Zenith Outdoor/Indoor Experience

May 2025 – April 2027 (24M)

Project 2024-1-CY01-KA220-ADU-000256958

D3.1 Technical Infrastructure and Core

Authors	TUW
Abstract	The achievements accomplished during the first activity of work package 3 lay out the foundation for further implementation of the guided outdoor AR tour. They provide 1) a framework that defines requirements regarding the core functionality of the guided tour; 2) a proposal of software and hardware solutions capable of those functionalities; and 3) a core implementation which enables further implementation activities. The impact for the future stages of the project is described in detail. The development of guided outdoor AR tours will benefit from the proposes unified data structure. The proposed hardware, the Apple Vision Pro, can be a vehicle for novel human-environment interactions that will be designed within future activities of this work package. As such, the deliverable is aligned with the broader goals of the Erasmus+ programme and contributes directly to the promotion of lifelong learning, digital heritage, and innovation in adult education. Moreover, this document reflects the collaborative and interdisciplinary nature of the project, highlighting the contributions of each partner involved.

THE CONSORTIUM

No.	Abbreviation	Partner Name	Logo
1	**GEO**	Geoimaging ltd	
2	**AUTH**	Aristotle University of Thessaloniki	
3	**TUW**	Technische Universität Wien	
4	**VIR**	Virtual Solutions	

DISCLAIMER

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them.

Co-funded by the European Union

Co-funded by the European Union

CONTENTS

The consortium 2
Disclaimer 2
Contents 3
Glossary 3
1. Introduction 4
2. Framework 5
2.1 AR core functionality 5
2.2 Unified data structure 7
3. AR Implementation Decisions 8
3.1 Hardware 8
3.2 Software 8
3.2.1 Requirements for Unity Setup 9
4. AR Implementation Details 9
4.1 Mixed Reality Setup 10
4.2 UI skeleton 10
4.3 API call 11
5. Summary and Conclusions 12

GLOSSARY

API	Application Programming Interface
AR	Augmented Reality
DLL	Dynamic Link Library
GDPR	General Data Protection Regulation
HTTP	Hyptertext transfer protocol
POI	Point of Interest
UI	User Interface

Co-funded by the European Union

1. INTRODUCTION

This deliverable is part of work package 3 which bundles the outdoor AR implementation activities of the ZOE project – Zenith Outdoor/Indoor Experience for Cultural Tours with AR Technology and 3D Platform – funded under the Erasmus+ KA220-ADU programme. ZOE aims to enhance adult education and cultural engagement by leveraging cutting-edge technologies, such as Augmented Reality (AR), 3D simulation, and gamified digital learning paths. The outcomes of this activity form the basis for the outdoor guided AR tour. In this document, we aim to shed light on decisions from various angles, such as usability, accessibility, novelty and robustness.

The purpose of this deliverable is to present the context, rationale, and initial developments related to the specific tasks and objectives outlined in work package 3. It serves to document the progress made by the consortium partners and to provide a structured account of the methodologies employed, tools developed, and coordination mechanisms applied. The content herein reflects the collaborative efforts of the project consortium and contributes to the overall goal of promoting accessible, inclusive, and technology-driven lifelong learning experiences.

This document is structured as follows: Section 2 describes a framework that will guide the activities of this work package, enabling the design and validation of an outdoor guided AR tour. It describes the core functionality of the AR prototype and introduces the concept of a unified data structure for the tour's content and logic. Section 3 discusses the key decisions regarding the implementation of the AR prototype, including hardware and software concerns. A solution is presented and its ability to meet the requirements is discussed: The Apple Vision Pro is selected as an AR target device due to its robustness during sunlight exposure and its capability of using eye movements as user-controlled input. Section 4 describes an artefact that implements the proposed solution, enabling the implementations required for subsequent activities. This includes core functionalities for the general project setup, UI dialogues and retrieving data from servers or other hardware. Finally, Section 5 highlights how the outcomes of this activity facilitate further efforts within and beyond this work package.

Co-funded by the European Union

2. FRAMEWORK

The following describes the framework necessary for developing the interconnected system, including the outdoor AR and indoor components. This framework is based on the results of the requirements questionnaire (A2.1) and will guide the project's subsequent activities and implementations. In this section, we will focus on the implications for the AR module and subsequent actions within this work package.

It is important to note that the project partners have agreed that outdoor-AR and indoor-3D content are similar in one direction only. This means that all guided tour interactions and content available in the AR setting will also be available in the indoor 3D system. However, this is not the case in the opposite direction: features in the indoor 3D system may not be implemented in the outdoor AR system. This decision was made because adding features to the indoor system takes significantly less time than adding them to the immersive outdoor AR system. Section 2.2 describes how a unified data structure is used to achieve this requirement for similarity.

2.1 AR core functionality

The requirements for the outdoor AR guided tour were developed together with all project partners, partially based on the results of the requirements questionnaire presented by AUTH at the first transnational meeting (A2.1). It should be noted that these requirements outline the fundamental components and interactions of the guided AR tour. These will inform the development of specific guided tour use cases and their content and location in A3.2 of this work package. All requirements will have a significant impact on the further development steps for this project. While technical factors influenced the selection of requirements, this document does not describe each technical limitation in a bottom-up approach. Instead, we first describe the requirements in a top-down approach, followed by their implications for the technical setup and how these were addressed.

The following requirements were agreed:

1. **Guided tour content is structured in a layered way.**

Overloading the scene with content can lead to cognitive overload and overstimulate a user. Therefore, content will be presented in a layered format. At the highest level, points

of interest (POIs) are annotated with a virtual icon to indicate to users that information can be revealed at this location. This virtual annotation object is presented in proximity to the corresponding real-world POI using georeferenced visualisation. Then, in response to specific user actions, the content in the lower layer can be revealed. Users can reveal further information in two ways, which are described below.

2. **Different media (Text, Audio, Images) can be embedded in the guided tour.**

A key strength of the guided outdoor AR tour is that digital content can be accessed while experiencing the real cultural heritage POIs. The following media types were selected for in situ presentation:

- Text
- Audio
- Image

The media can be added to a specific POI and is then presented at the location of that POI.

3. **POIs can be selected by Gaze&Pinch to reveal information.**

Gaze&Pinch is an interaction pattern designed for selection tasks in immersive environments. It uses eye-tracking technology to detect which virtual object the user is looking at. Then, by briefly pressing the thumb and index finger together, the selected object is confirmed, triggering a selection event. In our case, a user can look at a POI object in the upper layer, perform the pinch gesture, and information from the lower layer will be revealed.

The Gaze&Pinch interaction pattern offers several advantages. Most importantly, it addresses the 'monkey-arm syndrome', which is the fatigue that results from pointing at objects in immersive environments over prolonged periods of time. Furthermore, it can be used by all users who have control over their eye movements, as the pinch gesture can be replaced by any other trackable body movement. For example, users having impairments related to their arms could confirm the selected target by pressing their lips together. Finally, our requirement analysis revealed that users prefer to reveal information by looking at objects.

4. **Information can be revealed by moving closer to a POI.**

Finally, a key finding of the requirements analysis was that users prefer discovering information in their surroundings than have all the information revealed to them at once. To this end, we plan to utilize proximity-aware interactions. The Euclidian distance between a user and a POI is calculated, and if it falls below a certain threshold, information can be revealed. For example, a virtual POI object could only appear if the user gets close enough to a certain feature of the environment. This has multiple potential benefits. Firstly, the guided tour designer could ensure that learners experience parts of the environment in an order that maximizes understanding. Secondly, it could be used for gamification purposes by hiding multiple POIs in an environment that are only visible in close proximity, thereby motivating the user to explore the entire environment.

The described requirements form the toolset of creating an interactive, gamified, content-rich guided tour AR experience and, consequently, guide the further technical preparations described below.

2.2 Unified data structure

To fulfill the project goal of providing similar cultural heritage experiences for in-situ and indoor AR users, the project partners developed a unified data structure to store the content and logic of guided tours, including the four core functionalities mentioned above. A unified data structure enables the following workflow:

1. Setting up a guided tour from a web browser, including the selection of POIs and the upload of media content as described above.
2. Retrieving the guided tour either from the indoor 3d system or from the outdoor guided AR system via a web-based API.

This approach has the advantage of using a data structure that specifies the experience logic and presented media, ensuring similarity between the two systems. However, there is still flexibility to fine-tune the look and feel of the experience according to the inevitable differences between an indoor 3D simulation and an immersive AR experience. Finally, the data structure is extensible, allowing for additional functionalities relevant to future applications beyond the scope of the project.

While the data structure will be described in larger detail in A3.2 and in our project partner VIR's work package (A4.1 and A4.5), we would like to mention the three integral for our AR implementation decisions:

1. POIs with geocoordinates and assigned media
2. Proximity values for each POI
3. Actions (e.g. hide/show a POI, make a POI selectable/unselectable)

3. AR Implementation Decisions

This section describes the key decisions regarding the selected hardware and software for the guided AR tour, considering the core functionalities and the guided tour contents and logic discussed above.

3.1 Hardware

First, the Apple Vision Pro can be used in direct sunlight because the virtual objects are always visible. This is a key advantage over optical-see-through HMDs, which have severe contrast issues in bright sunlight conditions Second, the Apple Vision Pro comes with built-in eye-tracking capabilities of sufficient quality to utilize Gaze&Pinch interaction. A third benefit is that virtual content can be presented at a very high quality due to the screens' high resolution.

3.2 Software

In order to create immersive experiences for the Apple Vision Pro, Unity 6 was chosen. Unity is an industry-standard game engine for creating immersive experiences. It comes with a visual editor that provides building blocks and frameworks for creating experiences. Additionally, C# scripts can be created to add functionality to these experiences. One key reason for choosing Unity is that Dynamic Link Libraries (DLL) can be incorporated into the system. These DLL files are used by the GeoAR framework, which we will use for the calibration module (A3.3). Thus, choosing Unity guaranteed that calibration would be possible. Finally, Apple and Unity are currently collaborating, which secures support for using Unity to develop Apple Vision Pro experiences.

3.2.1 Requirements for Unity Setup

The setting up of the Unity project requires the following steps which will be further extended as we see need while implementing the further deliverables of WP 3. The following steps are necessary to set up the Unity project so that future activities are optimized:

1. **Project Setup for Augmented Reality**

Unity is a game engine originally used for creating desktop applications. In order to create AR environments, certain software packages must be installed, and the project settings must be configured. Although Unity usually has large standard libraries for AR development, it is important to ensure that they are compatible with the the Apple Vision Pro as a target device. This step ensures that passthrough video of the real environment will be correctly presented while the space is merged with virtual objects and is particularly important due to the fact that Apple Vision Pro became only available in 2024.

2. **UI Skeleton**

Both the calibration user dialogue and potential user interactions during the guided AR tour require UI windows to function within the application. Thus, this is the first implementation target after project setup is complete. The Gaze&Pinch interaction pattern is achieved by setting up the UI skeleton, which involves the simple selection of UI elements.

3. **API call**

Furthermore, the application requires the ability to retrieve external information from a server using an HTTP request. This is necessary to complete the calibration dialogue and retrieve the content and logic of the guided tours. Therefore, we need to implement a consumer of multiple API endpoints.

4. AR Implementation Details

This section provides a more detailed description of the initial setup of the Unity-based prototype, which will be expanded to incorporate the project's AR component. It serves

as the main tangible outcome of Activity 3.1, which will be expanded upon in subsequent activities of this work package, ultimately resulting in the final prototype deliverable 3.6.

4.1 Mixed Reality Setup

To build for the Apple Vision Pro from Unity, version 6 or newer is required. This version can only be accessed with a Pro license of Unity, which is available for free through the Unity Educator plan for graduate students and university faculty.

There are two project building pipelines for the Apple Vision Pro: Unity Polyspatial or Metal. The former brings visionOS features to the Unity experience, while the latter gives developers full control over interactions, lighting, etc. Since this project does not require full control, as would be necessary for psychological studies, we opted for the Polyspatial package. This means that, regardless of our implementation, some user safety and comfort features will be in place. For instance, Polyspatial will notify users if they step too close to a physical object. Additionally, virtual objects in direct proximity to the user will become transparent. The following project setup was created to enable both video passthrough and correct gaze&pinch interactions:

- SceneCamera prefab from the PolySpatial samples
- XRInteractionManager
- AR Session
- Input Action Manager with the PolySpatialInputActions selected
- Spatial Tap Select Filter
- Two XR Spatial Pointer Interactors with Primary- and SecondaryWorldTouch Action
- A Polyspatial UnboundedVolumeCamera to draw objects in World Space

This setup fulfills the desired functionality at this point but might be adjusted in the future. A detailed description will follow in the final prototype deliverable (A3.6).

4.2 UI skeleton

A user interface (UI) is necessary for the spatial calibration required for georeferenced positioning of virtual objects (see deliverable 3.3). UI dialogues may also be used in the guided tour. Thus, we implemented a basic UI skeleton that includes elements such as

buttons and input fields, as well as different screens that can be navigated.. The following logic was implemented:

- View Manager, holding references to all main screens and keeping track of the currently visible screen
- UI content GameObject, holding:
    - UI canvas, holding the visual design of all screens
    - EventSystem, defining the allowed UI actions
- View Script – an abstract class that is implemented by each UI screen

This logic can be flexibly extended by creating a specific GameObject for the visual design and placement of the necessary UI elements. A script implementing the View class is attached to this object, comprising all functionalities relevant to that screen. As UI input actions are handled independently of the UI itself, the gaze-and-pinch interaction pattern works for the UI dialogue due to the project setup explained above.

4.3 API call

At least two parts of the guided AR tour prototype require an API request: the spatial calibration dialogue, which retrieves the user's location from another device; and the loading of the guided tour logic and content from the shared database. Therefore, a simple API request has already been implemented within this activity. Using the System.Net.Http library, the following logic was implemented:

- Set a base API string (the url)
- Create an HttpClient object
- Call GetAsync(url) on the obect and store it in a HttpResponseMessage
- Call Content.ReadAsStringAsync() on the HttpResponseMessage

The last two function calls adhere to an asynchronous execution pattern, typically used for operations that cannot be completed within a single frame. Without this, the application would block and become unresponsive while waiting for the device to fetch all required data—an issue that is particularly critical in an AR environment.This logic is very flexible and can, basically, be used to handle any GET requests by very simple adaptations. To handle the content, the retrieved response message must correspond to the underlying data structure. This is best done after the data structure details are agreed upon.

5. SUMMARY AND CONCLUSIONS

This deliverable has provided an overview of the key activities, methodologies, and intermediate results related to the implementation of the outdoor AR tour environment within the ZOE project. The documented work reflects the commitment of the consortium to ensure the quality, relevance, and impact of the project's outputs in the fields of digital heritage, adult education, and immersive learning technologies.

Through collaborative planning and task execution, all necessary decisions have been made to ensure the development of innovative tools and learning experiences, including augmented reality (AR) applications, 3D simulations, and inclusive navigation mechanisms. This deliverable, thereby, lays a solid foundation for the successful continuation of the project.

During the described activity, all project partners collaboratively made decisions in a way that further activities are enabled as best as possible. This resulted in a framework that includes:

- Clear **requirements for the AR guided** tour that can be defined within an unified data structure, ensuring appropriate alignment between the indoor and outdoor prototypes;
- a **selection of hardware and software**, capable of implementing the required features of the guided AR tour as defined as part of work package 2;
- an **implementation of the core features** that enables further activities of this work package.

The achievements of this deliverable can be used in all of the further parts of work package 3 as discussed below:

A3.2 Guided AR tour: The AR prototype can extend the mixed reality project structure implemented during this activity. User dialogues of any kind can extend the UI skeleton implementation. Guided tour logic and media can be loaded from the database by extending the API implementation.

A3.3 Calibration of [Apple Vision Pro]: The user dialogue for calibration can be implemented by extending the UI skeleton implementation. The API implementation can be adapted to retrieve the user location from another device.

A3.4 Guide Recording Module: User interactions to start/stop recordings can be handled using the input action structure implemented in the mixed reality project structure.

A3.5 User Interaction Module: User interactions based on the Gaze&Pinch and proximity-aware interaction patterns defined in the requirements. Basic user interactions, tracked and handled by the implemented project setup, can be refined and extended.

A3.6 & A3.8 Technical Evaluation: The selected software, Unity, allows the logging of all functionalities of the AR guided tour, as well as the logging of user interactions.

The results outlined in this deliverable will directly inform the upcoming phases and contribute to the broader objectives of ZOE—namely, to democratize access to culture, promote lifelong learning, and integrate digital innovation into adult education. Further updates and refinements will be captured in future deliverables and evaluated through internal monitoring and stakeholder feedback.