Project Report: AR Museum Viewer

Project Title

AR Museum Viewer

Abstract

The **AR Museum Viewer** is an interactive web-based platform that allows users to explore, visualize, and interact with 3D models in both virtual and augmented reality (AR). Designed for museum applications, this system provides an immersive and informative experience where users can view 3D artifacts, learn their historical context, and interact with them using advanced AR features. The platform integrates **Firebase** for dynamic model hosting and leverages **React**, **Three.js**, and **Google's model-viewer** for rendering and interaction.

Table of Contents

---

1. Introduction

The AR Museum Viewer bridges the gap between traditional museum experiences and modern AR technology. It enables users to visualize 3D models of artifacts in both desktop and AR environments, providing accessibility and an enhanced learning experience.

---

2. Objectives

- Provide a platform to explore 3D artifacts from museums in an interactive and immersive manner.
- Enable users to upload and view custom 3D models.
- Offer historical context for artifacts via dynamic metadata fetched from Firebase.
- Support AR experiences on WebXR-compatible devices like HoloLens.

---

3. Features

1. **3D Model Viewing**: Users can view and interact with 3D models using camera controls and animations.
2. **Firebase Integration**: Models and their metadata are fetched dynamically from Firebase cloud storage.
3. **AR Compatibility**: Support for AR visualization using WebXR and device-specific viewers like Scene Viewer (Android) or Quick Look (iOS).
4. **Custom Model Upload**: Users can upload local `.glb` or `.gltf` files to view models instantly.
5. **Interactive Editor**: An editor feature with real-time control over lighting, background, and wireframe modes for models.

6. **History Display and TTS**: Metadata such as historical descriptions are displayed and read aloud using text-to-speech functionality.
7. **Responsive Design**: Optimized for different devices and screen sizes.

---

4. System Architecture

Frontend:

- Developed using **React** for UI interactivity and responsiveness.
- Integrated **@google/model-viewer** for AR and 3D rendering.
- Uses **@react-three/fiber** and **drei** for advanced model editing.

Backend:

- Firebase provides cloud storage for hosting 3D models and metadata.

Text-to-Speech:

- Utilizes the browser's `speechSynthesis` API for reading aloud artifact descriptions.

AR Framework:

- WebXR for AR visualization on compatible devices.
- Device-specific AR viewers (e.g., Scene Viewer for Android).

---

5. Technologies Used

- **React**: Frontend framework for dynamic rendering and interactivity.
- **Firebase**: Cloud storage for 3D models and metadata.
- **Three.js (@react-three/fiber)**: Rendering and controlling 3D objects.

- **Google Model-Viewer**: Simplified AR integration.
- **JavaScript APIs**: For text-to-speech, file handling, and WebXR compatibility.
- **CSS**: Custom styling for a professional and interactive design.

---

6. Implementation Details

Firebase Integration

- Models and metadata are fetched dynamically from Firebase Storage using `getStorage`, `ref`, and `listAll`.
- Metadata is parsed from a `metadata.json` file containing historical descriptions for each model.

Model Viewer Component

- Uses `<model-viewer>` for displaying 3D models with AR support.
- Provides functionalities like auto-rotate, camera controls, and environment lighting.

AR Compatibility

- WebXR integration enables immersive AR sessions for devices like HoloLens.
- Includes fallbacks for device-specific AR viewers when WebXR is not supported.

Editor Component

- Built with Three.js, allowing users to manipulate lighting, background, and wireframe modes.
- Supports real-time animation playback and pausing.

---

7. Challenges and Solutions

Challenge	Solution
AR compatibility across devices	Implemented device-specific fallbacks using WebXR and model-viewer features.
Dynamic loading of large 3D models	Optimized Firebase fetch requests and used React Suspense for fallback loading.
Text-to-speech interruptions	Added `speechSynthesis.cancel()` to handle overlapping speech requests effectively.
Ensuring wireframe toggle efficiency	Iterated through all mesh objects in the model to apply the wireframe setting dynamically.

---

8. Future Work

- **Integration of VR**: Add support for Virtual Reality (VR) interactions.
- **AI Metadata Generation**: Use AI to auto-generate artifact descriptions from model data.
- **Multi-User Collaboration**: Allow real-time shared AR experiences.
- **Expanded Model Formats**: Support additional 3D formats like `.obj` and `.fbx`.

---

9. Conclusion

The AR Museum Viewer provides an innovative approach to exploring artifacts in 3D and AR. By combining Firebase's dynamic hosting with cutting-edge rendering frameworks, this project delivers a scalable and engaging platform. Its extensible architecture makes it ideal for future enhancements like VR support and AI-driven features.

Project Repository: https://github.com/Mahanthb/ar-museum