Icono

Descripción generada automáticamente con confianza media

**Instituto Tecnológico de Costa Rica**

**Escuela de Ingeniería en Computación**

**Curso:** Inteligencia Artificial

**II Semestre 2023**

**Proyecto Final: Avance**

**Estudiante:**

Celina Madrigal Murillo – 2020059364

**Profesor:**

[Luis Alexander Calvo Valverde](https://tecdigital.tec.ac.cr/dotlrn/community-member?user_id=402439)

**Grupo:** 2

**Fecha de entrega:**

Martes 10 de octubre del 2023

Generador de Historias Ilustradas

# Resumen

El tema por desarrollar es un Generador de Historias Ilustradas que combina el lenguaje natural, la generación de imágenes y la traducción automática. La plataforma permitirá que los usuarios proporcionen una sinopsis breve de una historia con detalles clave y que luego esta se expanda utilizando el modelo ChatGPT. Posteriormente, se generará una imagen alusiva a la historia utilizando Stable Diffusion. Además, se integrará DeepL (se cambió a Google Translate) para traducir las historias a ingles haciéndoles más accesibles. La plataforma se implementará en PythonAnywhere para poder ser ejecutada en la nube.

# Avances a la fecha

1. Se investigó como se obtienen las APIs de ChatGPT. Para obtener una hay que registrarse o iniciar sesión en OpenAI platform, luego ir a View API keys y generar una. Al intentar hacer pruebas se notó que las APIs de ChatGPT tienen un modelo de precios que generalmente se basa en el consumo de tokens. Al crear una cuenta nueva esta incluye 5 dólares de crédito gratuito con vigencia de 3 meses. Para el proyecto se ha creado una cuenta desde cero, de ser necesario más crédito del que ya se incluye gratuitamente más adelante se valorará pagar lo requerido o crear una cuenta nueva.
2. Se investigó como utilizar la API de ChatGPT en Python. Para poder hacer uso de ella primero se tuvo que instalar una biblioteca llamada openai. Luego de esto se creó un código sencillo para probar el funcionamiento de la API. Primero se configura la clave del API, luego se crea una función para interactuar con la API y luego se utiliza la función con su respectivo prompt. El código realizado es el siguiente incluyendo su respectivo resultado:

Texto

Descripción generada automáticamente

Captura de pantalla de computadora

Descripción generada automáticamente

1. Se investigó como obtener la API de Stable Diffusion. Se ha decido utilizar Hugging Face, la cual es una comunidad de ciencia de datos, modelos e inteligencia artificial, que ha desarrollado el modelo de Difusores que permite el uso de modelos basados en difusión como la de Stable Diffusion. Se tomó esta decisión ya que es gratuito, simple y más accesible para lograr el objetivo del proyecto. Para ejecutar el modelo de Stable Diffusion solamente es necesaria la instalación de algunas bibliotecas.
2. Se investigó como utilizar el modelo de Stable Diffusion de Hugging Face. Al igual que con la API de ChatGPT se creó un código para comenzar a hacer pruebas. Primero se importan bibliotecas, luego se define el identificador del modelo pre-entrenado que se utilizará, se ajusta la resolución de la imagen, se define el prompt para generar la imagen, se genera la imagen y por último se muestra. Para ejecutar el código y mostrar la imagen se utilizó un cuaderno de Google Colab. A continuación, se muestra el código creado con su respectivo resultado:

Captura de pantalla de un celular

Descripción generada automáticamente

Una persona con un caballo en la tierra

Descripción generada automáticamente con confianza media

Se utilizó el siguiente video como guía: [Cómo crear imágenes con IA desde Python | TUTORIAL de Stable Diffusion - YouTube](https://www.youtube.com/watch?v=H4VGaC1Gs7o)

1. Se investigó como se obtiene la API de DeepL pero como es de pago se decidió cambiar a la de Google Translate la cual utiliza un modelo de aprendizaje automático para traducir entre más de 100 idiomas. Para utilizar esta API se debe instalar la biblioteca googletrans. A continuación, se muestra el código realizado para hacer pruebas con su respectivo resultado:

Captura de pantalla de un celular

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Se utilizó el siguiente video como guía: [Google Translate API with Python - YouTube](https://www.youtube.com/watch?v=VIc_DDBpE7U)

1. Se unió el código creado para probar la API de ChatGPT y el creado para probar la de Google Translate. Se modificaron los códigos para poder unirlos y a partir de una sinopsis dada por el usuario se genera la historia y esta se traduce a inglés. A continuación, se muestra el código con su respectivo resultado:

Texto

Descripción generada automáticamente

Imagen que contiene Texto

Descripción generada automáticamente

Imagen que contiene Texto

Descripción generada automáticamente

1. Se creó una cuenta gratuita en PythonAnywhere llamada Beginner account la cual tiene ciertas limitaciones. De ser necesario se podría pagar la cuenta Hacker la cual tiene un costo de 5 dólares al mes.
2. Se creó la Web App en PythonAnywhere. Se creó con Flask ya que se considera lo mejor para el propósito del proyecto. El enlace es el siguiente:

<http://celina19.pythonanywhere.com/>

# Cronograma del trabajo faltante

Las tareas pendientes son las siguientes:

* Estudiar a más profundidad cómo funciona y como se utiliza PythonAnywhere.
* Integrar las 3 funcionalidades (generación de la historia, generación de la imagen y la traducción de la historia) para que trabajen de manera conjunta.
* Crear un interfaz sencilla y amigable para la aplicación.
* Realizar el video de la aplicación y el articulo solicitado.

A continuación, se presenta el cronograma de las siguientes semanas para esas actividades:

|  |  |
| --- | --- |
| **Actividad** | **Fechas** |
| Estudio PythonAnywhere | Del 11/10/2023 al 18/10/2023 |
| Integrar las 3 funcionalidades | Del 19/10/2023 al 26/10/2023 |
| Crear la interfaz | Del 27/10/2023 al 03/11/2023 |
| Realizar el video y el artículo | Del 04/11/2023 al 6/11/2023 |