

# Generación automática de reglas procedurales a partir de objetos 3D aplicado al modelado de entornos urbanos (GENOME)

Juan Felipe Calvache Clavijo<sup>1</sup>

[Juan.calvache00@usc.edu.co](mailto:Juan.calvache00@usc.edu.co)

Universidad Santiago de Cali, Facultad de Ingeniería, Programa de Ingeniería de Sistemas (1)

## **Resumen**

El modelado 3D es una técnica de virtualización aplicada en diferentes campos como la arquitectura, el diseño, los videojuegos y entornos digitales, las características de las diferentes técnicas está dada por su complejidad o el consumo de tiempo, por lo tanto, el proyecto tiene como objetivo estudiar la técnica conocida como modelado procedimental, para la creación de una herramienta y un flujo de trabajo, que permita disminuir el tiempo en la creación de estructuras en entornos urbanos. El flujo de trabajo creado en el proyecto comprende tres fases, la primera tiene como objetivo la captura de información, se integran datos de sistemas de información geográfica (GIS) y análisis de imágenes fotográfica, la segunda toma los datos de la fase uno donde los procesas mediante el software, generando reglas procedurales que describen cómo modelar el edificio capturado. La tercera fase consiste en implementar las reglas generadas en la Fase 2 en el software **CityEngine**, esta herramienta utiliza las reglas procedurales para extruir axiomas 2D a modelos 3D completos, este proyecto fue desarrollado con el fin de crear una herramienta que permitiera la construcción de estas estructuras para que de esta forma se pudiera agilizar el proceso de construcción de entornos digitales.

*Palabras Clave:* Modelado procedimental, City Engine, Open Street Map, entornos digitales, Sistemas de información geográfica

## **Abstract**

3D modeling is a virtualization technique applied in different fields such as architecture, design, video games and digital environments, the characteristics of the different techniques are given by their complexity or time consumption, therefore, the project aims to study the technique known as procedural modeling, for the creation of a tool and a workflow, that allows to reduce the time in the creation of structures in urban environments. The workflow created in the project comprises three phases, the first aims to capture information, data from geographic information systems (GIS) and photographic image analysis are integrated, the second takes the data from phase one where it is processed by software, generating procedural rules that describe how to model the captured building. The third phase consists of implementing the rules generated in Phase 2 in the CityEngine software. This tool uses procedural rules to extrude 2D axioms into complete 3D models. This project was developed in order to create a tool that would allow the construction of these structures so that the process of building digital environments could be streamlined.

## 1. INTRODUCCIÓN

### 1.1 Marco teórico

El impacto de nuevas tecnologías ha producido varios cambios en las metodologías de planificación y diseño urbano, permitiendo abordar el desarrollo del crecimiento urbano orientado al futuro, los ingenieros buscan mejorar los diseños de las ciudades, para gestionar la complejidad de la urbanización (Badwi, Ellaithy, & Youssef, 2022). La generación de objetos tridimensionales es muy útil en campos como la simulación, la animación y la arqueología. La creación de objetos 3D presentan una complejidad en su desarrollo debido al nivel de detalle de los objetos y hay que tener en cuenta que al trabajar con las herramientas que tienen un enfoque de aplicación manual, el camino demanda un esfuerzo significativo (Díaz, 2022).

Una forma de mejorar la producción de los objetos, es adoptar tecnicas que permitan la creación de una forma rápida y ágil. En este estudio se adopta una técnica de modelado procedimental conocida como gramáticas de

división que permite la descomposición de una estructura, generando una huella de la fachada que permite identificar la zona de interés para aplicarle objetos o texturas (Badwi, Ellaithy, & Youssef, 2022).

La creación de entornos urbanos, ayuda a gestionar la planificación de manera inteligente, en la china antigua planificaban de manera amigable con la naturaleza, identificando zonas estratégicas que les permitiera adquirir recursos de una manera fácil (Qin, Mao, Hu, Zheng, & Xu, 2023). En la investigación se creó un flujo de trabajo, que tiene como objetivo ser una guía en la creación de entornos digitales, estos entornos permiten visualizar, analizar y simular con precisión los entornos urbanos (Yadav, Colucci, Boccardo, & Zlatanova, 2024). La herramienta desarrollada tiene el propósito de apoyar el proceso de generación de entornos, mejorando los tiempos en la creación de las reglas de manera manual.

Se utilizan datos GIS tomados de Open Street Map, estos son datos cartográficos recopilados desde cero por voluntarios, ellos realizan estudios sistemáticos sobre el terreno, generando datos que brindan características físicas del terreno como las carreteras o edificios por medio de etiquetas (Fitri, Marena, Hisam, Hafiz, & Izzati, 2022). Estos datos son utilizados en CityEngine para la representación del entorno por medio de una huella conocida como lotes, las reglas CGA pueden ser utilizadas en las huellas dando como resultado la creación de diferentes tipos de estructuras.

El desarrollo será llevado en python que es un lenguaje de programación que trabaja muy bien con machine learning y el manejo de datos (Rayhan & Gross, 2023). El software está compuesto por tres vistas que capturan y usan la información brindada por el usuario. Una vista cuenta con un algoritmo de inteligencia artificial conocido como YOLOV5, este permite entrenar el modelo haciendo uso de un conjunto de datos, el cual puede extraer características que componen las imágenes (Shi, Wang, & Guo, 2022), también se ha estudiado técnicas como gaussian splatting que es utilizado para la renderización de imágenes 2D a objetos en 3D (BERNHARD, Georgios, Thomas, & George, 2023) se pueden obtener objetos a partir de esta técnica y utilizarlos en CE.

Para desarrollar el software se aplicó la metodología PSP apoyada por la guía PMBok. La guía define la disciplina necesaria para satisfacer las necesidades de las partes interesadas (Dang, Sardroud, & Shirzadi, 2024). Se utilizaron las plantillas para establecer los planes de alcance, tiempo y calidad, al aplicar esto se tiene como objetivo generar una herramienta de calidad soportada por buenas prácticas.

## 1.2 Antecedentes

El modelado procedimental es la creación automatizada de contenido mediante algoritmos (Kopowski, Mrela, Mikołajewski, & Rojek, 2024). Este concepto ha sido desarrollado sobre la base de algunos sistemas de producción como lo es el L-system, las gramáticas de formas y las gramáticas de división, estas técnicas permiten la creación de estructuras a partir de un conjunto de parámetros de entrada.

El L-system es un sistema de reescritura basado en reglas que tienen como objetivo el modelado geométrico de plantas, fue creado por el biólogo Lindenmayer y fue adoptado en la comunidad científica para el desarrollo de nuevos enfoques, partiendo de esta técnica emergen otras como la gramática de formas, que permite generar y controlar la forma, es utilizada para la construcción y análisis de diseño (Wang, Song, & Tang, 2020), o la gramática de divisiones que fue introducida para agregar atributos como parámetros a las formas geométricas.

Muller amplió el concepto proponiendo el método de arquitectura generada por computadora (CGA) para la

generación de objetos 3D con un alto detalle, estos objetos son en particular objetos urbanos que permiten el modelado de entornos digitales, CGA es una metodología que ha potenciado el concepto del modelado procedimental y el diseño urbano debido a el conjunto de gramaticas de formas, como extrusión, taslación, escalado y división (Zhang, Wu, Liu, Zhang, & Li, 2022) .

En trabajos relacionados se estudiaron las diferentes aplicaciones, como el estudio de **Procedural modeling applied to the 3D city model of Bogota** donde establecieron un flujo de trabajo para el desarrollo de modelos urbanos en corto tiempo. El objetivo del estudio fue generar un modelo tridimensional aproximado del area estudiada, para que les permitiera analizar el impacto de una nueva autopista en la zona. Su flujo de trabajo utilizó datos SIG para generar la huella y la creación de una regla para la extrusión de 2000 modelos. El resultado fue una ciudad tridimensional que les permitió realizar el estudio de la nueva carretera (Alomía, Loaiza, Zúñiga, Luo, & Asoreycacheda, 2021).

En el estudio **GIS Based Procedural Modeling in 3D Urban Design** propusieron reglas CGA para proporcionar un flujo de trabajo completo de diseño urbano, el proyecto fue desarrollado en CityEngine, comienzan utilizando datos SIG que les permiten realizar una división de parcelas aplicando el modelado procedimental, esto permite delimitar los espacios de cada estructura, extruyen los edificios y los mapean con texturas. El resultado fue un flujo de trabajo interactivo completo para el diseño urbano 3D semiautomatizado a partir de la representación 2D de el área urbana tiempo real (Zhang, Wu, Liu, Zhang, & Li, 2022).

**Procedural modeling and layout method for a generic ancient Chinese city** es un un estudio que propone un metodo de diseño de ciudad basada en L-System, el cual es enfocado en las ciudades chinas antiguas, ellos mejoran el metodo brindando restricciones locales, agregando un factor simétrico y una restricción de distancia, para guiar la generación del diseño, de esta manera mejoraron el rendimiento del L-system, sus resultados experimentales mostraron que la estructura generada por L-System de la ciudad antigua china es más cercana al diseño real de una ciudad antigua china debido a la simétrica lograda (Qin, Mao, Hu, Zheng, & Xu, 2023).

### 1.3 Estado del arte

El modelado procedural ha permitido a proyectos como **Procedural 3D fountaing modeling** utilizar el modelado procedimental junto con inteligencia artificial para la creación de fuentes en 3D, utilizando un método de aplicación para crear modelos de fuentes que utiliza información de simetría de objetos, estos reciben parámetros de fuentes similares y parámetros como el tipo de texturas, dimensiones de la fuente, forma de la fuente y una imagen, se comienza a realizar las similitudes y a ejecutar las reglas, esta obtiene como resultado un modelo rápido de la fuente (Kurt & Bülbül, 2021).

**Geovisualization\_of\_Buildings\_AI\_vs\_Procedural\_Mod** propusieron un flujo de trabajo enfocado en el modelado procedural, el uso de SIG y el uso de IA, su objetivo, mejorar en gran medida la eficiencia y precisión de los procesos de diseño. Su aplicación comienza con la extracción de descripciones de edificios a partir de entradas textuales o visuales, el modelo se entrenan para comprender las relaciones espaciales y los estilos arquitectónicos y haciendo uso de modelado paramétrico les permite la creación de estructuras complejas (Nikçi, Župan, & Racetin, 2024).

La automatización en la generación de estructuras 3D se ha visto impulsada por el uso de recursos tecnológicos avanzados y diversos enfoques de modelado, como el modelado procedimental, los SIG y la inteligencia artificial. Estas herramientas permiten no solo la creación de representaciones precisas y detalladas de entornos urbanos, sino también optimizar el tiempo y los costos en comparación con métodos convencionales.

### 1.4 Justificación

El modelado procedimental es una herramienta para sintetizar automáticamente una gran cantidad de formas limpias (Ishtiaque, Takeo, & Oliver, 2023). El modelado procedimental de objetos en 3D establece un flujo de trabajo que al ejecutarse tiene como resultado la creación de diferentes estructuras, al analizar trabajos relacionados, se descubrió que existen diferentes técnicas que pueden ser aplicadas como lo son el L-System, modelado de gramática de formas, modelado de gramáticas divididas, estas técnicas han sido propuestos en diferentes investigaciones (Ming, Jieli, Yang, Ji, & Guanyao, 2022).

Combinar el modelado procedimental con SIG, permite el desarrollo de ciudades inteligentes como en la ciudad de Nanjin en china, utilizaron información geográfica para la planificación de la ciudad. Crearon un centro de datos con la información geoespacial y les permitió ser más efectivos en la toma de decisiones, además crearon un sistemas para aprovechar la información que generaban, esto les permite realizar monitoreos constantes de la ciudad y recibir alerta ante anomalías.

desde una perspectiva científica, este proyecto busca abordar una brecha existente en el campo del modelado procedimental mediante la integración de tecnologías avanzadas como la visión artificial y los sistemas de información geográfica. A diferencia de enfoques anteriores que dependen en gran medida de reglas predefinidas o la intervención manual, nuestra propuesta se enfoca en la automatización de la extracción de datos y utilizarlas para la generación del contenido, dentro de los datos esenciales se encuentran los SIG que permite representar las zonas urbanas y que también hoy en día aporta en la creación de ciudades inteligentes (Zhang D. , 2024).

El flujo de trabajo que se estableció, mejora la precisión de los modelos generados y también facilita la accesibilidad para usuarios no especializados. Se utilizan técnicas de IA, como redes neuronales que son utiles para la detección de objetos en imágenes, al ser entrenada puede funcionar perfectamente para mantener una representación digital de los aspectos del edificio (Sezen, Cakir, Atik, & Duran, 2022). Este enfoque interdisciplinario no solo contribuye al desarrollo de herramientas más eficientes para el modelado urbano, sino que también abre nuevas posibilidades para la planificación urbana, la simulación y el diseño de ciudades inteligentes.

## 1.5 Problema de investigación

Las representaciones virtuales en 3D han abierto un vasto campo de aplicaciones, pero también ha generado diferentes desafíos. Como toda tecnología emergente han surgido múltiples caminos para la obtención de resultados. Sin embargo la creación de estas formas no es tarea sencilla, requiere de intervención de artistas o personal calificado en diseño y modelado(Hossain, Shen, Igarashi, & Kaick, 2023).

En la aplicación de modelados de edificios y entornos urbanos, los metodos tradicionales implican procesos manuales o semiautomatizados basado en herramientas de diseño asistido por computadora (CAD) y software de sistemas de información geográfica (SIG). Estas herramientas permiten la creación de modelos tridimensionales a partir de planos en 2D pero la intervencion manual sigue siendo significativa, lo que incrementa considerablemente el tiempo y el esfuerzo requerido (Nikçi, Župan, & Racetin, 2024).

Estos retos también estan constituidos en la creación de las reglas que son utilizadas para representar el modelo de la estructura, los modelos son representaciones creadas a partir de lenguajes y reglas, esta aplicación es una manera efectiva de modelar, pero el aprendizaje de la sintaxis de los programas también presenta tiempo para una comprensión avanzada, por lo tanto hay una necesidad de explorar alternativas que optimicen el proceso de modelado, reduciendo la dependencia de la intervención humana.

## 1.6 Metas

Desarrollar un software para capturar información y generar reglas para la creación de estructuras 3D aplicando el método de modelado procedimental.

- Identificar los requerimientos esenciales y tecnologías para la generación automática de reglas procedurales en el modelado urbano 3D, mediante el análisis de las necesidades del usuario.
- Diseñar la estructura del sistema de generación de reglas procedimentales a partir de los requerimientos identificados.
- Construcción del sistema de generación de reglas aplicando el modelado procedimental de gramáticas de división propuesto en el proyecto.
- Establecer pruebas al sistema de generación de reglas evaluando la eficiencia y practicidad en comparación con los métodos tradicionales.

## 1.7 Estructura del artículo

La estructura del informe permite una mejor comprensión de la creación de reglas procedimentales, permitirá al lector conocer diferentes flujos de trabajo y aplicaciones, se observará la metodología desarrollada para la creación de la aplicación, las pruebas realizadas y los resultados obtenidos durante el proyecto.

# 2. MATERIALES Y MÉTODOS/METODOLOGÍA

## 2.1 Arquitectura del sistema

La arquitectura del sistema está compuesta por tres fases, como se observa en la **Ilustración 1**, la primera fase se integran datos GIS y análisis de imágenes fotográficas mediante técnicas de visión artificial. A través de GIS, se obtiene información geoespacial como las coordenadas del edificio, su tipo y el área del lote.

En la fase dos, los datos recopilados en la Fase 1 son procesados mediante el Software, una herramienta desarrollada específicamente para el proyecto GENOME. Este software utiliza gramáticas divididas para generar reglas procedimentales, que describen cómo modelar tridimensionalmente el edificio capturado. El software toma como entrada parámetros como la geometría del axioma 2D y los atributos arquitectónicos. La salida es un archivo de regla CGA.

En la tercera fase consiste en implementar las reglas generadas en la Fase 2 en **CityEngine**, esta herramienta utiliza las reglas procedurales para extraer axiomas 2D a modelos 3D completos. El usuario simplemente arrastra la regla procedimental generada a un lote 2D que represente la geometría del edificio capturado. En cuestión de segundos, CityEngine genera un modelo tridimensional que refleja con precisión el estilo arquitectónico y los atributos del edificio

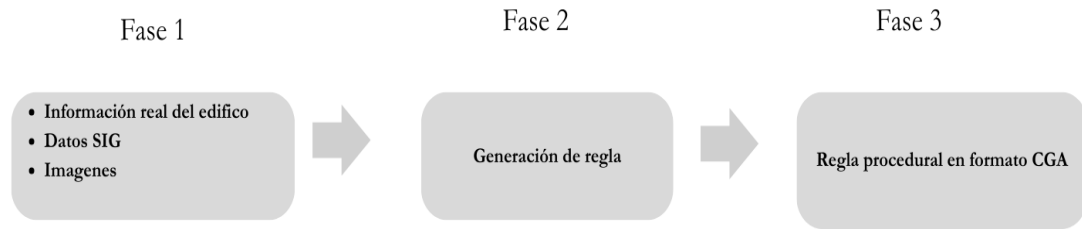


Figura 1: Fases de la arquitectura del sistema

## 2.2 Explicación de la metodología

Para el cumplimiento del desarrollo de este proyecto se plantea el uso de la metodología conocida como Proceso de Software Personal(PSP) fue creada y definida por Watts Humphrey del Software Engineering Institute en la Carnegie Mellon University como proceso de auto mejoramiento, diseñado para ayudar a controlar, administrar y mejorar la forma en que se trabaja, esta metodología son un conjunto de prácticas definidas en sus diferentes fases para la gestión de software y la mejora de la productividad personal en tareas de desarrollo de sistemas informáticos (Morejón & Armas, 2020).

Fuente: Elaboración propia

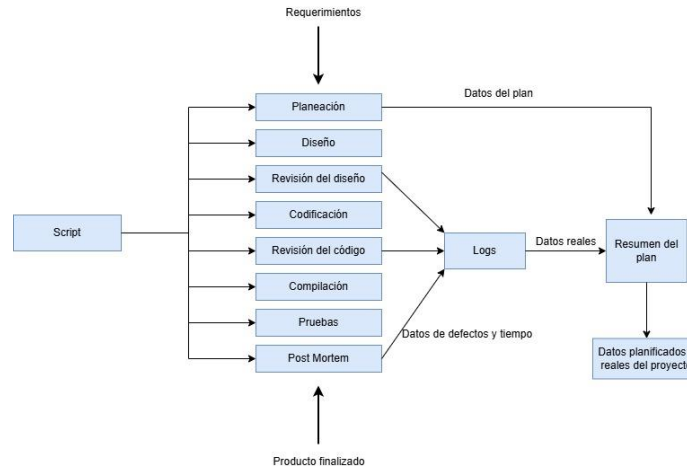


Figura 2: Imagen metodología PSP

## 2.3 Etapas de la metodología

### 2.3.1 Planeación

En esta etapa fueron definidos los objetivos específicos del proyecto y las necesidades del usuario. Se identifican las herramientas de forma que potenciaran el desarrollo. El uso de las guías de PMBok, sirvieron de apoyo para gestionar el alcance, tiempo y calidad del proyecto. Dado que la gestión del tiempo es crucial en la metodología, se utilizó Jira para la creación del cronograma y de las actividades con el fin de llevar un registro y cumplimiento del tiempo y las tareas.

### 2.3.2 Diseño.

La fase del diseño se enfoca en estructurar y conceptualizar la solución del proyecto, se emplean herramientas útiles para la elaboración de diagramas como casos de uso, diagramas de entidad-relación, diagramas de flujo, diagrama de secuencia. El diseño permite comprender claramente las funcionalidades del sistema y alinearlas con las necesidades del cliente. Los diagramas permitieron identificar los componentes del sistema, las entidades y sus relaciones, el comportamiento global del sistema y también el comportamiento de las vistas y la comunicación del sistema.

### 2.3.3 Código

En esta etapa se desarrolla el sistema basado en el diseño. Se realizaron actividades como, implementación de la lógica, seguimiento de estándares, documentación, registro del tiempo también se entrenó un modelo de inteligencia artificial que captura información por medio de imágenes. Se desarrollaron las vistas para enviar datos por campo de texto, cargar datos CSV e imágenes. Se desarrolló un módulo que se encarga de la generación de las reglas y cada vista cuenta con funcionalidades para borrar, descargar y volver al menú principal. Para el desarrollo se manejaron estándares como el IEEE830 y las guías del PMBok.

### 2.3.4 Compilación

En esta fase, se validan y preparan los componentes de desarrollados, las actividades realizadas fueron la detección de errores, correcciones, registro de defectos, preparación para pruebas. Esto garantizó la estabilidad y funcionalidad básica del sistema, por lo tanto, la compilación prepara el sistema para la fase de pruebas, asegurándose que no haya fallos estructurales.

### 2.3.5 Pruebas

En las pruebas se garantiza que el sistema cumpla con los requisitos funcionales y no funcionales, entre las actividades realizadas estuvieron las pruebas unitarias que sirvieron para evaluar los componentes individuales con el fin de verificar que funcionaran de manera adecuada, también se realizaron pruebas de integración para analizar el comportamiento del modelo en la herramienta, se realizaron validaciones para identificar que el sistema cumple con los requisitos del cliente y por último se registraron los resultados.

### 2.3.6 Post Mortem

Esta es la etapa final de la metodología, se realiza un análisis exhaustivo del proyecto para identificar aprendizajes y oportunidades para realizar mejoras, esta etapa implicó actividades como la evaluación del desempeño la revisión de métricas, la identificación de mejoras y la documentación donde se realiza un resumen de los hallazgos, lecciones aprendidas y estado final del proyecto.

## 3. RESULTADOS Y DISCUSIÓN

El proyecto mostro un porcentaje alto en el cumplimiento de los objetivos, que fueron establecidos al inicio. Como resultado se obtuvo una herramienta funcional y eficiente. Los resultados han sido plasmados en tablas y gráficos que permiten resaltar el comportamiento de las actividades clave. La metodología PSP permitió el registro detallado de los tiempos y métricas lo que facilitó identificar y corregir errores.

La **figura 2** representa las tareas cumplidas en cada etapa, todas las tareas fueron programadas dentro del cronograma. Documentar esta información ayudó a identificar etapas que fueron más críticas que otras, como lo fue la programación. El realizar una adecuada planeación permitió el cumplimiento con los plazos establecidos, asignando los recursos de manera efectiva.



Figura 3: Gráfico circular de las tareas cumplidas por fase

**Análisis de la desviación:** La desviación fue tomada de la función que se encuentra en la **figura 3**. Al realizar un análisis de la **figura 4**, se evidencia la desviación que se presentaron en tareas relacionadas con la elaboración

de diagramas y ajustes de diseño, lo que indica una subestimación de recursos como el tiempo, sin embargo, en otras etapas mostro una desviación negativa, lo que refleja un uso eficiente del tiempo asignado. La ilustración muestra que la desviación no fue mayor al 50% lo que confirma que el proyecto se mantuvo dentro de márgenes aceptables y sin alteraciones significativas

$$Desviación = \frac{Esfuerzo\ real\ (horas) - esfuerzo\ estimado\ (horas)}{Esfuerzo\ estimado\ (horas)} * 100$$

Figura 4: Función de desviación



Figura 5: Grafico de barras Desviación tiempo estimado Vs tiempo real

### Revisión de métricas

Tiempo de Ejecución:

El tiempo establecido en cada fase fue adecuado dentro de lo pactado, no se presentaron desviaciones anormales en la consecución de cada actividad, en algunas etapas hubo retrasos debido a errores encontrados que presentaron un grado significativo de complejidad, todos los errores fueron corregidos, en la **figura 5** se muestra el comportamiento del tiempo estimado y el real, el comportamiento de ambo muestra una gestión eficiente en cada etapa ya que ha sido cumplido durante los tiempos pactados.

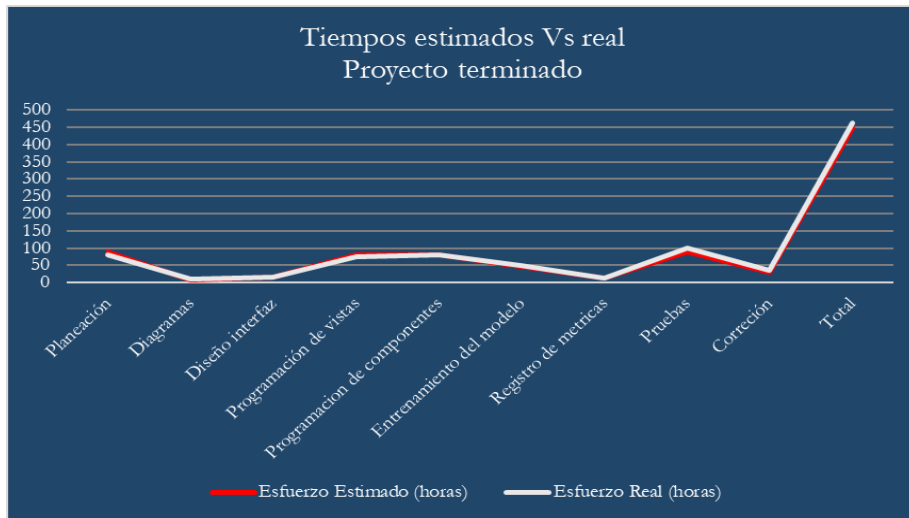


Figura 6: Gráfico de líneas, comportamiento del tiempo real sobre el estimado

### Calidad del Producto:

Para la calidad del producto se establecieron tres criterios, la funcionalidad, estabilidad y desempeño, a cada con un peso según su importancia. Cada criterio es medido con un porcentaje, el cual es hallado al dividir las tareas realizadas sobre las pactadas y multiplicarlo por cien, algunas son el análisis de los componentes, en el caso de las funcionalidades se califican para obtener un resultado que pondere la actividad, en la estabilidad se realizaron pruebas y se identificó el porcentaje de cumplimiento, por último el desempeño, que fue realizado en pruebas como la inicialización del sistema, recursos y pruebas.

En la **Tabla 1** se muestran los criterios, porcentaje en el cumplimiento y el cumplimiento del ponderado total que fue del 92% reflejando la buena gestión para el cumplimiento del proyecto. En general se ha logrado satisfacer las expectativas del cliente, cumpliendo con los objetivos establecidos. Sin embargo, hay un área de mejora que podría ser abordada para alcanzar el cumplimiento del 100%

Tabla 1. Tabla de ponderado cumplimiento final

Criterio	Cumplimiento (%)	Peso (%)	Ponderación (%)
Funcionalidad	100	50	50
Estabilidad	80	30	24
Desempeño	92	20	18,4
<b>Cumplimiento ponderado final</b>		<b>100</b>	<b>92,4</b>

**Tabla 2. Criterio de calificación**

<b>90%-100%: Excelente</b>
<b>75%-89%: Bueno</b>
<b>50%-74%: Satisfactorio</b>
<b>&lt;50%: Necesita mejoras</b>

**Métricas de evaluación de la herramienta**

En esta sección se presentan los resultados de la **Tabla 3**, comparando la generación manual de reglas CGA con el uso de la herramienta desarrollada, evaluando eficiencia, reducción de errores, flexibilidad y escalabilidad. Los tiempos para crear las reglas con la herramienta se obtuvieron mediante quince muestras por vista, calculando el tiempo desde la generación de la regla hasta su extrusión en CE. Al comparar con la generación manual, se observó una mejora significativa en el tiempo necesario para generar una regla básica, destacando la automatización del proceso. El software logra una mejora del 100% en eficiencia, optimizando tiempos, lo que representa un beneficio clave en entornos donde estos factores son esenciales.

**Tabla 3. Evaluación de la herramienta**

<b>Caso de Prueba</b>	<b>Método Manual (minutos)</b>	<b>Con Herramienta (minutos)</b>	<b>Diferencia (minutos)</b>	<b>% de Mejora</b>
Residencial	30	0:56	43	100%
Comercial	30	0:46	43	100%
Industrial	30	0:36	43	100%

**Defectos y Errores:**

En la **Tabla 4** se muestran los defecto y errores encontrados, cada defecto fue gestionado y corregido, el error de diseño de botones tomaba otra posición al expandir la ventana, se identificaron dentro del código y se encontró la forma de que estos mantuvieran su posición, esto también fue aplicado a los listbox y a los cuadros de texto, el problema de los campos de texto tuvo la misma solución que la de los botones, también las vistas presentaban redundancia de escritura de código, se creó una clase que permitiera a las vistas realizar llamados a las funciones para la optimización del código. Todos los errores fueron solucionados y documentados durante el tiempo que se había establecido en la fase de pruebas

Tabla 5. Errores corregidos

Defecto	Fase	Fecha	Descripción	Tiempo en corregir
Error diseño	Diseño	06/11/2024	Los botones se van de su posición	60min
Error diseño	Diseño	07/11/2024	Arreglo de campos de texto	120 min
Errores lógica	Lógica	07/11/2024	Arreglo del ListBox	200min
Errores lógica	Lógica	09/11/2024	Código repetido	240min
Errores lógica	Lógica	11/10/2024	Error en los campos que carga manual según la categoría del edificio.	180min

### Identificación de Mejoras

- **Áreas de Mejora:**

Una debilidad identificada fue que el desarrollo fue dirigido solo por una persona teniendo que cumplir con todas las actividades, haciendo que esta persona tuviera que utilizar intervalos adicionales para poder cumplir con las tareas, para la creación de software es necesario la intervención de varias personas, es importante las ideas de que estás pueden generar.

Se recomienda identificar bien todas las fases y utilizar marcos de referencia para el desarrollo ya que estos permiten establecer un alcance, tiempo y calidad del producto, además es importante que se tengan en cuenta las necesidades y recomendaciones de terceros para tener presente otros aspectos además se debe tener clara las actividades y ejecutarlas en el tiempo establecido para que se llegue a lo pactado antes o en el momento adecuado.

### Lecciones Aprendidas

Dentro de las lecciones aprendidas se identificó que la planificación debe ser exhaustiva y se deben considerar todos los aspectos necesarios del proyecto ya que en algunas actividades faltó análisis lo cual condujo a una desviación en el tiempo. Por otra parte, se encontraron errores de diseño a causa de un análisis previo deficiente, pero al realizar una validación continua permitió la identificación y corrección de los errores.

Como lecciones futuras es adecuado obtener una visión más clara y esto se logra involucrando a todas las partes interesadas también se recomienda implementar ciclos de retroalimentación de manera frecuente y realizar pruebas durante todas las fases para reducir errores.

## 4. CONCLUSIONES

El proyecto logro cumplir con sus objetivos, centrados en el desarrollo de una herramienta para la captura y generación de reglas que optimiza el tiempo y mejora la eficiencia en comparación con métodos manuales. El

desarrollo fue posible gracias al cumplimiento de los objetivos generales que permitieron identificar los requisitos, diseñar la estructura del sistema y codificar la herramienta.

Los resultados presentados en la **Tabla 3** evidencia el impacto positivo de la herramienta, destacándose una mejora significativa en los tiempos de creación de reglas en comparación con el proceso manual. Además, las reglas que son creadas por el software trabajan de manera adecuada en el ecosistema de CityEngine, posibilitando la creación de estructuras que fueron categorizadas como residenciales, industriales y comerciales. Por lo tanto, el trabajo realizado no solo cumplió con los objetivos planteados, sino que también demostró su relevancia práctica, al facilitar la gestión eficiente de entornos mediante la optimización del tiempo en actividades clases.

## REFERENCIAS

### I. REFERENCIAS

- Alomía, G., Loaiza, D., Zúñiga, C., Luo, X., & Asoreycacheda, R. (2021). Procedural modeling applied to the 3D city model of Bogota: A case study. doi:<https://doi.org/10.1016/j.vrih.2021.06.002>
- Badwi, I. M., Ellaithy, H. M., & Youssef, H. E. (2022). 3D-GIS Parametric Modelling for Virtual Urban Simulation Using CityEngine. doi:<https://doi.org/10.1080/19475683.2022.2037019>
- BERNHARD, K., Georgios, K., Thomas, L., & George, D. (2023). 3D Gaussian Splatting for Real-Time Radiance Field Rendering. doi:<https://doi.org/10.1145/3592433>
- Dang, F. Z., Sardroud, J. M., & Shirzadi, M. (2024). BIM Implementation for PMBOK Enhancement in the Construction Industry. doi:DOI: 10.29007/k7nf
- Díaz, D. J. (2022). Generación procedural de edificios 3D utilizando los lenguajes VEX/Python (houdini) a partir de datos georreferenciados por sistemas de información geográfica (SIG).
- Fitri, M. N., Marena, O., Hisam, O. A., Hafiz, M. Y., & Izzati, A. K. (2022). Suitability of Open Street Map (OSM) for 1:50,000 Topographic Map. doi:DOI: 10.1088/1755-1315/1051/1/012012
- Hossain, I., Shen, I.-C., Igarashi, T., & Kaick, O. v. (2023). Data-guided Authoring of Procedural Models of Shapes. doi:<https://doi.org/10.1111/cgf.14935>
- Kopowski, J., Mrelá, A., Mikołajewski, D., & Rojek, a. (2024). Enhancing 3D Printing with Procedural Generation and STL Formatting Using Python. doi:<https://doi.org/10.3390/app14167299>
- Kurt, M., & Bülbül, M. A. (2021). PROCEDURAL 3D FOUNTAIN MODELING. doi:<https://doi.org/10.22531/muglajsci.908164>
- Morejón, Y. L., & Armas, C. A. (2020). El Proceso de Software Personal en la asignatura Proyectos Informáticos para la formación del técnico medio en Informática. Obtenido de <http://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/1748>
- Nikçi, R., Župan, R., & Racetin, I. (2024). Geovisualization of Buildings: AI vs. Procedural Modeling. doi:<https://doi.org/10.3390/app14188345>
- Qin, X., Mao, W., Hu, Z., Zheng, H., & Xu, X. (2023). Procedural modeling and layout method for a generic ancient Chinese city. Obtenido de <https://link.springer.com/article/10.1007/s11042-023-16942-1>
- Rayhan, A., & Gross, D. (2023). The Rise of Python: A Survey of Recent Research. doi:David Gross
- Sacha, J., & Mohit, G. (2024). Radiance Fields from Photons. doi:10.48550/arXiv.2407.09386
- Sezen, G., Cakir, M., Atik, M. E., & Duran, Z. (2022). DEEP LEARNING-BASED DOOR AND WINDOW

DETECTION FROM BUILDING FAÇADE. doi:<https://doi.org/10.5194/isprs-archives-XLVIII-B4-2022-315-2022>

Shi, Y., Wang, N., & Guo, X. (2022). YOLOV: Making Still Image Object Detectors Great at Video Object Detection. doi:<https://doi.org/10.48550/arXiv.2208.09686>

Wang, X., Song, Y., & Tang, P. (2020). Generative urban design using shape grammar and block morphological analysis. doi:<https://doi.org/10.1016/j.foar.2020.09.001>

Yadav, Y., Colucci, E., Boccardo, P., & Zlatanova, S. (2024). A Geodatabase Design for the Development of a Digital Twin for Urban Environments: A Case Study from Turin, Italy. doi:<https://doi.org/10.5194/isprs-archives-XLVIII-4-2024-525-2024>

Zhang, D. (2024). Research on the Application of Geographic Information Systems in the Construction of Smart Cities. doi:DOI: 10.54254/2754-1169/96/2024MUR0112

Zhang, M., Wu, J., Liu, Y., Zhang, J., & Li, G. (2022). GIS Based Procedural Modeling in 3D Urban Design. doi:DOI: 10.3390/ijgi11100531