MediaVerse

A universe of media assets
and co-creation opportunities

D5.6

VR Collaboration, Social Media Analytics, and Immersive Storytelling Authoring Tools v3

Project Title	MediaVerse
Contract No.	957252
Instrument	Innovation Action
Thematic Priority	ICT-44-2020 Next Generation Media
Start of Project	1 October 2020
Duration	36 months

Deliverable title	VR Collaboration, Social Media Analytics, and Immersive Storytelling Authoring Tools v3
Deliverable number	D5.6
Deliverable version	V1.0
Previous version(s)	D5.3
Contractual Date of delivery	30.06.2023
Actual Date of delivery	14.07.2023
Nature of deliverable	Other
Dissemination level	Public
Partner Responsible	Vragments GmbH (VRAG)
Author(s)	Girish Vadlamudi (VRAG), Jens Brandenburg (VRAG), Stephan Gensch (VRAG), Dimitrios Ververidis (CERTH), Tasos Papazoglou Chalikias (CERTH), Spyros Papafragos (ATC)
Reviewer(s)	Rahel Luder (STXT), Florian Maillard (STXT), Violeta Vasileva (AS)
EC Project Officer	Luis Eduardo Martinez Lafuente

Abstract	This deliverable reports on the developments of the authoring tools supported by the MV platform during the final project year. It is sectioned into a) supporting functionality updates on MV nodes, b) updates on the Social Analytics Engine by ATC, and c) updates on the VR authoring tools as services by CERTH and VRAG.
Keywords	Accessibility, Authoring Tools, Immersive Storytelling, VR Sandbox

Copyright

Revision History

VERSION	DATE	MODIFIED BY	COMMENTS
V0.1	22/12/2022	Dimitrios Ververidis (CERTH)	VROdos developments and results
V0.2	27/12/2022	Dimitrios Ververidis (CERTH)	Review of Section 5 about VROdos
V0.3	22/05/2023	Stephan Gensch (VRAG)	Merging V0.2 and updating ToC
V0.4	14/06/2023	Spyros Papafragos (ATC)	Input in Social Analytics section
V0.5	16/06/2023	Stephan Gensch (VRAG)	Prepare partner input for internal review
V0.6	21/06/2023	Tasos Papazoglou Chalikias (CERTH)	Finalization of VROdos section
V0.7	23/06/2023	Rahel Luder (STXT), Florian Maillard (STXT), Violeta Vasileva (AS)	Review and minor adaptions
V0.8	25/06/2023	Manos Schoinas, Evangelia Kartsounidou (CERTH)	Comments and Review
V0.9	26/06/2023	Stephan Gensch (VRAG)	Incorporated comments from coordinator review
V1.0	14/07/2023	Evangelia Kartsounidou, Symeon Papadopoulos (CERTH)	QA and Final Document

Glossary

ABBREVIATION	MEANING
AP	Average Precision
BERT	Bidirectional Encoder Representations from Transformers
CNN	Convolutional Neural Network
DAM	Digital Asset Management
GDPR	General Data Protection Regulation
GUI	Graphic User Interface
HMD	Head-mounted Display
ML	Machine Learning
MV	MediaVerse
NAF	Networked-Aframe
NAS	Neural Architecture Search
SCRFD	Sample and Computation Redistribution for Efficient Face Detection
UC3	Use Case 3
UGC	User Generated Content
VR	Virtual Reality
WP	Work Package

Table of Contents

Index of Figures

Index of Tables

Executive Summary

This document presents the research, development, and improvements that have been carried out to advance the Social Analytics Engine and the capabilities of VROdos and Fader360, two innovative authoring tools integrated within the MediaVerse platform. Our work has been primarily driven by the goal of democratizing immersive digital content creation, by offering both intuitive and powerful tools for creators of all skill levels.

The first significant component of our work involves the adaptations made to the MediaVerse Node to better support the authoring tools. Updates have been made to the front-end, incorporating a button that opens the respective project in the authoring tool on every MediaVerse project detail page. On the backend, we added three new API endpoints to facilitate pagination details, enhancing the overall user experience.

In the domain of social analytics, we had to address significant changes announced by Twitter in 2023, including limitations to API access and pricing. To counter this, the Social Analytics Engine was updated with new functionalities, including the creation of PDF reports, improvements in trending and alerting, the introduction of manual sentiment editing, and the creation of a new web application for mass correction of entity extractions. To improve performance, a Kubernetes cluster was set up, along with the addition of a health-monitoring service for the Solr Server.

VROdos, our tool for creating multi-player WebVR experiences, saw a range of enhancements. We incorporated additional features like support for fog and masking, extended the variety of content, introduced texture compression, and met public Internet connectivity standards. Furthermore, we have adapted the equipment required for users to be cost-effective, not exceeding 2000 EUR. For integration with MediaVerse nodes, we focused on user and identity management, importing and storing media from and into MediaVerse nodes, and usage of MediaVerse support functions.

For Fader360, a suite of features has been rolled out, including the ability to incorporate 360-degree backgrounds for video and audio, subtitles display functionality, comprehensive asset and scene management panels, scene links for navigational purposes, and view mode features that offer a more immersive user experience. We have implemented a number of features to ensure a seamless interaction between MediaVerse and Fader360, such as using MediaVerse to store Fader360 story output as a text file, pulling subtitles from MediaVerse, and supporting MediaVerse project ID and node details.

In conclusion, our advancements in the Social Analytics Engine, VROdos, and Fader360 have not only enhanced the capabilities of these tools but also contributed significantly to the overall MediaVerse ecosystem. We believe that these enhancements will foster a more collaborative, inclusive, and creative environment for digital content creation and immersive storytelling. Plans involve continuous improvements in performance and user experience, with a keen focus on addressing user feedback and market trends.

1   Introduction

In an era defined by digital media, immersive and interactive experiences have become an essential component of storytelling and content creation. They not only enrich narratives, but also foster deeper audience engagement and offer novel ways to communicate and express ideas. However, creating these experiences often requires specialized tools and technical skills, which can pose challenges for content creators who might not have such expertise. This was the impetus for the development of intuitive and accessible authoring tools like Fader360 and VROdos, and their integration into the MediaVerse - a holistic platform designed to support diverse media experiences.

The motivation for the work described in this document is rooted in the recognition of these complexities and the desire to simplify the process of creating rich, interactive, and immersive digital content. We embarked on this journey with the vision of transforming storytelling by making cutting-edge authoring tools accessible to all users, regardless of their technical skills. The goal was to democratize the process of content creation, allowing everyone from artists and educators to businesses and amateurs to bring their visions to life in a digital environment.

To this end, we have invested our efforts in research, development, and improvement of the authoring tools integrated within MediaVerse. This entailed refining existing features, developing new functionalities, improving user experience, and ensuring seamless interoperability with the MediaVerse platform. We have also kept a close eye on the changes in social analytics, particularly on platforms like Twitter, to enhance content enrichment and performance tracking.

The primary motivation behind this research was to create authoring tools that offer both power and simplicity. Our aim was to empower users with the tools to create immersive experiences while removing the technical barriers that might hinder their creative process. This document offers insights into the development and integration of these authoring tools, their features, and their potential to transform digital storytelling.

1.1   Relation with Other Activities in MediaVerse

The immersive storytelling tools and the VR sandbox are high-level applications that make use of a variety of MediaVerse services, such as media processing and annotation, digital asset management, or accessibility tools, to name a few. These services are developed in the work packages WP3, WP4, and WP5. As such, there are many activities in other WPs that directly influence the work on the authoring tools.

1.1.1   Relation with Other Activities of Social Analytics Service

Social Analytics functionality developments are based on the Conceptual Architecture Design of WP2, as it is presented in D2.2 – Conceptual Design of the MediaVerse Framework[1] and the integration plans from WP6 that have defined the mechanism to integrate the functionality in a MediaVerse Node ecosystem. The WP2 and WP7 also provide the user feedback to clearly define the specifications for the fine-tuning of Social Analytics features.

---

[1] https://mediaverse-project.eu/wp-content/uploads/2021/10/MediaVerse_D2.2_Conceptual-Design-of-the-MediaVerse-Framework.pdf

1.1.2   Relation with Other Activities of VROdos and Fader360

VROdos and Fader360 are implementations towards VR experiences that use a variety of MediaVerse services, such as media processing and annotation, digital asset management, or accessibility tools, to name a few (see Figure 1: The position of VROdos and Fader360 with respect to MediaVerse architecture.). WP2 gives a good indication on the envisioned architecture that serves as a foundation to integrate the VROdos and Fader360 authoring tools that are also described in technical terms in this deliverable. Finally, WP7 provides the necessary feedback to refine the developments on these tools.
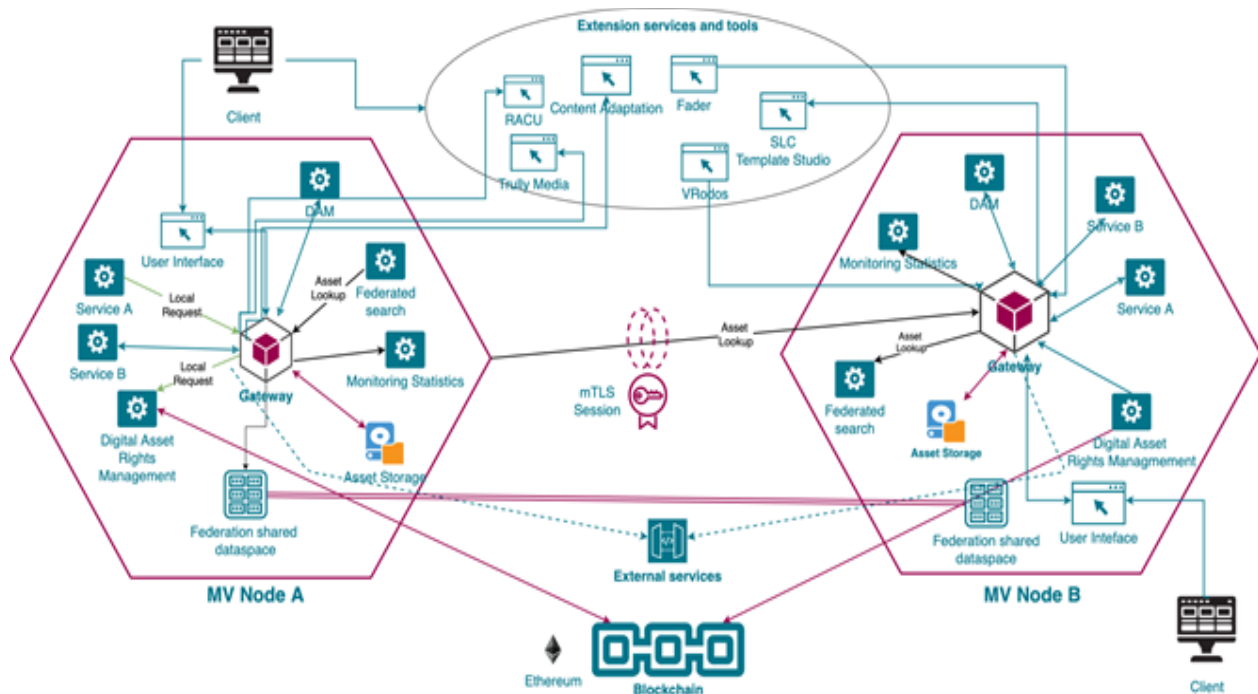
Figure 1: The position of VROdos and Fader360 with respect to MediaVerse architecture.

More precisely, WP2 provided the use cases and requirements engineering to guide feature implementation of for the current reporting period. WP3 provided media adaptations that allow authoring tools to incorporate and use them directly. WP4 provided the foundation for the digital asset management (DAM) to ensure the correct usage of media assets by internal and third-party tools. This includes proper adherence to copy- und usage rights, compensation, and attribution. Within WP6, the integration of the authoring tools into MediaVerse nodes has been finalized. Within WP7, the development progress of WP5 has been regularly tested and evaluated.

1.2   Deliverable Outline

MediaVerse Node Adaptations to Support Authoring Tools: This chapter discusses the feature and API updates that allow MediaVerse projects to open in an authoring tool such as Fader360. This includes additions to user authentication, project management, and asset retrieval functionalities.

Social Analytics for Content Enriching and Performance Tracking: Here, we explore the key areas of focus for the Social Analytics Engine. The chapter discusses the impact of changes in Twitter's API access and pricing, the creation of PDF reports, the refinements in alerting and trending mechanisms, advancements in analysis libraries, and infrastructure improvements for better performance.

VROdos for Authoring VR Experiences Based on 3D Geometries: This section presents VROdos, a web platform enabling users to create multi-player WebVR experiences via a 3D authoring tool. It covers the development of additional features, enhancing content variety, texture compression, meeting public internet connectivity standards, and the use of a projector to enhance user experience. The chapter also defines the final equipment requirements, the integration with MV node(s), and details on user and identity management. Furthermore, it covers how VROdos accesses and stores media from MV nodes, plans for future development, and the potential for exploitation in education, promotion, and entertainment.

Fader360 Immersive Storytelling Toolset – Fader360: This chapter delves into the integration of Fader360 with MediaVerse, focusing on the authentication process, how Fader360 accesses and stores media from MediaVerse, and the use of MediaVerse support functions. It then details the development efforts undertaken for the Fader360 frontend, with descriptions of various features such as 360-degree backgrounds, subtitles display, Asset Management and Properties panels, Scene Settings, Scene Links, and View mode functionalities.

Each chapter offers comprehensive insights into the development and integration of authoring tools with MediaVerse. The information is aimed at providing an understanding of how these tools enhance user experience, streamline workflow, and contribute to more efficient and immersive storytelling and content creation.

2   MediaVerse Node Adaptations to Support Authoring Tools

This chapter describes the development efforts of the final project year within the MV node to support third party authoring tools.

2.1   Feature Updates

A new button is provided on every project detail page within the MediaVerse node, allowing users to open the respective project in the authoring tool (see Figure 2). When the button is clicked, users will be redirected to the authoring tool through a link that contains several parameters, such as the project ID, the URL of the MediaVerse node (depending on the user's selected node), and the user's email address.
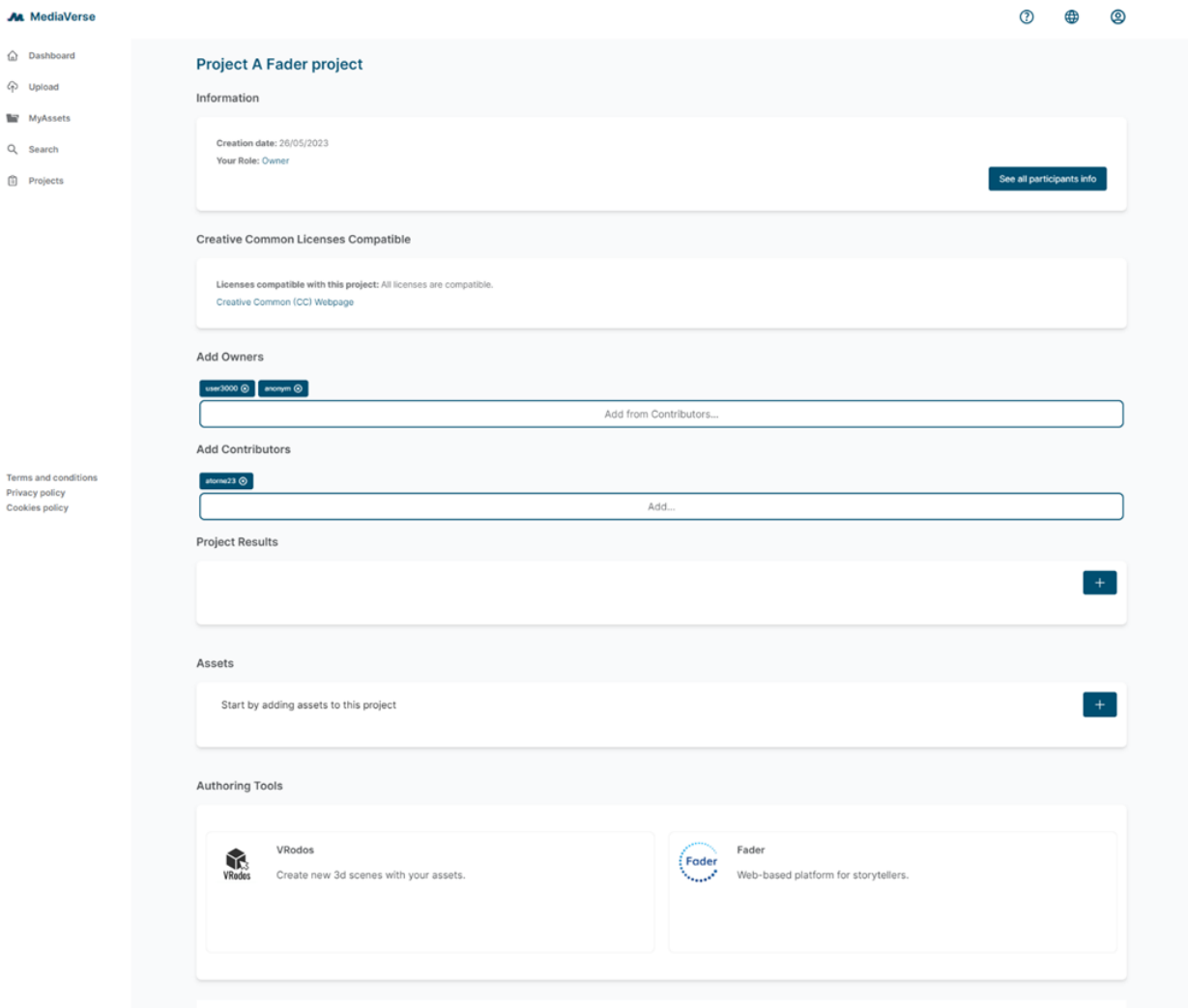
Figure 2: Project detail page with direct links to the authoring tools at the bottom.

In the context of Fader360, the email address is used to determine the user's login status within the Fader360 instance. The project ID is used to retrieve the project and its associated details in Fader360, while the MediaVerse node is crucial for any MediaVerse-related tasks in Fader360, including authentication, project management, and asset retrieval.

2.2   API Updates

In our process of retrieving information related to assets and projects in Fader360, we relied on the existing API endpoints. However, we encountered an issue when we realised that these endpoints lacked the necessary pagination information. To address this limitation, we made a formal request for the development of three additional endpoints that would incorporate the essential pagination details. Although it was technically feasible to modify the existing endpoints, we decided against it since they were already actively utilised by the UI. Instead, we opted to create distinct endpoints specifically tailored to handle pagination requirements. By doing so, we aimed to ensure a more efficient and seamless user experience while maintaining the integrity and stability of the existing system.

Here are the improved descriptions for the newly implemented API endpoints:

- GET /assets/paginated
  - Description: Retrieve a pageable subset of User Assets with pagination information.
  - Parameters:
    - None
  - Returns:
    - A subset of User Assets in a paginated format, along with pagination details such as total count, current page, and page size.
- GET /project/userList/all/paginated
  - Description: Retrieve all the projects of a User with pagination information.
  - Parameters:
    - None
  - Returns:
    - All projects associated with the User in a paginated format, including pagination details such as total count, current page, and page size.
- GET /project/{id}/assets/paginated
  - Description: Retrieve all Assets from a specific Project with pagination information.
  - Parameters:
    - id (path parameter): The unique identifier of the Project.
  - Returns:
    - All Assets belonging to the specified Project in a paginated format, along with pagination details such as total count, current page, and page size.

3   Social Analytics for Content Enriching and Performance Tracking

In D5.3 - VR Collaboration, Social Media Analytics, and Immersive Storytelling - Authoring Tools v2[2] we originally presented the Social Analytics Tool where we focused on the following three tasks:

1. The content ingestion from owned social media accounts for enriching a MediaVerse node content,
2. Content publishing in social media platforms, and
3. Performance tracking of shared content, including trending topic detection.

In this chapter, we present an overview of the modifications, enhancements, and adaptive measures undertaken regarding the Social Analytics Engine during the final phase of the MediaVerse project. These have been driven by both external changes and an ongoing internal commitment to improve the system's capabilities and performance.

Firstly, under Twitter Changes section, we address significant alterations to Twitter's API access and pricing initiated in early 2023. This necessitated a transition from the free streaming API access to a paid basic tier. The process involved modifying our backend application to align with Twitter's updated API v2 and adopting a RESTful API polling approach. This section underlines the project's capacity to adapt to external changes while ensuring the continuity of data collection.

In the Creation of PDF Reports section, we detail improvements to our report generation functionality, an essential feature of MediaVerse. The enhancements include bug fixes for stability, new features for enriched content, and a broader range of customization options. These changes are designed to improve the user experience and the utility of the generated reports.

In the Trending/Alerting section, we elaborate on modifications aimed at refining the accuracy and customization of our trend monitoring and alert system. These adjustments are part of our ongoing efforts to minimize false positives and provide more user-friendly visualizations of trend data.

In the Analysis Libraries section, we discuss the updates to our entity extraction and sentiment models. To increase the system's adaptability, we have integrated features allowing users to manually edit sentiment and named entities. The introduction of the new NER Editor application exemplifies our commitment to improving user interactivity within the MediaVerse project.

Finally, in the Infrastructure segment, we outline the measures taken to enhance the system's performance and scalability. These include setting up a new Kubernetes cluster and improving the Apache Solr server. These infrastructural enhancements are integral to ensuring that the MediaVerse project can effectively handle an increased volume of data and deliver faster, more efficient analysis.

In summary, this chapter provides a comprehensive overview of the significant developments and enhancements made to the MediaVerse project during the reporting period. Each change is part of our commitment to advancing the project's capabilities and ensuring that MediaVerse continues to provide a valuable, reliable, and user-friendly social media analytics tool.

---

[2] https://MediaVerse-project.eu/wp-content/uploads/2022/09/D5.3-V1.0.pdf

3.1   Twitter Changes

In early 2023, Twitter announced significant changes affecting their API access and pricing. More specifically, they have stopped free access and have imposed limitations in the number of retrieved posts on a monthly basis. To continue collecting content from Twitter, we have been forced to abandon the free streaming API access and move to a paid basic tier of ~100$/mo. This tier, however, does not offer stream API and instead requires RESTful API polling. Moreover, it allows a rate limit of 10K tweets/month. Twitter have also deprecated API v1, a radical change which we needed to comply with. In short, our steps involved:

- Investigating new Twitter changes.
- Creating a new account with basic-level access.
- Modifying our client backend application to consume Twitter API v2 instead of v1.
- Convert our streaming API fetching strategy to continuous polling with REST API.
- Monitor consumption to avoid rate-limit-exceeded issues.

Moreover, due to the high dependence on login-with-Twitter only option, we have switched to a username-based approach. This came together with a major redesign in the architecture of the platform, since the twitter_id has been used as a primary key in several DB tables. The login workflow has changed, affecting the user experience, considering the existing users which were facing the risk of being locked out of the platform. We have managed to make this transition smoothly without causing frustration to existing or new users. Finally, we have changed the way we store the passwords inside the database, applying encryption and salting techniques for extra security.

3.2   Creation of PDF Reports

The creation of reports is one of the most valuable features offered by the platform. We have managed to improve the functionality and make it stable by applying several bug fixes. In respect to offering new features, we have implemented the following:

- Preview of images and videos inside the report.
- Include sentiment and capability to edit before exporting.
- "Select-All" option before exporting.
- Enriched Insights section.
- Improved faceted selections.
- Customization of facets inside each aggregation.
- Option to export the PDF in low/high/medium resolution.
- Addition of a simplified trends section inside the report.

3.3   Trending/Alerting

Regarding the trending monitoring and alerting, we have worked on the improvement of the service by examining the reason for false positive alerts. Thus, we extended the rules and have also provided the option to customize the thresholds appropriately, to reduce noise and false alarms. The latter also includes setting the range (timeframe) in hours. The set of available rules is depicted in the following figure (Figure 3).

Figure 3: Set of rules to customize the alerting service.

We have also worked on improving the visualization of the trends (see Figure 4), as we have noticed that the huge amount of information was a source of frustration for users. We have cleaned up the view and provide a more simplified version of the diagrams, fitting all the necessary parts in one page in different tabs. We have also moved the alerts in a collapsible panel to allow more space for the diagrams.

As seen in Figure 4, we have enriched the functionality with the option to re-run past time slots and re-calculate the trends found inside them. This can work in conjunction with the "editing" feature for the named entities and sentiment in case the user needs to manually edit the diagrams.

Figure 4: Visualization of the trends.

A new feature has also been added to the platform. This refers to edit and save custom queries (Figure 5). This mechanism is combined with the option to receive real time alerts when there are updates (newly retrieved posts) based on the specific query. For this purpose, we have designed an entire parallel system to monitor on frequent intervals the storage for updates and send push notifications to the frontend client.
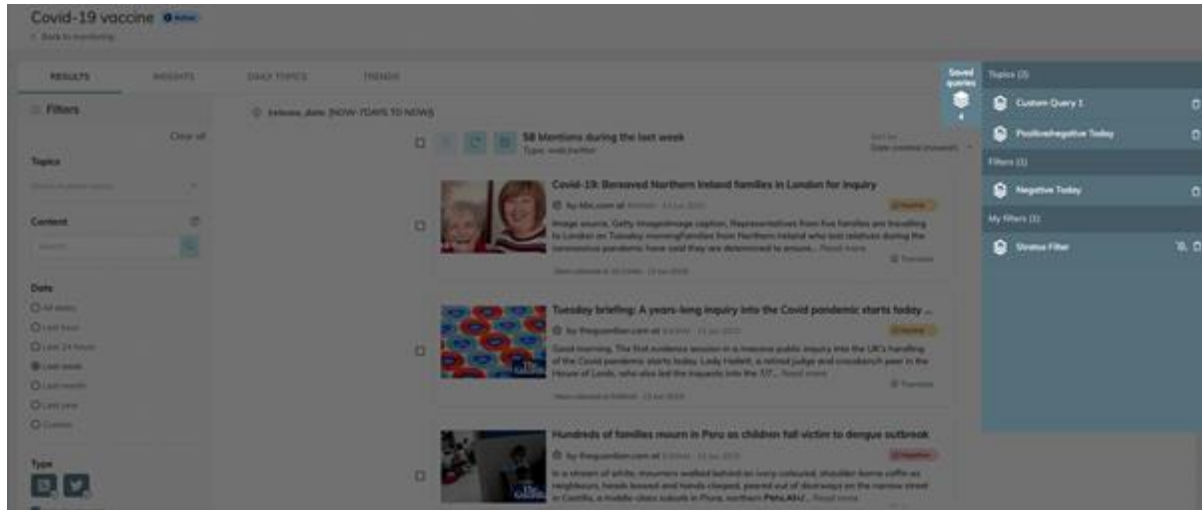
Figure 5: Mechanism for query customization.

The custom queries can be also saved as "topics", which can be combined with other facets in the left-side panel of the facets and filters (Figure 6).
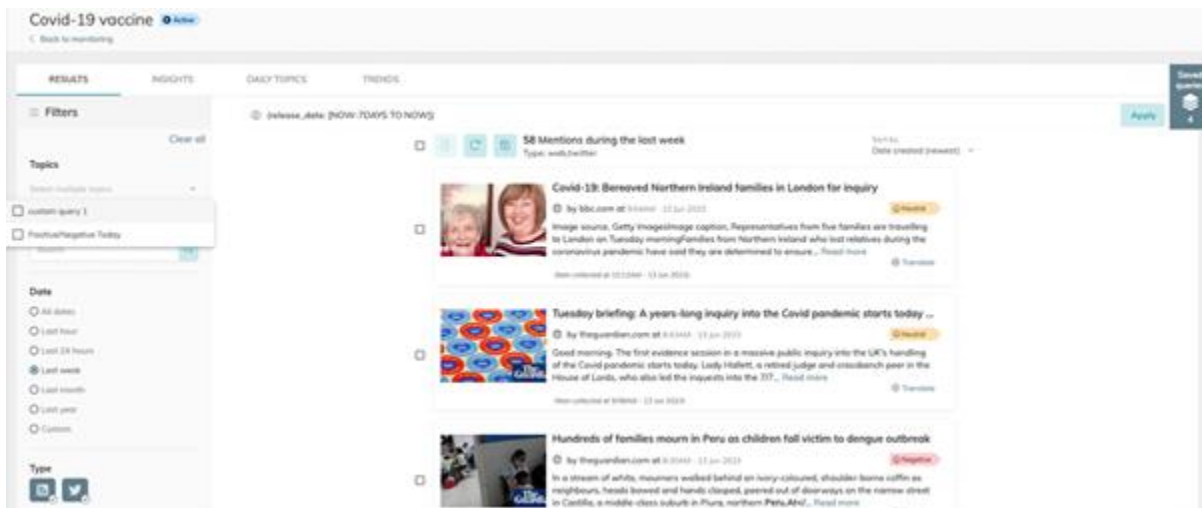
Figure 6: Custom queries combined with facets and filtering.

3.4   Analysis Libraries

We have improved the analysis pipeline by retraining our entity extraction and sentiment models. For this purpose, we have integrated a feature, which allows manual editing of sentiment directly on the interface by the

user (Figure 7). The change applied by the user is directly reflected on the user interface. This information is also stored on a database to be used for re-training the models.

Figure 7: Manual marking of sentiment.

A similar attempt has been made for editing the named entities. A whole new web application (NER Editor) has been implemented to allow users to mass correct the extracted entities (see Figure 8). This need was driven by the fact that certain entities were referring to the same person but were represented with different names, e.g., Biden, Joe Biden, and President Biden. To fix this issue, we have also tried to use the WikiData API for performing named entity disambiguation. However, we met significant obstacles and performance issues and eventually abandoned the attempt. By using the NER Editor, the user can replace all existing entities with the corrected value as well as add a rule to the NER library for future occurrences of the same entity. The set of current rules can be browsed and modified in the user interface of the NER Editor application.
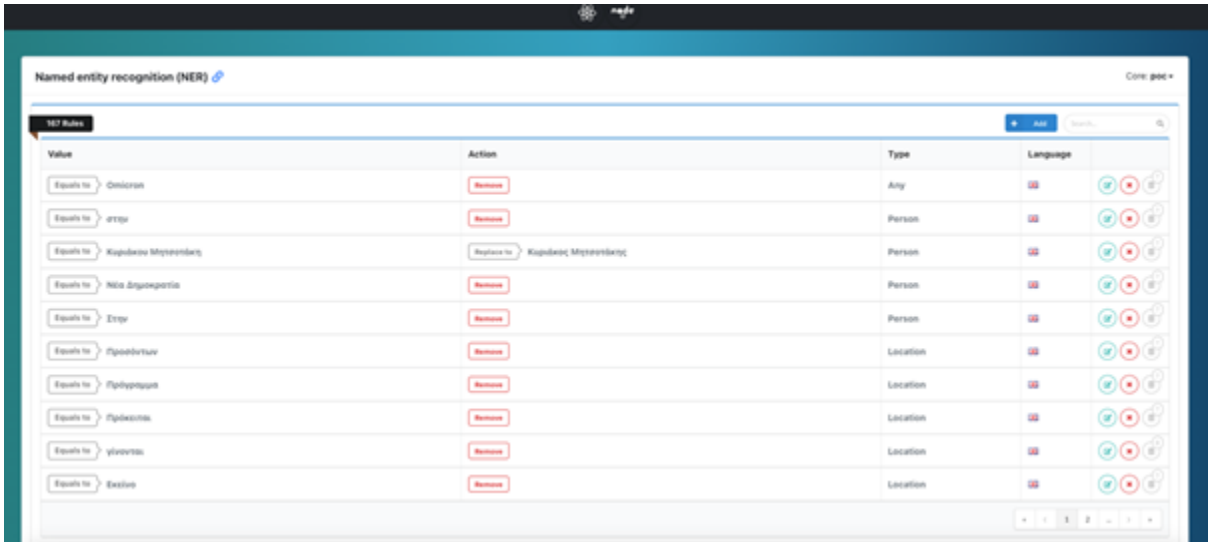
Figure 8: Manual replacement of entities.

Moreover, we have added basic rules to edit existing and future entities, as seen in Figure 9. For the matching value, the user may use: "starts with"; "includes"; "ends with". For the action, the user may select to "remove" or "replace" the specific entity with an alternative value.

Figure 9: Rules to edit existing and future entities.

3.5   Infrastructure

Due to the extremely high input rate of posts, we reached a limit with the analysis libraries and storage. This was causing severe bottlenecks and delays in the analysis pipeline. To keep up with the incoming stream the system was prone to data loss, selecting to sacrifice part of the input for avoiding lags in input data.

To improve analysis speed and performance, we have set up a new Kubernetes cluster to offer more hardware resources and better management of the containerized analysis libraries. This has been deployed on a private cloud infrastructure. A similar approach has been followed for Apache Solr server. The performance of Solr has been improved by converting the existing collections to use sharding and clustering for high-availability and faster retrieval (see Figure 10). The performance was also improved by applying small modifications to existing Solr queries, following the appropriate guidelines for query optimization.

Figure 10: Graph representation of collection sharding.

Finally, we have added a health-monitoring service by host-tracker, checking on 5-min intervals the uptime of Solr Server. For this purpose, we have created a health check endpoint to our backend to return a query result from Solr. This is then consumed by host-tracker to ensure that the search engine is up and running properly.

4   VROdos for Authoring VR Experiences Based on 3D Geometries

In the previous related deliverable, namely D5.3 - VR Collaboration, Social Media Analytics, and Immersive Storytelling - Authoring Tools v2[3], we presented VROdos, a web platform that allows users to create multi-player WebVR experiences through a web-based 3D authoring tool, as shown in Figure 11.
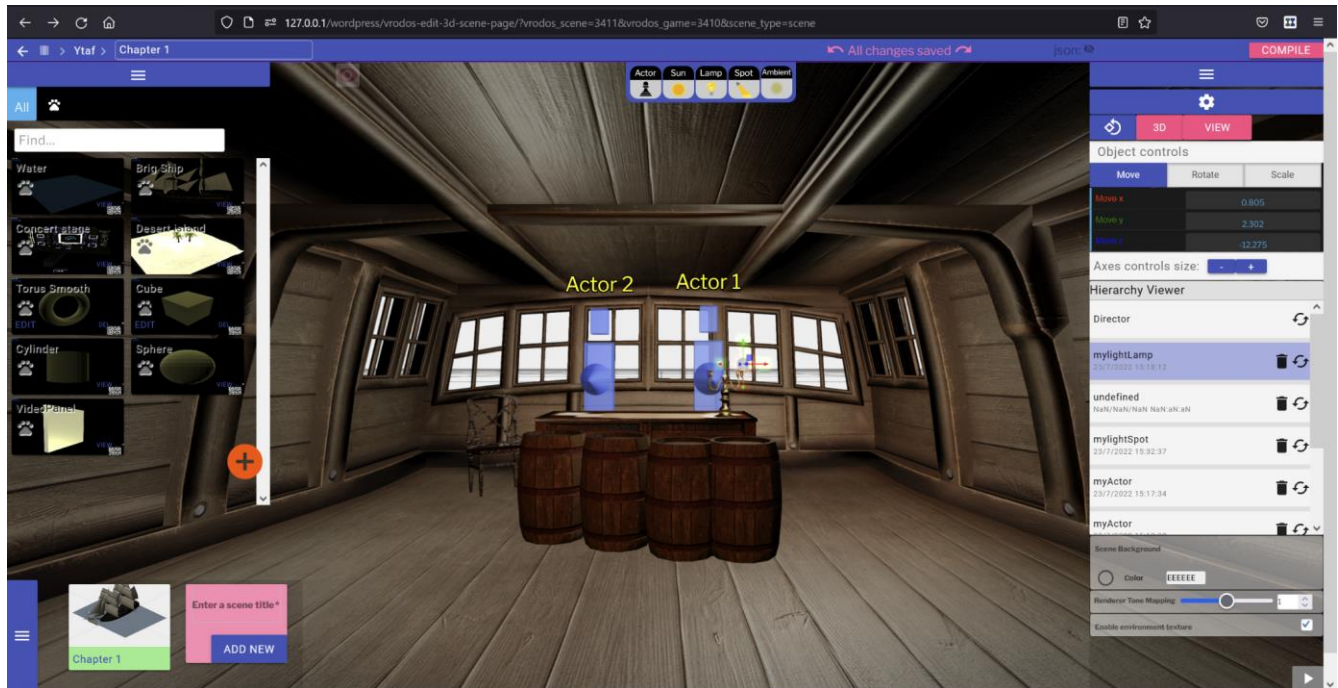
Figure 11: VROdos authoring tool.

The final experience was a virtual production stream, as shown in Figure 12. It is a virtual production scene that allows positioning user streams inside a virtual environment. Proper interfaces on the right-side allow configuring the parameters for green screen removal as the users should have a green background.

Figure 12: VROdos developments: virtual production stream and user experience targeting high-end PCs.

Figure 13 shows the overall architecture of the application. The proposed system is divided into four logical phases:

1. Setup,
2. Authoring,
3. Playing, and
4. Exploitation.
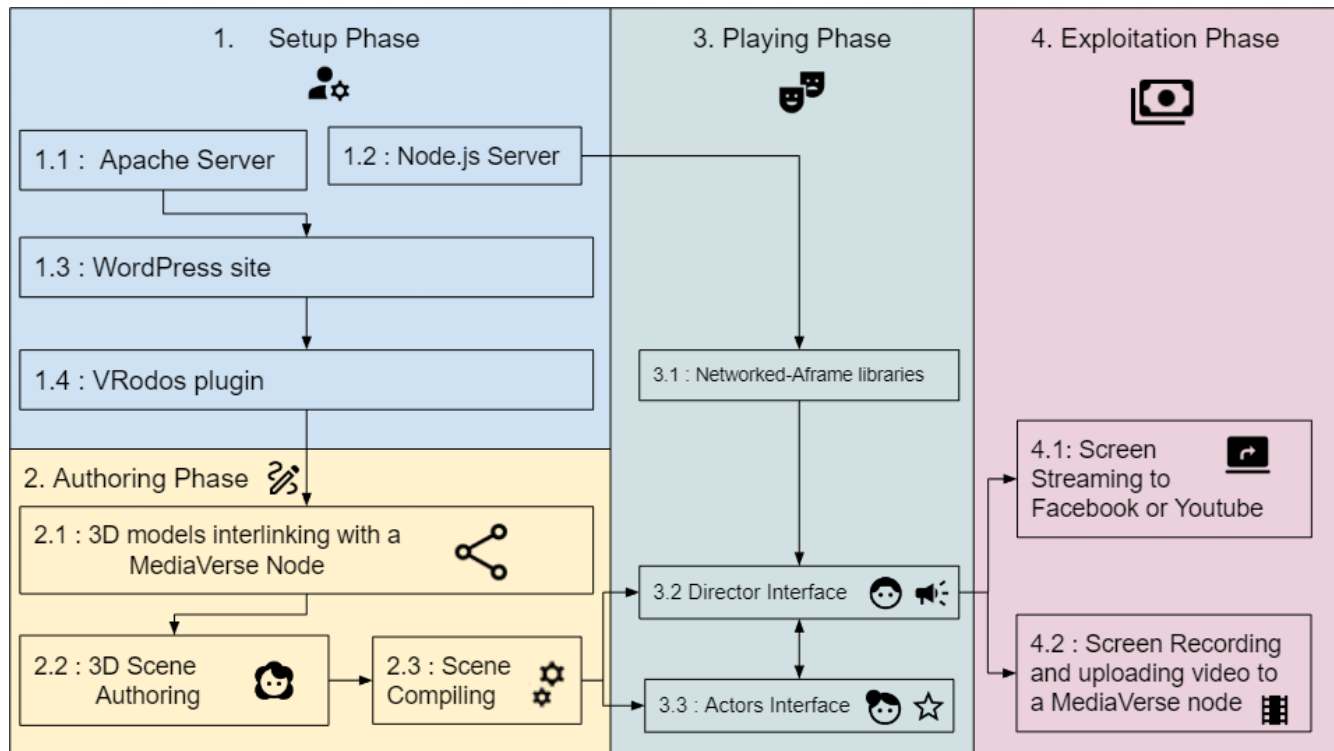
Figure 13: Design of the system as from previous deliverable (D5.3).

Overall, we have devised a custom web-based streaming kernel that performs rendering at a high-end PC in one of the peers participating in the production (called the Director), and streams back the rendered scene stream to the peers' smartphones that we refer to as Actors. The result is a virtual production movie or a live movie stream. On this basis, it can be either recorded in the Director's PC and then uploaded to a MediaVerse node, or it can be streamed live to social media such as Facebook or YouTube.

In total, four types of users are involved, namely:

1. the System Administrator who is responsible for the set up and maintenance of the system;
2. the Graphics Designer who is responsible for uploading the 3D models and setting up the 3D scenes;
3. the Director who is responsible for the coordination of the actors as well as for the rendering of the scene in a high-end PC; and
4. the Actor who participates through a smartphone.

The whole platform is completely web browser based. There is no desktop binary application used. The web libraries used are the state-of-the-art libraries, namely Three.js[4], Aframe[5], and Networked-Aframe (NAF)[6] that allow to construct virtual reality experiences for any kind of device that has a web browser, namely smartphones, tablets, laptops and PCs.

In the following sections, we describe the improvements made on the aforementioned platform, the testing and the evaluation by end users. In Section 4.1, we present the developments in the third year. In Section 4.2, we provide the piloting details for the authoring and playing parts. In Section 4.3, we provide details about the evaluation results of the tools. Finally, conclusions are drawn in Section 4.4.

4.1   Summary of Development

The developments regard technical improvements in the authoring and playing part of VROdos as well as the setup of the experiments for piloting the application. These are explained in the following subsections.

4.1.1   Additional Features

The new features are the support of fog for increasing realism and the support of masking for changing the z-order. A fog configurator for improving the realism of certain environments was made. Figure 14 presents the authoring interface for the fog configuration, while Figure 15 shows the environment with fog. Two types of fog are supported such as linear that has a near and a far distance limit, vs exponential fog that has an exponential density factor. The fog can be also coloured.
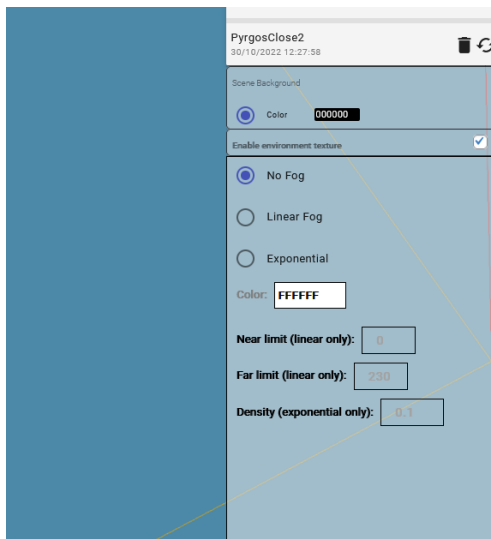
Figure 14: Authoring interface for fog configuration.

Figure 15: Environment with fog.

Masking was also developed for changing the z-order, where z-order means which object is rendered first or second or third, etc. The authoring tool is used to place the mask on the problematic rendering topology. For example, a mask was placed in the stairs in order to reverse the z-order of the water with the ship (Figure 16).

---

[4] Three.js, https://threejs.org/

[5] Aframe, https://aframe.io/

[6] Networked-Aframe, https://www.npmjs.com/package/networked-aframe

Thus, the water is not visible when looking inside the hatch (Figure 17). A mask object is just a plane generated by a design software and uploaded with the name "Mask".
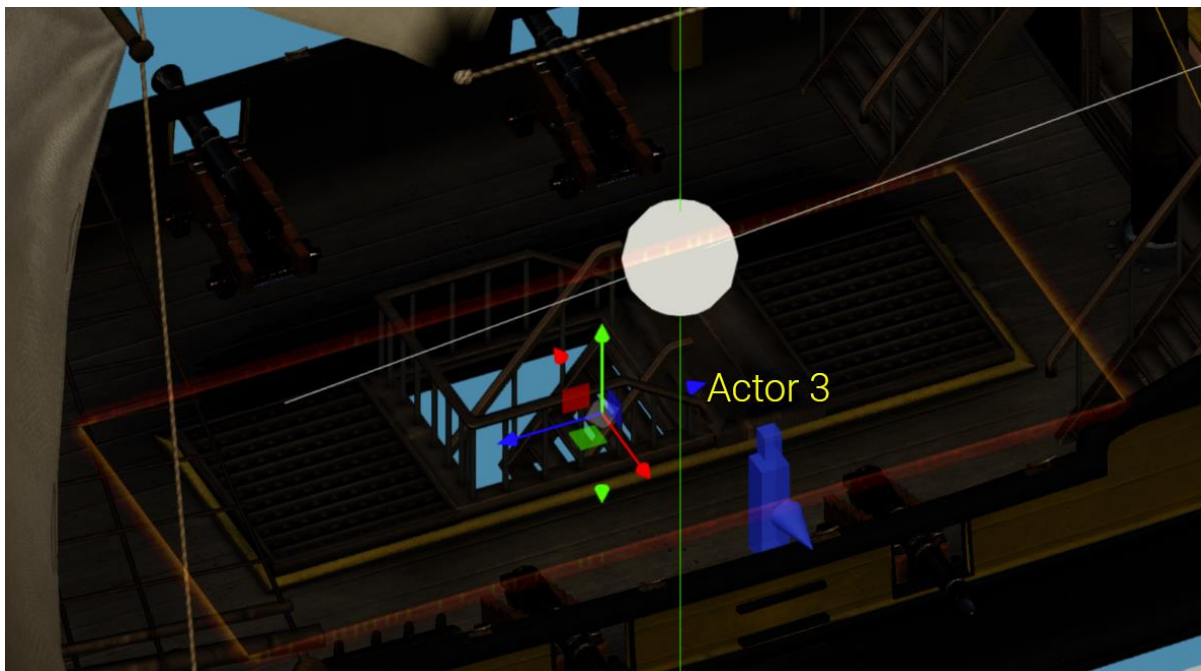
Figure 16: Placing a plane as a mask to change the z-buffer order of the sea against the ship.
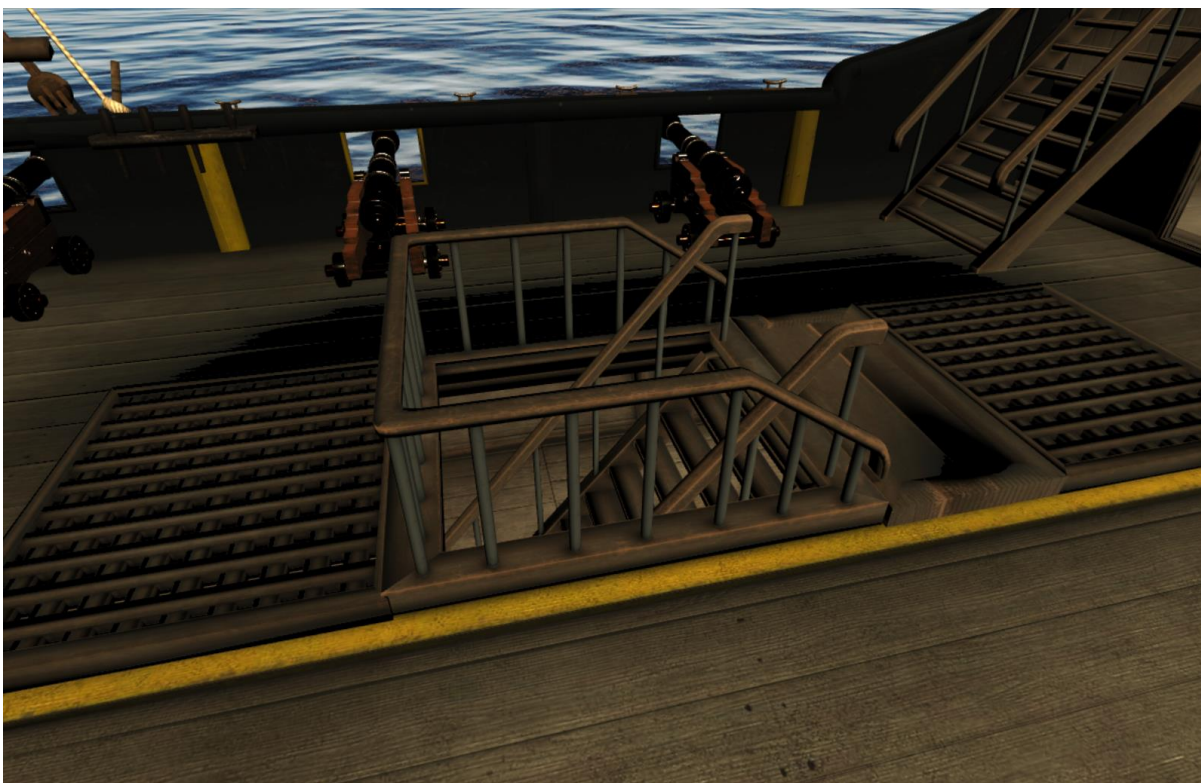
Figure 17: The sea does not penetrate the boat when rendering the final scene in Virtual Production mode.

4.1.2   Content Variety

More assets and scenes were constructed for increasing variety. The scenes include the Rebellions' Ship scene as shown in Figure 18 and Figure 19, and the Lava Rock scene as shown in Figure 20. In Figure 19, the cabin of the ship was enriched with historical photos (stemming from 1918 up to 1945) related to the Thermi municipality, Macedonia, Greece. The images were kindly provided by the cultural centre of Thermi and consist of a part of the newly published encyclopaedia for the history of Thermi. The second scene, as shown in Figure 20, regards theatrical plays that take place in a hypothetical hell, used by the local dramatic school of the Cultural Centre of Thermi.
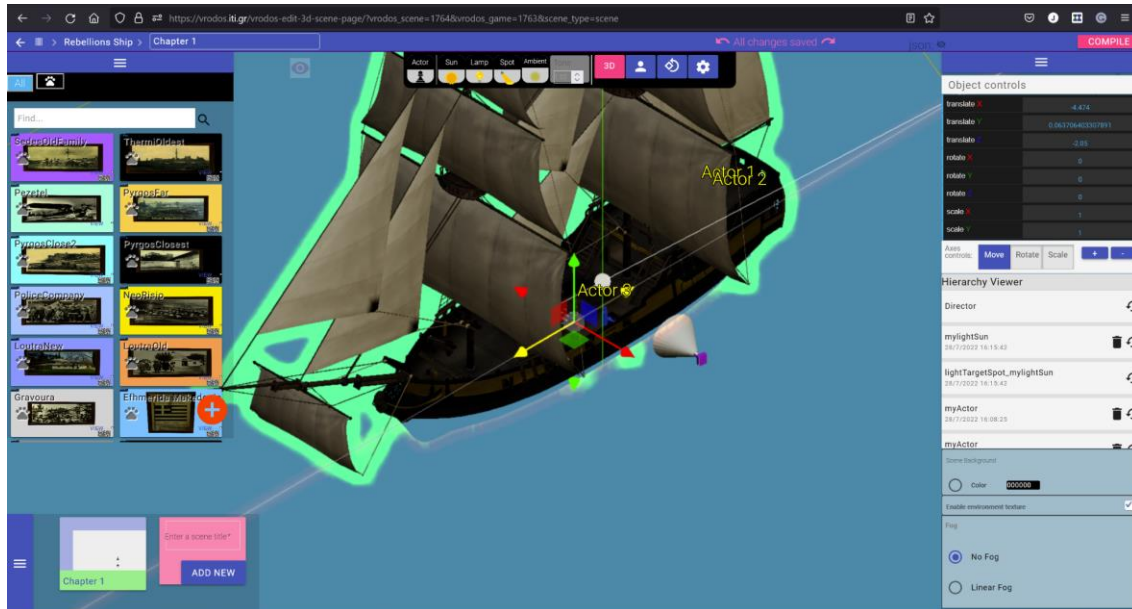
Figure 18: The rebellions' ship for cultural and historic themed virtual productions.
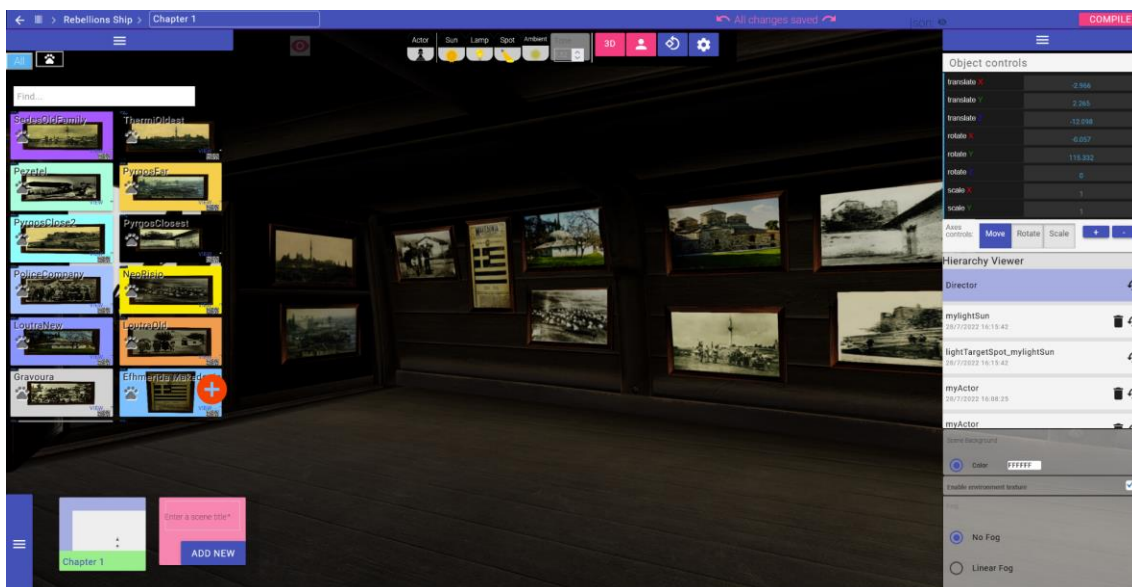
Figure 19: Enriching the rebellions' ship cabin with historical photos.

Figure 20: The second scene is the Lava rock scene for actors to perform dramatic acts.

4.1.3 Texture Compression

All 3D assets were resaved using Blender 3D with compressed textures, as JPEG instead of PNG format. These textures regarded diffusion, normal, metal-ness, and roughness maps. Indicatively, the size of the 3D model of the ship was reduced from 130 to 30 MBs without visible quality reduction. This allowed the scenes to load quicker and be rendered faster. This significantly improved the value of the proposed system as it has been made completely portable and can be run now on laptops with NVidia 3060 RTX graphics card or higher.

4.1.4 Meet Public Internet Connectivity Standards

The environment was configured for standard Internet connections such as 50 Mbps. The streams of players were downgraded to 800x600 pixels to maintain speed. The quality was adaptive to current Internet speed as 50 Mbps connections do not have a constant speed and fluctuate between 10 and 30Mbps in most of the cases.

4.1.5 Dealing with ICE Servers Issues

The use of Interactive Connectivity Establishment[7] (ICE) server for signalling users' IPs in peer-to-peer connections was better configured. The ICE servers are responsible for the initial interconnection of the peer users. They provide the IPs to each other to ensure that streams are well propagated. They do not intervene in data pipelining afterwards. Two kinds of ICE servers exist, namely STUN and TURN servers[8]. The STUN servers are easily found[9] and they work well for laptops and desktops. However, it was found that they do not work for smartphones in public networks with firewalls. Therefore, we used a TURN server that can override firewalls and

---

[7] Wikipedia: ICE Server: https://en.wikipedia.org/wiki/Interactive_Connectivity_Establishment
[8] Stackoverflow: ICE, STUN, and TURN servers: https://stackoverflow.com/questions/59484802/ice-vs-stun-vs-turn
[9] Github, list of available STUN servers: https://gist.github.com/mondain/b0ec1cf5f60ae726202e

work for smartphones. The Openrelay Turn server was used[10] as it was the only available open source TURN server. Other commercial TURN servers could also be used[11]. The use of port 443 was necessary as it is a prerequisite of the video and audio stream to be recorded from the web browser.

```javascript
var myPeerConnection = new RTCPeerConnection({
 iceServers: [
  {
    urls: "turn:openrelay.metered.ca:80",
    username: "openrelayproject",
    credential: "openrelayproject",
  },
  {
    urls: "turn:openrelay.metered.ca:443",
    username: "openrelayproject",
    credential: "openrelayproject",
  },
 ],
});
```

Figure 21: The final setup of the ICE servers for the Networked-Aframe.

4.1.6   Enhancing the Experience with a Projector

We aimed to increase the availability of the final result to physical audiences by the use of a projector. A projector was used to the pipeline so that an audience in the room can experience the result. Figure 22 shows the setup of the experiment. Left is the audience that inspects both the projected live image as well as the green screen stage. At the centre is a laptop that receives the actor's stream from the smartphone, performs the 3D rendering, projects the virtual environment on a white screen, and streams the whole video to Facebook live. On the right-side, there are the journalists - actors performing on a green stage. Figure 23 illustrates the setup.

---

[10] Free WebRTC TURN server: https://www.metered.ca/tools/openrelay/
[11] Xirsys TURN server: https://xirsys.com/

Figure 22: Panoramic Image of the setup.
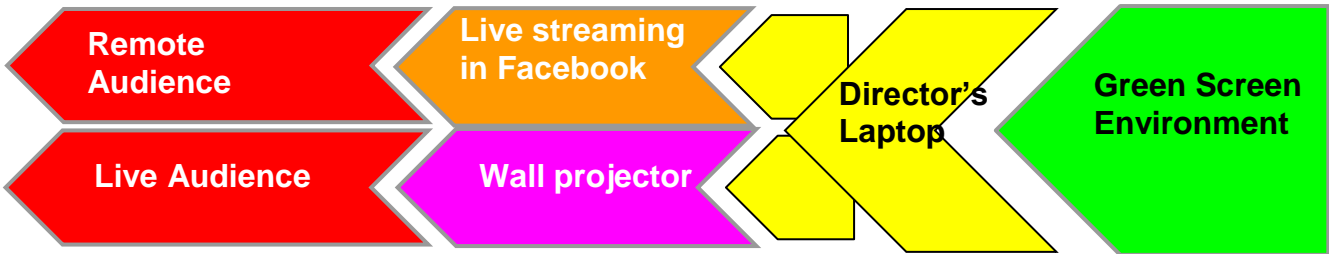
Figure 23: VROdos Setup.

The final design of VROdos has been transformed, as shown in Figure 24. The difference is that the Promotion – Monetization Phase (in the right part of Figure 24), has been improved by adding the "Projecting to a whiteboard" experience.
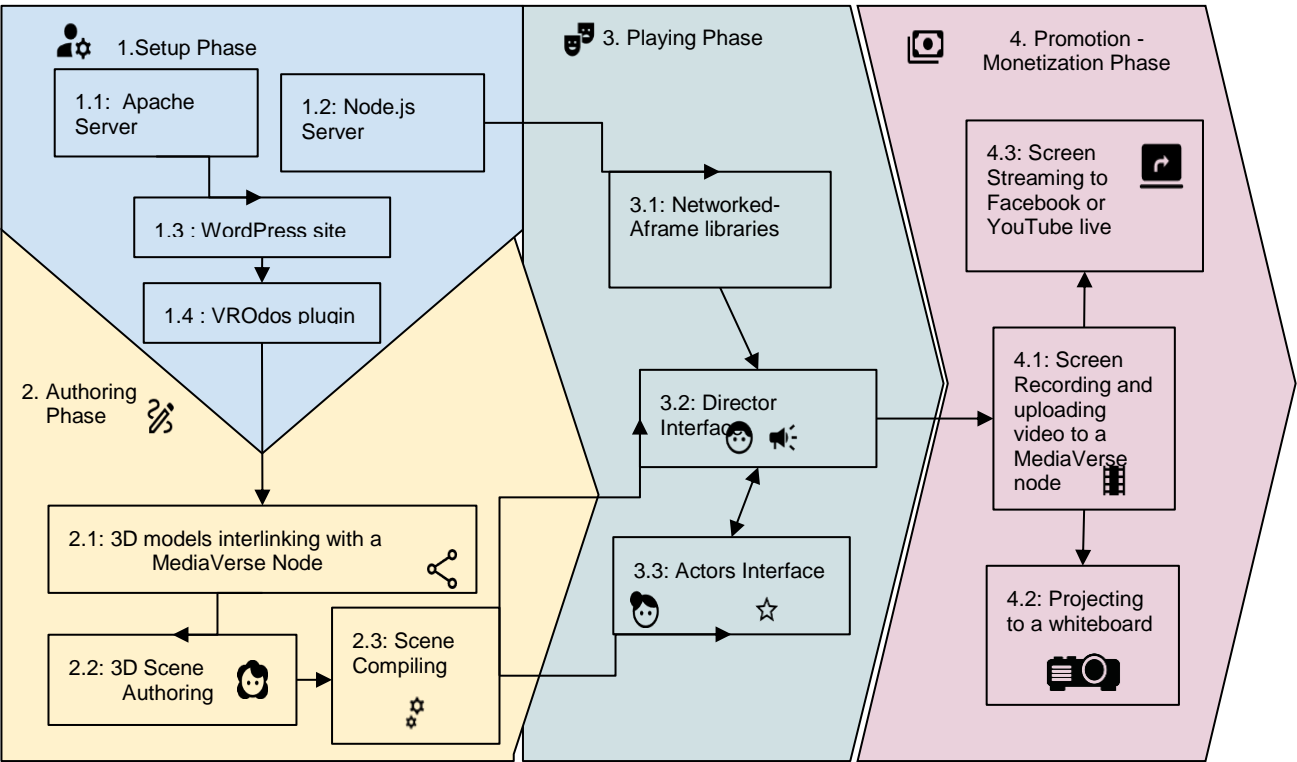
Figure 24: The final design of the VROdos platform. Node 4.2 was added to improve the experience.

4.1.7   Final Equipment Requirements Definition

The final equipment specifications are described below. The overall cost of the end-user's equipment does not exceed 2000 EUR. All actors can participate through their smartphones and using a Creator Studio Kit with a tripod and a green screen. The final requirements and the typical equipment used are described in Table 1. The final setup is shown diagrammatically together with some photos of the equipment in Figure 25.

Table 1: Hardware - Software requirements.

#	TYPE	SPECIFICATION	COST
1	Laptop	Software: Windows 10 or 11, Chrome Browser Hardware: A middle-end laptop is needed such as G15 5510, NVidia 3060 GPU or higher	1500 EUR
2	Smartphone(s)	Software: Chrome or Firefox browser. Hardware: A low-end phone with camera and microphone. Xiaomi POCO X3 with Android 10 was used.	200 EUR
3	Studio	Green Screen 4x4 metres (blue can also be used) Tripod with light for smartphones.	70 EUR
4	Projector (Optional)	HDMI connection Low end with 90 EUR can be used 800 x 480 or higher resolution	90 EUR (Optional)
5	Battery unit (Optional)	A 300 MAh battery unit can be used to avoid many cables in the area. Supply power for project (12V, 20 Watt) Supply power to PC (220V, 120 Watt) Cooling fan for PC (5V, 5 Watt)	300 EUR (Optional)
6	Server	Software: Ubuntu 21, Apache 2.0, Node.js 17, Mysql, WordPress 6.1, VROdos plugin. Hardware: No GPU is needed. Typical setup.	1000 EUR (provided by VROdos at CERTH)

Servers	Low-end server for VROdos Apache 2.0 server Node.js server 17 WordPress 6.1 No GPU is needed.	TURN server (openrelay: https://www.metered.ca/tools/open relay/)	MediaVerse node server

Internet connection (50Mbps), Wifi router

End-User Equipment	Laptop DELL G15 5510 NVidia 3060 RTX 16 GB RAM Intel CPU i7 10870HCooling fan of 20 EUR Total Cost: 1400	Smartphone Low end smartphone (Xiaomi POCO and Xiaomi Note 5 were used successfully) Cost 200 EUR	MyStudio Tripod for smartphone and Green screens Creator studio Cost: 70 EUR
	Optional: Projector Low end PVO YG-300 Pro Mini Projector: 800 x 480 pixels Cost 90 EUR	Optional: Battery unit Make the system completely portable without the need of external supply Cost: 300 EUR All powers 288Ah	

Figure 25: Hardware setup

4.2 Updates on Integration with MV Node(s)

The VROdos authoring tool uses MediaVerse user credentials to connect to a node, get an authorization token, then import projects and assets that are associated with the user in a MediaVerse node.

The collaborative session that VROdos offers can be recorded as a video file and then uploaded as a project outcome to the registered MediaVerse node.

4.2.1 User and Identity Management

The basic step for the integration of VROdos with the MediaVerse node is authentication with the same credentials. The integration of the authentication scheme between the VROdos tool and an MV node relies on the development of a custom plugin in the VROdos web platform that handles the communication between the two components, as presented in Figure 26.
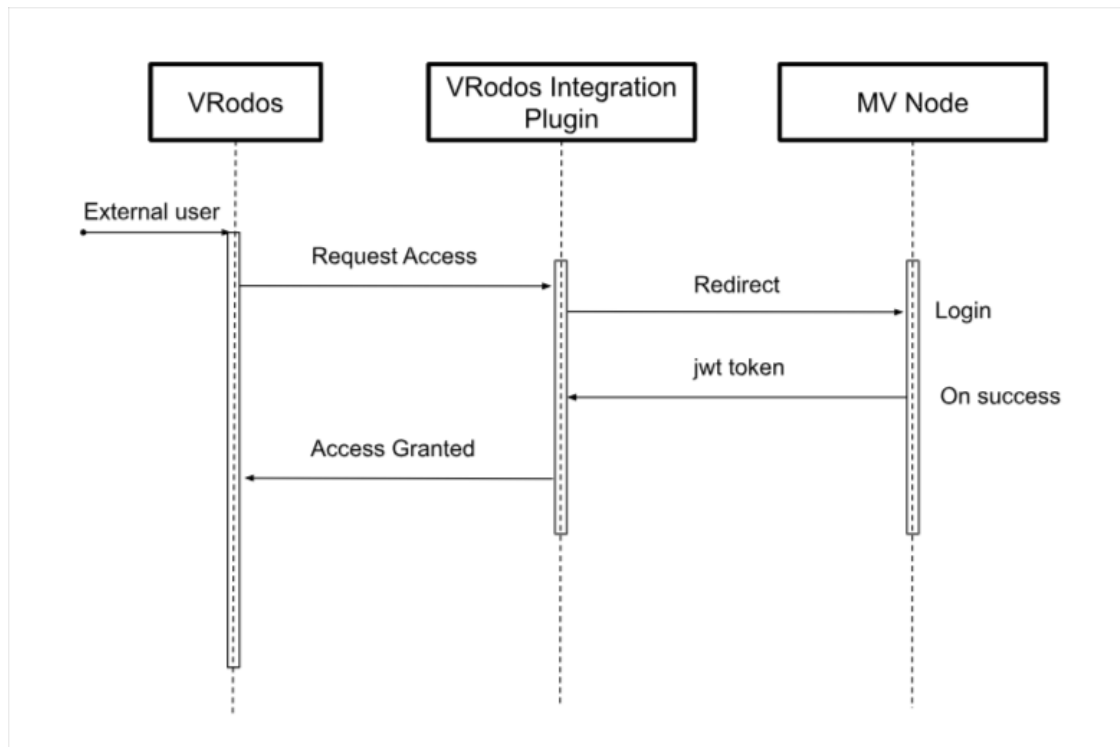
Figure 26: Sequence diagram of the authentication scheme between VROdos tool & an MV node.

This procedure consists of the following steps:

1. A user is registered in the MV Node.
2. By clicking on the appropriate VROdos button from inside the MediaVerse environment, the user is transported to the login page of VROdos.
3. User logs in to VROdos using MV Credentials, requesting access to said Node. The MV node information is saved automatically to VROdos.
4. On success, a user receives a valid JWT (JSON Web token) from the MV Node which is used from the VROdos plugin to allow the user access to the MediaVerse assets and projects.

4.2.2   Access Media from MV

After successfully logging in VROdos, the user has the option to import all project and asset data from a MediaVerse node. By clicking on a button from the login screen the import process runs in the background. VROdos, being a creation tool for collaborative 3D content, allows only 3D files to be imported as assets and more specifically in the .GLB (gltf) format. All files that are not of GLB type are ignored during the import process.

In essence, each project from MediaVerse is copied to VROdos, along with all assets that are available in each project, for users to use them in a user collaborative session. If a MediaVerse project exists in VROdos then the specific project is not imported from the MV node. Note that if a user changes the available assets inside a MediaVerse project, the changes are not mirrored in VROdos. The correct way to handle this would be to delete the project from VROdos, then rerun the import process.

4.2.3   Use of MV Support Functions

As previously mentioned, VROdos uses MV support functions to import project data from an MV node and export a video file to an MV node. More specifically VROdos:

- Uses the authentication API of a MediaVerse node to login a user and get a token back.
- Imports all projects and the assets of each project using MediaVerse API calls.
- Uses the MediaVerse API to upload a video file to a MV project and set that video as the project outcome.

4.2.4   Store Media into MV

VROdos supports session recording, while users are live in the collaborative environment. By using the controls at the bottom center as seen in Figure 27, the director can record a video of what the browser tab is showing, save it to their computer or click on the upload button for the resulting video to automatically be uploaded to the MediaVerse node the user is registered to. The video is uploaded as a project outcome file for the specific project the user is in while in the collaborative environment.

Figure 27: Users have access to the outcome media from inside their Project in the MediaVerse node.

4.3   Piloting

The testing of the platform has been conducted in the context of two piloting phases:

1. The Authoring Phase, in which the users-stakeholders design the virtual production environment; and
2. The Experiencing Phase, in which the environment is offering the final product to the end-users, i.e., a real-time virtual production movie.

More information about the piloting activities with VROdos and the evaluation results will be available in the forthcoming D7.3 - Final large-scale pilots evaluation report.

4.4 Requirements Update, Conclusion, Outlook, and Exploitation

4.4.1 Requirements Update

XR-02 / A 3D Scene can be Exported as a Video Clip.

A mechanism to record the contents of a browser tab has been implemented inside the VROdos as core functionality. While the collaborative session is in progress, the Director can record a video of what is happening in the environment, and subsequently can download that video or one-click upload it to a MediaVerse project as a project outcome.

XR-03 / Import 3D Objects.

We have made available an import mechanism for 3D objects using the VROdos plugin, for binary glTF (.glb) files. glTF is an open standard developed and maintained by the Khronos Group[12]. It supports 3D model geometry, appearance, scene graph hierarchy, and animation. It is a streamlined, interoperable format for the delivery of 3D assets that minimises file size and runtime processing by apps.

XR-04 / Decorate VR Walls.

This requirement is now deprecated to simplify the process of creating a scene, as users can upload any 3D objects they want, to decorate a scene.

XR-07 / User Presence in VR (Avatar) & XR-08 / User Presence in VR (Webcam).

Both in 3D and VR environments, user presence is shown only for the Actors by using their camera stream as input instead of a 3D avatar. The support of a 3D avatar instead of a camera stream is planned for a future iteration of the development of the plugin, as it is not a requirement for the collaborative aspect of the platform.

XR-08 / User Presence in VR (Webcam).

Both in 3D and VR environments, user presence is available for the Actors by using their camera stream as input instead of a 3D avatar. This way, a viewer of the experience that is navigating the VR space can be fully immersed in the environment, perceiving the actors as real persons.

XR-09 / Interfaces for VR Devices.

Throughout the implementation of the VROdos plugin, VR support has been of great importance. This is why a-frame[13] has been selected as the primary web framework for XR experiences. The experience creation flow has been implemented in a way that after users edit a 3D scene, it is compiled for WebXR, automatically supporting VR devices.

XR-10 / VR Live Audience.

Each collaborative scene that has been built from VROdos can be accessed from a browser link and has native support for VR mode using the a-frame framework. In essence a user with a link can access the content in 3D or VR mode.

---

[12] https://en.wikipedia.org/wiki/Khronos_Group
[13] https://aframe.io/

AUTH-03 / Connected External Software.

VROdos is considered an external software of MediaVerse. In section 4.2 we described the integration updates between VROdos and the MediaVerse platform. Summarising, VROdos authenticates users through VROdos, fetches project and asset content, and stores the derivative project outcome in MV node for publishing.

4.4.2   Outlook

Our primary focus for future development is to prioritise performance improvements and bug fixes, to minimise system memory consumption when loading multiple large 3D files. We are also actively working on adding support for more file types in the 3D editor to supercharge the collaborative content creation process further. Additionally, user interface tweaks will be implemented to enhance the overall user experience and make the task of using VROdos for creating experiences as enjoyable as the creative collaborative aspect of using them.

4.4.3   Exploitation

The VROdos WordPress plugin as well the VROdos MV Integration plugin are both open source and available on GitHub under Apache-2.0 license.

- **VROdos Software:** https://github.com/VROdos/VROdos
- **VROdos MV Integration plugin:** https://github.com/VROdos/mvnode_integration_plugin

By having the VROdos software as an open-source WordPress plugin, anyone can install it very easily on any WordPress instance and start building 3D environments quickly, while adding new functionality by forking the GitHub repository.

CERTH, being a research centre, is planning to use VROdos in future research projects, further expanding its capabilities and supercharging it into a more powerful creation tool.

4.4.4   Conclusions and Exploitation

The provided methodology for real-time virtual productions is of high value for educational, promotional, and entertainment goals. The proposed methodology exploits open-source web technologies to build interfaces, structures, and procedures that allow non-developers to be authors. It is the first time that web technologies for 3D rendering and virtual production have been used in real-time and with low-budget equipment. Of course, the quality is not comparable with high-budget movies, but the quality is sufficient for school environments, libraries, and local cultural centres. Furthermore, it can be significantly improved by making the following changes:

1. Having both Director and Actor devices connected to the same WiFi deteriorates the frame rate. To improve the frame rate, the Director and the Actor should access the Internet from separate WiFi routers or from a single high-speed connection line. This will increase the audio and video quality per actor as the compression algorithms will work less.
2. Increase the transmitted video stream resolution per actor from 800 x 600 pixels to 1920 x 1080 pixels. Although the smartphone camera had a definition of 1080p, only 600p was finally used because it was not efficient to transmit in real-time a HD stream in low bandwidth line - this is related also to what is described in step 1. By the use of HD video, the background will be removed more accurately, and also the actor in focus will have better resolution.

3. The use of a desktop PC with a good graphics card, e.g., 3080 RTX GPU (instead of a laptop that was used in the tests) will significantly increase rendering speed and will allow for more complex 3D environments.
4. Use a better video projector as the one used with 800x480 resolution is deteriorating the quality. 1920 x 1080 with 2000 lumens will be ideal.

The features offered by the platform as regards the pluralism of 3D graphics can be further expanded in order to increase realism. For example:

- Clouds and sky can be improved using these libraries:
    - https://github.com/Dante83/A-Starry-Sky
    - https://mrdoob.com/#/131/clouds
- Grass terrain can be improved using the following library:
    - https://github.com/jeromeetienne/threex.grass
- Custom terrain can be made using this library:
    - https://github.com/FarazzShaikh/Terrain-Builder

Overall, as an experience, the proposed system was evaluated well by the stakeholders and the viewers. As compared to our previous evaluation in 2019-2020 (Ververidis et al., 2021), we have significantly improved the user-friendliness and usability of the system by:

1. Improving the user interfaces;
2. Providing an undo-redo mechanism;
3. Improving the compiling speed of the final experience using A-frame instead of Unity3D. The compiling speed improvement has been reduced from 10 minutes to 2 seconds by changing the output format, namely, instead of compiling into Unity3D which is in C# language, we have developed a methodology to compile into Aframe, which is a lightweight web format (see D5.3). This made the platform much more flexible as the authors can preview the result in any phase of the authoring process;
4. By providing the means for stakeholders, such as actors to participate in real-time inside the experience through a green screen environment. By providing the capability of the stakeholders to participate in the experience through their smartphone cameras, we have increased the added value of the platform as it hooks to existing cultural scenarios such as dancing, theatrical playing, a book presenting, Interviewing VIPs, etc. This made the platform more attractive and well-accepted by the users.

The lesson received from this work is that the most important step for successful software is the requirements engineering phase which runs throughout 70% of the project. Most of the projects do not try to follow the needs of the users, live with them, but try to pose their own preferences either for simplifying the software development process or for following strict rules posed several years ago. We have tried to cope with the needs of the users by continuously receiving feedback during the developments. We have changed 2-3 times the final product vision in the first 2 years of the project, e.g., by removing the requirement of VR glasses that none has or wants to buy and trying to appeal to low-budget equipment. We have provided the means for the users to participate with a smartphone and remotely receive the rendered environment as a video stream. Now, even with low-end smartphones democratisation in participation is achievable. Also, we have provided a way for the stakeholders to be protagonists inside the virtual environment using their smartphone camera video stream, but not replaced with 3D avatars which makes them anonymous. In this manner, the resulting solution does not replace cultural stakeholders but surrounds them with a virtual space that is impossible to construct physically.

5   Fader360 Immersive Storytelling Toolset – Fader360

Fader360 is a pioneering web-based platform focused on immersive storytelling. It leverages a combination of interactive hotspots and 360° content to provide users with a unique authoring tool to create virtual reality experiences. Ideally, Fader360 is suited to aspiring journalists, filmmakers, and individuals seeking to share their narratives in a novel way.

Over the past year, Fader360 has undergone significant enhancements, primarily through integration with MediaVerse. This integration resulted in the expansion of Fader360's storytelling capabilities by linking them to the vast media resources and potential offered by MediaVerse, enriching the platform's ability to engage audiences globally in novel ways.

This strategic collaboration represents a considerable shift in the narrative landscape, pushing the boundaries of virtual reality storytelling. It has not only added new dimensions to the storytelling experience but has also significantly improved the quality of immersive experiences. By integrating with MediaVerse, Fader360 now provides its users with an extensive collection of media resources and tools, enabling them to create narratives that deeply resonate with audiences worldwide.

As Fader360 continues to refine and enhance its capabilities, it is progressively shaping the future of virtual reality storytelling. By positioning itself at the intersection of cutting-edge technology and narrative creation, it remains a driving force in the evolution of the industry.

5.1   Summary of Development

The latest upgrade of Fader360 signifies a considerable advancement, as it involves an extensive restructuring and optimization of the entire application. This upgrade covers both the frontend and backend, intending to address accumulated user feedback over the years while emphasizing simplicity and enhancing user experience.

The recent upgrades to Fader360 underscore a commitment to providing users with a more efficient storytelling platform. By acknowledging user feedback and utilizing cutting-edge technologies, Fader360 has made notable progress in simplifying the user experience and integrating smoothly with MediaVerse. This upgrade introduces new opportunities for storytellers, enabling them to craft immersive narratives effortlessly, utilize the benefits of MediaVerse, and bring their stories to life in remarkable ways.

5.1.1   Continuous Upgrade of Fader360 Backend

Simultaneously, Fader360's backend has been entirely reengineered to ensure complete integration with the MediaVerse platform. This integration yields significant benefits, as it provides users with seamless access to MediaVerse assets and projects. As a result, storytellers can easily import MediaVerse content into Fader360, initiating their story creation process with an array of rich media resources readily available. This incorporation not only extends the potential for storytelling but also optimizes the workflow by eliminating manual transfers or conversions. Section 5.2 describes in detail all the activities and features developed during this reporting period in relation to the integration of Fader360 into MediaVerse.

5.1.2   Complete Redesign of Fader360 Player and Editor

For the frontend, both the Editor and Player components have seen substantial improvements using state-of-the-art software stack. These updates modernize the application and incorporate user-friendly workflows that are straightforward to understand and navigate. The user experience within Fader360 has been thoughtfully redesigned to offer an intuitive and seamless journey, permitting storytellers to concentrate on their creative process without unnecessary complications.

The frontend stack was defined and deployed using React/JS[14], which forms the core of our application. We chose to display Fader360 Stories using the combination of React-Three-Fiber[15] and React-Three-Drei[16], the latter being a helpful additional component set and a wrapper around ThreeJS[17]. This combination provides a powerful and flexible platform for creating and displaying 3D content within the React framework. Styling within the application was managed using TailwindCSS[18], a highly customizable, low-level CSS framework. This was employed for both in-view HTML components and purely 2D HTML elements, ensuring a consistent and visually appealing user interface. We also incorporated Axios[19] for fetching backend data, providing a simple and efficient way to make HTTP requests from both the browser and Node.js[20] environment.

Another integral aspect of our development effort was the integration of Hls.js[21] to facilitate the streaming of Audio/Video content. This powerful JavaScript library enables users to play HLS streams directly in their browsers without the need for any plugins. We have also implemented a wide range of features to facilitate a more intuitive and user-friendly experience.

Spatial Background

One of the standout features of Fader360's front-end development is the ability to incorporate 360-degree backgrounds, for both video and audio. This innovative feature is a critical part of creating immersive experiences within the platform, enabling users to set visually rich and immersive backdrops that fully encase the viewer in the story (see Figure 28). But the versatility of this feature does not stop at scene backgrounds predetermined in the backend. Users can also harness any image or video asset available for their story, allowing them to tailor their narratives with a wide array of unique and engaging visual and auditory settings. This dynamic utilization of media not only expands the creative canvas for storytellers but also significantly enriches the sensory engagement of the audience, immersing them in a narrative that surrounds them in a full 360-degree spectacle.

---

[14] https://reactjs.org/: React is a JavaScript library for building user interfaces. It allows developers to create large web applications that can update and render efficiently in response to data changes.

[15] https://github.com/pmndrs/react-three-fiber: React Three Fiber is a React renderer for three.js on the web and react-native. It's a reconciler for three.js, enabling developers to define declarative 3D content for the web.

[16] https://github.com/pmndrs/drei: React Three Drei is a collection of reusable components for react-three-fiber. It helps speed up development by providing useful abstractions and helper functions.

[17] https://threejs.org/: Three.js is a cross-browser JavaScript library and Application Programming Interface (API) used to create and display animated 3D computer graphics in a web browser using WebGL.

[18] https://tailwindcss.com/: Tailwind CSS is a highly customizable, low-level CSS framework that gives you all of the building blocks you need to build bespoke designs without any annoying opinionated styles you have to fight to override.

[19] https://axios-http.com/: Axios is a promise-based HTTP client for the browser and Node.js. It makes it easy to send asynchronous HTTP requests to REST endpoints and perform CRUD operations.

[20] https://nodejs.org/: Node.js is an open-source, cross-platform JavaScript run-time environment that allows server-side execution of JavaScript code.

[21] https://hls-js.netlify.app/: Hls.js is a JavaScript library that plays HLS in browsers with support for MSE. It relies on HTML5 video and MediaSource Extensions for playback.
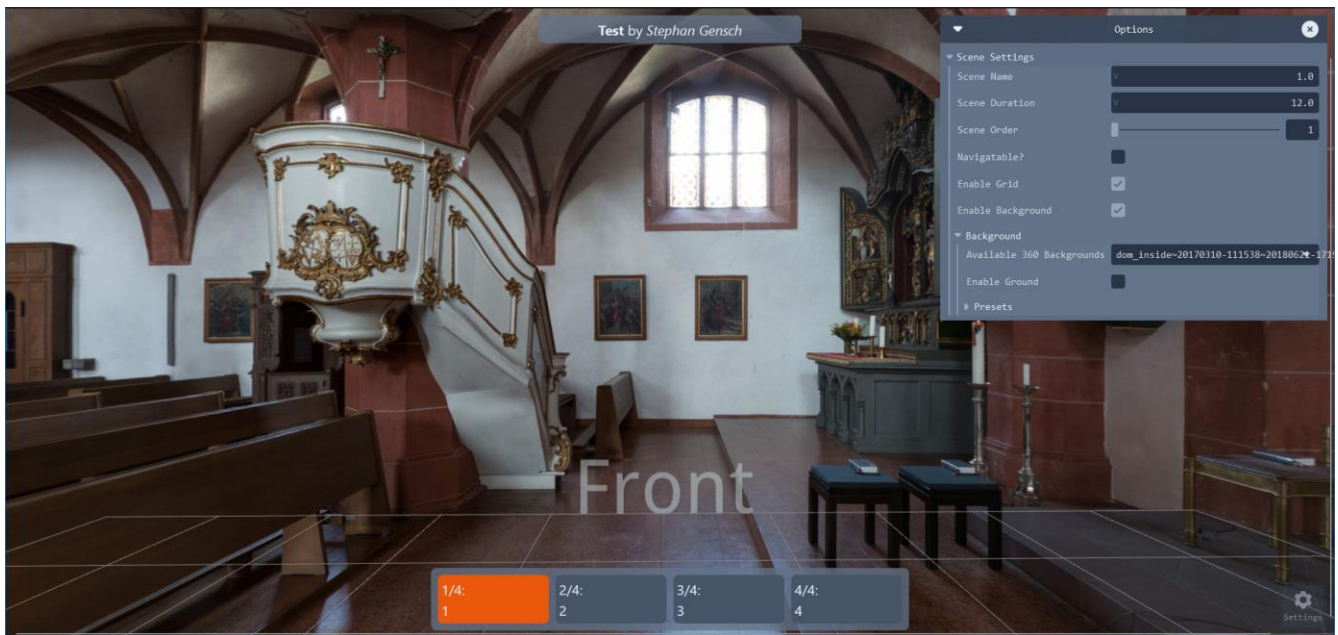
Figure 28: Setting a 360-degree background in Fader360 using the Scene Options panel.

Subtitles

Another significant feature of Fader360 is its in-built subtitle display functionality. Recognising the importance of accessibility and clarity in immersive storytelling, Fader360 offers the option to include subtitles in the narrative (see Figure 29). The platform can retrieve SRT (SubRip Subtitle) files from the MediaVerse node through the Fader360 backend, assuming these have been created and made available by the MediaVerse node. Fader360 then internally converts these files into WebVTT (Web Video Text Tracks), a widely supported standard format for displaying timed text in the video player. This automatic conversion ensures broad compatibility and functionality within the platform. The feature is not just about displaying text but also about providing flexibility and control to the users. They can navigate through settings that allow them to decide which subtitles they want to display, catering to multilingual content and enhancing the inclusivity of the platform. This fine level of control over the subtitle display contributes to a more personalised and engaging storytelling experience for a global audience.
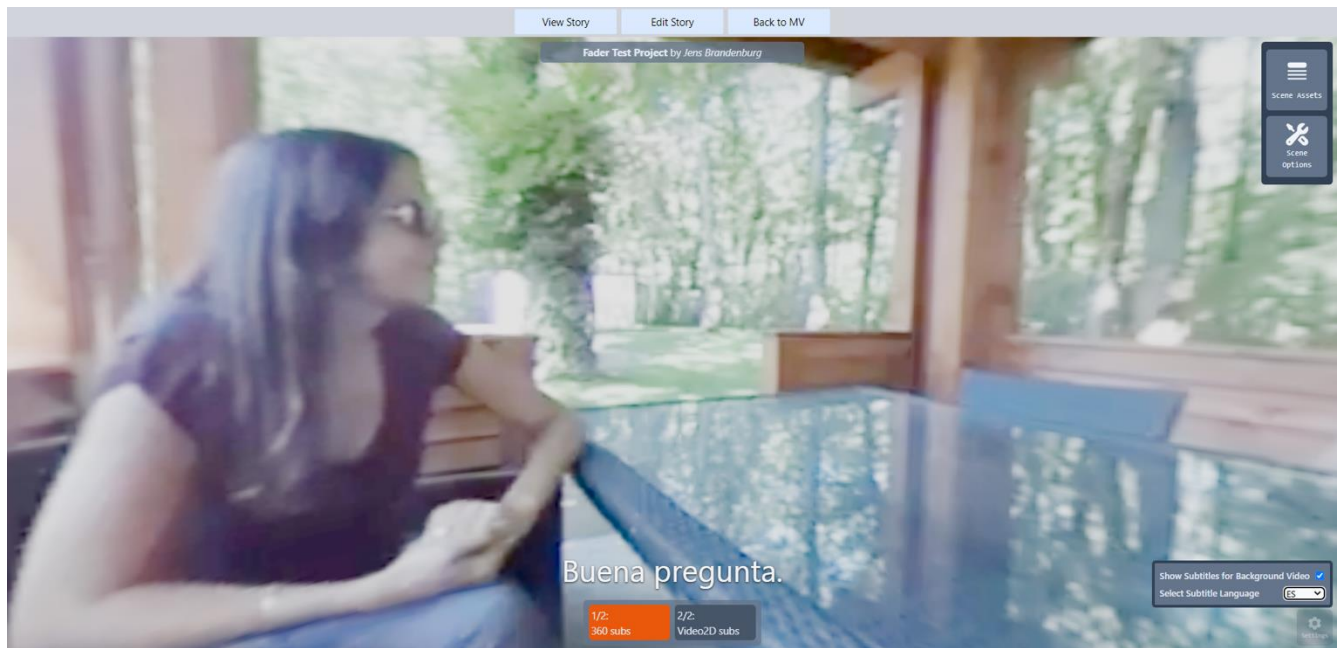
Figure 29: Display of subtitles for a 360-degree video.

Asset and Properties Management

The robust management tools complement the immersive features of Fader360. Central to these is the Asset Management Panel, designed to provide users with an organized and easy-to-use overview of all assets being used in a story. This helps to streamline the process of storytelling, enabling efficient tracking and management of different media elements that form a part of the narrative.

Going beyond mere asset management, Fader360 equips users with the Asset Properties Panel, a powerful feature that offers detailed control over individual assets. Figure 30 shows how Asset management and Asset Properties Panel appears in Fader360. With this panel, users can precisely configure an asset's position within the 3D space, creating a spatial arrangement that aligns perfectly with their vision. The panel also includes color settings for customizing visual aesthetics, fields for adding headers and descriptions to provide context, and asset-specific settings like loop and autoplay that further refine how the asset is integrated into the story. For maximum flexibility, there's even an option to delete an asset when no longer needed. These comprehensive controls over asset properties ensure that each media element contributes to the story in a way that's meticulously tailored to the narrative's needs, enhancing the overall coherence and impact of the storytelling experience.

Figure 30: Asset and asset properties panel to manage scene assets and their properties.

Scene Settings

The Scene Settings Panel (Figure 31) is another crucial feature within the Fader360 application. This tool gives users the freedom to shape their narrative's structure by defining the order of scenes, thus allowing them to create a storyline that flows according to their creative vision. In addition to this, users have the capacity to name each scene, contributing to the distinct identity and coherence of each part of the story.
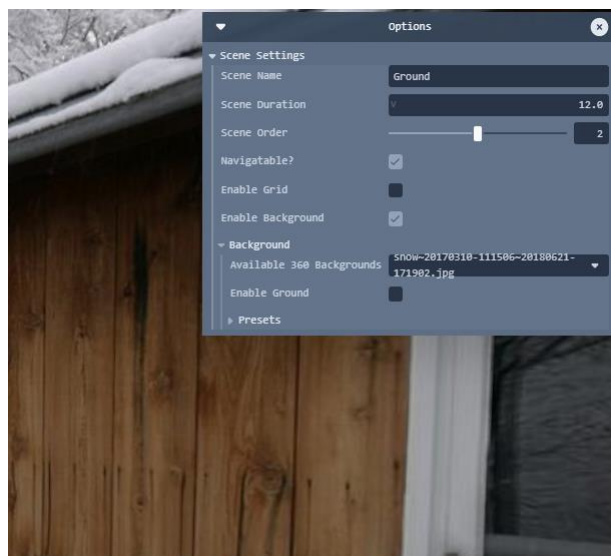
Figure 31: Scene settings panel.

Moreover, the panel provides various background options, extending the possibilities for visual storytelling. These options, readily available from the backend, enable users to create diverse, visually stunning backgrounds that enhance the immersive quality of each scene. With the Scene Settings Panel, Fader360 empowers users to create immersive narratives that effectively capture their vision and engage their audience on a deep level.

Interactive Scene Links

Fader360 offers the inclusion of Scene Links, a feature designed to enhance the interactive aspect of the storytelling process. These links, which can be added directly as text or converted from existing Image2D or TextCard assets, create connections between different scenes in the narrative. This feature not only allows for a more dynamic flow of the story but also lets the audience participate in the story by navigating through different scenes based on their interest (Figure 32).
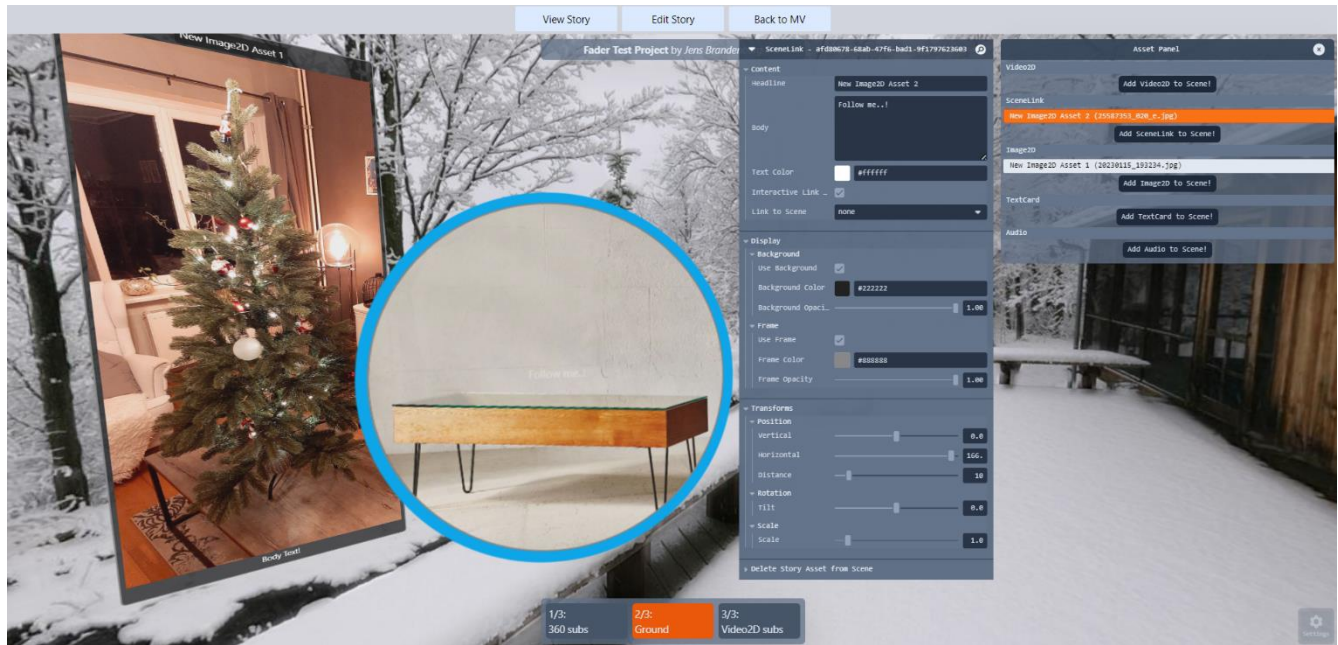
Figure 32: Scene link that allows interactively switching scenes when viewing a story.

Whether it is linking directly from one scene to another or using visual cues from Image2D or TextCard assets, the implementation of Scene Links provides a unique, interactive, and engaging way of storytelling. This function contributes significantly to the immersive nature of the Fader360 platform, facilitating a story structure that resonates with the audience and encourages exploration within the narrative space.

Viewing Features

Fader360 also includes several viewing features that enrich the user's interaction with the platform. Firstly, a View Mode has been implemented, which allows users to experience and explore other users' Fader360 stories. This feature promotes sharing and appreciation of diverse narratives, fostering a community of storytelling enthusiasts. For instance, Figure 33 shows the player functionality from the frontend perspective.

Figure 33: Screenshot of the player functionality of the frontend.

Secondly, Fader360 provides a feature that facilitates a smooth narrative flow by enabling looping through scenes according to a pre-defined scene order. 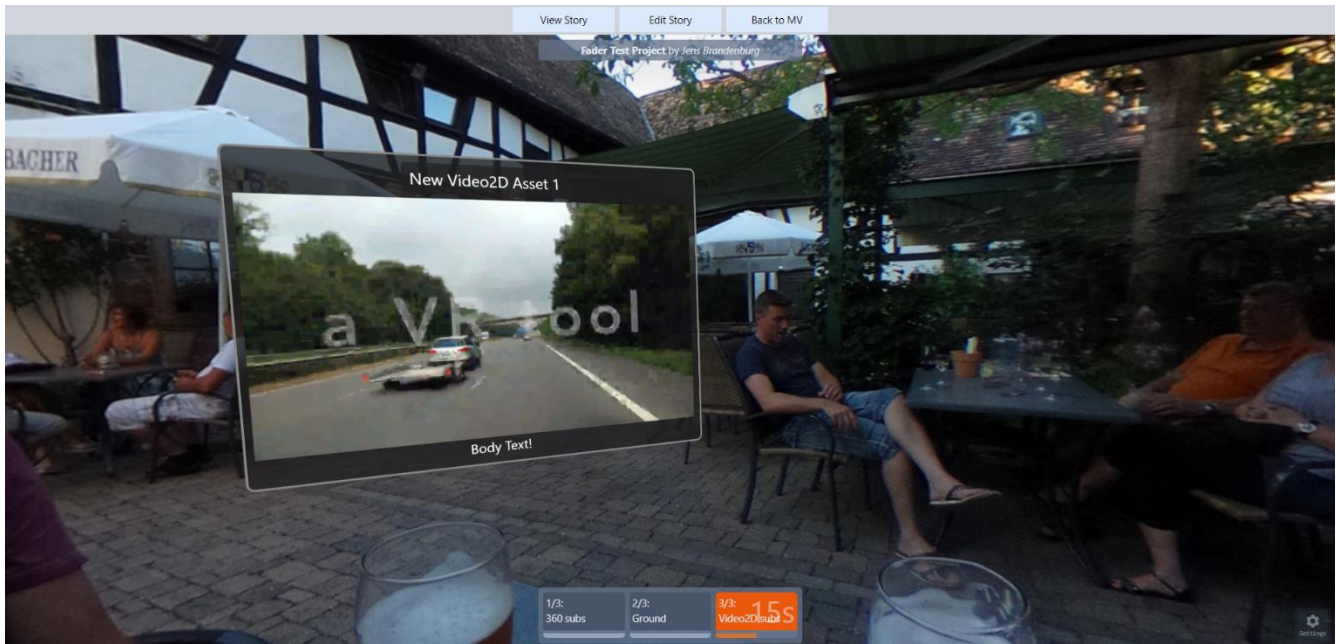This functionality allows the story to continue uninterrupted, automatically moving from one scene to the next, creating a seamless and immersive storytelling experience.

Lastly, to provide users with an intuitive understanding of their progress within a scene, Fader360 includes a progress display. This not only shows the total length of the scene but also provides a progress bar that indicates the elapsed time until the scene switch occurs. This real-time tracking feature enhances the user's engagement with the story, offering a clear sense of pacing and progression. Overall, these features aim to optimize the viewing experience, making the exploration of Fader360 stories a more immersive and enjoyable process.

All these developments have been aimed at enhancing user experience and increasing the versatility of Fader360's immersive storytelling potential. Our continued efforts will remain focused on user needs and technological advancements, ensuring that Fader360 remains at the forefront of immersive storytelling toolsets.

5.2   Updates on Integration with MV Node(s)

The integration between Fader360 and MediaVerse provides a streamlined authentication process for users. By allowing users to register using their existing MediaVerse credentials, Fader360 saves time and effort, thereby enhancing user experience. This process involves authenticating user credentials, acquiring additional user details from MediaVerse, and establishing a comprehensive user profile within Fader360. This tight integration creates a seamless connection between the two platforms and ensures user identity and data consistency.

Fader360's seamless integration with MediaVerse enables users to readily incorporate any MediaVerse asset or project into their Fader360 workflow. Through dedicated index pages for MediaVerse assets and projects, users can easily import content into Fader360 for creation and editing. This cohesive environment simplifies the process of importing projects, enhances productivity, and empowers users to efficiently create immersive stories.

Fader360 leverages MediaVerse's functionalities, such as the automatic generation and correction of subtitles, to enhance the user experience. Subtitle files are smoothly integrated with assets as they are imported into Fader360, and a dedicated synchronization button allows users to effortlessly maintain subtitle consistency across both platforms. This feature is available to all users, fostering a collaborative and inclusive environment.

To overcome the challenge posed by the lack of Fader360 story player support in MediaVerse, users can currently transfer their ready-to-publish stories back to MediaVerse as text files accompanied by Fader360 player links. These text files are treated as additional assets within MediaVerse, and users can monetize these and establish moral rights ownership. This solution ensures that users within the MediaVerse platform can easily engage with Fader360 stories. Plans include implementing a feature that allows users to lock a project once it is published, thereby preserving the integrity and consistency of the published content. We currently investigate the ability for other users to obtain rights to a Fader360 story and re-use or adapt it, taking into consideration some necessary clarifications on legal and technical issues.

5.2.1   User and Identity Management

Upon the initial engagement with Fader360, users can conveniently register using their existing MediaVerse credentials, assuming they have an account on a MediaVerse node. This process simplifies registration by negating the need for a separate Fader360 account, thus saving users' time.

During this registration phase, users are asked to provide their MediaVerse node, login details, and a unique username for the Fader360 environment (see Figure 34). These details are authenticated to verify user identity and access rights. Additionally, Fader360 obtains further information such as the user's first and last names, display name, and other pertinent details from MediaVerse. This data is used to create an in-depth user profile within Fader360. Once all information is gathered, a new user entry is added to the Fader360 database, linking the user's MediaVerse credentials to their Fader360 profile. This linkage facilitates a smooth connection between the two platforms, enabling users to access their MediaVerse assets and projects within Fader360.

Upon returning to Fader360, when a user enters their MediaVerse credentials, these details are cross-checked with the corresponding database entry for validation. If the data aligns, the user is authenticated and granted access to their Fader360 account. However, if any inconsistencies or errors arise during the verification process, an appropriate error message is displayed to guide the user towards resolving the issue, ensuring a seamless login experience.

This robust, secure authentication process promotes smooth integration between MediaVerse and Fader360, offering users a seamless transition between the two platforms. By utilizing their existing MediaVerse credentials, users can access Fader360's potent storytelling capabilities while maintaining a consistent user experience across both platforms.

Figure 34: Fader360 Login page with MediaVerse node and credentials.

5.2.2   Accessing Media from MV

After successful authentication via MediaVerse node credentials, Fader360 users gain immediate access to all their assets and projects. By seamlessly integrating with MediaVerse, users can integrate any asset or project into their Fader360 workflow via the "Add to Fader360" button, thus facilitating immediate use.

Fader360 includes a dedicated MediaVerse asset index page that displays previews of all the user's MediaVerse assets. Selecting "Add to Fader360" on an asset prompts the retrieval of detailed information about that asset from MediaVerse. The original asset is then downloaded to Fader360, which analyses the asset, extracts necessary data, and transcodes it to generate multiple versions with varying qualities and aspect ratios. Once complete, the asset's status updates to "ready," signifying it is prepared for integration into a Fader360 story. Figure 35 displays the MediaVerse asset index page in Fader360 environment.

Besides the asset index page, Fader360 also features a MediaVerse project index page (see Figure 36). Selecting a MediaVerse project provides an overview, including information about all associated users, the current user's role, and an "Add to Fader360" button. This button initiates the creation of a new Fader360 project based on the selected MediaVerse project, with each asset undergoing processing before assignment to the new project. Users can add the same MediaVerse project to Fader360 multiple times, offering the flexibility to create various Fader360 stories using the same project assets.
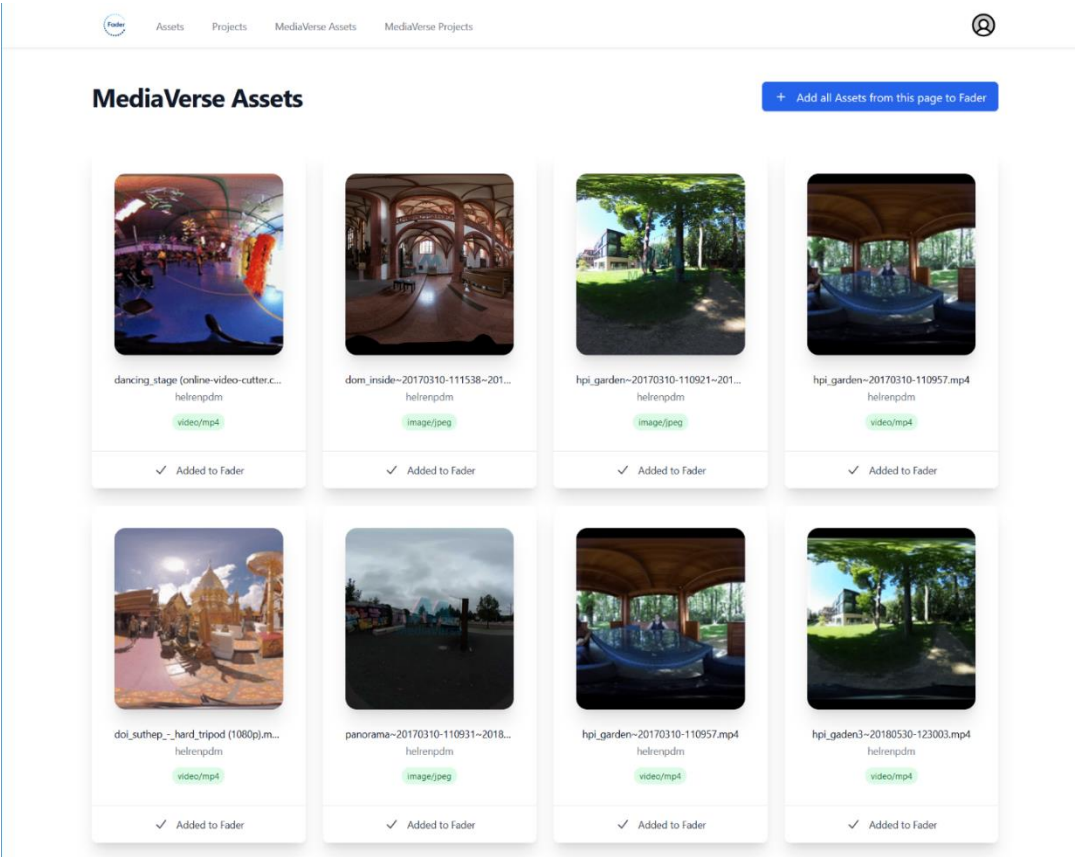
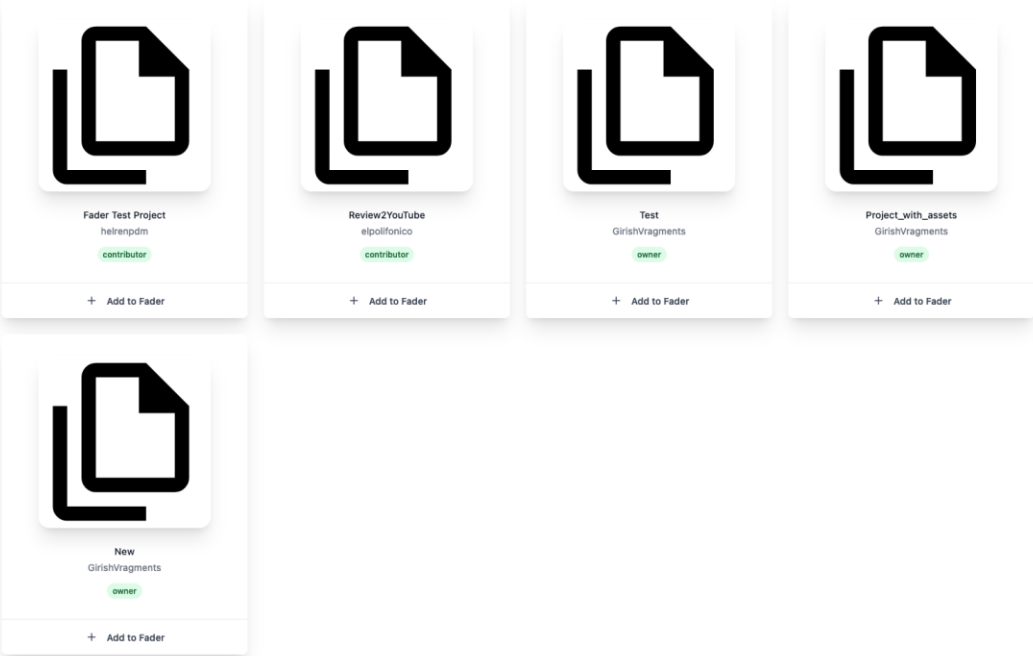*Figure 35: MediaVerse asset index page in Fader360.*

Figure 36: MediaVerse project index page in Fader360.

Users logged into both the MediaVerse node and Fader360 with the same credentials in the same browser session can use the Fader360 button on the MediaVerse project detail page (Figure 37). Selecting this button opens that specific project detail page in Fader360, facilitating easy project importation. If the user is logged only into MediaVerse, Fader360 will prompt them to log in. After logging in, Fader360 automatically redirects them to the appropriate MediaVerse project detail page, ensuring a smooth user experience.
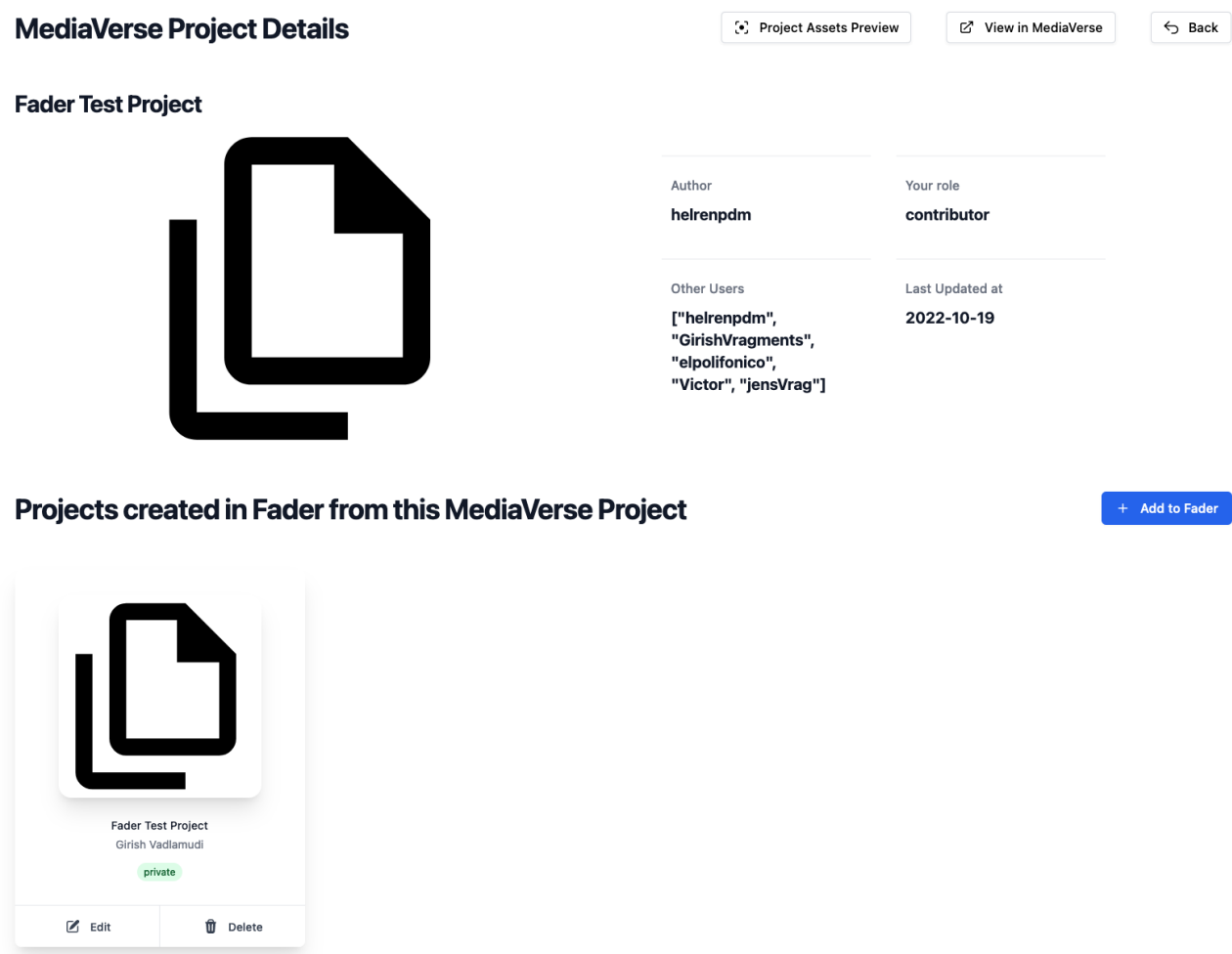
Figure 37: MediaVerse project detail page in Fader360

This integration between MediaVerse and Fader360 provides a harmonized user environment, promoting a fluid workflow and improved productivity. It simplifies the process of importing projects into Fader360 and allows users to leverage MediaVerse's advanced collaboration features alongside Fader360's sophisticated 360° story editing capabilities. By combining the strengths of both platforms, users are empowered to create engaging and immersive narratives more efficiently.

5.2.3   Utilization of MediaVerse Support Functions

In a bid to enhance user experience within Fader360, we have implemented the importation of asset subtitles from MediaVerse. MediaVerse offers features that facilitate automatic generation and correction of subtitles for assets. This process is streamlined for users, who can effortlessly upload video or audio assets to MediaVerse, and the platform will automatically append the relevant subtitles.

When users import these assets to Fader360, either directly or via a project, the subtitle files are smoothly integrated along with the assets. This ensures that users can immediately access synchronized subtitles within the Fader360 environment, improving overall usability and convenience.

As mentioned above, users retain the flexibility to modify subtitles within MediaVerse. To support this functionality, we have integrated a dedicated button within Fader360 that synchronizes subtitles between the two platforms (see Figure 38). By selecting this button, users can effortlessly update subtitles across both systems, maintaining consistency and accuracy.
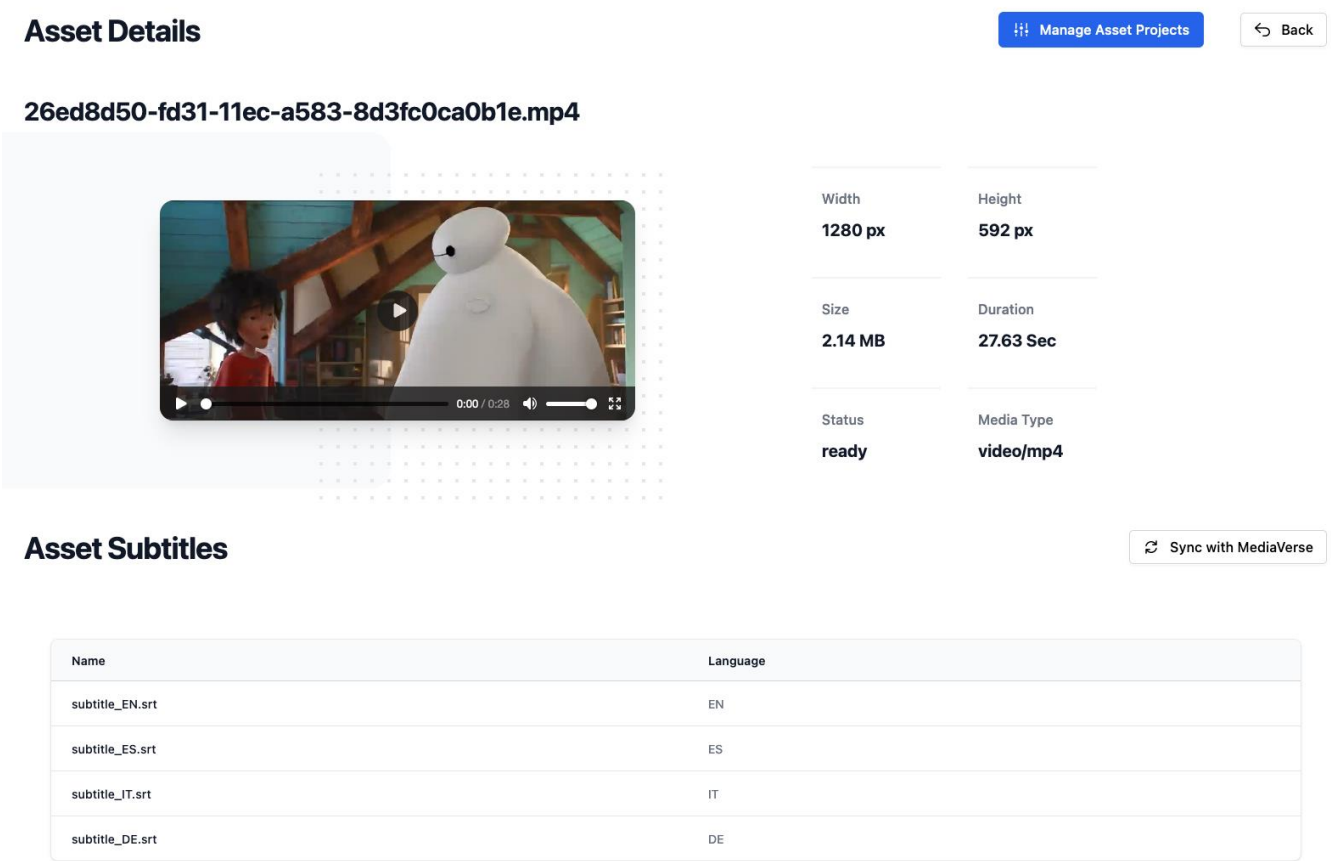
Figure 38: Asset detail page in Fader360 with asset subtitles and sync button.

Notably, this synchronization functionality is available to all users, regardless of their asset license. This means that whether a user is the creator or the owner of the asset he/she is able to utilize the subtitle synchronization feature in the same way. This contributes to a collaborative and inclusive environment for all Fader360 users.

5.2.4 Store Media into MediaVerse

Within the context of Fader360, the lack of Fader360 story player support in MediaVerse presents a challenge. This is due to a Fader360 story not being a containerized media item, like an image or video, for which standard web viewers or players exist, but it is rather an interactive website served through its own application (namely the Fader360 frontend which relies on content being served from the Fader backend). Nevertheless, we've developed a solution for users wishing to publish their story. They can transfer it back to MediaVerse as a text

file along with a Fader360 player link. This text file is treated as an additional asset within MediaVerse, allowing users to monetize it and claim moral rights ownership. Furthermore, users can designate the license type for this asset, providing additional flexibility for their content. As previously mentioned, ideas exist to improve this feature and allowing better sharing and re-use of Fader360 stories, but there are still legal and technical challenges to be solved.

For MediaVerse users interested in this asset, accessing the Fader360 story for viewing is straightforward. They simply need to select the provided link, which will open the story in the Fader360 player (see Figure 39). This seamless integration enables users within the MediaVerse platform to readily interact with and explore the Fader360 story.
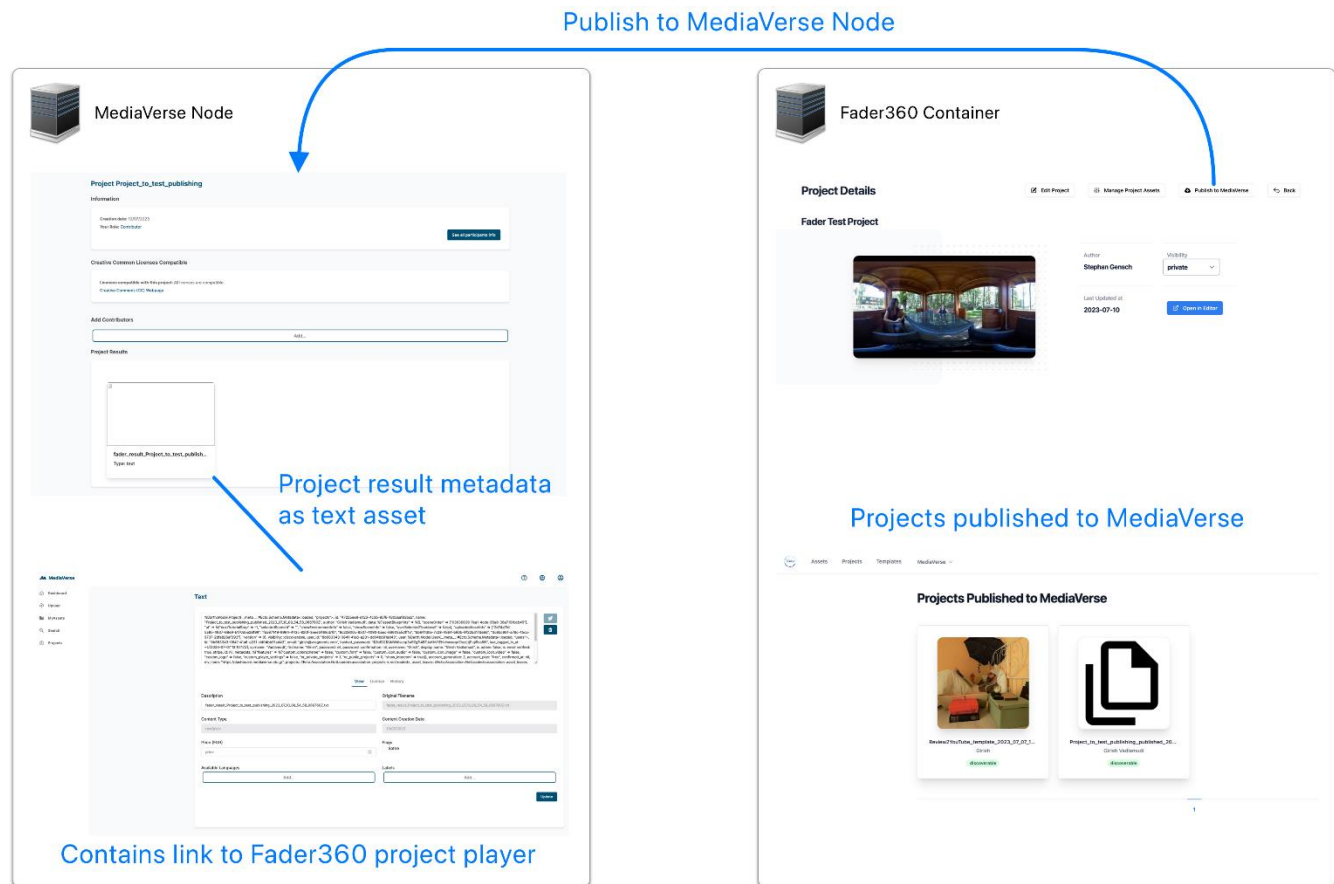
Figure 39: Schematic view of publishing a project result from Fader360 into a MediaVerse project.

However, once the project result has been transferred to MediaVerse, it is critical to avoid making any alterations. Therefore, we recommend caution and we advise against making any changes at this stage. To help manage this concern, we plan to introduce a feature that allows users to lock the project once it's been published to MediaVerse from Fader360. This planned feature aims to maintain the integrity and consistency of the published content within MediaVerse, ensuring it remains unmodified and intact.

5.3   Requirements Update, Outlook, and Exploitation

5.3.1   Requirements Update

This sub-section presents the updates on the requirements addressed by the authoring tool. Table 2 is an update on Table 3.1 - Developments to support the requirements across the technical components from D5.2 - Immersive Storytelling Authoring Tools v1[22] and D5.3 - VR Collaboration, Social Media Analytics, and Immersive Storytelling - Authoring Tools v2[23].

Table 2: Developments to support the requirements across the technical components.

REQUIREMENT	REQUIREMENT DESCRIPTION	DEGREE OF FULFILMENT
UX-02 / Content Information will be easily visible.	This refers to the display of asset information through Fader360 and will help users pick the right assets for their immersive story. Since the user is authenticated against an MV node, all available asset information can be acquired from the MV node DAM and presented to the user upon asset list and asset detail views.	**Fully implemented**
COLL-02 / The dashboard provides a concise overview of various types of information.	From the Fader360 authoring tool, only limited information needs to be available. Main API endpoints for Fader360 are the user, project and asset endpoints to acquire information and data and write information back on the project status, e.g., used assets to make that available to usage tracking within MediaVerse.	**Fully implemented**
COLL-04 / Project participants communicate via simple chat.	This feature is no longer directly associated with the authoring tools and should rather be a support function of a MediaVerse node itself.	**Not implemented or used**
COLL-05 / Project participants can add notes in the project.	This feature is no longer directly associated with the authoring tools and should rather be a support function of a MediaVerse node itself.	**Not implemented or used**
PLAY-01 / The player enables display/overlay of subtitles from .srt files.	Uploaded video to a MediaVerse node can make use of the accessibility features provided by STXT RACU workflow. Using this, subtitle text files (.srt) are returned to MediaVerse and can be referenced through the original video asset. When such an .srt file exists for a given video, it can be used by Fader360 to display the text inside a 360 story as an overlaid text card.	**Fully implemented for 360-degree and 2D video**
PLAY-02 / Accessibility extras can be switched on and off.	The display of subtitles in a story should be an option in the player - similar to a "CC" button on most web video players.	**Fully implemented for 360-degree and 2D video**
BRKR-04 / Incorporating segments will have implications on the usage rights of the project.	A user is generally only able to use assets (from their MediaVerse account) in a Fader360 project, that they have obtained the rights to download them for. Since these assets need transformations for technical reasons and they will be part of a collage, only such assets should be eligible to download and use in Fader360 that have the correct rights to use them in that fashion.	**Fully implemented in asset retrieval logic**

---

[22]https://MediaVerse-project.eu/wp-content/uploads/2021/10/MediaVerse_D5.2_Immersive-Storytelling-AuthoringTools-v1.pdf
[23] https://MediaVerse-project.eu/wp-content/uploads/2022/09/D5.3-V1.0.pdf

AUTH-01 / Mobile and desktop interfaces.	The Fader360 backend, as well as the editor and player are fully working on desktop interfaces. With mobile devices and VR, certain restrictions apply, such as default muting of audio content and necessary device capabilities to display VR content with rotation sensors active.	**Fully implemented on desktop with minor deviations on mobile**
AUTH-02 / MediaVerse enables content creators to do basic editing on audio, video, 360 Video.	Basic editing of assets should be supported outside of Fader360. Fader360 operates on finalised media assets.	**Not implemented**
AUTH-03 / External authoring tools can be connected with MediaVerse for easy updates.	A Fader360 user can log in to their MediaVerse account. Having done this, they can acquire updates on assets within their projects and update their stored information on a Fader360 project accordingly.	**Fully implemented**
AUTH-04 / The authoring UIs need to support various languages.	Fader360 is currently available in English, French and German. The localization can be extended, provided there is capacity to translate within the consortium.	**Full implementation pending**
AUTH-05 / Creators can add and edit various accessibility enhancements to content.	Fader360 is designed to support subtitles on videos, using generated .srt-files from a MediaVerse node.	**Done through MediaVerse node and accessibility toolset**
XR-02 / A 3D Scene can be exported as a video clip.	Due to the technical complexity of this issue (performance of capture and rendering) and various logical fallacies with interactive, non-linear content, a video export is not feasible. If users want to capture their experience, they should rather use screen capture software, which will yield a better result.	**Not implemented**
XR-05 / 360-video can be enhanced by adding objects/assets.	Fader360 supports adding various interactive and non-interactive content on top of 360-degree video.	**Fully implemented**
XR-06 / 360-video can be enhanced by hotspots with different functions.	Fader360 supports adding various interactive hotspots on top of 360-degree video.	**Fully implemented**
RGHT-03 / The system must be able to trace content sources (inside MV).	Fader360 relies on the MV-DAM to trace content sources and displays this accordingly to the user.	**Not implemented in Fader360, but done through MediaVerse association**
MNTR-01 / Monitor the consumption and re-use of content.	Since content used in a Fader360 story is downloaded and transcoded, there is a disconnect from the original asset to its derivative works. While statistics of a Fader360 story are generated and stored, this offers only relative monitoring on a project level, not on individual asset usage. However, if we look at the analogy of video production using assets from an MV node, this is quite comparable. Assets are downloaded and used in a new video. At that point, a direct link to the assets and their usage in a new video is lost. When the result is uploaded to MediaVerse (a Fader360 JSON project file or a video file in the latter scenario), asset references must be tagged inside MediaVerse.	**Not implemented or used**

5.3.2   Outlook

Our primary focus for future development is to prioritise the user-friendliness of the Fader360 interface, both on the frontend and backend. Building upon user feedback, we are committed to making significant quality of life improvements to enhance the overall user experience. Our goal is to streamline processes, eliminate pain points, and ensure a seamless and intuitive platform for storytellers.

Additionally, we are actively working on implementing support for 3D models within the Fader360 story. This exciting feature will enable users to leverage the capabilities of MediaVerse by directly importing and incorporating 3D models into their narratives. By expanding Fader360's compatibility with MediaVerse's 3D assets, we aim to provide users with even more creative possibilities and opportunities for immersive storytelling.

By prioritising user-friendliness and introducing support for 3D models, our development efforts in the future are aligned with creating a dynamic and empowering platform for storytellers. We will continue to listen to user feedback, identify areas for improvement, and implement enhancements that enhance the usability and creative potential of Fader360. Our aim is to enable users to craft captivating narratives with ease and push the boundaries of immersive storytelling.

5.3.3   Exploitation

In alignment with our project requirements, we are excited to announce that we will be making the code for both the Fader360 frontend and backend open source on GitHub in the upcoming weeks. By embracing open-source principles and fostering a community-driven approach, we encourage developers from around the world to contribute to the development of Fader360. This collaborative effort will enable continuous improvement and the addition of new features that cater to the evolving needs of storytellers.

We understand the importance of flexibility and customization, which is why we invite use case partners and interested individuals to leverage the open-source nature of Fader360. Whether it is a specific feature requirement or an enhancement tailored to their storytelling needs, anyone can easily access the latest codebase and begin working on their desired additions. This accessibility ensures that Fader360 remains a versatile platform, empowering storytellers to create immersive narratives that resonate with their audiences.

Furthermore, as part of our efforts to enhance the overall deployment experience, we have made significant changes to our deployment pipeline. This means that use case partners or individuals interested in hosting their own MediaVerse node can do so seamlessly. By pulling the necessary images, including Fader360, from the Docker Hub, the setup process becomes simplified, requiring minimal user effort. This allows individuals to have their own dedicated instance of Fader360 running alongside their personalised MediaVerse node, providing them with full control over their storytelling environment.

In summary, by open sourcing the Fader360 code and facilitating easy setup and deployment through streamlined processes, we aim to foster collaboration, innovation, and customization within the Fader360 community. We look forward to the contributions and advancements that will arise from this collective effort, enabling storytellers worldwide to unleash their creativity and share compelling narratives in new and exciting ways.

5.3.4   Conclusions and Future Work

Fader360, the immersive storytelling toolset by Vragments, is at the forefront of revolutionising storytelling through its ground-breaking web-based platform. By seamlessly blending interactive hotspots and mesmerising 360° content, Fader360 takes audiences on captivating journeys into unexplored realms. The recent collaboration with MediaVerse has solidified Fader360's position as a cutting-edge player in the industry. This integration empowers storytellers to leverage a vast array of media resources, tools, and possibilities, enabling them to create immersive narratives that resonate deeply with global audiences. The updates to Fader360 have streamlined the user experience, enhanced the integration with MediaVerse, and simplified the workflow, allowing storytellers to bring their narratives to life more efficiently.

The collaboration between Fader360 and MediaVerse signifies a significant shift in the storytelling landscape. By joining forces, these visionary platforms have unlocked new dimensions of storytelling and elevated the art of immersive experiences. Storytellers now have access to a vast array of media resources, tools, and possibilities, empowering them to create captivating narratives that resonate deeply with audiences worldwide. The seamless integration between Fader360 and MediaVerse allows for the effortless importation of MediaVerse content into Fader360, eliminating the need for manual transfers or conversions. This integration not only expands the possibilities for storytelling but also streamlines the workflow, enabling storytellers to focus on their creative process and bring their stories to life in extraordinary ways.

As Fader360 continues to push the boundaries of what is possible in virtual reality storytelling, it inspires and shapes the future of this dynamic and ever-evolving medium. The recent updates to Fader360 have made significant strides in simplifying the user experience and integrating seamlessly with MediaVerse. By addressing user feedback and incorporating the latest technologies, Fader360 has enhanced its platform, allowing storytellers to create immersive narratives with ease. With the seamless integration between Fader360 and MediaVerse, users can swiftly commence their story development using their existing MediaVerse assets, significantly reducing the time taken to get started on a project. By leveraging the strengths of both platforms, Fader360 empowers storytellers to create captivating narratives and brings the future of immersive storytelling closer than ever before.

In our continuous efforts to enhance Fader360, we are committed to improving both the frontend and backend systems to provide an exceptional user interface and experience. We value user feedback and actively seek it out to identify areas of improvement and address them promptly. By addressing user pain points and incorporating valuable feedback, we strive to establish a seamless and enjoyable user experience.

In terms of the backend, we are currently developing a feature that allows users to send their Fader360 story results back to MediaVerse. This integration will streamline the process and enable users to seamlessly share their stories across platforms. Additionally, we are implementing various quality of life improvements to the backend, including the introduction of filtering functionality to asset and project index pages. These enhancements aim to enhance efficiency and make it easier for users to navigate and manage their content within Fader360.

On the frontend side, we are working on an exciting new feature that will support the integration of 3D models from MediaVerse into Fader360. As MediaVerse already provides support for 3D models, this feature will allow users to directly import and incorporate these models into their Fader360 stories. This development is already

supported in the Fader360 backend, and we are actively working on extending this support to the frontend to provide users with even more creative possibilities.

Overall, our ongoing efforts in both the frontend and backend development of Fader360 reflect our dedication to continuously improving the platform and delivering an exceptional user experience. By incorporating user feedback, introducing new features, and streamlining integration with MediaVerse, we aim to empower storytellers and provide them with the tools they need to create immersive and captivating narratives.

6. Conclusions

During the recent reporting period, considerable progress has been made on the Social Analytics Engine, in addition to the continuous enhancements in VROdos and Fader360 platforms. By making use of advanced web technologies and leveraging the facilities provided by MediaVerse, all three platforms have taken strides towards improving user experience and expanding their respective capabilities.

In response to substantial changes to Twitter's API access and pricing structure in 2023, adaptations have been made to the Social Analytics Engine. The transition to a paid basic tier and the move from streaming API to RESTful API polling represent significant modifications to our approach. Additionally, we have transitioned from a login-with-Twitter only system to a username-based approach, decreasing our dependence on third-party platforms. Further improvements in the engine include developments in report generation, enhancements to the trend monitoring and alerting system, upgrades to the analysis libraries, and infrastructure improvements for better performance and data management.

VROdos, with its novel real-time virtual production methodology, continues to demonstrate its potential in a diverse range of contexts, from education to entertainment. The platform has seen important improvements in its interfaces, undo-redo mechanism, compilation speed, and stakeholder participation, thereby enhancing the overall user experience. The incorporation of smartphone cameras in the creation process stands as a significant step towards greater user participation.

Meanwhile, Fader360 continues to push the boundaries in the realm of immersive storytelling. By integrating 360° content and interactive hotspots, it offers unique narrative experiences for its users. Fader's collaboration with MediaVerse has extended its capabilities, offering users access to a broad array of media resources and tools.

In summary, this reporting period has seen substantive progress in the Social Analytics Engine, VROdos, and Fader360. These platforms, utilizing the resources provided by MediaVerse, continue to evolve, and expand their offerings, responding effectively to changing conditions and user feedback. As we move forward, we remain committed to addressing remaining challenges, integrating user feedback, and persistently seeking to improve user experiences.

7. References

Ververidis, D., Migkotzidis, P., Nikolaidis, E., Anastasovitis, E., Papazoglou-Chalikias, A., Nikolopoulos S., & Kompatsiaris, I. (2021). An authoring tool for democratizing the creation of high-quality VR experiences, Springer Virtual Reality, June 2021, DOI: https://doi.org/10.1007/s10055-021-00541-2