

MANUAL DE USUARIO CRUCERO EN ACECHO CLUE

Versión: 1.0

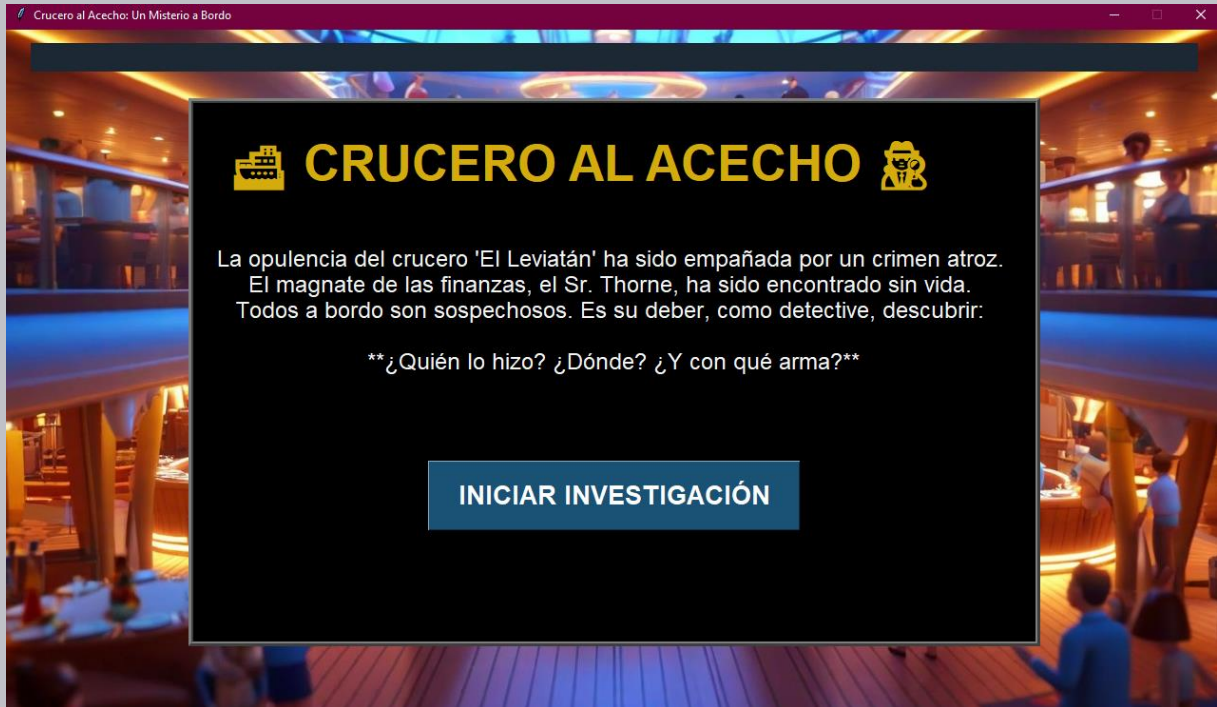
Autor: 7E David Monroy 21110382

Fecha: Octubre 2025



1. Introducción

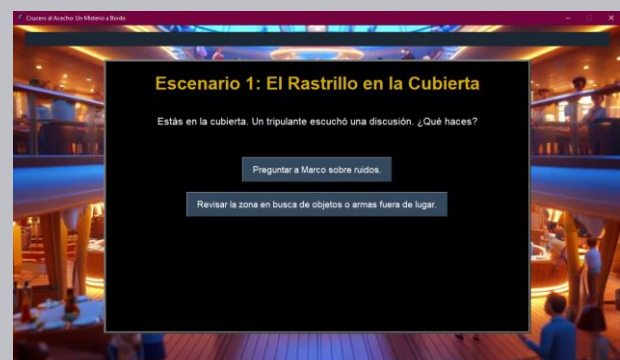
El presente manual tiene como finalidad describir de forma detallada el funcionamiento, lógica interna y uso del videojuego interactivo **“Crucero al Acecho: Un Misterio a Bordo”**, un simulador de investigación criminal desarrollado en Python con interfaz gráfica mediante Tkinter. Este documento ofrece orientación tanto para jugadores como para programadores interesados en entender la mecánica y estructura técnica del sistema.



2. Descripción general del juego

El jugador asume el rol de un detective a bordo del crucero ‘El Leviatán’, donde un crimen ha sacudido la tranquilidad del viaje. A través de una narrativa interactiva, deberá reunir pistas, tomar decisiones y realizar una acusación final identificando al culpable, la locación del crimen y el arma utilizada.

El juego combina elementos de novela visual, lógica deductiva y encadenamiento de pistas, en un entorno gráfico sencillo pero inmersivo. Cada partida genera un caso distinto mediante la selección aleatoria de uno de los cinco finales predefinidos.



3. Requisitos del sistema

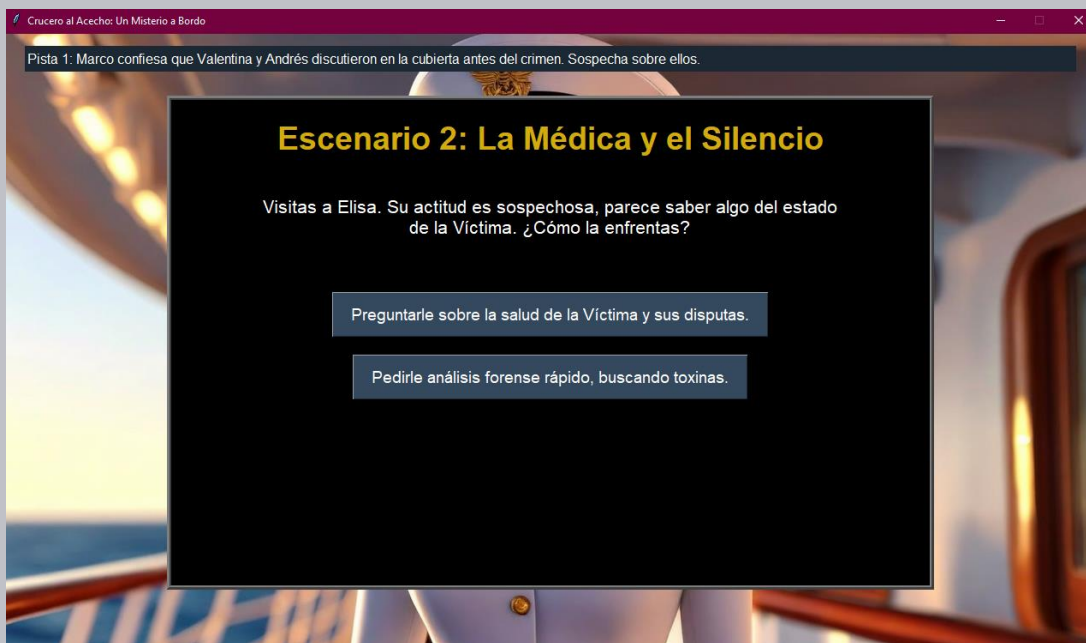
****Software:**** Python 3.10 o superior.

****Librerías necesarias:**** Tkinter (incluido), Pillow (para imágenes).

****Recursos:**** Carpeta 'recursos/' con las imágenes de cada personaje, locación y arma.

****Sistema operativo:**** Windows 10 o superior (compatible también con Linux y macOS).

****Resolución mínima:**** 1280x720 píxeles.



4. Instalación y ejecución

1. Descargue los archivos del proyecto, incluyendo el script principal y la carpeta 'recursos/'.

2. Verifique que las imágenes (.jpg) estén correctamente nombradas y ubicadas.

3. Instale la librería Pillow si no está disponible:

```
pip install Pillow
```

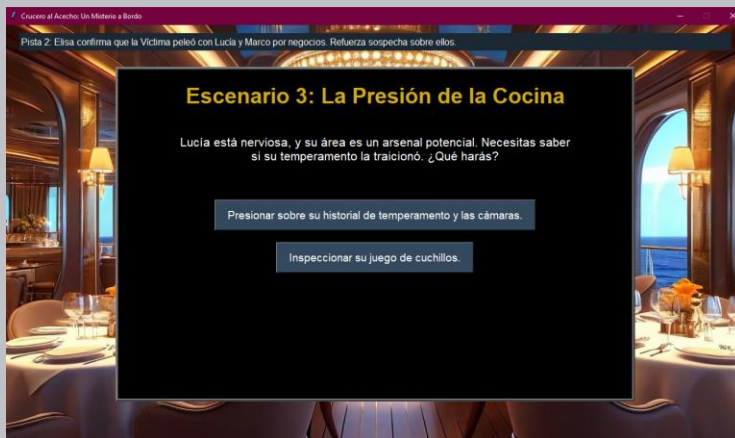
4. Ejecute el programa desde terminal o un entorno de desarrollo con:

```
python crucero_al_acecho.py
```

5. Interfaz del juego

La interfaz gráfica utiliza la biblioteca Tkinter para desplegar imágenes, texto y botones interactivos. Consta de las siguientes áreas principales:

- **Área de Fondo:** muestra la imagen del escenario actual.
- **Marco de Información:** área negra donde aparece la historia y las opciones de decisión.
- **Etiqueta de Pistas:** línea superior donde se registran los descubrimientos recientes.
- **Botones:** permiten avanzar, investigar o realizar la acusación final.

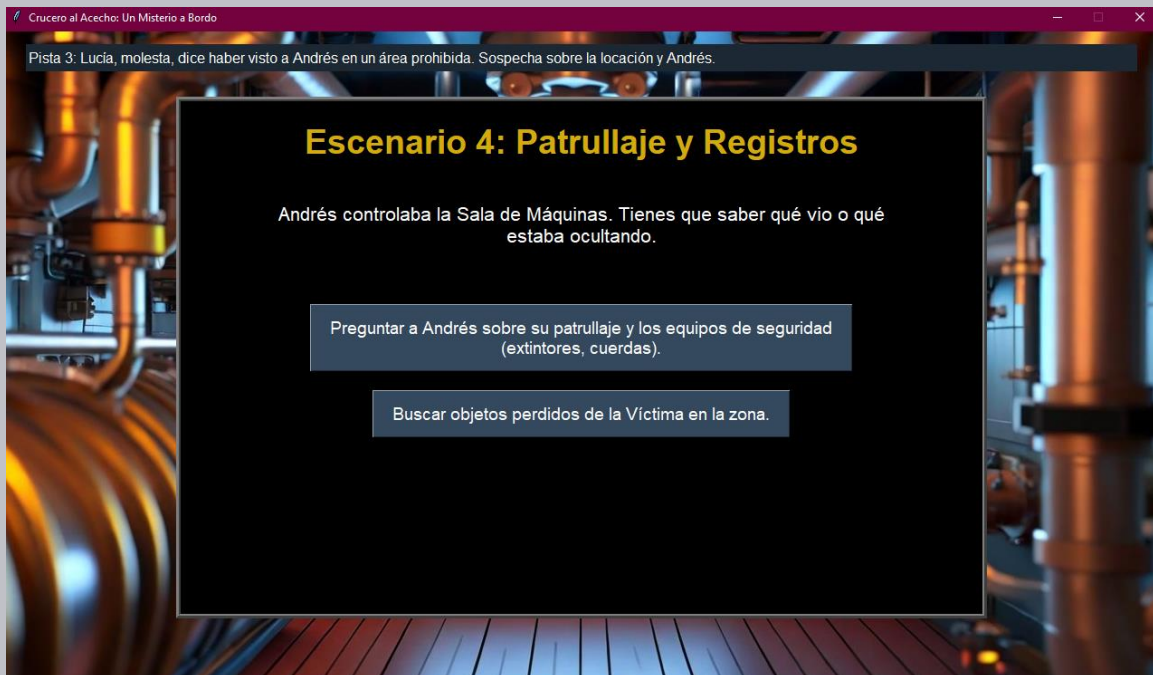


6. Cómo jugar

1. Al iniciar, el juego muestra una introducción con la historia del crimen.
2. Presione "INICIAR INVESTIGACIÓN" para avanzar a los escenarios.
3. En cada escenario, seleccione entre dos decisiones posibles para recolectar pistas.
4. Las pistas obtenidas se muestran en la parte superior del juego.
5. Tras los cinco escenarios, realice su acusación final seleccionando:
 - Culpable
 - Locación
 - Arma
6. Finalmente, el juego revela si su deducción fue correcta.

7. Controles y botones

- ****INICIAR INVESTIGACIÓN:**** Comienza la partida.
- ****Botones de decisión:**** Ejecutan la elección del jugador en cada escenario.
- ****ACUSAR:**** Evalúa la hipótesis del jugador frente a la solución real.
- ****CONTINUAR:**** Avanza al resumen o permite reiniciar el juego.
- ****JUGAR DE NUEVO / SALIR:**** Reinicia el sistema o cierra la ventana.



8. Mecánica interna (vista técnica)

El programa se basa en la clase principal ``CruceroAlAcecho``, heredada de ``tk.Tk``, que define la ventana gráfica y controla el flujo de los escenarios. Se utilizan estructuras de datos globales para manejar los personajes, locaciones, armas y combinaciones predeterminadas de finales. El atributo ``pistas`` funciona como un registro dinámico donde se almacenan los indicios recolectados por categoría.

Cada escenario está codificado en un diccionario ``escenarios_data``, que define título, texto, imagen y opciones. Las decisiones del jugador actualizan las listas de sospechosos, locaciones o armas mediante la función ``manejar_decision()``. El sistema de acusación final compara las selecciones del jugador contra el caso secreto generado al inicio.

8.1 Fundamento de la lógica basada en reglas, casos y encadenamiento hacía adelante

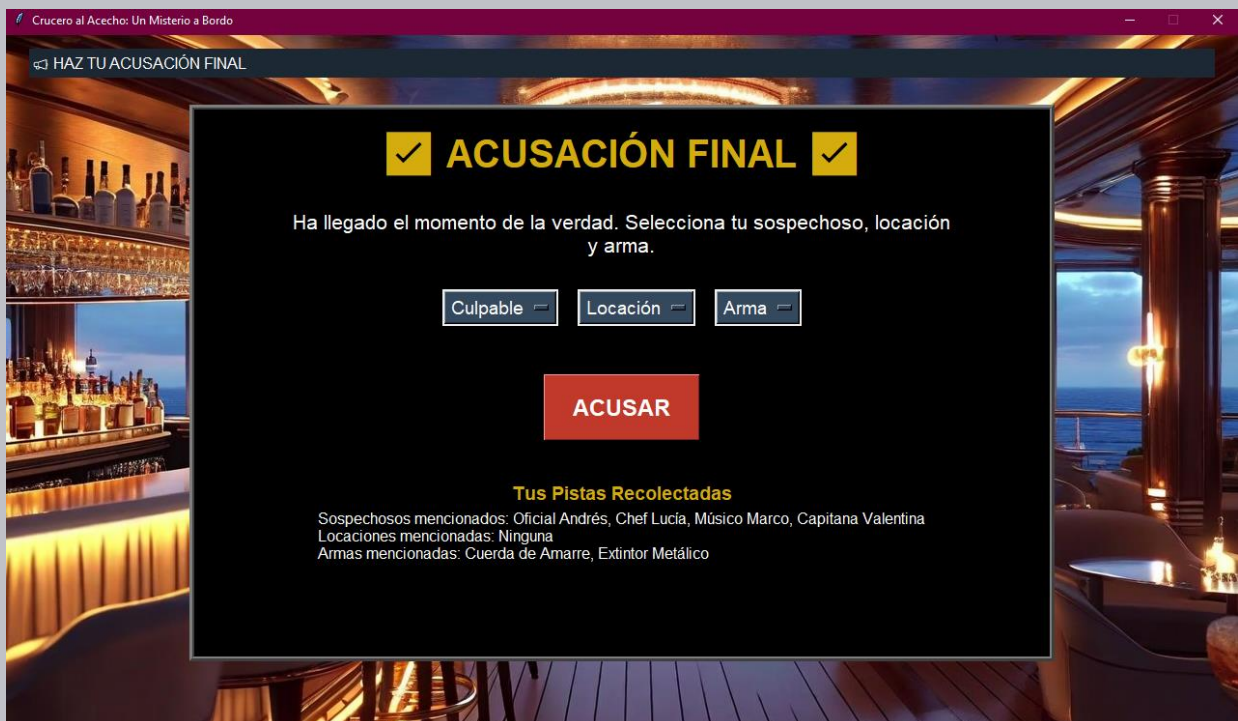
El juego utiliza un modelo de **encadenamiento hacia adelante**, típico en los sistemas expertos. A medida que el jugador selecciona opciones, el sistema analiza condiciones específicas para determinar qué pistas revelar. Cada escenario representa un **caso** dentro del conjunto de conocimiento predefinido.

Por ejemplo, si el jugador interroga a un personaje o inspecciona una locación, la función `manejar_decision()` activa reglas condicionales del tipo:

SI `escenario_actual == 1` Y `sujeto == 'marco'` → agregar `['valentina', 'andres']` a `pistas['culpable']`

SI `escenario_actual == 3` Y `tipo_pista == 'cuchillos'` → agregar `'cuchillo'` a `pistas['arma']`

Este proceso corresponde al razonamiento progresivo del encadenamiento hacia adelante, donde las nuevas evidencias conducen a conclusiones parciales hasta la evaluación final.



9. Consejos y estrategias

- Analice cuidadosamente cada pista antes de avanzar al siguiente escenario.
- Tome nota de los nombres que aparecen con frecuencia: suelen tener relevancia lógica.
- Cruce la información entre arma, locación y sospechoso para inferir el patrón correcto.
- No se apresure en la acusación final: una pista omitida puede cambiar el resultado.



10. Errores comunes y soluciones

****Error:**** El juego muestra 'Imagen no encontrada'.

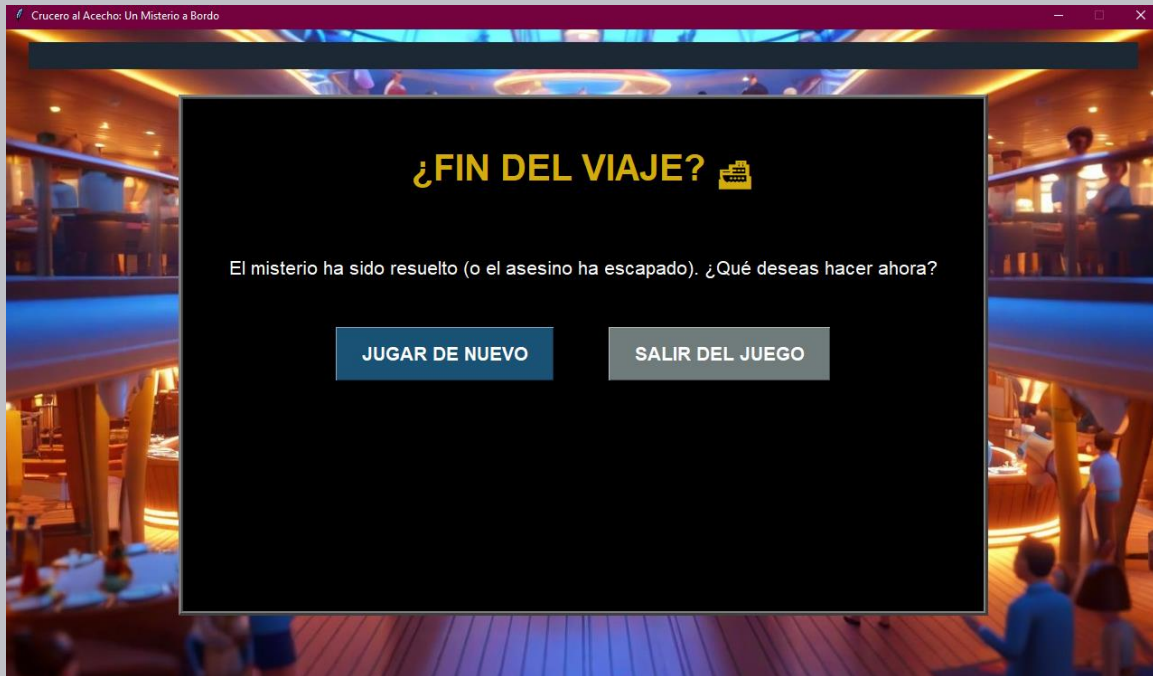
****Solución:**** Verifique que la carpeta 'recursos/' contenga los archivos .jpg con los nombres exactos.

****Error:**** La ventana no se abre o el juego se cierra al iniciar.

****Solución:**** Compruebe la instalación de Pillow y la versión de Python.

****Error:**** El texto se corta o no aparece completo.

****Solución:**** Ejecute el juego en una resolución mínima de 1280x720.



11. Créditos

Desarrollado por: David Monroy 21110382

Lenguaje: Python 3

Bibliotecas: Tkinter, Pillow

Diseño de interfaz y narrativa: David Monroy 21110382