

Escuela Superior Politécnica del Litoral

Facultad de Ingeniería en Electricidad y Computación

Creación de imágenes PixelArt para videojuegos usando modelos generativos de
Deep Learning

TECH-324

Proyecto Integrador

Previo la obtención del Título de:

Ingeniero/a en Ciencias de la Computación

Presentado por:

Dennisse Nicolle Aguirre Veliz

Homar Flavio Herrera Nieto

Guayaquil - Ecuador

Año: 2023 - 2024

Dedicatoria

Este proyecto está dedicado con gran afecto al Dr. Miguel Realpe, por sus enseñanzas y dedicación al campo de la Inteligencia Artificial y siempre proveernos la motivación para culminar el proyecto, a la comunidad de Open Source por nunca dejar de generar software libre.

Este proyecto está dedicado con gran afecto a todas aquellas personas que han albergado el anhelo de crear sus propios videojuegos, pero se han enfrentado a la incertidumbre de no saber por dónde empezar o de contar con recursos limitados para llevar a cabo su desarrollo.

Agradecimientos

Quiero expresar mi profundo agradecimiento a mis padres, quienes siempre me apoyaron, a Christian, mi mejor amigo quien supo estar a mi lado, a Rex, mi gato que siempre está a mi lado en las largas noches de estudio, a Carolina, quien me ayudo cuando se empezó a gestar el proyecto, a Alexandra, Jean y Carlos por siempre ser ese apoyo y rescate cuando las cosas iban mal y por último a mis amigos de Hawaii, en especial a Isaac, que me enseñaron la importancia de disfrutar la vida y de las pequeñas cosas.

Quiero expresar mi profundo agradecimiento a Dios y a mis padres, quienes han sido inquebrantables pilares de apoyo a lo largo de todas las etapas de mi vida. Asimismo, deseo extender mi gratitud a los distinguidos profesores de la universidad. Su dedicación y capacidad para guiarnos y orientarnos a lo largo de nuestra carrera profesional han dejado una huella en mi formación académica.

Declaración Expresa

Nosotros, Dennisse Nicolle Aguirre Véliz y Homar Flavio Herrera Nieto, acordamos y reconocemos que:

La titularidad de los derechos patrimoniales de autor (derechos de autor) del proyecto de graduación corresponderá al autor o autores, sin perjuicio de lo cual la ESPOL recibe en este acto una licencia gratuita de plazo indefinido para el uso no comercial y comercial de la obra con facultad de sublicenciar, incluyendo la autorización para su divulgación, así como para la creación y uso de obras derivadas. En el caso de usos comerciales se respetará el porcentaje de participación en beneficios que corresponda a favor del autor o autores. La titularidad total y exclusiva sobre los derechos patrimoniales de patente de invención, modelo de utilidad, diseño industrial, secreto industrial, software o información no divulgada que corresponda o pueda corresponder respecto de cualquier investigación, desarrollo tecnológico o invención realizada por mí/nosotros durante el desarrollo del proyecto de graduación, pertenecerán de forma total, exclusiva e indivisible a la ESPOL, sin perjuicio del porcentaje que me/nos corresponda de los beneficios económicos que la ESPOL reciba por la explotación de mi/nuestra innovación, de ser el caso.

En los casos donde la Oficina de Transferencia de Resultados de Investigación (OTRI) de la ESPOL comunique a los autores que existe una innovación potencialmente patentable sobre los resultados del proyecto de graduación, no se realizará publicación o divulgación alguna, sin la autorización expresa y previa de la ESPOL.

Guayaquil , 16 de Enero del 2024.

Dennisse A. V

Dennisse Nicolle Aguirre Veliz

Homar Flavio Herrera Nieto

Evaluadores

Luis Eduardo Mendoza M., Ph.D.

Profesor de Materia

Miguel Andres Realpe., Ph.D

Tutor de proyecto

Resumen

Este proyecto se orienta a abordar los desafíos presentes en el desarrollo de 'Mi Bosque 3D' y proyectos similares impulsados por ESPOL. Uno de los obstáculos destacados es la insuficiencia de recursos gráficos y la falta de diseñadores especializados en el estilo PixelArt, esencial para estos videojuegos distintivos. La solución propuesta consistió en desarrollar un prototipo funcional de un sistema de generación de imágenes mediante modelos generativos de Deep Learning. La implementación del prototipo incluyó una interfaz intuitiva con StableSwarmUI como interfaz de usuario y un backend construido con ComfyUI. Además, se integró la tecnología LoRa para refinar las imágenes generadas en el estilo PixelArt según las expectativas del usuario. El resultado final fue una interfaz que permitía a los usuarios generar, modificar y guardar imágenes en el distintivo estilo. La LoRa garantizó la mejora continua de las imágenes según las preferencias del usuario. Este prototipo se destacó por ser offline, lo que no solo lo hizo más accesible, sino que también ahorró tiempo y recursos en el desarrollo de personajes y objetos en este estilo. En conclusión, este sistema prototipo ofreció una solución integral y eficaz para superar los desafíos en la creación de contenido visual para videojuegos.

Palabras Clave: LoRa, ComfyUI, StableSwarmUI , IA

Abstract

This project aims to address the challenges in the development of 'Mi Bosque 3D' and similar projects driven by ESPOL. One of the prominent obstacles is the insufficient graphical resources and the lack of designers specialized in the PixelArt style, crucial for these distinctive video games. The proposed solution involved developing a functional prototype of an image generation system using generative models of Deep Learning. The implementation of the prototype included an intuitive interface with StableSwarmUI as the user interface and a backend built with ComfyUI. Additionally, LoRa technology was integrated to refine the generated images in the PixelArt style according to user expectations. The end result was an interface that allowed users to generate, modify, and save images in the distinctive style. LoRa ensured continuous improvement of the images based on user preferences. This prototype stood out for being offline, making it not only more accessible but also saving time and resources in the development of characters and objects in this style. In conclusion, this prototype system offered a comprehensive and effective solution to overcome challenges in creating visual content for video games.

Keywords: LoRa, ComfyUI, StableSwarmUI , IA

ÍNDICE GENERAL

Resumen	I
Abstract	II
ÍNDICE GENERAL.....	III
ABREVIATURAS	V
ÍNDICE DE FIGURAS	VI
ÍNDICE DE TABLAS	VII
CAPÍTULO 1	1
1.1 Introducción.....	1
1.2 Descripción del problema.....	2
1.3 Justificación del problema	2
1.4 Objetivos.....	3
1.4.1 Objetivo General	3
1.4.2 Objetivos Específicos.....	3
1.5 Marco teórico.....	3
1.5.1 PixelArt.....	3
1.5.2 Creación de Imágenes PixelArt.....	3
1.5.3 Generadores de imágenes con inteligencia artificial	4
1.5.4 Modelos generativos utilizados en Deep Learning	4
1.5.5 Stable Diffusion y LoRa en la generación de imágenes.....	5
CAPÍTULO 2	6
2. Metodología	6
2.1 Requerimientos	6
2.1.1 Requerimientos funcionales.	6
2.2 Alcance y limitaciones de la solución	6
2.3 Riesgos y beneficios de la solución.....	7
2.4 Usuarios de la solución.....	7
2.5 Prototipo	8

2.6	Diseño de la solución	9
2.6.1	Historias de Usuario en la Vista de Escenarios.	10
2.6.2	Vista Lógica	10
2.6.3	Vista de Desarrollo.....	11
2.6.4	Vista de Proceso	11
2.6.5	Vista de Despliegue o Física	12
2.7	Plan De Implementación	14
CAPÍTULO 3		15
3.1	Desarrollo	15
3.1.1	Desarrollo de la LoRa.....	15
3.1.2	Desarrollo del sistema	15
3.2	Resultados.....	17
3.2.1	Resultados de la LoRa.....	17
3.2.2	Resultados del sistema	17
3.3	Pruebas.....	21
3.4	Análisis de costos	22
3.4.1	Ordenador que satisface los requisitos mínimos	22
3.4.2	Computador con requisitos recomendados.....	22
3.4.3	Horas de trabajo	23
CAPÍTULO 4		24
4.1	Conclusiones.....	24
4.2	Recomendaciones	24
BIBLIOGRAFÍA.....		25
APÉNDICE A		27
ANEXOS.....		29

ABREVIATURAS

ESPOL Escuela Superior Politécnica del Litoral

IA Inteligencia Artificial

LoRa Low-Rank Adaptation

GANs Generative Adversarial Networks

VAEs Variational Autoencoders

ÍNDICE DE FIGURAS

<i>Figura 1 Prototipo inicial</i>	8
<i>Figura 2 configuración del prototipo inicial</i>	9
<i>Figura 3 Modelo de vistas</i>	9
<i>Figura 4 Diagrama de componentes</i>	11
<i>Figura 5 Diagrama de secuencia</i>	12
<i>Figura 6 Diagrama de despliegue</i>	13
<i>Figura 7 Plan</i>	14
<i>Figura 8 Diseño final del prototipo</i>	16
<i>Figura 9 Comfy</i>	16
<i>Figura 10 LoRa</i>	17
<i>Figura 11 Acceso directo</i>	17
<i>Figura 12 Pantalla principal</i>	18
<i>Figura 13 Generación de imagen a partir de un prompt</i>	18
<i>Figura 14 Generar imagen a partir de un prompt y una imagen subida</i>	19
<i>Figura 15 Historial de imágenes generadas</i>	19
<i>Figura 16 Negative prompt</i>	20
<i>Figura 17 ControlNet</i>	20
<i>Figura 18 Comparación de imágenes entre generadores</i>	21

ÍNDICE DE TABLAS

<i>Tabla 1 historia de usuario</i>	<i>10</i>
<i>Tabla 2 Tabla de comparación del tiempo de ejecución.....</i>	<i>21</i>
<i>Tabla 3 Características del ordenador</i>	<i>22</i>
<i>Tabla 4 Características del computador.....</i>	<i>22</i>

CAPÍTULO 1

1.1 Introducción

Los videojuegos son las formas de entretenimiento digital más conocidas en todo el mundo [1] y esto se debe a que su industria está en constante expansión. Dentro de este fascinante panorama, es posible explorar una amplia variedad de estilos artísticos que se utilizan en ellos. Uno de los más destacados es el famoso 'PixelArt', que reproduce imágenes mediante píxeles [2]. Este estilo artístico emergió durante los primeros períodos de la historia de los videojuegos, caracterizados por restricciones significativas en el hardware [3]. En ese contexto, las limitaciones se reflejaban en resoluciones reducidas y una paleta de colores sumamente limitada.

A pesar del progreso tecnológico a lo largo de los años, el PixelArt ha persistido como el pionero de los estilos utilizados en los videojuegos actuales, manteniendo su existencia en la industria y siendo empleado en diversos proyectos. Este fenómeno es evidente en la influencia destacada que ha tenido en juegos icónicos como "Super Mario Bros" y "Pac-Man" [4]. Muchas personas han decidido adoptar este estilo debido a su aprecio por lo retro y el ahorro de dinero; sin embargo, la creación de Pixel Art implica el dominio de limitaciones y técnicas específicas [5]. El proceso de creación implica una cuidadosa consideración de cada trazo, cada sombra, para lograr la estética distintiva y evocadora que define al PixelArt.

En la actualidad, se despliega un mundo de posibilidades para la creación de imágenes al estilo PixelArt, impulsado por la Inteligencia Artificial (IA). Aunque los creadores de arte suelen emplear aplicaciones dedicadas a la generación de gráficos píxel, como Pískel [6], la IA emerge como un compañero revolucionario al automatizar el proceso de creación de imágenes. Herramientas como DALL·E 2 ya posibilitan la generación de imágenes a partir de una descripción hecha en lenguaje natural.

Este proyecto tiene como objetivo la creación de un prototipo funcional de un sistema de generación de imágenes al estilo PixelArt de manera offline, utilizando tecnologías de inteligencia artificial como modelos de redes neuronales y de difusión, LoRa (Low-Rank Adaptation), OpenAI y DALL-E. La implementación se complementa con una interfaz gráfica intuitiva que facilite la interacción del usuario y permita hacer configuraciones básicas o avanzadas. Además, contará con un módulo de reentrenamiento sencillo para entrenar a una LoRa específica, junto con un ejemplo de LoRa reentrenada para generar imágenes PixelArt de animales propios de las Islas Galápagos.

1.2 Descripción del problema

Instituciones educativas como ESPOL destacan por liderar el desarrollo de videojuegos con fines educativos. Un ejemplo notorio es "Mi Bosque 3D", un videojuego inspirado en el bosque protector Prosperina [7]. En este entorno interactivo, los jugadores enfrentan desafíos educativos cuidadosamente diseñados para fomentar el aprendizaje, mientras exploran y desbloquean nuevos niveles. Cabe mencionar que este juego adopta el estilo distintivo de 'PixelArt'. No obstante, el proyecto enfrenta una fase prolongada de desarrollo debido a la escasez de recursos gráficos, como laptops o computadoras con tarjetas gráficas, y la falta de diseñadores dedicados a la creación de este estilo.

La falta de diseñadores puede llevar a una variedad de estilos en el videojuego, lo que perjudica la coherencia visual y la inmersión del jugador. La inconsistencia en el diseño gráfico puede causar confusión y afectar la identidad estética del juego, haciendo difícil que los jugadores se conecten emocionalmente con el entorno y los personajes. Por lo tanto, es preferible optar por un único estilo visual que respalde la narrativa del juego, proporcionando así una experiencia más clara y armoniosa para los jugadores. En este sentido, la creación de nuevos personajes mediante la transformación de imágenes preexistentes y objetos al estilo Pixel Art se vuelve esencial para mantener la integridad del diseño y asegurar que cada elemento contribuya de manera cohesionada al estilo general del juego.

1.3 Justificación del problema

La limitada cantidad de personal dedicado a la elaboración de imágenes PixelArt ha provocado retrasos en el desarrollo del videojuego "Bosque 3D". El proceso de creación y búsqueda de *assets*, que son esenciales para la construcción del juego, se ha visto afectado por esta restricción de personal. El diseñador inicia el proceso elaborando un boceto, y posteriormente, transforma ese boceto en píxeles, un proceso que requiere tiempo y habilidad. Hay personal que se dedica a programar el juego, como programadores, pero carecen de la especialización y habilidades necesarias para la creación de imágenes. Este desequilibrio en la distribución de roles dentro del equipo de desarrollo ha llevado a una falta de homogeneidad en el diseño del juego, comprometiendo su identidad estética y retro característica.

Por otro lado, los diseñadores especializados en PixelArt, que forman parte del equipo, aunque cuentan con las habilidades necesarias, se enfrentan a desafíos temporales significativos. La creación de imágenes en este estilo particular lleva su tiempo y, en muchas ocasiones, el proceso implica varias iteraciones para lograr la calidad y estética deseadas. Además, es importante tener en cuenta que estos diseñadores no solo dedican tiempo a la creación, sino

que también operan bajo ciertos costos asociados con su experiencia y esfuerzo. Por ello, surge la necesidad de desarrollar un prototipo de un sistema de generación de imágenes que permita reducir costos y agilizar de manera eficiente el proceso de creación.

1.4 Objetivos

1.4.1 Objetivo General

Desarrollar un prototipo funcional de un sistema de generación de imágenes con el estilo PixelArt, por medio del uso de Inteligencia Artificial, para mantener línea gráfica de un videojuego deseado.

1.4.2 Objetivos Específicos

1. Obtener un proceso exitoso de entrenamiento de modelos generativos de difusión empleando la metodología LoRa, para la producción de imágenes en el estilo PixelArt.
2. Diseñar una interfaz gráfica altamente intuitiva para la generación de nuevas imágenes utilizando modelos generativos de difusión entrenados previamente.
3. Integrar la interfaz gráfica con los modelos generativos de difusión para permitir a los usuarios generar imágenes PixelArt.

1.5 Marco teórico

1.5.1 PixelArt

El PixelArt es un estilo que emergió en la década de 1970, pero alcanzó su máxima popularidad en la década de 1980 [8]. Su impulso se consolidó gracias al éxito de consolas emblemáticas como la Nintendo Entertainment System (NES) y la Master System de Sega [9]. Títulos como Super Mario Bros y The Legend of Zelda destacaron la creatividad y expresión visual única que el PixelArt podía ofrecer [2]. Estos videojuegos no solo se convirtieron en referentes de sus respectivas consolas, sino que también contribuyeron significativamente a la popularización del estilo. Estas consolas adoptaron el estilo PixelArt debido a las notables limitaciones que tenían los dispositivos de esa época [4].

1.5.2 Creación de Imágenes PixelArt

En épocas anteriores, la creación de imágenes se encontraba limitada por diversos factores, destacando especialmente las restricciones en la resolución. No obstante, en la actualidad, se ha producido una auténtica revolución en este campo. Esta transformación significativa se

atribuye a los avances tecnológicos y las innovaciones en herramientas y técnicas. A pesar de estos avances, la creación de imágenes con el estilo distintivo del PixelArt sigue siendo un desafío considerable. Esto se debe a la necesidad de una meticulosa planificación de la ubicación de cada píxel, teniendo en cuenta detalladamente el contorno y la paleta de colores. La complejidad involucrada en la creación de imágenes con estilo PixelArt sigue demandando habilidad y atención minuciosa [10].

1.5.3 Generadores de imágenes con inteligencia artificial

En la intersección entre la creatividad artística y los avances tecnológicos, surgen los generadores de imágenes impulsados por Inteligencia Artificial (IA). La inteligencia artificial se dedica a capacitar a las máquinas para resolver problemas que, por lo general, son resueltos por seres humanos [11]. En la actualidad, herramientas como Midjourney o DALL-E 2 han llevado esta capacidad a un nuevo nivel [12]. Al darles instrucciones precisas y contextualizadas (*'prompt'*) [13], estas herramientas, que emplean modelos de aprendizaje profundo, pueden generar imágenes sorprendentemente realistas y detalladas.

1.5.4 Modelos generativos utilizados en Deep Learning

En el campo del Deep Learning, existen notables herramientas para la generación de imágenes, siendo dos de las más destacadas, las Redes Neuronales Generativas Adversariales (GANs, por sus siglas en inglés) y los Autocodificadores Variacionales (VAEs, por sus siglas en inglés).

Las GANs se componen de dos elementos fundamentales: el "generador" y el "discriminador". El generador tiene la tarea de producir datos sintéticos, mientras que el discriminador se encarga de evaluar si dichos datos son auténticos o falsos. Estos dos modelos operan en conjunto en un proceso iterativo. Con cada iteración, el generador mejora su capacidad para crear datos que sean cada vez más difíciles de distinguir de los datos reales. Al mismo tiempo, el discriminador perfecciona su habilidad para determinar qué tan auténticos son los datos generados [14].

Por otro lado, los VAEs se destacan por su enfoque en el uso de espacios latentes. En estos espacios, los VAEs mapean los datos de entrada, donde cada punto en el espacio representa una variación significativa en los datos. Estos modelos constan de dos componentes: un codificador y un decodificador. El codificador captura la información más relevante de los datos y la comprime en el espacio latente, mientras que el decodificador es responsable de tomar puntos en el espacio latente y generar datos reconstruidos a partir de ellos [15].

1.5.5 Stable Diffusion y LoRa en la generación de imágenes

Stable Diffusion es una herramienta que se emplea para generar imágenes a partir de descripciones. Hace uso de una técnica conocida como "difusión latente". Su código fuente es de acceso público, lo que significa que está disponible para cualquier persona interesada. La función principal de este modelo es seleccionar una imagen basada en una descripción o "prompt" proporcionado por el usuario. A continuación, la imagen generada se somete a un proceso de eliminación de ruido, lo que da como resultado la imagen final. Además, brinda la flexibilidad de ajustar o ingresar varios parámetros de la imagen generada, como sus dimensiones [16].

Por otro lado, LoRa es una técnica de reentrenamiento empleada en el ámbito del Aprendizaje Profundo, especialmente para la optimización de modelos de difusión. Esta técnica tiene la capacidad de ajustar modelos de gran tamaño, incluso cuando los recursos computacionales son limitados y, además, puede adaptar modelos extensos en situaciones donde se cuenta con un conjunto de datos reducido [17].

CAPÍTULO 2

2. Metodología

Después de las reuniones con el cliente, se acordó desarrollar un prototipo funcional para un sistema de generación de imágenes PixelArt, junto con la interfaz gráfica correspondiente. En este proyecto, se decidió implementar Stable Diffusion como motor principal para el interfaz, respaldado por un backend construido con la ayuda de ComfyUI. Es relevante señalar que se diseñó una modalidad offline para permitir el uso del sistema, incluso en situaciones donde no haya conexión a internet.

2.1 Requerimientos

2.1.1 *Requerimientos funcionales.*

- El sistema debe contar con una interfaz gráfica que permita a los usuarios crear imágenes estilo PixelArt mediante un prompt para generar la imagen.
- Los usuarios deben tener la capacidad de editar configuraciones básicas y avanzadas.
- Se debe permitir a los usuarios cargar imágenes junto con una descripción para generar otras imágenes a partir de ellas.
- El sistema debe posibilitar el entrenamiento de una LoRa utilizando un dataset de imágenes.
- Se debe incluir la funcionalidad de reentrenar una LoRa existente.

2.1.2 **Requerimientos no funcionales**

- La interfaz debe ser intuitiva y amigable.
- El sistema debe ser escalable.
- El sistema debe tener un tiempo eficiente al genera las imágenes.

2.2 Alcance y limitaciones de la solución

La solución se beneficiará de una interfaz gráfica desarrollada en Stable Diffusion, respaldada por un backend implementado en ComfyUI. La utilización de modelos generativos, como GANs (Generative Adversarial Networks) o VAEs (Variational Autoencoders), será clave en el proceso de creación de imágenes PixelArt, asegurando resultados visuales de alta calidad. La interfaz ha sido diseñada con la premisa de proporcionar a los usuarios un acceso intuitivo y amigable al sistema, permitiéndoles aprovechar sus capacidades de manera eficiente. La

incorporación de un módulo de reentrenamiento para la LoRa proporcionará a los usuarios la capacidad de ajustar el modelo según sus preferencias.

Por otro lado, es fundamental contar con una computadora que disponga de una tarjeta gráfica, ya que esta juega un papel esencial en el proceso de generación de imágenes. Además, el hardware utilizado debe contar con un espacio de almacenamiento adecuado para manejar la carga de operaciones que se llevarán a cabo durante la creación de imágenes. Es crucial destacar que el entorno donde se utilice el sistema debe ser bien ventilado, ya que el rendimiento de la máquina podría verse afectado por el calor generado durante procesos intensivos.

2.3 Riesgos y beneficios de la solución

Es importante señalar que existe la posibilidad de que la imagen generada no cumpla totalmente con las expectativas del usuario. Además, la efectividad y rendimiento de este prototipo de sistema están estrechamente ligados al hardware en el que se implementa. En este contexto, es crucial señalar que un hardware que no cumpla con los requisitos necesarios podría limitar significativamente la capacidad del sistema para operar a su máxima potencia. Entre los beneficios destacados se encuentra la capacidad de generar imágenes con un estilo distintivo de PixelArt. Además, la posibilidad de transformar una imagen a este estilo proporcionará una solución eficiente, reduciendo significativamente el trabajo manual requerido para diseñar en este complejo estilo visual. A medida que evolucione el sistema, se planea agregar mejoras adicionales, como la compatibilidad con dispositivos móviles, entre otras funcionalidades. Estas mejoras futuras contribuirán a mantener una estética coherente en los diseños, independientemente de la plataforma o dispositivo utilizado, ofreciendo así una experiencia unificada a los usuarios.

2.4 Usuarios de la solución

Entre los usuarios, se incluirán aquellos que interactuarán directamente con el sistema, entre ellos, los programadores y diseñadores. Los diseñadores, con su experiencia y conocimientos, podrán evaluar la calidad estética de las figuras generadas y determinar si cumplen con los estándares deseados. La capacidad de visualizar y ajustar las imágenes a través de la interfaz gráfica intuitiva les proporcionará una herramienta eficaz para la creación de contenido PixelArt, permitiendo una expresión creativa más fluida y ajustada a sus visiones artísticas. Por otro lado, los programadores, al utilizar el sistema, tendrán la oportunidad de personalizar y optimizar la configuración del modelo, adaptándolo a las necesidades específicas del

proyecto en el que estén trabajando. Podrán explorar y modificar parámetros, así como contribuir al desarrollo y mejora continua del sistema mediante su experiencia técnica, lo que garantizará una integración efectiva y eficiente en aplicaciones y entornos diversos. Adicionalmente, entre los usuarios se encuentran los jugadores, quienes podrán disfrutar aún más de los juegos que incorporen este estilo visual PixelArt. La coherencia visual ofrecida por el sistema contribuirá a una experiencia de juego más inmersiva y estéticamente atractiva.

2.5 Prototipo

En la fase inicial de prototipado, se ideó una versión simple y directa del sistema. En esta iteración, se proporcionaba al usuario un espacio para ingresar el prompt, acompañado por opciones claras como cargar, generar y descargar dispuestas debajo. Además, se incorporó un botón de configuraciones en el lado derecho para permitir al usuario ajustar parámetros específicos. Sin embargo, tras una reunión con el cliente, se destacó la necesidad de enriquecer la funcionalidad del sistema, ya que la versión inicial se percibió como demasiado básica. Asimismo, se señaló que la sección de configuraciones resultaba algo confusa, motivando la búsqueda de mejoras y adiciones para satisfacer las expectativas del cliente. Posteriormente, se desea señalar que la Figura 1 ilustra el diseño inicial del sistema.

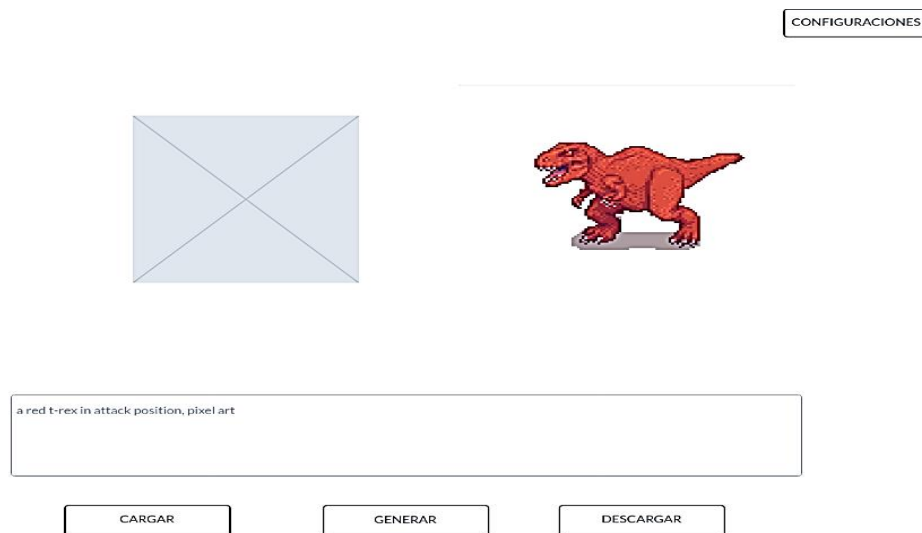


Figura 1 Prototipo inicial

Asimismo, la Figura 2 detalla la sección de configuraciones en la versión inicial del sistema:

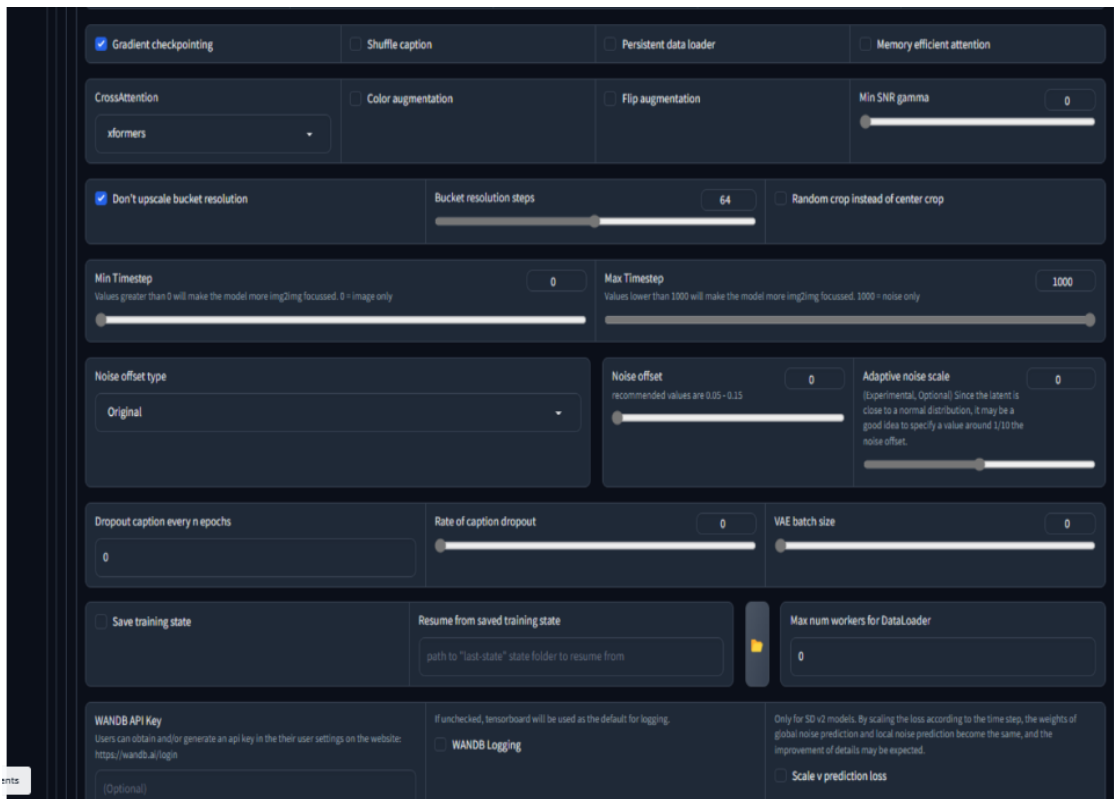


Figura 2 configuración del prototipo inicial

2.6 Diseño de la solución

A lo largo del tiempo, se ha observado que los diagramas de arquitectura a menudo presentan diversas dimensiones de manera poco clara. Como respuesta a este desafío, se ha desarrollado el modelo de "4+1 Vistas" de Kruchten, con el propósito de ofrecer una descripción más detallada y comprensible de la arquitectura del software. Este modelo se centra en cinco vistas concurrentes, que abarcan aspectos como la lógica, los procesos, la disposición física y el desarrollo del software como se muestra en la Figura 3 [18]:

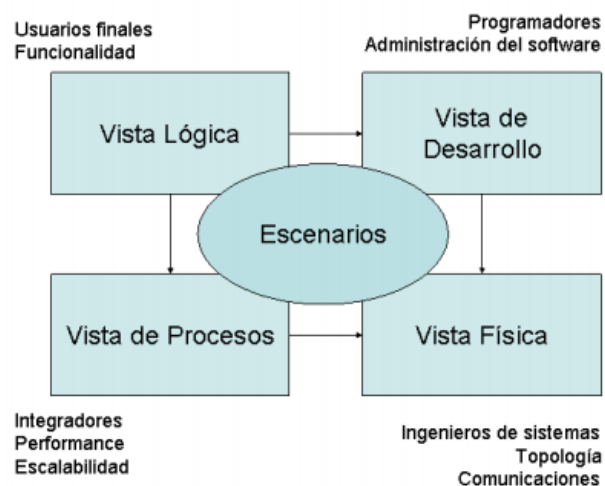


Figura 3 Modelo de vistas

2.6.1 Historias de Usuario en la Vista de Escenarios.

En respuesta a los requisitos establecidos, se ha elaborado un conjunto de historias de usuario, y a continuación, se presentará un desglose detallado de una de ellas, identificada con el ID PIXEL0001 en la Tabla 1. Para obtener una visión más completa y detallada, se invita a revisar el Apéndice A.

Tabla 1 historia de usuario

Enunciado de la historia			Criterios de Aceptación				
Rol	Funcionalidad	Resultado	# de escenario	Criterio de Aceptación	Contexto	Evento	Comportamiento o esperado
Como un cliente	Crear una imagen PixelArt a partir de un prompt ingresado en la interfaz gráfica.	Con la finalidad de crear una imagen en estética PixelArt acorde al prompt.	1	Imagen generada correctamente	En caso de que el prompt sea coherente.	Al dar click en generar.	Se genera la imagen en base al prompt y se almacena en el directorio.
			2	Imagen generada incorrectamente	En caso de que el prompt sea sin coherencia.		Se genera una imagen basada en la interpretación del prompt, la cual podría generar un resultado no deseado.
			3	Imagen no generada	En caso de que el prompt sea cero		El sistema no generará nada.

2.6.2 Vista Lógica

La fase inicial del proceso de generación de imágenes Pixel Art se inicia mediante la recopilación y preparación de un conjunto de datos que consiste en imágenes Pixel Art. Este conjunto de datos es esencial para permitir que el modelo de deep learning aprenda y capture las características distintivas del estilo Pixel Art durante el entrenamiento.

Después de la fase de entrenamiento, el modelo adquiere la capacidad de generar imágenes Pixel Art. Este proceso de generación puede ser iniciado ya sea en respuesta a un prompt introducido por el usuario o a partir de la imagen que esté último suba a través de la interfaz gráfica del prototipo.

2.6.3 Vista de Desarrollo

Para esta vista, se ha elaborado el siguiente diagrama de componentes con el objetivo de proporcionar más información acerca de la arquitectura interna del sistema, tal como se muestra en la Figura 4.

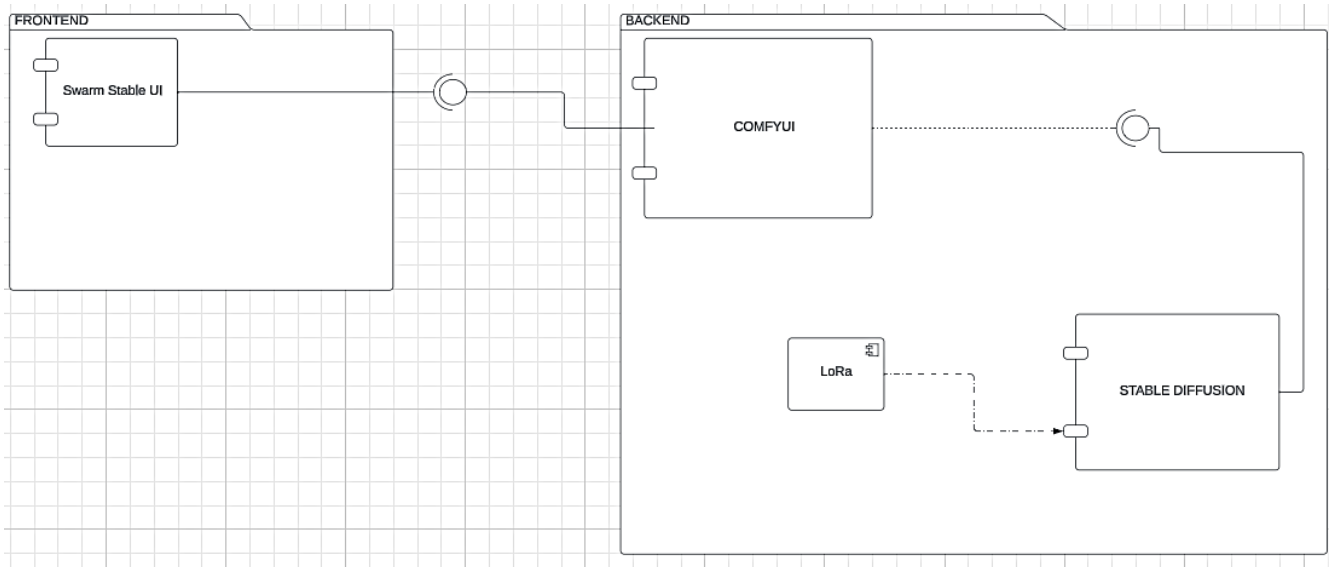


Figura 4 Diagrama de componentes

Lo más destacable de este diagrama, es que se elaboró la LoRa. Su presencia facilita la generación de imágenes con un sesgo específico, permitiendo dirigir la creación hacia un estilo artístico particular, en este caso, el PixelArt.

2.6.4 Vista de Proceso

Para esta vista, se ha elaborado el diagrama de secuencia, que se muestra en la Figura 5, con el propósito de representar las interacciones entre los componentes de este prototipo de sistema.

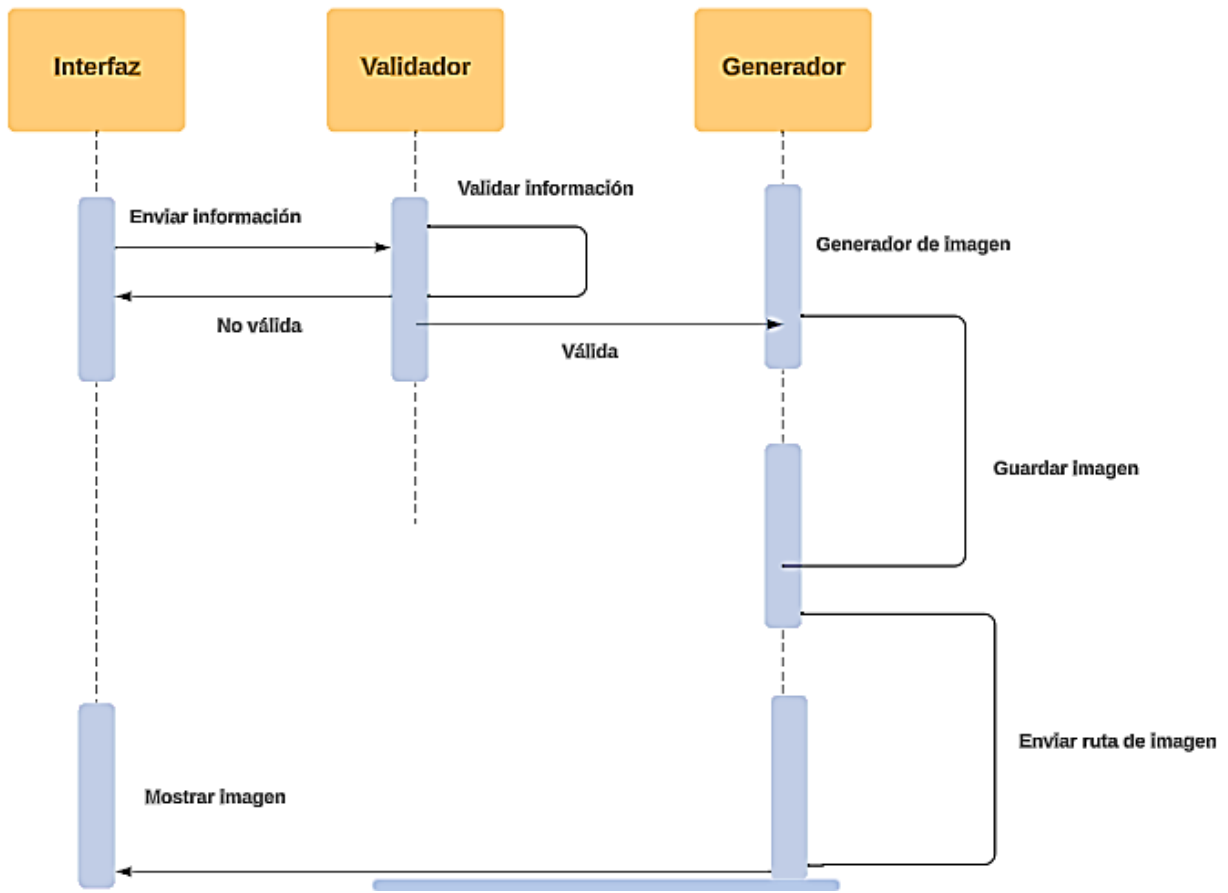


Figura 5 Diagrama de secuencia

Este diagrama proporciona una representación visual que facilita la observación de la interacción entre los tres componentes utilizados durante el proceso de generación de imágenes al estilo PixelArt.

2.6.5 Vista de Despliegue o Física

Para esta vista, se ha elaborado un diagrama de despliegue, que se muestra en la Figura 6, con el fin de mostrar cómo los componentes del software se distribuyen y se ejecutan en el hardware.

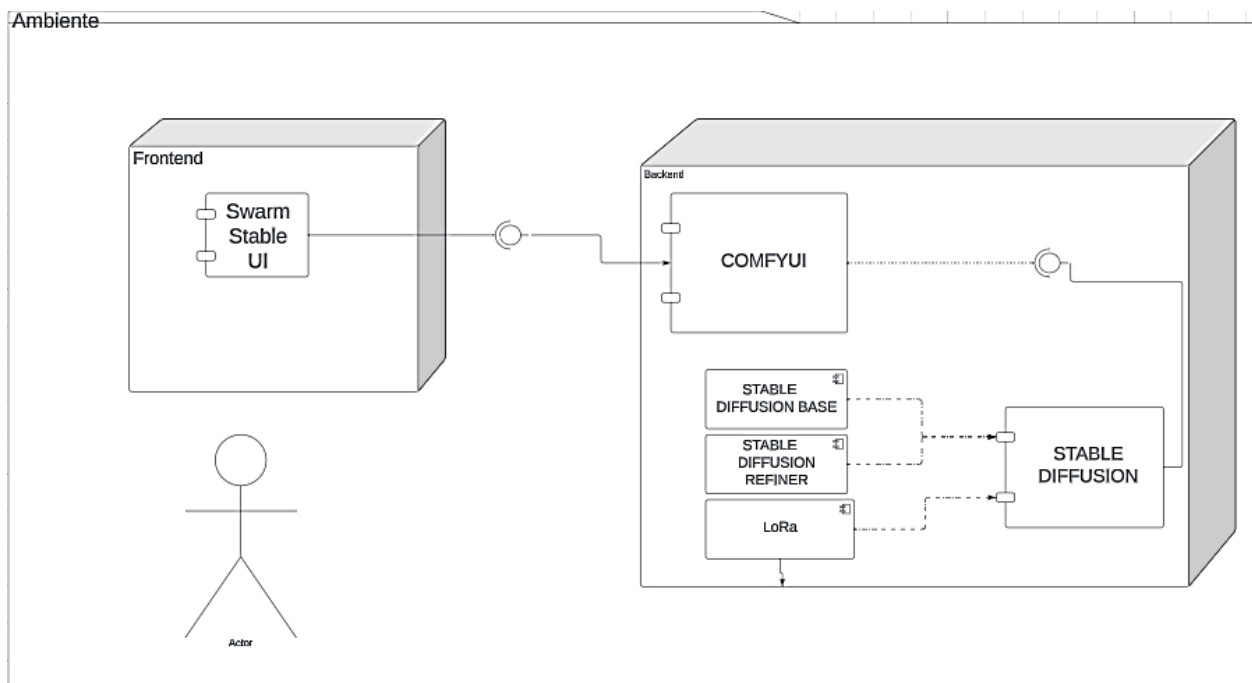


Figura 6 Diagrama de despliegue

Aquí se puede observar que se trabajan en frontend y backend. En el frontend se tiene la interfaz gráfica y en el backend el ComfyUI y el Stable Diffusion. El Stable Diffusion se compone de tres elementos fundamentales: Lora, Base y Refiner.

2.7 Plan De Implementación

El plan de implementación se puede visualizar en la Figura 7.

ID	TÍTULO	INICIO	FIN	(DÍAS)	L	M	X	J	V	L	M	X	J	V	L	M	X	J	V	L	M	X	J
1	Definición, planificación y estimación																						
1,1	Introducción al cliente	2/10/23	2/10/23	1																			
1.2	Introducción al proyecto	3/10/23	3/10/23	1																			
1.3	Levantamiento de requerimientos	3/10/23	5/10/23	3																			
1.4	Levantamiento de ambiente de desarrollo	4/10/23	5/10/23	2																			
1,5	Revisión de lineamientos y otros recursos de plataforma	5/10/23	5/10/23	1																			
2	Diseño de solución																						
2,1	Diseño prototipo de baja fidelidad	9/10/23	10/10/23	2																			
2.2	Revisión de prototipo con cliente	10/10/23	10/10/23	1																			
2,3	Diseño prototipo de alta fidelidad	11/10/23	16/10/23	4																			
2,4	Revisión de diseño con cliente	17/10/23	17/10/23	1																			
2,5	Mapeo de animales Galapagos	17/10/23	24/10/23	6																			
2,6	Investigación de ambiente para interfaz gráfica	17/10/23	19/10/23	3																			
2,7	Definición de parámetros modificables	18/10/23	20/10/23	3																			
2,8	Definición de como se realizará el port a UI	23/10/23	27/10/23	5																			
3	Desarrollo																						
3,1	Desarrollo de pantalla de prompt	30/10/23	6/11/23	6																			
3,2	Desarrollo de pantalla de ajustes	6/11/23	10/11/23	5																			
3,3	Desarrollo de selección de imagen	30/10/23	6/11/23	6																			
3,4	Gestor de descarga y almacenamiento de imágenes	6/11/23	10/11/23	5																			
3,5	Desarrollo de label de puntuación	13/11/23	17/11/23	5																			
3,6	Desarrollo de gestor de puntuación	20/11/23	24/11/23	5																			
3,7	Desarrollo de pantalla de entrenamiento	13/11/23	22/11/23	8																			
3,8	Gestor de entrenamiento	23/11/23	30/11/23	6																			
3,9	Refinamiento de parámetros de entrenamiento	4/12/23	8/12/23	5																			
3,10	Entrenamiento de LoRa para animales de Galapagos	27/11/23	4/12/23	6																			
3,11	Validación de LoRa y reentrenamiento	5/12/23	8/12/23	4																			
4	Revisión y retroalimentación																						
4,1	Pruebas con el cliente	11/12/23	15/12/23	5																			
4,2	Modificaciones	18/12/23	22/12/23	5																			
4,3	Pruebas con el cliente final	26/12/23	29/12/23	4																			
4,4	Desarrollo de manual técnico	2/1/24	5/1/24	4																			
4,5	Entrega final	15/1/24	15/1/24	1																			

Figura 7 Plan

Según lo observado en el plan de implementación, se tiene previsto dedicar 6 días al entrenamiento de la herramienta LoRa con animales, objetos, y otros elementos PixelArt. Durante este período, se llevará a cabo la comparación entre imágenes reales y aquellas generadas por la herramienta LoRa. Este proceso permitirá evaluar la eficacia del modelo en la representación visual de una variedad de elementos.

CAPÍTULO 3

3.1 Desarrollo

3.1.1 Desarrollo de la LoRa

Para el entrenamiento del modelo LoRa se utilizó un dataset de imágenes PixelArt previamente generado con inteligencia artificial y la interfaz gráfica de Kohya, siguiendo los siguientes pasos:

- **Preparación:** Se ha establecido una estructura organizativa con tres carpetas principales: "imágenes", "registró" y "modelo". Dentro de la carpeta "imágenes", se ha creado una subcarpeta destinada a contener las imágenes de entrenamiento, detallando en el nombre el número específico de pasos de entrenamiento que se utilizarán.
- **Etiquetado:** Se empleó la herramienta BLIP Captioning de la interfaz gráfica de usuario (GUI) de Kohya para generar archivos de etiquetas correspondientes a cada imagen. Posteriormente, se llevó a cabo una revisión de estos archivos para garantizar la corrección y precisión del etiquetado.
- **Configuración:** Se empleó el archivo predeterminado para la configuración del entrenamiento, especificando la ubicación de las carpetas previamente creadas. Además, se introdujo el modelo necesario y se procedió con el proceso de entrenamiento.
- **Resultados:** La fase de entrenamiento demandó aproximadamente 35 minutos, dando como resultado la generación de un archivo importante para la implementación del modelo. Este proceso se repitió varias veces con el objetivo de alcanzar el resultado deseado

3.1.2 Desarrollo del sistema

Durante la reunión con el cliente, se presentó inicialmente un diseño de prototipo de baja calidad, lo cual generó comentarios específicos sobre la aparente simplicidad de la interfaz. Tomando en cuenta estas observaciones, se procedió a realizar una nueva versión del diseño, asegurándose de incorporar todos los requisitos solicitados por el cliente.

A continuación se muestra el prototipo final, en la Figura 8.

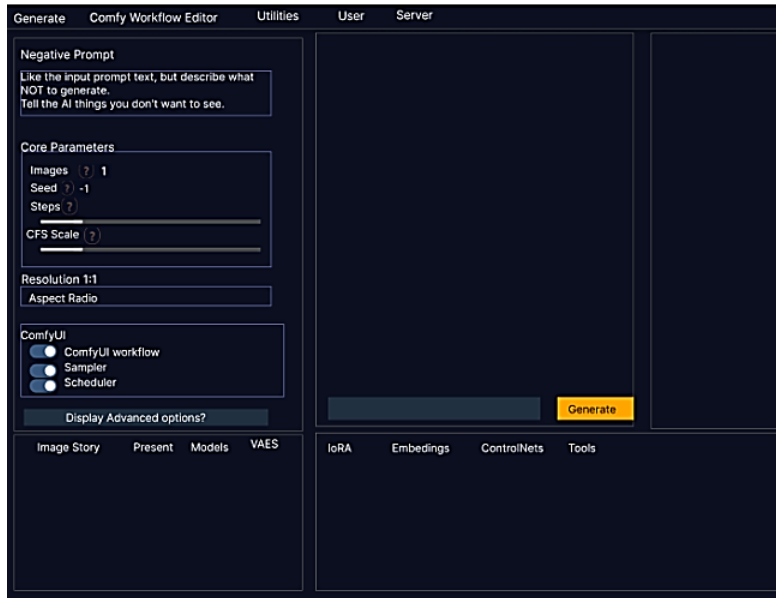


Figura 8 Diseño final del prototipo

Otro aspecto relevante en la interfaz es la presencia de una pestaña denominada "Comfy". Inicialmente, desde esta pestaña se generaban las imágenes, pero se percibió que resultaba algo complejo de manejar debido a la visualización de múltiples nodos que llevaban a cabo diversas operaciones, tal como se aprecia en la Figura 9.

Con el objetivo de optimizar la experiencia de usuario, especialmente para aquellos que no poseen un conocimiento técnico, se tomó la decisión de trasladar esta funcionalidad a una pestaña separada. De este modo, los usuarios especializados pueden ajustar parámetros específicos desde esa sección, mientras que la pantalla principal del prototipo presenta una interfaz más simplificada y accesible para todo tipo de personas, como se ilustra en la figura mostrada anteriormente.

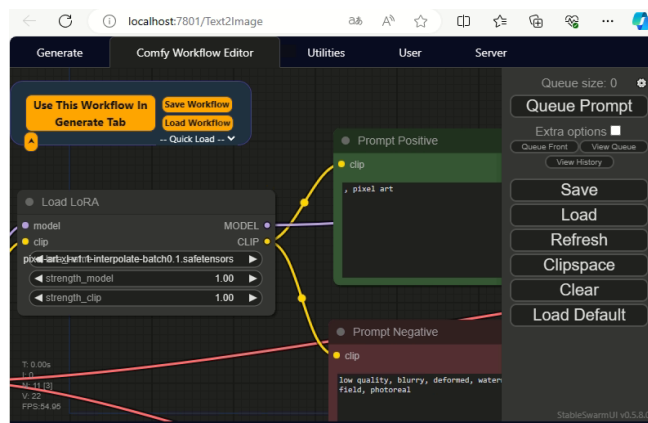


Figura 9 Comfy

3.2 Resultados

3.2.1 Resultados de la LoRa

Se ha logrado crear una LoRa y que esta se pueda usar en la interfaz gráfica como se muestra en la Figura 10.

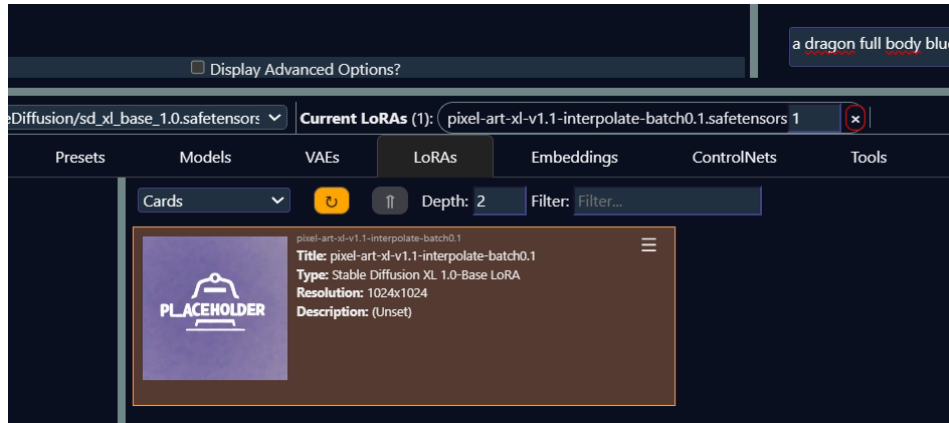


Figura 10 LoRa

Se ha logrado un avance significativo al generar la imagen mediante la introducción de ruido; en este proceso, la herramienta LoRa ha demostrado eficacia al refinar los pequeños detalles, como píxeles, proporcionando una delineación más precisa. Este enfoque ha superado las expectativas del cliente al mejorar la calidad general de los resultados obtenidos.

3.2.2 Resultados del sistema

Se logró crear el prototipo de sistema de generación de imágenes. En primer lugar, se creó un acceso directo como se muestra en la Figura 11 con el fin de proporcionar una vía rápida y eficiente para iniciar el proyecto.

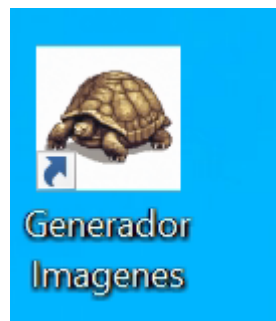


Figura 11 Acceso directo

Con esto, automáticamente se abre un campo de comando (cmd) y se muestra la interfaz gráfica por medio de un navegador como se muestra en la figura 12.

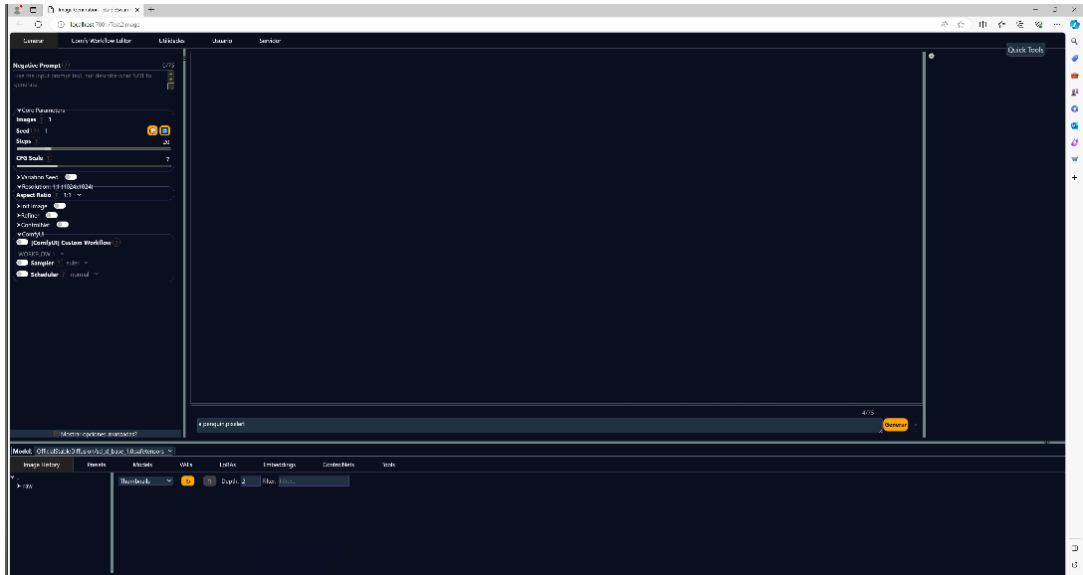


Figura 12 Pantalla principal

Se ha logrado la generación de una imagen de PixelArt a partir de un prompt ingresado por el usuario. En la interfaz, el proceso de generación se visualiza en tiempo real en un cuadro pequeño ubicado en el lado derecho. Para brindar a los usuarios un control adicional sobre el resultado, se han proporcionado parámetros modificables en el lado izquierdo de la interfaz como se observa en la Figura 13.

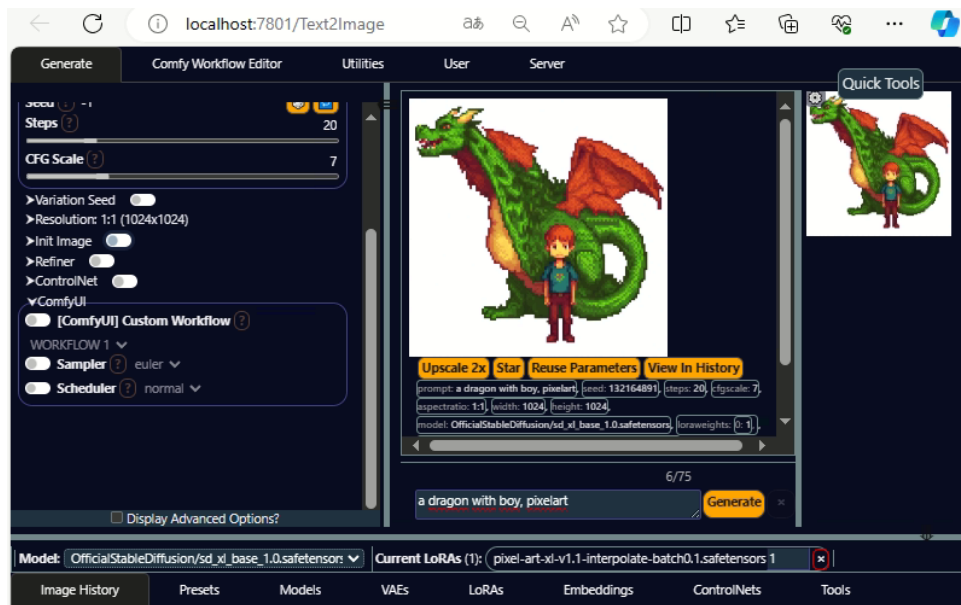


Figura 13 Generación de imagen a partir de un prompt

Asimismo, se ha logrado que el usuario tenga la opción de subir una imagen al hacer el prompt, como se observa en la Figura 14.

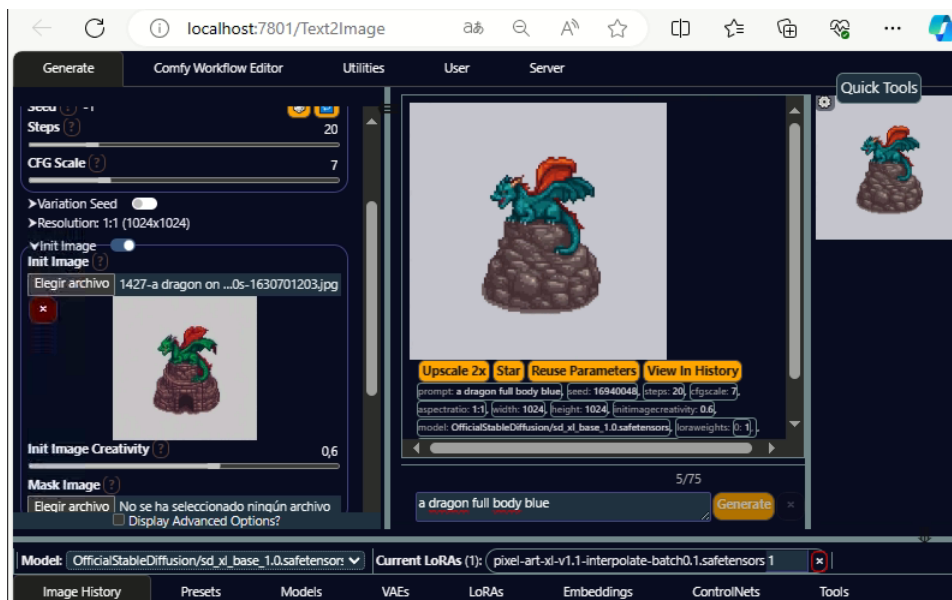


Figura 14 Generar imagen a partir de un prompt y una imagen subida

Además, el sistema guardará las imágenes que el usuario ha generado con la finalidad de que pueda acceder a su historial de creaciones en cualquier momento como se puede apreciar en la Figura 15.

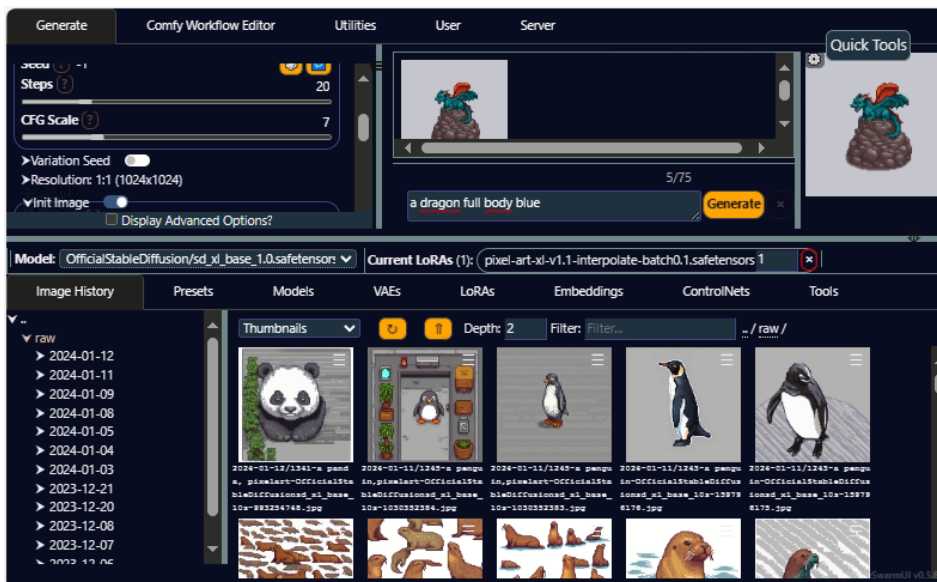


Figura 15 Historial de imágenes generadas

En la sección de parámetros principales, se ha logrado habilitar la funcionalidad que permite a los usuarios incluir "negative prompts". Esto implica que los usuarios puedan especificar qué elementos, situaciones, efectos o estados no desean que aparezcan en la imagen generada como se ve en la Figura 16.

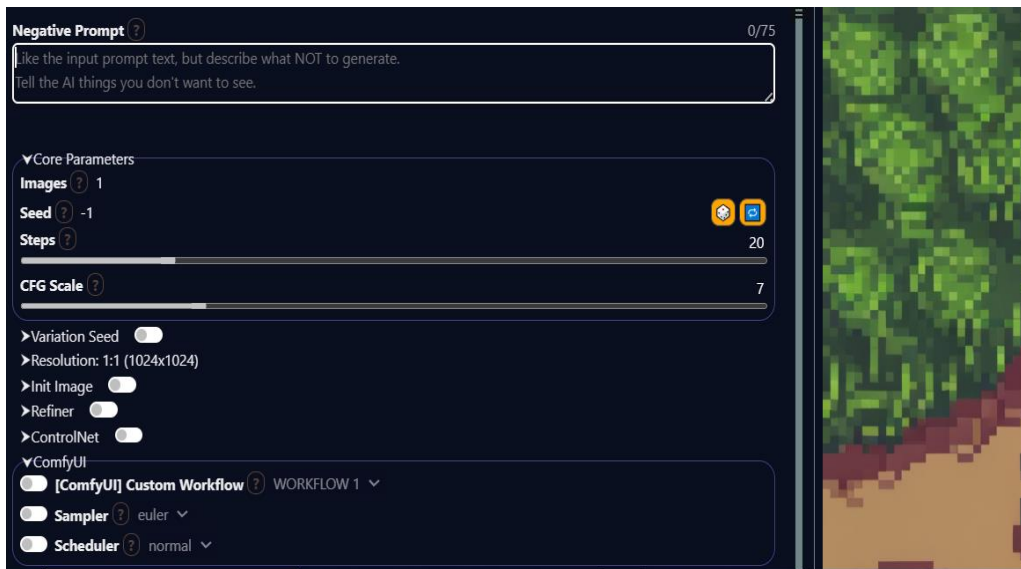


Figura 16 Negative prompt

Por otro lado, se ha incorporado la función ControlNet, que permite al usuario utilizar una plantilla como imagen de referencia para que el prompt se ajuste a su forma específica, tal como se muestra en la Figura 17.



Figura 17 ControlNet

3.3 Pruebas

Para evaluar el prototipo, se compararon las imágenes generadas con el generador de imágenes de Bing para evaluar su calidad visual. Se utilizó una muestra de 20 animales y se aplicaron el mismo prompt y condiciones para la generación en ambos generadores, con el prompt en inglés. No se proporcionó una imagen inicial ni ninguna referencia para guiar el proceso de generación. Las imágenes se generaron en dos escenarios distintos: animales con fondo blanco y animales en un entorno específico. Se proporciona un ejemplo de esto en la Figura 18.

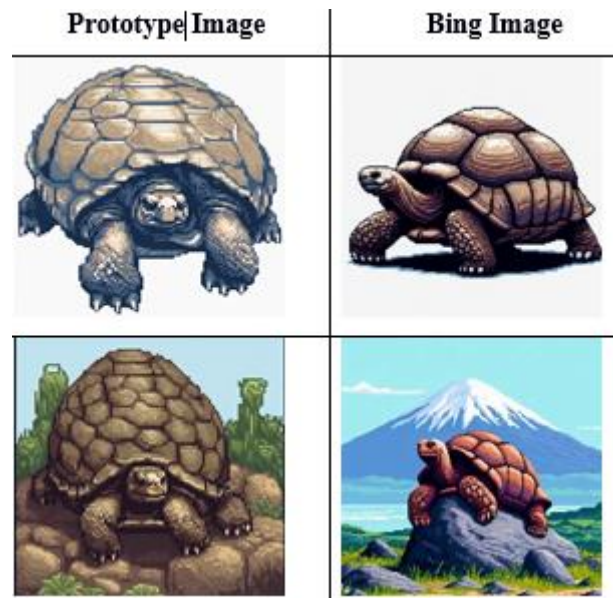


Figura 18 Comparación de imágenes entre generadores

Para obtener un análisis más completo y detallado de los resultados obtenidos, se invita al lector a consultar una tabla comparativa en los anexos. Esta tabla incluye los animales generados junto con sus respectivas situaciones para una evaluación más minuciosa.

Por otro lado, también se llevaron a cabo pruebas para evaluar el tiempo de ejecución en dos dispositivos distintos al generar imágenes con el prototipo, así como el tiempo que un ilustrador tarda en crear imágenes PixelArt. Los resultados de estas pruebas se presentan en la Tabla 2:

Tabla 2 Tabla de comparación del tiempo de ejecución

Tipo	Tiempo de ejecución
Pc 64GB RAM NVIDIA 3090 24GB I9 10TH GEN	10 – 20 sec
Laptop 32GB RAM NVIDIA 3050 TI 2GB I7 10TH GEN	30 - 90 sec
Ilustrador	40 -120 min

3.4 Análisis de costos

3.4.1 Ordenador que satisface los requisitos mínimos

STGAubron Gaming Desktop PC, Intel Core i7 3.4G up to 3.9G, GeForce RTX 2060 Super 8G GDDR6, 32G, 1TB SSD, WiFi, BT 5.0, RGB Keyboard/Mouse, RGB Mouse Pad, RGB BT Sound Bar, RGB Bluetooth Headset Mic, W10H64 : Electronic [19]

En la Tabla 3 se puede observar las características que satisfacen los requisitos mínimos necesarios para poner en marcha el proyecto:

Tabla 3 Características del ordenador

<i>Componente</i>	<i>Especificación</i>
Procesador	3.4 GHz core_i7
RAM	32 GB DDR3
Disco Duro	1 TB SSD
Coprocesador Gráfico	NVIDIA GeForce RTX 2060 Super 8G GDDR6
Marca del Chipset	NVIDIA
Descripción de la Tarjeta	Dedicada
Tamaño de RAM de la Tarjeta	8.00
Tipo Inalámbrico	802.11n, Bluetooth
Número de Puertos USB 2.0	6
Número de Puertos USB 3.0	3

El ordenador que cumple con todas estas características mínimas es valorado en aproximadamente en \$620.

3.4.2 Computador con requisitos recomendados

Corsair Vengeance i7400 Series Gaming PC - Liquid Cooled Intel® Core™ i9 12900K CPU - NVIDIA® GeForce RTX™ RTX 4090 GPU - 2TB M.2 SSD - 64GB Vengeance RGB DDR5 Memory - Black : Electronics [20]

En la Tabla 4 se puede observar las características que satisfacen los requisitos recomendados para el proyecto:

Tabla 4 Características del computador

Component	Specification
Screen Resolution	1920 x 1080
Processor	3.4 GHz core_i9_12900k
RAM	64 GB DDR5
Hard Drive	2 TB SSD
Graphics Coprocessor	NVIDIA GeForce RTX 4090
Chipset Brand	NVIDIA
Card Description	Dedicated

Graphics Card Ram Size	24 GB
Number of USB 3.0 Ports	2

El ordenador que satisface todas estas características recomendadas está valuado en aproximadamente \$3800.

3.4.3 Horas de trabajo

A lo largo de estos meses, cada miembro del grupo ha dedicado un total de 400 horas, abarcando diversas actividades como el levantamiento de requerimientos, investigación, desarrollo de código, adaptación de modelos, entre otras tareas cruciales para el proyecto. El valor estimado por hora de trabajo se establece en \$7 dólares americanos. Entonces el total de los recursos humanos sería \$2800.

CAPÍTULO 4

4.1 Conclusiones

- Se logró desarrollar un prototipo funcional de un sistema de generación de imágenes PixelArt por medio del uso de Inteligencia Artificial usando modelos generativos de Deep Learning con el fin de mantener la misma estética visual en todo el videojuego "Bosque 3D". Gracias a este proyecto, se pueden ahorrar recursos significativos en términos de tiempo y dinero, especialmente en el diseño de objetos y personajes en formato PixelArt. Además, cabe destacar que este prototipo permite generar imágenes de manera gratuita y trabajar sin necesidad de conexión a internet.
- Se logró entrenar exitosamente una LoRa con el objetivo de que la imagen que sea generada sea en estilo PixelArt.
- Se ha desarrollado una interfaz de usuario amigable, con el propósito de facilitar la creación de imágenes para los usuarios. En esta interfaz, se ha integrado de manera efectiva el modelo entrenado, permitiendo a los usuarios utilizarlo con facilidad en el momento que lo deseen.

4.2 Recomendaciones

- Utilizar una computadora con una tarjeta gráfica de alto rendimiento y una capacidad de almacenamiento considerable para que el proceso de generación sea más rápido.
- En futuras investigaciones, se podría explorar la posibilidad de permitir que los usuarios introduzcan un prompt en cualquier idioma y ampliar el entrenamiento de modelos de LoRa para mejorar la precisión en la generación de resultados.

BIBLIOGRAFÍA

- [1] J. Leon, "Surveilling the gamers: Privacy impacts of the video game industry," 2023. [Online]. Available: <https://www.sciencedirect.com/science/article/abs/pii/S187595212200060X>.
- [2] P. Giger, "MODERN PIXEL ART," 2013. [Online]. Available: <https://www.diva-portal.org/smash/get/diva2:832803/FULLTEXT01.pdf>.
- [3] G. Samuelson, "Pixel art - The Medium of Limitation," 2020. [Online]. Available: <https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1518210&dswid=-3572>.
- [4] V. Gonzalez, «Animación Pixel en videojuegos,» 2022. [En línea]. Available: https://repositorio.umayor.cl/xmlui/bitstream/handle/sibum/8880/Animaci%c3%b3n%20Pixel%20Art%20en%20Videojuegos%20y%20su%20rol%20en%20la%20actualidad_Valentina%20Valenzuela_20220912_SAG.pdf?sequence=1&isAllowed=y.
- [5] T. Zufri, "Pixel Art for Game Character Design," 2022. [Online]. Available: <https://journal.binus.ac.id/index.php/jggag/article/view/8565/4553>.
- [6] T. Martínez, «Diseño de personajes para videojuegos,» 2022. [En línea]. Available: <http://dspace.umh.es/handle/11000/28073>.
- [7] ESPOL, «Mi Bosque 3D,» [En línea]. Available: <https://mibosque.espol.edu.ec/teacher>.
- [8] K. Mary, "Graphical Style in Video Games," 2017. [Online]. Available: https://www.theseus.fi/bitstream/handle/10024/133067/Keo_Mary.pdf.
- [9] J. Maravall, «Pixel Art: Estética de la necesidad o elogio del medio,» 2015. [En línea]. Available: <https://revistas.um.es/reapi/article/view/236111>.
- [10] H. Kuang, "A Pixel image generation algorithm based on CycleGAN," 2021. [Online]. Available: <https://ieeexplore.ieee.org/document/9482118>.
- [11] I. Gorojovsky, «Cambio en las competencias valoradas frente a las nuevas tecnologías : Inteligencia Artificial y Machine Learning,» 2019. [En línea]. Available: <https://repositorio.udes.edu.ar/jspui/handle/10908/16763>.
- [12] B. Ali, "Generated Faces in the Wild: Quantitative Comparison of Stable Diffusion, Midjourney and DALL-E 2," 2023. [Online]. Available: <https://arxiv.org/pdf/2210.00586.pdf>.

- [13 D. Torres, «Manual de ChatGPT: Aplicaciones en investigación y educación universitaria,»
] 2023. [En línea]. Available: <https://digibug.ugr.es/handle/10481/85145>.
- [14 D. Manrique, «Redes neuronales aplicadas a la generación de imágenes,» 2023. [En línea].
] Available: <https://oa.upm.es/75103/>.
- [15 L. Calcagni, «Redes Generativas Antagónicas y sus aplicaciones,» 2020. [En línea].
] Available:
http://sedici.unlp.edu.ar/bitstream/handle/10915/101507/Documento_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y.
- [16 N. Dehouche, "What's in a text-to-image prompt? The potential of stable diffusion," 2023.
] [Online]. Available: [https://www.cell.com/heliyon/pdf/S2405-8440\(23\)03964-6.pdf](https://www.cell.com/heliyon/pdf/S2405-8440(23)03964-6.pdf).
- [17 Y. Gu, "Mix-of-Show: Decentralized Low-Rank Adaptation for," 2023. [Online]. Available:
] <https://arxiv.org/pdf/2305.18292.pdf>.
- [18 P. Kruchten, «Planos Arquitectónicos: El Modelo de “4+1” Vistas de la Arquitectura de
] Software,» 2006. [En línea]. Available:
<https://juliopezblog.files.wordpress.com/2021/04/planos-arquitectonicos-el-modelo-de-4-1-vistas-de-la-arquitectura-del-software.pdf>.
- [19 S. Global, «Amazon,» [En línea]. Available: <https://www.amazon.com/-/es/STGAubron-escritorio-alfombrilla-micr%C3%B3fono-auriculares/dp/B0BNBZYR5L>.
- [20 Amazon, «Amazon,» [En línea]. Available: Corsair Vengeance i7400 Series Gaming PC -
] Liquid Cooled Intel® Core™ i9 12900K CPU - NVIDIA® GeForce RTX™ RTX 4090 GPU - 2TB M.2 SSD - 64GB Vengeance RGB DDR5 Memory - Black : Electronics.
- [21 G. El-Din, «PIXEL ART AS A VISUAL STIMULUS IN GRAPHIC ARTS,» 2021. [En
] línea]. Available: https://jaars.journals.ekb.eg/article_177174.html.
- [22 J. d. I. Torre, «AUTOCODIFICADORES VARIACIONALES (VAE),» 2022. [En línea].
] Available: <https://arxiv.org/ftp/arxiv/papers/2302/2302.09363.pdf>.

APÉNDICE A









Historias de usuario completas:

Enunciado de la historia			Criterios de Aceptación				
Rol	Funcionalidad	Resultado	# de escenario	Criterio de Aceptación	Contexto	Evento	Comportamiento esperado
Como un cliente	Crear una imagen PixelArt a partir de un prompt ingresado en la interfaz gráfica.	Con la finalidad de crear una imagen en estética PixelArt acorde al prompt.	1	Imagen generada correctamente	En caso de que el prompt sea coherente.	Al dar click en generar.	Se genera la imagen en base al prompt y se almacena en el directorio.
			2	Imagen generada incorrectamente	En caso de que el prompt sea sin coherencia.		Se genera una imagen basada en la interpretación del prompt, la cual podría generar un resultado no deseado.
			3	Imagen no generada	En caso de que el prompt sea cero		El sistema no generará nada.
Como un cliente	Crear una imagen PixelArt a partir de un prompt y una imagen ingresada en la interfaz gráfica.	Con la finalidad de crear una imagen en estética pixel art acorde al prompt	1	Imagen generada correctamente	En caso de que el prompt sea coherente y la imagen esté en formato PixelArt	Al dar click en generar	Se genera la imagen en base al prompt y se la guarda en el directorio junto a la imagen que subió el usuario.
			2	Imagen generada incorrectamente	En caso de que el prompt sea coherente y la imagen no esté en formato PixelArt		Se toma como referencia la imagen subida y se la transforma en pixel art, tomando en cuenta también el prompt ingresado. Se genera la imagen y se la luego almacena en el directorio.
			3	Imagen no generada	En caso de que el cuadro de texto esté vacío		El sistema no generara nada.
Como un cliente	Entrenar una LoRa desde una interfaz gráfica.	Con la finalidad de mejorar las imágenes generadas.	1	LoRa entrenada desde cero	En caso de entrenar desde cero la LoRa y tener disponible el grupo de imágenes a mejorar	Al dar click en entrenar	Se genera la LoRa a partir del dataset entregado.

2	Lora reentrenada	En caso de reentrenar la LoRa existente y tener un Dataset.	Se genera la LoRa basada en el dataset y LoRa entregada.
---	---------------------	---	--

ANEXOS

Tabla comparativa de imágenes del prototipo vs el generador de imágenes de Bing

Animal	Prompt	Prototype Image	Bing Image
Galapagos Tortoise	Giant a giant Galapagos tortoise with white background, pixelart		
	a giant Galapagos tortoise on top of a rock, pixelart		
Galapagos iguana	marine a Galapagos marine iguana with white background, pixelart		
	a Galapagos marine iguana under the sea, pixelart		

Galapagos Sea Lion

Galapagos sea lion with white background, pixelart



Galapagos sea lion lying down, pixelart



blue footed booby with white background, pixelart

a blue footed booby with white background, pixelart



a Blue-footed booby flying, pixel art



Galapagos Flamingo

a Galapagos Flamingo, white background, pixelart



Galapagos Flamingo
surrounded by plants,
pixelart



Galapagos Albatross

a Galapagos Albatross,
white background,
pixelart



Galapagos Albatross in a
nest, pixelart



a Darwin's finch

a Darwin's finch, white
background, pixelart



a Darwin's finch, in a
grass, pixel art



a Galapagos penguin

a Galapagos penguin,
white background,
pixelart



Galapagos penguin, on
ice, pixelart

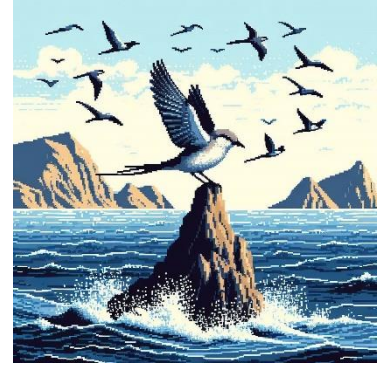


**a Galapagos
mockingbird**

Galapagos mockingbird,
white background,
pixelart



Galapagos mockingbird,
flying over the sea,
pixelart



a Galapagos petrel

Galapagos petrel, white
background, pixelart



Galapagos petrel, lying
down, pixelart



**a Cangrejo Sally
Lightfoot**

Sally Lightfoot Crab,
white background,
pixelart



Sally Lightfoot Crab,
walking in sand, pixelart

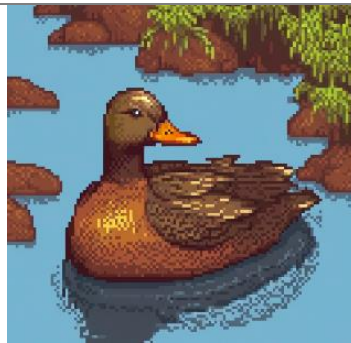


Galapagos duck

Galapagos duck, white background, pixel art

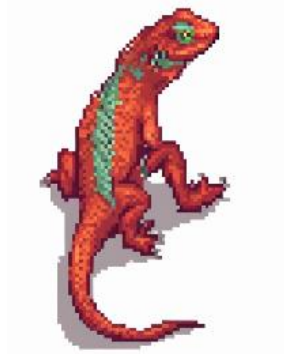


Galapagos duck, swimming in the water, pixel art



Lava lizard from the Galapagos

Lava lizard from the Galapagos, white background, pixelart



a Lava lizard from the Galapagos, on a rock, pixelart



Red lobster

a Red lobster, white background, pixelart



Red Lobster, using its tongs to cut plants, pixel art



green sea turtle

a green turtle, white background, pixelart



a green turtle swimming in the water, pixelart



Red-billed duck

a red-billed duck, white background, pixelart



a red-billed duck opening its wings, pixelart



Galapagos gecko

a Galapagos Gecko, white background, pixelart



a Galapagos Gecko, in the sand, pixelart



Galapagos mouse

a galapagos mouse, white background, pixelart



a galapagos mouse in a bush area, pixelart



Galapagos Crane

a Galapagos crane, white background, pixelart



Galapagos crane, flying over water, pixelart



a blue whale

a blue whale, white background, pixel art



a blue whale swimming ,
pixel art

