

MI JUEGO PYTHON

CREAMOS UN JUEGO PYTHON -{PARTE 1} (PLANIFICACIÓN Y ANÁLISIS)

**requisitos del profesor*

1. El juego tendrá una lista de 15 o 20 palabras (pueden ser más) de 5 letras de longitud. Esta lista estará oculta y no se mostrará al usuario
2. Al comenzar la partida, se tomará una palabra aleatoria de la lista
3. El juego se ejecutará infinitamente hasta que el usuario acierte la palabra de la lista
4. En cada iteración, se preguntará al usuario para que inserte una palabra (tendremos que controlar los errores que puedan producirse: longitud, tipo incorrecto, etcétera).
5. Si la palabra coincide, el usuario gana la partida.

**descripción del juego*

Este juego consiste en adivinar una palabra oculta de una lista predefinida de 15 a 20 palabras, todas de 5 letras. La lista de palabras no es visible para el jugador. En cada turno, el jugador debe introducir una palabra de 5 letras y el sistema le dará pistas sobre si su palabra es correcta o no. El juego sigue ejecutándose hasta que el jugador adivine correctamente la palabra.

**objetivos*

El objetivo es adivinar la palabra oculta a partir de las palabras propuestas por el jugador. Para hacer el juego más desafiante, el sistema puede ofrecer retroalimentación sobre cuántas letras de la palabra ingresada están en la palabra correcta, aunque en este diseño no proporcionaremos ese nivel de detalle; simplemente se indicará si la palabra es correcta o no.

1. **Nombre del juego:** *top secret animals*
2. **Listar las palabras:**
 - araña
 - abeja
 - burro
 - cabra
 - cebra
 - cerdo
 - cisne
 - erizo
 - koala
 - gallo
 - tigre
 - mosca
 - monos
 - morsa
 - perro
 - lince
 - pulpo
 - panda
 - ratón
 - zorro
3. **Selección aleatoria:** Usar el módulo random para seleccionar una palabra de la lista.
4. **Validación de entradas:** Asegurarse de que el jugador ingrese solo palabras de 5 letras.
5. **Control de errores:** Comprobar que el jugador no ingrese caracteres no alfabéticos ni palabras de longitud incorrecta.
6. **Interfaz amigable:** Incluir mensajes claros sobre lo que el jugador debe hacer.
7. **Reinicio del juego:** Una vez que se acierta la palabra, permite reiniciar el juego con una nueva palabra aleatoria.
8. **Portada del juego:**



**diario*

-15/11/2024

Hoy he creado el documento y he planificado su estructura y diseño.

-20/11/2024

Hoy he desarrollado las ideas y los requisitos.

-22/11/2024

Hoy he comenzado a desarrollar mi juego en python.

-27/11/2024 Hoy he arreglado, avanzado y dado forma a mi juego.

-29/11/2024

Hoy termine y depuré my python y explique la información y cada paso de mi python.

-04/12/2024

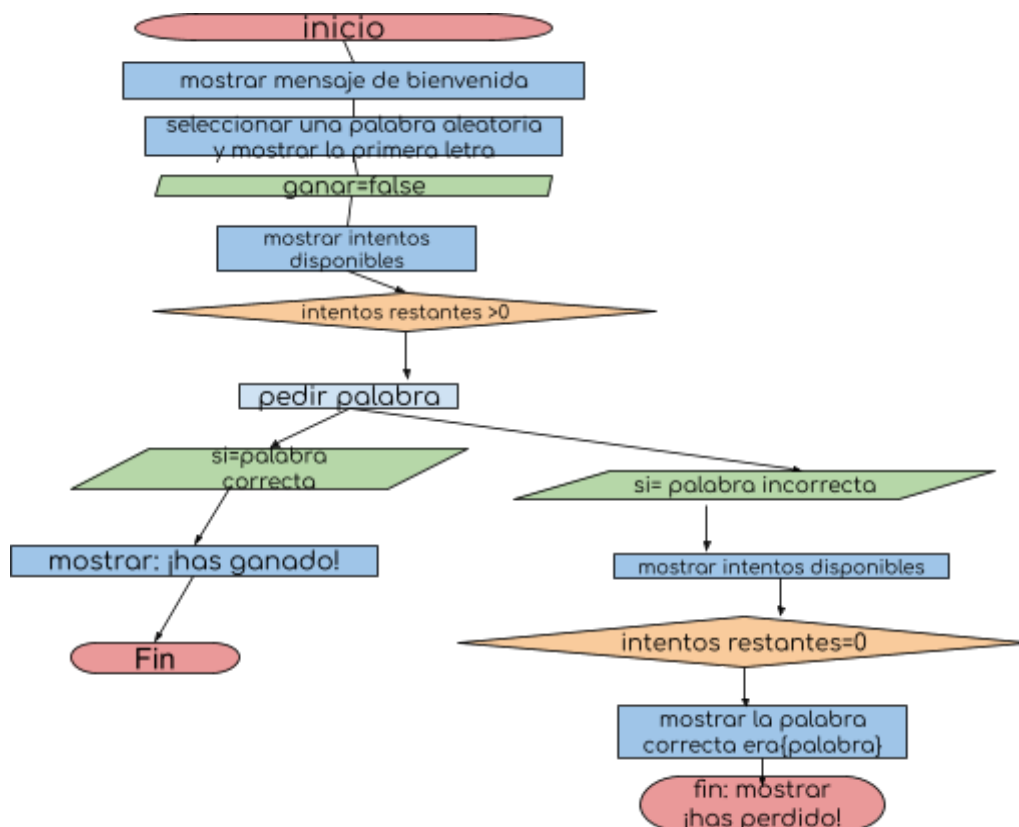
Hoy he perfeccionado y corregido mi juego. Luego he creado la representación de mi algoritmo y terminado los últimos detalles de mi trabajo

CREAMOS UN JUEGO PYTHON -{PARTE 2} (DISEÑO E IMPLEMENTACIÓN)

**requisitos del profesor*

6. Crear un algoritmo en el que indiquemos las fases del juego y todo su funcionamiento.
7. Implementar nuestro juego usando Python.

ALGORITMO



CREAMOS UN JUEGO PYTHON -{PARTE 2} (DEPURACIÓN)

Algunos de los **problemas** que he ido solucionando mientras hacía pruebas de mi juego son:

- Los mensajes que daba eran confusos
- Era difícil entender el código
- Sin pistas el juego era imposible

soluciones

- Poner mensajes claros para que se entienda bien lo que le pido al jugador
- Explicar cada parte del programa del programa para que se entienda lo que quiero hacer.
- Ofrecer una pista para darle juego al jugador

TRABAJO REALIZADO POR MARICARMEN LÓPEZ LÓPEZ