



DIRECCIÓN GENERAL DE UNIVERSIDADES
TECNOLÓGICAS Y POLITÉCNICAS



UNIVERSIDAD POLITÉCNICA DE VICTORIA

FEASIBILITY ANALYSIS TOOL FOR THE LEED ENERGY & ATMOSPHERE OPTIMIZE ENERGY PERFORMANCE CREDIT

TESINA

QUE PARA OBTENER EL GRADO DE
INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN

PRESENTA

DIEGO ALEXIS ORTIZ VÁZQUEZ

DIRECTOR

DR. MARCO AURELIO NUÑO MAGANDA

CO-DIRECTOR

ING. LUIS CARLOS LARA VÁZQUEZ

ORGANISMO RECEPTOR

JONES LANG LASALLE

CIUDAD VICTORIA, TAMAULIPAS, ABRIL DEL 2025

UNIVERSIDAD POLITÉCNICA DE VICTORIA

FEASIBILITY ANALYSIS TOOL FOR THE LEED ENERGY & ATMOSPHERE OPTIMIZE ENERGY PERFORMANCE CREDIT

T E S I N A
QUE PARA OBTENER EL GRADO DE
INGENIERÍA EN TECNOLOGÍAS DE LA
INFORMACIÓN

PRESENTA:
DIEGO ALEXIS ORTIZ VÁZQUEZ

DIRECTOR
DR. MARCO AURELIO NUÑO MAGANDA

CO-DIRECTOR
ING. LUIS CARLOS LARA VÁZQUEZ

ORGANISMO RECEPTOR
JONES LANG LASALLE

CIUDAD VICTORIA, TAMAULIPAS, ABRIL DE 2025



Ciudad Victoria,
Tamaulipas, a
**14 de Enero de
2025**

JONES LANG LASALLE. INC
Ing. Luis Carlos Lara Vázquez
PRESENTE



La Universidad Politécnica de Victoria tiene a bien presentar a **ORTIZ VAZQUEZ DIEGO ALEXIS** estudiante del programa académico de **INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN**, con número de matrícula **2030282** y seguro facultativo IMSS número **49-21-02-2159-1**; quién deberá realizar su práctica profesional de **ESTADÍA**, a partir del **08 de Enero de 2025** al **11 de Abril de 2025**, con duración de **600 horas** desarrollando el proyecto "**Feasibility analysis tool for the LEED Energy & Atmosphere Optimize Energy Performance Credit**" que le fué asignado.

Al concluir se le extenderá la carta de liberación al evaluarlo satisfactoriamente y además le solicitaremos su valiosa opinión respondiendo el formulario que se le enviará por mail, referente al desempeño del practicante, la información que nos proporcione, es de vital importancia para la mejora de los programas académicos que ofrece la Universidad Politécnica de Victoria para la formación de profesionistas altamente especializados.

El practicante deberá cumplir con el Reglamento Interno aplicable al personal en su centro de trabajo.

Sin otro particular.

ATENTAMENTE

M. A. OTHÓN CANO GARZA
DIRECTOR DE VINCULACIÓN

C.C.P. MARCO AURELIO NUÑOMAGANDA
ASESOR INSTITUCIONAL



UNIVERSIDAD POLITÉCNICA DE VICTORIA

Av. Nuevas Tecnologías 5902
Parque Científico y Tecnológico de Tamaulipas
Carretera Victoria Soto La Marina Km. 5.5
Cd. Victoria, Tamaulipas. C.P. 87138

Tel: (834) 1711100 al 10
www.upvictoria.edu.mx



JONES LANG LASALLE.INC

Chicago Illinois, a 14 de Enero del 2025
Asunto: Carta de Aceptación

M.A OTHÓN CANO GARZA
DIRECTOR DE VINCULACIÓN
UNIVERSIDAD POLITÉCNICA DE VICTORIA
PRESENTE

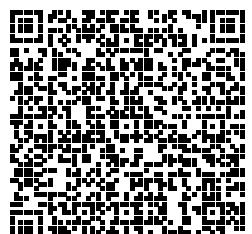
Hacemos de su conocimiento que hemos aceptado al estudiante **ORTIZ VAZQUEZ DIEGO ALEXIS** del programa académico de **INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN**, con número de matrícula **2030282** de la Universidad Politécnica de Victoria, para realizar su **ESTADÍA** en nuestra empresa **JONES LANG LASALLE.INC**, durante el periodo comprendido del día **08 de Enero de 2025 al 11 de Abril de 2025**, el estudiante estará colaborando en el proyecto "**Feasibility analysis tool for the LEED Energy & Atmosphere Optimize Energy Performance Credit**" con una carga horaria total de **600 horas**

Al concluir satisfactoriamente sus prácticas profesionales, se le entregará al estudiante su carta de liberación debidamente formalizada.

Con el objetivo de colaborar en la mejora de los programas académicos de la Universidad Politécnica de Victoria, estamos de acuerdo en responder a la brevedad posible el formulario de evaluación del desempeño del practicante previo a emitir la liberación de **ESTADÍA**.

Sin otro particular.

ATENTAMENTE
Ing. Luis Carlos Lara Vázquez
ASESOR EMPRESARIAL



JONES LANG LASALLE.INC

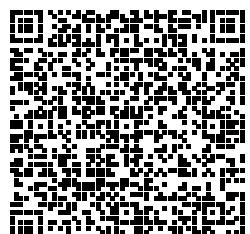
Chicago Illinois, a 09 de Abril del 2025
Asunto: Carta de Liberación

M.A OTHÓN CANO GARZA
DIRECTOR DE VINCULACIÓN
UNIVERSIDAD POLITÉCNICA DE VICTORIA
PRESENTE

Por medio de la presente me permito comunicarle que el estudiante **ORTIZ VAZQUEZ DIEGO ALEXIS** del programa académico de **INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN**, con número de matrícula **2030282** de la Universidad Politécnica de Victoria, terminó satisfactoriamente su **ESTADÍA** desarrollando trabajos y actividades directamente relacionadas con el proyecto **Feasibility analysis tool for the LEED Energy & Atmosphere Optimize Energy Performance Credit**, con una duración de **600 horas**, en el periodo comprendido del **08 de Enero de 2025 al 11 de Abril de 2025**

Sin otro particular por el momento, hago propicia la ocasión para enviarle un cordial saludo.

ATENTAMENTE
Ing. Luis Carlos Lara Vázquez
ASESOR EMPRESARIAL





CARTA DE ACEPTACIÓN DEL DOCUMENTO PARA SU IMPRESIÓN

Cd. Victoria, Tamaulipas a 10 de Abril de 2025

Diego Alexis Ortiz Vázquez
PRESENTE

Le comunico que el Programa Académico de Ingeniería en Tecnologías de la Información le ha otorgado la autorización para la impresión de su Tesina de Estadía Práctica cuyo título es:

**Feasibility analysis tool for the LEED Energy & Atmosphere
Optimize Energy Performance Credit**
ATENTAMENTE



Dr. Marco Aurelio Nuño Maganda
ASESOR INSTITUCIONAL

c.c.p Director de programa académico

EVALUACIÓN DE ESTADÍA

Rúbrica para evaluación de la presentación y el reporte de estadía

Nombre del alumno: DIEGO ALEXIS ORTIZ VÁZQUEZ

Calificación final: 100

Periodo: ENERO-ABRIL 2025

Ponderación	Aspecto a Evaluar	Competente 10	Independiente 9	Básico Avanzado 8	No Competente 5
40	Resultados y Actividades	Estrechamente relacionados al perfil de egreso de su programa académico	Parcialmente relacionados al perfil de egreso de su programa académico	Escasamente relacionados al perfil de egreso de su programa académico	Escasamente relacionados al perfil de egreso de su programa académico
30	Exposición de las actividades de la estadía	Detalladas y sustentadas con respecto a los resultados que se obtuvieron	Detalladas y sustentadas parcialmente con respecto a los resultados que se obtuvieron	Detalladas parcialmente con respecto a los resultados que se obtuvieron	Detalladas escasamente con respecto a los resultados que se obtuvieron
10	Material visual Lenguaje verbal	Uso el lenguaje y la terminología apropiadas; El material visual está organizado, adecuado y suficiente	Uso el lenguaje y la terminología apropiadas El material visual está parcialmente organizado y es suficiente	Uso el lenguaje y la terminología son parcialmente apropiadas; El material visual está parcialmente organizado y es suficiente	Uso el lenguaje y terminología es inapropiado; El material visual no está organizado y es insuficiente
10	Exposición en Idioma Inglés	Pronunciation is clear so language is easily understood (2.5) Uses fluent connected speech, occasionally disrupted by search for correct form of expression (2.5) Uses topic related vocabulary without problems (2.5) Responds to questions using varied and descriptive vocabulary and language structures (2.5)	Pronunciation is understandable, but there are slight errors (2.25) Speech is connected but frequently disrupted by search for correct form of expression (2.25) Uses some topic related vocabulary sufficient to communicate ideas (2.25) Responds to questions using simple but accurate vocabulary and language structures (2.25)	Pronunciation is understandable most of the time, marked native accent and many errors (2) Speaks with simple sentences, sometimes not connected, but is understood (2) Uses basic vocabulary to communicate ideas (2) Partly responds to simple questions, with limited vocabulary and language structures (2)	Pronunciation makes language very difficult to understand (1) Uses one-word/two-word utterances (1) Unable to communicate ideas due to lack of vocabulary (1) Uses isolated words or sentence fragments to respond to questions (1)
5	Respuesta a los cuestionamientos de los evaluadores	Clara y satisfactoria	Clara y parcialmente satisfactoria	Clara e insuficiente	Confusa e insuficiente
5	Autorización de tesina en tiempo y forma	Presenta en tiempo y forma	Presenta en tiempo y forma con la mayoría de requerimientos solicitados	Presenta en tiempo y con algunas limitantes de los requerimientos solicitados.	Presenta fuera de tiempo y con los mínimos requerimientos solicitados.

Dr. Marco Aurelio Nuño Maganda
ASESOR INSTITUCIONAL

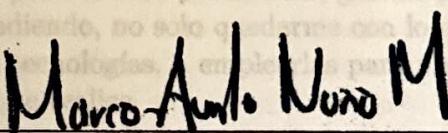
Dr. Héctor Hugo Avilés Arriaga
EVALUADOR

EVALUADOR DE INGLÉS

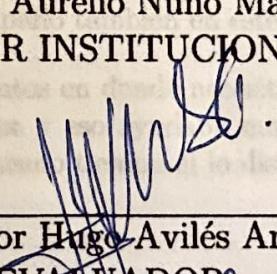
REGISTRO DE EVALUACIÓN DE EXPOSICIÓN DE ESTADÍA

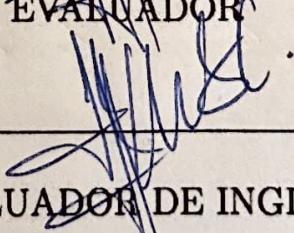
Siendo las 08:00 horas del día 15 de Abril de 2025, el alumno **Diego Alexis Ortiz Vázquez**, del programa académico **Ingeniería en Tecnologías de la Información**, con matrícula **2030282**, presentó la exposición de la estadía realizada durante el cuatrimestre **Enero-abril 2025**, en el **Jones Lang LaSalle**, con el proyecto titulado **Feasibility analysis tool for the LEED Energy & Atmosphere Optimize Energy Performance Credit**.

Una vez concluido el proceso de evaluación, y con base a la rúbrica establecida para éste propósito, se determina que la calificación de la estadía es 100.


Marco Aurelio Nuño M

Dr. Marco Aurelio Nuño Maganda
ASESOR INSTITUCIONAL


Dr. Héctor Hugo Avilés Arriaga
EVALUADOR


EVALUADOR DE INGLÉS

Agradecimientos

En este espacio quiero expresar mi más profundo agradecimiento a mi familia, quienes han sido un pilar fundamental a lo largo de este camino. Con su apoyo incondicional hoy me encuentro finalizando mis estudios.

A mi mamá, por quién siempre cada mañana se levantaba para hacerme un lonche para ir a la universidad y siempre se despedía de mi cada mañana. Por aquellos momentos en donde me sentía estresado y sin mucho ánimo, siempre me apoyaba de la manera que ella podía para salir adelante en cada trabajo y proyecto que se me atravesara.

A mi papá, quien al igual que mi mamá, me apoyaba mucho y me daba ánimos en los momentos donde me quería dar por vencido pero ellos siempre me daban un respiro más.

A mi hermano mayor, quien por él estoy finalizando esta carrera que él me recomendó que estudiara, me ha abierto los ojos a un mundo donde pueda ejercer sin necesidad de preocuparme por llegar temprano a la oficina y que pueda trabajar desde casa. Agradezco mucho su apoyo al principio de la carrera dándome una laptop con la cual me ha acompañado en todo este proceso de preparación para la vida profesional, gracias por tus consejos y ánimos para que siga estudiando y aprendiendo, no solo quedarme con lo que enseñan en la escuela, sino a abrirme a conocer nuevas tecnologías, a emplearlas para un propósito profesional y ganar experiencia en cada *script* que realice.

A mi hermano menor, quien me acompaña también en este viaje.

A mi perro Perry, por aquellos momentos en donde necesitaba darme un respiro y despegar mi mente, salíamos a pasear al parque y eso ayudaba mucho a mi persona para tener un momento de respiro, además que al mismo tiempo él lo disfrutaba.

Resumen

En este documento se registran los avances realizados para el desarrollo de una herramienta de análisis dirigida a los gerentes de proyectos en el área del desarrollo inmobiliario sustentable. En el área antes mencionada, muchos edificios conllevan a una serie de agentes que actúan en contra del medio ambiente lo que conlleva a ciudades que generan mucha más contaminación ambiental debido a la falta de atención que se les da a estos edificios.

Alrededor de todo el mundo, muchas empresas, organizaciones e inversionistas son partícipes en la lucha para conservar el ambiente y reducir los gases de efecto invernadero. Organizaciones como el *U.S. Green Building Council* ha contribuido a desarrollar un sistema para combatir con esta problemática, dicho sistema permite calificar a aquellos edificios que cumplen con los requisitos para evitar la propagación de gases de efecto invernadero y presevar el ambiente. A estos se les otorga un certificado llamado LEED y ha sido objeto de gran valor para todos interesados.

Sin embargo, el alcanzar este reconocimiento conlleva una gran agrupación de documentación y demás papeles para ser revisados y obtener un puntaje basado en la información antes proporcionada.

Para agilizar este proceso de certificación, en esta tesina se registran los avances realizados para el desarrollo de una herramienta web que permite a los gerentes de proyectos de JLL, por medio de llenado de información sobre su edificio a determinar un puntaje y certificado que puede alcanzar su edificio lo que llevaría a estos gerentes a mejorar y hacer cambios en la infraestructura de dicho edificio para obtener un puntaje mucho más alto que no solo dará prestigio al edificio, sino que también contribuirá al ambiente de manera positiva.

A pesar que el proyecto continúa en curso, se han establecido las bases para una plataforma que simplificará significativamente el proceso de evaluación LEED, permitiendo a los gerentes de proyectos tomar decisiones informadas en las etapas tempranas del diseño/construcción sustentable. Este avance representa un paso importante hacia la optimización de recursos y la promoción de prácticas de construcción más ecológicas en el sector inmobiliario.

Palabras clave: Sistema web, Empresas, Simulación Energética, Desarrollo Sustentable, LEED.

Summary

This paper records the progress made in developing an analysis tool for project managers in sustainable real estate development. In the area mentioned above, many buildings lead to a series of agents that act against the environment leading to cities that generate much more environmental pollution due to the lack of attention given to these buildings.

Around the world, many companies, organizations, and investors are involved in the fight to conserve the environment and reduce greenhouse gases. Organizations such as the U.S. Green Building Council have helped to develop a system to combat this problem, which qualifies buildings that meet the requirements to prevent the spread of greenhouse gases and preserve the environment. These are awarded a certificate called LEED and have been of great value to all interested parties.

However, achieving this recognition involves a large grouping of documentation and other paperwork to be reviewed and scored based on the information provided above.

To streamline this certification process, this bachelor's thesis records the progress made in the development of a web-based tool that allows JLL project managers, by filling in information about their building, to determine a score and certificate that their building can achieve which would lead these managers to improve and make changes to the building's infrastructure to obtain a much higher score that will not only bring prestige to the building but also positively contribute to the environment.

Although the project is still ongoing, the groundwork has been laid for a platform that will significantly simplify the LEED evaluation process, allowing project managers to make informed decisions in the early stages of sustainable design/construction. This development represents an important step towards optimizing resources and promoting greener building practices in the real estate sector.

Keywords: Web System, Enterprises, Energy Simulation, Real Estate Sustainable, LEED.

Índice

Agradecimientos	VII
Resumen	VIII
Summary	IX
Índice	X
1. Introducción	1
1.1. Definición del problema y justificación del proyecto	1
1.2. Objetivo general	2
1.3. Objetivos específicos	2
1.4. Alcances y limitaciones	2
1.5. Contenido	4
2. Fundamentos teóricos	5
2.1. Modelado Energético	5
2.2. LEED	6
2.2.1. Sistema LEED	7
2.2.2. Proceso de certificación LEED	8
2.3. OpenStudio	8
2.3.1. OpenStudio SDK	9
2.3.2. OpenStudio PAT	9
2.3.3. EnergyPlus	10
2.3.4. ¿Qué ventajas ofrece un software de simulación de energía?	11
2.3.5. EnergyPlus frente a otros software de simulación	12
2.4. La programación dentro del modelado energético	13
2.4.1. Python	13
2.4.2. Streamlit	13
2.4.3. Ruby	16
3. Sistema Propuesto	17
3.1. Análisis de requerimientos	17
3.1.1. Requerimientos funcionales	17
3.1.2. Requerimientos no funcionales	17
3.2. Arquitectura del Sistema	18
3.2.1. Capa de Presentación	18
3.2.2. Capa de Procesamiento	18
3.2.3. Capa de Datos	19
3.2.4. Diagrama de Arquitectura del Sistema	19
3.3. Diagrama de Contexto	19
3.4. Diagramas de Casos de Uso	20
3.5. Diseño de Pantallas del Sistema	21
3.6. Diseño de Entradas y Salidas	23

3.7. Algoritmos	26
4. Implementación del Sistema y Pruebas	35
4.1. Hardware Utilizado	35
4.2. Software Utilizado	36
4.3. Elementos a Evaluar	36
4.4. Pruebas	37
4.5. Problemas Encontrados y Correcciones	40
5. Conclusiones Finales	44
5.1. Conclusiones sobre el sistema desarrollado	44
5.2. Trabajos y perspectivas para el futuro	44
Índice de figuras	46
Índice de cuadros	47
Índice de algoritmos	48
Referencias	49

1. Introducción

1.1. Definición del problema y justificación del proyecto

LEED (Liderazgo en Energía y Diseño Ambiental) es uno de los sistemas de certificación de edificios verdes más utilizados en todo el mundo. Desarrollado por el Consejo de Edificación Verde de Estados Unidos (USGBC), LEED proporciona un marco para edificios verdes saludables, altamente eficientes y que ahorran costos [1].

En los Estados Unidos, la certificación LEED se ha vuelto cada vez más popular e influyente en la industria de bienes raíces comerciales. Muchos mercados de oficinas de alta calidad ahora tienen más del 50 % de su stock de Clase A certificado por LEED u otros estándares de construcción verde [2]. Esta alta tasa de adopción refleja la creciente importancia de las credenciales de sostenibilidad para las propiedades comerciales de alta gama.

JLL (*Jones Lang LaSalle*) es reconocida como el mayor proveedor de servicios de certificación LEED y otras certificaciones de edificios verdes a nivel mundial [3]. Como firma líder en servicios de bienes raíces comerciales, JLL ayuda a los clientes a navegar el complejo panorama de las certificaciones de sostenibilidad e implementar estrategias para mejorar el rendimiento de los edificios.

Entender y optimizar los créditos en la categoría de Energía y Atmósfera es crítico por varias razones [4] [5]:

- El uso de energía es un contribuyente importante al impacto ambiental y los costos operativos de uno edificio. La categoría de Energía y Atmósfera aborda directamente la eficiencia energética y el uso de energía renovable.
- Con el creciente enfoque en la descarbonización, el rendimiento energético se ha convertido en una prioridad para muchos propietarios y ocupantes de edificios. Un fuerte desempeño en esta categoría puede aumentar significativamente la puntuación LEED general de un proyecto.
- A medida que las regulaciones sobre el rendimiento energético de los edificios se vuelven más estrictas, sobresalir en los créditos de Energía y Atmósfera ayuda a preparar los edificios para el futuro a estándares en evolución.
- Muchos inquilinos corporativos ahora tienen objetivos ambiciosos de sostenibilidad, lo que hace que los edificios energéticamente eficientes y de bajas emisiones de carbono sean más atractivos en el mercado de arrendamiento.
- Los inversores están considerando cada vez más el rendimiento energético y las emisiones de carbono en su evaluación de activos inmobiliarios, lo que hace que las fuertes credenciales en Energía y Atmósfera sean valiosas para la valoración de activos.

Al desarrollar herramientas para estimar mejor los puntos LEED en la categoría de Energía y Atmósfera, JLL busca ayudar a los clientes a optimizar sus estrategias de sostenibilidad y

maximizar el valor de sus inversiones en edificios verdes.

1.2. Objetivo general

Llevar a cabo el desarrollo de una herramienta web accesible que permita a los gerentes de proyectos de sustentabilidad en JLL estimar de forma rápida e independiente el número de puntos LEED que su proyecto logre para el Crédito de Optimización del Rendimiento Energético de Energía y Atmósfera, esto considerando las últimas actualizaciones de la certificación LEED y sus diversos escenarios de edificios dentro de los Estados Unidos.

1.3. Objetivos específicos

- Diseñar una interfaz intuitiva para ingresar especificaciones de edificios y datos de ubicación
- Integrar e implementar visualizaciones interactivas para los resultados de rendimiento energético y estimaciones de puntos LEED con la infraestructura existente de *OpenStudio* con un diseño responsivo para varios dispositivos y tamaños de pantalla.
- Implementar y desarrollar APIs para facilitar la comunicación entre el *front-end* y los modelos de *OpenStudio* para asegurar una recuperación y procesamiento de datos eficiente para tiempos de respuesta rápidos.
- Integrar y mejorar la herramienta existente de geocodificación inversa de JLL para proporcionar al usuario datos climáticos relevantes para ubicaciones en los Estados Unidos.
- Realizar pruebas de usabilidad exhaustivas para asegurar que la herramienta sea más fácil de usar e intuitiva.
- Crear una documentación completa del usuario y crear materiales de capacitación para equipos internos y posibles usuarios externos.
- Configurar un proceso de implementación para actualizaciones y lanzamientos sin problemas para mantener la herramienta alineada con futuros cambios de LEED.

1.4. Alcances y limitaciones

El objetivo principal en este proyecto, el cual se encuentra actualmente en desarrollo y se prevé finalizar para junio de 2025, es desarrollar una herramienta web en donde tanto los gerentes de proyectos de JLL como sus clientes puedan acceder a ella, permitiendo a ambos usuarios estimar puntos LEED en la categoría de Optimización del Rendimiento Energético. Todo esto de la mano con la implementación de modelos energéticos de OpenStudio, se garantizarán resultados técnicos, precisos y alineados a los estándares actuales de modelado energético, con ayuda de las herramientas de desarrollo programables como Python y Ruby se busca desarrollar una herramienta web con una interfaz interactiva donde muestre gráficos

dinámicos basado en la entrada de datos del edificio, esto permitirá a los usuarios que manejen la herramienta obtener una estimación de certificado LEED para su edificio calificando los campos ingresados para un desarrollo sustentable que el proyecto genera para el medio ambiente, por el momento la aplicación se enfocará en su desarrollo para escritorio, para lo que llevarlo a dispositivos móviles como teléfonos o tabletas se desarrollará más adelante en el proyecto.

Para los usuarios que lleguen a manejar la herramienta, se dispondrá de guías y manuales de usuario completos para que estos usuarios puedan familiarizarse con la herramienta y esto pueda guiar al usuario a utilizar la herramienta.

La herramienta está limitada a estimar un puntaje basado en el territorio de los Estados Unidos manejados por JLL, por lo que otros países como México, Canadá o algún otro país extranjero no estará contemplado para esta estimación.

Otra limitación que se tiene para este proyecto es el tiempo que se tomará para desarrollarlo, ya que se tiene un contrato de seis meses para cumplir con dicho proyecto, por lo que este documento no podrá presentar en su totalidad el producto final desarrollado para la institución, sin embargo, se demostrará más adelante los avances y desarrollos obtenidos durante este proyecto, al igual que los problemas que se enfrentaron durante su desarrollo.

Otro obstáculo que se presenta para el desarrollo de este proyecto es la computadora en donde se trabaja para este proyecto, la empresa otorgó una computadora portátil para el desarrollo del antes mencionado, esto quiere decir que el equipo está manejado por la empresa, por lo que la instalación de programas no se puede hacer de manera manual y se requiere generar tickets al departamento de TI para que se apruebe la instalación de los o el programa que se requiere, una vez que se aprueba dicho ticket un especialista se conectará de manera remota al equipo para instalar el software que se necesita.

Por otra parte, hoy en día se tienen diversos frameworks que ayudan con la construcción de sistemas webs, algunos usando JavaScript para la parte de front-end como lo es React, Astro, Vue, etc y C# o PHP para la parte de back-end con Laravel, .NET y demás. En este proyecto se cuenta únicamente con Ruby y Python como lenguajes de programación lo que limita el uso de estos frameworks convencionales para el desarrollo de la plataforma, además que el asesor empresarial ya seleccionó el framework en donde se desarrollará este proyecto el cual es *streamlit*.

La precisión de los resultados depende de la correcta implementación y funcionalidad de los modelos energéticos de OpenStudio lo que lleva a que los problemas técnicos de dicha herramienta pueden afectar el rendimiento general del proyecto. Dado que el proyecto se enfoca únicamente en el Crédito de Optimización del Rendimiento Energético en la categoría de Energía y Atmósfera esto deja de lado otras áreas del sistema de certificación LEED y finalmente las actualizaciones y regulaciones de las certificaciones LEED pueden cambiar a través del tiempo lo que puede significar una demora en la adaptabilidad de estas nuevas regulaciones a los estándares LEED.

1.5. Contenido

Este documento está estructurado por los siguientes capítulos:

1. **Introducción:** Aquí se presenta el contexto actual del proyecto en donde se define el problema a resolver, la justificación, los objetivos y los alcances y limitaciones.
2. **Fundamentos teóricos:** En este segundo capítulo se abordan conceptos claves relacionados con el modelado energético, las certificaciones LEED y las herramientas de programación utilizadas alrededor de dicho proyecto.
3. **Sistema Propuesto:** Para el tercer capítulo se detalla el diseño y la arquitectura de la herramienta de análisis.
4. **Implementación del Sistema y Pruebas:** En el cuarto capítulo se describe el proceso del desarrollo hasta la fecha del proyecto mencionando las tecnologías utilizadas y las pruebas realizadas.
5. **Conclusiones Finales:** Finalmente, para el capítulo final del documento se resumen los logros alcanzados durante el desarrollo del proyecto y se proponen líneas de desarrollo futuro.

2. Fundamentos teóricos

El entorno construido, representado principalmente por los edificios, desempeñan un papel crucial en los esfuerzos globales por mitigar el cambio climático y lograr los objetivos de descarbonización. Los edificios no solo satisfacen las necesidades esenciales de vivienda, trabajo y recreación, sino que contribuyen a enfrentarse a grandes desafíos para la sostenibilidad a su impacto en el medio ambiente.

Estudios recientes muestran:

- Los edificios contribuyen al *40 %* de las emisiones globales de carbono, convirtiéndose en unos de los mayores emisores del cambio climático.
- En México, el sector edificado consume un *18 %* de la energía total generada en el mundo, reflejando así su papel en la creciente demanda energética global.
- Además, México genera un *12 %* de las emisiones de carbono a nivel global, generando una necesidad urgente de tomar medidas estratégicas sostenibles en el diseño, construcción y operación de edificios.

En este contexto, herramientas como el modelado energético y sistemas de certificación LEED surgen como enfoques claves para optimizar el desempeño energético de estos edificios y promover un desarrollo sustentable.

A continuación, se analizarán y detallaran conceptos claves y herramientas en donde se integran de manera efectiva la solución a lograr una sostenibilidad en un entorno construido.

2.1. Modelado Energético

Los países en vías de desarrollo presentan algunos problemas derivados de la energía de los cuales están profundamente ligados a la alta dependencia de las fuentes tradicionales de energía como el sol, viento, agua, calor, etc. El uso extensivo de estos recursos tradicionales trajo consigo en consecuencia efectos adversos en el medio ambiente tales como la deforestación, la disminución de la fertilidad del suelo y el deterioro de los ecosistemas. Frente a estos impactos de suma importancia, surge la necesidad de recopilar grandes cantidades de información que permitan comprender la relación existente entre los recursos disponibles y las demandas energéticas, para lograr esto se necesitan herramientas especializadas para analizar diferentes problemáticas para lograr resultados que contribuyan al proceso de planificación energética eficiente y sostenible [6].

Con esto, los modelos energéticos emergen como una herramienta estándar para la planificación, previsión y optimización de los recursos energéticos [7]. El modelado energético evalúa la viabilidad de diferentes fuentes de energía y permite diseñar sistemas más eficientes y sostenibles. Este enfoque permite mitigar el impacto ambiental de las actividades humanas y garantizar un desarrollo energético equitativo y alineado con las necesidades de las poblaciones en donde más se necesiten [8].

Existen diversos usos y aplicaciones en tema del modelado energético en que destacan proyectos relacionados con la sostenibilidad como lo pueden ser los siguientes [9]:

- **Gestión de residuos como combustible alternativo:** Se tratan de procesos de conversión de residuos en energía (WTE) en donde se reduce la emisión de metano dando en consecuencia opciones de combustibles de bajo carbono.
- **Transiciones en el sector transporte:** Se hace un análisis del cambio hacia combustibles y tecnologías de bajo carbono en un sector dominado por los combustibles fósiles.
- **Acceso a energía en zonas rurales:** Llevan a cabo la evaluación del impacto de mejorar el acceso a combustibles modernos en comunidades en desarrollo, contribuyendo a la reducción de la pobreza y al cumplimiento de los objetivos de desarrollo sostenible.

2.2. LEED

Las certificaciones LEED por sus siglas en inglés (*Leadership in Energy and Environmental Design*) se trata de un sistema de certificación desarrollado por el *U.S. Green Building Council (USGBC)*. Es una certificación que promueve el diseño, la construcción y el mantenimiento de edificios sostenibles y eficientes desde el punto de vista energético [10].

El modelado energético juega un papel muy importante en los procesos de certificaciones LEED en donde a través de herramientas específicas que serán mencionadas más adelante permiten analizar y diseñar múltiples escenarios en donde pueden optimizar variables en relación con el consumo de energía, las emisiones de gases de efecto invernadero y costos de operación [11].

Estas certificaciones proporcionan un marco para edificios ecológicos, altamente eficientes minimizando costos. Su objetivo es cumplir con estándares específicos para demostrar que se han implementado medidas significativas para reducir el consumo de agua, energía y combustible, reducir la cantidad de desechos que genera y ayudar a reducir la huella ambiental del edificio.

Otro de los beneficios que ofrecen estas certificaciones LEED son [12] [13] [14]:

- **Establece liderazgo en la industria de la construcción sustentable y mercado inmobiliario:** Hoy en día muchas empresas buscan que sus edificios tengan esta certificación dada la urgencia de la agenda global de 2040 en donde se busca reducir la huella de carbono en un 95 %.
- **Valida logros sustentables mediante un proceso de revisión externo e imparcial:** Con esto quiere decir que el *mánager* de quien se encarga de llevar a cabo la estrategia para que un edificio sea sustentable no es quien declara que éste sea sustentable, aquí entran en juego agencias externas que validan esta certificación como GBCI (*Green Business Certification Inc*) que avalan que el equipo de desarrollo siguen las

medidas establecidas para llevar a cabo esta certificación.

- **Otorga exposición internacional:** Esta certificación es reconocida a nivel global pero en algunos países tiene más auge como lo es Estados Unidos en donde es muy solicitada, por otra parte, en Europa también es reconocida, sin embargo, existen otras certificaciones locales que tienen más impacto.
- **Brinda la oportunidad de calificar para incentivos, fondos de inversión y premios privados y públicos:** En la mayoría de países desarrollados, se ofrece un incentivo por tener esta certificación que reducen la huella de carbono. En el caso de México, el único estado en donde este incentivo es otorgado es en la ciudad de Zapopan, Jalisco.

2.2.1. Sistema LEED

LEED funciona para todos los edificios en todas las fases de desarrollo, desde nueva construcción hasta los edificios existentes, así como para todos los sectores de la construcción, desde escuelas, hogares hasta hospitales y sedes corporativas. LEED posee 4 diferentes estándares para adaptarse a cualquier tipo de edificio, en la figura 1, se muestran estos 4 tipos de certificaciones que USGBC ofrece. A continuación se mencionan los siguientes:

- **LEED for Building Design + Construction (BD+C):** Está enfocada a aquellos edificios nuevos o en planes de construcción [15].
- **LEED for Interior Design + Construction (ID+C):** Sólo aplica para inmuebles dentro del edificio, es decir, edificios que se encuentran en obra negra donde se le hace una remodelación a un espacio en concreto [16].
- **LEED for Building Operations + Maintenance (O+M):** Esta certificación aplica para edificios en operaciones arriba de un año, aplica para edificios que cuenten o no con alguna certificación de LEED pero es necesario aplicar esta certificación cada 3 años dada la vigencia que tiene. Las certificaciones mencionadas no tienen una fecha de caducidad [17].
- **LEED for Neighborhood Development:** Estas certificaciones van más enfocadas a colonias o comunidades más completas, un ejemplo de una ciudad que posee esta certificación en México es el municipio de San Pedro Garza García en Nuevo León [18].



Figura 1: Certificaciones LEED

Para estas certificaciones es necesario cumplir una serie de requisitos que se convierten en puntos para ser acreedores de una certificación LEED en donde los mínimos de puntos necesarios para obtener uno de estos niveles de certificación es de 40 puntos, estos requisitos constan principalmente de aplicar medidas para reducir la huella de carbono tales como reducción de aguas residuales, consumo energético, uso de materiales bio-degradables, etc.

A continuación se presentan estos niveles de certificación junto con la puntuación para certificarse en estos niveles [4]:

1. **Platino:** +80 puntos.
2. **Oro:** 60 - 79 puntos.
3. **Plata:** 50 - 59 puntos.
4. **Certificado:** 40 - 49 puntos.

2.2.2. Proceso de certificación LEED

Para llevar a cabo el proceso de certificación LEED, es necesario registrar el proyecto en la plataforma de LEED Online, donde se implementarán las metodologías correspondientes. Los asesores encargados del proyecto deben cargar la documentación requerida, asegurándose de cumplir con los criterios establecidos para alcanzar la puntuación necesaria. Posteriormente, la documentación es sometida a una revisión preliminar por parte de los organismos certificadores, quienes evalúan el cumplimiento de los requisitos en función de la información proporcionada. Finalmente, se lleva a cabo una revisión final en donde basado con la puntuación obtenida, se certifica dicho edificio [19].

2.3. OpenStudio

Los edificios sustentables se categorizan como edificios de alto rendimiento ya que exigen un enfoque de diseño integrado para garantizar que todos los sistemas funcionen en conjunto de forma eficiente. Para conseguir resultados rentables y energéticamente eficientes, la integración de los sistemas debe tenerse en cuenta desde las primeras fases de diseño. Los diseñadores

de edificios necesitan un marco de software robusto que les permita realizar simulaciones multidisciplinares y exhaustivas de los edificios. Para dar una respuesta a esta necesidad entra OpenStudio el cual es una colección de herramientas para apoyar el modelado energético con ayuda de EnergyPlus [20]. Se trata de una herramienta de código abierto que permite el desarrollo de extensiones, herramientas y demás manipulaciones en donde se necesite en un sector en particular.

2.3.1. OpenStudio SDK

OpenStudio SDK actúa como un conjunto de herramientas para complementar el modelado energético. Este SDK ha sido diseñado para ser compatible con los motores de simulación de EnergyPlus permitiendo así el análisis y diseño de nuevas construcciones, calibración de modelos, evaluación de rehabilitaciones y el análisis a gran escala del potencial tecnológico [21]. Se puede entender que esta herramienta ofrece a los desarrolladores una estructura de alto nivel utilizando un enfoque orientado a objetos para representar los componentes de un edificio, todo esto por medio de creación de scripts en Python y Ruby.

2.3.2. OpenStudio PAT

Por sus siglas en inglés *Parametric Analysis Tool*, es otra de las herramientas que ofrece OpenStudio. En la figura 2, se muestra la interfaz gráfica de la herramienta corriendo en un sistema MacOS. Su principal propósito es explorar los múltiples escenarios de diseño y operación en donde se evalúa el impacto de diferentes variables en el rendimiento energético de una edificación todo esto a través de comandos de programación específicos para la manipulación de estos modelos [22]. PAT permite a su vez ejecutar simulaciones paramétricas en EnergyPlus utilizando diversas combinaciones para evaluar como afecta el desempeño energético. Además facilita el análisis de sensibilidad que ayuda a identificar aquellos parámetros con mayor impacto en resultados energéticos [23].

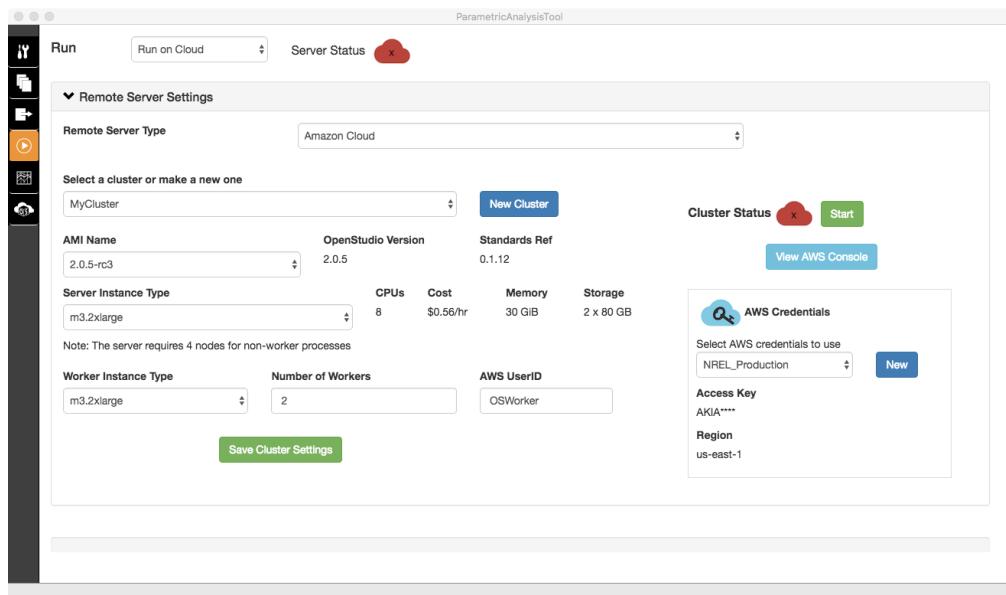


Figura 2: Interfaz gráfica de la herramienta OpenStudio PAT

2.3.3. EnergyPlus

EnergyPlus es un software avanzado de simulado energético desarrollado por el Departamento de Energía de los Estados Unidos [24]. El software está especializado para aquellos profesionales que se desempeñan en el sector inmobiliario y en el diseño del análisis del consumo energético de espacios, oficinas y edificios para llevar a cabo simulaciones térmicas y evaluar el rendimiento de sistemas HVAC. Estos sistemas HVAC por sus siglas en inglés *Heating, Ventilation and Air Conditioning* se tratan de sistemas que permiten la regulación de la temperatura, humedad y la calidad del aire en un espacio determinado como las infraestructuras mencionadas con anterioridad. El principal objetivo del software es el diseño de edificaciones eficientes y sostenibles que reduzcan el consumo de energía y las emisiones de carbono. En la figura 3, se muestra la interfaz gráfica de EnergyPlus corriendo en un sistema Windows.

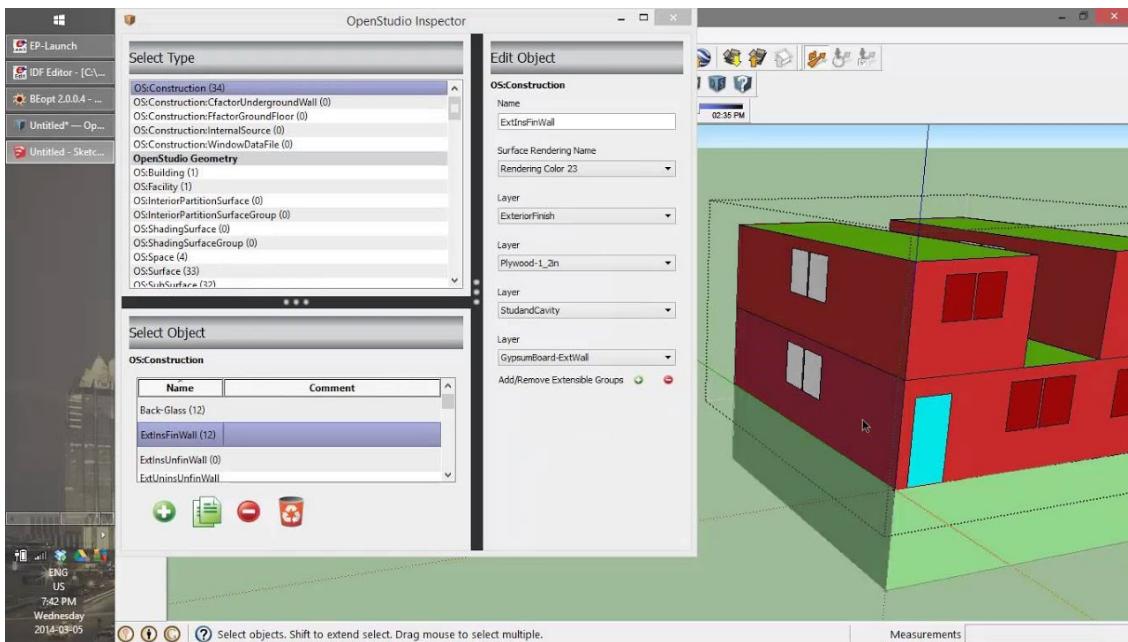


Figura 3: Software de EnergyPlus

2.3.4. ¿Qué ventajas ofrece un software de simulación de energía?

En el sector inmobiliario especialmente en el desarrollo y/o renovación de edificios sostenibles, estas herramientas como lo es EnergyPlus son esenciales. Dentro de las ventajas existentes en estos simuladores son [25] [26]:

- **Simular el costo energético:** Los clientes que contratan a estos consultores de modelado energético siempre buscan las mejores soluciones para sus intereses, por ello es que gracias a estas simulaciones de proporcionan estimaciones precisas del uso de energía en función de las características del edificio, el clima y los sistemas implementados.
- **Evaluación de costos energéticos:** Estas herramientas ayudan a calcular los costos enlazados al consumo de energía, permitiendo así optimizar el presupuesto y los recursos que estos clientes desean ahorrar.
- **Dimensionar los sistemas HVAC:** Llegar a determinar el tamaño ideal para sistemas de calefacción, ventilación y aire acondicionado garantiza un equilibrio entre la eficiencia y el confort que los clientes muchas veces buscan.

Una ventaja particular que ofrece esta herramienta de EnergyPlus es la predicción de comportamientos térmicos en función de la ubicación y las condiciones climáticas, esto quiere decir, que facilita la toma de decisiones informadas en la etapa de diseño o remodelación del espacio en particular [27]. En la figura 4, se muestra el uso del sistema HVAC abierto desde el software de OpenStudio.

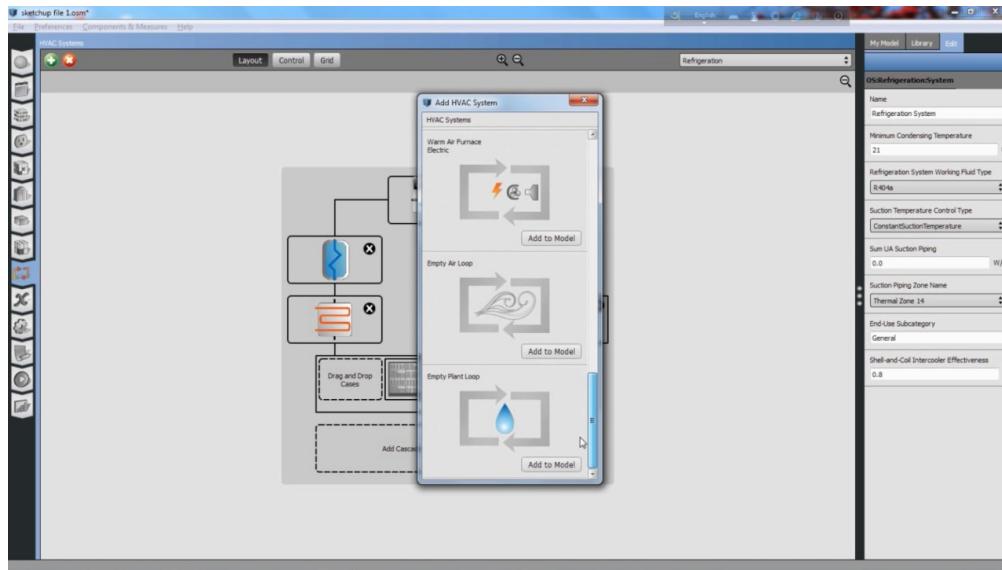


Figura 4: Uso de sistema HVAC en OpenStudio

2.3.5. EnergyPlus frente a otros software de simulación

A pesar de la diversidad de programas de simulación energética en el mercado, en particular EnergyPlus resalta más sobre los demás dado su enfoque detallado que ofrece y las capacidades avanzadas lo que en consecuencia a los ingenieros y consultores requieren para llevar a cabo sus proyectos con estos clientes. A diferencia de herramientas como eQUEST, la cual es una herramienta más amigable para usuarios principiantes, EnergyPlus ofrece una mayor precisión y flexibilidad en la simulación de sistemas más complejos [28]. Además, esta herramienta es de código abierto lo que significa que los usuarios pueden personalizar funciones en específico para satisfacer sus necesidades en proyectos que lo requieran. En la tabla [1] se abordan a grandes rasgos estas diferencias entre eQuest y EnergyPlus

Característica	eQuest	EnergyPlus
Motor de simulación	DOE-2	Propio (EnergyPlus)
Facilidad de uso	Alta (interfaz gráfica)	Baja (requiere scripting)
Detalle en modelado térmico	Medio	Alto
Simulación de transitorios	No	Sí
Modelado de HVAC avanzado	Limitado	Completo
Integración con BIM	No nativa	Sí
Código abierto	No	Sí

Cuadro 1: Comparación entre eQuest y EnergyPlus

2.4. La programación dentro del modelado energético

En el ámbito del modelado energético, las herramientas como OpenStudio dependen de lenguajes de programación para la creación, análisis y optimización de modelos de eficiencia energética. Entre los más utilizados son los dos mencionados anteriormente, Python y Ruby, ambos desempeñan un papel clave en la automatización, personalización y ejecución de simulaciones. A continuación, se abordarán la relevancia de estos dos lenguajes en el desarrollo sustentable y su aplicación en el modelado energético.

2.4.1. Python

En los últimos años, gracias al avance de las tecnologías se han creado estos lenguajes de alto nivel completamente legibles para los humanos, esto quiere decir que la sintaxis no es tan compleja como lo pueden ser los demás lenguajes disponibles en el medio, un ejemplo de estos lenguajes es Python, que cumple con estos criterios además de ser dinámico e interpretado [29]. A menudo estos lenguajes interpretados son mejores para prototipado de códigos de manera rápida. Y el hecho de ser dinámico quiere decir que las variables que se les puede asignar pueden cambiar a medida que el programador así lo deseé [30]. Por lo que para los que emplean el modelado energético les resulta bastante accesible de utilizar sin tener que complicarse demasiado conociendo la sintaxis de esta misma, además de ofrecer una variedad muy grande de librerías para emplearlas en herramientas de este propósito.

Con el apoyo de Python muchas herramientas y librerías han sido creadas bajo un propósito similar como lo es *Pandapower* que es una herramienta de análisis estático con el fin de optimizar sistemas eléctricos equilibrados [31] y también *PyLESA* con un mismo enfoque pero también con mediciones térmicas [32].

Principalmente, se busca mucho automatizar los procesos con *scripting* de Python y mejorar el flujo de trabajo de una herramienta para este ámbito de modelado energético.

2.4.2. Streamlit

Streamlit es un framework de desarrollo web para Python con el que se puede desarrollar páginas web completas en cuestión de horas [33]. Este framework ayuda positivamente a los científicos de datos al momento de querer mostrar un resultado dinámico, un nuevo modelo o un fragmento de un complicado análisis, alguna forma alterna de dar a conocer esta pieza de información sería por medio de un documento de *Word* o similar. Otra manera sería la creación de un sitio web completamente desde cero con otros frameworks disponibles para Python como *Django* o *Flask*, sin embargo, esto toma mucho tiempo, conocer mucho la herramienta y muchas veces es hasta frustrante. Aquí es donde entra Streamlit, en donde gracias a sus ventajas que ofrece de velocidad, fiabilidad y cargada de métodos pre-establecidos que permiten la construcción de estos sitios web con mucha facilidad [34].

A través de esta herramienta es como muchos usuarios han creado aplicaciones completas y complejas como lo son los LLMs (*Large Language Models*), visualizar datos, información geográfica hasta videojuegos con este framework. A continuación se mostrarán algunos usos

de este framework creados por la comunidad de Streamlit:

■ MathGPT

- Este usuario creó una inteligencia artificial capaz de resolver problemas matemáticos, sin embargo, tiene un costo de 10 dólares (USD) dicha herramienta. En la figura 5, se muestra la aplicación del usuario corriendo en la nube de Streamlit.

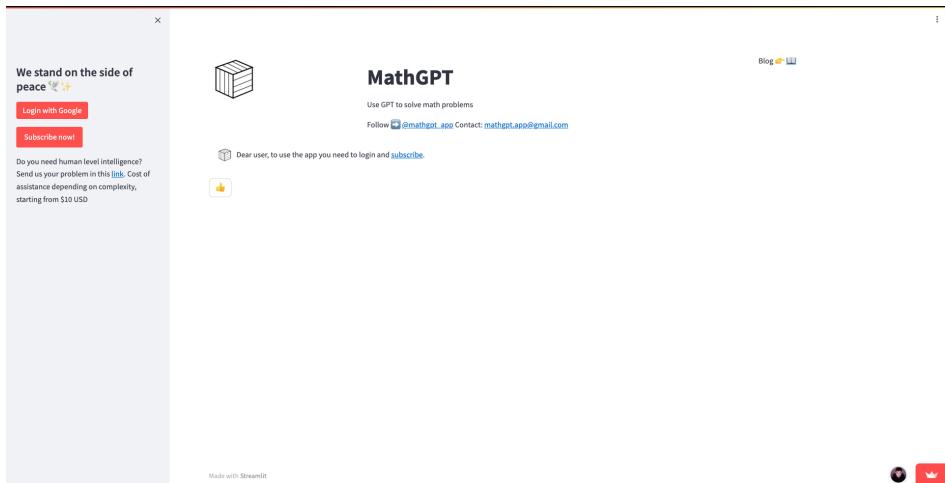


Figura 5: Aplicación de MathGPT hecha con Streamlit

■ Goodreads Analysis App

- Otro usuario creó esta herramienta para analizar tus tipos de lecturas que realizas organizándolas por la cantidad de libros leídos a lo largo de los años, cuando fueron lanzados estos libros, los porcentajes de aceptación por parte de las reseñas, etc. Todo esto tipo de información recopilada es mostrada a través de diagramas que permite una lectura de datos mucho más amigable. En la figura 6, se muestra la aplicación de *Goodreads Analysis App* ejecutándose en la plataforma de Streamlit.

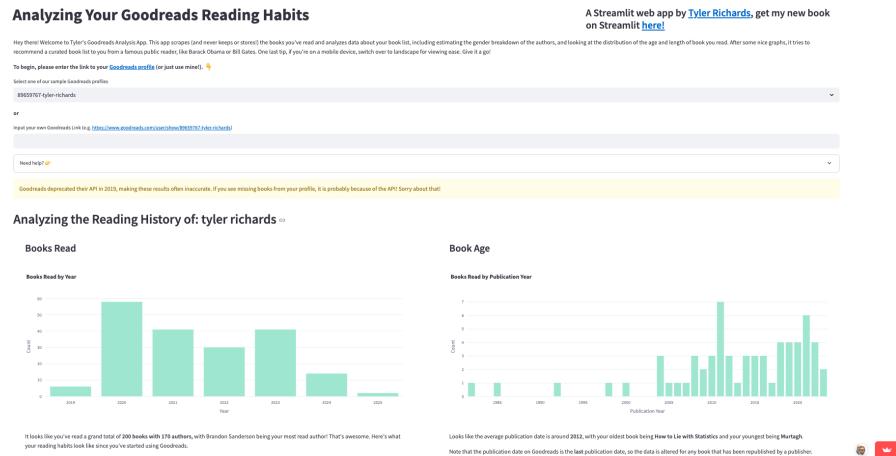


Figura 6: Aplicación Goodreads Reading Habits hecha con Streamlit

■ Migration Network

- Otro usuario desarrolló una red de migraciones de varios estados en los Estados Unidos, el cual muestra un índice de migración y cantidad de personas que migran de un estado a otro. La aplicación muestra un mapa en que se muestran a manera de grafos los estados a los que las personas migran de un estado a otro, para una mejor visualización, del lado izquierdo el usuario puede establecer los parámetros para visualizar de una mejor manera de a donde van o donde emigran estas personas de estados. En la figura 7, está la aplicación de antes mencionada ejecutándose en la plataforma de Streamlit.

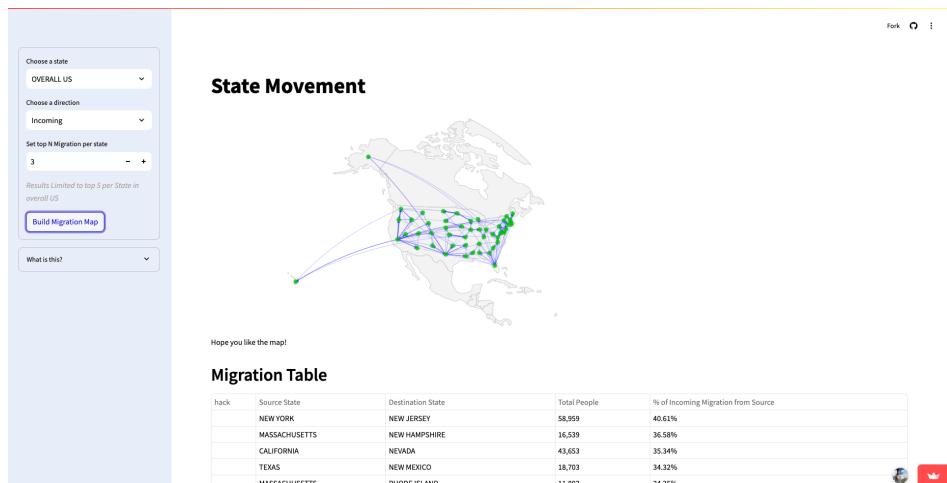


Figura 7: Aplicación Migration Network hecha con Streamlit

Con estos ejemplos de usos de aplicación con Streamlit, los gerentes de modelado energético pueden generar estas herramientas para presentarlos a clientes potenciales de manera rápida

y eficiente, además Streamlit posee un despliegue de aplicación autónomo en su página para que cualquier otra persona pueda acceder a ella de manera remota.

2.4.3. Ruby

Ruby es muy similar a Python en muchos aspectos, es un lenguaje de programación dinámico de modo que los usuarios pueden leer dicho código con facilidad, además se trata de un lenguaje orientado a objetos como Java, cada valor especial se maneja en este lenguaje de programación lo que permite ser muy versátil a la hora de querer desarrollar un programa en específico [35].

La idea de traer estos dos lenguajes de programación es que su aplicación en este mundo del desarrollo sustentable es el hecho de lograr la automatización de procesos más fácil por medio de código, tal como lo puede ser el *web-scraping* (técnica de extracción de información de sitios web de manera automática) para obtener los últimos detalles de la certificación LEED, datos geo-climáticos, etc.

La automatización de procesos trae consigo una gran productividad al momento de mejorar la carga de trabajo de un proceso en particular. Como por ejemplo; Roth, Bull, Criswell, Ellis, Glazer, Goldwasser, Kruis, Parker, Philip y Reddy crearon una herramienta de automatización de procesos para la mejora de la productividad del modelado energético por medio de Ruby, OpenStudio y demás herramientas [36]. Con ayuda de Ruby se pueden crear frameworks para este mismo medio como lo hicieron Hale, Macumber, Weaver y Shekhar con un framework análisis de modelado energético junto con otros lenguajes como C++ y C# [37].

3. Sistema Propuesto

3.1. Análisis de requerimientos

Con el propósito de desarrollar un proyecto dirigido a gerentes de proyectos y clientes de estos mismos, a continuación, se exponen aquellos requisitos funcionales y no funcionales que para la fase final del desarrollo del proyecto se compromete a cumplir:

3.1.1. Requerimientos funcionales

- Desarrollar una interfaz de usuario intuitiva con la suficiente capacidad de ingresar especificaciones de edificios y datos de ubicación.
- Implementar una visualización interactiva para los resultados de rendimiento energético y estimaciones de puntos LEED además de integrarlo con la infraestructura existente de OpenStudio.
- Crear un diseño responsivo para los diferentes tamaños de pantalla y dispositivos.
- Emplear y desarrollar APIs para facilitar la comunicación entre el front-end y los modelos de OpenStudio.
- Proporcionar al usuario entradas para seleccionar diferentes tipos de edificios tales como oficinas, almacenes, comercios, etc.
- Utilizar métodos de cálculo avanzados para estimaciones más precisas de puntos LEED.
- Validar los resultados obtenidos de la puntuación con resultados conocidos de certificación LEED.
- Crear una documentación completa de usuario para dicha herramienta y preparar documentación técnica para mantenimiento y actualizaciones futuras.

3.1.2. Requerimientos no funcionales

- Asegurar una recuperación y procesamiento de datos eficiente para tiempos de respuesta rápidos.
- Integrar y mejorar la herramienta existente de geocodificación inversa de JLL para proporcionar al usuario datos climáticos relevantes para ubicaciones en los Estados Unidos.
- Realizar pruebas exhaustivas de la herramienta en varios escenarios y pruebas de usabilidad.
- Desarrollar materiales de capacitación para equipos internos y posibles usuarios externos.

- Configurar un proceso de implementación de actualizaciones y lanzamientos sin problemas.
- Establecer sistemas de monitoreo para el rendimiento y uso de la herramienta.

3.2. Arquitectura del Sistema

Para esta arquitectura de la herramienta propuesta se evalúa el nivel que un edificio puede obtener para una certificación LEED basado en el puntaje de criterios energéticos y ambientales en donde el edificio existe. Para ello, la herramienta corre simulaciones y combinaciones de diseños alternativos usando PAT (*Parametric Analysis Tool*) de por medio a manera de simulación. Además, el sistema usa la API de Google Maps para la geolocalización de dicho edificio, en adición con esta API por medio de varios archivos de Excel se obtienen más datos relevantes sobre la ubicación ingresada, como lo es la zona climática, latitud y longitud de la ubicación y demás datos que son necesarios para las simulaciones.

En este proyecto se manejan tres capas de arquitectura; **capa de presentación, capa de procesamiento y capa de datos**.

3.2.1. Capa de Presentación

En esta capa de presentación se aborda todo lo relacionado con el *front-end* del sistema.

- El sistema consiste en un dashboard interactivo desarrollado con Streamlit, un framework de Python con el que se desarrolla de manera ágil y sencilla la impresión de los datos obtenidos.
- El dashboard permite a los gerentes de proyectos ingresar los datos de estos edificios como lo pueden ser; dimensiones, materiales, energía eléctrica, energía hidráulica, etc.
- Una vez ingresado los datos, se determina el tipo de certificación que el edificio en cuestión logra obtener, así mismo como una visualización de los datos ingresados.

3.2.2. Capa de Procesamiento

La capa de procesamiento aborda el *back-end* de la aplicación.

- El *back-end* utiliza Python y Ruby para el procesamiento de los datos ingresados por parte del usuario y los comunica con la API de OpenStudio.
- La llamada a la API de Google Maps es requerida para obtener con precisión los datos geográficos de donde se está llevando a cabo el proyecto de la certificación del edificio, misma en donde las coordenadas recibidas son comparadas con este archivo de Excel con todas las zonas climáticas de diferentes proveedores (*Department of Energy & OneBuilding*).

- Este archivo en donde se tiene los datos climáticos son obtenidos por medio de *web scraping* con un *script* de Python en el que se corre periódicamente para obtener los datos más nuevos y actualizados de las zonas climáticas.

3.2.3. Capa de Datos

En la sección de capa de datos, se abordan las herramientas en donde se obtienen datos relevantes para que el sistema funcione plenamente.

- Se hace uso de la herramienta de **OpenStudio** para la simulación del consumo energético además de la eficiencia del edificio (se implementa la librería de OpenStudio en el sistema para esta simulación).
- La API de **Google Maps** es vital para el sistema, con ello se recuperá la localización precisa de la dirección ingresada para devolver la zona climática y demás datos relevantes que el sistema regresa para determinar la certificación LEED.

3.2.4. Diagrama de Arquitectura del Sistema

A continuación, en la figura 8, se muestra la interacción entre los componentes del sistema.

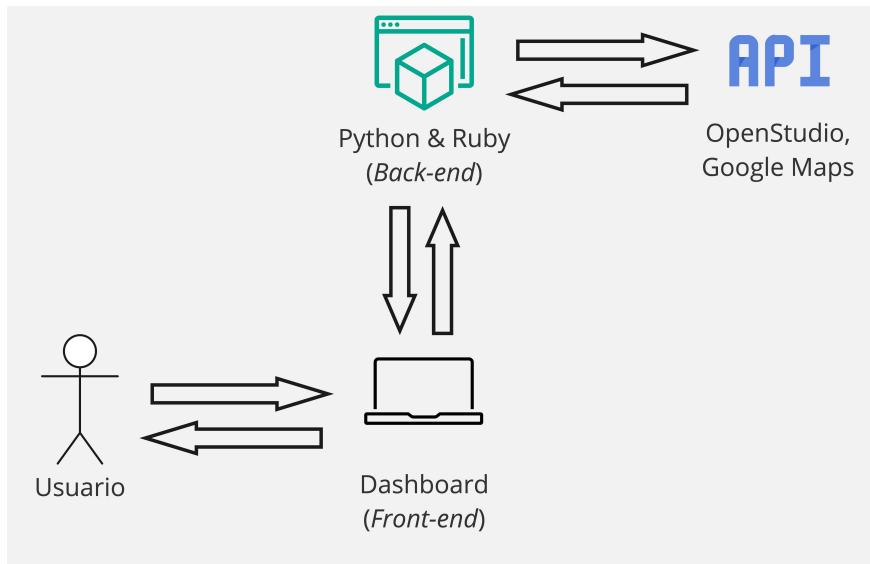


Figura 8: Diagrama de Arquitectura del Sistema

3.3. Diagrama de Contexto

En la figura 9, se muestra un diagrama de contexto en donde se ilustra una visión general de dicho sistema, aquí se presenta como el usuario (gerente de proyectos) y otras entidades externas interactúan con el sistema. Primeramente, la herramienta de análisis (dashboard) se comunica con el usuario, OpenStudio y las APIs de geolocalización con el fin de evaluar el desempeño energético del edificio.

El sistema espera recibir datos por parte del usuario, con ayuda de OpenStudio y las APIs, se devuelve un análisis detallado para conocer el tipo de certificado LEED que el edificio puede ser acreedor.

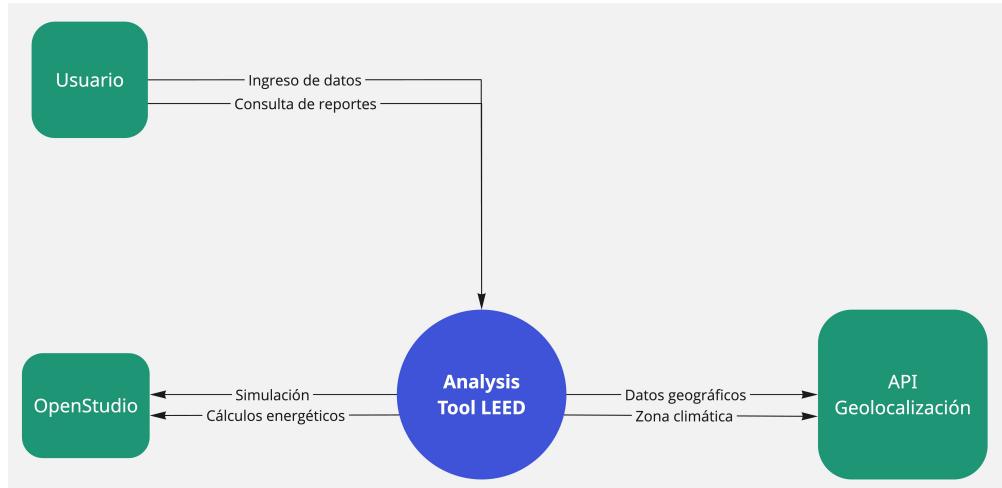


Figura 9: Diagrama de Contexto del sistema

De acuerdo con el flujo que el presente diagrama expone, el usuario es el encargado de ingresar los datos de su edificio con el que está trabajando, una vez ingresado estos datos, el sistema se pone en contacto con la API de geolocalización para obtener los datos climáticos que se requieren para tomarlos en cuenta al momento que se pasan a OpenStudio para la simulación energética, cuando OpenStudio termina su procesamiento de simulación...

3.4. Diagramas de Casos de Uso

En la figura 10, se propone el siguiente diagrama de casos de uso para el sistema de la herramienta de análisis.

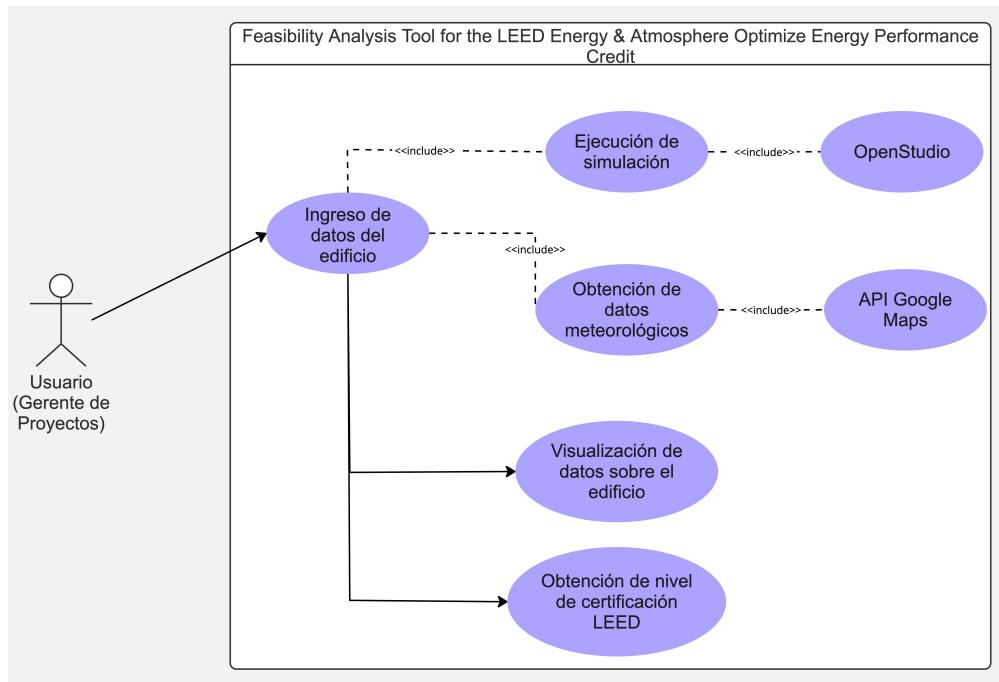


Figura 10: Diagrama de casos de uso

Lo que se propone en el presente diagrama, el agente que interactúa con el sistema en cuestión son los gerentes de proyectos que llevan a cabo la dirección de la certificación LEED de un edificio, la idea es que este agente por medio del *dashboard* propuesto ingrese los datos e información solicitada de su edificio.

Cuando el usuario ingresa la dirección de donde se ubica dicho edificio, por medio de la API de Google Maps y el archivo de los datos meteorológicos, se regresan datos relevantes de la ubicación como lo es la estación meteorológica más cercana, coordenadas geográficas y la zona climática. Datos que son relevantes para posteriormente mandarlos a OpenStudio para llevar la simulación energética.

Una vez terminado este proceso de simulación, la herramienta arroja un resultado de la posible certificación que logra alcanzar el edificio. Esto permite al gerente de proyectos realizar mejoras y cambios para alcanzar la certificación más alta que la gran mayoría de los clientes desean obtener.

3.5. Diseño de Pantallas del Sistema

Para la elaboración de la herramienta, se decidió desde un principio utilizar el framework de Streamlit para su creación, se investigó acerca de las capacidades y límites que el framework puede ofrecer a los usuarios. Una de las características que ofrece esta herramienta es la manera sencilla de presentar información acerca de un conjunto de datos con un solo método para llamar una gráfica, también la posibilidad de llenar información como si se tratara de un formulario, etc. Uno de los puntos débiles que tiene dicho framework es la dificultad de

personalizar a fondo la interfaz de usuario, la comunidad de Streamlit ha desarrollado algunos componentes que el framework por defecto no incluye como una barra de navegación, para ello es necesario instalar las librerías de estos usuarios por medio de la paquetería de Python.

The screenshot shows a clean, modern web interface for a sustainability analysis tool. At the top center, there is a title: "Analysis tool for LEED Energy & Atmosphere Optimize Energy". Below the title, the interface is divided into two main sections: "Project Information" on the left and "Building Information" on the right. Both sections contain several input fields with dropdown menus and text input boxes. In the "Project Information" section, fields include "Select a certification type" (dropdown), "Select a rating system" (dropdown), "Address" (text input), "Climate Zone" (dropdown), "Station Name" (dropdown), and "Data Source" (dropdown). In the "Building Information" section, fields include "Building Orientation" (dropdown), "Roof Effective R-Value (ft².°F·h/BTU)" (text input), "Exterior Wall Effective R-Value (ft².°F·h/BTU)" (text input), "Slab Insulation Effective R-Value" (text input), "SHGC" (text input), and "U-Factor" (text input). At the very top of the page, there is a small navigation bar with icons for home and search.

Figura 11: Diseño de pantalla principal de la herramienta web

A continuación, en la figura 11, se presenta un boceto de *Figma* de la pantalla principal de la herramienta. La idea de presentar una pantalla principal sin tener que acceder por medio de un verificador de identidad, es debido a que la herramienta solo será compartida entre los gerentes de proyectos de la empresa JLL.

En esta primer pantalla, se tiene una pequeña barra de navegación en la parte superior representado con íconos, la idea es tener una sección principal de la herramienta para el llenado de información del edificio en cuestión y por otra parte, una sección en donde se explique de manera sencilla y rápida el funcionamiento de esta herramienta, donde el gerente puede seguir los pasos para llenar la información de su edificio y ver los resultados arrojados por medio de la simulación de *OpenStudio*. En esta primer pestaña se tiene la herramienta con distintos campos de llenado de información dividida en dos columnas. La primer columna se utilizó para llenar información general del proyecto, en donde se ingresa el tipo de certificado LEED que se desea obtener los cuales son 4 tipos; **BD+C**, **ID+C**, **O+M**, **ND**, dependiendo de la certificación que el gerente desee, se despliegan los diferentes sistemas de evaluación enfocados al tipo de proyecto que el gerente se encuentre manejando en ese momento.

En la parte de *address*, el usuario ingresa la dirección en donde se está llevando a cabo

el proyecto. Aquí por medio de la API de Google Maps, recolecta las coordenadas de la dirección ingresada, revisa en el archivo *csv* alguna estación meteorológica que se encuentre más cercana y una vez que la encuentra, llena la información recuperada en los campos que están desactivados, se propuso que estén desactivados ya que son datos de alta importancia para la herramienta, dado que en la certificación LEED los datos como la zona climática, la fuente de información y el nombre de la estación son datos que no deben ser equívocos, de lo contrario perdería la oportunidad de recibir esta certificación. Es por ello que en la herramienta se aplica esta regla para que los gerentes no toquen estos datos sensibles.

En la segunda columna, el gerente debe llenar los datos del edificio a ser certificado, cada uno de estos campos posee un *placeholder* con información extra para que sea llenado el campo. Más abajo de la pantalla principal viene un botón para llevar a cabo la simulación energética y recibir un tipo de certificado LEED.

Este diseño de pantalla está inspirado de otro proyecto de la empresa con un cliente confidencial, en la figura 12, se muestra un poco sobre este diseño de pantalla que JLL tiene actualmente con un cliente confidencial. En dicho diseño, se tiene tres columnas de las cuales las primeras dos es para el llenado de información y datos sobre un tipo de edificio, mientras que para la última columna esta censurada la imagen del edificio con el que se llevará a cabo dicho proyecto y una previsualización de los datos ingresados en formato JSON.

```

{
  "project_information": {
    "gfa": null,
    "version": 20.5,
    "location": null,
    "climate_zone": null,
    "orientation": null,
    "data_source": null,
    "distance_to_station": null,
    "http_link": null,
    "latitude": null,
    "longitude": null
  },
  "building_envelope": {
    "shared_adibatic_walls": {
      "Roof": false,
      "North": false,
      "East": false,
      "South": false,
      "West": false
    }
  }
}
  
```

Figura 12: Diseño de pantalla de proyecto con cliente confidencial de JLL

3.6. Diseño de Entradas y Salidas

Durante el desarrollo de este proyecto, el asesor empresarial presentó un código de Ruby el cual se conecta con la nube privada de JLL enviando los datos de un análisis en formato JSON en donde procesa dicha información y los devuelve en forma de archivos ZIP.

El contenido de entrada es el archivo JSON que el usuario genera en la herramienta cuando

llena la información de los campos solicitados. En la figura 13, se visualiza el archivo de entrada para ser enviado a la nube de JLL, esto con con ayuda de la extensión de JSON Crack de Visual Studio Code permite visualizar como luce este dato de entrada.*script* de Ruby.

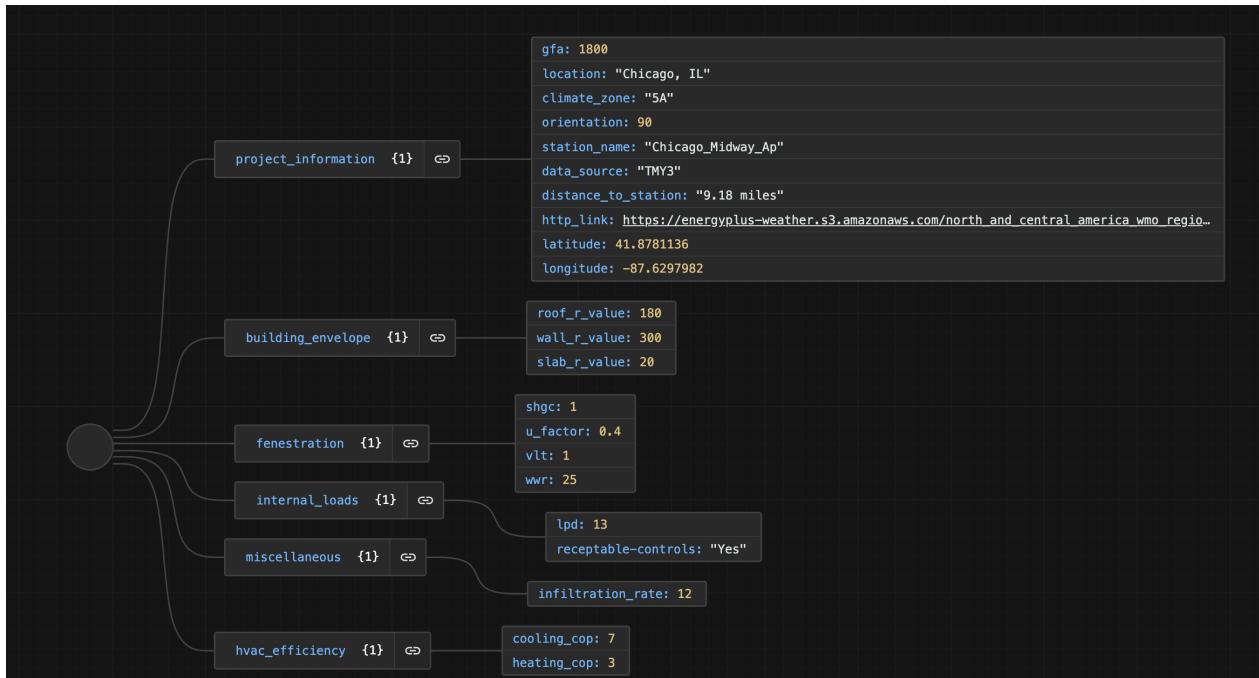


Figura 13: JSON de entrada representado de manera visual con ayuda de la extensión JSON Crack

Para este archivo de entrada, el archivo JSON hasta este momento puede aceptar los campos que se ven reflejados en la imagen, para ello el gerente de proyectos debe llenar cada uno de los campos para que pueda llenar el archivo JSON. Hasta este punto de desarrollo de la aplicación, el archivo se guarda en la raíz del directorio del proyecto, la idea que se plantea más adelante es que este archivo sea enviado a la nube y procese la información almacenada.

Por el momento lo que sigue es que con el archivo de Ruby se corre para tomar el archivo JSON ya generado con que se envía a la nube privada de JLL para hacer procesos y simulaciones que finalmente cuando termina retorna varios archivos en formato ZIP a la raíz del directorio en donde se está programando la herramienta.

Con fines de evitar mostrar todos los archivos ZIP que genera este *script* de Ruby, se mostrará una carpeta y su contenido dentro de ella.

Esta carpeta lleva por nombre: “data_point_0a03f04f-d4cb-43cc-a597-db81c7e95b69” el cual está compuesto por varios dígitos que fungen como identificador único de un resultado obtenido. Dentro de este archivo ZIP se encuentran dos archivos: **eplusout.sql** y **datapoint.zip**. En la figura 14, se muestra el contenido de este archivo ZIP junto con el nombre de los archivos que incluye.

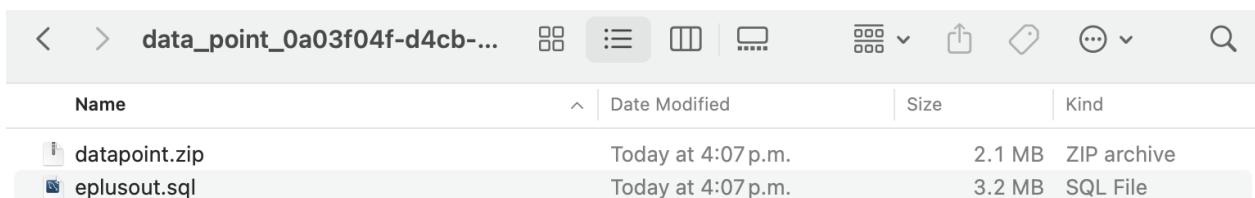


Figura 14: Contenido del archivo ZIP de los diferentes resultados obtenidos del *script* de Ruby

El primer contenido almacenado en este archivo SQL son una serie de números y letras alfanuméricos. Por otra parte, hay otro archivo ZIP nombrado *datapoint.zip*, dentro de este otro archivo existen diferentes archivos que son una colección de reportes generados en base al archivo JSON de entrada, en la tabla 2, se comparte el contenido de este archivo **datapoint.zip**:

Contenido del archivo datapoint.zip
Eplusout.audit
Eplusout.bnd
Eplusout.eio
Eplusout.end
Eplusout.err
Eplusout.eso
Eplusout.json
Eplusout.mdd
Eplusout.mtd
Eplusout.rdd
Eplusout.shd
Eplusout.sql
Eplusout.csv
Eplusspsz.csv
Eplusbtl.htm
Epluszszz.csv
In.idf
In.osm
Measure_attributes.json
Objectives.json
Pre-preprocess.idf
Results.json
Run.log
Sqlite.err
Started.job
Stdout-energyplus

Cuadro 2: Contenido del archivo datapoint.zip

3.7. Algoritmos

Una de las funcionalidades que hacen que el sistema determine las zonas climáticas así también como las coordenadas de la estación meteorológica más cercana en base a la dirección ingresada por parte del usuario es un *script* de Python en el que se lleva a cabo un *web scraping* de plataformas en donde se obtienen esta información y es concentrada en una archivo CSV para ser leído posteriormente en el archivo principal del sistema.

Este archivo funge como un actualizador de datos climáticos que se compone de varias funciones que serán presentadas a continuación:

Algoritmo 1 extract_doe_links()

```

1: Establecer URL al contenido raw de GitHub para datos meteorológicos de EnergyPlus
2: Enviar solicitud GET a URL
3: if la solicitud es exitosa then
4:     Analizar datos JSON de la respuesta
5:     Inicializar lista vacía doe_links
6:     for cada característica en datos analizados do
7:         Extraer propiedades y geometría
8:         if existe enlace 'all' en propiedades then
9:             Extraer información relevante (nombre de estación, coordenadas, etc.)
10:            if el nombre de la estación no coincide con el patrón CZ##RV2 then
11:                Crear diccionario con información extraída
12:                Añadir diccionario a doe_links
13:            end if
14:        end if
15:    end for
16:    return doe_links
17: else
18:     return lista vacía
19: end if
```

Antes de pasar a explicar el primer algoritmo, es necesario mencionar que la obtención de los datos climáticos para los Estados Unidos de América se pueden obtener de diferentes fuentes, una de las más confiables y es manejada por el gobierno de este mismo país es el **Departamento de Energía** por sus siglas en inglés *Department of Energy (DOE)* y otra fuente llamada *OneBuilding.org*. Ambas fuentes de información, tienen datos relevantes de datos climáticos, el primero se limita a únicamente exponer la información meteorológica de los Estados Unidos de América, por otra parte, *OneBuilding* recopila esta misma información a nivel global.

Esta función en particular emplea el uso de *web scraping* para la fuente de DOE que está en un repositorio de GitHub en formato *geojson*. Dado el formato, se analizó los datos en formato *JSON* para posteriormente almacenar cada uno de los datos en una lista, la información que se presenta en el *geojson* es un hipervínculo específico que lleva por encabezado "all" este

hipervínculo es esencial ya que la dirección obtenida tiene datos relevantes para poblar el archivo CSV como el nombre de la estación, las coordenadas en donde se encuentra, etc. Al momento de estar llevando a cabo este *script* se percató de un error en el nombre de una estación que comparte una expresión regular CZ##RV2, se trata de información de prueba que no aporta nada a los datos climáticos requeridos por lo que se decide separarlos del diccionario que se desea crear y no incluirlos dentro.

Algoritmo 2 extract_links_from_excel(url)

- 1: Enviar solicitud GET a url
 - 2: Leer datos Excel del contenido de la respuesta
 - 3: Convertir datos Excel a DataFrame de pandas
 - 4: **return** columnas específicas como diccionario
-

En esta otra función, los datos que se extraen de la fuente de *OneBuilding*, los hipervínculos llevan a tablas en formato Excel. Lo que realiza esta función en específico es la extracción de encabezados de información específica que están contenidas en dichos archivos, fungo como una función auxiliar para otra función que hace uso de ella más adelante del código.

Algoritmo 3 extract_years(tmyx_version)

- 1: Encontrar todos los años de 4 dígitos en tmyx_version
 - 2: Encontrar todos los rangos de años en tmyx_version
 - 3: **if** se encuentran rangos de años **then**
 - 4: **return** año final máximo de los rangos
 - 5: **else if** se encuentran años individuales **then**
 - 6: **return** año máximo
 - 7: **else**
 - 8: **return** 0
 - 9: **end if**
-

Dentro de la fuente de datos que contienen los archivos de *OneBuilding*, existen estas versiones nombradas como “tmyx” el cual es un acrónimo de *Typical Meteorological Year*, estos son conjuntos de datos climáticos registrados en un cierto periodo de año que pueden ser conjuntos de años como por ejemplo del 2002 al 2017 o simplemente registros de un solo año.

Más adelante de los algoritmos, se hace uso de esta extracción de años para dicha fuente de información, el algoritmo se encarga de obtener por medio de el hipervínculo recuperado el periodo de tiempo en el que está registrado dicha información que pueden un rango de tiempo o un solo periodo de tiempo.

Algoritmo 4 update_csv()

- 1: Establecer ruta del archivo de salida
 - 2: Llamar a extract_doe_links() para obtener datos DOE
 - 3: Inicializar lista de URLs de Excel
 - 4: Inicializar lista vacía excel_data
 - 5: **for** cada url en URLs de Excel **do**
 - 6: Llamar a extract_links_from_excel(url)
 - 7: Extender excel_data con el resultado
 - 8: **end for**
 - 9: Abrir archivo de salida para escritura
 - 10: Escribir encabezado CSV
 - 11: Escribir datos DOE en CSV
 - 12: Procesar datos de OneBuilding:
 - 13: Extraer información relevante
 - 14: Calcular prioridad basada en años
 - 15: Escribir datos procesados en CSV
 - 16: Imprimir mensaje de éxito
-

En el algoritmo 4, esta función es la más importante que está dentro de la funcionalidad del *script* ya que es la responsable de actualizar y poblar el archivo *csv* con los datos más nuevos de ambas fuentes meteorológicas. Este empieza por nombrar un nuevo archivo en formato *csv*, posteriormente se llama a la función de extraer hipervínculos de DOE, más adelante se declaró una lista con varios hipervínculos correspondientes a la fuente de *OneBuilding*, cada uno de estos hipervínculos fueron seleccionados cuidadosamente desde la fuente en donde se destaca la versión de TMYX ya que es la versión con la que se decidió trabajar para este proyecto. Cada uno de estos hipervínculos son procesados por medio de una llamada la función *extract_links_from_excel* donde se pasa como parámetro la lista de los hipervínculos. Ya que se tienen ambas fuentes declaradas en variables distintas, con el archivo *csv* que se creó con anterioridad se declaró los encabezados que van a componer estos archivos los cuales son 9 encabezados, se llenan en base a la información obtenida por ambas fuentes meteorológicas.

Se pasó a escribir y poblar este archivo con los datos de DOE y *OneBuilding*. Para los archivos de *OneBuilding* se categorizó los períodos de tiempo dando un peso más significativo a los períodos más recientes, esto con el fin de tener los datos más nuevos de cada uno hoja de cálculo que se estén actualizando con el tiempo.

Por último, escribió los datos recopilados en dicho archivo creado con anterioridad y devuelve en consola un mensaje ya sea de error o éxito según se haya procesado la actualización de datos.

Algoritmo 5 check_duplicated(file_path)

```

1: Inicializar defaultdict duplicados
2: Abrir file_path y leer CSV
3: for cada fila en CSV do
4:     Crear clave a partir de valores de fila
5:     Añadir índice de fila a duplicados[clave]
6: end for
7: Filtrar duplicados para mantener solo aquellos con más de una ocurrencia
8: if existen duplicados then
9:     Imprimir información de duplicados
10: else
11:     Imprimir "No se encontraron duplicados"
12: end if
```

Aquí el algoritmo 5 se encarga de leer el archivo *csv* ya creado y con datos de ambas fuentes pobladas de información y se encarga de revisar si existen datos que se repiten para posteriormente y los devuelve mostrando cuales fueron estos archivos repetidos.

Algoritmo 6 delete_duplicates(file_path)

```

1: Inicializar conjunto vacío visto
2: Inicializar lista vacía filas_a_mantener
3: Abrir file_path y leer CSV
4: for cada fila en CSV do
5:     Crear clave a partir de valores de fila
6:     if clave no está en visto then
7:         Añadir clave a visto
8:         Añadir fila a filas_a_mantener
9:     end if
10: end for
11: Abrir file_path para escritura
12: Escribir filas_a_mantener en CSV
13: Imprimir mensaje de éxito
```

Al igual que el algoritmo anterior, este lee el archivo creado con la información ya poblada y se encarga de borrar aquellos datos fueron encontrados como duplicados para posteriormente eliminarlos.

Algoritmo 7 main()

```

1: Llamar a update_csv()
2: Llamar a check_duplicated() con ruta de archivo CSV
3: Llamar a delete_duplicates() con ruta de archivo CSV
```

Finalmente, en la función principal del *script* se manda a llamar la función de actualizar el archivo *csv*, una vez finalizado este proceso, manda a llamar las funciones de revisar y eliminar los datos duplicados.

Este *script* en particular planea ejecutarse cada cierto tiempo con el propósito de registrar los nuevos datos meteorológicos para la herramienta de análisis.

El siguiente *script* tiene como finalidad mostrar la interfaz gráfica de la herramienta con apoyo del framework *Streamlit* en donde se muestran las distintas funciones que tiene la herramienta para su funcionalidad.

Algoritmo 8 process.address.chunk(df_chunk)

```

1: Importar bibliotecas necesarias
2: Definir GOOGLE_API_KEY
3: Cargar datos del clima EPW desde archivo CSV
4: function PROCESS_ADDRESS_CHUNK(df_chunk)
5:   for all dirección en df_chunk do
6:     Obtener datos geográficos
7:     Crear dataframe con datos geográficos
8:   end for
9:   Concatenar df_chunk con dataframe de datos geográficos
10:  for all fila en df_chunk do
11:    if coordenadas son válidas then
12:      Procesar fila
13:      Añadir fila procesada a lista de filas procesadas
14:    end if
15:  end for
16:  return DataFrame de filas procesadas
17: end function
```

En este algoritmo 8, es en donde se cargan las bibliotecas utilizadas para la creación de la interfaz gráfica, principalmente usando el framework de *Streamlit* y demás bibliotecas para su correcto funcionamiento. En este primer algoritmo se inicializa una variable para usar la API de Google que está resguardada en un archivo oculto en el sistema. Posterior a ello, se carga el archivo *csv* con el que se pobró de información de las fuentes de DOE y OneBuilding para obtener datos climáticos y demás información relevante para ser procesado a *OpenStudio*. En este primer algoritmo de la herramienta principal se creó un *dataset* para obtener los datos geográficos de la dirección ingresada por parte del gerente de proyectos. La función principal de este algoritmo es procesar la dirección ingresada para posteriormente ser llamada en la función principal del sistema.

Algoritmo 9 find_closest_weather_stations

```

1: function FIND_CLOSEST_WEATHER_STATIONS(latitude, longitude, country)
2:   Filtrar datos del clima basado en el país
3:   Calcular distancia a cada estación meteorológica
4:   Ordenar estaciones por distancia
5:   Encontrar estaciones dentro de 5 millas
6:   Seleccionar la estación más cercana
7:   Obtener información de otras estaciones cercanas
8:   return información de la estación más cercana y otras estaciones
9: end function
```

Esta función permite determinar la estación meteorológica más cercana a la dirección ingresada por el usuario. Dentro de dicha función se hace un filtro de los datos climáticos del archivo *csv* basado en el país que sea ingresado. Esto dado que por el momento la herramienta planea utilizarse para el país de Estados Unidos de América, sin embargo, si a futuro esta herramienta mejora exponencialmente, se planea mejorarlo para que funcione en cualquier parte del mundo. Una vez que los datos climáticos, pasa por un cálculo de distancia en millas entre la dirección ingresada con la estación meteorológica, posterior a ello, hay una condición para obtener las estaciones más cercanas, esta condición es un radio de distancia de menor o igual a 5 millas para encontrar con dicha estación, esta condición puede ser modificada únicamente por el programador de modo que el gerente de proyectos no deba de preocuparse por ingresar un radio de búsqueda de dicha estación. Basado en esta condición, la función ingresa en el archivo *csv* para encontrar la estación más cercana de *DOE* o *OneBuilding* según la distancia. Una vez que lo determina, pasa por otro proceso en donde obtiene otras estaciones más cercanas al primer resultado, esto se aplicó para un trabajo futuro con la herramienta en donde el usuario pueda escoger una de las estaciones que estas dos fuentes ofrece y así pueda seleccionar la que más le parezca. Finalmente la función retorna estos valores en donde únicamente importa la estación más cercana para agregarlo a los campos de texto deshabilitados.

Algoritmo 10 parse_address_string

```

1: function PARSE_ADDRESS_STRING(address_string)
2:   Analizar dirección con expresión regular
3:   return componentes de la dirección (calle, ciudad, estado, código postal)
4: end function
```

La función de analizar el *string* de la dirección ingresada utiliza la librería de expresiones regulares de *Python* para la extracción de los componentes de dicha dirección, dando como resultado un retorno de los componentes importantes de la dirección como la calle, ciudad, estado y código postal en un diccionario de datos.

Algoritmo 11 address_to_fips_and_coordinates(address_string)

```

function ADDRESS_TO_FIPS_AND_COORDINATES(address_string)
    Hacer solicitud a la API de Google Maps
    if geocodificación es exitosa then
        Extraer información relevante
        if dirección está en Estados Unidos then
            Hacer solicitud a la API del Censo
            Extraer información adicional
        end if
        Obtener información de estaciones meteorológicas
    end if
    return diccionario con toda la información recopilada
end function

```

Este algoritmo en particular permite convertir la dirección ingresada en coordenadas geográficas y un código FIPS en caso de que la ubicación se encuentre dentro de los Estados Unidos de América. Estos códigos FIPS son una serie de números que identifican a un estado o condado dentro de dicho país, los cuales estos datos son de alta prioridad para la certificación LEED.

Esta función realiza una llamada a la API de Google Maps para geocodificar la dirección ingresada, posteriormente pasa por una validación en donde si la geocodificación es exitosa (nunca se ha presentado un caso contrario) extrae únicamente los datos más relevantes de esta misma, pasa también por otra verificación en donde si la ubicación ingresada está dentro del territorio de los Estados Unidos de América, realiza una llamada a una API de FIPS llamada *Census* que es administrada por el gobierno de los Estados Unidos. A partir de este punto si ambas condiciones se cumplen, entonces se devuelve un diccionario con toda esta información recopilada. En caso de que solamente se cumpla la primera condición, los códigos FIPS no son agregados.

Algoritmo 12 is_valid_address_format(address_string)

```

function IS_VALID_ADDRESS_FORMAT(address_string)
    Verificar formato de dirección con expresión regular
    return resultado de la verificación
end function

```

En esta otra función, funge como un verificador si la cadena de texto ingresada por el gerente tiene un formato de dirección válido, esto es ya que en la pantalla principal en donde se solicita este campo de información, el formato de la dirección puede ser como (calle, ciudad, estado, país y código postal), no necesariamente el usuario debe ingresar una dirección así completa, con el simple hecho de ingresar una ciudad con el nombre del estado completo o abreviado es más que suficiente para que el código pueda interpretar correctamente dicha localización, todo esto gracias a las expresiones regulares que son aplicados en esta función.

Algoritmo 13 display_homepage()

```

1: function DISPLAY_HOMEPAGE
2:   Mostrar título “Analysis tool for LEED Energy & Atmosphere Optimize Energy Per-
   formance Credit”
3:   Crear dos columnas col1 y col2
4:   En col1:
5:     Mostrar subtítulo “Project Information”
6:     certification_type ← Selección de usuario [“BD+C”, “ID+C”, “O+M”, “ND”]
7:     rating_system ← show_certification_type(certification_type)
8:     if rating_system no está vacío then
9:       Mostrar “Certification type: ” + certification_type
10:      Mostrar “Rating system: ” + rating_system
11:      Mostrar línea horizontal
12:    end if
13:    address ← Entrada de texto del usuario
14:    if address no está vacío then
15:      Intentar:
16:        result ← address_to_fips_and_coordinates(address)
17:
18:        if result[‘country’] es “United States” then
19:          zone ← get_cz_from_fips_and_change(result[‘fips_code’], fips_df)
20:
21:        else
22:          zone ← get_cz_from_closest_weather_station(result[‘latitude’], re-
   sult[‘longitude’])
23:
24:      end if
25:      Actualizar estado de sesión con zona climática, nombre de estación, fuente de
   datos, etc.
26:      Si hay error:
27:        Mostrar mensaje de error
28:      else
29:        Mostrar advertencia para ingresar dirección
30:      end if
31:      Mostrar zona climática, nombre de estación, fuente de datos y distancia (entradas
   deshabilitadas)
32:      En col2:
33:      Mostrar subtítulo “Building Information”
34:      Obtener entradas del usuario para orientación del edificio, valores R, SHGC y Factor
   U
35:      Almacenar entradas en el estado de sesión
36:    end function

```

Pasando al siguiente algoritmo 13, su función principal es mostrar la pantalla principal de

la herramienta en donde el gerente de proyectos va llenando la información de su proyecto. Para la visualización del llenado de datos se tomó como referencia un proyecto de la empresa con un cliente confidencial para la interfaz de usuario, en este caso se declaró dos columnas, la primera para el llenado de la información del proyecto en donde se debe seleccionar el tipo de certificación LEED que el gerente esté deseando obtener, de ahí pasa una función extra que únicamente muestra el sistema de calificación a obtener basado en la selección del tipo de certificado, esto debido a que cada tipo tiene un enfoque distinto de calificación por lo que el gerente debe seleccionar cuidadosamente cual es la que se busca calcular. Después de ello, con la dirección ingresada se hace un proceso en el que se verifica si dicha dirección pertenece a los Estados Unidos de América, esto se hace para la obtención de códigos FIPS que por el momento aún no son importantes, pero que más adelante durante el desarrollo del proyecto será utilizado. En caso de que la dirección tenga algún error particular, muestra dicho mensaje de error, para que por último llene 4 campos de datos de manera automática con información obtenida de este último proceso de ingreso de dirección. Por otro lado, en la segunda columna, el gerente continua llenando la información acerca de el edificio en cuestión, se presentan más campos de entrada de datos en donde cada uno de los campos que son mostrados en esta función tienen restricciones de ingreso de datos como lo puede ser el campo de “*building_orientation*” donde únicamente se aceptan valores entre 0 y 360 en números, además cada campo tiene su propio *placeholder* con información sobre el tipo de valor que debe ser ingresado en cada campo. Hasta este momento se continua planeando el siguiente paso con la recolección de estos datos.

Para finalizar esta sección de algoritmos, se describirá a grandes rasgos los que realiza el código de Ruby para la conexión a la nube, sin embargo, como dicho código fue realizado por el asesor empresarial, únicamente se hablará sobre su funcionamiento y no se mostrará pseudocódigo de ello.

Este último código se conecta a la red privada de JLL, cuenta con diferentes funciones para establecer una conexión robusta y segura para el envío de grandes cantidades de datos. En este caso el archivo en JSON que contiene la información de los datos del proyecto son procesados de modo que cada dato que se encuentra dentro de ella debe generar un ciclo de “completado”. Para que el código no esté limitado por el internet, se tiene establecido un *timeout* para cada caso en donde se tome un poco más de tiempo de lo esperado a cada dato. Una vez que termina este proceso, se descargan los resultados realizados en la nube con el análisis obtenido en formato zip que incluyen reportes muy detallados sobre cada prueba realizada así también como su estado obtenido.

4. Implementación del Sistema y Pruebas

4.1. Hardware Utilizado

Durante el proceso de firmado de contrato con la empresa para desarrollar este proyecto, el asesor institucional mencionó que se hará llegar una computadora portátil para trabajar en dicho proyecto. De acuerdo con el asesor empresarial, muchos de los empleados de JLL se les proporciona una computadora portátil para llevar a cabo sus trabajos de manera segura y evitar que se haga uso de dispositivo personales que pueden llevar al secuestro y robo de información delicada, por lo que para este proyecto se optó por un equipo seleccionado previamente por dicho asesor.

Se mencionó que este equipo está monitoreado y configurado por la empresa y que cada programa que se deba instalar debe pasar por un *ticket* con el departamento de TI, una vez que esta solicitud sea aprobada, uno de los especialistas de dicho departamento se conectará de manera remota al equipo para instalar dicho software. Si se trata de instalarlo por su cuenta, se debe ingresar contraseña de administrador para ello.

El equipo otorgado cuenta con las siguientes características:

- **Marca:** DELL
- **Procesador:** Intel Core Ultra 7
- **Disco Duro:** 500GB M.2
- **Memoria RAM:** 32GB

Esta computadora fue otorgada para acceder a los recursos de la empresa sin problema alguno, participar en las reuniones y desarrollar el presente proyecto. Al término del contrato con la empresa, la computadora deberá ser regresada a la empresa cuidando que no presente ninguna falla o alteración en su sistema operativo. Esto tomado del contrato firmado por la empresa en donde se responsabiliza del uso adecuado de dicho equipo. En la figura 15, se anexa una imagen de la computadora otorgada por la empresa:

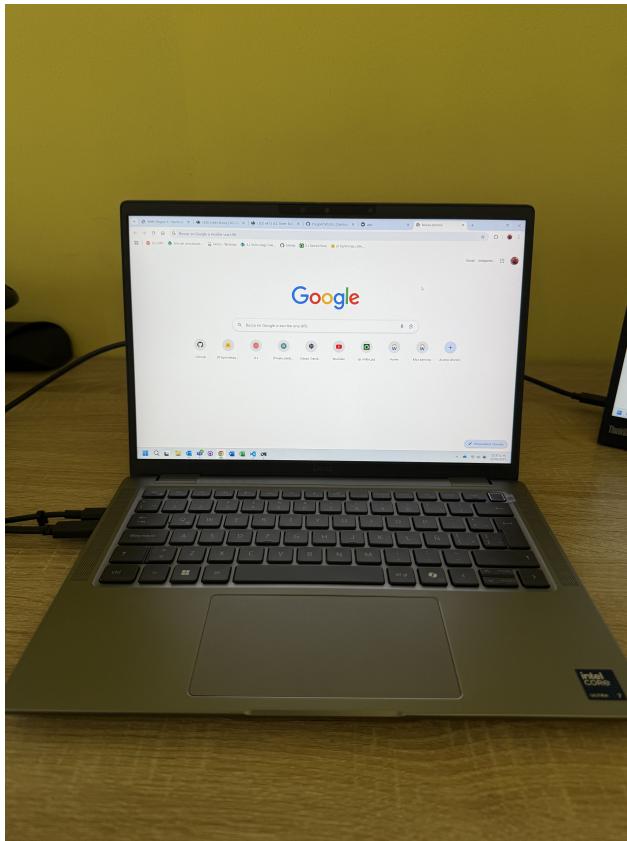


Figura 15: Computadora portátil otorgada por la empresa

4.2. Software Utilizado

Principalmente se escogió como entorno de desarrollo **Visual Studio Code** para programar la herramienta. Además, los lenguajes de programación utilizados fueron **Python** en su versión 3.9.13 y **Ruby** en la versión 3.2.6, se usaron estas versiones debido a la compatibilidad que tienen con las librerías de *OpenStudio* y demás. La herramienta para crear la interfaz gráfica de la herramienta en concreto fue con el framework de *Streamlit*.

En adición a esto, se empleó el software de *OpenStudio* para llevar a cabo la simulación energética. Además, se integró el uso de la API de Google Maps para la geolocalización de la dirección ingresada.

4.3. Elementos a Evaluar

Hasta el momento del desarrollo de la herramienta y con los *scripts* ya creados. Se buscó obtener los datos meteorológicos de ambas fuentes de información (*DOE* y *OneBuilding*) de modo que estos datos sean ejecutados por medio de un *script* de *Python* y sean plasmados en un archivo *CSV*. El resultado que debe arrojar este *script* dentro del archivo *CSV* debe corresponder a la plantilla que se tiene con un recurso similar otorgado por el asesor empresarial, además de no tardar demasiado durante la ejecución de este mismo.

Durante el desarrollo del proyecto, se planteó que el gerente pueda seleccionar entre diferentes estaciones meteorológicas condicionado con un radio de 5 millas sobre la ubicación ingresada, con esto, el gerente puede seleccionar algún tipo de estación que más le convenga para su proyecto dado la ubicación o la fuente de datos que más le funcione.

A su vez, para el llenado de información cada dato de entrada, se busca que únicamente se acepte el tipo de dato que cada *placeholder* está solicitando en sus respectivos campos, al mismo tiempo que cada campo tenga coherencia al momento de llenar los datos, por ejemplo, si se está solicitando un valor tipo numérico, si el gerente trata de ingresar el dato con letras, esto no está permitido y se debe mostrar un mensaje de advertencia sobre esto.

4.4. Pruebas

La primera tarea en desarrollarse fue el *script* de los datos meteorológicos de ambas fuentes de información (*DOE* y *OneBuilding*). Lo primero fue consultar con el asesor empresarial acerca de dónde se puede obtener la información requerida a lo cual se proporcionó dos distintos hipervínculos, uno de la fuente de *OneBuilding* y otro un repositorio para *DOE*.

En la figura 16, se muestran capturas de pantalla de ambas fuentes de datos para la extracción de datos climáticos. En la figura 16a, se muestra una captura de pantalla de la página de *OneBuilding* se cuenta con diferentes pestañas en donde se almacenan datos climáticos de las distintas regiones del mundo en hojas de cálculo. Por otro lado, en la figura 16b, se muestra una captura de pantalla del repositorio de la fuente de *DOE* el cual es un archivo *GeoJSON*.

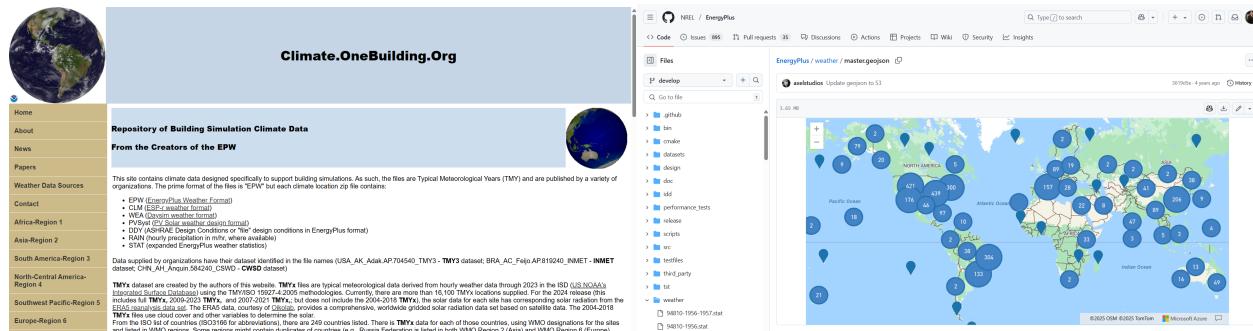


Figura 16: Fuentes de datos meteorológicos

A partir de ambas fuentes, se creó un *script* que recupera la información meteorológica de ambas fuentes en donde son plasmadas en un solo archivo *CSV*. Sin embargo, durante un tiempo se tuvo que hacer mejoras a este *script* debido a una serie de *bugs* en donde algunos encabezados faltaba información. Una vez solucionados estos errores, se agregó dicho *script* al archivo principal de la herramienta en donde con ayuda de la API de Google Maps, se hicieron varias pruebas de entrada de diferentes ubicaciones para obtener los distintos datos

climáticos de las zonas ingresadas, con apoyo del archivo *CSV* que contiene dichos datos climáticos.

Cuando se corre el *script* de *update-weather-data.py* en la terminal se muestran diferentes mensajes con información relevante a considerar de acuerdo a la funcionalidad que se planteó para dicho código. En la figura 17, se anexa el *log* de dicho código. Dentro de el primero extrae toda la información de las fuentes de datos de *OneBuilding* y *DOE*, una vez que llena el archivo con la información recopilada procede a encontrar aquellos datos que se encuentran repetidos en el archivo antes mencionado, para ello muestra explícitamente cuales fueron esos archivos que se repitieron y se borraron durante este proceso, para finalmente mostrar un resultado de datos duplicados eliminados exitosamente.

```
CSV file updated successfully! Check 'resources/epw_weather_data_updated.csv' for the updated data.
Duplicates found:
Value ('603900', 'ALGIERS', 'IWEC', '36.72', '-3.25', '/Africa_wmo_region_1/DZA/DZA_Algeria.603900_IWEC.DZA.zip', 'doe') is duplicated in rows 1, 3047
Value ('725905', 'Climate_Zone_1', 'CTZR2V1', '40.8', '-124.2', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ01RV2/CZ01RV2.zip', 'doe') is duplicated in rows 1067, 1260
Value ('C000080', 'Climate_Zone_02', 'CTZR2V1', '38.4', '-122.7', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ02RV2/CZ02RV2.zip', 'doe') is duplicated in rows 1068, 1261
Value ('724938', 'Climate_Zone_3', 'CTZR2V1', '37.7', '-122.2', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ03RV2/CZ03RV2.zip', 'doe') is duplicated in rows 1069, 1262
Value ('724945', 'Climate_Zone_4', 'CTZR2V1', '37.4', '-122.4', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ04RV2/CZ04RV2.zip', 'doe') is duplicated in rows 1078, 1263
Value ('723940', 'Climate_Zone_5', 'CTZR2V1', '34.9', '-120.4', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ05RV2/CZ05RV2.zip', 'doe') is duplicated in rows 1071, 1264
Value ('722970', 'Climate_Zone_6', 'CTZR2V1', '33.9', '-118.5', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ06RV2/CZ06RV2.zip', 'doe') is duplicated in rows 1072, 1265
Value ('722900', 'Climate_Zone_7', 'CTZR2V1', '32.7', '-117.2', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ07RV2/CZ07RV2.zip', 'doe') is duplicated in rows 1073, 1266
Value ('696140', 'Climate_Zone_8', 'CTZR2V1', '33.6', '-117.9', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ08RV2/CZ08RV2.zip', 'doe') is duplicated in rows 1074, 1267
Value ('722880', 'Climate_Zone_9', 'CTZR2V1', '34.15', '-118.15', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ09RV2/CZ09RV2.zip', 'doe') is duplicated in rows 1075, 1268
Value ('C000086', 'Climate_Zone_10', 'CTZR2V1', '33.88', '-118.27', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ10RV2/CZ10RV2.zip', 'doe') is duplicated in rows 1076, 1269
Value ('723800', 'Climate_Zone_11', 'CTZR2V1', '38.21', '-121.2', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ11RV2/CZ11RV2.zip', 'doe') is duplicated in rows 1077, 1270
Value ('720830', 'Climate_Zone_12', 'CTZR2V1', '38.51', '-121.5', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ12RV2/CZ12RV2.zip', 'doe') is duplicated in rows 1078, 1271
Value ('723800', 'Climate_Zone_13', 'CTZR2V1', '38.51', '-119.71', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ13RV2/CZ13RV2.zip', 'doe') is duplicated in rows 1079, 1272
Value ('C000081', 'Climate_Zone_14', 'CTZR2V1', '35.71', '-117.71', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ14RV2/CZ14RV2.zip', 'doe') is duplicated in rows 1080, 1273
Value ('7u7185', 'Climate_Zone_15', 'CTZR2V1', '32.81', '-115.61', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ15RV2/CZ15RV2.zip', 'doe') is duplicated in rows 1081, 1274
Value ('725957', 'Climate_Zone_16', 'CTZR2V1', '41.3', '-122.31', '/north_and_central_america_wmo_region_4/USA/CA-Zones/CZ16RV2/CZ16RV2.zip', 'doe') is duplicated in rows 1082, 1275
Duplicates removed from 'resources/epw_weather_data_updated.csv' successfully!
```

Figura 17: Log del script *update-weather-data.py*

Esta funcionalidad se ve reflejada en el la interfaz de la herramienta en el campo donde se debe agregar la dirección, en la figura 18, se tiene el resultado obtenido de dicho *script* en donde se extraen datos meteorológicos de manera automática para el llenado de información. Con la finalidad de demostrar esta prueba, se ingresó la ciudad de Chicago en un formato de Ciudad, Estado. Para este campo particular no se limita a este formato en particular, se puede agregar toda una dirección completa de domicilio o simplemente el nombre de la ciudad y los resultados son agregados de manera automática en menos de 1 segundo.

Performance Credit

Project Information

Select a certification type ?

LEED Certification type... ▼

✖ Please select a certification type.

Address
Chicago, IL

Climate Zone
5A

Station Name
Chicago-Midway AP

Data Source
TMY3

Distance to Station
9.18 miles

Figura 18: Datos climáticos de la ciudad de Chicago, Illinois agregados de manera automática con el *script* de datos meteorológicos

Durante este proceso, se pensó en una forma en que el gerente pueda escoger entre diferentes estaciones climáticas basado en un radio de n cantidad de millas sobre la ubicación ingresada. Para llevar a cabo esta idea, se escribió en un *script* en paralelo al proyecto principal, trayendo la funcionalidad de la API de Google Maps para la geolocalización y presentación de dichas estaciones cercanas a la dirección ingresada.

En la figura 19, se tiene el *script* paralelo al proyecto en donde se realizó esta prueba de la estación meteorológica más cercana basado en la dirección ingresada. En este caso, el usuario una vez que ingresó la dirección se expone la estación más cercana dentro de un radio de 5 millas, debajo de este primer resultado está una opción para descargar los datos climáticos, esto se hizo basado en una instrucción que solicitó el asesor empresarial para su uso en un futuro. El otro botón es para mostrar más estaciones meteorológicas que se encuentran más lejos de la estación principal encontrada, en esta prueba se encontró cinco estaciones de las cuales dos perteneces a *OneBuilding* y las otras tres a *DOE*. La idea de este es *script* es poder hacer uso de ella más adelante en el proyecto para que el usuario pueda escoger entre una estación u otra dependiendo del criterio de gerente de proyectos.

Address (Street, City, State, Country, Zip Code)
San Antonio, Texas

Address Information and Closest Weather Station:

Address	Climate Zone	Representative City	Country	Latitude	Longitude	Station ID	Station Name	Distance to Station (miles)	Data Source	Priority	HTTP Link	Host	County	State
San Antonio, Texas	2A	San Antonio	United States	29.4252	-98.4946	T49058	San.Antonio-Brooks.AFB	5.84	SRC-TMYx	5	http://climate.onebuilding.org/WMO_Region_4_North_and_Central_America/USA_L_onebuilding	Bexar County	Texas	

[Open Weather Data for San.Antonio-Brooks.AFB](#)

[Show Other Nearby Stations](#)

Other Nearby Stations:

Station ID	Station Name	Data Source	Host	Distance to Station (miles)	TMYx Version	Priority	HTTP Link
722535	San Antonio-Kelly AFB	TMYx	doe	6.02	TMY3	1	https://energyplus-weather.s3.amazonaws.com/north_and_central_america_wmo_j
722535	San.Antonio-JB.San.Antonio-Kelly.Field Annex	SRC-TMYx	onebuilding	6.07	TMYx-2009-2023	1	http://climate.onebuilding.org/WMO_Region_4_North_and_Central_America/USA_L
722533	San.Antonio-Stinson.Muni.AP	SRC-TMYx	onebuilding	6.09	TMYx-2009-2023	1	http://climate.onebuilding.org/WMO_Region_4_North_and_Central_America/USA_L
722533	San Antonio-Stinson AP	TMY3	doe	6.72	TMY3	1	https://energyplus-weather.s3.amazonaws.com/north_and_central_america_wmo_j
722536	San Antonio	TMY2	doe	7.37	TMY2	1	https://energyplus-weather.s3.amazonaws.com/north_and_central_america_wmo_j

[Open Weather Data for San Antonio-Kelly AFB](#)

[Open Weather Data for San Antonio-JB.San.Antonio-Kelly.Field Annex](#)

[Open Weather Data for San Antonio-Stinson.Muni.AP](#)

[Open Weather Data for San Antonio-Stinson AP](#)

[Open Weather Data for San Antonio](#)

Figura 19: Las 5 estaciones meteorológicas más cercanas a la ciudad de San Antonio, Texas

4.5. Problemas Encontrados y Correcciones

Uno de los problemas que llevó mucho tiempo de desarrollo fue la creación del *script* de los datos climáticos de las fuentes de información de *DOE* y *OneBuilding*.

El primer problema comenzó desde la recolección de datos, una ventaja muy grande que tiene la fuente *OneBuilding* frente a la otra, es la simplicidad de obtener sus datos, esto quiere decir que en la barra de navegación lateral se tienen siete diferentes regiones como se muestra en la figura 20, lo que permite acceder con facilidad a las distintas regiones para obtener sus datos climáticos. El problema apareció debido a que en la región de **North-Central America-Region 4** existen dentro de la vista existen varios archivos de Excel de los cuales tomé el URL de cada uno de ellos para que hiciera el *web scrapping* correspondiente para obtener así los datos climáticos de Norte América. Sin embargo, en dicha información se recopiló datos de otros países además de Estados Unidos, por lo que se tuvo que modificar los URLs para únicamente dejar los correspondientes a Estados Unidos.

The screenshot shows the Climate.OneBuilding.Org website. On the left is a sidebar with a globe icon at the top, followed by a vertical list of links: Home, About, News, Papers, Weather Data Sources, Contact, Africa-Region 1 (highlighted with a red border), Asia-Region 2, South America-Region 3, North-Central America-Region 4, Southwest Pacific-Region 5, Europe-Region 6, and Antarctica-Region 7. The main content area has a blue header "Climate.OneBuilding.Org". Below it is a section titled "Repository of Building Simulation Climate Data From the Creators of the EPW". It contains a bulleted list of file formats: EPW (EnergyPlus Weather Format), CLM (ESP-r weather format), WEA (Daysim weather format), PVsyst (PV Solar weather design format), DDY (ASHRAE Design Conditions or "file" design conditions in EnergyPlus format), RAIN (hourly precipitation in mm/hr, where available), and STAT (expanded EnergyPlus weather statistics). A note below states: "This site contains climate data designed specifically to support building simulations. As such, the files are Typical Meteorological organizations. The prime format of the files is "EPW" but each climate location zip file contains:". At the bottom of the content area, there is a note about TMYx datasets, ERA5 reanalysis data, and cloud cover.

Figura 20: Distintas regiones disponibles de la fuente *OneBuilding.org*

Hasta este momento la fuente de *OneBuilding* se encontraba recolectando los datos climáticos de manera correcta. Sin embargo, cuando se trató de recopilar los datos de la fuente de *DOE* el primer obstáculo que se presentó fue la forma de obtenerlo basado en un archivo *GeoJSON* de un repositorio de GitHub. Se tuvo que investigar la manera de recopilar dicha información proveniente de un repositorio como el anterior mencionado para reunir la información climática.

En la figura 21 ,se muestra una imagen de como se puede obtener los datos climáticos de un punto específico marcado en el mapa del repositorio. A partir de la selección de cualquier punto se despliega información en forma de tabla y varios enlaces para descargar los diferentes tipos de archivos según el usuario necesite. Para la recolección de datos que se solicitó, se encuentra en el encabezado de “all”. Sin embargo, se tuvo otro problema el cual fue que no realizaba correctamente la recolección de los datos de esta fuente, la de *OneBuilding* si lo reunía pero de *DOE* no lo hacía. Este problema perduró por dos semanas de trabajo, debido a la extracción de datos de un repositorio, al término de estas dos semanas e investigación sobre la extracción de estos datos, logró funcionar el *script* de los datos climáticos.

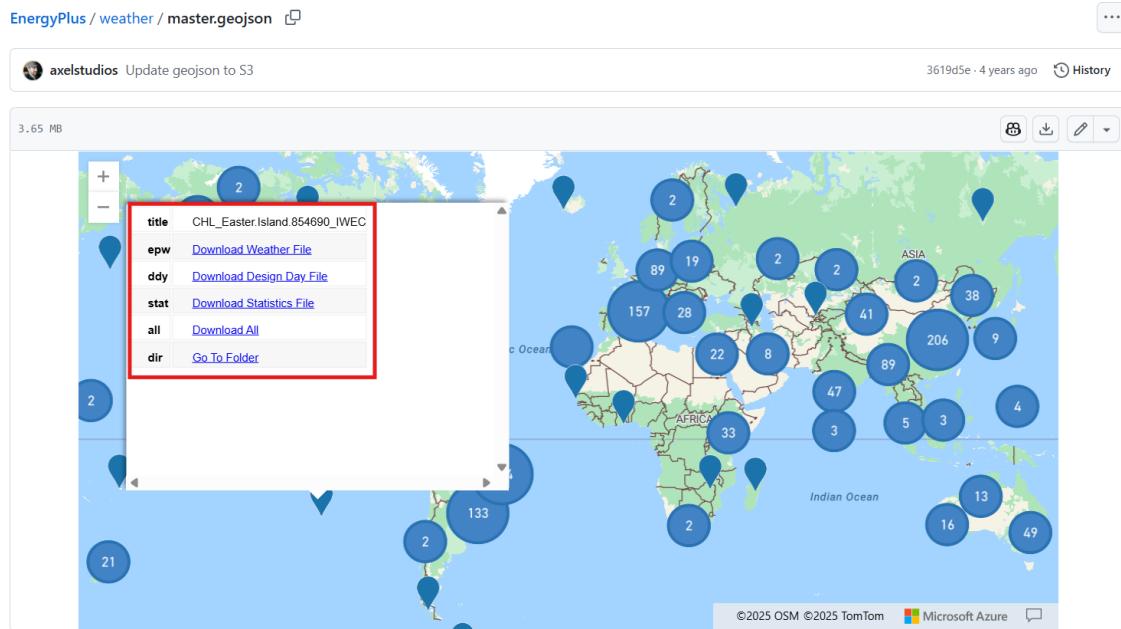


Figura 21: Despliegue de información sobre un punto en el mapa de la fuente *DOE*

Sin embargo, cuando se analizó el resultado arrojado del *script*, se observó que había demasiadas inconsistencias. En la figura 22b se intenta reflejar los errores que se obtuvieron en aquel momento en donde el resultado arrojado muchas veces fallaba el nombre de la estación, el *id* y la fuente de datos. Este problema sucedió debido a la falta de expresiones regulares para la obtención de datos del URL que se estaba recolectando. La solución a este problema fue la implementación de expresiones regulares más robustas para el filtro de datos y su acomodo en sus respectivos encabezados. Dando como resultado el obtenido en la figura 22a, en donde ahora el resultado arrojado del *script* mejoró significativamente el archivo *CSV* de los datos climáticos.

	A	B	C	D	E		A	B	C	D	E		
1	<i>id</i>	station_name	data_source	lat	lon	http_link	1	<i>id</i>	station_name	data_source	lat	lon	http_link
8428	719220	La Ronge AP	CWEC2016	55.15	-105.267	/WMO_Re	1	id	station_name	data_source	lat	lon	http_link
8429	718660	Saskatoon Intl AP	CWEC	52.17	-106.72	/WMO_Re	2416		/USA_UT_Hanksville.AP.724735_TMY3		38.37	-110.72	/north_an
8430	718610	Broadview	CWEC2016	50.368	-102.571	/WMO_Re	2417		USA_UT_Moab-Canyonlands.Field.72_TMY3		38.58	-109.54	/north_an
8431	715560	Last Mountain Lake	CWEC2016	51.417	-105.25	/WMO_Re	2418		USA_UT_Ogden-Hill.AFB.725755_TM1TMY3		41.12	-111.97	/north_an
8432	718660	Saskatoon Intl AP	CWEC2016	52.167	-106.717	/WMO_Re	2419		USA_UT_Ogden-Hinkley.AP.725750_1TMY3		41.2	-112.02	/north_an
									USA_UT_Provo.Muni.AWOS.725724_1TMY3		40.22	-111.72	/north_an

(a) Resultado correcto del *script*

(b) Ejemplo de resultado erróneo del *script*

Figura 22: Resultados arrojados el *script* de recolección de datos climáticos

Otro problema muy particular relacionado a los datos arrojados de la dirección ingresada, fue para la ciudad de Richmond, Virginia. Cuando el usuario ingresa una dirección en particular, uno de los resultados que devuelve esta búsqueda con ayuda de la API de Google Maps, es la zona climática *ASHRAE* que dependiendo de la zona o región tiene un valor como lo puede ser: 4B, 4A, etc. Estos valores son obtenidos dentro de otro archivo *CSV* que ya estaba creado con anterioridad y para dicha ciudad la zona climática tenía un valor de 4A y en su defecto

arrojaba un **NA** lo que significaba que no había una zona climática registrada para dicha localización. El problema de esto estaba relacionado con un código *FIPS* de dicho archivo *CSV*. Este código por alguna razón estaba incorrecto, se buscó en internet el código *FIPS* de dicha ciudad de Virginia, se revisó que ambos códigos coincidieran, al notar que no eran similares, se modificó y finalmente se solucionó este *bug* que tenía dicha ciudad.

En la figura 23 se muestra como aparecía dicho *bug* en la figura 23b remarcado en cuadro rojo y su solución en la figura 23a remarcado en cuadro verde.

Address (Street, City, State, Country, Zip Code)

Address Information and Closest Weather Station:

Address	Climate Zone	Representative City	Country	Latitude
Richmond, VA	4A	Richmond	United States	37.5407

(a) Resultado con zona climática

Address (Street, City, State, Country, Zip Code)

Address Information and Closest Weather Station:

Address	Climate Zone	Representative City	Country	Latitude
Richmond, VA	NA	Richmond	United States	37.5407

(b) Resultado sin zona climática

Figura 23: *Bug* de zona climática de la ciudad de Richmond, Virginia

5. Conclusiones Finales

5.1. Conclusiones sobre el sistema desarrollado

Para este proyecto se enfocó mucho en el desarrollo de las características que la herramienta debe presentar para un correcto funcionamiento. Muchas veces se trabajó en análisis de datos para corroborar que los resultados obtenidos por parte de los *scripts* de Python funcionaran correctamente de modo que la automatización de estos códigos agilizara la manera de trabajar con los datos que se planean analizar en un futuro.

Se resolvieron muchísimos problemas relacionados con *bugs* para los diferentes *scripts* que forman parte del proyecto principal. De los cuales, muchos de ellos fueron en parte por insuficiencia de uso de expresiones regulares para los URLs de las fuentes meteorológicas.

Se trabajó muy por encima la parte de la simulación de los datos ingresados con *OpenStudio* dado que esos códigos forman parte de otro proyecto confidencial de la empresa que depende de varios factores como el uso adecuado de la herramienta *Parametric Analysis Tool (PAT)*, el cálculo de las diferentes combinaciones de valores en cada unidad de medida, etc.

El desarrollo de este proyecto mejoró el llevar a cabo proyectos de análisis de datos por medio de Python y sus diferentes librerías, además de una ayuda significativa al momento de emplear el framework de *Streamlit* que abre la posibilidad de usarlo para diferentes proyectos de inteligencia artificial o mostrar resultados obtenidos de manera rápida y sin tantos problemas para el diseño de páginas web.

5.2. Trabajos y perspectivas para el futuro

Para cuando la herramienta esté terminada al 100 %, se espera que una versión beta sea liberada y monitoreada de manera segura y cerrada en la nube de JLL, para que únicamente los gerentes de proyectos de desarrollo sostenible puedan probar esta herramienta para obtener algunos resultados de sus proyectos con edificios que buscan esta certificación, con una estimación de puntaje obtenido, los gerentes de proyectos podrán ver que campos o alternativas deben cuidar más y/o ponerle más atención para que su edificio logre la certificación LEED más alta posible. Hasta este punto del documento, se ha hecho mención del desarrollo de las características principales que este proyecto busca lograr de modo que en un futuro se aplicarán más características y mejoras para que este proyecto salga a ser probado y recibir una retroalimentación por parte de estos usuarios objetivos para mejorar significativamente la aplicación y logre ser una herramienta que los gerentes de proyectos puedan contar en su día a día.

Durante esta estadiá, muchas veces se trabajó en paralelo al proyecto principal para apoyar al equipo de norteamérica

Como se mencionó con anterioridad, la mayoría de los *scripts* que dan la funcionalidad a ciertas herramientas y automatizan el proceso de la misma, pertenecen a otros proyectos confidenciales de la empresa, los cuales se garantizó el permiso de usarlos para esta herra-

mienta en particular.

Más adelante se plantea que la característica de permitir al gerente de proyectos seleccionar una estación u otra dependiendo de su preferencia para registrarla en la herramienta propuesta. Misma que ayudará a estos gerentes a obtener un resultado simulado del certificado que el edificio en que están trabajando.

Índice de figuras

1. Certificaciones LEED	8
2. Interfaz gráfica de la herramienta OpenStudio PAT	10
3. Software de EnergyPlus	11
4. Uso de sistema HVAC en OpenStudio	12
5. Aplicación de MathGPT hecha con Streamlit	14
6. Aplicación Goodreads Reading Habits hecha con Streamlit	15
7. Aplicación Migration Network hecha con Streamlit	15
8. Diagrama de Arquitectura del Sistema	19
9. Diagrama de Contexto del sistema	20
10. Diagrama de casos de uso	21
11. Diseño de pantalla principal de la herramienta web	22
12. Diseño de pantalla de proyecto con cliente confidencial de JLL	23
13. JSON de entrada representado de manera visual con ayuda de la extensión JSON Crack	24
14. Contenido del archivo ZIP de los diferentes resultados obtenidos del <i>script</i> de Ruby	25
15. Computadora portátil otorgada por la empresa	36
16. Fuentes de datos meteorológicos	37
17. <i>Log</i> del <i>script update-weather-data.py</i>	38
18. Datos climáticos de la ciudad de Chicago, Illinois agregados de manera automática con el <i>script</i> de datos meteorológicos	39
19. Las 5 estaciones meteorológicas más cercanas a la ciudad de San Antonio, Texas	40
20. Distintas regiones disponibles de la fuente <i>OneBuilding.org</i>	41
21. Despliegue de información sobre un punto en el mapa de la fuente <i>DOE</i>	42
22. Resultados arrojados el <i>script</i> de recolección de datos climáticos	42
23. <i>Bug</i> de zona climática de la ciudad de Richmond, Virginia	43

Índice de cuadros

1. Comparación entre eQuest y EnergyPlus	12
2. Contenido del archivo datapoint.zip	25

Índice de algoritmos

1. extract_doe_links()	26
2. extract_links_from_excel(url)	27
3. extract_years(tmyx_version)	27
4. update_csv()	28
5. check_duplicated(file_path)	29
6. delete_duplicates(file_path)	29
7. main()	29
8. process_address_chunk(df_chunk)	30
9. find_closest_weather_stations	31
10. parse_address_string	31
11. address_to_fips_and_coordinates(address_string)	32
12. is_valid_address_format(address_string)	32
13. display_homepage()	33

Referencias

- [1] Ali Amiri, Juudit Ottelin y Jaana Sorvari. “Are LEED-certified buildings energy-efficient in practice?” En: *Sustainability* 11.6 (2019), pág. 1672.
- [2] Sofia Dermisi. “Effect of LEED ratings and levels on office property assessed and market values”. En: *Journal of Sustainable Real Estate* 1.1 (2009), págs. 23-47.
- [3] Peter T Belisle. “Sustainability/climate change”. En: *Strategic planning for energy and the environment* 30.4 (2011), págs. 71-78.
- [4] Emre Caner Akcay y David Ardit. “Desired points at minimum cost in the “Optimize Energy Performance” credit of leed certification”. En: *Journal of Civil Engineering and Management* 23.6 (2017), págs. 796-805.
- [5] Jin Ouk Choi et al. “LEED credit review system and optimization model for pursuing LEED certification”. En: *Sustainability* 7.10 (2015), págs. 13351-13377.
- [6] Sargunam Jebaraj y Sumathy Iniyian. “A review of energy models”. En: *Renewable and sustainable energy reviews* 10.4 (2006), págs. 281-311.
- [7] Lirong Zhou et al. “Energy consumption model and energy efficiency of machine tools: a comprehensive literature review”. En: *Journal of Cleaner Production* 112 (2016), págs. 3721-3734.
- [8] Peter Lopian et al. “A review of current challenges and trends in energy systems modeling”. En: *Renewable and sustainable energy reviews* 96 (2018), págs. 156-166.
- [9] Toshihiko Nakata, Diego Silva y Mikhail Rodionov. “Application of energy system models for designing a low-carbon society”. En: *Progress in Energy and Combustion Science* 37.4 (2011), págs. 462-502.
- [10] Sam Kubba. *LEED practices, certification, and accreditation handbook*. Butterworth-Heinemann, 2009.
- [11] Raymond J Cole y Maria Jose Valdebenito. “The importation of building environmental certification systems: international usages of BREEAM and LEED”. En: *Building research & information* 41.6 (2013), págs. 662-676.
- [12] Ji-Myong Kim, Kiyoung Son y Seunghyun Son. “Green benefits on educational buildings according to the LEED certification”. En: *International Journal of Strategic Property Management* 24.2 (2020), págs. 83-89.
- [13] Peng Wu et al. “A decade review of the credits obtained by LEED v2. 2 certified green building projects”. En: *Building and Environment* 102 (2016), págs. 167-178.
- [14] G Christopher Wedding y Douglas Crawford-Brown. “Improving the link between the LEED green building label and a building’s energy-related environmental metrics”. En: *Journal of Green Building* 3.2 (2008), págs. 85-105.
- [15] Osama Marzouk. “Evolution of the (Energy and Atmosphere) credit category in the LEED green buildings rating system for (Building Design and Construction: New Construction), from version 4.0 to version 4.1”. En: *Journal of Infrastructure, Policy and Development* 8.8 (2024), págs. 10-24294.
- [16] Michelle Cottrell. *Guidebook to the LEED Certification Process: For LEED for New Construction, LEED for Core and Shell, and LEED for Commercial Interiors*. Vol. 2. John Wiley & Sons, 2011.
- [17] Tony Robinson. “Operations & Maintenance”. En: *High-Performance Buildings*. River Publishers, 2020, págs. 341-370.

- [18] Pedro Morales et al. “Qualitative analysis of sustainable vertical dwelling in Nuevo Leon, Mexico”. En: *Journal of Green Building* 6.1 (2011), págs. 58-72.
- [19] Salman Azhar et al. “Building information modeling for sustainable design and LEED® rating analysis”. En: *Automation in construction* 20.2 (2011), págs. 217-224.
- [20] Rob Guglielmetti, Dan Macumber y Nicholas Long. “OpenStudio: an open source integrated analysis platform”. En: (2011).
- [21] David Goldwasser et al. “The life cycle of an openstudio measure: Development, testing, distribution, and application”. En: *ASHRAE & IBPSA-USA SimBuild Conference 2016*. Vol. 7. ASHRAE/IBPSA-USA. 2016, págs. 222-229.
- [22] Andrew Parker et al. “A parametric analysis tool for building energy design workflows: Application to a utility design assistance incentive program”. En: *ACEEE summer study on energy efficiency in buildings*. Vol. 12. 2014.
- [23] Ajla Aksamija. “Methods for integrating parametric design with building performance analysis”. En: *ARCC Conference Repository*. 2018.
- [24] Joana Sousa. “Energy simulation software for buildings: review and comparison”. En: *International Workshop on Information Technology for Energy Applications-IT4Energy, Lisbon*. Citeseer. 2012, págs. 1-12.
- [25] Kristen S Cetin et al. “Development and validation of an HVAC on/off controller in EnergyPlus for energy simulation of residential and small commercial buildings”. En: *Energy and Buildings* 183 (2019), págs. 467-483.
- [26] T Agami Reddy. “Literature Review on Calibration of Building Energy Simulation Programs: Uses, Problems, Procedures, Uncertainty, and Tools.” En: *ASHRAE transactions* 112.1 (2006).
- [27] Nelson Fumo, Pedro Mago y Rogelio Luck. “Methodology to estimate building energy consumption using EnergyPlus Benchmark Models”. En: *Energy and Buildings* 42.12 (2010), págs. 2331-2337.
- [28] Hema Sree Rallapalli. *A comparison of EnergyPlus and eQUEST whole building energy simulation results for a medium sized office building*. Inf. téc. Arizona State University, 2010.
- [29] Ivet Challenger-Pérez, Yanet Díaz-Ricardo y Roberto Antonio Becerra-García. “El lenguaje de programación Python”. En: *Ciencias Holguín* 20.2 (2014), págs. 1-13.
- [30] Clayton Miller, Christian Hersberger y Marcus Jones. “Automation of common building energy simulation workflows using Python”. En: (2013).
- [31] Leon Thurner et al. “pandapower—an open-source python tool for convenient modeling, analysis, and optimization of electric power systems”. En: *IEEE Transactions on Power Systems* 33.6 (2018), págs. 6510-6521.
- [32] Andrew Lyden, Graeme Flett y Paul G Tuohy. “PyLESA: A Python modelling tool for planning-level Local, integrated, and smart Energy Systems Analysis”. En: *SoftwareX* 14 (2021), pág. 100699.
- [33] Mohammad Khorasani, Mohamed Abdou y J Hernández Fernández. “Web Application Development with Streamlit”. En: *Software Development* (2022), págs. 498-507.
- [34] Tyler Richards. *Streamlit for Data Science: Create interactive data apps in Python*. Packt Publishing Ltd, 2023.
- [35] David Flanagan y Yukihiro Matsumoto. *The Ruby Programming Language: Everything You Need to Know*. O'Reilly Media, Inc., 2008.

- [36] Amir Roth et al. "Scripting Frameworks For Enhancing EnergyPlus Modeling Productivity". En: *2018 Building Performance Analysis Conference and SimBuild*. Vol. 8. ASHRAE/IBPSA-USA. 2018, págs. 312-319.
- [37] Elaine Hale et al. *Flexible Framework for Building Energy Analysis*. Inf. téc. National Renewable Energy Lab.(NREL), Golden, CO (United States), 2012.