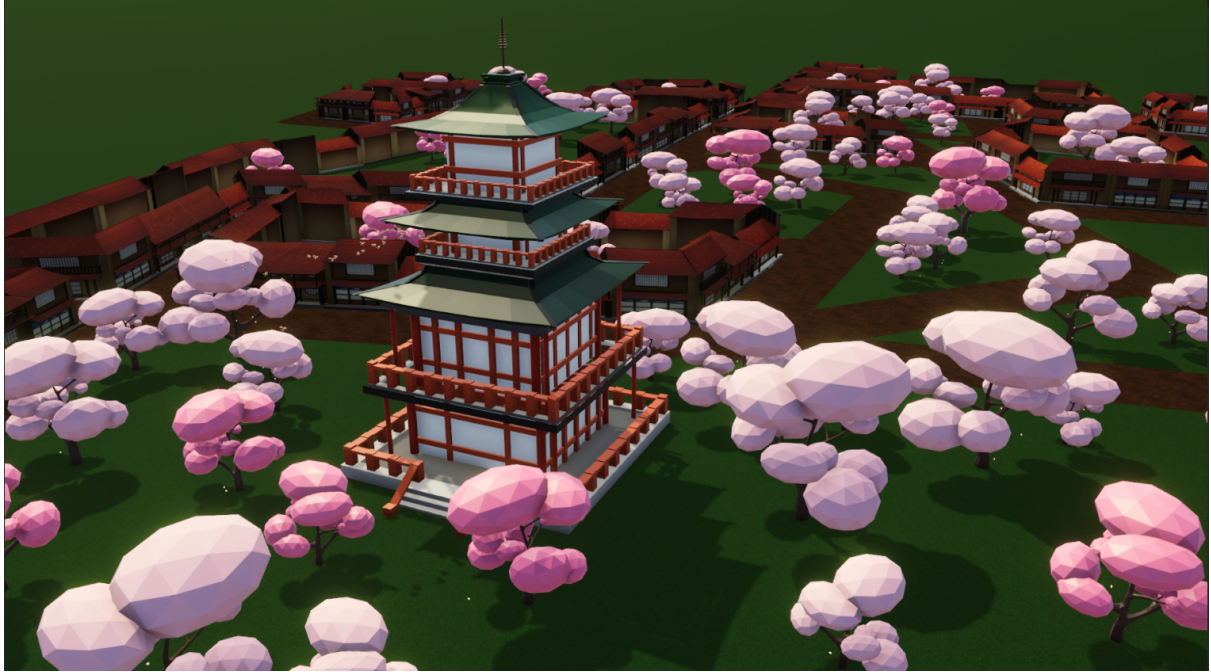


# Manual de Usuario

---



## Recursos a instalar

Lo único que deberás incluir en tu proyecto es un conjunto de tags y layers. Mostramos la configuración necesaria para estos se muestra en estas capturas:

| Tags  |               | Layers          |                |
|-------|---------------|-----------------|----------------|
| Tag 0 | Pared         | Builtin Layer 0 | Default        |
| Tag 1 | BuildingLot   | Builtin Layer 1 | TransparentFX  |
| Tag 2 | NPC           | Builtin Layer 2 | Ignore Raycast |
| Tag 3 | Pajaro        | User Layer 3    | Obstaculos     |
| Tag 4 | Depredador    | Builtin Layer 4 | Water          |
| Tag 5 | AgenteVirtual | Builtin Layer 5 | UI             |
|       |               | User Layer 6    | Player         |
|       |               | User Layer 7    | Aldeanos       |

El orden de los tags y layers dentro del proyecto puede ser relevante, así que se recomienda que tengan este mismo orden en vuestro proyecto.

## Interfaz de uso

Para utilizar el generador de ciudades **CityBuilder**, lo primero que tenemos que hacer es añadir el script **CityBuilder** (se encuentra en la carpeta **Scripts**) como componente en un GameObject cualquiera de nuestra escena. Tras esto, tenemos que modificar todos sus parámetros a nuestro gusto. Para ello, vamos a explicar cada uno de ellos:

### Topología de la ciudad

El primer parámetro que nos encontramos es **Punto Origen**, al cual tenemos que asignarle el transform de un GameObject (ej. el GameObject que tiene esta componente). La posición de este transform será el punto a partir del cual se empezará a crear nuestra ciudad. El siguiente es **Modo Completo**, que nos servirá para indicar si queremos modificar unos parámetros más avanzados sobre la topología de la ciudad (se comentarán más adelante).

A continuación, tenemos **Topology**, que especifica el tipo de ciudad que queremos crear (Raster, Branching, Radial\_Raster o Radial\_Branching). **Long\_PR** y **Long\_SR** establecen lo <sup>lungo</sup> largo que son las calles principales y secundarias (siempre menores que las principales), respectivamente. **Anchora Carretera** <sup>Larghezza</sup> marca la anchura de las calles de la ciudad. **Prob Branching PR** y **Prob Branching SR** sirven para establecer la probabilidad con la que se puede ramificar una carretera principal o secundaria, respectivamente. **Segment Limit PR** y **Segment Limit SR** sirven para limitar el crecimiento de cada tipo de calle. Y en **Road Material** establecemos el material que queremos que tengan nuestras calles.

Luego tenemos parámetros más avanzados para las ciudades. Dos de ellos son **Snap Length** y **Vertex Radio**, que sirven para indicar a una distancia máxima para saber si un extremo de una calle puede unirse con una intersección u otra calle cercana, y el segundo para marca una distancia máxima para saber si podemos unir una nueva intersección entre calles en alguno de sus extremos por cercanía. Otro de ellos es **Min Angle**, en el cual podemos escoger el ángulo mínimo entre dos calles conectadas a un mismo extremo que sería aceptable (no se aplica a todos los casos).

Además de todos estos, existen 4 parámetros concretos para topologías radiales:

- **Num Circles:** número de circunferencias para nuestra ciudad radial
- **Num Segs Circle:** con cuantos puntos queremos crear las circunferencias (nivel detalle de estas)
- **Circles Distance:** distancia entre circunferencias
- **Num PR Branches:** calles principales que salen desde el centro de la ciudad hasta las afueras de la misma (limitado por **Num Segs Circle**)

## Edificación de la ciudad

En esta sección, hemos simplificado la parametrización a 4 valores muy sencillos:

- **House Length:** largo de los edificios
- **House Depth:** profundidad de los edificios
- **House Height:** altura de los edificios
- **House Model:** lista de todos los prefabs que se colocarán como edificios. La probabilidad de colocar cada prefab es equiprobable. Se recomienda encarecidamente que la escala de todos estos sea 1 en todas sus dimensiones para evitar tamaños incorrectos en la colocación.

## Decoración de la ciudad

En este apartado, se necesitan 2 listas y un valor de densidad:

- **Arbolado:** es una lista de elementos formados por 3 valores: el prefab que se va a utilizar como arbolado de la ciudad (**Modelo**), el radio que ocupa el prefab en escena (**Radio**) y la probabilidad de aparición de este modelo (**Probabilidad**). Si la suma de probabilidades entre todos los elementos de arbolado es distinta de 0, se ponen todas como equiprobables.
- **Plant Density:** densidad de arbolado en la ciudad (Baja, Media o Alta)
- **Edificios Característicos:** es una lista semejante a la de **Arbolado**, pero para indicar posibles edificios emblemáticos en nuestra ciudad.

## Seres Vivos de la ciudad

Aquí podemos encontrar todo lo referente a IA (aldeanos y pájaros) como el sistema de jugador por defecto que le hemos dejado para explorar sus ciudades. Con **AI** se indica si quiere inteligencia artificial en la ciudad o no. **Population Density** controla el tamaño de la población de la ciudad. **Birds** sirve para saber si queremos generar pájaros además de aldeanos en la ciudad. Por último, **Default Player Sys** nos permite establecer en el centro de la ciudad que hayamos creado un jugador en primera persona de 2 tipos: uno cuyo movimiento es similar al de un humano, y otro como si fuera una cámara libre que puede volar por toda la ciudad.

## Botones de creación

Dentro del inspector, podemos encontrar una serie de botones al final de nuestra componente para crear directamente la ciudad con **Generar Ciudad**, o ir paso a paso con el resto de botones. Si vamos paso a paso, podemos volver a un paso anterior con el botón **Paso Anterior**; y resetear la creación de ciudades con **Resetear CityBuilder**. Cuando hayamos creado por completo una ciudad, podemos generar el sistema de jugador por defecto con el botón **Colocar Default Player System**.

## Otros aspectos

Cabe destacar que este creador de ciudades funciona tanto en modo edición como en modo ejecución. Las ciudades creadas se pueden guardar sin problemas como un prefab en nuestro proyecto al tener todo lo que necesita dentro de ella (excepto el **Default Player System**). Sin embargo, se recomienda guardar las ciudades sin IA, ya que las posiciones a las que se puede dirigir la IA son fijas desde que se crean.

### DEFAULT PLAYER SYSTEM

Para controlar al jugador del **Default Player System**, los controles son los siguientes:

- **W-S/Flecha arriba-Flecha abajo:** para moverse hacia delante y hacia atrás

- **C:** habilitar o deshabilitar la cámara libre. Esta se controla igual que el personaje en primera persona, exceptuando que esta puede volar y atravesar toda la escena.
- **E:** esconder cámara de seguimiento de la IA de la ciudad (si hay)
- **P:** cambiar entre la visión de pájaros (si hay) y entre visión de aldeanos
- **Z y X:** para cambiar la visión de IA entre los distintos aldeanos de la escena

El ratón sirve para rotar la cámara de nuestro personaje.