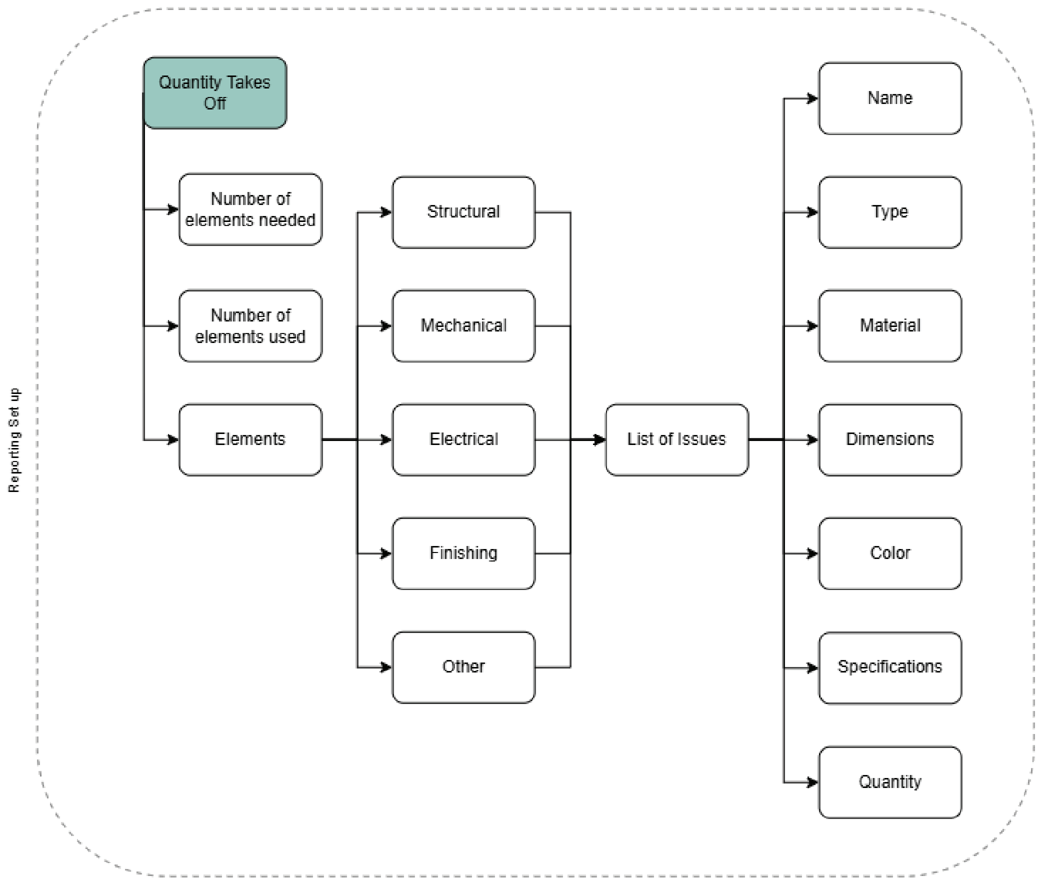

Figure 24 - ARsembly BIM AR Architecture α, Section 3

5.2.5. Wireframes and Storyboards and Usability Test

Wireframes and storyboards were also divided between general UI and BIM AR UI. General UI tested authentication logic (necessity 13) and device compatibility (necessity 12) on an android device (Figure 25). The BIM AR UI validated object placement and guideline display using AR markers (Figure 26).

Design patterns implemented in this test included:

1) Sign up screen
2) Top navigation bar
3) AR marker recognition

The usability test conducted with Unity identified multiple issues:

1) Color contrast was too bright
2) Text size was too small
3) Authentication failed to function properly
4) AR marker recognition was unstable

Figure 25 - Wireframes and storyboard of General UI

Figure 26 - Wireframes and storyboard of BIM AR UI

5.2.6. Virtual Prototyping

The inputs from the previous segments led to the construction of the first BIM AR prototype. This initial version aimed to observe and analyse the interaction between the General UI and the BIM AR UI, identifying possible overlaps, logic improvements, or interface issues. In this phase, the full architectures were implemented and tested, including:

1) Staring setup: authentication process, role selection and AR guidelines.
2) General Setup: Project selection, user profile and settings
3) Core setup: AR engine and Assembly instructions

However, the reporting setup was only drafted at this stage, as priority was given to testing the AR engine and general functionality to obtain substantial user feedback. The virtual prototyping process was repeated four times, each iteration incorporating insights from the previous round. The outcomes were as follows:

Firstly, positive reception to the newly introduced color range and the discovery of a security flaw were the main results of the first virtual prototype (Figure 27). The color scheme was chosen based on the investigator's design preferences, as no formal studies were conducted due to time constraints. This phase also implemented the Look and Feel stage from the UI framework.

Secondly, the AR functionalities were tested (Figure 28), revealing two key issues:

1) Overlaps between General UI and BIM AR UI occurred across multiple panels
2) The login feature failed to communicate with the API as expected

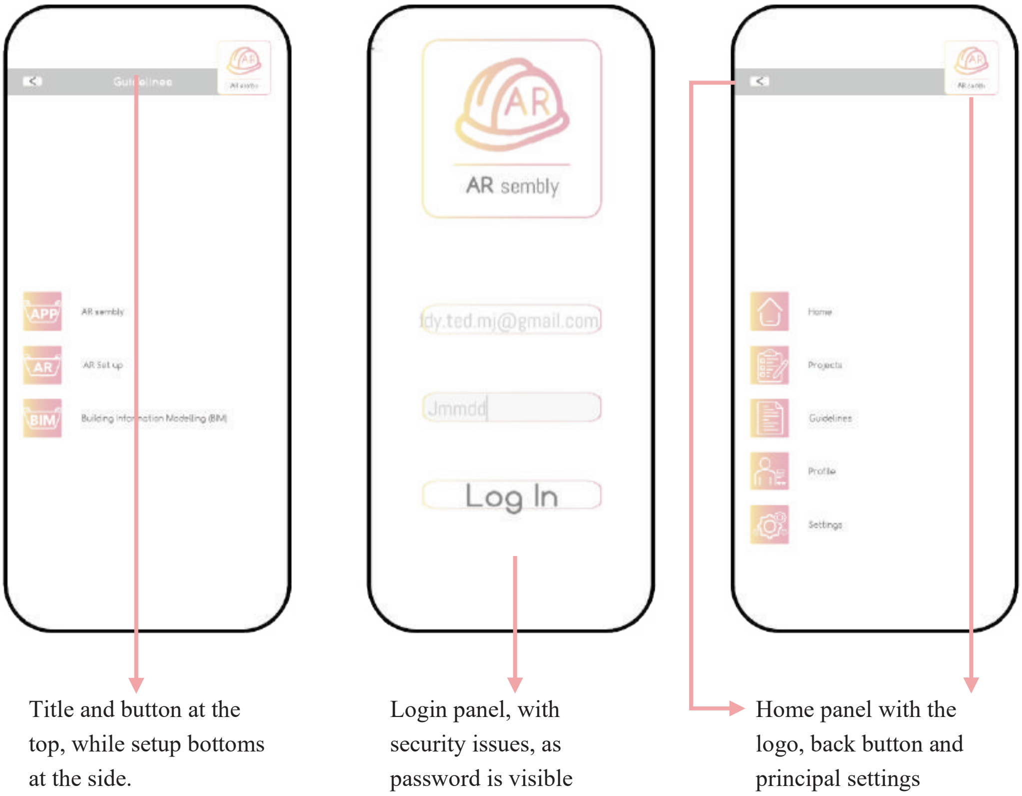

Figure 27 – First Virtual Prototype

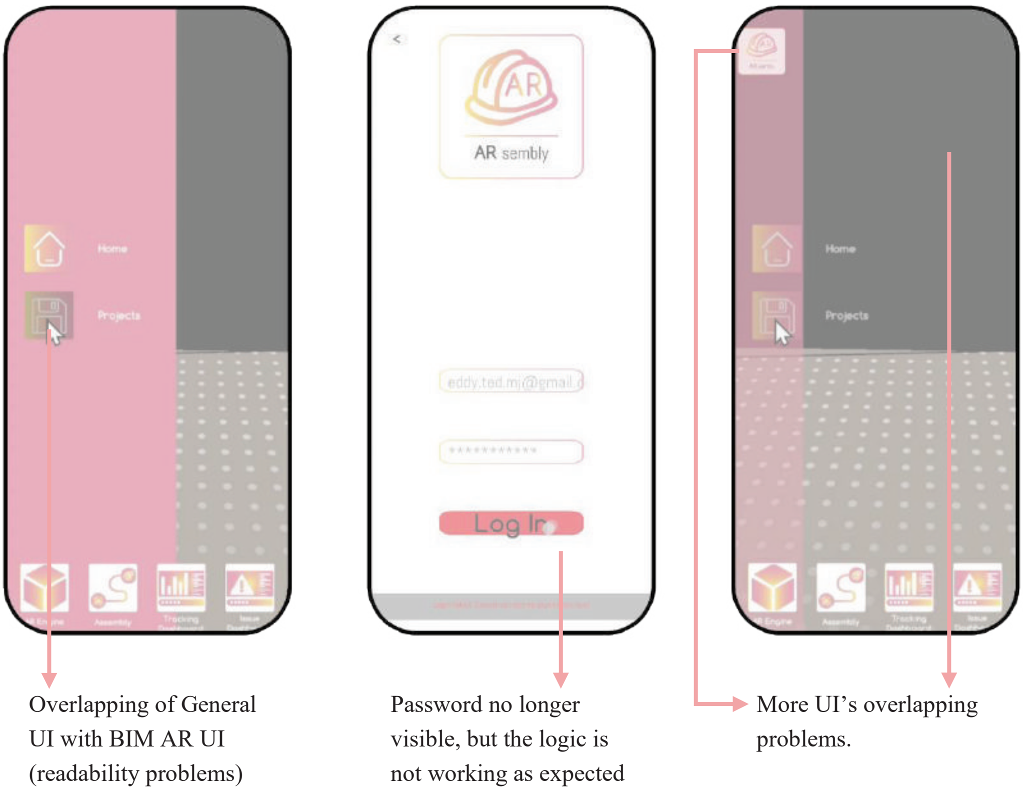

Figure 28 – Second Virtual Prototype

During the third iteration, the security flaw was corrected by hiding the password field. Additionally, new operations were added, including toggle buttons (checkboxes) to toggle model visibility, a home menu within the side bar, and animated transitions upon clicking buttons (Figure 29).

In the final prototype, the AR marker method for model placement was replaced with the AR plane approach, in response to feedback gathered during the wireframes, storyboards, and usability testing phases. Moreover, overlapping between the side menu and the AR plane was detected early during this iteration. The cause was identified and corrected immediately. As a result, the prototype displayed correct alignment of the AR plane, menus, buttons, and logical interaction flow (Figure 30).

Figure 29 – Third Virtual Prototype

Figure 30 – Fourth Virtual Prototype

The virtual prototyping stage proved to be a valuable and innovative step, effectively identifying critical design misalignments prior to the general testing phase defined in Hernández Castro's framework. The issues detected directly contributed to the refinement and seamless integration of General UI and BIM AR UI, ultimately improving both development and user experience.

5.2.7. Architecture β

As previously stated, both architecture α and architecture β were divided into two branches: the General UI and the BIM AR UI. These structures evolved based on results from the literature review, benchmarking analysis, and user profiling.

5.2.7.1. Architecture β (General UI)

The section is the updated version of the architecture α with the output's implementation from the wireframes and storyboards, usability test and virtual prototyping. Some of the main changes are as follows:

1) Login and signup functions now require acceptance of the Terms and Conditions (T&C)
2) A new panel was added prior to project selection to display setup guidelines

3) A new functionality allows projects to be added directly via API and linked to specified buttons, with projects still categorized as assembly or installation
4) A new project status called Archive was introduced for completed projects
5) A Save Settings function was added for project configurations

Additionally, active projects were limited to two categories (Assembly and Installation) and three status with three projects per status (Working in Progress, Pending, Complete) with no limit on Archived projects, as this list serves purely historical purposes (Figure 31).

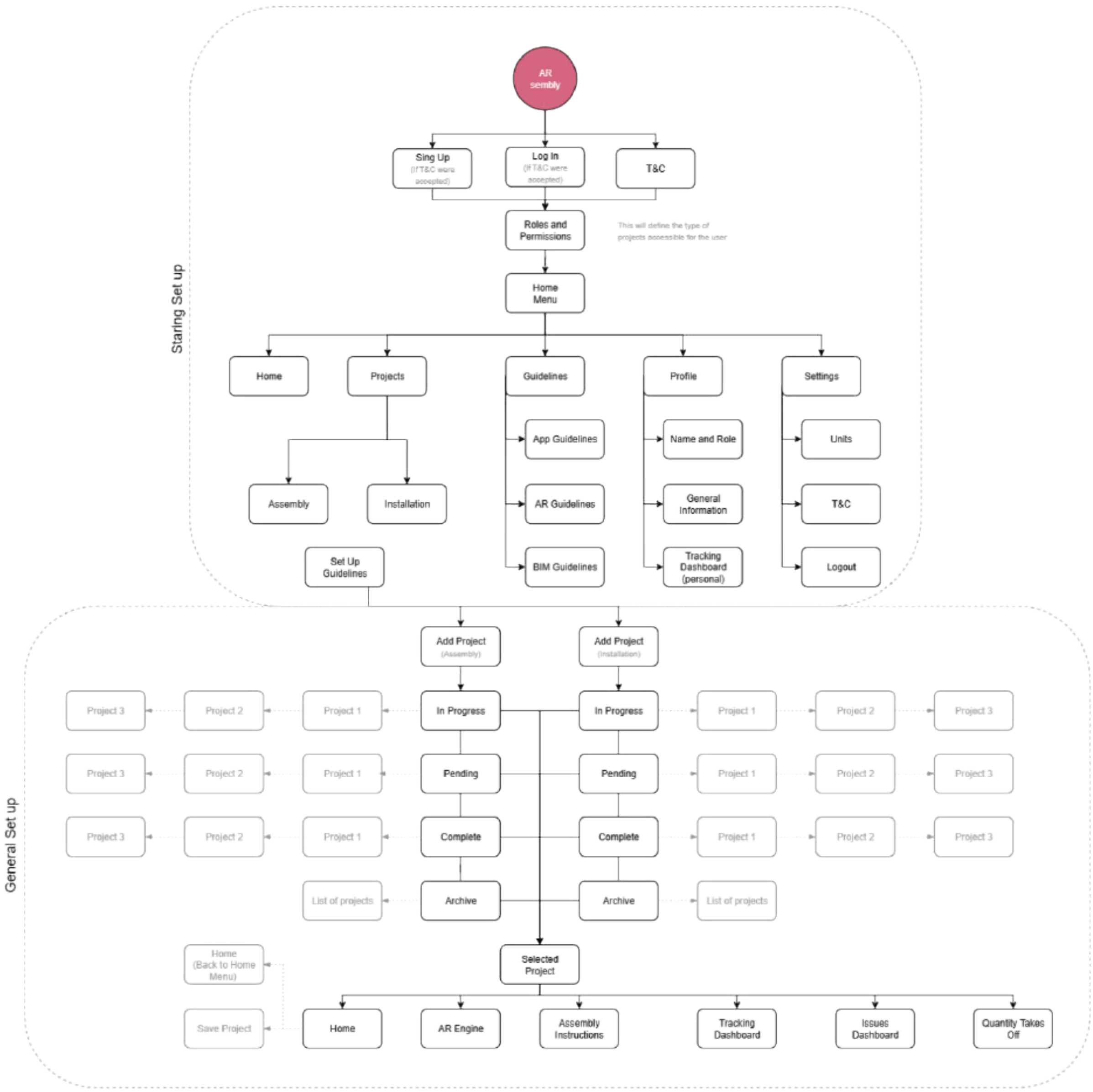

Figure 31 - ARsembly General UI Architecture β

5.2.7.2. Architecture β (BIM AR UI.)

Due to time and resource constraints, the reporting setup was not implemented in this iteration, allowing a more focused development of the starting, general, and core setups. This section presents the key improvements from architecture α to architecture β, especially regarding the AR engine (Figure 32):

1) Transition to AR placement method
2) Model manipulation features for rotation and scale adjustment
3) Subdivision of the BIM model into eight groups, aligned with the AR framework

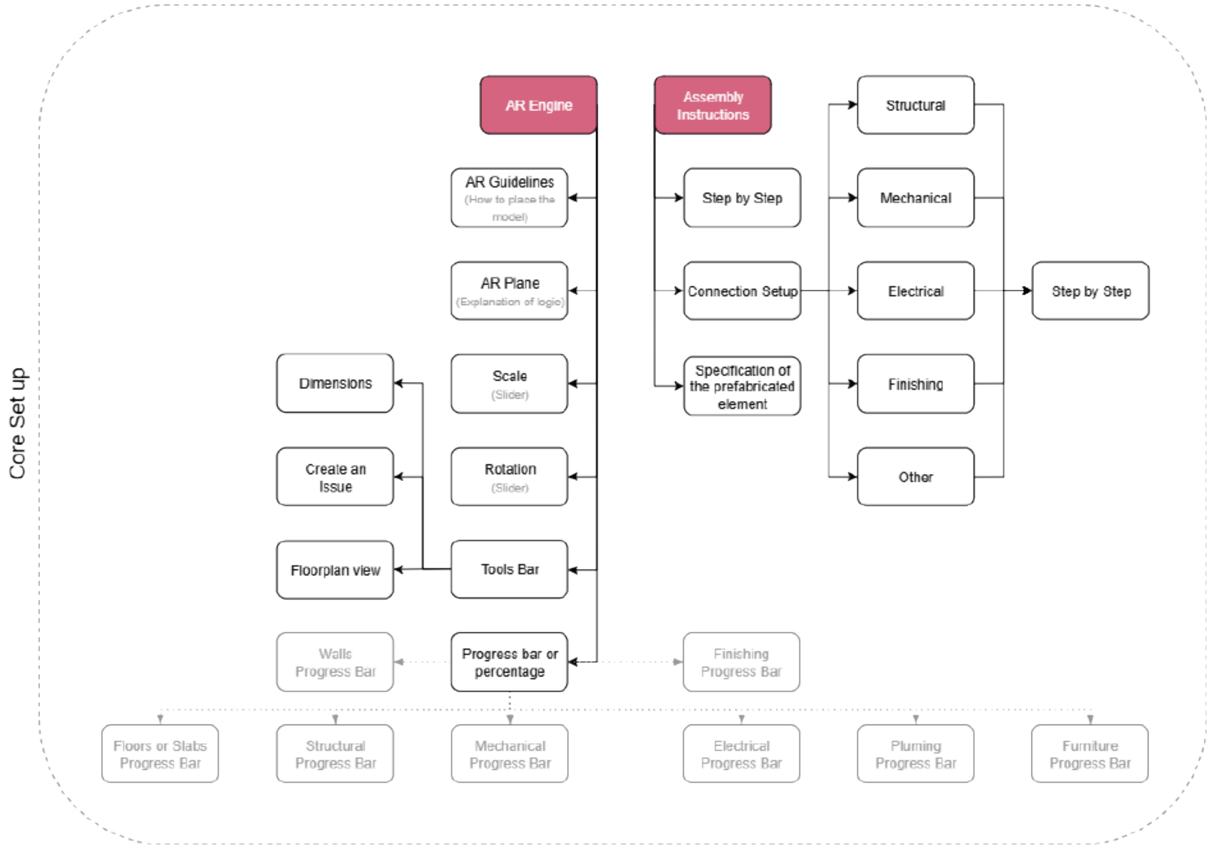

Figure 32 – ARsembly BIM AR UI Architecture β, Section 1

The reporting setup remains unchanged at this point due to the constraints mentioned (Figure 33 and Figure 34).

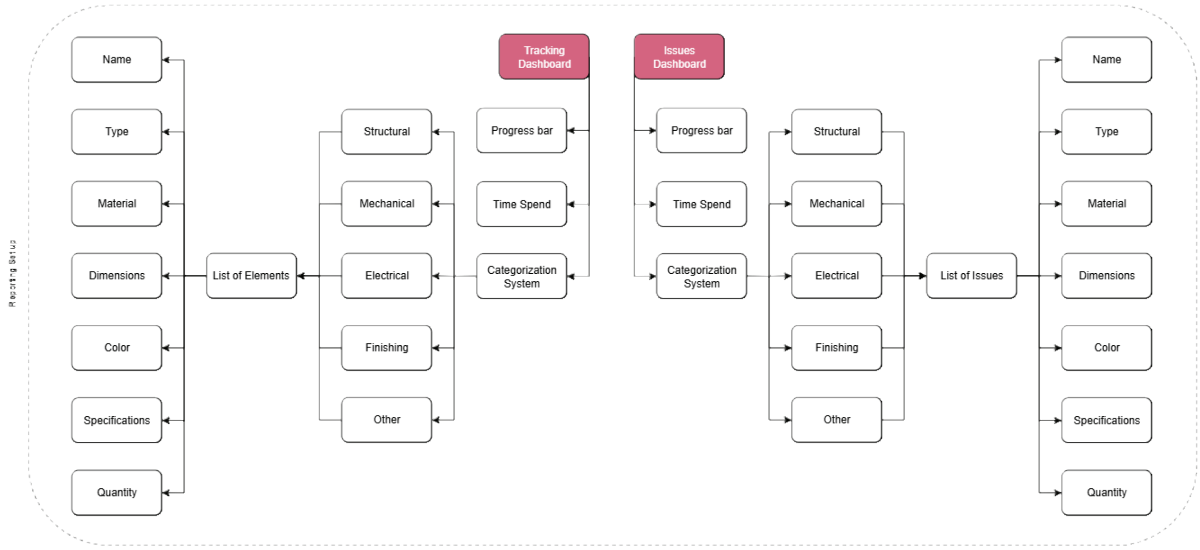

Figure 33 - ARsembly BIM AR UI Architecture β, Section 2

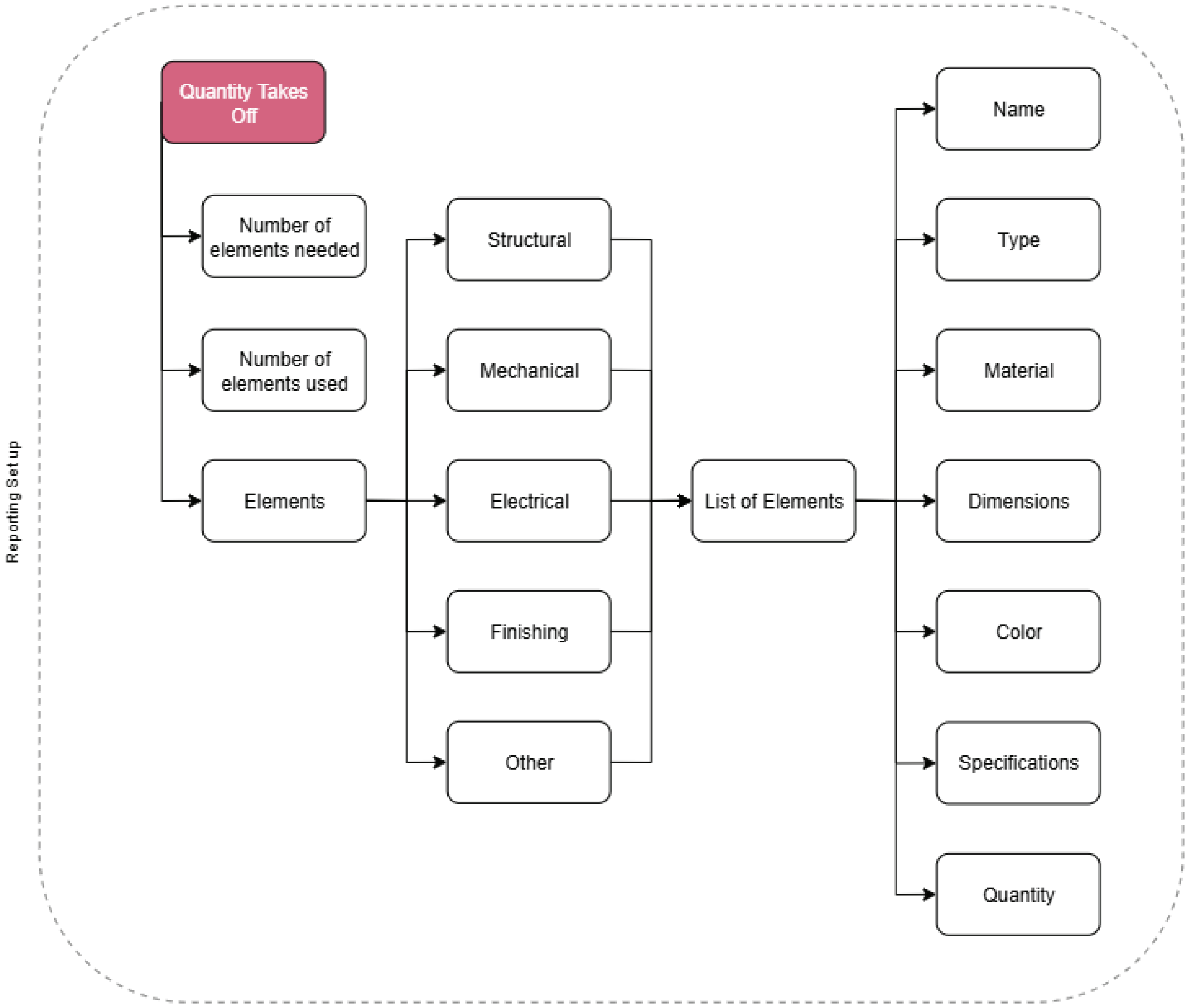

Figure 34 - ARsembly BIM AR UI Architecture β, Section 3

5.3. Implementation of proposed BIM AR Framework

The UI structure developed in the previous stages enabled a comprehensive understanding of the software's functionality, serving as a visual guide during the execution of the BIM AR framework. The implementation was targeted toward Android devices, using the GLTF file format, a location based (AR plane) display method, and AR Foundation for placement (Figure 35).

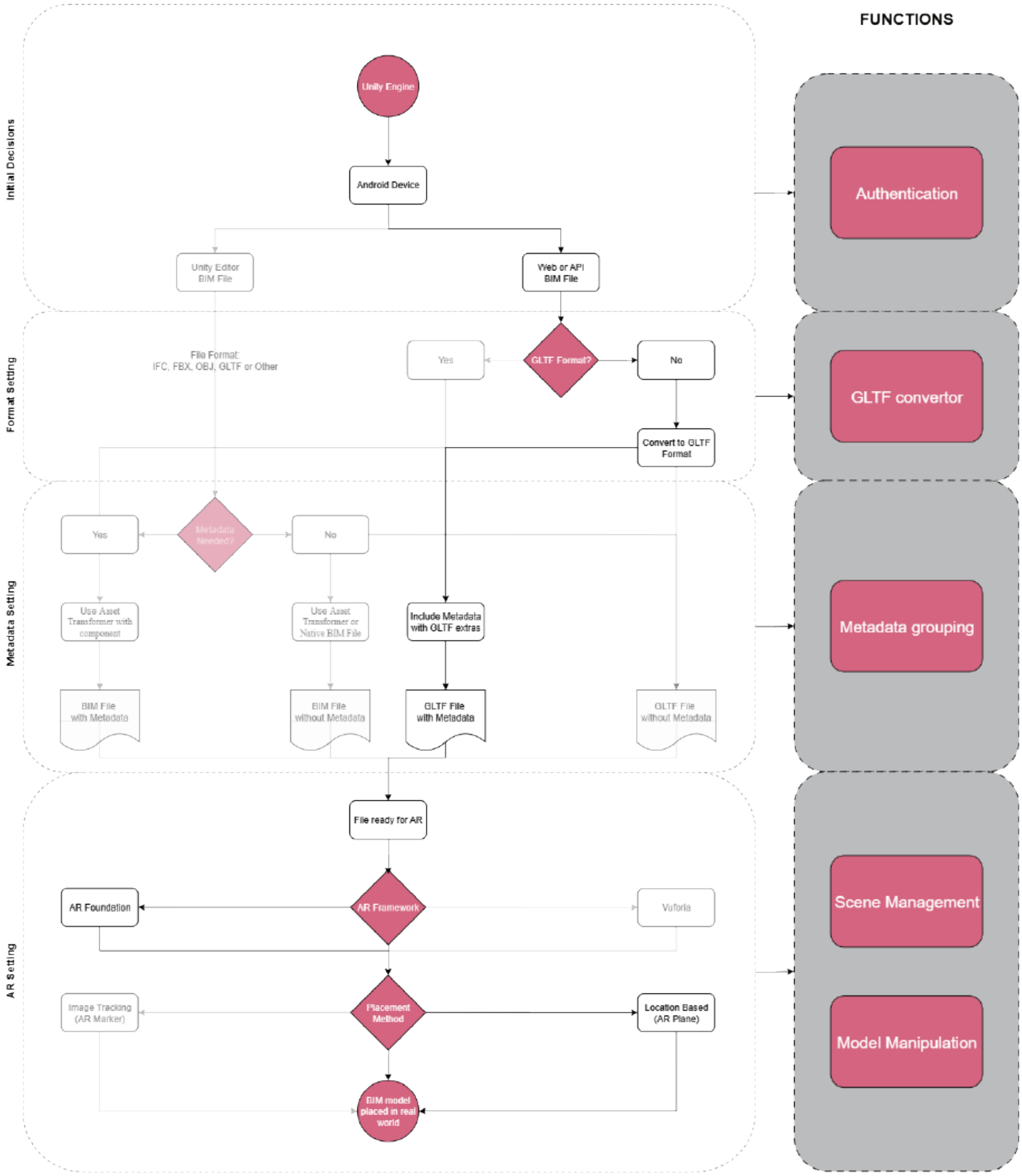

Figure 35 - Implemented BIM AR Framework for Unity Engine with Defined Fuctionalities

To execute each stage of the framework, various technical functions were developed using Unity engine tools, external scripts, and API connections. The correlation between framework stages and technical implementation is as follows (Figure 35):

1) Authentication function addressing the Initial decision stage.
2) Creation of a GLTF convertor for format setting
3) Addition of GLFT extras via coding for metadata setting
4) Unity scene management tools for AR framework
5) Model manipulation once the 3D is placed in the real world

The functions and general logic will be presented on the next sections.

5.3.1. Authentication and general logic

The BIM AR tool connects to an external API (Application Programming Interface), named "AR Factory", developed by José Luís Duarte Granja, which stores all user data, projects, and product information.

Authentication was managed via Docker containers that allow interaction between the Unity based BIM AR software and the external API. The authentication process comprises five steps (Figure 36):

1) Docker connection established between API and software
2) C# script stores the token and API endpoint (location)
3) The software sends a data request to the API
4) The API responds with the requested information
5) Software stores the data in Unity editor or mobile device

The Figure 37 illustrate a segment of the created script in C# programming language to manage the authentication process.

The complete software logic consists of sixteen steps (Figure 38 and 39):

- Steps 1 to 5: Authentication
- Steps 6 to 11: Role selection, user information display, and basic settings
- Steps 12 to 16: BIM model selection and loading

It’s important to note that BIM models can be loaded via two methods (Figure 39):

- API based (automatic via C#)
- Direct URL input (manual fallback)

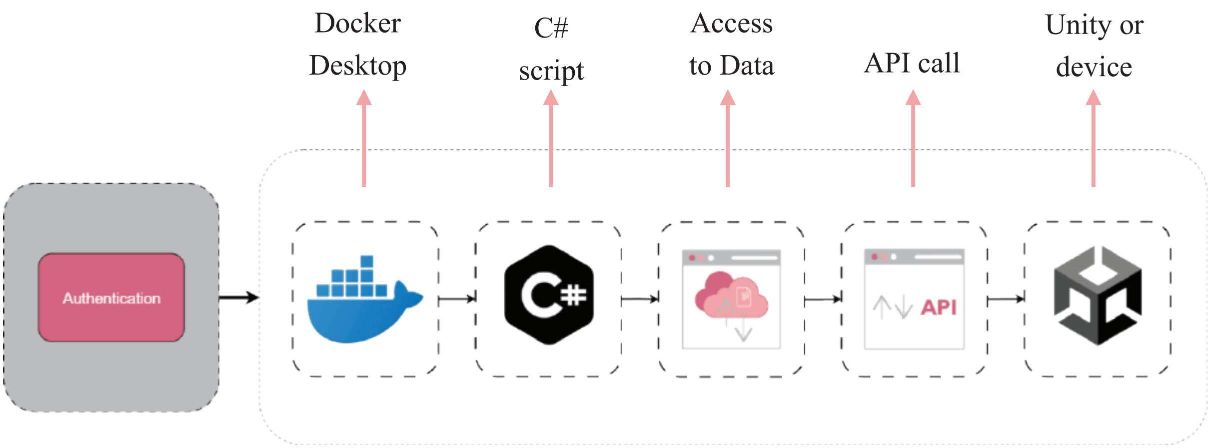

Figure 36 – Authentication Function Steps

Figure 37 – Section of C# Script for Manage Authentication Function

Figure 38 – ARsembly Final General User Interface 1

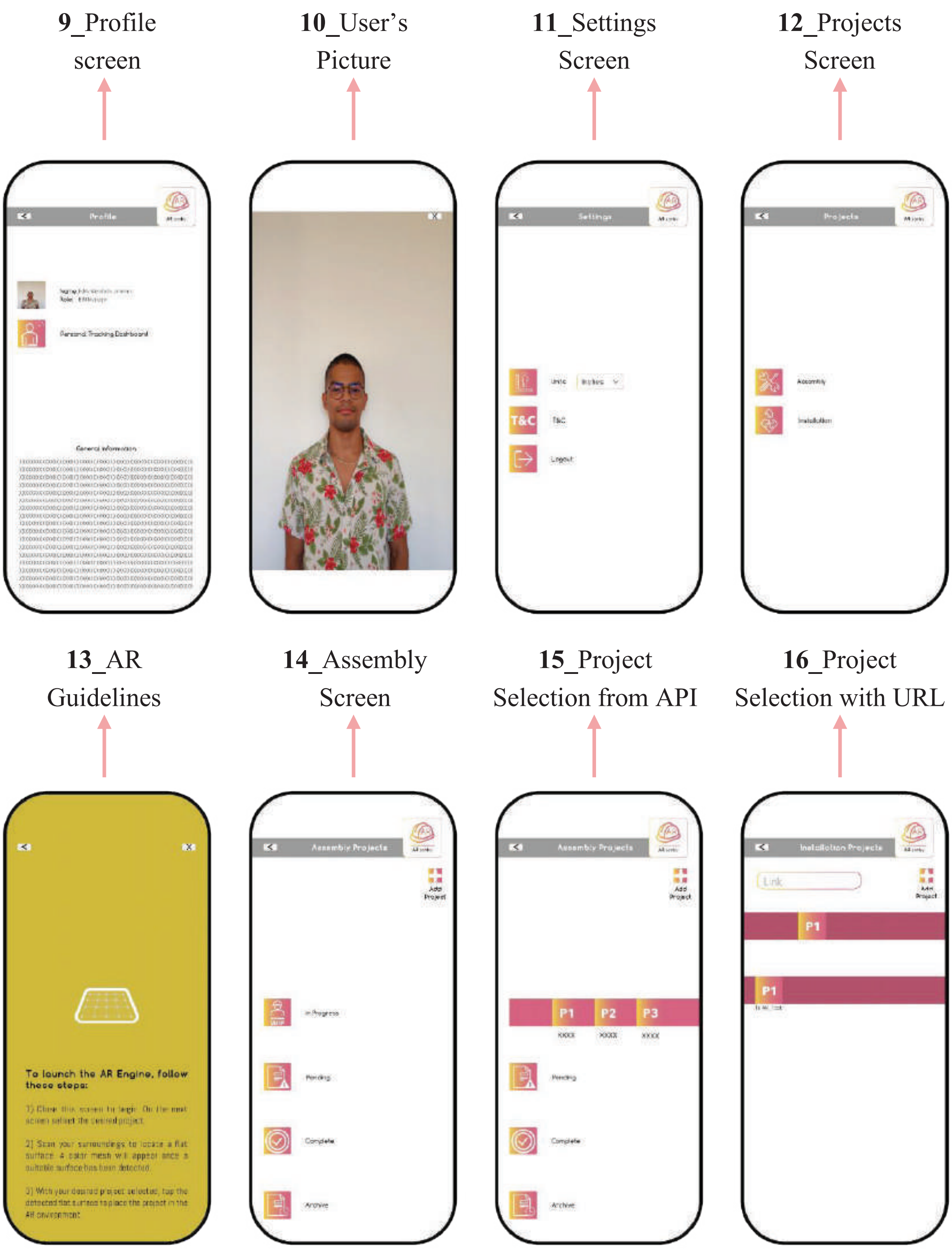

Figure 39 - ARsembly Final General User Interface 2

5.3.2. GLTF Convertor and Metadata grouping

Before loading the BIM model into the BIM AR software, its format must be converted from IFC or other BIM formats to GLTF, as FBX and IFC are not supported for runtime loading. This was achieved using Python scripting and the ifcOpenShell module (Figure 40).

Given that the tool supports assembly of prefabricated elements, it was necessary to organize BIM elements into eight categories, in line with the IFC schema provided by buildingSMART(Buildingsmart, 2020) (Figure 41):

1) Walls: "IfcWall", "IfcWallStandardCase"
2) Floors or slab: "IfcSlab", "IfcFloor"
3) Structural elements: "IfcBeam", "IfcColumn", "IfcMember", "IfcPlate", "IfcReinforcingBar
4) Mechanical elements: "IfcDistributionFlowElement", "IfcMechanicalFastener"
5) Electrical elements: "IfcElectricDistributionElement", "IfcElectricalElement", "IfcLamp"
6) Plumbing: "IfcFlowFitting", "IfcFlowSegment", "IfcSanitaryTerminal"
7) Furniture: "IfcFurniture"
8) Finishing elements: "IfcCovering", "IfcSurfaceStyle"

These groupings were added as GLTF "extras", ensuring both geometry and classification were included. Once converted, the GLTF files are uploaded to the API and retrieved at runtime using the Unity glTFast package and C# scripting (Figure 42).

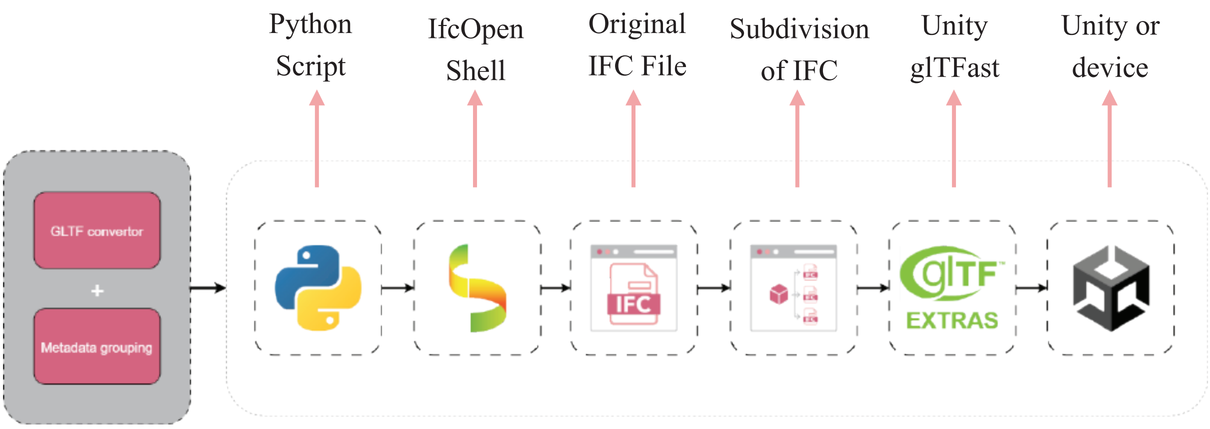

Figure 40 – GLTF Convertor and Metadata grouping Functions Steps

```python
import ifcopenshell
import ifcopenshell.geom
import trimesh
import os
import json
import numpy as np

# Define IFC types for each group
GROUP_DEFINITIONS = {
    "Walls": ["IfcWall", "IfcWallStandardCase"],
    "Floors or slab": ["IfcSlab", "IfcFloor"],
    "Structural elements": ["IfcBeam", "IfcColumn", "IfcMember", "IfcPlate", "IfcReinforcin
    "Mechanical elements": ["IfcDistributionFlowElement", "IfcMechanicalFastener"],
    "Electrical elements": ["IfcElectricDistributionElement", "IfcElectricalElement", "IfcL
    "Plumbing": ["IfcFlowFitting", "IfcFlowSegment", "IfcSanitaryTerminal"],
    "Furniture": ["IfcFurniture"],
    "Finishing elements": ["IfcCovering", "IfcSurfaceStyle"],
}

def get_group(product):
    for group_name, ifc_types in GROUP_DEFINITIONS.items():
        if any(product.is_a(t) for t in ifc_types):
            return group_name
    return None

def get_color_from_styles(ifc_product):
    try:
        styles = ifcopenshell.util.element.get_style(ifc_product)
        for style in styles:
            if hasattr(style, "Styles"):
                for s in style.Styles:
```

Figure 41 – Section of Python Script for GLTF Conversion

```csharp
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.Networking;
using TMPro;
using System;
using System.Collections;
using System.Collections.Generic;
using System.IO;
using UnityEngine.SceneManagement;

Unity Script | 0 references
public class ModelSelectionManager : MonoBehaviour
{
    // UI REFERENCES

    // BACKEND CONFIG
    [Header("API")]
    public string apiEndpoint = "http://localhost:8004/api/ifcmodel/";
    private const string AUTH_SCHEME = "JWT";

    // RUNTIME STATE
    private List<IFCModelInfo> allModels = new();
    private bool isLoading = false;
    private string authToken;
    private string activeMenu = "Assembly";
```

Figure 42 – Section of C# Script for GLTF file Loading

5.3.3. Scene Management and Model Manipulation

Following successful authentication, metadata inclusion, and file retrieval, the scene management system becomes active. The separation of UI logic (General UI vs BIM AR UI) translates into scene separation inside the Unity tool. AR functionality is only triggered after the model is selected.

Upon model selection, the tool opens the mobile camera to project the BIM model into the real world environment. Users then access the manipulation functions, which include (Figure 43).:

1) 360° rotation by a slider control.
2) Scaling adjustment by a slider control.
3) Assembly tracking, users toggle between completed/incomplete elements, using checkboxes and a visual progress slider (Figure 46)

All manipulation functions were coded in C#, using Unity UI elements (sliders, icons, toggles, labels, etc.) (Figure 44 and 45).

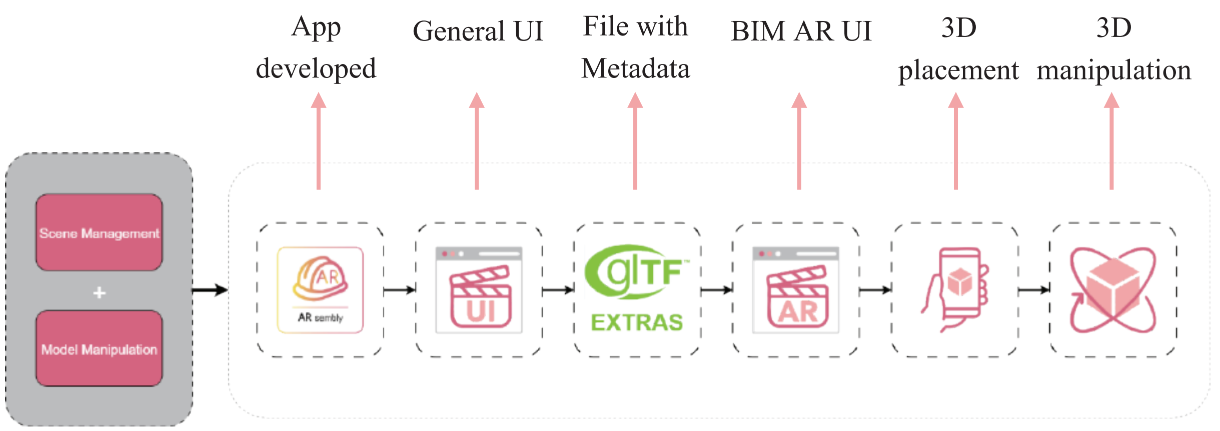

Figure 43 – Scene Management and Model Manipulation Functions Steps

```
    UnityEngine;
using UnityEngine.SceneManagement;
using UnityEngine.XR.ARFoundation;
using System.Collections;
using Unity.XR.CoreUtils;
using UnityEngine.XR.Management;

Unity Script | 0 references
public class SceneSwitcher : MonoBehaviour
{
    private XROrigin arOrigin;
    private ARSession arSession;
    private Camera arCamera;

    /// From Menu to AR
       0 references
       public void LoadAREngineSceneFromMenu()
    {
        PlayerPrefs.SetInt("ReturnFromMenu", 1);
        SceneManager.LoadScene("2_ArEngineScene");
    }

    /// From AR to Menu

    0 references
    public void LoadMenuSceneFromAR()
    {
        PlayerPrefs.SetInt("ReturnFromMenu", 0);

        // Find & disable only the AR visuals:
        arOrigin = FindAnyObjectByType<XROrigin>();
        arSession = FindAnyObjectByType<ARSession>();

        if (arOrigin != null) arOrigin.gameObject.SetActive(false);
        if (arSession != null) arSession.enabled = false;

        // Immediately go back to Menu:
        SceneManager.LoadScene("1_MenuScene");
```

Figure 44 - Section of C# Script for Scene Management Function

```
    UnityEngine;
 sing UnityEngine.UI;
using TMPro;

Unity Script | 3 references
public class ModelManipulator : MonoBehaviour
{
    private string modelId;
    private Camera arCamera;
    private Vector2 touchStartPos;
    private bool isMoving;

    [Header("UI References")]
    public Slider scaleSlider;
    public Slider rotationSlider;
    public TextMeshProUGUI scaleValueText;
    public TextMeshProUGUI rotationValueText;

    Unity Message | 0 references
    private void Start()
    {
        arCamera = Camera.main;

        if (scaleSlider != null) scaleSlider.wholeNumbers = true;
        if (rotationSlider != null) rotationSlider.wholeNumbers = true;
    }

    2 references
    public void Initialize(string id)
    {
        modelId = id;

        if (scaleSlider != null)
        {
            scaleSlider.onValueChanged.AddListener(SetScale);
            scaleSlider.value = Mathf.Round(transform.localScale.x * 100);
            UpdateScaleText(scaleSlider.value);
        }

        if (rotationSlider != null)
        {
            rotationSlider.onValueChanged.AddListener(SetRotation);
            rotationSlider.value = Mathf.Round(transform.eulerAngles.y);
            UpdateRotationText(rotationSlider.value);
        }
```

Figure 45 - Section of C# Script for Model Manipulation Function

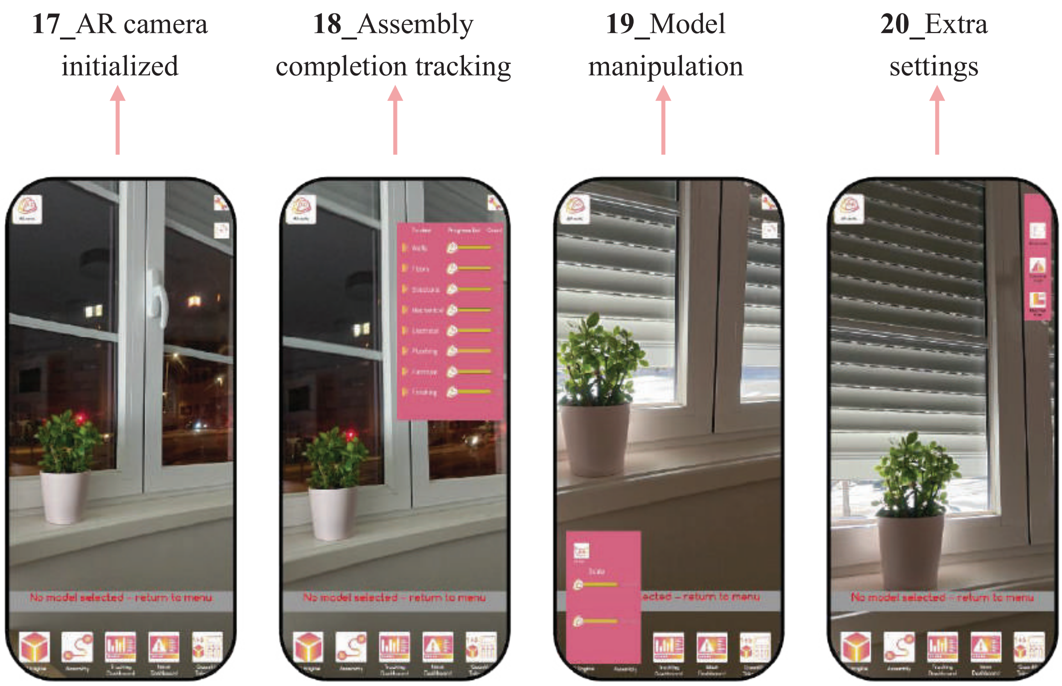

Figure 46 - ARsembly Final BIM AR User Interface 1

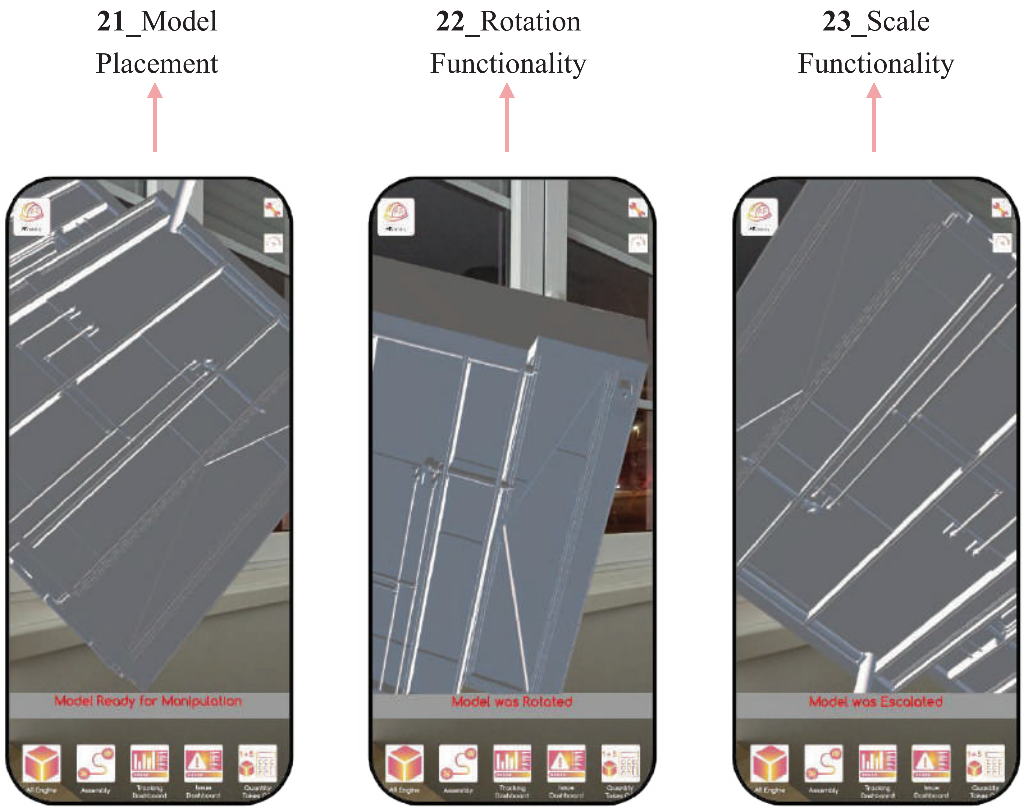

Figure 47 – ARsembly Final BIM AR User Interface 2

5.4. Final Considerations

This chapter presented the partial implementation of the proposed BIM AR tool, focusing on the validation of both the UI framework and the technical BIM AR integration. The virtual prototyping phase was essential in identifying and resolving early design flaws, such as UI overlaps, unstable AR marker use, and login issues, leading to an improved interface and better user experience.

The evolution from Architecture α to β showed clear progress in both logic and usability. Key enhancements included Terms and Conditions control, streamlined project handling, metadata support, and model manipulation features confirming the tool's alignment with prefabricated assembly workflows.

BIM AR framework implementation with Unity and Android successfully integrated runtime logic, API communication, and model loading using GLTF format. The eight IFC based model groups supported structured visualization and future task management.

A key insight from this chapter is that BIM AR UI design is not static or universal, but dynamic and context sensitive. As Chu (2023) highlights, task complexity directly influences UI structure, interaction logic, and feedback mechanisms. This was validated through the tool’s testing, where AR usability improved with refined design choices tailored to task complexity.

Overall, the chapter confirmed the framework’s viability and set the foundation for testing and evaluation in the next phase.

6. CONCLUSIONS AND LIMITATIONS

6.1. Final Conclusions

This dissertation sets out to explore and implement a replicable, technically grounded, and open source Building Information Modeling, Augmented Reality (BIM AR) solution, tailored to real world challenges in the Architecture, Engineering, Construction and Operations (AECO) sector. The result was a digital system architecture grounded in technical feasibility and informed by design logic, user segmentation, and iterative evaluation.

The research confirmed that while BIM and AR are increasingly relevant in AECO workflows, especially in on site assembly and decision making, their integration remains fragmented often limited by cost, software compatibility, or implementation complexity. This gap defined the challenge and validated the relevance of the research.

The proposed solution "**ARsembly**" emerged as a prototype that responds directly to this context. Across the thesis, two frameworks were proposed and developed: the BIM AR framework (Chapter 3), and the User Interface (UI) framework (Chapter 4). The AR framework tackled a series of technical challenges such as BIM model preparation, file format transformation, runtime rendering, and mobile device limitations. A key outcome was the creation and validation of a GLTF based runtime pipeline, selected for its efficiency, metadata management, and compatibility with free and open source tools, particularly important for ensuring accessibility and scalability. The use of GLTF allowed the system to bypass the limitations of more restrictive formats like FBX or IFC when used in mobile environments.

Simultaneously, the UI framework approached the complexity of human interaction in AR by defining a modular structure separating general functions from AR specific ones. This was iterated through Architecture α and β, wireframes, and virtual prototyping. The resulting structure supported clarity, usability, and visual hierarchy, especially for non expert users.

The process also relied on benchmarking, user segmentation, necessity mapping, and practical prototyping. These steps were not just methodological; they grounded the system in real multidisciplinary workflows and functional requirements found in real AECO scenarios. The implementation of Architecture α and the usability testing phase helped expose issues like AR marker instability, layout clarity, and the need for visual feedback, all of which directly influenced improvements later integrated into the prototype.

In Chapter 5, the full implementation of the BIM AR framework and the partial implementation of the UI framework demonstrated the practical feasibility of the proposed approach. The resulting tool "**ARsembly**" enabled user authentication via API, real time BIM model loading, GLTF metadata grouping, and in place model manipulation through AR. These functionalities were achieved using Unity, C# scripting, the glTFast package, Docker based API integration, and custom Python scripts.

A key insight throughout this process is that BIM AR UI design cannot be universal and it is not static. As noted by Chu (2023), the structure and effectiveness of an AR interface depend on the task's complexity, user type, and context. This reinforces the value of adaptive, task sensitive, and user driven design, especially in AECO contexts where technical accuracy, timing, and spatial constraints are critical. This principle held true during this research and reinforced the importance of modular, adaptive, and user centered design thinking in AR development for construction applications.

Therefore, the objective of developing a technically grounded, partially implemented, and context-sensitive BIM AR system has been successfully achieved. The frameworks are viable, the logic is validated through real testing, and the foundation is laid for full scale development and future research.

In conclusion, the thesis successfully fulfilled its aim: to develop and test a BIM AR integration system that is technically sound, aligned with AECO needs, and based on accessible, open source tools.

6.2. Research Limitations

Despite its structured methodology and practical outcomes, this research was bounded by several limitations:

1) Reporting and tracking features partial implementation
   The reporting and tracking setups originally designed to track project and assembly progress were only drafted and not implemented. Priority was given to the core AR pipeline, interface structure, and runtime logic due to time and resource constraints.

2) Restricted format support for runtime loading
   The research focused on free and accessible solutions. Therefore, proprietary options like Asset Transformer (formerly Pixyz) were excluded. This limited runtime loading to GLTF format, as FBX and IFC formats are not supported at runtime by Unity without premium plugins.

3) Testing and validation scope
   Usability tests were limited to internal sessions during wireframing and prototyping. Broader evaluations involving external users such as field technicians, assembly workers, or industry stakeholders could provide deeper insights into real world usability and functional gaps.

4) Platform constraint to android devices
   The system was optimized for Android using Google ARCore via Unity's AR Foundation. The lack of testing or deployment for iOS or WebAR platforms means the tool remains platform dependent, although the architecture allows future adaptation.

5) Lack of 4D integration or progress synchronization

Although the BIM data pipeline managed metadata and grouped elements according to IFC schema, the tool did not synchronize with 4D construction planning tools or scheduling software. This limited the scope to real time model visualization without lifecycle tracking.

6.3. Future Development and Recommendations

Building on the outcomes and limitations, several future development pathways are identified:

1) Implementation of reporting and tracking logic
   Full development of the reporting module would enable users to document assembly progress, track on site issues, and visualize task completion. Integration with BIM Execution Plans (BEPs) and CDEs could create a closed data loop.

2) Cross platform development
   Expanding support to iOS (via ARKit) and Web based AR could increase accessibility and use across diverse projects.

3) Expanded user testing
   Field level testing should be conducted to evaluate tool performance in real scenarios. This could include comparative tests with traditional documentation methods and feedback from multidisciplinary teams.

4) IFC and IDS validator integration
   Future versions could include native or plugin based support for IFC validation against Exchange Information Requirements (EIRs) using emerging IDS standards. This would improve consistency and enable smart filtering of model data for AR deployment.

5) Advanced interaction models
   Incorporating Natural User Interfaces (NUIs) such as gestures or voice recognition could further simplify interaction.

REFERENCES

Abboud, R. (2014) "Architecture in an Age of Augmented Reality: Opportunities and Obstacles for Mobile AR in Design, Construction, and Post-Completion.," in. Available at: https://api.semanticscholar.org/CorpusID:110038356.

A-Frame (2025) *VR Headsets & WebXR Browsers – A-Frame*. Available at: https://aframe.io/docs/1.7.0/introduction/vr-headsets-and-webxr-browsers.html (Accessed: August 31, 2025).

Alizadehsalehi, S., Hadavi, A. and Huang, J.C. (2020) "From BIM to extended reality in AEC industry," *Automation in Construction*, 116. Available at: https://doi.org/10.1016/j.autcon.2020.103254.

Autodesk (2025) *GAMMA AR | ACC & BIM360 | Autodesk App Store*. Available at: https://apps.autodesk.com/BIM360/en/Detail/HelpDoc?appId=5167916362864767928&appLang=en &os=Web (Accessed: August 31, 2025).

Ayer, S.K., Messner, J.I. and Anumba, C.J. (2016) "Augmented Reality Gaming in Sustainable Design Education," *Journal of Architectural Engineering*, 22(1), p. 4015012. Available at: https://doi.org/10.1061/(ASCE)AE.1943-5568.0000195.

Azhar, S., Hein, M. and Sketo, B. (2008) "Building Information Modeling (BIM) : Benefits , Risks and Challenges," in. Available at: https://api.semanticscholar.org/CorpusID:2418765.

Azuma, R.T. (1997) "A Survey of Augmented Reality," *Presence: Teleoperators and Virtual Environments*, 6(4), pp. 355–385. Available at: https://doi.org/10.1162/pres.1997.6.4.355.

Babylon.js (2023) *Proper and efficient way to working with large BIM models*. Available at: https://forum.babylonjs.com/t/proper-and-efficient-way-to-working-with-large-bim-models/39559 (Accessed: August 31, 2025).

Babylon.js (2025) *Babylon Native: cross-platform native apps with the power of Babylon.js*. Available at: https://www.babylonjs.com/native/ (Accessed: August 31, 2025).

Bademosi, F., Blinn, N. and A Issa, R.R. (2019) *Use of augmented reality technology to enhance comprehension of construction assemblies*, *Journal of Information Technology in Construction (ITcon)*. Available at: http://www.itcon.org/2019/4.

Balali, V., Zalavadia, A. and Heydarian, A. (2020) "Real-Time Interaction and Cost Estimating within Immersive Virtual Environments," *Journal of Construction Engineering and Management*, 146(2), p. 4019098. Available at: https://doi.org/10.1061/(ASCE)CO.1943-7862.0001752.

Boan, T., Jiajun, L. and Bosché, F. (2025) "AUTONOMOUS MIXED REALITY FRAMEWORK FOR REAL-TIME CONSTRUCTION INSPECTION," *Journal of Information Technology in Construction*, 30, pp. 852–874. Available at: https://doi.org/10.36680/j.itcon.2025.035.

Börsting, I., Karabulut, C., Fischer, B. and Gruhn, V. (2022) “Design Patterns for Mobile Augmented Reality User Interfaces—An Incremental Review†,” *Information (Switzerland)*, 13(4). Available at: https://doi.org/10.3390/info13040159.

Buildingsmart (2020) *Core data schemas*. Available at: https://standards.buildingsmart.org/IFC/RELEASE/IFC4/FINAL/HTML/ (Accessed: August 31, 2025).

Chelaru, B., Onuțu, C., Ungureanu, G., Volf, I. and Șerbănoiu, A.A. (2025) “BIM and cloud-based tools integration for BREEAM in energy-efficient building design: A case study,” *Results in Engineering*, 26. Available at: https://doi.org/10.1016/j.rineng.2025.105493.

Chi, H.-L., Chen, Y.-C., Kang, S.-C. and Hsieh, S.-H. (2012) “Development of user interface for tele-operated cranes,” *Advanced Engineering Informatics*, 26(3), pp. 641–652. Available at: https://doi.org/10.1016/j.aei.2012.05.001.

Coronado, E., Itadera, S. and Ramirez-Alpizar, I.G. (2023) “Integrating Virtual, Mixed, and Augmented Reality to Human–Robot Interaction Applications Using Game Engines: A Brief Review of Accessible Software Tools and Frameworks,” *Applied Sciences (Switzerland)*, 13(3). Available at: https://doi.org/10.3390/app13031292.

Cryengine (2025a) *CRYENGINE | Documentation - VR - Oculus Rift*. Available at: https://www.cryengine.com/docs/static/engines/cryengine-3/categories/1638401/pages/21267378 (Accessed: August 31, 2025).

Cryengine (2025b) *CRYENGINE | Support: General*. Available at: https://www.cryengine.com/support/view/general (Accessed: August 31, 2025).

Cunha, R.R., Arrabal, C.T., Dantas, M.M. and Bassanelli, H.R. (2022) “Laser scanner and drone photogrammetry: A statistical comparison between 3-dimensional models and its impacts on outdoor crime scene registration,” *Forensic Science International*, 330, p. 111100. Available at: https://doi.org/10.1016/j.forsciint.2021.111100.

Cupix (2025) *Spatial Intelligence Platform for Construction & Facility Management | Cupix*. Available at: https://www.cupix.com/ (Accessed: August 31, 2025).

Dalux (2025) *TwinBIM - Augmented Reality (AR) – Dalux HelpCenter*. Available at: https://dalux.zendesk.com/hc/en-us/articles/360015612953-TwinBIM-Augmented-Reality-AR#h_01HMXN47SY4G50XNV6C2KYK3TH (Accessed: August 31, 2025).

Darfdesign (2025) *Augmented Reality platform for Architecture and Interior Design - DARF DESIGN*. Available at: https://www.darfdesign.com/arki.html (Accessed: August 31, 2025).

Davila Delgado, J.M., Oyedele, L., Demian, P. and Beach, T. (2020) “A research agenda for augmented and virtual reality in architecture, engineering and construction,” *Advanced Engineering Informatics*, 45, p. 101122. Available at: https://doi.org/10.1016/j.aei.2020.101122.

Dickson, P.E., Block, J.E., Echevarria, G.N. and Keenan, K.C. (2017) "An experience-based comparison of unity and unreal for a stand-alone 3D game development course," in *Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE*. Association for Computing Machinery, pp. 70–75. Available at: https://doi.org/10.1145/3059009.3059013.

Dinis, F.M., Guimaraes, A.S., Carvalho, B.R. and Martins, J.P.P. (2017) "Development of virtual reality game-based interfaces for civil engineering education," in *IEEE Global Engineering Education Conference, EDUCON*. IEEE Computer Society, pp. 1195–1202. Available at: https://doi.org/10.1109/EDUCON.2017.7943000.

Emi (2024) *About Official Console Ports – Godot Engine*. Available at: https://godotengine.org/article/about-official-console-ports/ (Accessed: August 31, 2025).

Epicgames (2025a) *Augmented Reality Overview in Unreal Engine | Unreal Engine 5.6 Documentation | Epic Developer Community*. Available at: https://dev.epicgames.com/documentation/en-us/unreal-engine/augmented-reality-overview-in-unreal-engine (Accessed: August 31, 2025).

Epicgames (2025b) *Datasmith Plugins Overview | Unreal Engine 5.5 Documentation | Epic Developer Community*. Available at: https://dev.epicgames.com/documentation/en-us/unreal-engine/datasmith-plugins-overview?application_version=5.5 (Accessed: August 31, 2025).

Epicgames (2025c) *Importing IFC Files into Unreal Engine using Datasmith | Unreal Engine 5.5 Documentation | Epic Developer Community*. Available at: https://dev.epicgames.com/documentation/en-us/unreal-engine/importing-ifc-files-into-unreal-engine-using-datasmith?application_version=5.5 (Accessed: August 31, 2025).

Fan, X., Chai, Z., Deng, N. and Dong, X. (2020) "Adoption of augmented reality in online retailing and consumers' product attitude: A cognitive perspective," *Journal of Retailing and Consumer Services*, 53, p. 101986. Available at: https://doi.org/10.1016/j.jretconser.2019.101986.

Fernandes, G.A., Cunha, G.G., Mota, T. and Lopes, C. (2011) "The Advanced Visualization Station: a Mobile Computing Center for Engineering Project Support." Available at: https://www.msc-les.org/proceedings/mas/mas2011/mas2011_464.html (Accessed: August 31, 2025).

Funk, M., Bachler, A., Bachler, L., Kosch, T., Heidenreich, T. and Schmidt, A. (2017) "Working with augmented reality? A long-term analysis of in-situ instructions at the assembly workplace," in *ACM International Conference Proceeding Series*. Association for Computing Machinery, pp. 222–229. Available at: https://doi.org/10.1145/3056540.3056548.

GAMMA AR (2025) *Why you should go with GAMMA AR*. Available at: https://gamma-ar.com/why-gamma-ar/ (Accessed: August 31, 2025).

Gherghina, A., Olteanu, A. and Tapus, N. (2013) "A marker-based augmented reality system for mobile devices," in *2013 11th RoEduNet International Conference*. IEEE, pp. 1–6. Available at: https://doi.org/10.1109/RoEduNet.2013.6511731.

Godot Engine (2024) *Compiling for Linux, *BSD*. Available at: https://docs.godotengine.org/en/stable/contributing/development/compiling/compiling_for_linuxbsd.html (Accessed: August 31, 2025).

Godot Engine (2025) *Deploying to Android* . Available at: https://docs.godotengine.org/en/stable/tutorials/xr/deploying_to_android.html (Accessed: August 31, 2025).

Graham, K., Pybus, C., Arellano, N., Doherty, J., Chow, L., Fai, S. and Grunt, T. (2019) "Defining Geometry Levels for Optimizing BIM for VR: Insights from Traditional Architectural Media," *Technology|Architecture + Design*, 3(2), pp. 234–244. Available at: https://doi.org/10.1080/24751448.2019.1640541.

Grazina, J.F.L. (2013) "Realidade aumentada aplicada a BIM para a monitorização do progresso e controle de produção na construção." Available at: https://run.unl.pt/handle/10362/11005 (Accessed: August 31, 2025).

Haji Rasouli, A., Banihashemi, S., Drogemuller, R., Fazeli, A. and Mohandes, S.R. (2022) "Augmented reality in design and construction: thematic analysis and conceptual frameworks," *Construction Innovation*, 22. Available at: https://doi.org/10.1108/CI-01-2022-0007.

Hajirasouli, A. and Banihashemi, S. (2022) "Augmented reality in architecture and construction education: state of the field and opportunities," *International Journal of Educational Technology in Higher Education*. Springer Science and Business Media Deutschland GmbH. Available at: https://doi.org/10.1186/s41239-022-00343-9.

Hamledari, H. and Fischer, M. (2021) "Construction payment automation using blockchain-enabled smart contracts and robotic reality capture technologies," *Automation in Construction*, 132. Available at: https://doi.org/10.1016/j.autcon.2021.103926.

Hartson, Rex. and Pyla, P.S.. (2012) *The UX Book : Process and Guidelines for Ensuring a Quality User Experience*. Elsevier Science.

Hernández-Castro, F. (2016) "Usability cookbook: metodología para el análisis y diseño de aplicaciones," *RepositorioTEC* [Preprint].

Hertzum, M. (2020) "Usability Testing: A Practitioner's Guide to Evaluating the User Experience," *Synthesis Lectures on Human-Centered Informatics*, 1(1), pp. i–105. Available at: https://doi.org/10.2200/s00987ed1v01y202001hci045.

Jeffri, N.F.S. and Awang Rambli, D.R. (2021) "A review of augmented reality systems and their effects on mental workload and task performance," *Heliyon*. Elsevier Ltd. Available at: https://doi.org/10.1016/j.heliyon.2021.e06277.

Jenuwa, N., Yusop, N., Ali, S., Md Zain, N. and Zainal Abidin, Z. (2025) "Enhancing Construction Site Safety Through the Adoption of Safety Technologies," *BORNEO AKADEMIKA*, 9, pp. 82–94. Available at: https://doi.org/10.24191/BA/v9i1/118982.

Johansson, M. and Roupé, M. (2024) "Real-world applications of BIM and immersive VR in construction," *Automation in Construction*, 158. Available at: https://doi.org/10.1016/j.autcon.2023.105233.

Kim, S., Nussbaum, M.A. and Gabbard, J.L. (2019) "Influences of augmented reality head-worn display type and user interface design on performance and usability in simulated warehouse order picking," *Applied Ergonomics*, 74, pp. 186–193. Available at: https://doi.org/10.1016/j.apergo.2018.08.026.

Li, X., Yi, W., Chi, H.-L., Wang, X. and Chan, A.P.C. (2018) "A critical review of virtual and augmented reality (VR/AR) applications in construction safety," *Automation in Construction*, 86, pp. 150–162. Available at: https://doi.org/10.1016/j.autcon.2017.11.003.

Lindholm, J., Johansson, P. and Yitmen, I. (2025) "Toward a BIM information delivery framework based on common data environment," *Construction Innovation*, 25(7), pp. 139–157. Available at: https://doi.org/10.1108/CI-06-2024-0185.

Liu, B. and Chen, H. (2025) "BIMCoder: A Comprehensive Large Language Model Fusion Framework for Natural Language-Based BIM Information Retrieval," *Applied Sciences (Switzerland)*, 15(14). Available at: https://doi.org/10.3390/app15147647.

Machado, R.L. and Vilela, C. (2020) "Conceptual framework for integrating bim and augmented reality in construction management," *Journal of Civil Engineering and Management*, 26(1), pp. 83–94. Available at: https://doi.org/10.3846/jcem.2020.11803.

Mah, O.B.P., Yan, Y., Tan, J.S.Y., Tan, Y.X., Tay, G.Q.Y., Chiam, D.J., Wang, Y.C., Dean, K. and Feng, C.C. (2019) "Generating a virtual tour for the preservation of the (in)tangible cultural heritage of Tampines Chinese Temple in Singapore," *Journal of Cultural Heritage*, 39, pp. 202–211. Available at: https://doi.org/10.1016/j.culher.2019.04.004.

Makransky, G. and Klingenberg, S. (2022) "Virtual reality enhances safety training in the maritime industry: An organizational training experiment with a non-WEIRD sample," *Journal of Computer Assisted Learning*, 38(4), pp. 1127–1140. Available at: https://doi.org/10.1111/JCAL.12670.

Marín-Vega, H., Alor-Hernández, G., Bustos-López, M., López-Martínez, I. and Hernández-Chaparro, N.L. (2023) "Extended Reality (XR) Engines for Developing Gamified Apps and Serious Games: A Scoping Review," *Future Internet*. Multidisciplinary Digital Publishing Institute (MDPI). Available at: https://doi.org/10.3390/fi15120379.

MCAD (2025) *¡MCAD presenta GAMMA AR: La innovación en construcción!* Available at: https://mcad.co/mcad-presenta-gamma-ar-la-innovacion-en-construccion/ (Accessed: August 31, 2025).

Musarat, M.A., Khan, A.M., Alaloul, W.S., Ayub, S. and Bérenger, M. (2025) "Inspecting the benefits and challenges of virtual reality in construction projects of Malaysian industry," *Innovative Infrastructure Solutions*, 10(8). Available at: https://doi.org/10.1007/s41062-025-02153-0.

Norouzi, N., Shabak, M., Embi, M.R. and Khan, T. (2015) “The Architect, the Client and Effective Communication in Architectural Design Practice,” *Procedia - Social and Behavioral Sciences*, 172, pp. 635–642. Available at: https://doi.org/10.1016/j.sbspro.2015.01.413.

Olsen, D. and Kim, J. (2025) “Exploring the Use of Augmented Reality Paired with GPS for Construction Installation Activities,” in W. Collins, A.J. Perrenoud, and J. Posillico (eds.) *Proceedings of Associated Schools of Construction 61st Annual International Conference*. EasyChair (EPiC Series in Built Environment), pp. 492–499. Available at: https://doi.org/10.29007/w47v.

Pan, N.-H. and Isnaeni, N.N. (2024) “Integration of Augmented Reality and Building Information Modeling for Enhanced Construction Inspection—A Case Study,” *Buildings*, 14(3), p. 612. Available at: https://doi.org/10.3390/buildings14030612.

Panya, D.S., Kim, T. and Choo, S. (2023) “An interactive design change methodology using a BIM-based Virtual Reality and Augmented Reality,” *Journal of Building Engineering*, 68, p. 106030. Available at: https://doi.org/10.1016/j.jobe.2023.106030.

Peixoto Helena and FEUP (2018) *FEUP creates an educational laboratory for Civil Engineering*. Available at: https://noticias.up.pt/2018/01/05/feup-cria-um-laboratorio-educacional-para-a-engenharia-civil/ (Accessed: August 31, 2025).

Pereira da Silva, N., Eloy, S. and Resende, R. (2022) “Robotic construction analysis: simulation with virtual reality,” *Heliyon*, 8(10), p. e11039. Available at: https://doi.org/10.1016/j.heliyon.2022.e11039.

PlayCanvas (2025a) *Editor | PlayCanvas*. Available at: https://playcanvas.com/products/editor (Accessed: August 31, 2025).

PlayCanvas (2025b) *Platforms | PlayCanvas Developer Site*. Available at: https://developer.playcanvas.com/user-manual/xr/platforms/ (Accessed: August 31, 2025).

PlayCanvas (2025c) *Running the Engine in Node.js | PlayCanvas Developer Site*. Available at: https://developer.playcanvas.com/user-manual/engine/running-in-node/ (Accessed: August 31, 2025).

Puri, N. and Turkan, Y. (2020) “Bridge construction progress monitoring using lidar and 4D design models,” *Automation in Construction*, 109, p. 102961. Available at: https://doi.org/10.1016/j.autcon.2019.102961.

Rashidi, A., Maalim, A., Gutierrez-Bucheli, L., Maxwell, D. and Arashpour, A.Prof.M. (2022) “Evaluating the Effectiveness of BIM-based Virtual Reality for Coordinating the Design Disciplines in a Building Renovation Project,” in *IOP Conference Series: Earth and Environmental Science*. Available at: https://doi.org/10.1088/1755-1315/1101/3/032019.

Revolti, A., Dallasega, P., Schulze, F. and Walder, A. (2023) “Augmented Reality to support the maintenance of urban-line infrastructures: A case study,” *Procedia Computer Science*, 217, pp. 746–755. Available at: https://doi.org/10.1016/j.procs.2022.12.271.

Rokooei, S., Shojaei, A., Alvanchi, A., Azad, R. and Didehvar, N. (2023) “Virtual reality application for construction safety training,” *Safety Science*, 157, p. 105925. Available at: https://doi.org/10.1016/j.ssci.2022.105925.

Sabil, D. and Erizal (2023) “Penerapan Buidling Information Modeling (BIM) 5D pada Proyek Gedung Simpang Temu Dukuh Atas, Jakarta Pusat,” *Jurnal Teknik Sipil dan Lingkungan*, 8(02), pp. 95–104. Available at: https://doi.org/10.29244/jsil.8.02.95-104.

Sacks, R., Wang, Z., Ouyang, B., Utkucu, D. and Chen, S. (2022) “Toward artificially intelligent cloud-based building information modelling for collaborative multidisciplinary design,” *Advanced Engineering Informatics*, 53, p. 101711. Available at: https://doi.org/10.1016/j.aei.2022.101711.

Safira, A.D. and Hidayat, B. (2024) “Penerapan BIM Dan AR Menggunakan Autodesk Revit Dan Gamma AR Pada Pengawasan Pembangunan Struktur,” *Jurnal Bangunan, Konstruksi & Desain*, 2(1), pp. 28–39. Available at: https://doi.org/10.25077/jbkd.2.1.28-39.2024.

Sawhney, A., Riley, M. and Irizarry, J. (2020) *Construction 4.0: An Innovation Platform for the Built Environment*. Available at: https://doi.org/10.1201/9780429398100.

SmartTek (2025) *AResidence: AR House Tour*. Available at: https://smarttek.solutions/portfolio/ar-house-tour/ (Accessed: August 31, 2025).

Solmaz, S., Dominguez Alfaro, J.L., Santos, P., Van Puyvelde, P. and Van Gerven, T. (2021) “A practical development of engineering simulation-assisted educational AR environments,” *Education for Chemical Engineers*, 35, pp. 81–93. Available at: https://doi.org/10.1016/j.ece.2021.01.007.

Sommerville, Ian. (2017) *Software engineering*. 10th ed. Edinburgh: Pearson India Education Services.

Surri, D.A. and Kuncoro, T. (2025) “Integration of BIM and AR in Supporting Construction Project Management Performance: A Systematic Literature Review,” *G-Tech: Jurnal Teknologi Terapan*, 9(3), pp. 1289–1296. Available at: https://doi.org/10.70609/g-tech.v9i3.7218.

Trimble (2021) *Trimble Connect Product Updates - Trimble Inc.* Available at: https://community.trimble.com/viewdocument/trimble-connect-for-mobile-2132?CommunityKey=2ba352bf-190c-4712-8473-09e331f39bf2&tab=librarydocuments (Accessed: August 31, 2025).

Trimble (2025) *Trimble Connect - What is Trimble Connect?* Available at: https://docs.browser.connect.trimble.com/what-is-trimble-connect (Accessed: August 31, 2025).

Unity (2025a) *Unity - Manual: AR development in Unity*. Available at: https://docs.unity3d.com/6000.0/Documentation/Manual/AROverview.html (Accessed: August 31, 2025).

Unity (2025b) *Unity Asset Transformer: 3D CAD Data Visualization Software | Unity*. Available at: https://unity.com/products/unity-asset-transformer (Accessed: August 31, 2025).

Unity (2025c) *Unity glTFast Documentation | Unity glTFast | 6.13.1*. Available at: https://docs.unity3d.com/Packages/com.unity.cloud.gltfast@6.13/manual/index.html (Accessed: August 31, 2025).

Unity (no date) *What platforms are supported by Unity? – Unity Support Help Center*, *2025*. Available at: https://support.unity.com/hc/en-us/articles/206336795-What-platforms-are-supported-by-Unity (Accessed: August 31, 2025).

Unreal (2025) *Unreal Engine (UE5) licensing options - Unreal Engine*. Available at: https://www.unrealengine.com/en-US/license (Accessed: August 31, 2025).

Urbancová, H., Vrabcová, P. and Pacáková, Z. (2024) "Communication from below: Feedback from employees as a tool for their stabilisation," *Heliyon*, 10(7), p. e28287. Available at: https://doi.org/10.1016/j.heliyon.2024.e28287.

Van, B.P., Wong, P. and Abbasnejad, B. (2025) "A systematic review of criteria influencing the integration of BIM and Immersive Technology in building projects," *Journal of Information Technology in Construction*. International Council for Research and Innovation in Building and Construction, pp. 243–297. Available at: https://doi.org/10.36680/j.itcon.2025.011.

Verdelho Trindade, N., Ferreira, A., Madeiras Pereira, J. and Oliveira, S. (2023) "Extended reality in AEC," *Automation in Construction*. Elsevier B.V. Available at: https://doi.org/10.1016/j.autcon.2023.105018.

VSite (2025) *vSite AR | Construction-grade Augmented Reality*. Available at: https://www.vgis.io/augmented-reality-bim-gis-ar-aec-civil-construction-productivity-engineering-grade-mr-mixed-reality/ (Accessed: August 31, 2025).

Wang, X. (2009) "Augmented Reality in Architecture and Design: Potentials and Challenges for Application," *International Journal of Architectural Computing*, 7(2), pp. 309–326. Available at: https://doi.org/10.1260/147807709788921985.

W.F. Chen and J.Y. Richard Liew (2002) *The Civil Engineering Handbook*. Edited by W.F. Chen and J.Y.R. Liew. CRC Press. Available at: https://doi.org/10.1201/9781420041217.

Wonderland Engine (2025a) *Quick Start - Augmented Reality | Wonderland Engine*. Available at: https://wonderlandengine.com/getting-started/quick-start-ar/ (Accessed: August 31, 2025).

Wonderland Engine (2025b) *Wonderland Editor Now Available on Flathub | Wonderland Engine*. Available at: https://wonderlandengine.com/news/wonderland-editor-on-flathub/ (Accessed: August 31, 2025).

Wonderland Engine (2025c) *Wonderland Engine Features | Wonderland Engine*. Available at: https://wonderlandengine.com/about/features/ (Accessed: August 31, 2025).

Xu, Y., Barba, E., Radu, I., Gandy, M., Shemaka, R., Schrank, B., Macintyre, B. and Tseng, T. (2011) "Pre-patterns for designing embodied interactions in handheld augmented reality games," in *2011 IEEE*

International Symposium on Mixed and Augmented Reality - Arts, Media, and Humanities, ISMAR-AMH 2011, pp. 19–28. Available at: https://doi.org/10.1109/ISMAR-AMH.2011.6093652.

Xyzreality (2025) *Best augmented reality construction app | XYZ Reality*. Available at: https://www.xyzreality.com/augmented-reality-for-construction (Accessed: August 31, 2025).

Yang, D., de Vries, B. and van der Schaft, L. (2021) "The construction workers' preference and acceptance of innovations in data provision: A stated choice experiment study in the Netherlands," *Journal of Building Engineering*, 35, p. 101970. Available at: https://doi.org/10.1016/j.jobe.2020.101970.

Youngblut Kristina (2023) "Explore GAMMA AR's Augmented Reality for BIM Model Visualization and Collaboration," *AUGIWORLD, Management Strategies and Principles* [Preprint]. Available at: https://issuu.com/augi/docs/aw202308hr/32 (Accessed: August 31, 2025).

Yu, Y., Jeon, H. and Koo, B. (2022) "Enhancing the Stability and Placement Accuracy of BIM Model Projections for Augmented Reality-Based Site Management of Infrastructure Projects," *Applied Sciences*, 12(21), p. 10798. Available at: https://doi.org/10.3390/app122110798.

Zou, Z., Arruda, L. and Ergan, S. (2018) "Characteristics of models that impact transformation of BIMS to virtual environments to support facility management operations," *Journal of Civil Engineering and Management*, 24(6), pp. 481–498. Available at: https://doi.org/10.3846/jcem.2018.5689.