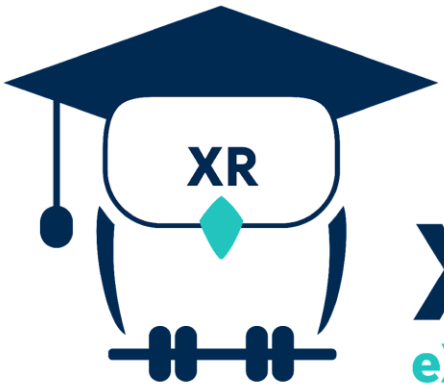
Ref. Ares(2023)8897770 - 29/12/2023

Horizon Innovation Action
HORIZON-CL4-2022-HUMAN-01-19



XR4ED
eXtended Reality for EDucation

D3.3
XR4ED Architecture and XR Modules specification

DISCLAIMER
This document is issued within the frame and for the purpose of the XR4ED project. This project has received funding from the European Union's Horizon Europe Framework Programme under Grant Agreement No. 1010931159. The opinions expressed and arguments employed herein do not necessarily reflect the official views of the European Commission. The dissemination of this document reflects only the author's view and the European Commission is not responsible for any use that may be made of the information it contains. This deliverable is subject to final acceptance by the European Commission. This document and its content are the property of the XR4ED consortium. The content of all or parts of this document can be used and distributed provided that the XR4ED project and the document are properly referenced. Each XR4ED partner may use this document in conformity with the XR4ED Consortium Grant Agreement provisions.

Funded by
the European Union

XR4ED Project Profile

Project title	eXtended Reality for EDucation
Project acronym	XR4ED
Grant agreement No.	101093159
Start date of project	1 January 2023
Duration of project	36 months
Project website	http://www.XR4ED-h2020.eu

Funded by the European Union

Partners

CYENS CENTRE OF EXCELLENCE	CYENS	RESEARCH	CYPRUS
konnektable TECHNOLOGIES	KONNEKTABLE	SME	IRELAND
21 Sci science 27 Co communication	SciCo	SME	GREECE
F6S	F6S	SME	IRELAND
INFINITIVITY DESIGN LABS	Infinity Design Labs	SME	FRANCE
KAINOTOMIA	KAINOTOMIA	SME	GREECE
NUROGAMES	NUROGAMES	SME	GERMANY
CRI	CRI	INDUSTRY	Luxemburg
SIMAVI	Software Imagination & Vision SRL	SME	ROMANIA
UPC	Universitat Politecnica de Catalunya	RESEARCH	SPAIN
CELLOCK	CELLOCK	SME	CYPRUS

Document information

Related work package	WP3	
Document reference	D3.3	
Related deliverables	D3.1, D3.2	
Type		DEM - Demonstrator, Pilot, Prototype
		DMP - Data Management Plan
	✓	R - Document, Report
Status	Final	
Version	4	
Due date	31/12/2023	
Submission date	29/12/2023	
Lead partner	(NURO)	
Contributing partners	(CELLOCK), (CYENS)	
Reviewers	CYENS	
Keywords		

Dissemination level	
✓	PU: Public
	SEN: Sensitive

List of contributors	
Name	Partner (short name)
Yash Shekhawat	NURO
Holger Kuerten	NURO
Haris Zacharatos	CELLOCK
Fotis Liarokapis	CYENS

Revision History

Document History			
Version	Date	Change editors	Changes
1	30/11/2023	Yash Shekhawat (NURO)	Initial version of the document
2	15/12/2023	Holger Kuerten (NURO), Haris Zacharatos (CELLOCK)	Inputs in various sections
3	28/12/2023	Yash Shekhawat (NURO)	Final version for review
4	29/12/2023	Fotis Liarokapis (CYENS)	Final version

Quality control		
Role	Partner (short name)	Approval date
Deliverable leader	Nurogames	28/12/2023
Quality manager	CYENS	29/12/2023
Project coordinator	CYENS	29/12/2023

Table of contents

XR4ED Project Profile 2
Partners 3
Document information 4
Revision History 5
List of figures 8
List of acronyms 9
Executive summary 10
1 Introduction 11
1.1 Overview 11
1.2 Structure of the Document 11
2 XR4ED Main Components 12
2.1 XR Module 12
2.2 Marketplace 12
2.3 Avatars and Templates 13
2.3.1 Avatars 13
2.3.2 Templates 13
2.4 Unique aspects of XR4ED Platform 14
2.4.1 Unique Functionalities 14
2.4.2 Unique Technical aspects 15
3 XR4ED Architecture 16
3.1 Architecture design principles 16
3.1.1 Microservices-based architecture 16
3.2 Architecture design process 21
3.3 Current Architecture 22
3.4 Communication between components 23
4 XR Module 25
4.1 XR Module Architecture 25
4.2 Functional Specifications 27

4.2.1 Module wide functional specifications 29
4.3 Technical Specifications 31
4.4 Current XR Module Specifications 33
5 Conclusion 35

List of figures

Figure 1 XR4ED Initial Microservices 19
Figure 2 XR4ED Components 23
Figure 3 XR Module components 25
Figure 4 Portal Hopper - 3rd person view 28
Figure 5 Portal Hopper - VR Mode 29
Figure 6 World Builder login screen 34
Figure 7 World Builder - New project screen 34

List of acronyms

Acronym	Description
XR4ED	Extended Reality for Education
XR	Extended Reality
VR	Virtual Reality
API	Application Interface
DRM/CRM	Digital/Content Rights Management
GIS	Geographic Information System
SDK	Software Development Kit
UI/UX	User Interface/ User Experience
FPS	Frames per Second
FAQs	Frequently asked questions
CI/CD	Continuous integration/continues deployment

Executive summary

The specific deliverable entitled “XR4ED architecture and XR Module specification” gathers information of the architecture of the XR4ED platform and the specifications of XR Module. The deliverable elaborates on the design principles and methodology for the architecture of the platform.

The XR Module, a central component of the XR4ED platform, is designed for creating, presenting, and experiencing content in a 3D environment, applicable both on-screen and in virtual reality (VR). The module's core functionalities include the World Builder, Portal Hopper, Interaction Editor, Test Environment, Templates, and Plug-ins Support, along with Server Integration. These features collectively facilitate the creation and interaction within immersive 3D worlds, significantly enhancing the educational and interactive experience.

The Marketplace, another integral part of XR4ED, serves as a centralized location for accessing, purchasing, and downloading digital XR assets. It manages products, facilitates content distribution, and handles shopping and transactions while ensuring digital rights management. Moreover, Avatars and Templates in XR4ED significantly contribute to user experience and interactivity, offering personalization, variety, and flexibility.

1 Introduction

1.1 Overview

The XR4ED D3.3 document offers an insightful overview of the XR4ED Architecture and the XR Module, essential components of the XR4ED platform. This deliverable outlines their combined role in enhancing Extended Reality (XR) experiences, with a focus on educational and interactive applications.

The XR4ED Architecture, built on a 3-tier system, includes a user-interface focused Presentation Tier (using React and React Native), an Application Tier with business logic in Node.js, and a Data Tier for database management using PostgreSQL. This structure ensures modularity, scalability, and efficient data handling.

The XR Module, integral to the platform, facilitates the creation and interaction with 3D content in both on-screen and VR environments. Key features include the World Builder for creating 3D "Worlds," the Portal Hopper for visualizing and interacting with these worlds, and the Interaction Editor for defining user interactions.

Additionally, the document highlights the Marketplace for XR content distribution and the use of Avatars and Templates to enhance user engagement in virtual spaces.

1.2 Structure of the Document

The structure of this document is as follows:

- Section 2 introduces the XR4ED Main Components with the 3 subsections about the major building blocks of the system as well as the uniqueness of the system.
- Section 3 includes the architecture with the main focus on the design, the microservices, the current architecture and the communication methods.
- Section 4 introduces the XR Module with the architecture, functional and technical specifications.
- Section 5 presents the conclusions.

2 XR4ED Main Components

2.1 XR Module

The XR Module in the XR4ED platform is a comprehensive tool designed for creating, presenting, and experiencing content in a 3D environment, both on screen and in virtual reality (VR). Its main functionalities include:

- **World Builder**: This is a user-friendly editor within the XR Module, used for creating 3D content, referred to as "Worlds". It allows users to import 3D models, audio, images, textures, and videos, and to construct interactive 3D environments. The World Builder supports various file types and enables users to scale, move, rotate objects, add basic shapes, assign textures or colors, and set up lighting and skyboxes.
- **Portal Hopper**: This component is used to visualize and run the worlds created by the World Builder. It operates on various platforms, including Windows PCs, Apple PCs, and Android-based VR headsets. The Portal Hopper offers different modes of interaction, including VR, 3rd-person, and 1st-person views, and it enables users to navigate and interact within the 3D worlds.
- **Interaction Editor**: Within the World Builder, this tool allows users to define interactive behaviors and rules, such as setting up single or multi-user environments, creating sequences or sets of goals, and defining how users can interact with objects in the world.
- **Test Environment**: This feature enables creators to test and verify the visuals and interactions in the worlds they've built, ensuring they work as intended before uploading to servers or saving as templates.
- **Templates and Plug-ins Support**: The XR Module comes with a set of predefined templates (like classrooms, auditoriums, labs, etc.) and supports plug-ins to add new functionalities, such as multiplayer components or specific experiments.
- **Server Integration**: Worlds created with the World Builder can be uploaded to servers, allowing other users to access and run them using the Portal Hopper.

The XR Module's versatility and ease of use make it a core component of the XR4ED platform, enabling the creation and sharing of immersive, interactive 3D experiences for educational and other purposes.

2.2 Marketplace

In the XR4ED platform, the Marketplace serves as a centralized hub for users to access, purchase, and download various digital assets and content related to Extended Reality (XR). The primary functions and features of the Marketplace are:

- **Manage Products**: It allows for the uploading and management of XR-related products. This includes handling product descriptions, categorizations, and any associated metadata.

- **Content Distribution**: Users can browse and purchase various XR content, such as 3D models, immersive environments, educational materials, and interactive experiences.
- **Shopping and Transactions**: The Marketplace includes features like a shopping cart for accumulating desired products, a checkout process for transactions, and secure payment systems to facilitate purchases.
- **User Accessibility**: Designed to be user-friendly, it enables easy access to a wide range of XR content for different users, including educators, students, and XR enthusiasts.
- **Digital Rights Management**: It handles the licensing of products, ensuring that creators' rights are protected while providing users with clear information regarding the usage rights of purchased content.

2.3 Avatars and Templates

In the XR4ED platform, the Avatars and Templates play a crucial role in enhancing the user experience and interactivity within the virtual environments.

2.3.1 Avatars

Personalization: Avatars in XR4ED allow users to represent themselves in the virtual world. These digital personas can often be customized, offering a range of options for appearance, attire, and accessories, thereby enhancing the personalization of the experience.

Interaction and Presence: Avatars facilitate interaction within the virtual environment, making it possible for users to communicate, collaborate, and engage with the content and with each other in a more immersive and human-like manner.

2.3.2 Templates

Pre-designed Frameworks: Templates in XR4ED are pre-designed frameworks or environments that users can utilize to create or customize their own virtual worlds quickly. They serve as starting points, reducing the time and effort required to build complex scenes from scratch.

Variety and Flexibility: These templates might include a variety of settings such as classrooms, laboratories, or meeting spaces, each tailored to different use cases. They provide flexibility and ease for users, especially those who might not have extensive skills in 3D design.

Both avatars and templates are integral to the XR4ED platform, significantly contributing to its usability, accessibility, and overall appeal to a wide range of users, particularly in educational and collaborative contexts.

2.4 Unique aspects of XR4ED Platform

The unique aspects of the XR4ED system can be categorized in 2 aspects, functionality based and technical uniqueness.

2.4.1 Unique Functionalities

- **Intuitive World Builder**: The World Builder within the XR Module is a hallmark of the platform's functionality. It empowers users, regardless of their technical expertise, to easily create and manipulate 3D environments ("Worlds"). This tool supports various file types for importing 3D models, audio, images, and videos, enabling educators and students to construct interactive and rich 3D educational content.
- **Versatile Portal Hopper**: The Portal Hopper is designed to visualize and interact with the 3D worlds created by the World Builder. Its versatility lies in its compatibility with multiple platforms, including PCs and VR headsets, and its support for various modes of interaction, such as VR, 3rd-person, and 1st-person views. This flexibility enhances the platform's usability across different hardware and user preferences.
- **Dynamic Interaction Editor**: This component of the World Builder allows users to define interactive behaviors and rules within the 3D environments. It facilitates the creation of both single and multi-user experiences, enabling educators to design interactive and collaborative learning scenarios, which are crucial in modern pedagogy.
- **Marketplace Integration**: The XR4ED Marketplace serves as a centralized hub for accessing, purchasing, and downloading digital XR assets. This feature simplifies the process of obtaining necessary resources for immersive experience creation, providing a seamless flow from resource acquisition to content development.
- **Customizable Avatars and Templates**: Enhancing user engagement, the platform offers customizable avatars and a variety of templates. Avatars allow users to have a personalized presence in the virtual world, while templates provide ready-to-use frameworks for quick development of virtual environments, catering to different educational scenarios.
- **Unified Authoring Tools**: The platform includes innovative authoring tools that integrate seamlessly with the VR environment. These tools enable dynamic creation and modification of VR content, streamlining the process from concept to classroom deployment. This feature significantly reduces the technical barriers traditionally associated with VR content creation.
- **Focus on Collaborative and Immersive Learning**: XR4ED is purpose-built for education, emphasizing collaborative and immersive learning experiences. Its tools and functionalities are geared towards making learning more engaging, interactive, and effective, aligning with contemporary educational goals.
- **Accessibility and User-Friendliness**: The platform's design prioritizes accessibility and ease of use, making advanced XR technologies approachable for educators and learners. This focus on

user-friendliness is evident in the intuitive interfaces and guided workflows, which lower the entry barrier to adopting XR in education.

2.4.2 Unique Technical aspects

- **Sophisticated 3-Tier Architecture**: At its core, the XR4ED platform is built on a 3-tier architecture, comprising the presentation tier, application tier, and data tier. This structure enhances the platform's modularity and scalability, enabling it to efficiently handle the complexities and demands of XR environments.
- **State-of-the-Art Presentation Tier**: Utilizing React and React Native, the presentation tier of the platform is designed for creating highly responsive and interactive user interfaces. This ensures a seamless and engaging user experience, crucial for immersive learning.
- **Robust Application Tier**: Written in Node.js, the application tier contains the business logic that supports the platform's core functionalities. The use of a microservices architecture in this tier allows for the independent development and scaling of different components, contributing to the platform's agility and robustness.
- **Secure and Efficient Data Tier**: The data tier employs PostgreSQL for database management, ensuring secure and efficient handling of data. This tier is critical for managing user information, content data, and transaction records, maintaining the platform's integrity and reliability.
- **Advanced Integration with VR Technologies**: The XR4ED platform is seamlessly integrated with various VR technologies and headsets. This integration is vital for providing a consistent and high-quality immersive experience across different hardware setups. Working with open standards such as OpenXR allows for maximum compatibility and future proofing.
- **Optimized Performance for Real-Time Interactions**: The technical design of the platform prioritizes high performance, crucial for real-time interactions in XR environments. This includes efficient data processing and minimal latency, ensuring smooth and immersive experiences for users.
- **Comprehensive Security Measures**: The XR4ED platform incorporates robust security protocols and measures. This includes secure data handling, authentication processes, and protection against unauthorized access, preserving the integrity and confidentiality of user data.
- **Scalability and Future-Proofing**: The platform's technical architecture is designed to be scalable and adaptable to future technological advancements. This ensures that the XR4ED platform can evolve with emerging trends and innovations in XR technology. The use of microservices allows the ability to add new services without any significant change in the overall system design.
- **Interoperability and Flexibility**: A key technical feature of the XR4ED platform is its interoperability with various systems and tools. This flexibility enables the platform to integrate with different educational technologies and software, enhancing its utility and applicability in diverse educational settings.

3 XR4ED Architecture

The XR4ED platform employs a sophisticated 3-tier architecture designed to support the complexities of Extended Reality (XR) experiences. It features a range of microservices, including user authentication, license management, product handling, and file management. The architecture is divided into a presentation tier built with React and React Native for user interfaces, an application tier containing the business logic written in Node.js, and a data tier responsible for database management using PostgreSQL. This structure ensures a scalable, flexible, and efficient framework, capable of supporting the diverse needs of an XR ecosystem.

3.1 Architecture design principles

The architecture of the XR4ED platform is designed based on a set of core principles to ensure it meets the complex demands of Extended Reality (XR) environments. Some key principles based on the structure and features of the platform:

- **Modularity and Scalability**: The use of microservices helps achieving a modular approach, allowing different components of the platform to be developed, updated, and scaled independently. This ensures the platform can grow and adapt to changing needs and technologies.
- **User-Centric Design**: The architecture emphasizes user experience, ensuring that the platform is accessible and intuitive for various user groups, including educators, students, and XR developers.
- **Interoperability and Flexibility**: Given the diverse range of functionalities (like user authentication, file management, and product management), the architecture is designed to support interoperability and flexibility, enabling seamless integration of different systems and tools.
- **Security and Reliability**: Essential for any digital platform, particularly one handling user data and commercial transactions, the architecture is incorporating robust security measures and ensure high reliability and uptime.
- **Performance Optimization**: Performance is critical in XR environments to ensure real-time interaction and immersion. The architecture is optimized for high performance, with efficient data processing and minimal latency.
- **Future-Proofing and Innovation**: The architecture is designed with future advancements in mind, allowing for the integration of emerging technologies and innovative features in the field of XR.

3.1.1 Microservices-based architecture

Microservice-based architecture, as utilized in the development of platforms like XR4ED, represents a paradigm shift in software design, focusing on decomposing applications into smaller, independent

units known as microservices. Each microservice in this architecture is designed to perform a specific function, operating autonomously, and communicating with other services through well-defined APIs and protocols. This approach contrasts sharply with traditional monolithic architectures, where all functions are intertwined within a single application.

In a microservice architecture, services are developed, deployed, and scaled independently, offering significant advantages in terms of agility and flexibility. This independence allows for rapid development and deployment cycles, as changes to one service do not necessitate alterations or redeployments of other services. Furthermore, each service can be written in the most appropriate programming language and framework for its specific task, fostering technological diversity and innovation.

The main benefits of a microservices-based architecture are:

- **Enhanced Scalability**: Microservices can be scaled independently, allowing for more precise scaling of individual components of an application based on specific demands.
- **Improved Fault Isolation**: In microservices, a failure in one service does not necessarily bring down the entire system, which enhances overall system reliability.
- **Flexible Technology Stack**: Different microservices can be written in different programming languages or use different technology stacks, allowing for the use of the best tool for each specific job.
- **Easier Maintenance and Updating**: Smaller, well-defined services are easier to understand, maintain, and update compared to a large monolithic code base.
- **Faster Development and Deployment**: Independent services can be developed and deployed more rapidly, enabling faster iteration and quicker release of new features.
- **Targeted Scaling and Optimization**: Resources can be allocated and optimized for individual services based on their specific needs, leading to more efficient use of infrastructure.
- **Enhanced Team Productivity**: Smaller, focused teams can work on individual microservices, reducing coordination overhead and increasing development speed.
- **Seamless Integration and Automation**: Microservices lend themselves well to continuous integration and deployment practices, supporting automated testing and deployment processes.
- **Resilience and Reliability**: The decoupled nature of microservices means the system is less likely to fail globally, enhancing the overall resilience of the application.
- **Adaptability to Business Changes**: Microservices are well-suited to agile and flexible business models, as they can be easily adapted or expanded to accommodate new business requirements or changes in direction.

The main microservices used in XR4ED, as shown in Figure 1, include the following:

- User Authentication Microservice:
    - This service manages all aspects of user identity and access control. It handles user login, logout, and registration processes.

- It also manages user permissions and role assignments, ensuring that users have appropriate access levels based on their roles within the platform.

- Licenses Microservice:
  - Focused on managing different types of licenses within the platform, this service handles license key authentication and tracks license expiration.
  - It plays a critical role in digital rights management, ensuring that content and software use comply with licensing agreements.
- Products Microservice:
  - This service is responsible for managing the products available on the XR4ED platform. It includes functionalities for uploading content and managing product metadata.
  - It categorizes products, and manages the shopping cart, checkout process, and payment systems, providing a complete e-commerce solution within the platform.
- File Management Microservice:
  - It manages all aspects of file handling within the platform. This includes uploading files, identifying file types, and storing files efficiently.
  - The service also facilitates the downloading of files, ensuring that users can access and retrieve digital assets as needed.

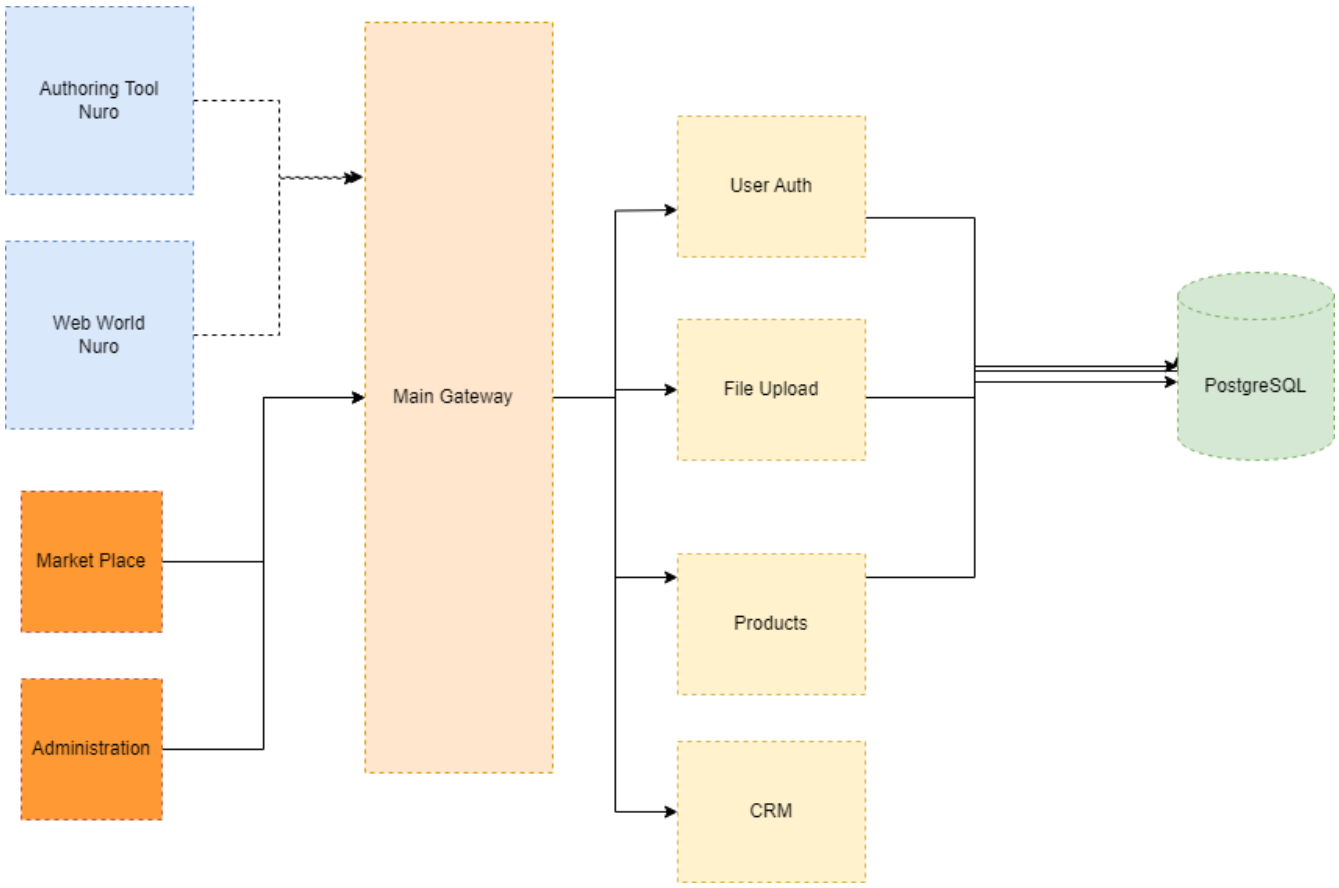


Figure 1 XR4ED Initial Microservices

Figure 1, depicts the initial microservices envisioned for XR4ED platform. The architecture helps in easily adding new microservices in the future, in case the need arises.

In terms of workflow, for example, when a user logs into the XR4ED platform, the User Authentication microservice verifies the user's credentials and assigns appropriate access rights. If the user interacts with the Marketplace to purchase a product, the Products microservice handles the product listings, shopping cart, and transaction processes. Meanwhile, the Licenses microservice ensures that any content accessed or purchased is in compliance with proper licenses and DRM (Digital Rights Management). Throughout these interactions, the File Management microservice facilitates the storage and retrieval of any files needed.

By structuring these functionalities into separate microservices, XR4ED achieves a high degree of modularity, scalability, and flexibility, allowing each service to be optimized for its specific task and ensuring the overall system's efficiency and reliability.

3.1.2 3-tier architecture

The XR4ED platform utilizes a 3-tier architecture, an approach that divides the system into three distinct layers, each with its specific responsibilities and functionalities. This architectural model enhances the platform's scalability, maintainability, and efficiency.

Presentation Tier:

- The first layer is the Presentation Tier, built using React and React Native for mobile applications. This tier is responsible for the user interface and user experience. It presents the system's data and functionality in a format that users can easily interact with.
- Communication with other tiers is achieved through REST API calls. This tier is crucial for ensuring a seamless and intuitive user experience, adapting the presentation of data and functionalities according to user actions and requests.

Application Tier (Backend):

- The second layer, known as the Application or Logic Tier, is written in Node.js. This tier contains the business logic of the platform, processing user commands, making logical decisions and evaluations, and performing calculations.
- It is structured on a microservices architecture, allowing different aspects of the platform to be handled by distinct, independently deployable services. This tier also integrates Geographic Information System (GIS) functionalities using the GEO Server and GIS API, which are vital for processing and managing geographical data, a key aspect in many XR applications.

Data Tier:

- The third layer is the Data Tier, which includes databases and data management systems. In XR4ED, PostgreSQL is used for storing all related data. This tier manages the storage and retrieval of data, ensuring that the platform has a reliable and consistent data storage mechanism.
- The Data Tier is fundamental for data integrity and security. It handles all the data used by the platform, including user information, product details, and other essential data, ensuring that data is correctly stored, retrieved, and maintained.

In summary, the 3-tier architecture of XR4ED effectively separates concerns, with each tier handling specific aspects of the platform's functionality. The Presentation Tier manages user interaction, the Application Tier handles the business logic, and the Data Tier takes care of data storage and management. This structured approach not only facilitates easier maintenance and scalability but also allows for more efficient development and smoother integration of new features or technologies.

3.2 Architecture design process

The design process of the XR4ED architecture, utilizing a 3-tier model, followed a comprehensive and methodical approach, ensuring that the final system met the complex requirements of an Extended Reality (XR) platform:

- **Requirement Analysis:**
  - The initial phase involved a deep understanding of the functional and non-functional requirements specific to XR applications, including user interaction needs, system performance, security requirements, and scalability considerations.
- **Defining the System's Scope:**
  - The scope of XR4ED was clearly defined, outlining the types of users it would serve, the nature of XR interactions it would support, and the extent of data processing and management involved.
- **High-Level Architectural Planning:**
  - A decision was made to adopt a 3-tier architectural style, delineating the separation of the presentation, application, and data management aspects of the platform.
- **Selecting Technology Stack:**
  - The technology stack for each tier was carefully chosen: React and React Native for the Presentation Tier, Node.js for the Application Tier, and PostgreSQL for the Data Tier.
- **Designing Each Tier:**
  - The Presentation Tier was designed with a focus on user interface and experience, tailored to the needs of XR users.
  - The Application Tier was structured around a microservices architecture, detailing the business logic, API endpoints, and inter-service communication.
  - The Data Tier's design involved creating a robust and secure database schema and data storage strategy.
- **Defining Inter-Tier Communication:**
  - The communication protocols between the tiers were established, ensuring seamless data flow and interactions between the user interface, business logic, and data storage components.
- **Security Planning:**
  - Security measures were integrated at each tier, considering the sensitive nature of XR data and the need for secure user interactions.
- **Prototyping and Feedback:**
  - A prototype of XR4ED was developed to test the fundamental aspects of the 3-tier architecture and gather initial feedback from users and stakeholders.
- **Detailed Design and Documentation:**
  - Detailed design documents were prepared for each tier, specifying the implementation details, interfaces, and interactions within the architecture.

- **Development and Iteration:**
  - The development process was iterative, allowing for adjustments based on continuous testing and feedback, ensuring alignment with XR requirements.

The experience and knowledge of each of the partner involved in the development of XR4ED platform equipped the consortium with specific knowledge and skillset to carry out each of the task.

3.3 Current Architecture

Figure 2 delineates the contemporary architecture of the XR4ED platform, presenting a layered structure that underpins the development and deployment of Extended Reality (XR) environments. At the apex of the architecture are critical components such as Avatars, Templates, and Assets—integral elements that form the building blocks for XR content creation. These components, along with a suite of additional plugins, are developed to equip users with a robust set of tools for crafting immersive experiences within the platform.

Central to the architecture are the microservices, which function dually as conduits for inter-component communication and as the creators of essential platform-level functionalities. This includes systems for user authentication, product management, and a Digital Rights Management (DRM) system, ensuring that the platform operates with efficiency and security.

The ecosystem is designed to be extensible, with third-party providers being able to introduce specialized functionalities tailored to the platform's needs, such as chat-based asset creation tools, enhancing the platform's versatility and creative potential.

The architecture's main frontend components are constituted by the XR Module and the Marketplace. The Marketplace boasts a standalone web application UI that provides a dedicated space for asset transactions and is seamlessly integrated with the authoring tool for ease of access.

Within the XR Module lie two principal elements: the authoring tool, which includes the world builder and editor functionalities, and the Web World, also known as the Portal Hopper. The Portal Hopper operates as the player for worlds constructed using the XR4ED platform, facilitating the exploration and interaction within these virtual spaces. This architecture acts as an guide to the first version of the platform and will be updated in the future.

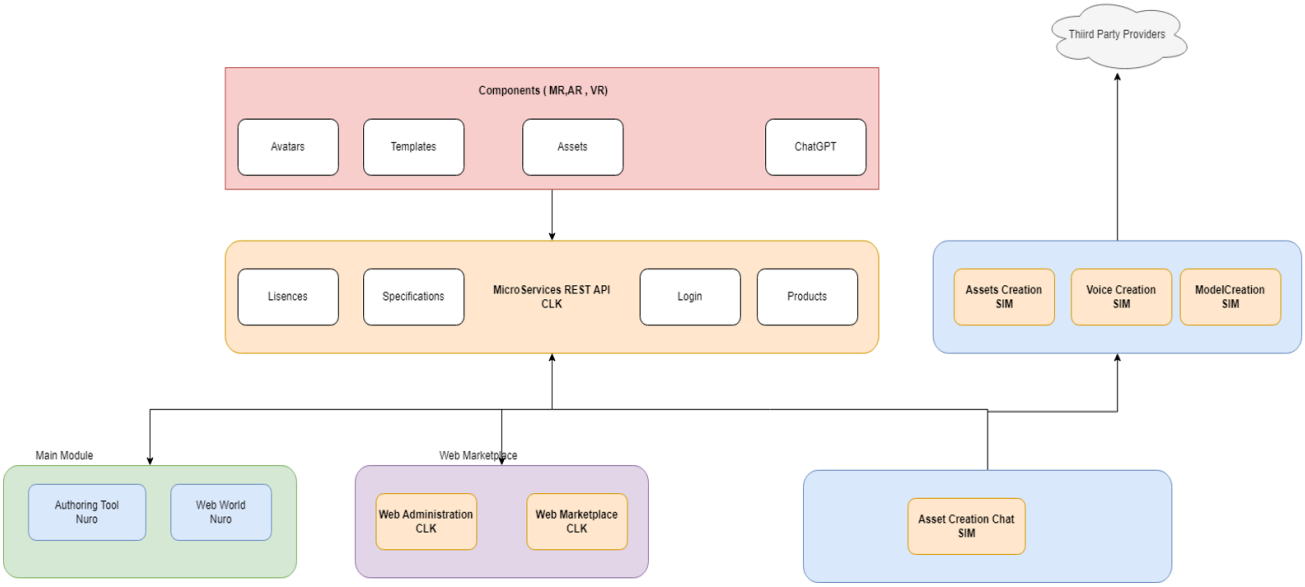


Figure 2 XR4ED Components

3.4 Communication between components

The Main Gateway, as shown in Figure 1, within the XR4ED platform acts as the central nervous system for routing communication between the platform's front-end and its array of microservices. It serves as the single point of entry for all external API requests, effectively decoupling the client-side interactions from the backend processes. Once a request reaches the Main Gateway, it authenticates the request, determines the appropriate service needed, and forwards the request to the relevant microservice, whether that's for user authentication, file upload, product information, or CRM. The gateway also handles responses from these services, delivering them back to the requesting client.

Additionally, the Main Gateway incorporates critical functions such as load balancing to distribute network traffic evenly across services, ensuring optimal performance and reliability. It enforces security protocols, manages API version control, and implements rate limiting to safeguard against system overload. The Main Gateway not only streamlines API management but also provides a layer of abstraction that enhances security, facilitates scalability, and ensures a seamless user experience across the XR4ED platform.

The XR4ED platform harnesses the power of REST (Representational State Transfer) APIs to enable seamless interactions between clients and services within its architecture. REST APIs operate over standard HTTP methods, such as GET, POST, PUT, and DELETE, which are universally recognized and understood across the web. This allows for a stateless communication protocol that is lightweight, flexible, and scalable, making it ideal for the platform's diverse and extensive interactions. By using REST

APIs, the platform ensures that data can be easily accessed and manipulated by various parts of the system in a consistent and standardized manner, enhancing interoperability and integration across different services and components. It's a choice that underpins the system's ability to provide a responsive and robust user experience, crucial for the demands of an Extended Reality environment.

4 XR Module

The XR Module is a cornerstone component of the XR4ED platform, engineered to facilitate the creation, visualization, and interaction with immersive 3D content. It is a sophisticated suite of tools designed to empower educators, creators, and users to build and experience virtual environments for various applications ranging from education to professional training.

4.1 XR Module Architecture

Figure 3 showcases the placement of XR Modules inside the XR4ED system and communication of the XR Module with the various XR4ED components.

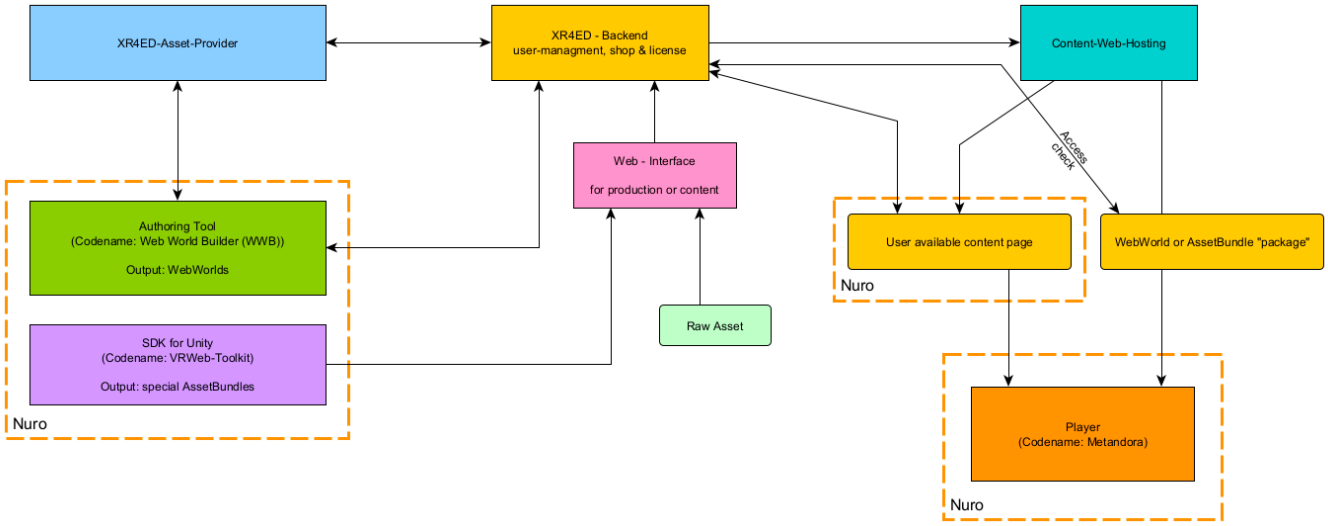


Figure 3 XR Module components

The architecture of the XR Module within the XR4ED platform orchestrates the creation, management, and delivery of Extended Reality (XR) content through a well-defined workflow.

Central to the XR Module's architecture is the **Authoring Tool**, also known as the **World Builder** (WWB), which creators use to develop immersive 3D environments termed WebWorlds. This tool outputs content ready for integration and distribution within the XR4ED ecosystem.

Complementing the Authoring Tool is the **SDK for Unity**, also known as VRWeb-Toolkit, which generates specialized AssetBundles for use in Unity-developed XR environments. These AssetBundles are complete packages that include a range of assets necessary for creating rich XR experiences.

Both the Authoring Tool and the Unity SDK interface seamlessly with the XR4ED-Backend. The XR4ED backend is the operational core for user management, facilitating transactions in the marketplace, and

administering licenses. It ensures that the XR Module's components are synchronized, and user interactions are efficiently managed.

The "**Raw Assets**" generated are then either directly utilized in content production or made accessible through the Web Interface for content curation and presentation.

For content distribution, the architecture includes a **Content-Web-Hosting service**. This service hosts the created WebWorlds or AssetBundle packages, making them available for user access over the web.

The architecture integrates a user-centric content page for any specific user generated or imported assets. Here, users can browse and access the XR content available, facilitated by a straightforward and accessible interface.

The culmination of the architecture is the "Portal Hopper," also known as Metandora, where the XR content comes to life. This component is dedicated to rendering the WebWorlds and deploying the AssetBundles, thus enabling users to interact with and experience the immersive content produced by the XR Module.

This architecture highlights a streamlined and efficient content flow from creation to end-user engagement, with each component fulfilling a specific function within the XR Module's ecosystem. It encapsulates the platform's vision to deliver a comprehensive and user-friendly suite for XR content creation, management, and consumption.

Furthermore, the flow of information and data is shown in Figure 3. The flow of data and interaction within the XR4ED platform starts with users uploading raw assets through a dedicated web interface. This interface is designed to facilitate the production and management of content, a process that is handled efficiently by the XR4ED backend services. Once uploaded, these assets are stored in the XR4ED Asset Provider, which serves as a repository for the raw assets to be accessed by the authoring tools, such as the World Builder and Editor, as well as the SDK for Unity, allowing for the creation and manipulation of immersive worlds.

These constructed worlds are then hosted on a high-speed storage system, optimized for rapid read and write operations. This ensures that users can load new environments quickly, enhancing the overall experience by minimizing wait times. Before these environments are made available to users, they undergo a conversion process into Unity Asset Bundles[1] using a specialized server. This conversion is accompanied by an access check, performed by the XR4ED backend, to verify user permissions and ensure secure distribution of content.

Furthermore, the backend plays a pivotal role in content discovery, presenting users with a content page that displays the available worlds they can access. This curated selection simplifies the user's navigation and choice of environments they wish to explore or utilize.

[1] https://docs.unity3d.com/Manual/AssetBundlesIntro.html

Finally, the complete and vetted environments, now packaged as Unity Asset Bundles, are loaded into the Player, known as the Portal Hopper. The Portal Hopper is the user-facing application that renders and allows interaction with the XR environments, providing a seamless and immersive experience. Through this well-orchestrated flow, the XR4ED platform ensures a streamlined process from asset upload to the delivery of interactive and engaging virtual environments.

4.2 Functional Specifications

Here are the comprehensive functional specifications for the specific components XR Module within the XR4ED platform:

World Builder:

- An editor for creating 'Worlds', which are 3D content environments.
- Supports importing fbx and obj file types for 3D models.
- Supports audio file imports including .mp3 and .wav formats.
- Allows importing images and textures in BMP, JPG, PNG, and GIF formats.
- Supports video imports in MP4 and WMF formats.

World Editor:

- Enables users to scale, move, and rotate objects within the world.
- Allows the addition of basic shapes like spheres and cubes.
- Users can assign textures or colors to models and their surfaces.
- Interactive behavior can be assigned to objects, such as picking up items or pressing buttons.
- Features for setting up lighting and skyboxes are included.

Interaction Editor:

- Users can define rules for interactions within the world.
- Allows the setup of single-user or shared multi-user environments.
- Enables the creation of sequences and goal sets within the world.

Test Environment:

- Provides a space to create a World and run the Portal Hopper for testing.
- Ensures visuals and interactions work as intended before release.

Templates and Plug-ins:

- Worlds can be saved as templates for various projects.
- A predefined set of templates is available, such as classroom, auditorium, and lab environments.

- Supports plug-ins that can modify the World Builder's UI and add new functionalities like multiplayer features or specific experiments.

Portal Hopper:

- Visualizes and runs the worlds created by the World Builder.
- Operable on Windows PCs, Apple PCs, and Android-based VR headsets.
- Offers VR-Mode, 3rd-person, and 1st-person views (VR-Mode requires a connected VR headset) shown in Figure 4 and Figure 5.
- Allows user movement and interaction as defined in the World Builder.
- Supports networked environments for both single and multi-user experiences.

Figure 4 Portal Hopper - 3rd person view

Figure 5 Portal Hopper - VR Mode

4.2.1 Module wide functional specifications

User Experience (UX) Design:

- The XR Module boasts an intuitive UI with drag-and-drop functionalities, context-sensitive menus, and customizable control panels.
- It incorporates accessibility features such as text-to-speech, high-contrast visuals, and compatibility with assistive technologies.

Performance Metrics:

- Asset import and world rendering are optimized for efficiency, targeting load times of fewer than 10 seconds for standard-sized assets.
- VR experiences aim for a consistent 90 FPS frame rate to ensure smooth visualization and prevent motion sickness.

Compatibility and Integration:

- Fully compatible with leading VR headsets like Oculus Rift, HTC Vive, and others, ensuring a broad user base.
- Offers API integrations for major third-party platforms, enabling content import/export and data synchronization.

Content Management:

- Version control is built-in for content, allowing creators to track changes and revert to previous versions when necessary.
- A centralized asset library enables effective management and organization of 3D models, textures, and other media.

Networking and Collaboration:

- Real-time collaboration tools are embedded, allowing multiple users to work on world-building simultaneously.
- All network communications are secured with end-to-end encryption, adhering to the latest cybersecurity standards.

Analytics and Reporting:

- The module includes analytics for measuring user interactions, dwell time, and other engagement metrics within the worlds.
- Customizable reports can be generated by creators to assess performance and inform future design decisions.

Customization and Extensibility:

- Users can leverage an SDK to develop custom functions and extensions, enhancing the platform's versatility.
- The platform allows end-users to personalize their avatars and environments through an extensive set of customization options.

Support and Documentation:

- Extensive documentation is provided online, including tutorials, FAQs, and a comprehensive API reference.
- Support is facilitated through a dedicated helpdesk, community forums, and a suite of training materials for both users and developers.

Quality Assurance:

- Prior to release, new features undergo rigorous testing across multiple devices and user scenarios.
- A feedback system is implemented directly within the platform, enabling users to report issues or suggest enhancements quickly.

Deployment and Maintenance:

- The XR Module uses a continuous integration/continuous deployment (CI/CD) pipeline, ensuring smooth rollouts of updates.
- Regular maintenance windows are scheduled during off-peak hours with clear communication to users to minimize disruption.

Security Features:

- Robust user authentication mechanisms, including multi-factor authentication, are in place to protect user accounts.
- Adherence to international privacy standards, such as GDPR, ensures user data is handled with the utmost care and compliance.

4.3 Technical Specifications

The technical specifications for the XR Module in the XR4ED platform encapsulate the system's hardware and software requirements, as well as its technical capabilities. Here's a detailed rundown of the technical specifications:

System Architecture:

- Built on a microservices architecture to ensure modularity and ease of maintenance.
- Utilizes a 3-tier system comprising the presentation, logic, and data tiers.

Hardware Requirements:

For World Builder:

- CPU: Intel i7 4 core CPU or equivalent.
- GPU: GeForce 1060 or equivalent, capable of 3D rendering.
- RAM: Minimum of 16 GB, with 32 GB recommended for optimal performance.

For Portal Hopper:

- CPU: Intel i7 4 core CPU or equivalent.
- GPU: VR-ready graphics card.
- RAM: Minimum of 8 GB, with 16 GB recommended.

Software Requirements:

- Supports major operating systems including Windows 10, macOS, and Android for VR headsets.
- For development: Unity SDK support for content creation, with a custom toolkit for XR development.

APIs and Protocols:

- RESTful API design for seamless interaction between the frontend, backend, and microservices.
- WebSockets for real-time bi-directional communication, essential for multi-user collaboration features.

Database and Storage:

- PostgreSQL for relational data management, supporting complex queries and transactions.
- Supports cloud storage solutions for asset hosting and distribution, with backup and recovery protocols in place.

Networking:

- Designed to operate over both IPv4 and IPv6 networks.
- Uses TLS 1.3 for secure data transmission across the network.
- Required 50 MB/sec connection with latency below 10ms for real time communication in multi user VR environments.

Security:

- Implements industry-standard encryption for data at rest and in transit.
- Regular security audits and compliance with OWASP top 10 for web application security.

Extended Reality Support:

- Compatible with various XR platforms, providing native support for headset tracking, controllers, and room-scale XR.
- Supports standard VR file formats for assets, including .gltf for 3D models.

Performance Optimization:

- Utilizes efficient algorithms for rendering and physics calculations to maximize performance.
- Adaptive quality settings to balance visual fidelity with performance across diverse hardware configurations.

Scalability:

- Designed for horizontal scaling, allowing additional resources to be added as user load increases.
- Load balancers distribute traffic to ensure even load across servers and prevent bottlenecks.

These technical specifications are crafted to ensure the XR Module delivers a powerful and immersive experience, capable of supporting the complex demands of XR environments for a wide user base. They reflect a commitment to high performance, security, and a seamless user experience, essential for the success of the XR4ED platform.

4.4 Current XR Module Specifications

Currently, the XR Module consisting of the World Builder and Portal Hopper with the core functionalities working. The current version of the world builder helps the user build 3D worlds using drag and drop as well as by importing assets. The user can also choose from the predefined template of environements and build on top of that. Figure 6 shows the login screen for the world builder where the user can also sign up to use the platform. Figure 7 depicts the new project screen for the world builder with a few existing templates.

Figure 6 World Builder login screen

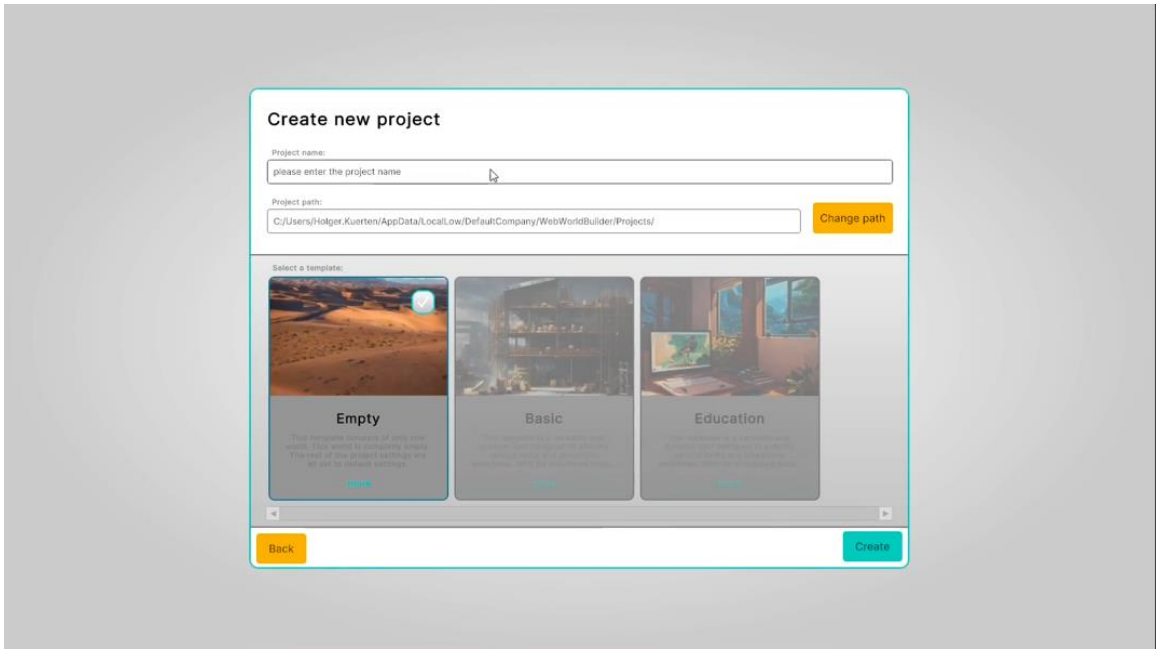


Figure 7 World Builder - New project screen

5 Conclusion

The development and deployment of the XR4ED platform, as detailed in this document, marks a significant step towards simplifying and democratizing the creation of virtual reality (VR) environments and experiences, particularly in educational settings. Traditional barriers, such as the need for extensive programming knowledge and familiarity with a variety of game engines and tools, have been addressed through the platform's comprehensive architecture and user-friendly interface.

Key to this advancement is the XR4ED platform's ability to streamline the process of accessing and utilizing diverse resources, including 3D models and avatars. By integrating these resources into a unified system, the platform significantly reduces the complexity and technical know-how required from educators to create immersive VR content. This approach not only makes the technology more accessible but also ensures that educators can focus on the pedagogical aspects of their content without being hindered by technological constraints.

The platform's XR Modules play a crucial role in this regard. They enable users to work within a VR environment to dynamically select and integrate 3D models from the XR4ED Marketplace, thereby creating customized VR environments and content for their educational modules. This integrated approach, combined with the platform's robust backend and database support, ensures a seamless experience for both content creators and end-users.

In conclusion, the XR4ED platform is poised to become a leading solution in the realm of XR systems for education, learning, and training within the European Union. Its innovative design and user-centric approach promise to enhance the quality and reach of educational experiences, leveraging the power of XR technology to create engaging, effective, and accessible learning environments.