# **Documentation for Tech Stack**

This document provides an overview of the technology stack used in the project. The stack includes various programming languages, tools, libraries, and services that are used to create a comprehensive solution for the application. Each component in the stack plays a crucial role in the system, working together to provide a seamless experience for the end-users.

## **1. C++**

C++ is a general-purpose programming language that offers high performance and fine-grained control over system resources. It is used in the project to create and optimize core components, such as the integration with MetaHuman and the conversion of Face Mesh to 3D NPC models.

## **2. Python**

Python is an easy-to-learn, versatile programming language that is widely used for AI and machine learning tasks. In this project, Python is employed to implement AI algorithms such as SST (Speech to Text), TTS (Text to Speech), and LLM (Large Language Model). It also facilitates the interaction with Azure Cognitive Services and OpenAI Chat GPT 3.5/4.0.

## **3. UE5 (Unreal Engine 5)**

Unreal Engine 5 (UE5) is a cutting-edge game engine that provides powerful tools for creating high-quality 3D experiences. In this project, UE5 serves as the foundation for the application, handling graphics rendering, user interaction, and the integration of various technologies like MetaHuman and Quixel Bridge.

## **4. Voice SDK by Meta**

The Voice SDK by Meta provides tools for capturing and processing microphone data from Oculus Quest 2 devices. The 'Dictation' feature is used in this project to collect voice input from users, enabling a more intuitive and immersive experience.

## **5. Azure Cloud Computing**

Azure Cloud Computing provides scalable, on-demand computing resources to execute AI algorithms with increased speed and efficiency. In this project, Azure is used to host and compute the AI algorithms implemented in Python, ensuring faster response times and improved performance.

## **6. Azure Cognitive Services**

Azure Cognitive Services is a suite of APIs and services that enable developers to build intelligent applications. In this project, the following Cognitive Services are utilized:

* STT (Speech to Text): Converts spoken language into written text.
* TTS (Text to Speech): Generates natural-sounding speech from text.

## **7. HTTP REST Requests**

HTTP REST Requests are used to send and receive data between different components in the system. In this project, REST Requests are employed to send queries to Azure Cognitive Services and OpenAI Chat GPT 3.5/4.0, and to receive the corresponding responses.

## **8. OpenAI Chat GPT 3.5/4.0**

OpenAI Chat GPT is a state-of-the-art language model that generates human-like text. In this project, GPT 3.5/4.0 is used to generate engaging and contextually relevant text based on user input. This text is then used to drive the interactions between the user and the application.

## **9. MetaHuman**

MetaHuman is a machine learning technology that can convert Face Mesh data into high-quality 3D NPC model files. In this project, MetaHuman is used to create realistic and visually stunning 3D characters based on user input.

## **10. Quixel Bridge**

Quixel Bridge is a tool that simplifies the process of importing and managing 3D assets in Unreal Engine. It is used in this project to handle the rendering and integration of 3D models generated by MetaHuman.

## **11. MetaHuman SDK**

The MetaHuman SDK is a machine learning-based solution for generating procedural animations based on audio input. In this project, the SDK is employed to create lifelike animations that synchronize with the speech generated by.

## **12. Workflow Description of the Tech Stack**

**User input:** User speaks into the Oculus Quest 2 microphone.

**Voice SDK by Meta:** Captures and processes the microphone data.

**Azure Cognitive Services - STT:** Converts the captured speech to text.

**OpenAI Chat GPT 3.5/4.0:** Processes the text input and generates a relevant response.

**Azure Cognitive Services - TTS:** Converts the generated response text to speech.

**MetaHuman SDK:** Generates procedural animations based on the generated speech.

**MetaHuman:** Converts Face Mesh data into 3D NPC model files.

**Quixel Bridge:** Handles rendering and integration of the 3D models in Unreal Engine 5.

**UE5 (Unreal Engine 5)**: Renders the scene with the 3D models and animations, displaying the final output to the user.

## **12. Visualization of the Workflow Diagram**

