

Manual de uso para la ejecución de simulaciones

En este anexo se procede a describir la forma de proceder para poder ejecutar simulaciones sobre un escenario. Se va a explicar la forma de compilar y ejecutar el proyecto, los archivos necesarios para poder ejecutar y los dos tipos de ejecuciones posibles: evaluaciones individuales y evaluaciones desatendidas.

1. Compilación y ejecución del proyecto

Como ya se ha comentado en el listado de tecnologías en la memoria principal (Sección 1.2) este proyecto está desarrollado en el lenguaje de programación *Java*. Por tanto, el primer requisito para poder ejecutarlo es tener instalado el *JDK*, el cual puede descargarse en el sitio oficial de *Oracle*: <https://www.oracle.com/java/technologies/downloads/>. La versión utilizada en este proyecto es: *jdk1.8.0_201*.

Para la compilación y ejecución del proyecto, se utiliza *Apache Ant*. Se ha de tener instalada esta herramienta y se recomienda incluir en la variable de entorno “*PATH*” la ruta a donde se ha instalado para poder ejecutarse desde cualquier ruta del sistema.

Por último, se ha de clonar el repositorio del proyecto de *GitHub* (<https://github.com/silarri/simulCARS>). Dentro del proyecto, se ha de acceder a: “*simulCARS/Simulator2021/MainSimulator/*”. En esta ruta se encuentra el archivo “*build.xml*”, script “*Ant*” creado para compilar y ejecutar el proyecto. Los comandos que se han de ejecutar desde la consola (Windows o Linux) son:

Limpiar y compilar:

```
ant clean  
ant build
```

Y para ejecutar, se invoca a la clase principal:

```
ant MainSimulator
```

El uso de este comando en lugar de “*ant run*” se debe a que se pueden definir otros “targets” para probar pantallas concretas de la GUI (de este proyecto o de futuras implementaciones). Para ello, bastaría con incluir otra etiqueta “target” equivalente a la nombrada *MainSimulator* y modificar su nombre de etiqueta, nombre de clase y *classpath*.

2. Archivos necesarios para poder ejecutar

Para poder ejecutar la aplicación, no es necesario ningún archivo específico de inicio, es decir, se puede acceder a la aplicación y entrar al editor de mapas “*File → Create or Edit Map*” para crear y/o editar un mapa que será simulado en un futuro. Sin embargo, para llevar a cabo la simulación de un mapa para probar sistemas de recomendación sobre el mismo, hacen falta algunos ficheros:

1. **Ficheros con el contenido del mapa a simular:** Estos ficheros son tres y contienen la información correspondiente al mapa sobre el que simular. Estos archivos son el fichero de ítems, el de salas y el del grafo.

El archivo de ítems, contiene el número total así como todos ellos con sus correspondientes características (incluidas las coordenadas de su posición en el mapa); el archivo de salas contiene la información general del mapa (nombre, anchura, altura, número de salas, etc.) y todas las salas con sus correspondientes elementos básicos para la generación de mapas (esquinas, puertas y escaleras) dotados de valor con sus coordenadas; y el archivo del grafo, similar al de salas pero conteniendo este la correspondencia de “IDs” de los elementos (para la creación del grafo) así como las conexiones entre elementos conectables (puertas y escaleras).

La localización de estos archivos será, siguiendo la estructura del proyecto: “*Simulator2022/resources/maps*”. En dicha carpeta se podrán encontrar varias subcarpetas con mapas creados. Se pueden encontrar los siguientes mapas: museo *MoMA*, actualizado para el nuevo simulador generalizado; escenario simple y escenario con salas poligonales complejas (con múltiples lados y ángulos convexos), para probar el correcto funcionamiento de las trayectorias de los

usuarios durante la simulación en distintas de salas; y centro comercial *GranCasa*, con todos los ítems y que se ha de probar con los nuevos datos generados.

2. **Bases de datos:** Se necesitan tanto la base de datos con toda la posible información del mapa creado como las bases de datos “modelo” que se crearán para los usuarios en cada simulación y en las que irán almacenando información. En función del tipo de comunicación, utilizarán o bien una única BD centralizada o cada usuario almacenará la información en su propia BD de valoraciones así como una con la información intercambiada (algoritmo *P2P*).

El archivo correspondiente a la base de datos con toda la información actuará como un “oráculo”, es decir, contendrá todas las valoraciones que cada usuario daría a cada ítem bajo cada contexto posible que pueden originarse a lo largo de la simulación, es decir, todas las posibles combinaciones de usuarios, ítems y contextos: “ $n^{\circ}\text{registros} = n^{\circ}\text{usuarios} * n^{\circ}\text{ítems} * n^{\circ}\text{contextos}$ ”. Esta información se encontrará en la tabla “*user_item_context*”, tabla principal a rellenar con datos para la simulación al crear una nueva fuente de datos para un nuevo escenario (con *AUTO-DataGenCARS* por ejemplo), además de las tablas “*user*”, “*item*” y “*context*” en caso de requerir nuevos datos encontrados en dichas tablas (nuevos o diferentes usuarios, ítems y/o contextos en la nueva fuente de datos). El archivo referente a la BD de usuarios es similar al anterior pero tendrá la tabla de valoraciones vacía y será copiada y renombrada para almacenar las valoraciones de los usuarios (en función del tipo de red seleccionada para la comunicación de los usuarios, una centralizada o una para cada usuario). El archivo referente a la cola de información de las comunicaciones es análogo al anterior pero servirá de modelo para almacenar la información intercambiada en las comunicaciones cuando se utilice la red *P2P*.

Estos archivos deben estar situados en la carpeta “*Simulator2022/resources/db*” del proyecto.

3. **Algoritmos de simulación:** Dada la generalización del Simulador, se ha de disponer del fichero que contiene los algoritmos disponibles y las características de estos (uso de red y redes que utiliza). Este archivo se ha de encontrar en la ruta “*Simulator2022/resources/recommenders*”. En caso de añadir un nuevo algoritmo, se debe implementar su comportamiento en la función de recomendación.

La distribución esperada de todos los diferentes ficheros necesarios para la

simulación en los diversos directorios puede observarse en la Figura 1.

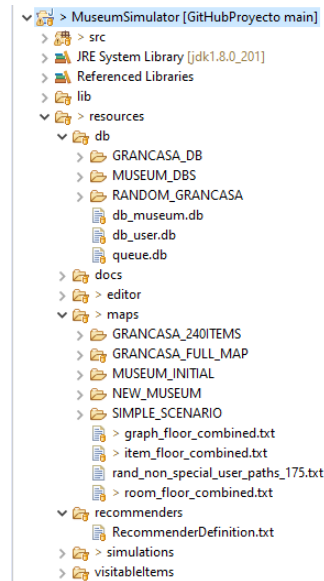


Figura 1: Estructura del proyecto y archivos necesarios para la simulación

3. Ejecuciones individuales mediante GUI

Una vez se disponga de los archivos necesarios para la simulación, se puede proceder a simular. Para ello, lo primero es definir la configuración de la simulación a realizar. Esto se puede hacer en la ventana a la que se accede mediante “*Simulation* → *Configuration*”.

En esta ventana, se podrán definir los parámetros deseados para la simulación. Se mostrarán por defecto los datos de la simulación del museo. La adaptación al nuevo escenario (GranCasa) así como cualquier nuevo mapa que pueda ser creado es sencilla, pues únicamente hay que modificar la escala (“*1 km represents in pixels [pixel]*” = “4000 (pixels)” en caso del mapa de GranCasa). El resto de parámetros se pueden modificar al gusto independientemente del escenario, como aumentar la velocidad de movimiento de los usuarios o el tiempo real por segundo de iteración. Este parámetro permite realizar simulaciones más rápidas que el tiempo real que representan, pero se ha de mantener bajo o igual al tiempo real simulado cuando se apliquen algoritmos de filtrado colaborativo basado en usuarios (como se explica en el Anexo C).

Opcionalmente, se pueden modificar las opciones de visualización (escalar el mapa para una visualización diferente o bien activar/desactivar la GUI en caso de delays con las operaciones de refresco de pantalla) en la opción del menú “View”.

Tras haber definido la configuración, se puede proceder a simular: “*Simulation* → *Start*”. Esta acción debería haberse habilitado tras definir la configuración.

4. Ejecuciones/evaluaciones desatendidas

Una nueva funcionalidad incorporada a este simulador son las evaluaciones desatendidas. Dicha funcionalidad permite llevar a cabo ejecuciones por lotes, sin interfaz gráfica, para recopilar múltiples resultados (probando diferentes algoritmos o diferentes parámetros de configuración en cada una) de forma automática.

Para llevar a cabo las evaluaciones desatendidas, debemos acceder a “*Simulation* → *Neglected Evaluations*”. Se abrirá una ventana emergente que pedirá seleccionar el archivo (.txt) donde se encuentran las configuraciones a ejecutar y la carpeta donde se desean almacenar los resultados de cada simulación. El archivo de configuraciones para cada simulación se deberá crear siguiendo el paradigma **clave-valor**, de manera análoga al uso de la clase *java.util.Properties*, es decir, especificando cada parámetro de configuración seguido de su número de simulación (por ejemplo: “*timeAvailableUser_1*”) e igualado al valor que se le desee proveer (por ejemplo: 1).

Se pueden encontrar ejemplos de archivos de configuraciones en la carpeta “*Simulator2022/resources/simulations*”.

5. Ejemplo guiado de simulación de un escenario - GranCasa

Tras lo comentado en los apartados anteriores se procede a presentar un ejemplo práctico de una simulación. Para este ejemplo se utilizará el mapa de GranCasa, creado con el Editor de Mapas.

En primer lugar, es necesario colocar los archivos necesarios para la simulación en las rutas correctas. Se necesitan los archivos referentes al mapa, generados con el editor; los archivos referentes a la base de datos, generados con herramientas externas (recomendado *DataGenCARS*); y el archivo con la información de los recomendadores (no modificar a no ser que se añadan por código nuevos recomendadores). Estos archivos se han de ubicar en las ubicaciones tal y como muestra la Figura 1.

Con los archivos necesarios, se puede proceder a ejecutar el proyecto. Al hacerlo se abre la ventana principal de la aplicación sobre la cual se plasmará la información de la simulación una vez establecida la configuración. Para establecer la configuración, se accede a la ventana *Configuration* “*Simulation* \rightarrow *Configuration*” (Figura 2).

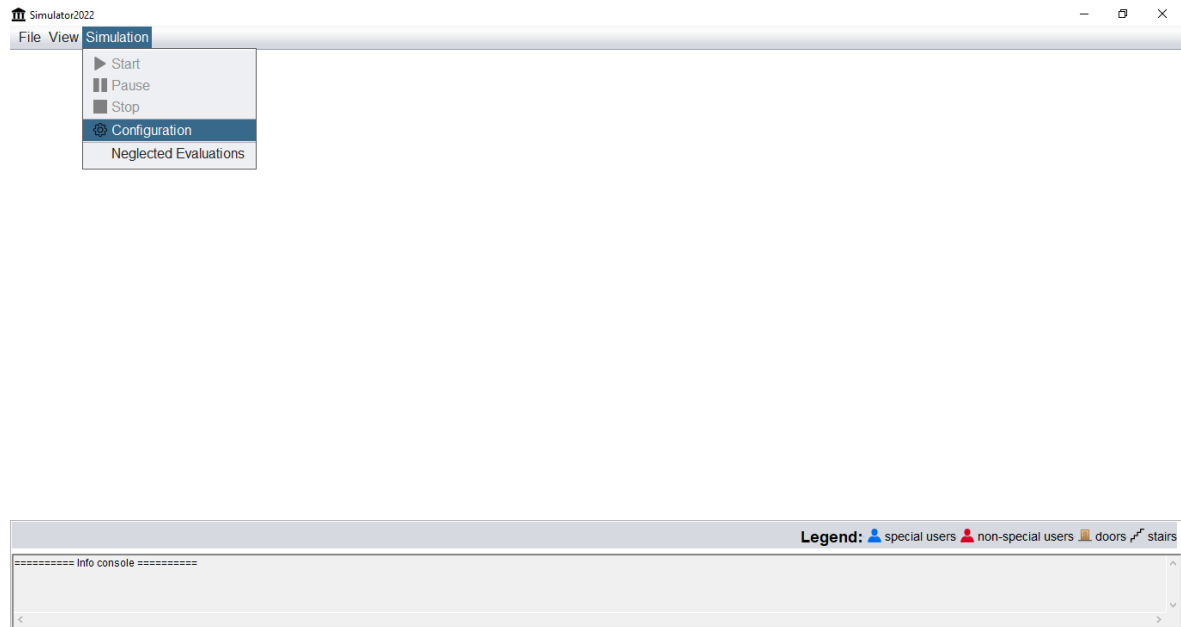


Figura 2: Acceso a la ventana *Configuration*

Con esta acción, se muestra una ventana que permite definir todos los parámetros de la configuración de la simulación a realizar. Cabe destacar que la escala del mapa de GranCasa es: 4 píxeles equivalen a 1 metro, o lo que es lo mismo, “*1 km represents in pixels = 4000*”. El resto de parámetros pueden establecerse tal y como se desee llevar a cabo la simulación. En el caso de la Figura B.3, se simulará a velocidad x30 (“*Real time per iteration = 30*”), se aplicará la estrategia *NPOI* para establecer los caminos de los usuarios sin recomendador, y el algoritmo de recomendación será *K-Ideal*. Cuando se dispone de la configuración deseada, se pulsa el botón “*Save and load combined floors*”.

Configuration

Parameters for the simulation:

FT,RAND,C

FT,NPOI,P2P

Time available for the user [hour]

1

Delay observing painting [seconds]

30

Real time per iteration [seconds]

30

Iteration time/Screen refresh time [seconds]

1

Time for the paths [hour]

1

User velocity [km/h]

3

1 km represents in pixels [pixel]

4000

TTL of the items to propagate [seconds]

180

Time on the stairs [seconds]

60

Minimum time to update recommendation [seconds]

30

Communication range [m]

250

Maximum knowledge base size [Mb]

1

Communication bandwidth [Mbps]

54

Latency of transmission [s]

1

Time to change the mood [seconds]

1800

☐ Use fixed seed for randomness

Seed number

0

Parameters for the experimentation:

Number of special users [1-176]

1

Number of non-special users [1-176]

175

File name

npoi_non_special_user_paths_175.txt

☒ Generation of dynamic non-special user paths

Path strategy

Near POI (NPOI)

Recommendation algorithm

K-Ideal (K-Ideal)

Threshold of recommendation [1-5]

2.5

Threshold of similarity [0-1]

0.5

How many items to recommend

10

Type of network

Select a type of network

Propagation Strategy

Select a strategy

Probability of user disobedience

0.4

Number of votes received

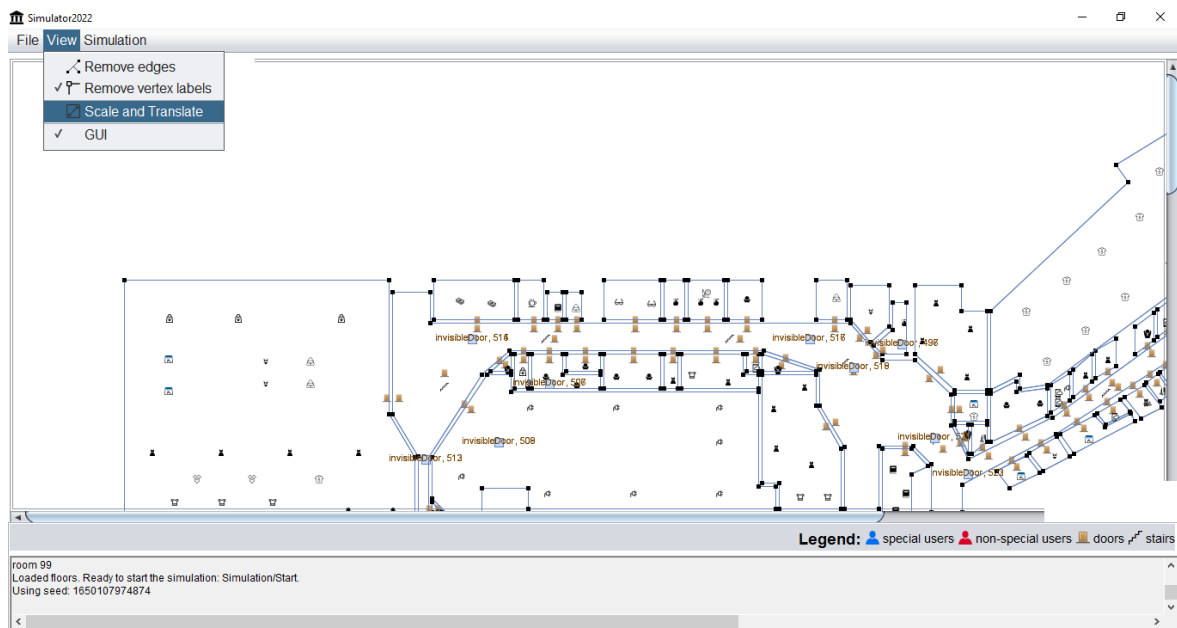
40

Save and load combined floors

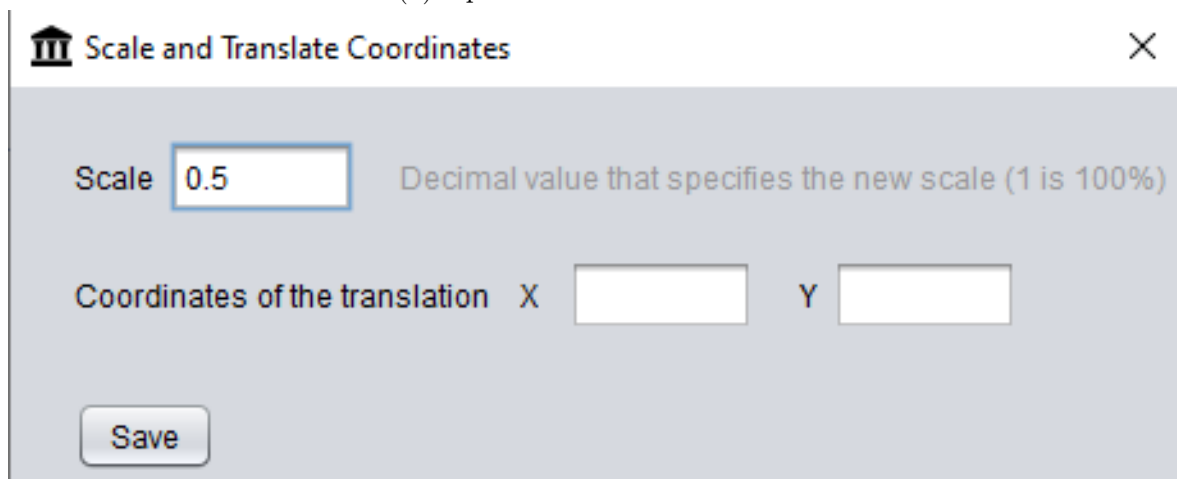
Figura 3: Ventana *Configuration* - Ejemplo de configuración para la simulación

Sin embargo, como se puede observar en la Figura B.3, este mapa es más grande que el utilizado originalmente, es decir, el de las plantas 4 y 5 del museo MoMA, por lo que para visualizarlo mejor antes de comenzar con la simulación se puede escalar el mapa (Figura B.4a).

En este caso, se aplicará una escala que lo reduzca a la mitad y, como no se le quiere aplicar ningún movimiento, se dejan los campos referentes a las coordenadas de traslación vacíos (Figura B.4b).



(a) Opción *Scale and Translate*



(b) Ventana *Scale and Translate*

Figura 4: Proceso *Scale and Translate*

Tras confirmar la configuración (y es calado si así se desea), se procede a cargar en el simulador la información de los ficheros (referente al mapa). Cuando toda la

información se ha cargado correctamente, se puede visualizar el mapa (si la GUI está activada) y se muestra en la consola localizada en la parte inferior de la ventana un mensaje que indica que se puede proceder a simular. Para ello, como indica el mensaje, hay que acceder a “*Simulation* → *Start*” (Figura 5). Cuando se lleva a cabo dicha acción, se inicializan las trayectorias a seguir por los usuarios sin recomendador y las conexiones con las bases de datos y comienza a ejecutarse el algoritmo de simulación. También se puede visualizar en dicha imagen el cambio tras escalar, pudiendo observarse ahora dos de las tres plantas del centro comercial.

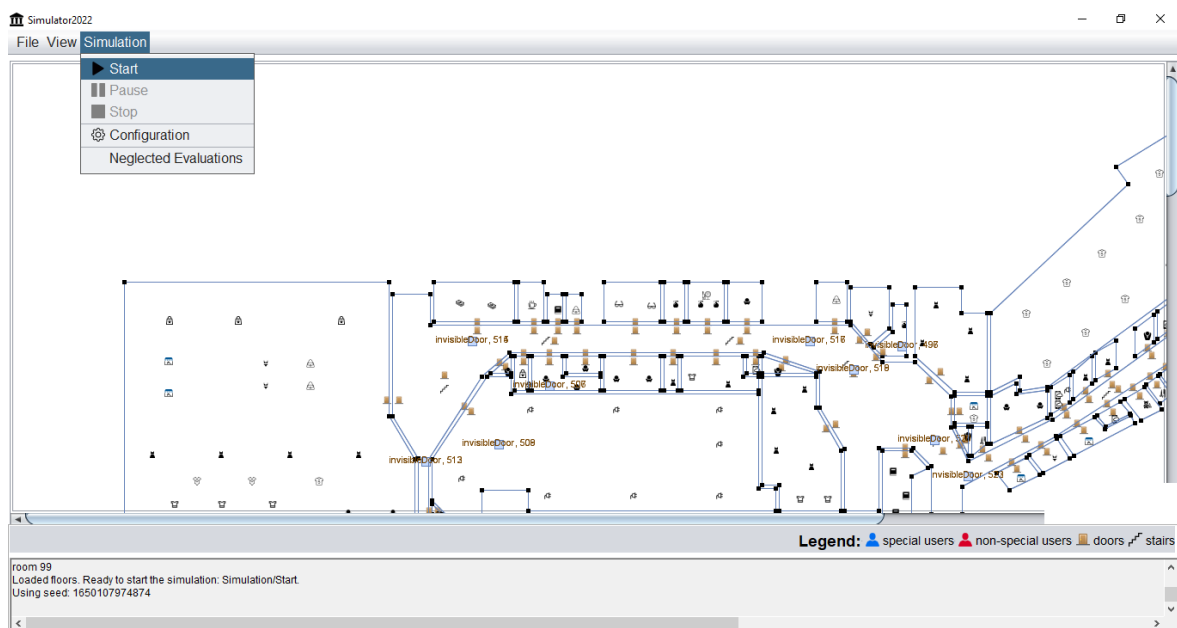


Figura 5: Comenzar la simulación

Una vez comenzada la simulación, se muestran por la consola de la ventana de visualización mensajes que indican la evolución de la simulación. Además, si la GUI está activada, se verá la evolución de la simulación en el panel principal de la pantalla (escenario, movimiento de los usuarios en cada instante de la simulación e información de los elementos del escenario si se coloca el cursor sobre alguno de ellos). Esto se puede ver en la Figura 6.

Una vez terminada la simulación, se muestra por consola un mensaje que indica el fin de la misma, por lo que ya se puede cerrar la aplicación o proceder con otra acción (otra simulación o acceder al editor).

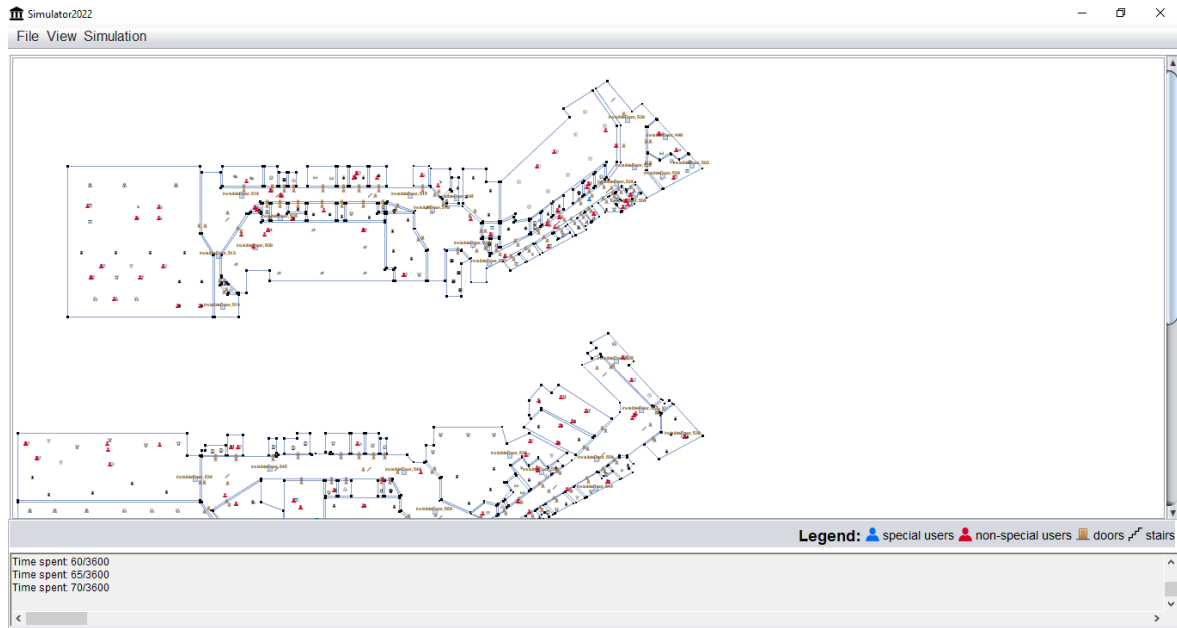


Figura 6: Simulación en marcha

6. Ejemplo guiado de simulación de un escenario - museo MoMA u otros escenarios

La ejecución para cualquier tipo de mapa es equivalente a la mostrada en el apartado anterior (Sección 5), pues los procesos y acciones a seguir son los mismos. La única diferencia que permite simular otros mapas (el del museo *MoMA* por ejemplo) se encuentra en los archivos que se ubiquen en las rutas especificadas en el Apartado 2. Estos ficheros contienen la información del mapa sobre el cual simular y la información de los usuarios, ítems, contextos y valoraciones (de cada usuario a cada ítem en cada contexto) para el escenario que se desea simular.

Para simular otro escenario, por ejemplo el escenario del museo *MoMA*, se deben sustituir los siguientes ficheros:

- Dentro del propio proyecto, en la ruta en la que se deben situar los ficheros referentes al mapa (`MuseumSimulator/resources/maps`), se encuentran además diversas carpetas con los mapas disponibles. En dichas carpetas se ubican los archivos referentes a los ítems, salas y grafo del escenario correspondiente. Para utilizar el mapa del museo en las simulaciones, se copia el contenido (los tres ficheros) de la carpeta `MuseumSimulator/resources/maps/MoMA_Museum` en la ruta del mapa que utiliza el simulador (`MuseumSimulator/resources/maps`), sustituyendo los archivos que se estuvieran utilizando.

- De forma análoga a la gestión de ficheros del escenario, para utilizar la base de datos de otro escenario (museo) se ha de acceder a la ruta `MuseumSimulator/resources/db` y sustituir los archivos de la carpeta del mapa que se quiera simular (en este caso *MUSEUM_DBS*) por los que haya en la carpeta raíz de las bases de datos (*db*).

Los ficheros referentes a los mapas de los diferentes escenarios que se encuentran disponibles (diferentes carpetas en `MuseumSimulator/resources/maps`) han sido generados mediante la nueva herramienta implementada en este proyecto (creador de mapas). La única excepción es el caso del mapa del museo, el cual ya se encontraba disponible en el proyecto original, el cual no se encontraba adaptado a las nuevas características de este proyecto (sobre todo en lo referente a la gestión de “subsalas”). Esta adaptación, sin embargo, es prácticamente automática, pues basta con cargarlo en el creador de mapas y volver a guardarlo, quedando así aplicadas las modificaciones pertinentes para poder ser reconocido por el simulador.

Respecto a las bases de datos (y la información contenida en ellas) de los escenarios, tanto en el caso del proyecto original (museo) como en el caso del nuevo escenario (*GranCasa*), han sido generados con la ya mencionada herramienta *AUTO-DataGenCARS*.

Anexos C

Parámetros de configuración del simulador

En este Anexo se explican los parámetros de configuración del simulador, los cuales definen e impactan directamente en la simulación a realizar. La Figura C.1 muestra la ventana de configuración, en la cual se muestran todos ellos. Además, se crean dos botones para iniciar simulaciones rápidas (FT, Fast Test) usando distintas trayectorias (RAND y NPOI) y algoritmos de recomendación y uso de red (UBCF con red Centralizada o P2P). Son útiles para pruebas rápidas o depuración.

Configuration

Parameters for the simulation: FT,RAND,C FT,NPOI,P2P

Time available for the user [hour] 1

Delay observing item [seconds] 30

Real time per iteration [seconds] 1

Iteration time/Screen refresh time [seconds] 1

Time for the paths [hour] 1

User speed [km/h] 3

1 km represents in pixels [pixel] 6597

TTL of the items to propagate [seconds] 180

Time on the stairs [seconds] 60

Minimum time to update recommendation [seconds] 30

Communication range [m] 250

Maximum knowledge base size [Mb] 1

Communication bandwidth [Mbps] 54

Latency of transmission [s] 1

Time to change the mood [seconds] 1800

☐ Use fixed seed for randomness

Seed number 0

Parameters for the experimentation:

Number of special users [1-176] 1

Number of non-special users [1-176] 175

File name

☐ Generation of dynamic non-special user paths

Path strategy Select a strategy

Recommendation algorithm Select an algorithm

Threshold of recommendation [1-5] 2.5

Threshold of similarity [0-1] 0.5

How many items to recommend 10

Type of network Select a type of network

Propagation Strategy Select a strategy

Probability of user disobedience 0.4

Number of votes received 40

Save and load combined floors

Figura 1: Ventana de configuración del simulador

Los parámetros disponibles que permiten definir la configuración de la simulación se clasifican en dos grupos:

- *Parámetros de la simulación*: sirven para definir las características que afectarán a la simulación que se desea realizar. Se podrán definir parámetros como: el tiempo de simulación, velocidad de movimiento de los usuarios, tiempo real que representa cada iteración de la simulación, etc.
- *Parámetros para la experimentación*: determinan el tipo de experimento que se quiere realizar. Se podrá definir el algoritmo que adoptarán los usuarios sin recomendador en su trayectoria, el algoritmo de recomendación que probarán los usuarios con recomendador, los parámetros a considerar por el algoritmo de recomendación (umbrales de valoración dada, similitud y número de ítems a recomendar) y la red de comunicaciones que utilizarán los usuarios (en caso de que el algoritmo de recomendación permita diferentes tipos de comunicación).

Cada uno de ellos tiene un propósito específico que permite al usuario plantear diferentes experimentos con la variación de sus valores. Los parámetros configurables son:

- Time available for the user: tiempo (en horas) del que disponen los usuarios durante la simulación.
- Delay observing item: tiempo (en segundos) que estará cada usuario viendo cada ítem.
- Real time per iteration: tiempo real (en segundos) que representa cada iteración. Este parámetro permite ejecutar simulaciones mucho más rápidas que el tiempo real que están simulando. El valor introducido equivale a la velocidad de simulación (si el valor introducido es 5, la simulación se ejecutará a velocidad x5).
- Iteration time/Screen refresh time: cada cuántos segundos se ha de refrescar la pantalla, mostrando la evolución de la simulación de forma visual (para simulaciones con GUI).
- Time for the paths: tiempo límite (en horas) hasta el cual se puede generar la trayectoria de los usuarios.
- User speed: velocidad media (km/h) a la que se mueven los usuarios.

- 1 km represents in pixels: parámetro de la escala. Su valor equivale al número de píxeles que representan un kilómetro (relación entre kilómetros reales y píxeles en la interfaz).
- TTL of the items to propagate: cuántos segundos permanece la información de las votaciones de los usuarios en la red de comunicación *P2P*.
- Time on stairs: cuánto tiempo le cuesta a los usuarios subir/bajar escaleras para moverse entre plantas.
- Minimum time to update recommendation: si el número de votos que recibe el usuario con recomendador es mayor que umbral máximo de votos (parámetro de experimentación), tiempo que le cuesta al algoritmo de recomendación actualizar la trayectoria del usuario.
- Communication range: rango máximo de comunicación (en metros) cuando se utiliza la red *P2P*. Su representación se observa en la Figura 1 de [2].
- Maximum knowledge base size: tamaño máximo (en Mb) de almacenamiento disponible en la cola de información cuando se utiliza *P2P*.
- Communication bandwidth: ancho de banda (en Mbps) disponible en la comunicación entre usuarios de la red *P2P*.
- Latency of transmission: segundos de retardo en la transmisión de comunicaciones (en *P2P*).
- Time to change the mood: tiempo mínimo para modificar el estado de ánimo de los usuarios.
- Seed number: en caso de activarse, número de semilla establecido para la simulación (pudiendo así realizar simulaciones repetibles con el mismo valor de semilla).
- Number of special/non-special users: número de usuarios con y sin recomendador durante la simulación. La suma de estos no puede superar el máximo establecido, pero sí permitir cualquier combinación hasta dicho número.
- Path strategy: estrategia de trayectoria que seguirán los usuarios sin recomendador durante la simulación. Los valores posibles equivalen a los algoritmos simples (RAND, ALL y NPOI).

- Recommendation algorithm: algoritmo que utilizarán los usuarios con recomendador (special users). Los algoritmos disponibles se explican en la Sección 2.1.
- Threshold of recommendation: umbral mínimo de valoración a utilizar en la aplicación de recomendaciones. Su valor puede oscilar entre 1 y 5 (valores posibles de valoraciones). Utilizado en los algoritmos no simples.
- Threshold of similarity: valor del umbral mínimo de similitud entre usuarios (perfil de usuario), aplicando únicamente recomendaciones basadas en aquellas proporcionadas por otros usuarios similares. El valor introducido debe estar entre 0 y 1. Utilizado en los algoritmos no simples.
- How many items to recommend: número máximo de ítems a recomendar entre los disponibles, ordenados por orden de puntuación (valoración).
- Type of network: tipo de red de comunicación utilizada. Puede ser centralizada o *P2P*. Únicamente aplica si el algoritmo de recomendación es UBCF. En cualquier caso se emplea una única base de datos para almacenar toda la información generada por los usuarios durante la simulación (como la red centralizada) pero no se utiliza para la comunicación entre los usuarios.
- Propagation Strategy: estrategia de propagación de la información para la red *P2P*.
- Probability of user disobedience: probabilidad con la que los usuarios con recomendador desobedecerán las recomendaciones que se les han brindado y visitarán otro ítem.
- Number of votes received: si el usuario recibe un número de votos mayor al definido en este parámetro, se actualiza su trayectoria a partir de toda la información disponible.

El valor especificado en estos parámetros impacta directamente sobre las simulaciones, modificando sus características, su desempeño y/o sus resultados. Se pueden realizar múltiples pruebas con combinaciones diferentes de estos para ver su impacto sobre las simulaciones y las valoraciones de los usuarios.

Mapas creados con el editor

En este Anexo se presentan los escenarios creados con el editor, tanto el escenario del centro comercial *GranCasa* como el del Campus Río Ebro.

1. Mapa *GranCasa*

El escenario principal utilizado como caso de estudio en este proyecto es el mapa de *Grancasa*. Este es el motivo por el cual su elaboración se pone un especial detalle.

Para la construcción del escenario en el editor, se toman como referencia dos fuentes, la página del *Colegio Oficial de Arquitectos de Aragón* (<https://www.coaaragon.es/Default.aspx?Cod=202>) y el sitio web del propio Centro Comercial *Grancasa* (<https://www.grancasa.es/mapa/#/>). Es necesario combinar la información encontrada en ambas para poder representar dicho mapa, además de un trabajo adicional que se comenta a continuación.

En la primera referencia únicamente se encuentran los planos de la planta baja y, aunque dispone de puntos de referencia, no se encuentran las acotaciones (medidas) de cada uno de los elementos (tiendas, pasillos, etc.). En cambio, en este enlace sí se comentan las dimensiones totales del centro, 400x90 metros, lo que se toma como referencia para la creación de las diferentes salas. De la web oficial del centro se pueden extraer los mapas de todas las plantas con sus respectivas tiendas/restaurantes, aunque de nuevo sin ningún tipo de medidas.

Por este motivo, la mayor parte del trabajo realizado en la elaboración de este escenario se realiza fuera del editor de mapas, imprimiendo las tres plantas del centro comercial de las páginas comentadas, midiendo cada elemento (altura, anchura y posición) y aplicando la equivalencia de dichas medidas frente a las dimensiones totales del mapa (transformadas a centímetros sobre el papel). Esta tarea suma un tiempo mucho mayor del necesario para crear un mapa directamente con el editor si se

dispone de las medidas pertinentes (el 70 % aproximadamente), pero se realiza ya que se quiere una representación del escenario de prueba lo más próxima y fiel a la realidad.

Con las medidas obtenidas, basta con “dibujar” las salas como se podría hacer con cualquier otra herramienta de dibujo común. Con el lápiz y la barra de estado se colocan los elementos sobre el mapa y, si las medidas no han sido completamente precisas en una primera instancia, se mueven los objetos con la herramienta de movimiento o bien desde la ventana de propiedades del elemento, proporcionando con esta última la posición exacta deseada (en píxeles o en metros). Finalmente se establecen las conexiones entre puertas y escaleras.

Una vez obtenidas las medidas y creadas las plantas en el editor, se han de crear los tipos de ítems que componen el mapa y que pueden visitar los usuarios durante la simulación. En este caso, los ítems a visitar son secciones dentro de una tienda. Se elige este como el tipo de ítems ya que las tiendas completas son demasiado generales, con mucha diferencia de tamaño entre ellas, y sobre todo porque muchas tiendas venden diferentes tipos de productos y engloban varias categorías diferentes.

Las secciones así como sus tipos o categorías se extraen del apartado de tiendas del sitio oficial del centro (<https://www.grancasa.es/tiendas/>). El escenario final está formado por 26 tipos de ítems diferentes a partir de las categorías y subcategorías encontradas en este enlace. La creación de estos tipos de ítems conlleva la búsqueda y descarga de sus correspondientes iconos representativos en formato *PNG*. De nuevo esta tarea implica un tiempo mucho mayor del necesitado para la creación de los tipos de ítems en el editor.

Además, se buscan una a una las tiendas y las categorías en las que se encuentran para crear instancias de los ítems a lo largo del mapa. En aquellas salas que aparecen en más de una categoría en la web, se crea al menos un tipo de ítem por cada categoría. Además, en las tiendas con una extensión mayor, se crean múltiples secciones a lo largo de su extensión (por ejemplo, en la tienda *Decathlon* se incluyen las secciones de fútbol, baloncesto, montaña, atletismo, etc.). Las tiendas con unas dimensiones pequeñas y que comprenden una única categoría contienen una única sección que engloba la propia tienda. El escenario final contiene un total de 283 secciones de tiendas diferentes.

En las Figuras 2, 4 y 6 se muestran las plantas creadas con el editor, las cuales se pueden comparar con las Figuras 1, 3 y 5, extraídas del sitio oficial y

tomadas como referencia en la construcción del mapa con el editor. Cabe destacar que en el diseño final, no todas las tiendas con ángulos convexos se dividen en subsalas (o en tantas subsalas como ángulos convexos contienen) si todos sus ítems son accesibles mediante una líneas rectas sin interferir con las paredes.



Figura 1: Mapa del sitio oficial de *Grancasa* - Planta 0

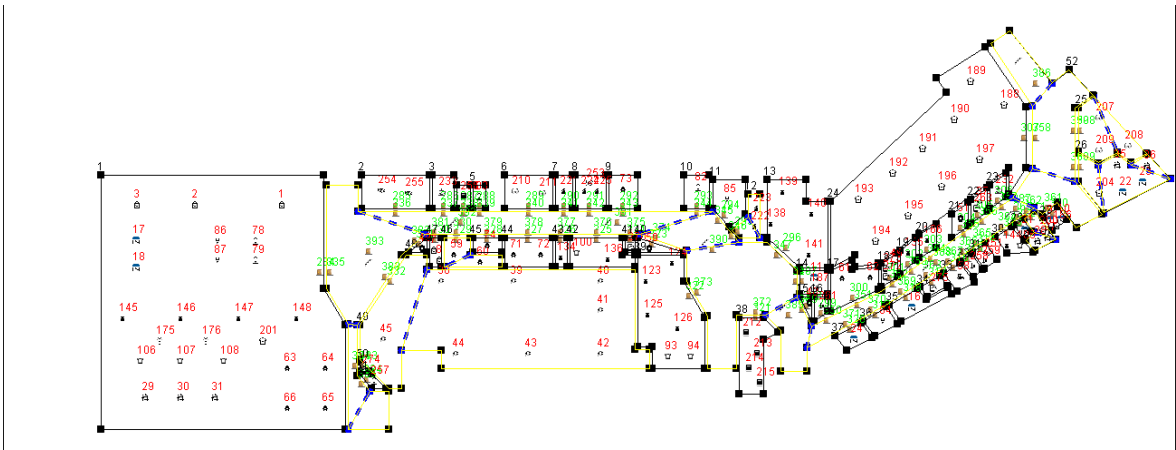


Figura 2: Mapa de *Grancasa* - Planta 0

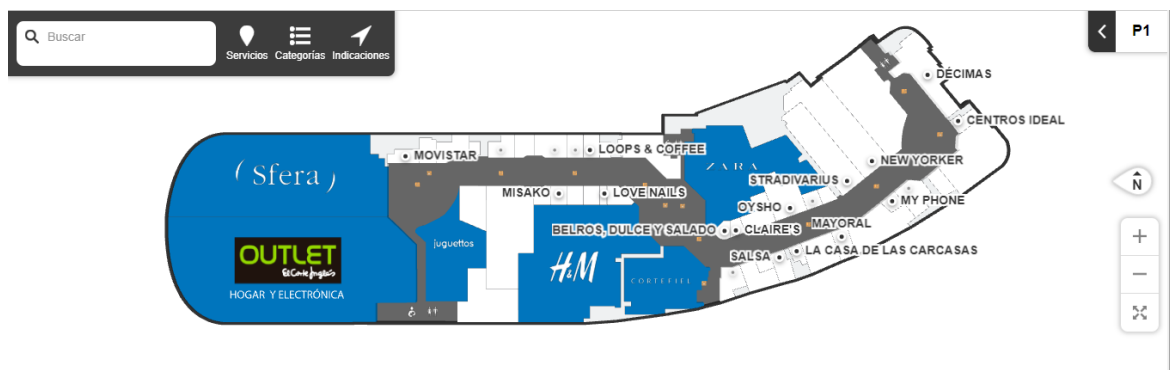


Figura 3: Mapa del sitio oficial de *Grancasa* - Planta 1

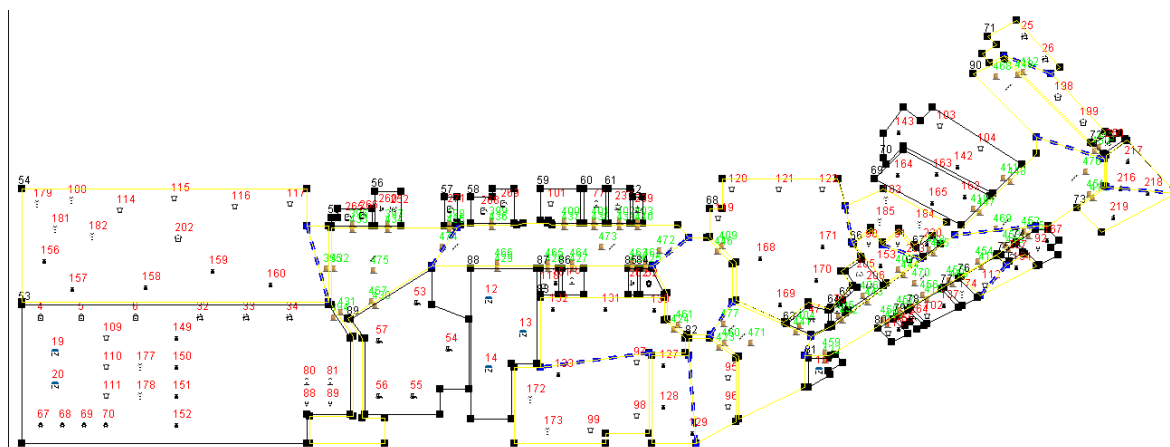


Figura 4: Mapa de *Grancasa* - Planta 1

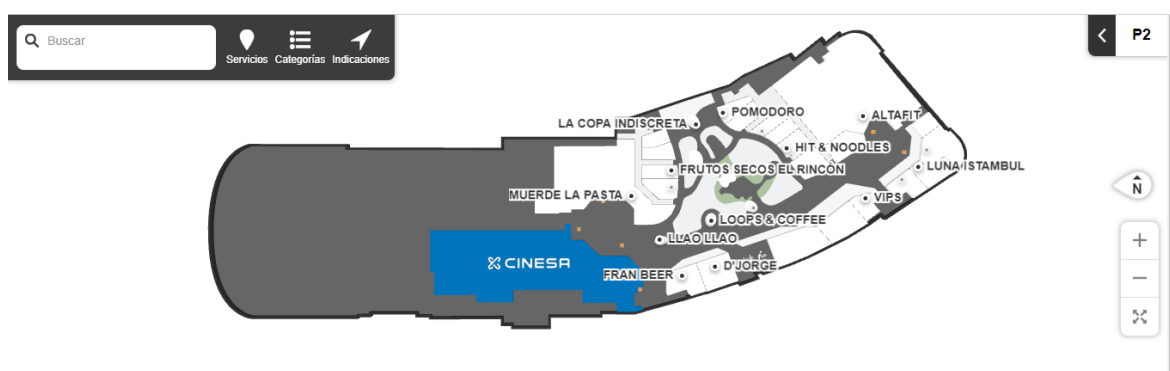


Figura 5: Mapa del sitio oficial de *Grancasa* - Planta 2

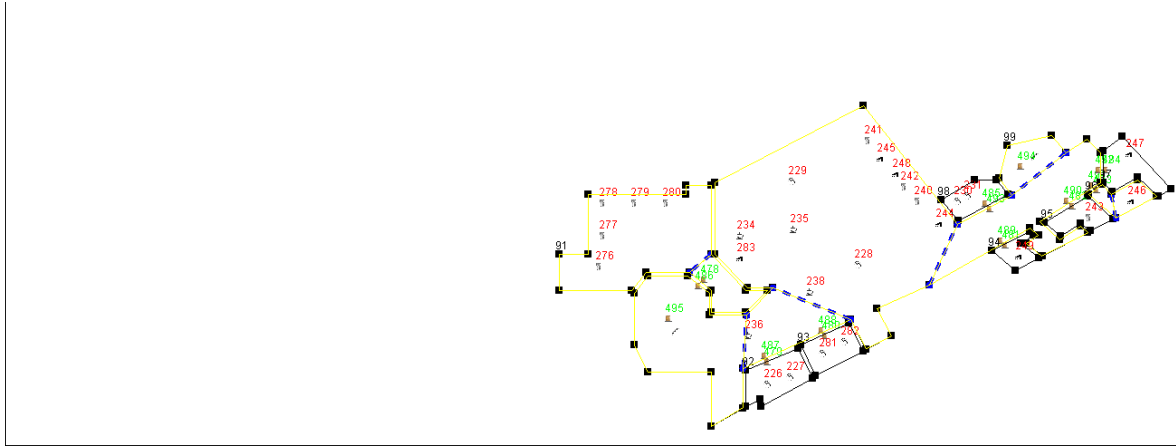


Figura 6: Mapa de *Grancasa* - Planta 2

2. Mapa Campus Río Ebro

Para comprobar el alcance de esta herramienta se crea un escenario al aire libre. La idea es simular un evento social, desarrollado al aire libre pero conteniendo también edificios, y en el que los ítems visitables sean personas que se moverán durante la simulación (ítems móviles).

El escenario creado se basa en el Campus Río Ebro, de la Universidad de Zaragoza. El caso de estudio a simular es una jornada de intercambio, en la cual varios expertos de diferentes lugares se reparten por el campus y se van desplazando compartiendo sus conocimientos con aquellos alumnos que lo deseen. En este caso, los expertos se interpretan como ítems visitables y los alumnos como los usuarios de la simulación.

Para la construcción de este escenario se han consultado dos fuentes, la imagen del campus encontrada en la página de la Escuela Universitaria de Ingeniería Técnica Industrial de Zaragoza (<http://euitiz.unizar.es/es/betancourt.php?seccion=situacion>) y la herramienta *Google Maps* (<https://www.google.es/maps/>). Dado que no se encuentran medidas en la primera fuente, para la elaboración de este escenario se han tomado unas medidas aproximadas extraídas de *Google Maps* como muestra la Figura 7.

La Figura 8 muestra el mapa completo y las Figuras 9, 10 y 11 muestran en mayor detalle cada una de las plantas. Estas figuras han sido extraídas del editor de mapas, sin utilizar el objeto divisor en subsalas para mayor claridad de las imágenes. Sin embargo, en caso de utilizar este escenario, habría que hacer uso de dicho objeto para el correcto movimiento de los usuarios y expertos durante la simulación.

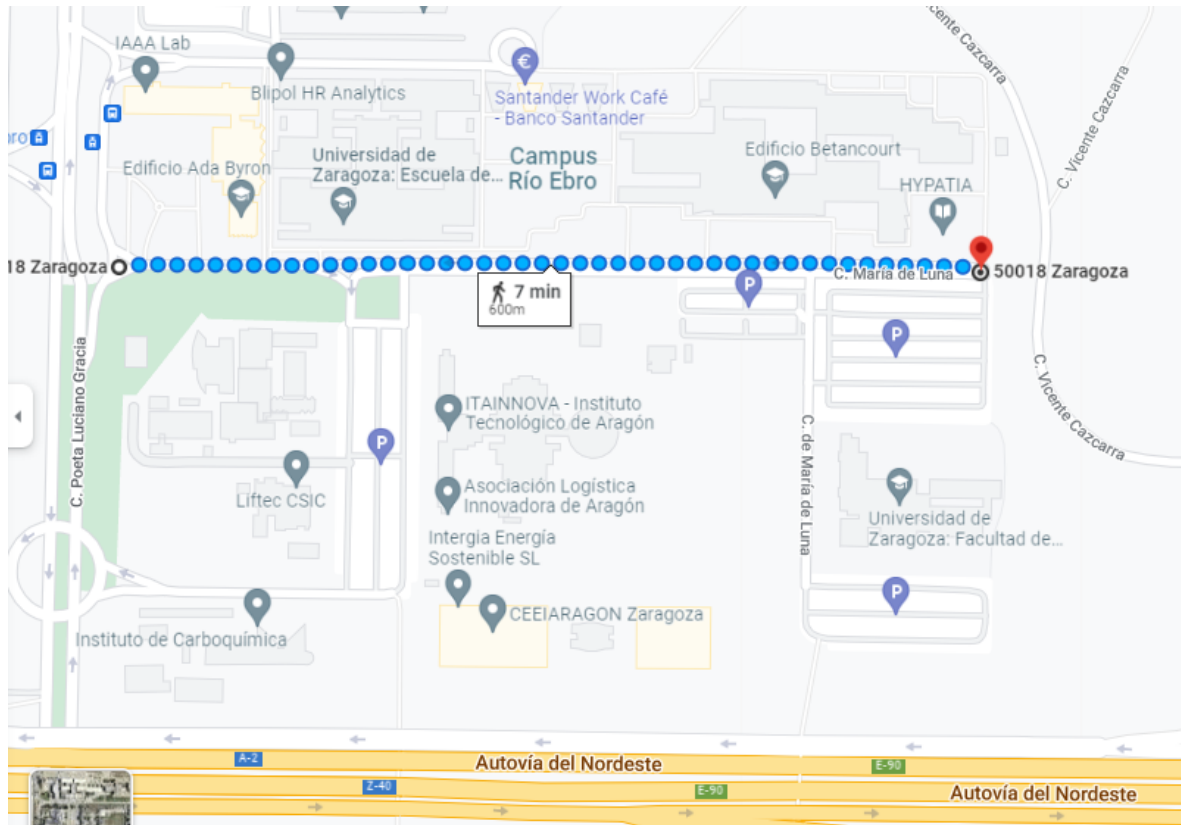


Figura 7: Mapa Campus Río Ebro - Referencia *Google Maps*

En este escenario se observa el alcance y las limitaciones de este proyecto. Por un lado, sí es posible representar el mapa, como se puede observar en las figuras. Sin embargo, existen otro tipo de limitaciones de visualización o actuación.

Cuando se representa un escenario con elementos con dimensiones tan dispares (escenario al aire libre con edificios representados en menor dimensión), se encuentra un problema con la escala. Si el mapa se representa como en las figuras de este apartado, es complicado representar las salas del interior de los edificios, pues su tamaño sería demasiado pequeño y los ítems contenidos en ellas se solaparían. Por otro lado, si se amplía el mapa para poder representar el interior de los edificios, el mapa creado tendría un tamaño muy grande, dificultando la visión en el simulador.

Por otro lado, se encuentran ítems con comportamientos propios más amplios que los atributos asignados a los tipos de ítems que se pueden crear con el editor. Por tanto, sería necesario extender la clase ítem y añadir el comportamiento requerido (trayectorias) y su posibilidad de creación en el editor. Por otro lado, habría que reconocer y tratar también este comportamiento en el simulador como el movimiento de los usuarios de la simulación (moviendo también los ítems con trayectoria).



Figura 8: Mapa Campus Río Ebro - Escenario Completo

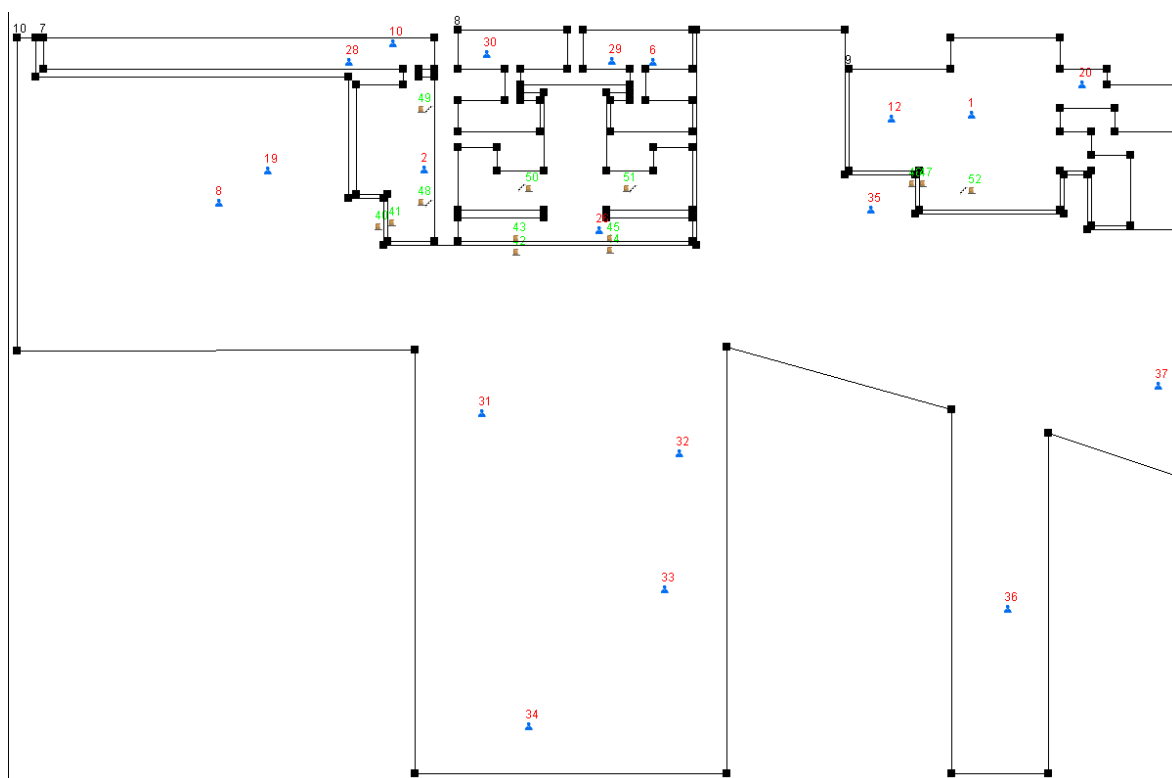


Figura 9: Mapa Campus Río Ebro - Planta calle

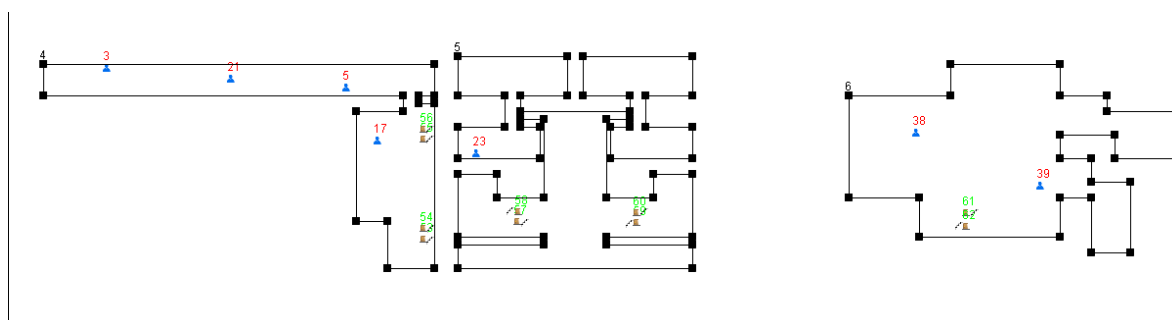


Figura 10: Mapa Campus Río Ebro - Primera planta (edificios)

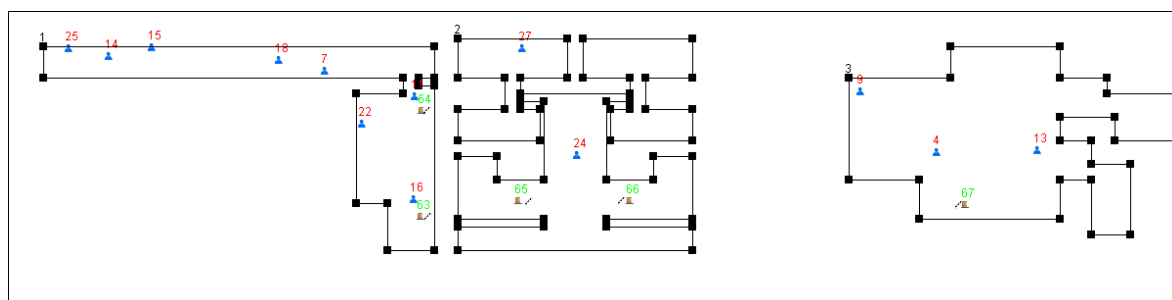


Figura 11: Mapa Campus Río Ebro - Segunda planta (edificios)