



Granja de Pou's

Proyecto de Programacion Orientado a Objetos

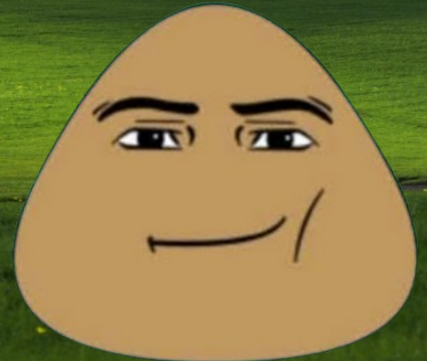
Grupo 18

Miguel Soto Delgado

Sebastián Moya

Maximiliano Muñoz
Hinojosa

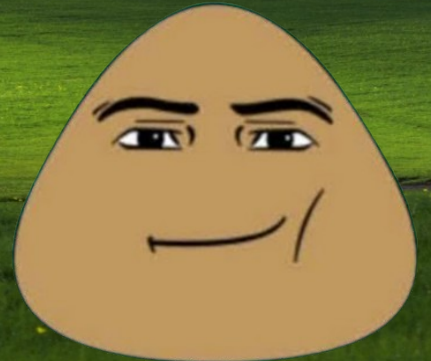
Joaquín Jara
Hananías





Problemática

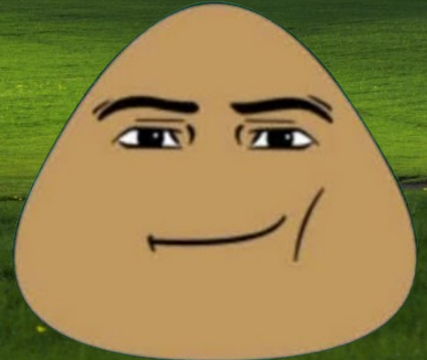
La intención de nuestro proyecto es implementar un Learning Language model con retroalimentación con la finalidad de entrenar una IA de manera local y simular conversaciones entre un número de entidades





Analisis

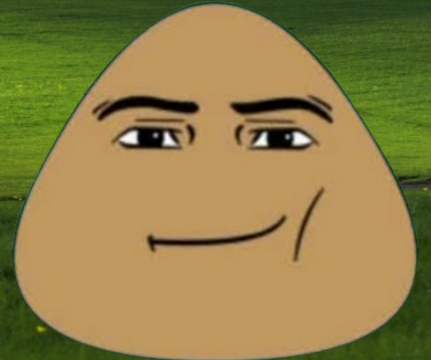
Para efectuar esto, sera necesario crear un LLM de manera local y ejecutar prompts para poder generar los dialogos de los Pou. Sumado a esto, se necesitara una interfaz grafica para poder mostrar en tiempo real las interacciones entre dichas entidades





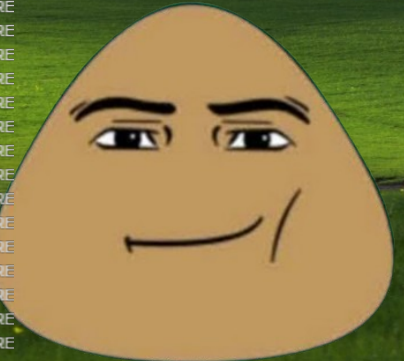
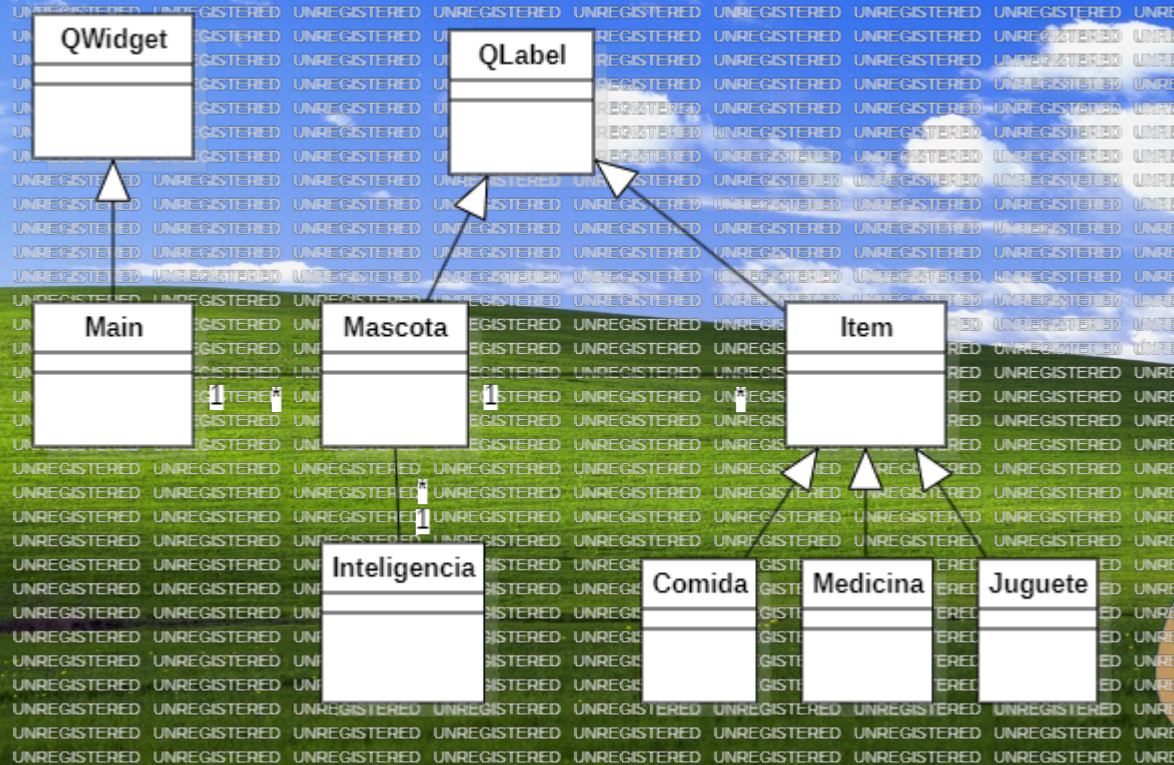
Diseño

En nuestro caso, planteamos un modelo usando la librería Transformers (de Hugging Face) y el modelo GPT2 para el modelo. Admeas de el framework PyQt6 para el interfaz.



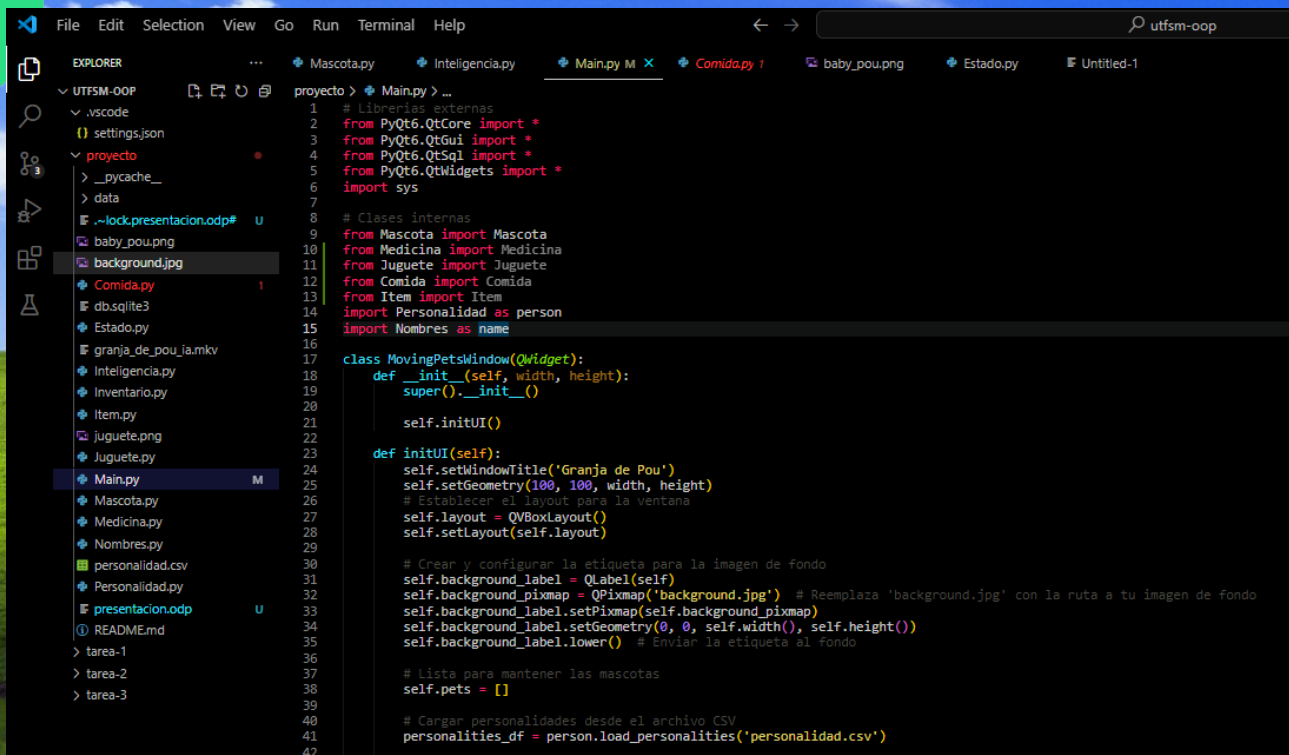


Modelo de Clases





Estructura

A screenshot of a Visual Studio Code editor showing a Qt project structure. The Explorer panel on the left shows a project named 'UTFSM-OOP' with a 'proyecto' folder containing files like 'Comida.py', 'Estado.py', 'Main.py', and 'presentacion.odp'. The main editor shows the code for 'Main.py', which includes imports for PyQt6 and various classes, and a 'MovingPetsWindow' class with an 'initUI' method.

```
1 # Librerías externas
2 from PyQt6.QtCore import *
3 from PyQt6.QtGui import *
4 from PyQt6.QtSql import *
5 from PyQt6.QtWidgets import *
6 import sys
7
8 # Clases internas
9 from Mascota import Mascota
10 from Medicina import Medicina
11 from Juguete import Juguete
12 from Comida import Comida
13 from Item import Item
14 import Personalidad as person
15 import Nombres as name
16
17 class MovingPetsWindow(QWidget):
18     def __init__(self, width, height):
19         super().__init__()
20
21         self.initUI()
22
23     def initUI(self):
24         self.setWindowTitle('Granja de Pou')
25         self.setGeometry(100, 100, width, height)
26         # Establecer el layout para la ventana
27         self.layout = QVBoxLayout()
28         self.setLayout(self.layout)
29
30         # Crear y configurar la etiqueta para la imagen de fondo
31         self.background_label = QLabel(self)
32         self.background_pixmap = QPixmap('background.jpg') # Reemplaza 'background.jpg' con la ruta a tu imagen de fondo
33         self.background_label.setPixmap(self.background_pixmap)
34         self.background_label.setGeometry(0, 0, self.width(), self.height())
35         self.background_label.lower() # Enviar la etiqueta al fondo
36
37         # Lista para mantener las mascotas
38         self.pets = []
39
40         # Cargar personalidades desde el archivo CSV
41         personalities_df = person.load_personalities('personalidad.csv')
42
```





Demostracion

<https://youtu.be/1i-xwWnfGFE>

