

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

Diseño de un agente virtual basado en metahumanos para establecer puntos de consulta de procesos internos de la universidad

PROYECTO INTEGRADOR

Previo la obtención del Título de:

Ingeniero en Telemática

Presentado por:

Jordan Javier Miranda Trujillo

Karen Fernanda Santillan Segura

GUAYAQUIL - ECUADOR

Año: 2023

DEDICATORIA

Jordan Miranda

El presente proyecto lo dedico a esa persona que sin tener la obligación, me dio apoyo económico, el cual me permitió dedicarme a estudiar. Por otra parte, gracias por esas palabras de “Vi tu esfuerzo y voy a apoyarte, porque se que llegarás lejos”.

Karen Santillan

El presente proyecto lo dedico con profunda gratitud y afecto sincero a mis padres Oscar Santillan y Mariuxi Segura, a mis hermanas Jamile y Doménica, a Manicricri, a mis amigos, a mi abuelo, a mis abuelas y a mi familia en general, quienes siempre me brindaron su más sincero apoyo y ayudaron durante mi vida académica.

AGRADECIMIENTOS

Jordan Miranda

Mi mas sincero agradecimiento a esa persona que sin tener la obligación, me dio apoyo económico, el cual me permitió dedicarme a estudiar. Por otra parte, gracias por esas palabras de “Vi tu esfuerzo y voy a apoyarte, porque se que llegarás lejos”.

Karen Santillan

En este punto crucial de mi carrera, agradezco profundamente a quienes han sido mi constante fuente de inspiración y sólido apoyo. En primer lugar agradezco a mis padres, hermanas, Manicricri, amigos y familia. Agradezco sinceramente a nuestra tutora, Dra. Rebeca Estrada, por su extraordinaria guía tanto en este trabajo como a lo largo de la carrera, a la Ing. María José por su apoyo y orientación, a Jordan por su dedicación en el proyecto, a mis compañeros y docentes por su soporte incondicional.

DECLARACIÓN EXPRESA

Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; Jordan Miranda y Karen Santillan damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual



Jordan Miranda



Karen Santillan

EVALUADORES

Mgtr. María José Ramírez
PROFESOR DE LA MATERIA

Rebeca Estrada Pico Ph.D.
PROFESOR TUTOR

RESUMEN

En el actual escenario de la educación superior las universidades deben afrontar diversos desafíos para lograr informar a sus estudiantes acerca de los procedimientos internos y servicios ofrecidos por los departamentos involucrados. Considerando lo mencionado y tomando en cuenta que ESPOL posee diversos canales informativos que pueden presentar dificultades, el presente proyecto se centró en establecer el diseño y prototipo de un agente virtual basado en metahumanos para establecer puntos de consulta de procesos internos de la universidad, para ello se definieron diversos objetivos que incluyen el desarrollo del metahumano, identificar los procesos más consultados, bosquejar una estructura física y realizar las pruebas respectivas. Asimismo se determinó la metodología a utilizar con sus respectivas herramientas y tecnologías, para la definición del metahumano se implementó Metahuman Creator, Quixel Bridge Plugin, Unreal Engine, Mixamo y Audio2Face. Para el entrenamiento del agente se configuró una red neuronal en una máquina virtual en Azure con información previamente recolectada, además se definieron los escenarios de evaluación. Al finalizar las pruebas se obtuvo un tiempo de procesamiento aproximado de 0.09 ms, un tamaño promedio de 192 Bytes, un formato para los temas de consulta, una velocidad de 23 consultas atendidas en 5 minutos y una precisión que fluctúa de 2 a 5 consultas erróneas. Finalmente, se concluyó que la creación de un agente virtual mejora la eficiencia y el acceso a servicios de consulta sobre procesos académicos para los estudiantes, destacando la capacidad del agente virtual de poder instruirse sobre cualquier modelo de negocio a través de una red neuronal, lo que le otorga mayores características de adaptabilidad en diversos sectores.

Palabras Clave: agente virtual, consultas, emulador, hardware, metahumano, Orange Pi, sistema, software

ABSTRACT

In the current scenario of higher education, universities must face several challenges in order to inform their students about internal procedures and services offered by the departments involved. Considering the above, and taking into account that ESPOL has several information channels that may present difficulties, this project focused on establishing the design and prototype of a virtual agent based on metahumans to establish points of consultation of internal processes of the university, for this several objectives were defined including the development of the metahuman, identify the most consulted processes, outline a physical structure and perform the respective tests. Likewise, the methodology to be used was determined with their respective tools and technologies. For the definition of the metahuman, Metahuman Creator, Quixel Bridge Pluggin, Unreal Engine, Mixamo and Audio2Face were implemented. For the training of the agent, a neural network was configured in a virtual machine in Azure with previously collected information, and the evaluation scenarios were also defined. At the end of the tests, an approximate processing time of 0.09 ms and an average size of 192 Bytes, a format for query topics, a speed of 23 queries served in 5 minutes and an accuracy fluctuating from 2 to 5 erroneous queries were obtained. Finally, it was concluded that the creation of a virtual agent improves efficiency and access to consultation services on academic processes for students, highlighting the capacity of the virtual agent to be able to learn about any business model through a neural network, which gives it greater adaptability characteristics in various sectors.

Keywords: emulator, hardware, metahuman, Orange Pi, queries, software, system, virtual agent

ÍNDICE GENERAL

RESUMEN	i
ABSTRACT	iii
ABREVIATURAS	vii
SIMBOLOGÍA	ix
INDICE DE FIGURAS	ix
INDICE DE TABLAS	xi
INDICE DE CÓDIGOS DE PROGRAMA	xiii
1 INTRODUCCIÓN	1
1.1 Descripción del problema	2
1.2 Justificación del problema	3
1.3 Objetivos	4
1.3.1 Objetivo general	4
1.3.2 Objetivos específicos	4
1.4 Alcance y limitaciones	4
1.5 Marco teórico	5
1.6 Estado del arte	6
2 METODOLOGÍA	9
2.1 Metahuman Creator	10
2.2 Unreal Engine	11
2.3 Quixel Bridge Pluggin	12
2.4 Audio2face	13
2.5 Mixamo	13

2.6	Voz a texto	13
2.7	Procesamiento del Lenguaje Natural	13
2.8	Texto a Voz	14
2.9	Microsoft Azure	14
2.10	Orange Pi 5B	15
2.11	Panel de Visualización	15
2.12	Descripción de escenarios	16
2.12.1	Despliegue del metahumano en emulador en la PC	16
2.12.2	Despliegue del metahumano en Orange Pi	17
3	ANÁLISIS DE RESULTADO	19
3.1	Base de datos para consultas y respuestas	19
3.2	Tiempo de respuesta	22
3.3	Velocidad y precisión de procesamiento de consultas	25
4	CONCLUSIONES Y LINEAS FUTURAS	29
4.1	Conclusiones	29
4.2	Recomendaciones	30
4.3	Trabajo Futuro	31
	BIBLIOGRAFÍA	33
	APÉNDICES	34
A	Esquema para definir las consultas en la red neuronal del sistema	37
B	Requerimientos y configuración de Android para Unreal Engine	37
C	Lanzamiento y empaquetado para dispositivos Android	39
D	Costo de implementación del prototipo	40

ABREVIATURAS

ESPOL Escuela Superior Politécnica del Litoral

FIEC Facultad de ingeniería en electricidad y computacion

SIMBOLOGÍA

NLP Natural language processing

TTS Text to Speech

STT Speech to Text

ÍNDICE DE FIGURAS

2.1 Arquitectura del sistema	9
2.2 Diagrama de tecnologías para configuración del metahumano	10
2.3 Entorno de creación y configuración del metahumano	11
2.4 Configuraciones del metahumano en Unreal Engine	12
2.5 Configuración en Quixel Bridge para admisión del metahumano	12
2.6 Conexión física de los componentes del hardware	15
2.7 Sistema implementado en emulador en la PC	16
2.8 Sistema implementado en hardware	17
3.1 Tiempo de ejecución de petición al sistema	23
3.2 Tiempos obtenidos por petición en la Orange Pi	23
3.3 Tiempos obtenidos por petición en la PC	24

ÍNDICE DE TABLAS

3.1	Modelo de consultas y respuestas del agente virtual	20
3.2	Tiempo total de procesamiento del sistema	25
3.3	Resultados de la velocidad sobre una misma consulta: préstamo de un libro	26
3.4	Resultados de la velocidad y precisión sobre 5 temas seleccionados	27
1	Costo de implementación del proyecto	40

ÍNDICE DE CÓDIGOS DE PROGRAMA

1	Esquema para definir consulta	37
---	-------------------------------	-------	----

CAPÍTULO 1

1. INTRODUCCIÓN

En la actualidad, el uso de la tecnología ha posibilitado el desarrollo de distintas herramientas que permiten gestionar y agilizar procesos en diversos ámbitos de la sociedad. Esto ha revolucionado la forma en que las instituciones educativas administran la comunicación y conexión con sus estudiantes y personal laboral. En este contexto, el diseño de un agente virtual basado en metahumanos surge como una solución innovadora cuyo objetivo es agilizar de manera eficiente la resolución de consultas que posean los estudiantes universitarios de ESPOL. Uno de los mayores desafíos de las universidades es brindar, de forma adecuada, la amplia gama de información sobre los procesos internos que éstos poseen. En este sentido, la creación de un agente virtual que integre características de un metahumano permitirá una experiencia más personalizada de tal forma que se mejore la interacción con los usuarios. El principal objetivo del presente trabajo es poder mejorar la eficiencia y la satisfacción de los estudiantes de la universidad brindando un canal de comunicación efectivo a la vez que reduce la carga de trabajo del personal administrativo. Durante la elaboración de este informe, se explorará en detalle el proceso de diseño, desarrollo e implementación del respectivo prototipo considerando aspectos técnicos, estéticos y funcionales. Asimismo, se analizará los posibles beneficios que se generen en términos de eficiencia, experiencia de usuario y mejora de la comunicación.

1.1 Descripción del problema

En el actual escenario de la educación de nivel superior las universidades deben afrontar diversos desafíos para lograr informar a sus estudiantes acerca de los procedimientos internos y servicios ofrecidos por los departamentos involucrados, puesto que de esta forma la institución brinda al alumnado una mayor confianza y participación, el poder de tomar decisiones informadas y aprovechar los recursos disponibles al máximo, ofreciendo así una trayectoria académica de excelencia. Dichos desafíos surgen debido a la complejidad existente en las operaciones de la universidad y del aumento de la demanda de información solicitado por el alumnado. Entre dichos desafíos el que más se destaca es la falta de un sistema adecuado de consulta y resolución de dudas, el cual a su vez se relaciona con:

- **Comunicación fragmentada:** La universidad cuenta con diversos canales de comunicación para establecer interacción con los estudiantes, tales como redes sociales, correo electrónico, portal web y oficinas físicas. La diversidad mencionada puede generar una comunicación fragmentada e ineficaz, dado que el alumnado puede presentar dificultades entre los diferentes puntos de contacto.
- **Tiempo de respuesta extenso:** a menudo los estudiantes deben esperar un tiempo de respuesta o asistencia prolongado para obtener una solución a un requerimiento específico, generando frustración y un aumento de los índices negativos en la experiencia de usuario.
- **Limitación en la experiencia de usuario:** los sistemas actuales resultan poco amigables, con escasa personalización e ineficaces ante la ausencia de interacción y respuestas concretas a las dudas expuesta por los estudiantes.
- **Carga laboral administrativa:** el personal sufre un aumento en las actividades laborales al tener que resolver los cuestionamientos rutinarios y al ejecutar actividades repetitivas. Lo anterior disminuye el recurso humano que se puede emplear en actividades más complejas, aumenta los índices de error debido al factor humano y en consecuencia afecta la calidad de respuestas brindadas a los usuarios.

1.2 Justificación del problema

ESPOL posee diversos canales informativos para la resolución de dudas o consultas, sin embargo, estos pueden no satisfacer de manera adecuada las problemáticas de los estudiantes considerando el volumen de información que posee. Dado lo anterior, se pretende diseñar un agente virtual que sirva como un punto de consulta rápido y eficiente fundamentado en las siguientes razones:

- Mejorar la experiencia de los estudiantes: ofreciendo un nivel de interacción más cercano al humano y de forma personalizada al brindar una rápida resolución y precisión.
- Uso eficiente de recursos: reduciendo la carga laboral administrativa de la universidad, permitiéndoles eliminar tareas reiterativas y realizar un mayor enfoque en actividades más demandantes.
- Respuestas de calidad: proporcionando respuestas más consistentes y precisas a consultas habituales sobre información general o de procesos internos al disminuir los errores o imprecisión humana, lo cual a su vez garantiza la integridad de los datos y aumenta la confiabilidad de la universidad.
- Escalabilidad y flexibilidad: la implementación de un agente virtual permite la escalabilidad y personalización del proceso de interacción según las necesidades de los estudiantes al permitir agregar y adaptar diversas funcionalidades para mejorar la gestión de la información.
- Competitividad: al utilizar tecnología de vanguardia se genera una mayor atracción en los postulantes, puesto que se ofrece un servicio moderno y eficiente durante sus estudios en ESPOL.

1.3 Objetivos

1.3.1 Objetivo general

Desarrollar un agente virtual basado en metahumanos a través de la implementación de flujo conversacional para la generación de respuestas automatizadas a las consultas realizadas por estudiantes sobre procesos internos de ESPOL.

1.3.2 Objetivos específicos

- Identificar al menos 8 procesos internos de la universidad que los estudiantes desconocen para el entrenamiento adecuado del agente virtual.
- Bosquejar una estructura física que permita la interacción entre los usuarios y el agente virtual.
- Realizar pruebas exhaustivas del agente virtual estableciendo escenarios y parámetros respectivos.

1.4 Alcance y limitaciones

El proyecto incluirá el diseño y desarrollo de un agente virtual abarcando la creación de un prototipo con su respectiva programación para la debida interacción con los usuarios. En primeras instancias el metahumano será instruido con información general sobre procesos de la universidad, información general sobre FIEC y Telemática.

Se llevarán a cabo pruebas exhaustivas con el propósito de evaluar la precisión de la información y la eficacia brindada en las soluciones mencionadas por el metahumano. Luego de ello, se realizarán las debidas modificaciones en base a los resultados obtenidos. La implementación piloto del prototipo se ejecutará con el propósito inicial de atender consultas específicas en base a la información previamente recolectada y programada.

En cuanto a las limitaciones, se destaca la complejidad existente de algunos procesos internos, de forma que el metahumano no será capaz de abordar todos los escenarios que puedan aparecer en su implementación final. Asimismo, la precisión de los datos

otorgados a los usuarios se verá afectada por las distintas actualizaciones en los sistemas que alojan la información de la universidad.

Es necesario destacar que el agente virtual no simboliza una solución estática, por el contrario, este puede adaptarse y expandirse de tal forma que aborde diversos tipos de información. Sin embargo, las adaptaciones requeridas para cumplir lo mencionado demandarán una mayor cantidad de recursos y tiempo.

Adicionalmente, se especifica que la red neuronal en donde se definirá la información necesaria para que el metahumano sea capaz de resolver las dudas de los usuarios estará alojada en Azure. Dado lo anterior, se deben considerar aspectos como ancho de banda adecuado, baja latencia, disponibilidad del servicio, temas de seguridad y limitaciones en la escalabilidad para asegurar una correcta funcionabilidad del sistema.

1.5 Marco teórico

El proyecto de crear un agente virtual de tipo metahumano con la capacidad de resolver consultas relacionadas con los procesos realizados en ESPOL tiene como fundamento el uso de tecnologías avanzadas, las mismas que son primordiales en el campo de la comunicación humano-maquina dado que permiten que las aplicaciones comprendan y generen lenguaje natural, característica esencial en la creación de agentes virtuales.

Voz a texto

La primera tecnología a utilizarse es el reconocimiento de voz (Speech to text), la cual se encarga de la conversión de la voz humana a texto a través del uso de lingüística computacional, es decir, usa algoritmos lingüísticos que se encargan de clasificar las señales auditivas de las palabras habladas. En resumen, cuando una persona habla genera vibraciones, las cuales son captadas para luego traducirlas a lenguaje digital mediante el uso de convertidores de una señal analógica, como lo es la onda del sonido, a una señal digital (Trivedi et al., 2018).

Texto a voz

Por otra parte, otra de las tecnologías usadas es la generación de voz a partir de texto (Text to speech), esta se encarga de la síntesis de texto a voz generando una salida audible que puede ser comprendida por los seres humanos. Ahora bien, la salida de voz se genera mediante el uso de algoritmos y modelos de síntesis de voz para transformar el texto procesado en señales de audio, representando el habla humana (Trivedi et al., 2018).

Procesamiento del lenguaje natural

Ahora bien como núcleo conversacional se utiliza algoritmos de deep learning y redes neuronales. En este caso se implementa el procesamiento natural del lenguaje (Natural language processing), el cual proporciona a los sistemas computacionales la capacidad de interpretar, manipular y comprender el lenguaje humano. NLP funciona realizando el preprocesamiento de las palabras a través de la creación de tokens. Asimismo, los datos preprocesados son usados para crear los modelos NLP (Nadkarni et al., 2011).

1.6 Estado del arte

En el pasar del tiempo se han utilizado agentes virtuales conversacionales en diferentes áreas, un ejemplo claro de eso se puede observar en el trabajo de investigación de (Giudici et al., 2022), el cual consistía tener una IA que pueda recibir peticiones por voz desde un micrófono central y manejar ciertos sensores de la casa similar a lo que realiza Alexa con los productos IoT.

Por otra parte, los agentes virtuales con procesamiento natural de lenguaje se han utilizado en la medicina, un ejemplo se puede visualizar en la investigación de (Boumans et al., 2022), que consistió en un estudio de última generación con agentes virtuales que poseen un diseño 3d y la capacidad de realizar conversaciones verbales con un grupo de adultos mayores que padecen amnesia.

De igual manera, los agentes virtuales se han utilizado en otras áreas de la medicina, eso se puede observar en el estudio de (Rudschies et al., 2023), en el cual se logra observar cómo se utiliza modelos de procesamiento de lenguaje debido a su capacidad para mejorar la accesibilidad, la disponibilidad y respaldar los tratamientos clínicos.

Ahora bien, en época de pandemia se utilizó este tipo de agentes virtuales como chatbot para poder dar un diagnóstico en la predicción de síntomas en base a conversaciones textuales con personas contagiadas de Covid-19 (VARELA-TAPIA et al., n.d.).

Los agentes virtuales también se han definido como una herramienta en el ámbito social, tal y como lo demuestra el estudio de (Tanaka and Nakamura, 2021), donde se define un agente virtual basado en un modelo 3D realista brindado por la plataforma Greta y generado con caracteres Autodesk. Este agente utiliza tecnología Text-to-Speech con el objetivo de entrenar a los usuarios considerando los rasgos autistas que estos puedan tener.

Debido a los avances tecnológicos relacionados a realidad virtual se han generado diversos instrumentos de soporte para diversas situaciones, lo mencionado se puede observar en el estudio realizado por (Ye et al., 2021), en el cual se diseña un agente virtual con reconocimiento de posición, permitiendo una navegación asistida en realidad virtual.

En el ámbito comercial, la implementación de estos agentes ha permitido establecer sistemas de recomendaciones para diversos procesos, un ejemplo de esto se visualiza en el estudio que llevaron a cabo (Anastasia et al., 2021), donde se implementa una metodología de diseño centrado en el usuario y natural conversational framework para definir una experiencia de usuario basada en la ciencia conversacional.

Uno de los mayores intereses relacionados a estos mecanismos son sus implementaciones en el ámbito educativo, en los estudios realizados por (Zojaji and Peters, 2019) y (Benitez Sandoval et al., 2022), se implementa tecnología de realidad virtual para crear agentes virtuales con la capacidad de dar soporte tanto en actividades sociales como en el aprendizaje de los estudiantes.

CAPÍTULO 2

2. METODOLOGÍA

En este capítulo se describirá la arquitectura tecnológica y física utilizada en el proyecto, la cual se puede visualizar en Fig. 2.1, además de una breve explicación de cada herramienta y cómo estas se encuentran relacionadas con el presente trabajo. La información necesaria para entrenar el sistema se obtiene de diversas fuentes de información, entre las cuales se destacan: experiencias propias, información brindada por docentes, personal administrativo y dudas comunes extraídas del apartado de Preguntas Frecuente del sitio web de FIEC. Para un mejor entendimiento la arquitectura se dividirá en dos partes: software y hardware.

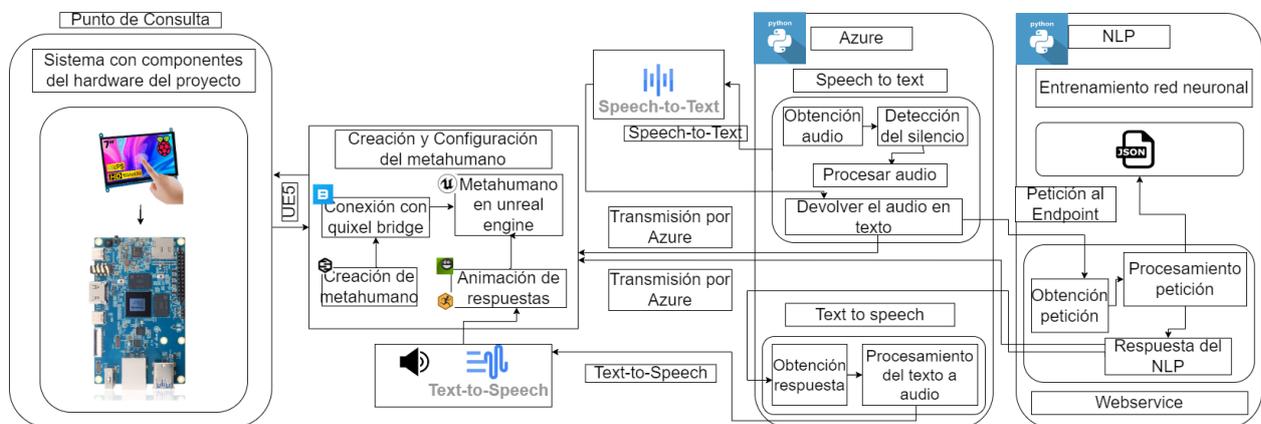


Figura 2.1: Arquitectura del sistema

• Software

Con el fin de elaborar la definición del metahumano que se presentará a los usuarios, se ha establecido una arquitectura que integra tres herramientas fundamentales: Metahuman creator, Quixel bridge y Unreal engine, las cuales pueden ser visualizadas en la Fig. 2.2. Se procede a describir cada una de ellas:



Figura 2.2: Diagrama de tecnologías para configuración del metahumano

2.1 Metahuman Creator

Metahuman Creator es un framework completo y gratuito basado en la nube ofrecido por la empresa Epic Games, el cual permitirá la creación y el diseño del metahumano definido para la implementación del presente proyecto. Posteriormente, se deberá exportar como archivo Metahuman que contiene la información requerida por unreal engine. En la Fig. 2.3 se puede visualizar el entorno que ofrece la herramienta para las actividades mencionadas.

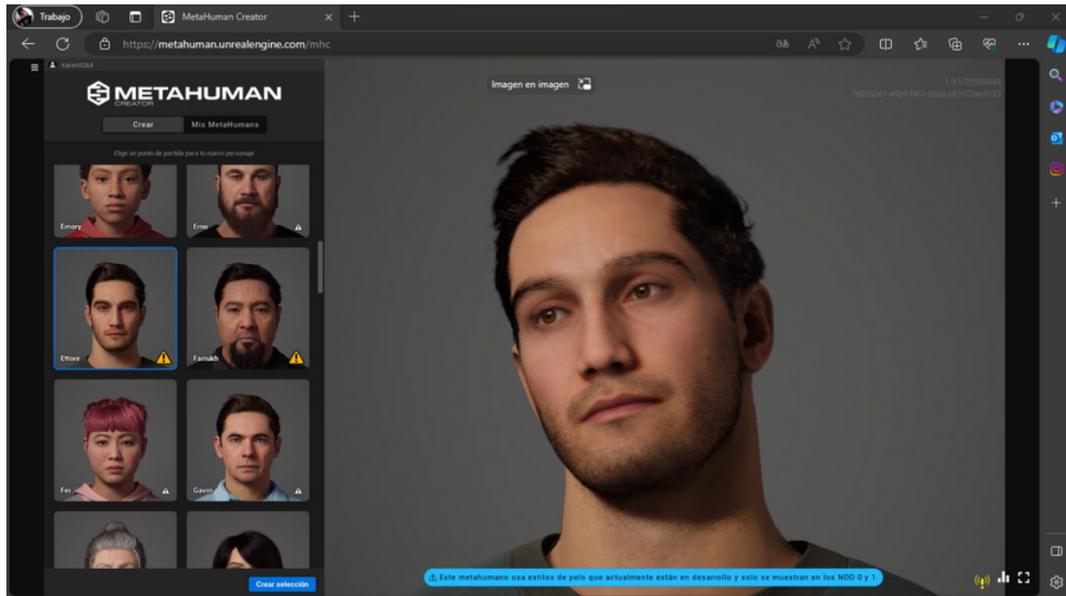


Figura 2.3: Entorno de creación y configuración del metahumano

2.2 Unreal Engine

Unreal Engine es un motor de juegos desarrollado por la empresa Epic Games, utilizado como una potente herramienta para crear videojuegos, componentes de realidad virtual y aumentada, entre otros. Este instrumento permitirá crear el ambiente para el metahumano, para ello se deberá importar el archivo MetaHuman previamente creado y se deberá realizar las configuraciones necesarias para que el metahumano sea admitido en el entorno. Lo mencionado con anterioridad se encuentra reflejado en la Fig. 2.4.

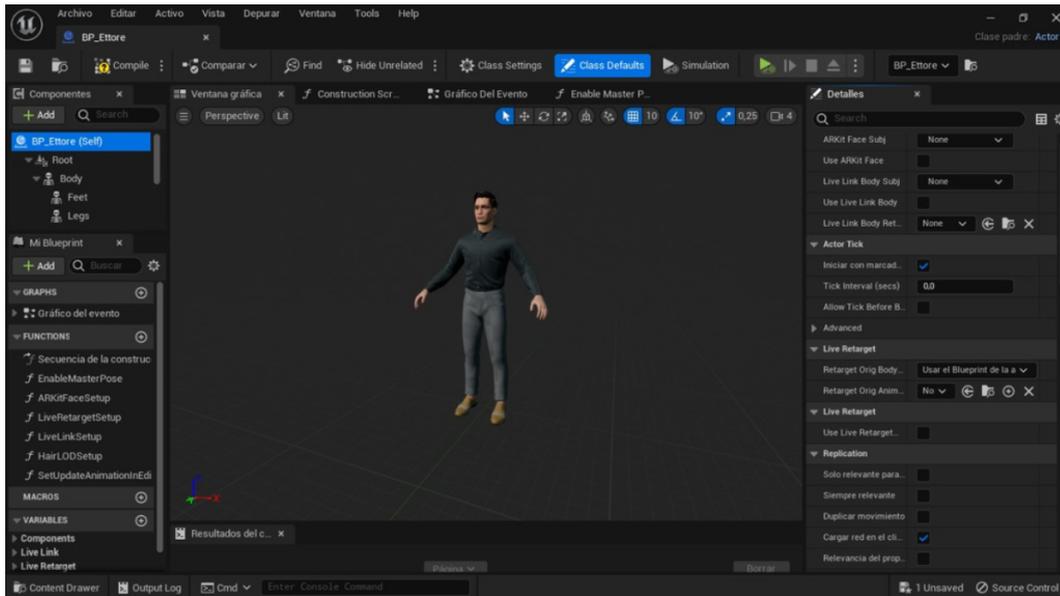


Figura 2.4: Configuraciones del metahumano en Unreal Engine

2.3 Quixel Bridge Pluggin

Quixel Bridge Pluggin es una herramienta que permite conectar el resultado del Metahuman Creator y Unreal Engine, la cual facilitará la importación y debida configuración del metahumano en el proyecto de unreal engine. Además, se logrará realizar una mejor personalización dado que se utilizan los activos descargados de Megascans. La configuración necesaria para la conexión se muestra en la Fig. 2.5

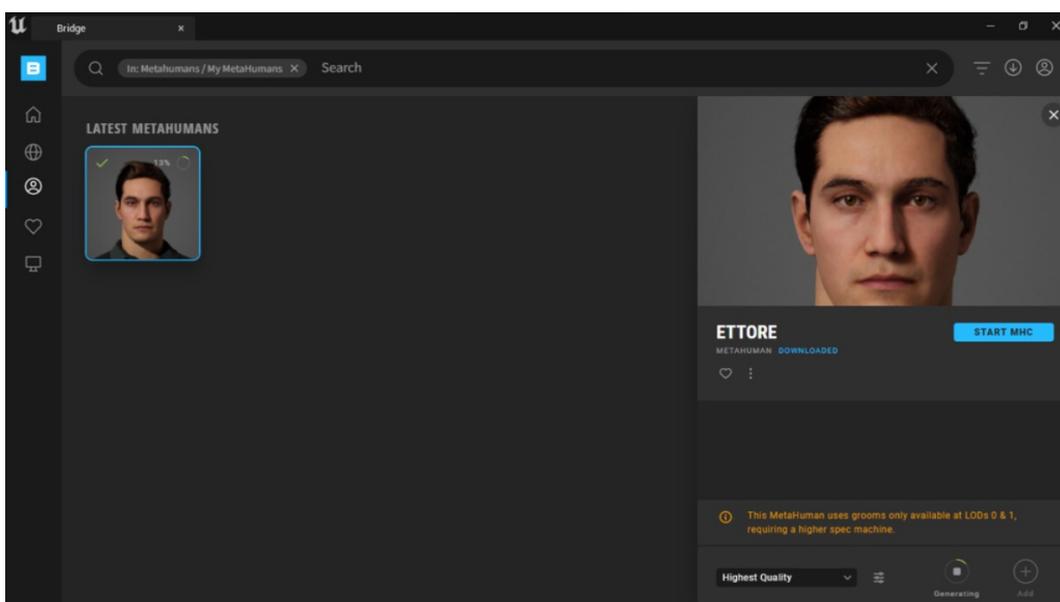


Figura 2.5: Configuración en Quixel Bridge para admisión del metahumano

2.4 Audio2face

La herramienta Omniverse Audio2Face es una aplicación desarrollada por Nvidia e impulsada por Inteligencia Artificial, la cual permite a través del uso de archivos en formato de audio generar animaciones faciales. Esta permitirá establecer las animaciones correspondientes para cada respuesta brindada por el metahumano de este trabajo.

2.5 Mixamo

La plataforma en línea Mixamo ofrece una diversidad de herramientas para realizar animación 3D de una forma más rápida y accesible, simplificando el proceso de animación de personajes 3D al usar tecnologías como motion capture y rigging automático. Para realizar la animación corporal del metahumano se utiliza esta herramienta en conjunto con Unreal Engine generando un agente más interactivo.

2.6 Voz a texto

Esta es una solución que permite la conversión de una señal analógica, como lo es las ondas sonoras producidas al hablar, a una señal digital entendible por el computador.

2.7 Procesamiento del Lenguaje Natural

El Procesamiento del Lenguaje Natural (NLP, por sus siglas en inglés) es una rama de la inteligencia artificial que se ocupa de la interacción entre las computadoras y el lenguaje humano. En el contexto de este proyecto, el NLP desempeña un papel crucial al permitir la interacción del usuario con el metahumano a través del lenguaje natural como comandos de voz o preguntas.

2.8 Texto a Voz

La conversión de texto a voz es una tecnología que convierte el texto escrito en discurso audible. En el caso de este proyecto, la implementación de Text to Speech es utilizada para dotar al metahumano de la capacidad de hablar permitiendo una experiencia más inmersiva y realista.

2.9 Microsoft Azure

La plataforma de servicios en la nube proporcionada por la empresa Microsoft se define como un conjunto de servicios de computación en la nube para implementar y administrar aplicaciones y servicios. El uso de esta herramienta en el proyecto permite realizar la conversión de voz a texto a través de la aplicación de sus servicios, asimismo como el levantamiento de una máquina virtual para la ejecución de la API establecida.

• Hardware

En el marco del presente proyecto integrador, resulta imprescindible explorar y describir los componentes esenciales que conforman la infraestructura técnica para la implementación del agente virtual. En este contexto, el hardware ejerce un papel principal en la materialización y visualización del asistente. En esta sección, se realiza un enfoque en dos elementos fundamentales: la Orange Pi 5B y el panel de visualización, cuya conexión se observa en la Fig. 2.6. La elección de ambos dispositivos se realizó en base a criterios fundamentales como rendimiento y capacidad para soportar el sistema definido, a la vez que brinda una agradable experiencia al usuario. A continuación, se brindan más detalles sobre los dispositivos mencionados.



Figura 2.6: Conexión física de los componentes del hardware

2.10 Orange Pi 5B

La Orange Pi 5B es una placa de desarrollo de bajo costo basada en el sistema en un chip (SoC) Allwinner H616. Su capacidad de procesamiento y E/S la hace adecuada para aplicaciones que requieren un hardware compacto pero capaz, como podría ser el caso de ejecutar el servidor de metahumano.

2.11 Panel de Visualización

El panel de visualización es un componente crucial en la experiencia de usuario. Puede ser un monitor, pantalla de realidad virtual o cualquier dispositivo de salida que permita al usuario interactuar y visualizar el entorno del metahumano. La elección del panel de visualización puede depender de la naturaleza específica de la aplicación, como juegos, experiencias de realidad virtual, o interfaces gráficas más tradicionales.

2.12 Descripción de escenarios

Para evaluar el funcionamiento del sistema se pretende establecer dos escenarios que permitan analizar el desempeño del metahumano tanto en un escenario basado solo en un emulador en la PC y un segundo escenario que incluye el uso de la Orange Pi como hardware.

2.12.1 Despliegue del metahumano en emulador en la PC

Este proceso se inició con la definición de cuáles serían los temas de consulta, identificando a los actores y áreas claves. Se realiza la definición de la red neuronal y la estructura del sistema en unreal engine. Asimismo, se determinó la estética y alcance de la animación del metahumano. En este escenario, la virtualización del agente y su capacidad para responder las consultas se evalúan a través de la generación de un emulador en la PC. El metahumano puede ser visualizado en la Fig. 2.7.



Figura 2.7: Sistema implementado en emulador en la PC

2.12.2 Despliegue del metahumano en Orange Pi

Además de las acciones descritas en el primer escenario, se añade la generación de una APK para sistemas Android en Unreal Engine, por lo cual se realizan las respectivas configuraciones para el proceso de packaging. En la Orange Pi se realiza la instalación del sistema operativo Android 13 para ejecutar el APK generado y levantar el sistema del agente virtual, tal y como se muestra en la Fig. 2.8.

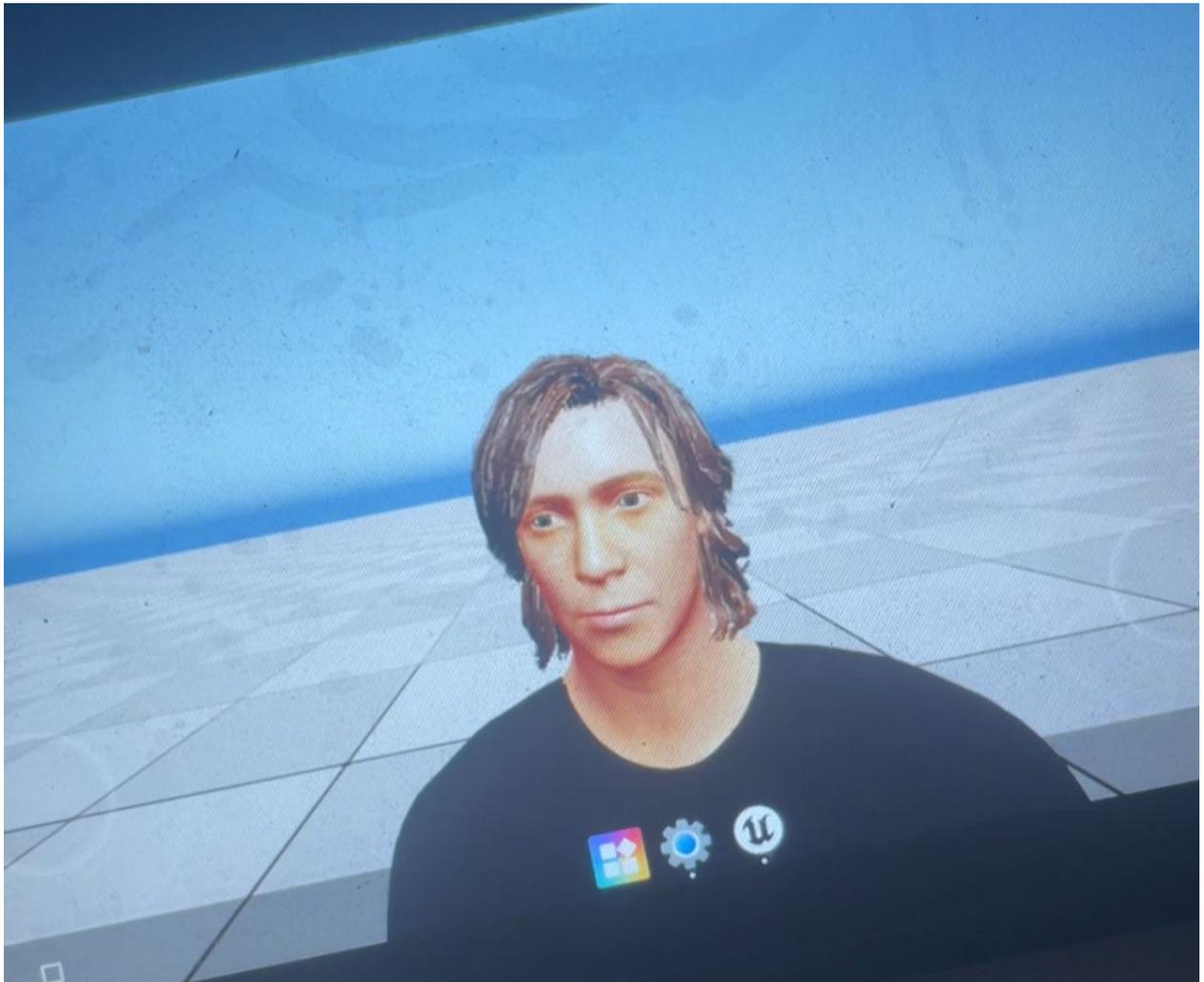


Figura 2.8: Sistema implementado en hardware

CAPÍTULO 3

3. ANÁLISIS DE RESULTADO

Durante el desarrollo del presente trabajo se ha explorado e incorporado diversas tecnologías emergentes que permitieran mejorar la eficiencia y acceso a servicios de consulta académicos; para lo cual se realizó un mayor enfoque en desarrollar un agente virtual que posea la capacidad de resolver consultas relacionadas a estos procesos, brindando una experiencia más agradable para los usuarios de ESPOL. En el siguiente capítulo, se pretende detallar y analizar los resultados obtenidos de la simulación e implementación del agente virtual para resolver consultas sobre procesos internos de la universidad. Para lograr lo mencionado se establecieron parámetros como tiempo de respuesta, velocidad y eficiencia de procesamiento de consultas, estabilidad de sistema en tiempos prolongados y precisión de las respuestas sobre consultas planteadas. De esta forma, no solo se valida el funcionamiento técnico del sistema, sino que se brinda un análisis que permita identificar mejoras futuras.

3.1 Base de datos para consultas y respuestas

Al brindar respuestas erróneas o no responder ante una consulta, puede generar confusión o resultados inesperados y en consecuencia esto termina afectando la experiencia de los usuarios. Para asegurar la calidad de las respuestas brindadas, se estableció un formato de respuesta basado en la información recolectada de fuentes primarias como: sitio web de la facultad (FIEC), sitio web de la universidad, docentes, personal administrativo de secretaría de FIEC y coordinador de la carrera de ingeniería en Telemática. En la Tabla 3.1, se logra visualizar los temas de consulta y sus respectivas respuestas.

Tabla 3.1: Modelo de consultas y respuestas del agente virtual

Tema de consulta	Respuesta
Préstamo de un libro	Para prestar un libro, necesita acercarse a la biblioteca, elegir el libro a prestar y acercarse a recepción con el libro y el carnet, para que el encargado lo ayude con su solicitud.
Préstamo de un cubículo	Para prestar un cubículo, necesita acercarse a recepción con el carnet, para que el encargado lo ayude con su solicitud.
Encargado del laboratorio de Redes avanzadas	El encargado del Laboratorio de redes avanzadas es, el Ingeniero Carlos Cedeño.
Coordinador de la carrera	El coordinador de la carrera de telemática es el PhD. José Córdova García, con el cual puedes comunicarte a través de los siguientes correos electrónicos jecordov@espol.edu.ec , jecordova@fiec.espol.edu.ec , telematica@fiec.espol.edu.ec
Acreditaciones que posee la carrera	La carrera de ingeniería en telemática posee actualmente dos acreditaciones ABET y EURACE las cuales aseguran que los estudiantes y graduados politécnicos de telemática cumplen con los estándares globales de calidad exigidos por ambas acreditadoras.

Proceso de anulación de semestre

Para realizar el proceso de anulación del semestre debes dirigirte a la página web de la facultad www.fiec.espol.edu.ec, en el apartado de enlaces de interés encontrarás la sección información para el estudiante donde deberás desplegar aquella que indica Instructivo del proceso de anulación de materia o materias del periodo académico. En este apartado te indican todas las consideraciones sobre el proceso y se adjunta la solicitud necesaria para el mismo. Finalmente, la solicitud realizada y documentos deben ser enviados a los siguientes correos: subdecanato@fiec.espol.edu.ec y secretaria-subdecanato@fiec.espol.edu.ec

Cambio de carrera

Para gestionar el cambio de carrera debes verificar primero si te encuentras dentro del plazo estipulado para realizarlo, dicha información la puedes encontrar en el calendario académico institucional publicado en la página web de la universidad. Luego deberás dirigirte a www.polimatico.espol.edu.ec y realizar la solicitud de cambio de carrera. Debes considerar que si el cambio es hacia una carrera que pertenece a un área de conocimiento distinta a la que actualmente te encuentras, deberás rendir los exámenes que la Oficina de Admisiones determine.

Oficinas de profesores	Las oficinas de los docentes, tanto principales como técnicos, se encuentran distribuidas en los diversos bloques de FIEC. Para conocer, de forma exacta, en que bloque y oficina se encuentra un docente específico puedes dirigirte a la página web de la facultad www.fiec.espol.edu.ec , al desplegar el menú FIEC encontrarás el apartado Directorio electrónico donde se describe el edificio, oficina, teléfono y su respectivo tipo, además del cargo de los distintos docentes de la facultad.
------------------------	--

Considerando lo expresado en la Tabla 3.1, el sistema posee actualmente 8 tópicos relacionados a las consultas que poseen los estudiantes. Dado que la red neuronal puede estar en constante cambio es posible agregar más tags para aumentar la capacidad del metahumano, por lo cual se adjunta un formato de los parámetros que se deben definir en el Apéndice 1.

3.2 Tiempo de respuesta

El tiempo de respuesta de un sistema, de forma general, resulta un aspecto crítico y de mayor importancia dado que influye en la satisfacción del usuario al utilizar el agente, permite aumentar la productividad en el entorno, mejorar la calidad del servicio entregado, entre otros. Para lograr determinar el tiempo de respuesta se realiza la solicitud tal y como se muestra en la Fig. 3.1. Para el posterior análisis se considerará el tiempo de “Process” y de “Download” mostrados en las Fig. 3.2 y Fig. 3.3, con un mayor enfoque en el tiempo de “Process”. También es necesario destacar los tiempos que se pueden observar en la Fig. 3.2 como “TCP Handshake”, que señala el momento en que inicia la conexión TCP entre cliente y servidor y “Transfer Start” que indica el momento en que inicia la transferencia de los datos entre cliente y servidor después de establecer la conexión TCP. Asimismo, se puede visualizar que el tiempo de download es mayor para la petición de la Orange Pi en comparación a la ejecutada en el emulador, 11.67 ms

y 7.33 ms respectivamente, esto debido a que el proceso de transmisión del paquete se efectúa sobre un enlace de conexión inalámbrica para el hardware, mientras que en la PC dicha transmisión se produce sobre un enlace de conexión alámbrico, lo cual produce diferencias de velocidad de transmisión, interferencia, latencia y ancho de banda compartido. Por otro lado, el tiempo total es el mismo para ambos escenarios, por lo cual, para el análisis se unificarán los resultados en una tabla.

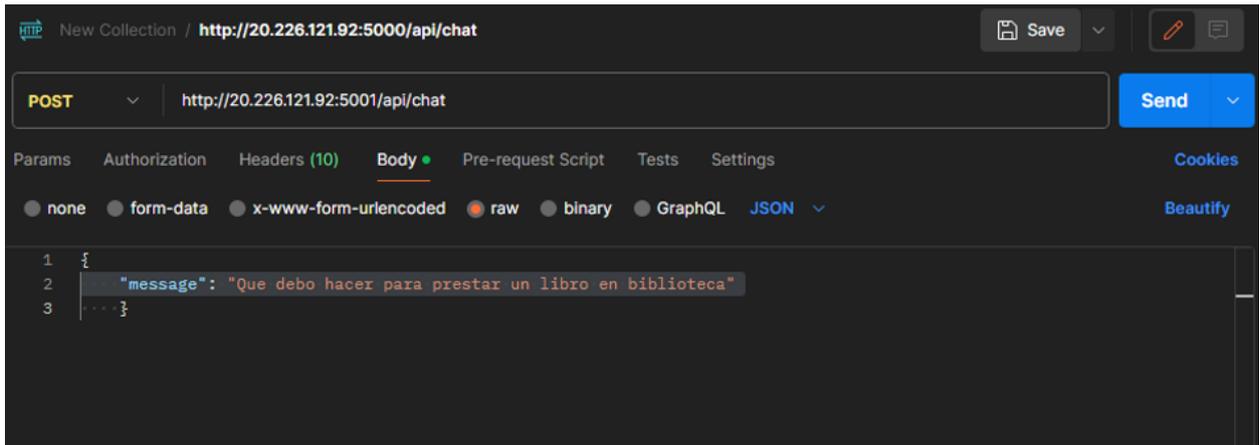


Figura 3.1: Tiempo de ejecución de petición al sistema

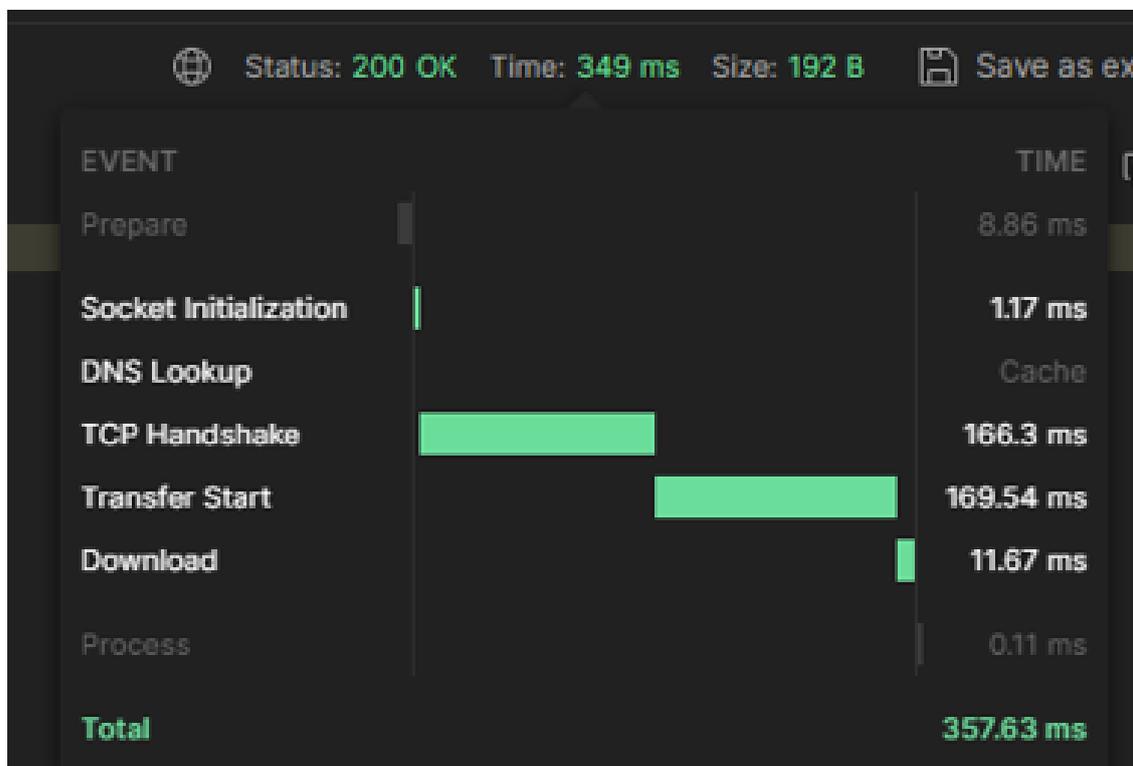


Figura 3.2: Tiempos obtenidos por petición en la Orange Pi

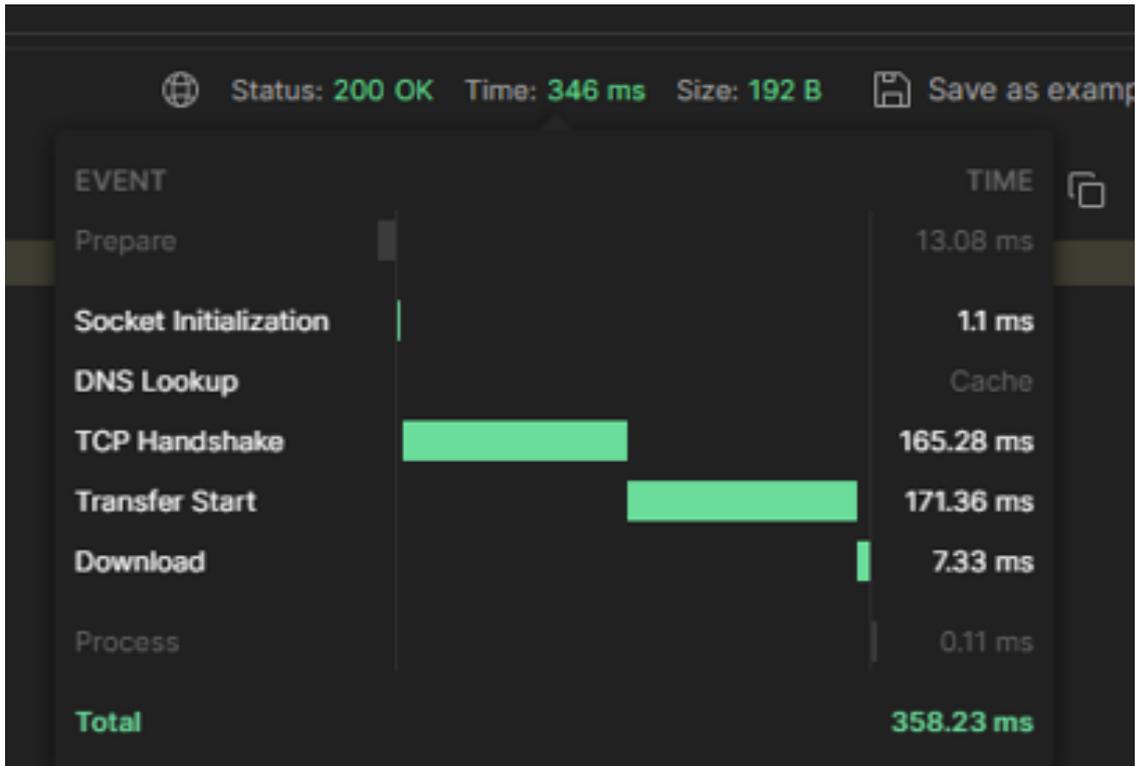


Figura 3.3: Tiempos obtenidos por petición en la PC

Este mismo procedimiento se realiza para 4 consultas más de las definidas en la Tabla 3.1 y sus resultados se describen en la Tabla 3.2.

Tabla 3.2: Tiempo total de procesamiento del sistema

Tema de consulta	Tamaño de respuesta (Bytes)	de Tiempo de procesamiento (process)	de Tiempo de download
Préstamo de un libro en biblioteca	192	0.11 ms	7.33 ms
Préstamo de un cubículo en biblioteca	195	0.07 ms	7.98 ms
Encargado del laboratorio de Redes Avanzadas	188	0.12 ms	10.73 ms
Acreditaciones que posee la carrera de telemática	196	0.05 ms	17.72 ms
Proceso de anulación de semestre	191	0.09 ms	21.55 ms

Tal y como se observa en la Tabla 3.2, se logra visualizar que no se guarda mucha relación entre el tamaño de la respuesta y los tiempos relacionados a esta, esto debido a que el procesamiento de la red neuronal se aloja en la nube con una conexión estable.

3.3 Velocidad y precisión de procesamiento de consultas

La velocidad y precisión son parámetros esenciales en el procesamiento de las consultas, puesto que influyen de forma directa en la capacidad que posee el sistema para procesar grandes volúmenes de solicitudes sin disminuir o afectar la calidad de las respuestas. Para el cálculo de la velocidad, se considera la cantidad de consultas procesadas en un intervalo de tiempo específico, mientras que la eficiencia se obtiene considerando cuántas de esas consultas se atendieron de forma satisfactoria en el mismo periodo.

Tabla 3.3: Resultados de la velocidad sobre una misma consulta: préstamo de un libro

Consulta	Tiempo de consulta	velocidad
Pasos para prestar un libro	11.55 segundos	312 consultas por hora
¿Cómo puedo prestar un libro en la biblioteca?	11.72 segundos	300 consultas por hora
¿Cómo realizo el préstamo de un libro?	11.59 segundos	312 consultas por hora
Procedimiento para prestar un libro en la biblioteca?	11.91 segundos	300 consultas por hora
¿Cuál es la forma en la que puedo prestar un libro?	11.72 segundos	300 consultas por hora

En la Tabla 3.3 se visualizan los resultados de la velocidad obtenida al realizar de distintas formas la consulta relacionada al préstamo de un libro en la biblioteca, obteniendo en promedio 25 consultas en 5 minutos para este tópico.

Tabla 3.4: Resultados de la velocidad y precisión sobre 5 temas seleccionados

Tema de consulta	Tiempo promedio de consulta	Velocidad	Eficiencia
Préstamo de un libro en biblioteca	11.70 segundos	312 consultas por hora	23 de 26 consultas son correctas
Préstamo de un cubículo en biblioteca	10.84 segundos	336 consultas por hora	24 de 28 consultas son correctas
Encargado del laboratorio de Redes Avanzadas	9.74 segundos	372 consultas por hora	26 de 31 consultas son correctas
Acreditaciones que posee la carrera de telemática	17.66 segundos	204 consultas por hora	14 de 17 consultas son correctas
Proceso de anulación de semestre	28.62 segundos	120w consultas por hora	8 de 10 consultas son correctas

Tal y como se observa en la Tabla 3.4, se obtuvo una velocidad promedio para cada tipo de consulta y una precisión que fluctúa entre 2 y 5 consultas que presentaron errores en sus respuestas, es necesario destacar que pueden darse fallos al procesar la consulta debido a variables que depende del usuario: tono de voz, velocidad del habla, ruido externo y de la conexión a internet. Asimismo se indica que la precisión en la resolución brindada por el sistema puede depender de cómo el usuario plantea la consulta para que el metahumano pueda procesarla.

CAPÍTULO 4

4. CONCLUSIONES Y LÍNEAS FUTURAS

Durante el desarrollo de los anteriores capítulos se ha explorado de manera detallada los componentes y herramientas necesarias para el desarrollo del mismo. En el siguiente capítulo se abordará los resultados e implicaciones derivadas del trabajo realizado.

4.1 Conclusiones

- Al finalizar el presente trabajo sobre el diseño de un agente virtual basado en metahumanos para establecer puntos de consulta de procesos internos de la universidad se destacan los siguientes puntos: creación de un agente virtual que permita mejorar la eficiencia y el acceso a servicios de consulta sobre procesos académicos que puedan generarse en el alumnado, definición de temas o dudas más consultadas por los estudiantes con el objetivo de capacitar al metahumano para brindar respuestas específicas y pertinentes y el análisis de los resultados obtenidos a través de la implementación y simulación del metahumano, tanto en software como en hardware, obteniendo datos sobre el tiempo de respuesta, velocidad, eficiencia relacionada, y precisión en las respuestas.
- La definición de los temas de consulta acerca de la universidad, de FIEC y de telemática permitió establecer el alcance de la fase inicial del metahumano, brindándole la capacidad de resolver las dudas que poseen los usuarios. Además, se destaca que el agente virtual puede aprender el modelo de negocio a través de una red neuronal, dándole mayores características de adaptabilidad en diversos sectores.

- El sistema ofrece un tiempo de procesamiento aproximado de 0.09 ms con un tamaño de respuesta promedio de 192 Bytes. Sin embargo, es necesario rectificar que estos valores no poseen la exactitud necesaria, ya que los mismo pueden variar según la extensión de la consulta o de la respuesta, debido a fallos en la conexión o por afectaciones en la recepción de la consulta.
- La implementación del agente virtual basado en metahumanos se presenta como una innovación en este ámbito, dado que combina el uso de tecnologías de modelamiento 3D con la inteligencia artificial para solventar y ayudar en la resolución de consultas que usualmente deben ser atendidas por el personal administrativo de la universidad.
- Asimismo los resultados obtenidos en este trabajo demuestran que el metahumano es una herramienta de gran valor con la capacidad de procesar consultas mejorando la accesibilidad a información de procesos de interés en tiempo real.

4.2 Recomendaciones

Aunque el tiempo actual de respuesta puede considerarse adecuado, es fundamental mejorar el mismo, optimizando los procesos y la eficiencia en la resolución de las consultas planteadas. Lo anterior es posible al indagar y aplicar soluciones que mejoren la capacidad de procesamiento del agente virtual, como el uso de hardware con mayores capacidades computacionales.

Realizar una mayor indagación sobre las razones que conllevan a fallos en la animación cuando se ejecuta el APK en la Orange Pi para posteriormente buscar soluciones que permitan corregir los errores y mejorar la estética del metahumano en el hardware.

Actualmente, el metahumano posee respuestas definidas para 8 tipo de consultas, por lo cual, se recomienda ampliar el alcance de las soluciones brindadas, aumentando las definiciones en la red neuronal o utilizando las bases de datos disponibles de la universidad.

4.3 Trabajo Futuro

- Como trabajo futuro se puede indagar la posibilidad de conectar las bases de datos de la universidad con el sistema del agente virtual para que el metahumano este en la capacidad de responder a cualquier consulta que realicen los estudiantes o aspirantes.
- Realizar pruebas regulares que permitan evaluar la efectividad y precisión en la ejecución del sistema al escuchar, procesar y responder las consultad mencionadas. Con lo anterior, se identificarán posibles problemas o áreas de mejora para después implementar medidas correctivas que aumenten las capacidades del metahumano.
- Investigar y explorar tecnologías emergentes, tales como sistemas operativos adecuados en base al software utilizado, plugins que permitan simplificar el proceso de definición en Unreal Engine y herramientas que posibiliten el alojamiento de la red neuronal para el adecuado funcionamiento del agente virtual.
- Optimizar el procesamiento de la red neuronal al migrar, total o parcialmente, el procesamiento desde la nube hacia la Orange Pi. Este enfoque permitiría almacenar localmente las consultas más frecuentes, eliminando así la necesidad de enviar dichas solicitudes a la nube. Como resultado, se logra una reducción significativa en los tiempos de consulta y un menor consumo de ancho de banda.
- Configurar el metahumano de tal forma que pueda implementar un aprendizaje automático, es decir, si el metahumano no posee la respuesta a una consulta planteada, este estará en la capacidad de almacenar dicha duda junto a la respuesta respectiva obtenida de las bases de la universidad.

BIBLIOGRAFÍA

- Anastasia, N., Harlili, & Putri Yulianti, L. (2021). Designing embodied virtual agent in e-commerce system recommendations using conversational design interaction. *2021 8th International Conference on Advanced Informatics: Concepts, Theory and Applications (ICAICTA)*. <https://doi.org/10.1109/ICAICTA53211.2021.9640258>
- Benitez Sandoval, E., Vasquez Rojas, D., Parada Cereceres, C., Anzueto Rios, A., Barde, A., & Billinghamurst, M. (2022). Prototyping a virtual agent for pre-school english teaching. *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. <https://doi.org/10.1109/VRW55335.2022.00180>
- Boumans, R., van de Sande, Y., Thill, S., & Bosse, T. (2022). Voice-enabled intelligent virtual agents for people with amnesia: Systematic review. *JMIR aging*, 5(2), e32473.
- Giudici, M., Crovari, P., & Garzotto, F. (2022). Candy: A framework to design conversational agents for domestic sustainability. *Proceedings of the 4th Conference on Conversational User Interfaces*, 1–8.
- Nadkarni, P. M., Ohno-Machado, L., & Chapman, W. W. (2011). Natural language processing: An introduction. *Journal of the American Medical Informatics Association*, 18(5), 544–551.
- Rudschies, C., Rings, S., Kruse, L., Schauenburg, G., Marmarshahi, H., & Zimmer, C.-N. (2023). Psychotherapy with the help of chatgpt? current technical and ethical boundaries of intelligent virtual agents.
- Tanaka, H., & Nakamura, S. (2021). Virtual agent design for social skills training considering autistic traits. *2021 43rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. <https://doi.org/10.1109/EMBC46164.2021.9630741>
- Trivedi, A., Pant, N., Shah, P., Sonik, S., & Agrawal, S. (2018). Speech to text and text to speech recognition systems-a review. *IOSR J. Comput. Eng*, 20(2), 36–43.
- VARELA-TAPIA, E. A., SANGACHA-TAPIA, L. M., ACOSTA-GUZMÁN, I. L., ORDOÑEZ-CHÁVEZ, F. E., & ACOSTA-VARELA, C. I. (n.d.). Modelo con nlp y machine learning en la predicción de síntomas en base a conversaciones textuales con personas contagiadas de covid-19.
- Ye, Z.-M., Chen, J.-L., Wang, M., & Yang, Y.-L. (2021). Paval: Position-aware virtual agent locomotion for assisted virtual reality navigation. *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. <https://doi.org/10.1109/ISMAR52148.2021.00039>
- Zojaji, S., & Peters, C. (2019). Towards virtual agents for supporting appropriate small group behaviors in educational contexts. *2019 11th International Conference on Virtual Worlds and Games for Serious Applications (VS-Games)*. <https://doi.org/10.1109/VS-Games.2019.8864528>

APÉNDICES

A Esquema para definir las consultas en la red neuronal del sistema

Código 1: Esquema para definir consulta

```
{  
  " intents": [  
    {  
      " tag": " NombreDelTag",  
      " responses": [  
        " Respuesta que dará el metahumano a la consulta planteada"  
      ],  
      " patterns": [  
        " Forma en la que el usuario puede consultar",  
        " Pueden definirse diversas formas"  
      ]  
    }  
  ]  
}
```

B Requerimientos y configuración de Android para Unreal Engine

1. Pasos para realizar configuración de Android SDK y NDK

- A. Instalar Android Studio
- B. Configurar Android Studio para usarlo por primera vez
- C. Configurar las herramientas de línea de comandos para Android
- D. Finalizar la instalación de Android Studio en su sistema operativo
- E. Configurar Android NDK

Para mayor información sobre el proceso de instalación y configuración dirigirse a la documentación de Unreal Engine o

ingresar a la siguiente url: <https://docs.unrealengine.com/5.3/en-US/how-to-set-up-android-sdk-and-ndk-for-your-unreal-engine-development-environment/>

2. Pasos para configurar Android en Unreal Engine

A. Verificar los requisitos previos para el desarrollo en Android

B. Crear un proyecto que se desee migrar a Android

C. Configurar el Editor Unreal para Android

1. Dirigirse a el apartado de “Editar”
2. Abrir la pestaña de “Configuración de proyecto”
3. Dentro de la ventana de Configuración del proyecto, vaya a Plataformas > Android.
4. En Empaquetado APK, habrá una advertencia que dice “El proyecto no está configurado para la plataforma Android”. Haga clic en el botón “Configurar ahora” para configurar automáticamente su proyecto para escribir los archivos necesarios de la plataforma.
5. Rellene el Nombre del paquete Android con un nombre apropiado de empresa y proyecto.
6. Si el botón “Aceptar licencia SDK está activado”, haz clic en él para aceptar el acuerdo de licencia SDK de Android. Si has aceptado previamente este acuerdo, no necesitarás completar este paso.

D. Configuración del Editor y PIE para previsualizaciones móviles

1. En la barra de herramientas, haga clic en Configuración > Vista previa del nivel de renderización y, a continuación, seleccione uno de los niveles de renderización disponibles para Android.
2. Haga clic en el menú desplegable situado junto al botón “Reproducir” de la barra de herramientas. Elija uno de los modos disponibles de “Vista previa móvil” correspondiente al nivel de renderización que haya seleccionado.
3. El editor mostrará ahora el proyecto de forma visualmente coherente con el renderizador de destino. Además, cuando pulses el botón PIE, el proyecto se

iniciará en una ventana independiente configurada con una relación de aspecto estándar para móviles y configurada para emular una pantalla táctil utilizando el ratón. Estos ajustes no afectarán al empaquetado para dispositivos móviles, pero garantizarán que tengas una vista previa precisa cuando trabajes en el editor.

Para mayor información sobre el proceso configuración dirigirse a la documentación de Unreal Engine o ingresar a la siguiente url: <https://docs.unrealengine.com/5.3/en-US/setting-up-unreal-engine-projects-for-android-development/>

C Lanzamiento y empaquetado para dispositivos Android

A. Lanzamiento del proyecto

1. Primero se debe asegurar que se encuentra abierto el Nivel que se desea probar.
2. Dirigirse a la Barra de Herramientas Principal y hacer clic en el botón “Plataformas” que muestra más opciones.
3. En el menú Plataformas, en la sección Inicio rápido, seleccione el dispositivo Android de la lista haciendo clic sobre él.
4. Mientras el Nivel se está lanzando en el dispositivo, el progreso se mostrará en la esquina inferior derecha de la pantalla.
5. Una vez finalizado el despliegue, el proyecto debería empezar a ejecutarse automáticamente en el dispositivo Android. Si el proyecto no se inicia automáticamente, puede iniciarlo buscando la aplicación en su dispositivo y pulsando sobre ella para iniciarla.

B. Empaquetado del proyecto

1. Dirigirse a la barra de herramientas principal y haga clic en Plataformas > Android.
2. Active la opción Android (Multi:ASTC,DXT,ETC2) y haga clic en “Package Project”.
3. Cuando aparezca el diálogo “Package Project”, elija el directorio donde desea guardarlo.

4. De click en “Select Folder” y Unreal Editor comenzará a empaquetar su proyecto.
5. Si navega a la carpeta donde se ha guardado su build, la carpeta contendrá los archivos APK y OBB necesarios para instalar su proyecto en un dispositivo Android. También hay un par de archivos .bat que puede utilizar para instalar o desinstalar automáticamente su compilación en un dispositivo conectado.

Para mayor información sobre el proceso configuración dirigirse a la documentación de Unreal Engine o ingresar a la siguiente url: <https://docs.unrealengine.com/5.3/en-US/setting-up-unreal-engine-projects-for-android-development/>, desde la sección 5 del sitio web.

D Costo de implementación del prototipo

Tabla 1: Costo de implementación del proyecto

	Valor (\$)	Proyección costo: 4 meses	Total (\$)
Configuración y mantenimiento del sistema	875	3500	3500
Orange Pi	137		137
de Visualización	46		46
Máquina virtual en Azure	7,59	30,37	30,37
Parlantes	15		15
Kit con fuente de alimentación, adaptador de corriente y ventilación	21		21
Total de costos			3749