



**Escuela Superior
de Ingeniería y Tecnología**
Universidad de La Laguna

Trabajo de Fin de Grado

Generación procedimental de la ciudad de San Cristóbal de La Laguna en el Siglo XVI

*Procedural generation of the city of San Cristóbal de La
Laguna in the 16th century*

Miriam Rodríguez Méndez

La Laguna, 4 de julio de 2020

D. **Fernando Andrés Pérez Nava**, con N.I.F. 42.091.420-V profesor Titular de Universidad adscrito al Departamento de Ciencia de la Computación e Inteligencia Artificial de la Universidad de La Laguna, como tutor

D. **Alejandro Pérez Nava**, con N.I.F. 43.821.179-S profesor Asociado de Universidad adscrito al Departamento de Ciencia de la Computación e Inteligencia Artificial de la Universidad de La Laguna, como cotutor

C E R T I F I C A (N)

Que la presente memoria titulada:

"Generación procedimental de la ciudad de San Cristóbal de La Laguna en el Siglo XV"

ha sido realizada bajo su dirección por D. **Miriam Rodríguez Méndez**, con N.I.F. 78.642.528-P.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 4 de julio de 2020

Agradecimientos

A mis tutores, por guiarme a lo largo de todo el desarrollo y estar pendientes de cada paso.

A mi pareja, por apoyarme en todo momento y estar a mi lado cuando se presentaban dificultades.

A mi familia y amigos, por acompañarme a lo largo de estos meses y aguantar conmigo.

Licencia



© Esta obra está bajo una licencia de Creative Commons Reconocimiento 4.0 Internacional.

Resumen

La ciudad de San Cristóbal de la Laguna fue considerada Patrimonio de la Humanidad por la Unesco en el año 1999, esta privilegiada distinción le fue otorgada gracias a la buena conservación de su casco histórico a lo largo de los años. Ha sido una ciudad modelo para muchas otras ciudades de hispanoamérica, como se constata en el primer plano cartográfico conocido, elaborado a finales del siglo XVI por el arquitecto italiano Leonardo Torriani.

Transmitir la historia y todo el conocimiento que guarda la ciudad mediante medios convencionales no es trabajo sencillo. Hoy en día con las facilidades que proporciona internet hay muchos usuarios que prefieren ver un vídeo a leer un artículo. Por esto se está convirtiendo en algo necesario adaptar los contenidos históricos a las tecnologías audiovisuales.

Con la visión artificial y el procesamiento de imágenes es posible procesar estos planos después de haber sido digitalizados para generar modelos 3D de la ciudad. En este trabajo se abordará la reconstrucción automatizada en 3D de la ciudad de la Laguna en el siglo XVI. El objetivo será la generación automática de edificaciones que sigan el patrón particular de la ciudad. Dicha generación estará limitada a las manzanas que aparecen en el plano que constituyen aproximadamente el 95 por ciento de las edificaciones.

Este modelo 3D podrá ser de aplicación para productos audiovisuales y de realidad virtual y en particular será utilizado en el proyecto de la fundación Cajacanarias "Reconstrucción Histórica Virtual de San Cristóbal de La Laguna".

Palabras clave: manzanas, casas, modelado procedimental, modelado 3D

Abstract

The city of San Cristóbal de la Laguna was considered a World Heritage Site by Unesco in 1999, this privileged distinction was granted thanks to its good conservation over the years. It has been a model city for many other cities in Latin America, as can be seen in the first known cartographic map prepared at the end of the 16th century by the Italian architect Leonardo Torriani, in which its original layout is depicted.

Transmitting the history and all the knowledge that it keeps is not an easy job.

Nowadays with the facilities provided by the internet there are many users who prefer to watch a video than read an article. For this reason, it is becoming necessary to adapt historical content to audiovisual technologies.

With artificial vision and image processing it is possible to process these plans after they have been digitized to generate 3D models of the city. This work will address the automated 3D reconstruction of the city of La Laguna in the 16th century.

The objective will be the automatic generation of buildings that follow the particular pattern of the city. This generation will be limited to the blocks that appear in the plan that constitute approximately 95 percent of the buildings.

This 3D model may be applied to audiovisual and virtual reality products and in particular will be used in the Cajacanarias Foundation project "Virtual Historical Reconstruction of San Cristóbal de La Laguna".

Keywords: *blocks, houses, procedimental modeling, 3D modelling*

Índice general

Índice de Figuras	III
Índice de Tablas	IV
1. Introducción	1
1.1. Antecedentes	2
1.2. Motivación	2
1.3. Objetivos	3
2. Generación Procedural de Contenidos	5
2.1. Introducción	5
2.2. Modelado Procedimental	6
2.3. Diseño de la ciudad	7
2.4. Generación de Parcelas y Manzanas	7
2.5. Edificaciones y Fachadas	8
3. Herramientas y Tecnologías	9
3.1. Control de Versiones: Github	9
3.2. Python (v.3.7)	9
3.3. PyCharm Community Edition (v.2020.1)	10
3.4. QGIS (v.3.10.5)	10
3.5. Mapshaper	10
3.6. Blender (v.2.82.7)	10
3.7. Procesamiento de los datos	10
3.7.1. Archivos Shapefile	10
3.7.2. Archivos OBJ	11
4. Desarrollo	12
4.1. Estudio previo	12
4.2. Plano 2D	12
4.2.1. Lectura de los datos	12
4.2.2. Extracción de los datos	13
4.2.3. Preparación de los datos	17
4.3. Plano 3D	22
4.3.1. Añadir profundidad a los muros	22
4.3.2. Ajustar altura del plano al terreno real	23
4.3.3. Generación de archivos OBJ	23
4.3.4. Modelo final 3D	24

5. Conclusiones y líneas futuras	32
5.1. Conclusiones	32
5.2. Líneas Futuras	32
6. Summary and Conclusions	34
7. Presupuesto	35
7.1. Justificación del presupuesto	35
Bibliografía	37

Índice de Figuras

1.1. Primer plano cartográfico de San Cristóbal de La Laguna realizado por Leonardo Torriani [1]	1
1.2. Reconstrucción virtual de la ciudad ibérica de Ullastret.	3
2.1. Primera variedad y segunda variedad de subdivisión de parcelas.	7
4.1. Ejemplo de muros y vacíos.	13
4.2. Plano de las manzanas de San Cristóbal de La Laguna, extraído de QGIS. . .	15
4.3. Planos de los muros y los vacíos, extraídos de QGIS.	17
4.4. Distancias acumuladas hasta cada punto del Muro.	17
4.5. Combinaciones de muros en manzanas.	18
4.6. Manzana con muro con la combinación 1-0-1.	19
4.7. Tipos de segmentos en una manzana.	20
4.8. Generación de las esquinas en las manzanas.	21
4.9. Plano 2D con la base de las casas.	22
4.10 Plano 2D aumentado con la base de las casas.	23
4.11 Ejemplo de manzana con casas mal generadas.	24
4.12 Ejemplo fichero OBJ.	25
4.13 Modelo final 3D visto desde arriba.	26
4.14 Modelo final 3D vista perfil derecha.	27
4.15 Modelo final 3D vista perfil izquierda.	27
4.16 Modelo final 3D aumentado.	28
4.17 Modelo final 3D con más detalle.	28
4.18 Modelo final 3D con terreno real.	29
4.19 Ejemplo de manzana con todas las casas bien generadas.	30
4.20 Ejemplo de manzanas con muros.	30
4.21 Ejemplo de manzana con vacío.	31

Índice de Tablas

4.1. Datos que guarda cada manzana.	14
4.2. Datos que guarda cada muro/vacío.	16
4.3. Datos y porcentajes de las casas en el modelo.	21
7.1. Resumen de presupuesto	36

Capítulo 1

Introducción

El conocimiento de la historia es enriquecedor pues nos permite entender cómo somos en la actualidad, qué elementos han modelado nuestro entorno y qué errores no debemos repetir ya que conocemos sus consecuencias.

Conseguir transmitir este conocimiento es complicado cuando se sabe que el 80 por ciento de los usuarios que consumen internet prefieren un vídeo a una lectura de texto. Por esto, se está intentando adaptar los contenidos históricos a las tecnologías audiovisuales. Los avances actuales en este campo permiten recrear y reproducir contenidos y conocimientos históricos.

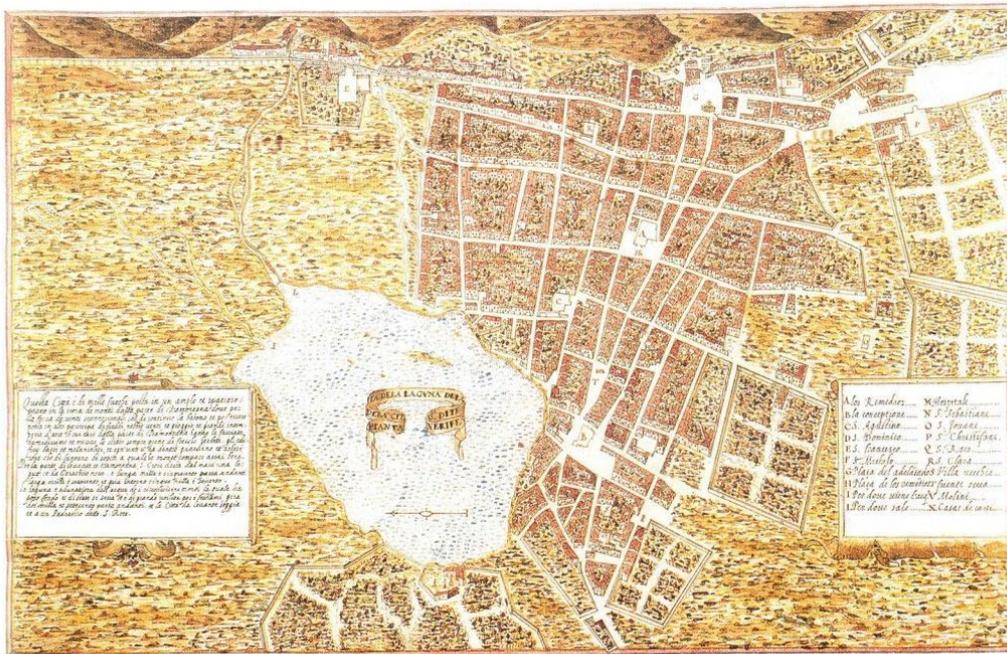


Figura 1.1: Primer plano cartográfico de San Cristóbal de La Laguna realizado por Leonardo Torriani [1]

Con este Trabajo de Fin de Grado se pretende recrear y construir en 3D el primer plano cartográfico existente de la ciudad de San Cristóbal de La Laguna , el cual se puede ver en la figura 1.1 (cedido por turismo de La Laguna [1]), realizado por el arquitecto Leonardo Torriani en el año 1588 cumpliendo órdenes de Felipe II para mejorar la defensa de las islas Canarias.

1.1. Antecedentes

Existen varias técnicas de creación de modelos 3D en el patrimonio y muchas de ellas se han utilizado durante décadas, sin embargo, han sido utilizadas de forma puntual debido a su elevado coste y su complejidad técnica. En los últimos años, gracias a los avances en hardware y software, se han desarrollado nuevos escáneres de utilización mucho más sencilla y precio razonable. Además de estos escáneres, han surgido nuevos software de creación de modelos 3D a partir de fotogrametría.

Existen dos sistemas de escaneado, los que utilizan escáneres láser y los que utilizan escáneres de patrones de luz blanca. Estos primeros emiten un haz láser que impacta en múltiples puntos del objeto a escanear, el escáner mide la distancia exacta a cada punto y a partir de este cálculo reconstruye la forma tridimensional del objeto. El sistema de escaneado de luz blanca estructurada, proyecta con luz blanca diferentes patrones (líneas verticales, horizontales, diagonales) sobre el objeto a escanear y una cámara lee las deformaciones de estas líneas sobre la superficie, a partir de esta información calcula su forma tridimensional.

El uso de fotogrametría ha servido a lo largo de décadas para crear relieves, este método requería de personal especializado y de complejos visores estereoscópicos. Gracias a la aparición de software que permite la creación de fotogrametría simplemente a partir de series de fotos y con un ordenador convencional, este recurso ha pasado a ser utilizado en sectores muy diversos, uno de los cuales es el patrimonio. Sigue una metodología muy sencilla, se toman series de fotos desde diferentes puntos de vista del objeto a modelar. El programa parte de puntos coincidentes de las fotografías para calcular desde que lugar exacto se han tomado y a partir de aquí recrear una nube de puntos, que nos lleva a la forma del objeto.

Un tercer método de construcción de modelos 3D es el modelado directo en programas de creación 3D como 3DS Max o Blender. Permite modelar objetos de los cuales se tiene información pero que han desaparecido, o sobre los que se quieren realizar modificaciones. Este es el caso de este trabajo, el cual se enfoca en modelar un plano cuyas estructuras en el mismo ya no existen, pero se cuenta con la información de las mismas.

Se han realizado varios modelos que se han llevado a la realidad virtual, como puede ser la Catedral de Colonia [2], cuenta con un modelado 3D de su interior y se puede recorrer con las gafas de RV.

En España se cuenta con la reconstrucción virtual de la ciudad ibérica de Ullastret [3], ver figura 1.2, gracias a la información obtenida en las últimas excavaciones, se realizó una reconstrucción completa de la ciudad y su entorno que sirvió de base para la creación de una experiencia de realidad virtual.

1.2. Motivación

Como se ha comentado anteriormente, la actualización y el estar al día con las nuevas tecnologías se está convirtiendo en algo esencial hoy en día. Para poder darle la importancia que tiene al legado histórico de las islas, y concretamente al de la ciudad de San Cristóbal de La Laguna es prioritario trasladar toda esta información y datos a medios



Figura 1.2: Reconstrucción virtual de la ciudad ibérica de Ullastret.

más atractivos y actuales. Es lo que se pretende conseguir con este Trabajo de Fin de Grado, convertir la historia en algo atractivo y enseñar a través de tecnologías 3D.

La recreación 3D de una ciudad a partir de un plano donde aparecen cientos de edificaciones requiere de personal cualificado lo que incrementa significativamente los costes. Es por ello que desde hace tiempo se trabaja en automatizar este proceso. Así se han utilizado desde hace tiempo técnicas de procesamiento de imágenes para crear texturas naturales, simular efectos especiales y generar modelos naturales complejos como árboles, ríos y cascadas. Estas técnicas de procesamiento pueden también aprovecharse para generar algunas geometrías y texturas en un entorno urbano.

Sin embargo, un área urbana es en su mayor parte un modelo que no sigue los patrones de la naturaleza, y que requiere la creación de edificios situados a lo largo de las calles y organizado en patrones propios.

En este trabajo se abordará la reconstrucción automatizada en 3D de La Laguna, en este caso los patrones de la red vial ya vienen dados y se generarán automáticamente las edificaciones que sigan el patrón particular de la ciudad. Dicha generación estará limitada a las manzanas que aparecen en el plano que constituyen aproximadamente el 95 por ciento de las edificaciones.

1.3. Objetivos

El objetivo principal del proyecto consistirá en la creación de un modelo en 3D a partir del plano elaborado por Leonardo Torriani(1.1). Este objetivo se descompone en un conjunto de actividades que son las siguientes:

- Documentación histórica.

- Digitalización de las manzanas del plano.
- Elaboración de un generador aleatorio con los elementos de las edificaciones.
- Generación procedimental de las bases de las edificaciones.
- Generación del modelo 3D de la ciudad.
- Incorporación del modelo 3D de la ciudad sobre el modelo digital del terreno.

Capítulo 2

Generación Procedural de Contenidos

2.1. Introducción

Diseñar y modelar ciudades cuenta con una larga historia relacionada con diseño gráfico, con aplicaciones desde la arquitectura, urbanismo a los simuladores de vuelo. El uso de los gráficos por ordenador, el modelado interactivo, el renderizado, visualización y simulación de modelos complejos de ciudades 3D se ha convertido en un tema de interés general. El proceso del diseño urbanístico podría llevar desde meses hasta años en su desarrollo, incluso contratando a muchos diseñadores provenientes de diferentes especialidades. Así que cualquier esfuerzo para poder reducir el tiempo del ciclo de diseño o de mejorar la calidad del modelo es una inversión beneficiosa. El diseño y modelado de ciudades a gran escala sería beneficioso para numerosas aplicaciones de gran importancia incluyendo las siguientes.

- Entretenimiento. Videojuegos y cine.
- Planificación y simulación urbanística. Visualizar y predecir el efecto de los cambios en la población, economía y adoptar políticas urbanísticas en la ciudad o país.
- Generación y visualización de mapas.
- Respuesta de emergencia. Crear modelos para capacitar al personal de respuesta a emergencias en los diseños urbanos actuales.

Modelar ciudades y espacios urbanos en general es una tarea complicada para los gráficos por ordenador, la visión por computador y la visualización. Entender, describir y modelar la geometría (crear modelos geométricos 2D y 3D) y el comportamiento de ciudades (simular el desarrollo urbanístico con el paso del tiempo) es un reto significativo.

Desde un punto de vista geométrico, las ciudades y los espacios urbanos son una colección de estructuras hechas por el hombre dispuestas en edificios interconectados, situados a lo largo de parcelas, manzanas y calles distribuidas sobre el terreno. Los entornos urbanísticos densos son particularmente difíciles de modelar geoméricamente porque son muy complejos y se expanden mucho, abarcando desde unos pocos hasta cientos de kilómetros cuadrados. Tradicionalmente, el modelado 3D de ciudades ha sido una tarea más bien manual que consume grandes cantidades de recursos. Con el constante crecimiento de requisitos de cantidad y de calidad en contenido urbano, hay

una necesidad imperativa de encontrar una solución alternativa que permita un modelado geométrico semiautomático y más rápido.

2.2. Modelado Procedimental

El modelado procedimental (MP) ha sido un tema de investigación activo durante más de treinta años y se aplica a una gran variedad de áreas como pueden ser el modelado de texturas, plantas, terrenos, edificaciones, áreas urbanas, redes de carreteras, ríos o creación de arte. No existe una única definición de modelado procedimental, se mueve en una amplia variedad de técnicas de generación que pueden producir semi-automáticamente un tipo de contenido específico basado en un conjunto de parámetros.

El MP está relacionado con muchas áreas y muchos modelos procedimentales son esencialmente representaciones generativas de procesos inspirados en la naturaleza, como el desarrollo de plantas o procesos centrados en el hombre, tales como simulaciones urbanas.

Una de las principales ventajas del MP, y probablemente la razón principal de su gran atractivo, es su capacidad de generación de datos. Un simple conjunto de parámetros de entrada o unas pocas reglas de generación del modelo procedimental produce una amplia variedad de modelos. Otra propiedad esencial del MP es su compresión de datos, un modelo geométrico bastante complejo puede ser representado por un modelo procedimental compacto y un conjunto de parámetros, mientras que la geometría real se genera solo cuando es necesario.

El MP tiene el potencial de reducir drásticamente la cantidad de esfuerzo de modelado requerido para crear contenido digital. Además como sus métodos son normalmente estocásticos (con comportamientos no deterministas), el MP puede crear una variedad de resultados usando un único conjunto de parámetros.

A pesar de todas las ventajas ya mencionadas y su alta productividad, los métodos de MP actuales aún no ofrecen una alternativa factible al modelado manual. La razón principal es el poco grado de control de la mayoría de modelos procedimentales. Requieren que los usuarios manipulen reglas de MP complicadas y parámetros cuyos efectos en la salida rara vez son predichos.

El MP puede usar varios sistemas de producción, el Sistema -L [4] es de los más relevantes y utilizados. Los sistemas L son un subconjunto de las gramáticas formales desarrollados para describir el crecimiento de las células vegetales. Más adelante se adaptaron para describir plantas enteras y estructuras arborescentes. Otros sistemas utilizan las Gramáticas de Formas [5]. Estas son gramáticas diseñadas para representar la subdivisión del espacio. En lugar de modelar crecimiento, como los Sistemas L, las gramáticas de forma modelan los objetos mediante la descomposición en componentes cada vez más pequeños y detallados. Esto las hace ideales para representar la formación de un edificio, comenzando por un volumen inicial, y dividiéndolo en secciones, pisos, habitaciones, hasta llegar a los detalles más pequeños, como ladrillos o baldosas.

2.3. Diseño de la ciudad

Una ciudad procedimental es un modelo complejo y normalmente jerárquicamente estructurado. Su procedimiento de generación generalmente opera de arriba hacia abajo, posiblemente comenzando a un nivel muy alto al generar una división por zonas del terreno disponible en zonas de la ciudad, como pueden ser el centro de la ciudad y las afueras, y terminando al colocar parcelas de edificios individuales.

Una vez definidas las zonas de la ciudad a alto nivel es necesario generar una red de calles y carreteras para obtener un modelo urbano en 3D. En [6] puede encontrarse una solución basada en el uso de sistemas L para hacer crecer una red de carreteras, similar al crecimiento de las ramas de una planta. A partir de un segmento semilla, crecen segmentos de carretera adicionales. La principal dificultad radica en controlar el proceso de crecimiento para producir una red vial final deseada.

2.4. Generación de Parcelas y Manzanas

Tras la definición del viario urbano aparecen un conjunto de manzanas que deben ser subdivididas en parcelas. La meta deseada para un método automático de subdivisión incluye proporcionar una salida (que la parcela tenga acceso a la calle), produciendo solo formas de parcelas plausibles y admitiendo una variedad de estilos de subdivisión.

En el diseño urbano, una manzana consta generalmente de una de las dos variedades de parcelas tal y como se ve en la figura 2.1. La primera variedad tiene parcelas cuya parte frontal está a lo largo de una calle y la parte posterior está adyacente a otra parcela de la misma variedad. La segunda variedad incluye parcelas que también pueden ser adyacentes a las calles, pero que pueden incluir una serie de parcelas interiores separadas por pequeños caminos o callejones, en lugar de las calles del viario. Hay varios algoritmos de generación de parcelas [7] para ambos tipos.

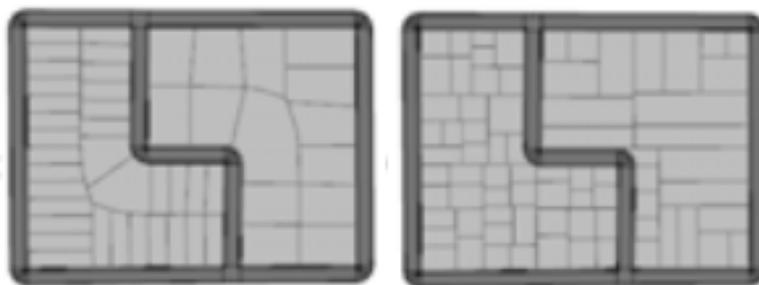


Figura 2.1: Primera variedad y segunda variedad de subdivisión de parcelas.

En el plano de Torriani se observa de forma mayoritaria la primera variedad. Una posible aproximación para la generación de las edificaciones en la laguna consistiría en generar el parcelario y posteriormente las edificaciones dentro del parcelario. Sin embargo, los algoritmos de parcelación asumen que las manzanas están cerradas y que todo el frente de la manzana se puede subdividir. Este no es el caso del plano de Torriani que tiene manzanas abiertas y zonas de las manzanas que no se pueden parcelar debido a que no lo estaban en aquel tiempo. Debido a esto se ha diseñado un algoritmo propio para resolver el problema.

2.5. Edificaciones y Fachadas

El ultimo paso en la generación procedimental de ciudades pasa por el levantamiento de los edificios y sus fachadas [8]. La mayoría de los métodos en esta categoría utilizan alguna forma de sistema de reescritura formal, como un Sistema-L, una gramática dividida o una gramática de formas, como base para generar un modelo de construcción 3D a partir de una forma de parcela 2D. Estos métodos pueden emplearse para crear edificios detallados y convincentes, pero requieren mucho esfuerzo pues la generación de la gramática adecuada para los edificios no es una tarea sencilla. Recientemente se han presentado algunos métodos alternativos intentan reconstruir automáticamente las gramáticas a partir de conjuntos de datos del mundo real, como las fotografías de fachadas de edificios.

Capítulo 3

Herramientas y Tecnologías

Antes de profundizar en las diferentes fases que tuvieron lugar durante desarrollo de este trabajo, se explicarán una a una las herramientas y tecnologías utilizadas.

3.1. Control de Versiones: Github

GitHub [9] es una plataforma de desarrollo colaborativo de software para alojar proyectos utilizando el sistema de control de versiones Git. El código se almacena de forma pública, aunque también se puede hacer de forma privada, creando una cuenta de pago.

Cuenta con funciones muy útiles a la hora de trabajar con código, permite crear diferentes versiones y guardarlas secuencialmente. Así se puede volver a una versión anterior siempre que se desee.

Gracias a esta herramienta se ha creado un repositorio donde se han almacenado todos los ficheros y los códigos generados a lo largo del desarrollo, repositorio [10]. De esta forma los códigos permanecen en la nube sin posibilidad de extravíos o eliminaciones accidentales.

3.2. Python (v.3.7)

Python [11] es un lenguaje de programación interpretado cuya filosofía hace hincapié en la legibilidad de su código. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional.

Este lenguaje de programación ha sido utilizado para la automatización del proceso de extracción de las coordenadas de las manzanas, así como de la creación de los polígonos de las casas en cada manzana.

Este ha sido el lenguaje elegido para la realización del trabajo por diversos motivos, primero para aprender y profundizar más en el propio lenguaje, y porque cuenta con infinidad de librerías muy útiles y dedicadas a programas específicos.

3.3. PyCharm Community Edition (v.2020.1)

PyCharm [12] es un entorno de desarrollo integrado de Python, en inglés *Integrated Development Environment (IDE)*. Permite entre otras cosas, descargar e instalar librerías, autoindexado, sincronizar con el control de versiones Git.

Se ha utilizado la versión Community que es gratuita, cuenta con una versión Profesional más completa y de pago. Con la versión gratuita se ha podido completar el trabajo con soltura y facilidad.

3.4. QGIS (v.3.10.5)

QGIS [13] es un Sistema de Información Geográfica (SIG) de Código Abierto licenciado bajo GNU - General Public License . QGIS es un proyecto oficial de *Open Source Geospatial Foundation (OSGeo)*. Soporta numerosos formatos y funcionalidades de datos vector, datos ráster y bases de datos.

Ha sido empleado para visualizar archivos ESRI Shapefile (SHP) del Casco Histórico de la ciudad de San Cristóbal de La Laguna. Concretamente los archivos referentes a las manzanas, y los muros y los vacíos pertenecientes a estas. Gracias a poder visualizar dichos archivos se ha podido asegurar en todo momento que la generación automatizada se estaba realizando correctamente.

3.5. Mapshaper

Mapshaper [14] es una herramienta online para la simplificación de archivos shape. Lee y escribe formatos Shapefile, GeoJSON y TopoJSON.

Es muy fácil e intuitiva y se ha usado principalmente para la visualización de los archivos Shapefile(.shp) en línea.

3.6. Blender (v.2.82.7)

Blender [15] es un programa informático multiplataforma, dedicado especialmente al modelado, iluminación, renderizado, animación y creación de gráficos tridimensionales. También de composición digital utilizando la técnica procesal de nodos, edición de vídeo, escultura y pintura digital.

3.7. Procesamiento de los datos

3.7.1. Archivos Shapefile

Shapefile [16] Este es un formato de archivo informático propietario de datos espaciales desarrollado por la compañía ESRI. Es un formato multiarchivo, es decir está generado por varios ficheros informáticos. El número mínimo requerido es de tres y tienen las extensiones siguientes:

.shp - es el archivo que almacena las entidades geométricas de los objetos.
.shx - es el archivo que almacena el índice de las entidades geométricas.
.dbf - es la base de datos donde se almacena la información de los atributos de los objetos.

3.7.2. Archivos OBJ

Los archivos *OBJ* [17] son un formato de archivo usado para objetos tridimensionales que contienen las coordenadas 3D (líneas poligonales y puntos), mapas de texturas, y otra información de objetos. Contienen un formato de imagen 3D estándar que se puede exportar y abrir en varios programas de edición de imágenes.

Este es el formato de archivo que se va a utilizar para generar el modelo final 3D.

Capítulo 4

Desarrollo

4.1. Estudio previo

Antes de comenzar a programar o a esquematizar la estructura del proyecto fue necesario un pequeño estudio previo, el cual comenzó por una visita al ex-convento de Santo Domingo en San Cristóbal de La Laguna, el principal objetivo de esta visita era ver y estudiar la maqueta basada en el plano de Torriani que está situada en este ex-convento.

Después de haber estudiado la maqueta se pasó a desarrollar un primer enfoque para la realización del programa. Ya se había trabajado anteriormente con *Python* así que no resultó demasiado complicado comenzar a programar, el estudio conllevó la búsqueda de librerías específicas que pudieran ser de provecho y también elegir la mejor forma de formar las estructuras de datos para los elementos básicos del mapa: manzanas, muros y vacíos.

Se han seguido unos pasos muy bien definidos, esto ha hecho que la investigación y el estudio de las herramientas haya sido progresivo y ordenado.

4.2. Plano 2D

Una vez realizado el estudio previo y tras haber planificado cómo se van a estructurar los ficheros en el programa se pasa a reconstruir el plano 2D. Este plano va a constar de manzanas y de los muros y vacíos existentes en cada una de ellas, en el caso de que tengan tal y como se ven en la figura 4.1. Se han utilizado 3 archivos con formato *Shapefile*, uno para las manzanas, otro para los muros y el último para los vacíos.

Este proceso es el paso previo antes de generar el plano 3D, por lo tanto la principal finalidad de la construcción de este plano 2D es la de facilitar la lectura de datos para la generación de los ficheros finales.

El plano en cuestión es el mostrado en la figura 4.2, esta figura muestra las manzanas sin muros ni vacíos.

4.2.1. Lectura de los datos

Se comienza leyendo cada uno de los ficheros *Shapefile*, los cuáles contienen la información geométrica y de los atributos de cada objeto existente. Esto se hará tanto



Figura 4.1: Ejemplo de muros y vacíos.

para las manzanas como para los muros y vacíos, con esta lectura de los datos se guardará dicha información relevante para futuras modificaciones y operaciones. Estos archivos *Shapefile* se visualizan previamente con *QGIS* o con *Mapshaper* para comprobar que los datos son correctos.

4.2.2. Extracción de los datos

Una vez leídos los datos se procede a la extracción de los mismos, de forma ordenada, empezando por las manzanas y terminando por los muros y los vacíos. La finalidad de esta extracción es la de guardar todos los datos en una estructura de datos ordenada y de uso sencillo.

Manzanas

En el caso de las manzanas es necesario tener acceso a las coordenadas de cada una, al igual que a las distancias acumuladas y las distancias punto a punto ya que toda esta información será utilizada más adelante. A su vez, cada manzana contará con su conjunto único de casas y fachadas, de las cuales hablaremos más adelante. En la tabla 4.1 se muestran los datos dentro de cada manzana.

Profundizando un poco más en 'tipo', como ya se explica en la tabla hace referencia al tipo de segmento, puede ser el comienzo de una casa (CC), el Comienzo de un Muro (CM), el Comienzo de un Vacío (CV), el Final de un Muro (FM) o el Final de un Vacío (FV). Esto será de mucha ayuda más adelante cuando se proceda a la construcción de fachadas y casas en las manzanas, ya que donde haya algún segmento que sea CM o CV no será posible construir o generar fachadas ni casas.

Identificador	Número que identifica a cada manzana, es único.
Coordenadas x	Todas las coordenadas 'x' que conforman cada uno de los vértices de la manzana.
Coordenadas y	Todas las coordenadas 'y' que conforman cada uno de los vértices de la manzana.
Distancias vértice a vértice	La distancia que existe entre un vértice y el inmediatamente posterior.
Distancias total acumuladas	Las distancias que existen entre el vértice de inicio y cada uno de los vértices.
Tipos de segmentos	Los tipos de segmentos que conforman la manzana. [CM, FM, CV, FV, CC]
Vectores (x, y)	Componentes 'x' e 'y' de los vectores que conforman la manzana.
Fachadas	Conjunto de fachadas construidas en la manzana.
Casas	Conjunto de casas construidas en la manzana.

Tabla 4.1: Datos que guarda cada manzana.



Figura 4.2: Plano de las manzanas de San Cristóbal de La Laguna, extraído de QGIS.

Muros y Vacíos

En cuanto a los muros y vacíos se seguirá el mismo proceso y se guardará la misma información para ambos. En la tabla 4.2 se explican brevemente los datos más relevantes guardados.

Cada muro/vacío está formado por tres puntos, dos de estos puntos representados por un 1 hacen referencia al inicio y al final del muro/vacío, y el tercer punto representado por un 0 hace referencia a por donde pasa dicho muro/vacío. En las figuras 4.3a y 4.3b se pueden apreciar estos tres puntos que delimitan las estructuras.

Es necesario destacar que la determinación del inicio y final del muro es ambigua ya que depende del sentido con el que se recorra la manzana. Aunque se podría fijar el sentido se prefiere facilitar el trabajo del operador que digitalize el plano y se tiene una marca indistinta (el valor 1) para inicio y final. Se le obliga, no obstante a colocar un punto con valor 0 dentro del muro para evitar la ambigüedad.

A la hora de guardar la información obtenida de los *Shapefiles*, esta información llega desordenada, es decir, estos tres puntos pertenecientes a cada muro/vacío no están agrupados, así que para comprobar a qué manzana pertenece cada muro/vacío se ha de comprobar cada manzana, y dentro de cada manzana comprobar segmento a segmento para ver a qué segmento pertenece cada punto obtenido del *Shapefile*, cabe mencionar

Identificador	Número que identifica la manzana a la que pertenece dicho muro/vacío.
Coordenadas x	Todas las coordenadas 'x' que conforman cada uno de los vértices del muro/vacío.
Coordenadas y	Todas las coordenadas 'y' que conforman cada uno de los vértices del muro/vacío.
Segmentos	Los segmentos de la manzana en los que se encuentra el muro/vacío.
Orden de los puntos	Se guardan los '1' y el '0' en el orden en el que se encuentran en la manzana.
Distancias acumuladas	Las distancias que hay desde el vértice de inicio de la manzana hasta cada punto del muro/vacío.

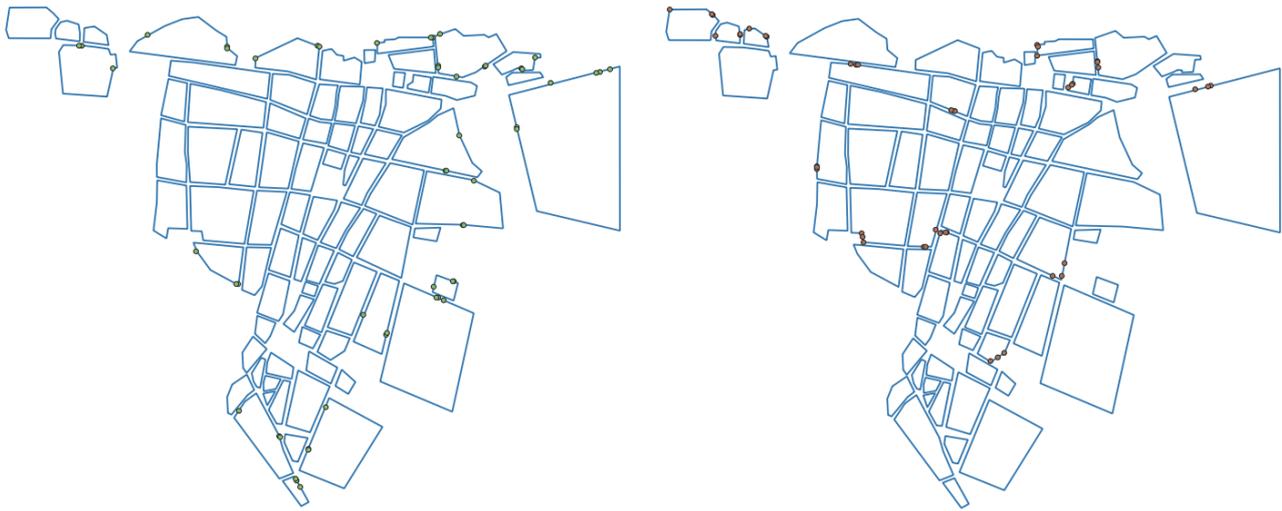
Tabla 4.2: Datos que guarda cada muro/vacío.

internamente que el recorrido de la manzana se realiza en sentido horario desde el vértice de partida.

Para poder organizar los puntos y guardar ordenadamente las coordenadas y los segmentos a los que pertenecen, se guardan las distancias, tanto la distancia acumulada desde el origen de la manzana hasta cada punto, como la distancia del punto al vértice. Para obtener estos tres puntos en orden, se aplica una técnica muy sencilla que consiste en comparar las distancias acumuladas, esta distancia es la que existe entre el vértice origen de la manzana y llega hasta el punto del muro/vacío en cuestión como se puede observar en la figura 4.4. Esta distancia se ordena de menor a mayor y de esta forma se obtienen los puntos ordenados.

Una vez se tengan ordenados los puntos en relación a las distancias es necesario conocer donde empieza y donde acaba el muro/vacío. Existen 3 combinaciones posibles de muros/vacíos, dependiendo de dónde se encuentre su inicio, medio y fin. Se ha de controlar cada una de estas combinaciones para una correcta implementación, y se debe tener en cuenta que los muros/vacíos siempre siguen la secuencia 1-0-1 para establecer el sentido.

- **Combinación 1-1-0.** En esta combinación lo primero que se encuentra es el final del muro/vacío (1), a continuación el inicio (1) y por último el medio (0). Se podría pensar que al encontrar el primer 1 ya se consideraría el inicio, pero se debe tener en cuenta el siguiente número que en este caso es otro 1, esto significa que el segundo 1 va a ser el inicio. En la figura 4.5b se muestra un ejemplo con este tipo de combinación.
- **Combinación 0-1-1.** En esta combinación lo primero que se encuentra es el punto medio (0), por lo tanto el siguiente punto tendrá que ser el final del muro/vacío y el



(a) Plano muros San Cristóbal de La Laguna. (b) Plano vacíos San Cristóbal de La Laguna.

Figura 4.3: Planos de los muros y los vacíos, extraídos de QGIS.

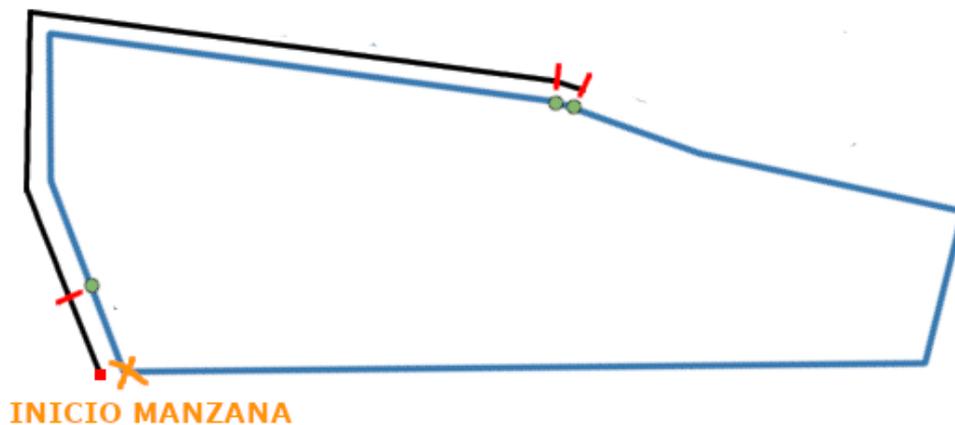


Figura 4.4: Distancias acumuladas hasta cada punto del Muro.

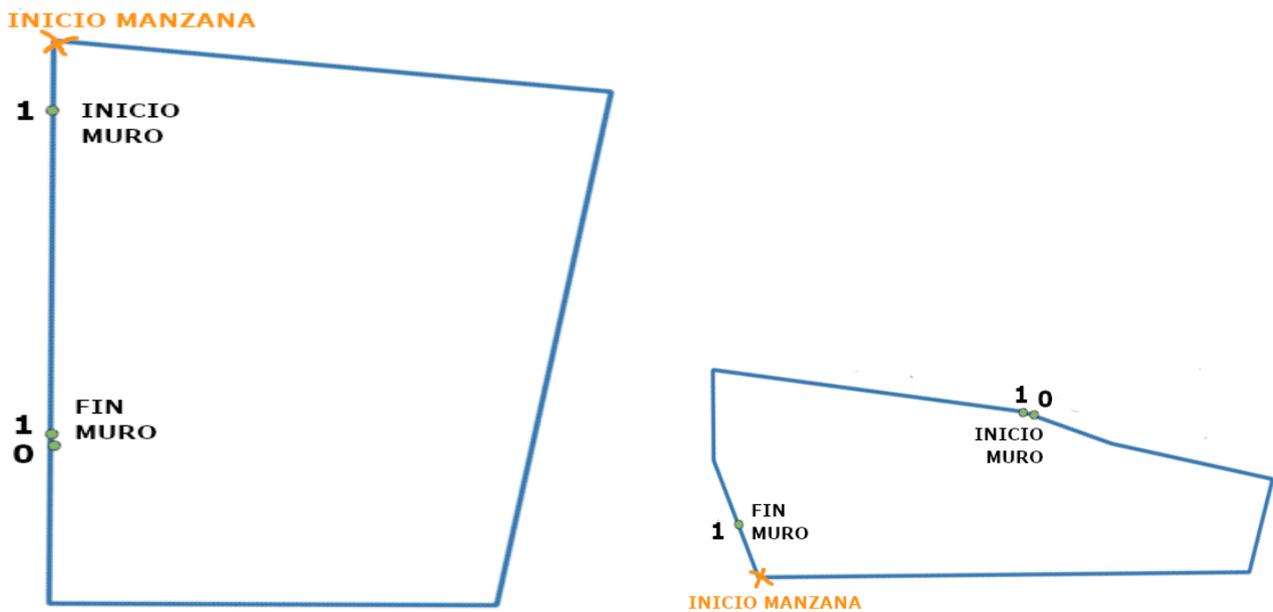
último punto será el inicio. En la figura 4.5a se muestra un ejemplo con este tipo de combinación.

- **Combinación 1-0-1.** Esta última combinación es la más sencilla ya que se encuentran los puntos de forma ordenada, se empieza por el inicio, a continuación el medio y para finalizar el final del muro/vacío. En la figura 4.6 se muestra un ejemplo con este tipo de combinación.

Todas estas combinaciones posibles existen porque los segmentos de la manzana empiezan a recorrerse siempre por el mismo punto, así que dichas combinaciones sirven de referencia a la hora de identificar correctamente la distribución de los muros/vacíos.

4.2.3. Preparación de los datos

Después de haber establecido los datos referentes a las manzanas y a los muros y vacíos, se procede a la reordenación de los segmentos en el caso de las manzanas con



(a) Manzana con muro con la combinación 0-1-1. (b) Manzana con muro, combinación 1-1-0.

Figura 4.5: Combinaciones de muros en manzanas.

muros o vacíos y a la edificación de fachadas y casas dentro de cada manzana.

Muros y Vacíos

Para una correcta construcción del modelo es necesario tener la información lo más clara posible, para esto se va a especificar de qué tipo es cada segmento de cada manzana. En apartados anteriores se explicó los tipos de segmentos con los que se trabajan y en este apartado se van a asignar estos tipos con los segmentos.

En los casos en los que el punto del muro/vacío no coincida con un vértice de la manzana, como en la figura 4.6, se añade dicho punto a las coordenadas de la manzana y se incrementa el número de vértices y segmentos en la misma. Si por el contrario el punto del muro/vacío coincide o se encuentra a una distancia muy pequeña de algún vértice de la manzana se cuenta el vértice ya existente como punto del muro/vacío, por lo tanto el tipo de segmento sería modificado de 'CC' a 'CM' ó 'CV'.

En la figura 4.7 se muestran los tipos de segmentos en una manzana que contiene un muro. Esta manzana cuenta con 6 segmentos y sus correspondientes 6 vértices, cada segmento empieza en el vértice marcado. El muro abarca desde el primer 'CM' hasta el 'FM' y todos los segmentos que se encuentran en medio han de marcarse como 'CM' para dejar constancia de que siguen perteneciendo al muro y para que a la hora de construir las casas no sea posible mientras sean de tipo 'CM' o 'CV'.

Además de modificar los tipos en las manzanas con muros o vacíos, también será necesario añadir las coordenadas de dichos muros o vacíos a las propias coordenadas de las manzanas, así a la hora de construir las casas se tiene toda la información en el mismo sitio y con acceder a los datos de la manzana es suficiente.

El resto de manzanas que no contengan ni muros ni vacíos no sufrirán ninguna modificación y todos sus segmentos serán del tipo 'CC'.

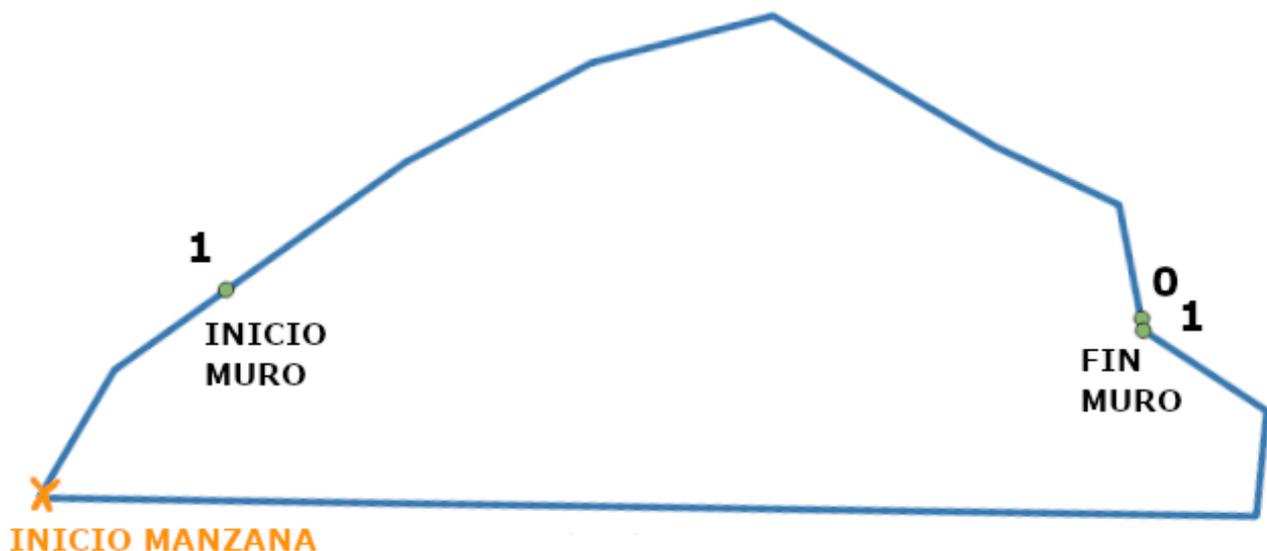


Figura 4.6: Manzana con muro con la combinación 1-0-1.

Construcción de Fachadas

Una vez se tengan las manzanas preparadas y correctamente modificadas en el caso de contener algún muro o vacío, se pasa a la primera fase de construcción en la que se generarán las fachadas. Este paso intermedio antes de crear las casas se añadió para que la implementación fuera más sencilla y desglosada.

A la hora de generar las fachadas, se ha optado por rellenar segmento a segmento de cada manzana, siempre teniendo en cuenta que el tipo de segmento sea 'FM', 'FV' o 'CC', los segmentos que sean 'CM' o 'CV' también serán una fachada pero dicha fachada servirá para más adelante darle profundidad a los muros y contar con ellos en el modelo 3D. Las fachadas contarán con una longitud mínima y una longitud máxima, así se asegura que no se generen fachadas muy pequeñas y poco realistas. La longitud de cada fachada se ha calculado con una distribución Normal [18] estimada a partir de las casas del plano, para obtener longitudes en un rango controlado y de forma aleatoria, una vez se obtiene la longitud se comprueba que realmente es mayor que la longitud mínima y menor que la longitud máxima.

Con este proceso existe la posibilidad de que queden espacios sin fachadas dentro de los segmentos, esto ocurrirá porque el espacio que quede es demasiado pequeño, en algunos casos se arregla sumando las longitudes de la fachada pequeña con la inmediatamente anterior siempre y cuando dicha suma no supere la longitud máxima, en el caso de que la supere se dejaría el hueco de esta fachada y contaría como casa no generada.

Construcción de Casas

En el proceso de construcción de casas, se comienza construyendo las casas que hacen esquina en las manzanas. Para esto se van a unir las fachadas que se encuentran antes y después de cada esquina, como se muestra en la figura 4.8.

Primero se recorren las fachadas anteriormente generadas en cada manzana y se comprueba nuevamente que la longitud se encuentra entre el mínimo y el máximo

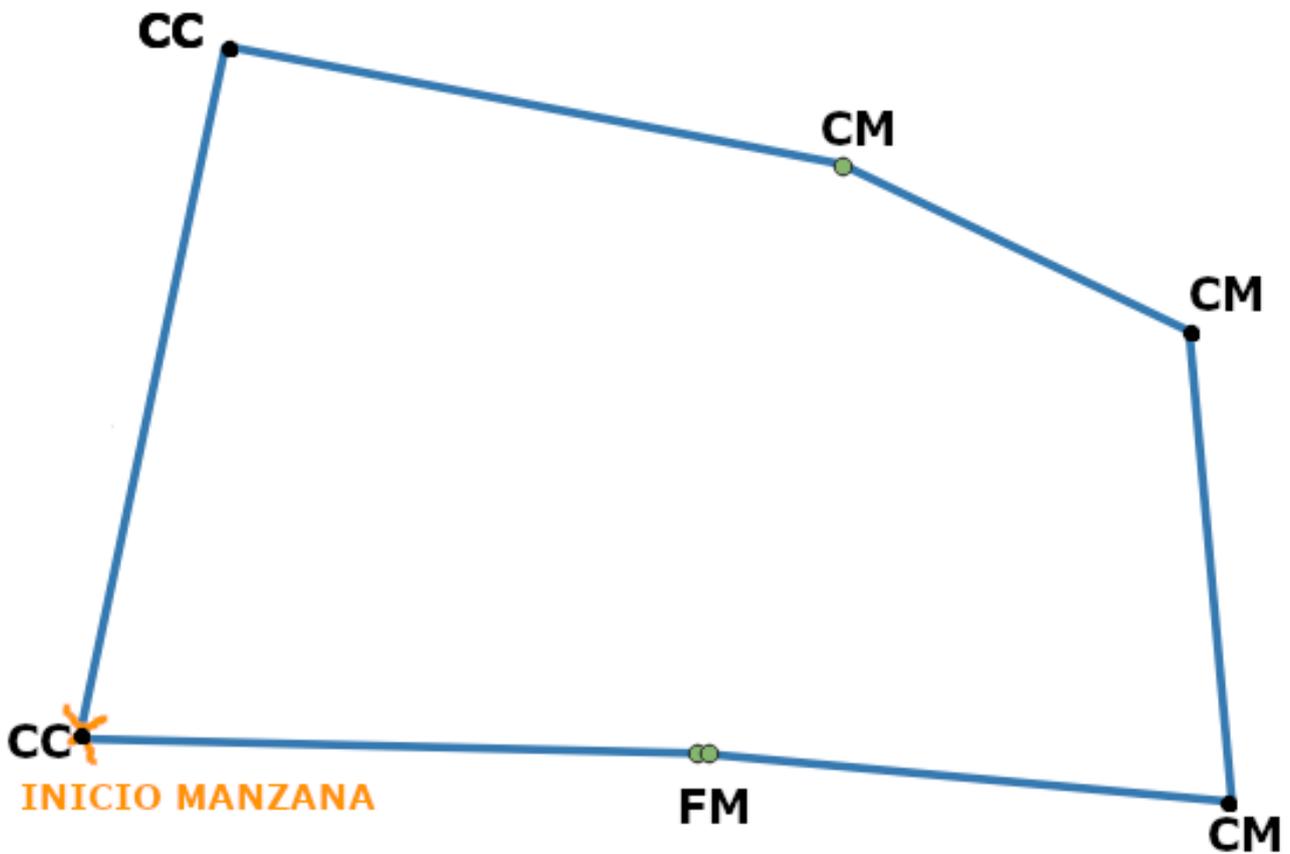


Figura 4.7: Tipos de segmentos en una manzana.

establecido. Para generar las casas de las esquinas se comprueba que la fachadas se encuentren en segmentos diferentes además de estar en un segmento de tipo 'FM', 'FV' o 'CC' y de no pertenecer ya a una esquina, en caso de pertenecer a una esquina se construye una casa normal. Una vez se realicen las comprobaciones, se suman las longitudes de ambas fachadas y se generan los puntos restantes para completar la base de la casa.

Para generar las casas que no pertenecen a una esquina se comprueba la longitud de la fachada, que no pertenezca a una esquina y que se encuentre en un segmento del tipo 'FM', 'FV' o 'CC', es entonces cuando se generan los puntos restantes para completar la base de la casa.

En ambos casos se aplica la altura correspondiente al igual que con la longitud con una distribución Normal.

A continuación se muestra el plano 2D finalizado en las figuras 4.9 y 4.10, donde se puede apreciar esos pequeños huecos en donde no caben fachadas, también las zonas donde hay vacíos y muros.

Se ha realizado un conteo del número y porcentaje de casas que se han generado correctamente, las que no se han generado y las que se han generado mal, se puede observar en la tabla 4.3. De esta manera se tiene un control más exhaustivo de la precisión en el modelo 2D generado.

Se considera que una casa ha sido bien generada cuando cumpla con los requisitos de tamaño en cuanto a longitud, profundidad y altura, y cuando no intersekte con ninguna

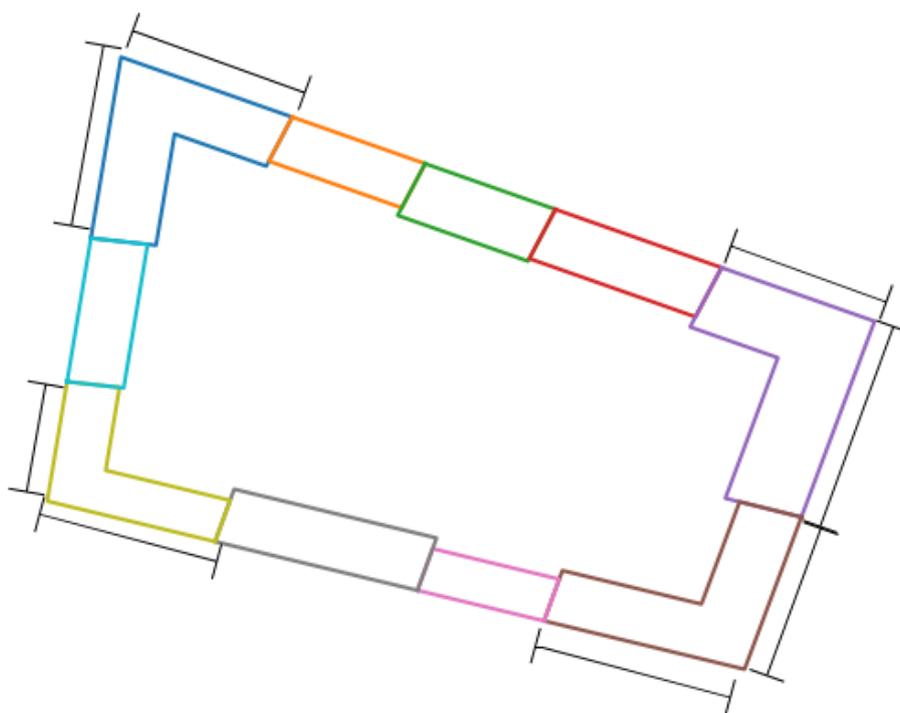


Figura 4.8: Generación de las esquinas en las manzanas.

Número total de casas	828
Número total de casas bien generadas	726 (87,7 %)
Número total de casas mal generadas	72 (8,7 %)
Número total de casas no generadas	30 (3,6 %)

Tabla 4.3: Datos y porcentajes de las casas en el modelo.

otra casa. Por otro lado, una casa mal generada (ver figuras 4.11a y 4.11b) interseca con alguna otra casa o no cumple los requisitos de tamaño. Para comprobar que no existe intersección, cada casa cuenta con un polígono formado por cada uno de los vértices de la misma. Se comprueba con los métodos propios de los polígonos, en este caso con el método *intersects*. Por último, una casa es considerada no generada cuando no exista en el plano, es decir, es demasiado pequeña para cumplir los requisitos de tamaño, se trata de tamaños tan pequeños que son difíciles de localizar en el plano a simple vista.

El número de casas no generadas es tan elevado porque están incluidas las casas de tres manzanas que no se han podido incluir finalmente en el modelo, de esto se hablará más detalladamente en el capítulo de líneas futuras, ya que en este modelo no se contemplan las manzanas que tengan más de un muro o vacío. Las casas de estas tres manzanas se han calculado y se han añadido como casas no generadas, en total son 23 casas las pertenecientes a estas manzanas lo que convierte el número de casas no generadas a 7 en el caso de que no se contara con estas tres manzanas.

Las casas no generadas y las casas mal generadas son las que tendrán que modelar los diseñadores desde cero, teniendo en cuenta que se ha conseguido un 87,7 por ciento de casas bien generadas, el trabajo que tienen que realizar estos diseñadores se ha conseguido reducir considerablemente.



Figura 4.9: Plano 2D con la base de las casas.

4.3. Plano 3D

El paso final de este proyecto es el de generar el modelo en un plano 3D, pero para poder hacer esto primero se deben realizar un par de modificaciones o ajustes para conseguir que el modelo final sea lo más fiel posible a la realidad.

4.3.1. Añadir profundidad a los muros

El primer ajuste a realizar es darles profundidad a los muros, esto se consigue haciendo lo mismo que se hizo anteriormente con las fachadas y las casas, creando un polígono en la base de cada uno de los muros. Este paso pretende hacer visibles los muros en el modelo 3D final para que aparezcan representados.

En el caso de los vacíos, como bien indica su nombre no son una estructura física, así que no es necesario ningún ajuste ni modificación para que se encuentren representados en el modelo final.

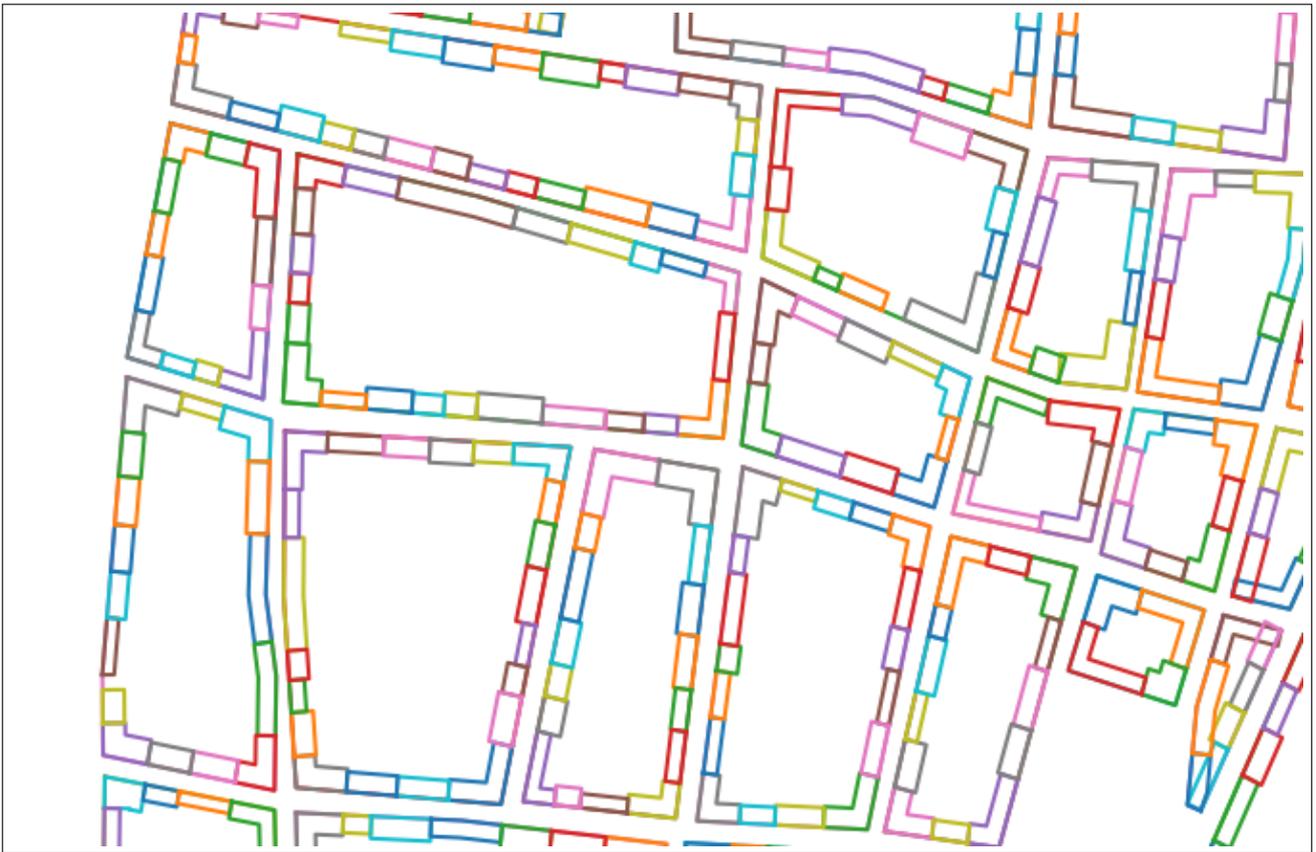


Figura 4.10: Plano 2D aumentado con la base de las casas.

4.3.2. Ajustar altura del plano al terreno real

Hasta ahora el modelo generado es un modelo plano y todavía en 2D, pero la realidad es que es muy difícil que un terreno real sea completamente plano. Por esto hace falta ajustar la base de todo el modelo para que sea igual a la realidad, ya que en el terreno real existen algunas pequeñas montañas, valles, los cuales convierten el terreno en un terreno irregular y por lo tanto, no plano.

Para realizar esta modificación, se ha utilizado un fichero OBJ ya explicado anteriormente, dicho fichero contiene todas las coordenadas en altura de la base de cada polígono perteneciente al modelo, esto incluye a las casas y a los muros.

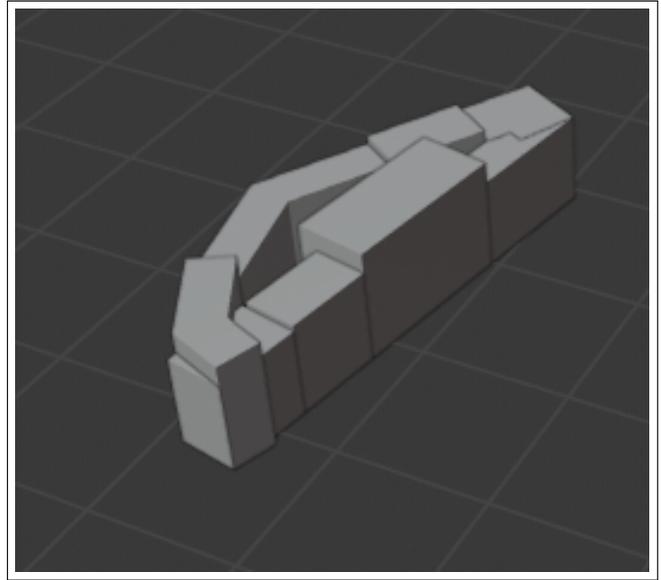
4.3.3. Generación de archivos OBJ

Se trabajará con un OBJ para las casas y con otro para los muros, teniendo así separadas las coordenadas de ambas estructuras. Para obtener ambos OBJ se extraen del modelo 2D todas las coordenadas que se han obtenido en el proceso, y además de esto será necesario aplicar una altura a las casas y a los muros.

En un fichero OBJ se separan los datos entre tipo de polígono, vértices de dicho polígono y las caras que forman el polígono. Como se muestra en la figura 4.12 la primera línea es el tipo de polígono, en este caso un Cubo, el segundo recuadro recoge las coordenadas de los vértices, un vértice por línea, y el último recuadro representa cada una de las caras del polígono, los números representan qué vértice compone cada cara.



(a) Casa mal generada



(b) Casa mal generada perfil.

Figura 4.11: Ejemplo de manzana con casas mal generadas.

Para generar la altura de la edificación se realizará el mismo proceso que con la longitud y la profundidad, utilizando una distribución Normal para generar aleatoriamente esta altura, contando con que tiene un mínimo y máximo establecido. Una vez se tenga la altura, simplemente se suma a las coordenadas de cada vértice de la base de las casas y se obtiene finalmente un polígono. El modelo final será un conjunto de polígonos, dicho conjunto estará formado por las casas y los muros.

4.3.4. Modelo final 3D

Cuando se hayan generado los archivos OBJ se importan a Blender, tanto el archivo de las casas como el de los muros. Una vez importados se aplicarán las texturas del plano real que se encuentran en un archivo .MTL, este archivo contiene información de texto sobre patrones de textura. En la imagen 4.13 se puede apreciar el modelo final, desde esta vista se comprueba como cada manzana está posicionada correctamente en el terreno.

En las imágenes 4.14 y 4.14 se aprecian las alturas de cada estructura, tanto de las casas como de los muros, así como su profundidad.

Tanto en la figura 4.16 como en la figura 4.17 se puede observar aquellas manzanas que tienen vacíos y como el modelo no ha generado ninguna estructura en los mismos.

Para comprobar que realmente se ha ajustado al terreno real se adjunto la figura 4.18 donde se puede ver con un mayor detalle las bases de las estructuras a diferentes alturas.

En las figuras 4.19a y 4.19b se puede observar un ejemplo de manzana sin muros ni vacíos y con todas las casas correctamente generadas.

En las figuras 4.20a y 4.20b se observa unos ejemplos de manzanas con muro, y en las figuras 4.21a y 4.21b una manzana con vacío, todas generadas correctamente. En

```
o Cube0
v 2079.633589392226 -582.636111980795 0
v 2086.1026920180725 -550.856645331325 0
v 2126.1507521360777 -560.0218730200227 0
v 2121.8497030379795 -578.8155875573656 0
v 2101.0550422198344 -574.0566104547055 0
v 2098.525733202825 -586.4818410007642 0
v 2079.633589392226 -582.636111980795 26.894611174517454
v 2086.1026920180725 -550.856645331325 26.894611174517454
v 2126.1507521360777 -560.0218730200227 26.894611174517454
v 2121.8497030379795 -578.8155875573656 26.894611174517454
v 2101.0550422198344 -574.0566104547055 26.894611174517454
v 2098.525733202825 -586.4818410007642 26.894611174517454
f 1 2 3 4 5 6
f 7 8 9 10 11 12
f 1 2 8 7
f 2 3 9 8
f 3 4 10 9
f 4 5 11 10
f 5 6 12 11
f 6 1 7 12
```

Figura 4.12: Ejemplo fichero OBJ.

todos los ejemplos se adjunta una imagen con el plano real para que se compruebe que realmente en ese lugar de la manzana existe un muro o vacío.



Figura 4.13: Modelo final 3D visto desde arriba.



Figura 4.14: Modelo final 3D vista perfil derecha.



Figura 4.15: Modelo final 3D vista perfil izquierda.



Figura 4.16: Modelo final 3D aumentado.

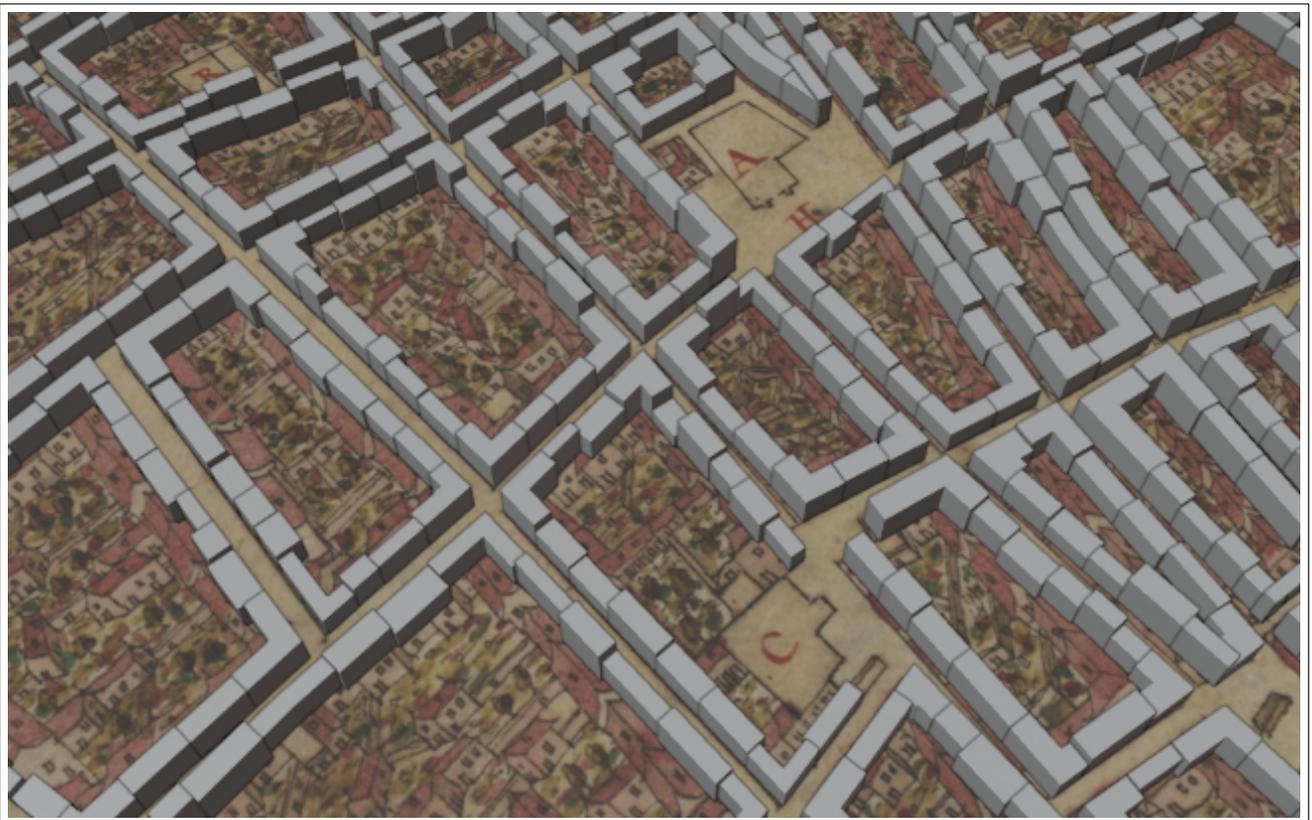


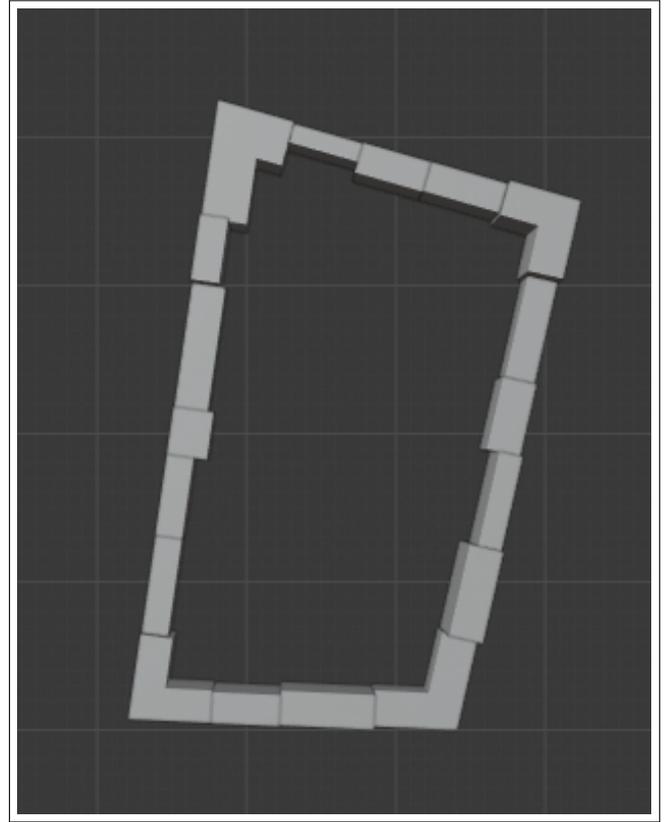
Figura 4.17: Modelo final 3D con más detalle.



Figura 4.18: Modelo final 3D con terreno real.

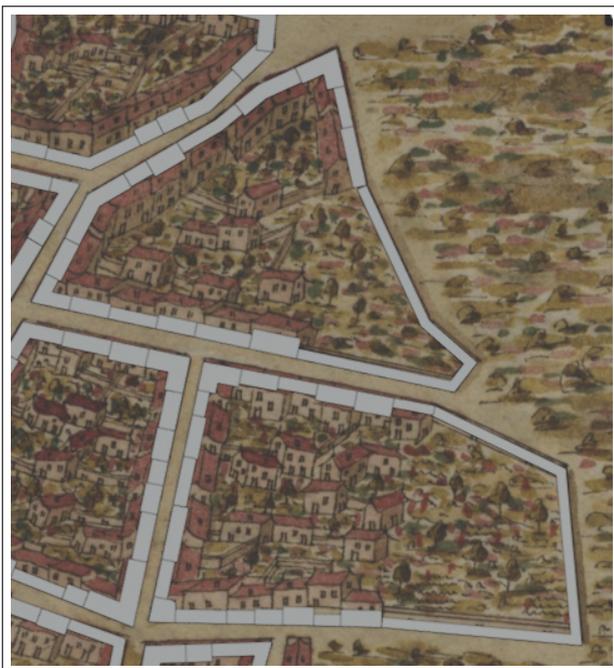


(a) Manzana con todas las casas, plano real.

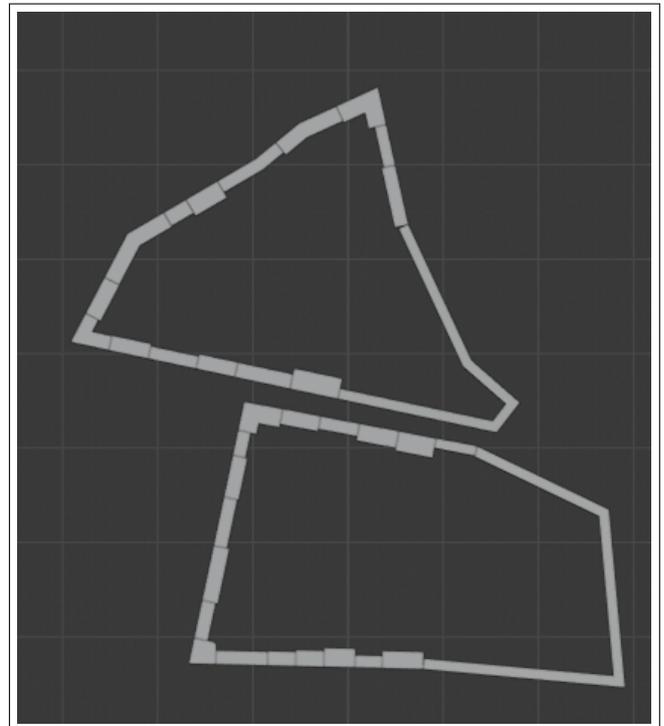


(b) Manzana con todas las casas.

Figura 4.19: Ejemplo de manzana con todas las casas bien generadas.

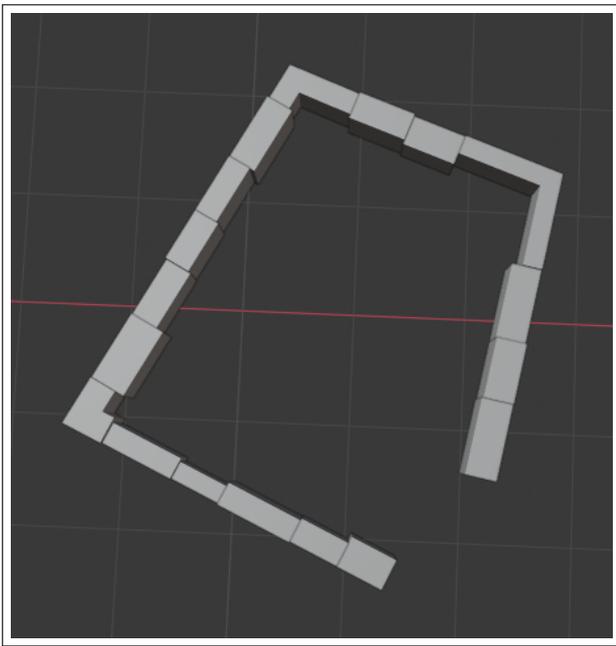


(a) Manzana con muro, plano real.

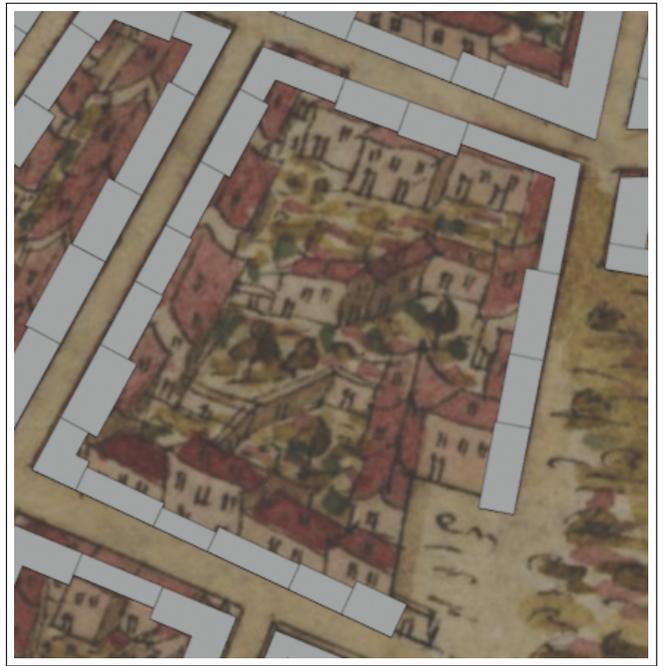


(b) Manzana con muro.

Figura 4.20: Ejemplo de manzanas con muros.



(a) Manzana con vacío, plano real.



(b) Manzana con vacío.

Figura 4.21: Ejemplo de manzana con vacío.

Capítulo 5

Conclusiones y líneas futuras

5.1. Conclusiones

El resultado final de este proyecto ha sido el esperado, con la obtención de un modelo 3D de las edificaciones del plano de Torriani de la ciudad de San Cristóbal de La Laguna.

En el proceso se ha pasado por diferentes etapas, primero un estudio previo y familiarizarse con las nuevas tecnologías a utilizar a lo largo del desarrollo, como han podido ser Blender, Python y los archivos Shapefile entre otras ya mencionadas. Después de haber extraído la información de los Shapefile, se ha procedido a la estructuración de la misma para que en su posterior utilización el acceso a los datos fuera más fácil e intuitivo, con la generación de un plano 2D como paso intermedio. Con los datos correctamente establecidos y la generación de los OBJ, se pasa a ejecutar el último paso generando el modelo 3D. Añadiendo texturas y terreno real a dicho plano 3D se ha concluido el desarrollo total del proyecto.

La generación procedimental de modelos como este no es tarea fácil y está ligada al trabajo de muchos diseñadores que invierten su tiempo en generar una a una las casas de cada manzana. Con este proyecto se pretende agilizar el trabajo de estos diseñadores, ya que con una sola ejecución, pasándole como parámetros los archivos Shapefile que se quieran generar, se obtiene el modelo 3D en forma de archivos OBJ que pueden importarse fácilmente a Blender.

Habiendo visto los resultados obtenidos, el porcentaje total de casas bien generadas es suficiente para reducir la carga de trabajo manual en el modelado. Por lo tanto, los objetivos propuestos han sido cumplidos en gran medida ya que todas esas casas que no se han podido generar o están mal generadas pueden ser rectificadas a mano sin problema.

5.2. Líneas Futuras

Como líneas futuras el programa puede ser mejorado y ampliado para conseguir resultados mucho más ambiciosos y realistas. Entre estas líneas futuras se encuentran:

- Incluir manzanas que tengan más de un muro o vacío. Como ya se ha comentado a lo largo del documento existen tres manzanas que se han excluido de este modelo 3D, porque actualmente el programa no genera modelos con manzanas que contengan

más de un muro o vacío, esto se incluye en las líneas futuras porque es posible implementarlo en el programa actual.

- Incluir tejados a las casas del modelo 3D. Este punto se planteó en algún momento del desarrollo pero por falta de tiempo no ha podido ejecutarse. Una vez obtenido el modelo 3D final, en Blender es posible añadir un script que incluya tejados en cada una de las casas generadas.
- Hacer que el tamaño de las casas sea dinámico dependiendo del área de las manzanas. Como se ha podido observar, hay algunas casas que se han generado mal y se han superpuesto unas con otras, la mayoría de casos ocurren porque el área de la manzana es muy pequeña y las casas tienen establecido un mínimo y máximo para longitud, altura y profundidad. Sin embargo, para estas manzanas tan pequeñas es necesario modificar estos parámetros para que el tamaño de las casas se adapte al de la manzana.
- Corregir el proceso de generación de fachadas para que no existan fachadas demasiado pequeñas. Esto requería de cambios mas sustanciales en el código. Conseguir que no se generen fachadas muy pequeñas jugando con los tamaños de las demás y adaptándolos para que al final todos los tamaños sean aptos.
- Aplicar este proceso al resto de mapas de Torriani de las islas como a los de Teguisse, Arrecife, Betancuria, Las Palmas, Telde, Santa Cruz de Tenerife, Garachico, Santa Cruz de la Palma o San Sebastián de la Gomera.

Capítulo 6

Summary and Conclusions

The end result of this project has been as expected, with obtaining a 3D model of the plane Torriani city of San Cristobal de La Laguna.

In the process, it has gone through different stages, first a preliminary study and getting familiar with the new technologies to be used throughout the development, such as Blender, Python and Shapefile files, among others already mentioned. After having extracted the information from the Shapefiles, we have proceeded to structure it so that its subsequent use makes accessing the data easier and more intuitive, with the generation of a 2D plan as an intermediate step. With the data correctly established and the generation of the OBJs, the last step is executed, generating the 3D model. Adding textures and real terrain to said 3D plane has completed the total development of the project.

The procedural generation of models like this is not an easy task and is linked to the work of many designers who invest their time in generating the houses on each block one by one. This project aims to streamline the work of these designers, since with a single execution, passing the Shapefile files to be generated as parameters, the 3D model is obtained in the form of OBJ files that can be easily imported into Blender.

Having seen the results obtained, the total percentage of well-generated houses is enough to reduce the manual workload in modeling. Therefore, the proposed objectives have been largely met since all those houses that could not be generated or are poorly generated can be rectified by hand without problem.

Capítulo 7

Presupuesto

La estimación del presupuesto para este proyecto se ha calculado, en parte, en función de una estimación del salario que podría percibir un ingeniero informático junior.

7.1. Justificación del presupuesto

Justificación del presupuesto mostrado en la tabla 7.1, en el que se muestra desglosado el tiempo empleado en el desarrollo el proyecto y su coste.

- El equipo informático utilizado tiene un coste de compra de 700 euros, a esto se le ha aplicado la amortización correspondiente a un año, lo que supone un 25 por ciento del precio total.
- El beneficio industrial supone un 15 por ciento del coste total, el coste total es de 5370,00€, añadiéndole este beneficio formaría un total de 6175,00€.
- Dado que en todo momento se ha utilizado software libre y gratuito para desarrollar el proyecto, este apartado no constituye un gasto real para el proyecto.
- El resto de presupuesto, es decir, el gasto derivado de la mano de obra se ha calculado en base a una estimación de lo que podría cobrar un ingeniero informático como desarrollador junior empleando su propio PC como herramienta principal de trabajo.

Con lo anterior, la cantidad total final asciende a 6607,00€ que es la suma de todos los costes del desarrollo realizado.

Concepto	Cantidad	Coste Unidad	Coste Total
Equipo informático	N/A	N/A	175,00€
Documentación	20h.	13,00€	260,00€
Análisis e investigación	30h.	17,00€	510,00€
Desarrollo	170h.	20,00€	3400,00€
Modelado y pruebas	30h.	20,00€	600,00€
Beneficio Industrial (15 %)	N/A	N/A	805,00€
Total			6175,00€
Total + IGIC (7 %)			6607,00€

Tabla 7.1: Resumen de presupuesto

Bibliografía

- [1] **Turismo La Laguna** - fuente.
<https://www.turismodelalaguna.com/la-mirada-cartografica-en-la-evolucion-de-la-ciudad-de-torriani-a-los-planificadores-contemporaneos-mapas-y-planos/>.
- [2] **Cologne Cathedral St. Maternus Chapel, en línea.** - fuente.
<https://www.youtube.com/watch?v=pIpCeE4PZB0>, consultaen06/03/2018.
- [3] **Ullastret 3D, caminar por una ciudad ibérica de hace 2.000 años, en línea.** - fuente.
<http://patrimoni.gencat.cat/es/historias/ullastret-3d-caminar-por-una-ciudad-ib> consultaen06/03/2018.
- [4] **Sistema-L** - fuente.
<https://es.wikipedia.org/wiki/Sistema-L>.
- [5] **Gramática Formal** - fuente.
https://es.wikipedia.org/wiki/Gram%C3%A1tica_formal.
- [6] **Procedural modeling of cities** - fuente.
Y. I. H. Parish and P. Müller, "Procedural modeling of cities,"in Proceedings of the Annual Conference on Computer Graphics and Interactive Techniques (ACM SIGGRAPH '01), pp. 301–308, August 2001.
- [7] **Procedural Generation of Parcels in Urban Modeling** - fuente.
Vanegas, Carlos & Kelly, Tom & Weber, Basil & Halatsch, Jan & Aliaga, Daniel & Müller, Pascal. (2012). Procedural Generation of Parcels in Urban Modeling. Computer Graphics Forum. 31. 681-690. 10.1111/j.1467-8659.2012.03047.x.
- [8] **Procedural modeling of buildings** - fuente.
P. Müller, P. Wonka, S. Haegler, A. Ulmer, L. Van Gool: Procedural modeling of buildings, SIGGRAPH 2006.
- [9] **Control de Versiones: GitHub** - fuente.
<https://github.com/>.
- [10] **Repositorio TFG** - fuente.
<https://github.com/alu0100785050/TFG-miriam/tree/master>.
- [11] **Python** - fuente.
<https://www.python.org/>.
- [12] **PyCharm** - fuente.
<https://www.jetbrains.com/es-es/pycharm/>.

- [13] **QGIS** - fuente.
<https://www.qgis.org/es/site/>.
- [14] **Mapshaper** - fuente.
<https://mapshaper.org/>.
- [15] **Blender** - fuente.
<https://www.blender.org/>.
- [16] **Archivos shapefile** - fuente.
<https://es.wikipedia.org/wiki/Shapefile>.
- [17] **Archivos OBJ** - fuente.
<https://www.reviversoft.com/es/file-extensions/obj>.
- [18] **Distribución Normal** - fuente.
https://es.wikipedia.org/wiki/Distribuci%C3%B3n_normal.