

Manual de Usuario de EBO2 para Juegos Serios: Modos Autoadaptativo y Therapist-In-The-loop

Antonio Blanco, Rosa Rina y Alicia Condón

Robolab

Equipo EuroAGE+

Resumen

Este manual proporciona instrucciones detalladas para utilizar el robot **EBO2** en entornos de terapia con juegos serios. Se describen dos modos de funcionamiento (*autoadaptativo* y *therapist-in-the-loop*), los pasos de configuración inicial (conexión del robot, dirección IP, y configuración de la API de OpenAI) y cada uno de los juegos implementados. Está dirigido tanto a ingenieros como a terapeutas, explicando todo de forma sencilla y paso a paso para facilitar su uso por cualquier persona.

Índice

1. Introducción	2
2. Acceso al Código Fuente	2
3. Conexión del Robot EBO2 y Configuración de la IP	2
3.1. Configuración de la IP	3
4. Modos de Operación de Cada Aplicación	4
4.1. Modo Autoadaptativo: ebo_adaptative_games	4
4.2. Modo Therapist-in-the-loop: ebo_serious_games	5
4.3. Funciones Comunes en Ambos Modos	6
5. Juegos Disponibles en la Plataforma EBO2	7
5.1. Juegos Conversacionales	7
5.1.1. Conversación	7
5.1.2. Storytelling	7
5.2. Juegos Cognitivos	8
5.2.1. Simón Dice	8
5.2.2. Pasapalabra	8
6. Configuración de la API de OpenAI	9

1. Introducción

Este manual proporciona instrucciones detalladas para utilizar el robot **EBO2** en entornos terapéuticos con juegos serios. Se describen dos modos de funcionamiento principales: el *modo autoadaptativo*, que opera de forma autónoma, y el *modo therapist-in-the-loop*, que requiere supervisión activa por parte del terapeuta.

El documento está pensado para ser útil tanto a personal técnico como a terapeutas. Por ello, todas las explicaciones se presentan de forma clara y sencilla, con instrucciones paso a paso para tareas como abrir una terminal o ejecutar scripts.

Los juegos disponibles son: **Simón Dice**, **Pasapalabra**, **Storytelling** y **Conversación**. Estos juegos están diseñados para promover la atención, la memoria, las capacidades cognitivas y la interacción social en personas mayores.

El manual incluye:

- Conexión del robot EBO2 a la red y configuración de IP.
- Configuración de la API de OpenAI para generación de lenguaje natural.
- Instrucciones detalladas para utilizar cada modo de funcionamiento.
- Descripción de los juegos y sus adaptaciones según el modo utilizado.

El objetivo es que cualquier miembro del equipo pueda utilizar el sistema correctamente en el futuro.

2. Acceso al Código Fuente

El código utilizado para ejecutar los juegos serios con el robot EBO2 está disponible públicamente en los siguientes repositorios de GitHub:

- **Modo Autoadaptativo:** https://github.com/AntonioBlanco00/ebo_adaptative_games
- **Modo Therapist-in-the-loop:** https://github.com/AntonioBlanco00/ebo_serious_games

Estos repositorios contienen todos los archivos necesarios para ejecutar las aplicaciones, así como scripts de configuración y documentación técnica adicional. Se recomienda revisarlos antes de comenzar el proceso de instalación o configuración.

3. Conexión del Robot EBO2 y Configuración de la IP

Para que el sistema funcione correctamente, es necesario realizar una serie de conexiones previas. En primer lugar, el ordenador (portátil o de escritorio) que se encargará de lanzar la aplicación de juegos debe estar conectado a una red Wi-Fi. En entornos fuera del laboratorio, como colegios o residencias, esta conexión puede establecerse compartiendo datos móviles desde un teléfono.

El siguiente paso consiste en conectar el robot EBO2 a esa misma red Wi-Fi. Una vez ambos dispositivos estén conectados a la misma red, será necesario indicarle a la aplicación de juegos (ubicada en el PC) cuál es la dirección IP del robot EBO2 para poder establecer la comunicación.

3.1. Configuración de la IP

Para configurar la IP del robot en la aplicación, se debe ejecutar el siguiente comando:

```
python3 actualizar_configs.py
```

Este comando debe ejecutarse dentro de la carpeta correspondiente:

- En el modo autoadaptativo: dentro del directorio ebo_adaptative.
- En el modo therapist-in-the-loop: dentro del directorio ebo2.

Al ejecutar el comando, se abrirá una ventana como la que se muestra en la Figura 1, en la cual se debe introducir la dirección IP del robot EBO2.

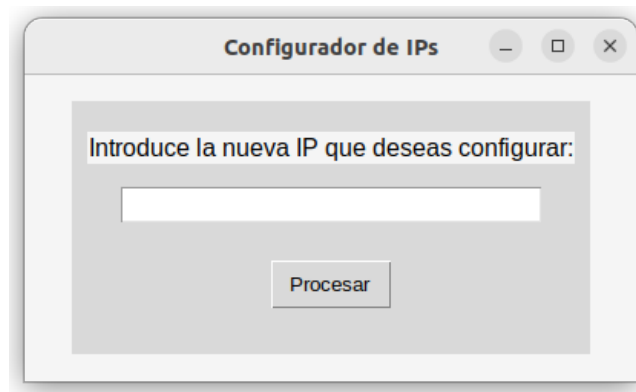


Figura 1: Ventana para introducir la dirección IP del robot.

Para obtener la IP del robot, se debe abrir una terminal en EBO2. Esto puede hacerse con el atajo Ctrl + Shift + T o haciendo clic derecho y seleccionando “Abrir en una terminal”. Una vez abierta, ejecutar el siguiente comando:

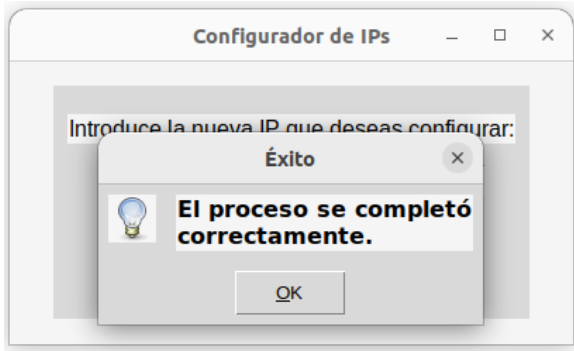
```
ifconfig
```

La dirección IP se encuentra en la sección wlan0, junto a la etiqueta inet. Ver Figura 2.

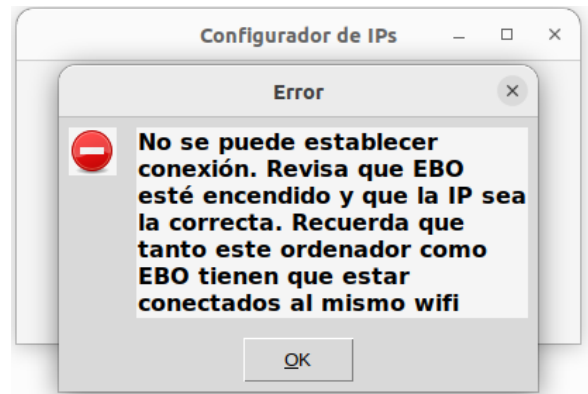
```
wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.50.33 netmask 255.255.255.0 broadcast 192.168.50.255
    inet6 fe80::b822:3947:f61e:51a1 prefixlen 64 scopeid 0x20<link>
    ether e4:5f:01:b1:4d:bb txqueuelen 1000 (Ethernet)
    RX packets 8348 bytes 11773278 (11.7 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2251 bytes 182952 (182.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figura 2: Ubicación de la dirección IP en la salida del comando ifconfig en EBO2.

Una vez introducida la IP en la ventana correspondiente, si aparece un mensaje de "completado", la configuración ha sido exitosa y el sistema está listo para utilizarse. En caso de error, es posible que la IP esté mal escrita o que alguna de las conexiones Wi-Fi (ya sea la del robot o la del PC) esté fallando. Ver Figuras 3a y 3b.



(a) Mensaje de confirmación cuando la IP se configura correctamente.



(b) Ejemplo de mensaje de error por IP incorrecta o conexión fallida.

Figura 3: Mensajes posibles tras la configuración de la IP.

4. Modos de Operación de Cada Aplicación

En este apartado se explica cómo ejecutar cada uno de los modos disponibles en la plataforma EBO2, así como sus características y requisitos particulares. Es importante que, antes de lanzar cualquiera de los modos, se haya configurado correctamente la IP del robot, tal como se describió en secciones anteriores.

4.1. Modo Autoadaptativo: ebo_adaptative_games

Este modo permite que el robot EBO2 se adapte automáticamente a cada usuario. La primera vez que una persona juegue, será necesario crear un nuevo perfil de usuario. En sesiones posteriores, bastará con seleccionar su nombre desde una lista de usuarios existentes.

Este modo utiliza una memoria DSR con agentes, lo cual puede requerir configuraciones adicionales o instalaciones previas en ciertos ordenadores. Además, los juegos se ejecutan mediante yakuake, por lo que este terminal debe estar instalado en el sistema.

Para lanzar este modo, se debe acceder al directorio ebo_adaptative_games y ejecutar el siguiente comando en una terminal:

```
bash autolaunch.sh
```

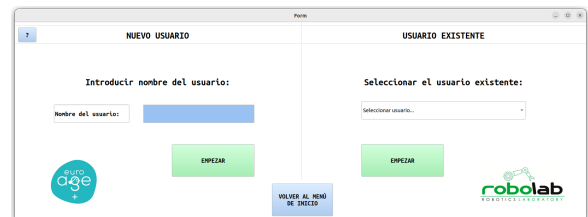
Al ejecutar el script, se desplegará un menú principal desde el cual se pueden lanzar los distintos juegos. Estos se adaptan de manera progresiva en función del desempeño y características del usuario.



(a) Menú principal tras ejecutar `autolaunch.sh`.



(b) Creación de nuevo usuario en juego conversacional.



(c) Carga de usuario en el juego Simón Dice.

Figura 4: Interfaz del modo autoadaptativo en distintas etapas.

4.2. Modo Therapist-in-the-loop: `ebo_serious_games`

En este modo, es el terapeuta quien introduce manualmente los datos del usuario y configura los parámetros de cada juego antes de cada sesión. Esto permite un mayor control sobre la intervención, adaptándola en tiempo real según el criterio clínico.

Este modo no requiere del sistema DSR, y funciona mediante componentes independientes. Por tanto, su instalación y ejecución son más sencillas que las del modo autoadaptativo.

Para ejecutar este modo, se debe acceder al directorio `ebo_serious_games` en el robot y, dentro de la carpeta `EB02`, lanzar el siguiente comando en una terminal:

```
bash iniciar_juegos.sh
```

Además, siguiendo las instrucciones disponibles en el repositorio de GitHub, es posible configurar este modo para que se ejecute mediante un acceso directo de escritorio, facilitando su uso sin necesidad de abrir una terminal.

Figura 5: Menú del juego Simón Dice en modo therapist. La interfaz es similar en todos los juegos, con la diferencia de que los datos del usuario se introducen manualmente.

4.3. Funciones Comunes en Ambos Modos

Ambos modos de funcionamiento incluyen algunas herramientas comunes que resultan útiles tanto para los terapeutas como para el personal técnico:

- **Generación de resultados:** en el menú principal de cada modo existe un botón para generar informes de resultados. Estos informes recogen información sobre el desempeño de los usuarios en los diferentes juegos, facilitando el análisis y seguimiento terapéutico.
- **Control manual del robot:** al iniciar cualquiera de los modos, también se lanza automáticamente un menú de control manual de EBO2. Este menú permite realizar acciones básicas como mover el robot, modificar su moción o que diga algo en concreto. Es especialmente útil en caso de incidencias durante las sesiones. Ver Figura 6.

Figura 6: Interfaz del menú de control manual del robot EBO2, disponible en ambos modos.

5. Juegos Disponibles en la Plataforma EBO2

Los juegos integrados en la plataforma EBO2 están diseñados para estimular distintas capacidades cognitivas, comunicativas y funcionales en personas mayores. Todos los juegos pueden ejecutarse tanto en modo autoadaptativo como en modo therapist-in-the-loop.

En ambos casos, el rol del terapeuta durante la sesión es el mismo: observar, acompañar y, si lo considera necesario, intervenir. La diferencia principal entre modos reside en la configuración previa al inicio del juego, no en su desarrollo.

5.1. Juegos Conversacionales

5.1.1. Conversación

El juego de *Conversación* permite al usuario mantener un diálogo natural con el robot EBO2 sobre temas cotidianos como aficiones, recuerdos personales o la vida diaria. El objetivo principal es fomentar la comunicación verbal, estimular la expresión emocional y reforzar el vínculo entre el usuario y el robot.

Durante la interacción, el terapeuta cumple el rol de transcriptor, escribiendo en una aplicación auxiliar lo que el usuario va diciendo. De este modo, EBO interpreta el contenido y responde de forma coherente, manteniendo un diálogo fluido. Esta dinámica permite una experiencia conversacional natural, incluso cuando el sistema no utiliza reconocimiento automático del habla.

Modos de operación:

- **Modo autoadaptativo:** el terapeuta selecciona el usuario, y el robot configura automáticamente todos los parámetros antes de iniciar el diálogo.
- **Modo therapist-in-the-loop:** el terapeuta introduce manualmente los datos del usuario antes de comenzar cada sesión.

5.1.2. Storytelling

Storytelling es un juego narrativo en el que EBO guía al usuario a través de una historia relacionada con una tarea cotidiana (por ejemplo, preparar un zumo o poner la mesa). En ciertos momentos clave, el usuario debe tomar decisiones simples que afectan el curso de la historia.

El objetivo es estimular la memoria episódica, la comprensión verbal y la toma de decisiones mediante un entorno interactivo y estructurado.

Modos de operación:

- **Modo autoadaptativo:** el sistema selecciona automáticamente una historia no repetida y adecuada al usuario registrado.
- **Modo therapist-in-the-loop:** el terapeuta introduce los datos manualmente, y la historia se selecciona desde el menú.

5.2. Juegos Cognitivos

5.2.1. Simón Dice

Simón Dice es un juego de memoria y atención en el que EBO presenta una secuencia de luces y sonidos que el usuario debe repetir en el mismo orden. La dificultad aumenta progresivamente a medida que se aciertan secuencias más largas.

Este juego está pensado para ejercitar la memoria a corto plazo, la concentración y la coordinación.

Modos de operación:

- **Modo autoadaptativo:** la dificultad del juego se ajusta automáticamente en función del rendimiento del usuario.
- **Modo therapist-in-the-loop:** el terapeuta define manualmente los parámetros antes de comenzar.

5.2.2. Pasapalabra

Pasapalabra es un juego basado en definiciones. EBO enuncia una palabra clave o pista, y el usuario debe adivinar la palabra correspondiente. Si no lo sabe, puede “pasar palabra”, y esa definición se repetirá más adelante.

El papel del terapeuta será marcar en la aplicación si el usuario acertó, falló o pasó la palabra. Este juego trabaja la memoria semántica, el vocabulario y la fluidez verbal.

Modos de operación:

- **Modo autoadaptativo:** el sistema selecciona automáticamente un rosco adecuado al perfil del usuario.
- **Modo therapist-in-the-loop:** el terapeuta puede elegir manualmente el rosco a utilizar o dejarlo en modo automático.

6. Configuración de la API de OpenAI

Los juegos **Conversación** y **Storytelling** utilizan modelos de lenguaje generativo a través de la API de OpenAI. Esta integración permite que el robot genere respuestas naturales, adaptadas al contexto y al perfil del usuario. Para que estos juegos funcionen correctamente, es necesario realizar una configuración previa del acceso a la API.

1. Crear una cuenta en OpenAI

Para comenzar, es necesario registrarse en la plataforma de OpenAI (<https://platform.openai.com>) y acceder al apartado *API Keys*, desde donde se podrá generar una clave de acceso secreta. Esta clave será utilizada por el sistema para autenticar todas las peticiones que realiza a los servidores de OpenAI.

2. Crear el archivo `.env`

Una vez obtenida la clave, debe crearse un archivo llamado `.env` dentro del directorio `ebo_gpt`. Este archivo debe contener la variable de entorno `OPENAI_API_KEY` con el valor correspondiente:

```
OPENAI_API_KEY="sk-XXXXXXXXXXXXXXXXXXXXXXXXXXXX"
```

Este paso es fundamental, ya que la librería de OpenAI utilizada por el sistema leerá esta clave automáticamente desde el entorno para autorizar las conexiones. Puedes encontrar más detalles sobre esta configuración en el repositorio oficial del proyecto Robolab-GPT: <https://github.com/AntonioBlanco00/robolab-gpt>.

3. Crear un asistente en la web de OpenAI

Desde la misma plataforma de OpenAI, en la sección *Assistants*, es posible crear asistentes personalizados para adaptarlos al comportamiento deseado. Cada asistente se configura con:

- Un nombre (por ejemplo, *Storytelling*).
- Un modelo (como `gpt-4`).
- Un *prompt* inicial o instrucción del sistema que define su personalidad y forma de actuar.

Una vez creado, se asignará automáticamente un **ID de asistente**, que será necesario para vincularlo desde la aplicación.

4. Asignar asistentes en `assistants.txt`

Dentro del directorio `src` del proyecto se encuentra el archivo `assistants.txt`. Cada línea de este archivo representa un asistente disponible en la aplicación, con el siguiente formato:

```
NombreVisible;ID_asistente
```

Por ejemplo:

Storytelling;asst_abc123xyz

El nombre visible es el que se mostrará en el menú de selección del asistente dentro de la aplicación, mientras que el ID corresponde al identificador del asistente en la plataforma de OpenAI.

5. Funcionamiento durante el juego

Una vez lanzado el juego (ya sea en modo autoadaptativo o therapist-in-the-loop), el terapeuta selecciona el asistente deseado y el sistema configura automáticamente la conversación con el modelo GPT correspondiente.

En ambos juegos conversacionales, el terapeuta tiene el rol de *transcriptor*: introduce por escrito lo que el usuario va diciendo a través de una aplicación auxiliar. El sistema interpreta ese texto, genera una respuesta mediante la API de OpenAI, y el robot vocaliza la respuesta generada, manteniendo una conversación coherente y adaptativa.

Resumen

- Solo los juegos **Conversación** y **Storytelling** hacen uso de la API de OpenAI.
- Es necesario crear una cuenta, obtener una clave API y definir un archivo `.env`.
- Se deben configurar asistentes en la web de OpenAI y enlazarlos desde `assistants.txt`.
- El terapeuta transcribe las respuestas del usuario, y el robot responde a través del modelo.