

GUÍA DE USUARIO AVENTURA CONVERSACIONAL

[project.py](#)

En este fichero nos encontramos con la inicialización de una mazmorra vacía (línea 29) y dos rutas de 'webhook' (líneas 39-44 y 49-67), una para el 'chatbot' de configuración y otra para el de juego.

La función "save_context" es utilizada para llevar un seguimiento del contexto actual y el contexto anterior usados, siempre y cuando no se produzca un error.

[game_manager.py](#)

En la función "default_welcome_intent_game":

- dngn.reset(): elimina todos los datos posibles guardados de la mazmorra (línea 14)
- dngn.set_mins(): Configura los requisitos mínimos de la mazmorra (línea 15)
- dngn.aux(): genera una mazmorra preconfigurada (línea 16)

Estas tres llamadas deberían ser eliminadas, su única razón de ser es puramente de testeo (evitar tener que configurar una mazmorra cada vez que se necesite probar).

El resto de las operaciones se resumen en:

- 1º: extraemos del *req* ('json' recibido por el 'webhook') los datos importantes, ya sean parámetros propios introducidos por el usuario o información guardada en el 'outputcontext'
- Se opera con la información requerida
- Se devuelve un mensaje personalizado

[conf_manager.py](#)

En la función "default_welcome_intent_conf":

- dngn.reset(): elimina todos los datos posibles guardados de la mazmorra (línea 18)
- dngn.aux(): genera una mazmorra preconfigurada (línea 19)

Estas dos llamadas deberían ser eliminadas, su única razón de ser es puramente de testeo (evitar tener que configurar una mazmorra cada vez que se necesite probar).

En general el resumen del conjunto de funciones es igual que el explicado anteriormente.

[dungeon.py](#)

En este fichero se encuentran las funciones que se encargan de administrar la propia mazmorra. Por lo general son autoexplicativas. La función, previamente mencionada, "dngn.aux()" es utilizada para testear. Se recomienda no eliminar dada su utilidad, pese a que no fuera a ser ejecutada en un posible producto final.

[fallback.py](#)

Este fichero se planteó para ofrecer una respuesta personalizada a cada 'intent' en caso de que el sistema no reconociera la frase introducida por el usuario. Finalmente se optó por devolver una respuesta genérica debido al incremento de 'intents' experimentado. Se recomienda eliminar. No obstante, en caso de querer continuar con la idea original, el esquema ofrecido es un ejemplo de implementación válido.

game.py

En este fichero se encuentra la funcionalidad referente al juego. Las funciones se encargan de:

- `initialite_items()`: esta función se encarga de asignar los 'items' a todas las habitaciones, objetos que contengan otros objetos y personajes que posean objetos. Este proceso es necesario ya que en el 'chatbot' de configuración toda la información se guarda en forma de 'string'.
- `build_rooms_aranged()`: esta función es utilizada para generar un mapa. Para ello, se utiliza la primera habitación como referente y se van añadiendo el resto de las habitaciones en las direcciones libres que existan.
- `build_items()`: esta función asigna los objetos a las habitaciones de manera aleatoria y priorizando aquellas habitaciones que no contengan objetos previamente.
- `build_characters()`: esta función asigna los personajes a las habitaciones de manera aleatoria y priorizando aquellas habitaciones que no contengan personajes previamente.

map_builder.py

Este fichero contiene la funcionalidad referente a la generación de mapas. Utiliza la librería 'pygame' que sirve para dibujar y generar imágenes. Esencialmente, este fichero se encarga de producir una imagen en formato ".png" cuyo tamaño se ajusta al número de habitaciones y su disposición. A su vez, muestra los personajes, objetos o jugadores que se encuentran en cada habitación.

if room.visited or not room.visited (línea 56), este condicional debería ser modificado y eliminar la segunda condición. De esta manera se dibujarán exclusivamente aquellas habitaciones que sí hayan sido visitadas.

En caso de querer generar un mapa que distribuya su tamaño en función exclusivamente de las habitaciones visitadas y no de todas las habitaciones existentes (pese a que luego solo se mostraran en la imagen aquellas visitadas) bastaría con generar una lista de habitaciones auxiliar en el que solo se encuentren aquellas visitadas.

Extra

En términos generales todas las funciones son simples y autoexplicativas. Toda la mazmorra se basa en la administración y configuración de objetos (todas las clases, comentadas y no comentadas en este fichero). La funcionalidad de cada uno de los sistemas conversacionales sigue el mismo esquema sencillo.