



Escuela Politécnica Nacional

Métodos Numéricos

Proyecto Bimestral

Seguidor solar con dos grados de libertad

Integrantes:

David Ignacio Egas Solís

Aubertin Leandro Ochoa Vélez

Erick Ezequiel Romero Basantes

Alexis Fernando Bautista Mino

Profesor:

Jhonathan Zea

12 de febrero del 2025

Índice

Índice	2
1. Objetivos.....	3
1.1. Objetivo General.....	3
1.2. Objetivos Específicos	3
2. Introducción.....	3
3. Metodología.....	4
3.1. Solución del problema	4
3.2. Desarrollo matemático.....	4
3.3 Diagrama de flujo	8
4. Análisis y Resultados.....	8
4.1. Validaciones.....	8
4.2. Pruebas de simulación	10
Prueba 1	10
Prueba 2	11
Prueba 3	12
5. Conclusiones.....	12
6. Recomendaciones	13

Figura 1 Ubicación inicial del panel con respecto al suelo	4
Figura 2 Sistema de referencia	5
Figura 3 Sistema P' proyectado en P	5
Figura 4 Sistema S' proyectado en S	6

1. Objetivos

1.1. Objetivo General

Desarrollar un programa en Python con una interfaz gráfica que simule el funcionamiento de un seguidor solar de dos grados de libertad (pitch y roll). El programa calculará en tiempo real la posición óptima del panel solar en función de la elevación y el azimut del sol, considerando como ubicación de referencia la Escuela Politécnica Nacional. La simulación permitirá visualizar la trayectoria del sol y el ajuste dinámico del panel para maximizar la captación de luz solar.

1.2. Objetivos Específicos

- Desarrollar el modelo matemático que permita calcular los ángulos de control pitch y roll del panel solar a partir de la posición solar (elevación θ y azimut α).
- Implementar un programa en Python para calcular y simular en tiempo real la orientación del panel solar en función de la posición del sol.
- Diseñar una interfaz gráfica que permita al usuario ingresar la fecha, hora y duración de la simulación, mostrando como cambia la orientación del panel solar a lo largo del tiempo.
- Visualizar la trayectoria del sol y la respuesta del panel solar mediante gráficos dinámicos que permitan analizar su comportamiento a lo largo del día.
- Asegurar que el panel solar se oriente de manera óptima, manteniéndose perpendicular a la luz solar durante todo momento durante la simulación.

2. Introducción

La energía solar se ha convertido en una de las fuentes de energía renovable más importantes en la actualidad debido a su disponibilidad y bajo impacto ambiental. Sin embargo, para maximizar su eficiencia, es fundamental optimizar la captación de la luz solar a lo largo del día. En este contexto, los seguidores solares desempeñan un papel clave, ya que permiten ajustar la orientación de los paneles solares de manera dinámica para mejorar la absorción de la radiación solar.

Este informe presenta el desarrollo de una simulación en Python para un seguidor solar de dos grados de libertad (pitch y roll), diseñado para ajustar automáticamente la posición del panel en función de la elevación y el azimut del sol. La ubicación de referencia para la simulación será la Escuela Politécnica Nacional, permitiendo obtener resultados realistas basados en coordenadas geográficas específicas.

El programa desarrollado integra un modelo matemático que calcula los ángulos de inclinación necesarios para optimizar la orientación del panel solar. Además, cuenta con una interfaz gráfica, a través de la cual el usuario puede ingresar la fecha, la hora y la

duración de la simulación. La visualización dinámica de la trayectoria del sol y la respuesta del panel facilitarán el análisis del comportamiento del sistema a lo largo del día.

3. Metodología

3.1. Solución del problema

El objetivo de este proyecto es implementar y desarrollar un seguidor solar con dos grados de libertad, realizando su simulación a lo largo del día. Para lograr esto, el sistema necesita calcular los ángulos de giro (*roll* y *pitch*), que son los encargados de direccionar el panel y ajustar su posición. Estos cálculos se basan en los ángulos azimutal y de elevación, que describen la trayectoria del sol a lo largo del día buscando maximizar la generación de energía eléctrica mediante el posicionamiento perpendicular del panel solar con respecto a los rayos del sol durante todo el día.

3.2. Desarrollo matemático

Para este proyecto es importante saber que partimos de un sistema de tres ejes que es paralelo al suelo (Figura 1) cuando todavía no ha sufrido algún tipo de cambio o rotación. Una vez el sistema haya sido rotado de alguna manera este sistema de referencia pasa a ser un nuevo sistema de ejes coordenados P' .

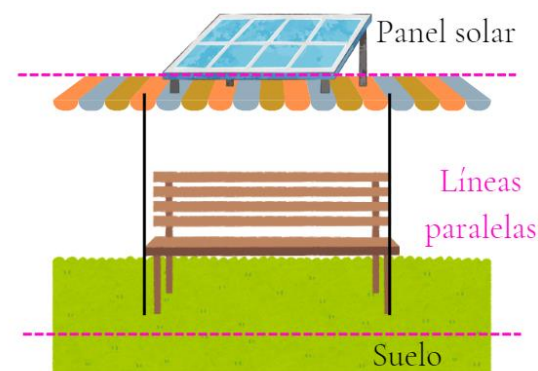


Figura 1 Ubicación inicial del panel con respecto al suelo

Este análisis se realiza desde la perspectiva y ubicación del panel, utilizando su longitud y latitud. Para analizar las rotaciones del panel, es necesario emplear matrices de rotación. Estas matrices nos permitirán calcular la posición del panel basándonos en los ángulos de elevación y azimut, determinando así los ángulos de movimiento pitch y roll, además de aplicar conceptos como el vector normal y el vector director de un plano.

Para determinar estos elementos primero definimos el sistema de referencia que usaremos para calcular dichos ángulos y que usaremos por convención a lo largo del desarrollo matemático.

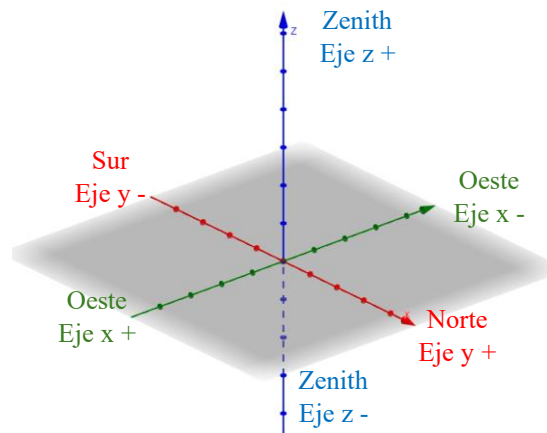


Figura 2 Sistema de referencia

Una vez definido dicho sistema de referencia, procedemos a calcular la matriz de rotación del ángulo *roll* que se obtiene al proyectar el giro del nuevo sistema llamado P' dentro del sistema original P.

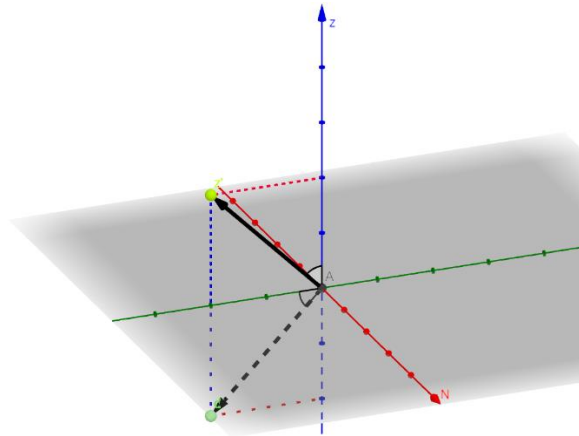


Figura 3 Sistema P' proyectado en P

Con la ayuda de la gráfica (Figura 3) podemos decir que la matriz asociada al giro es:

$$R_{\text{roll}} = \begin{pmatrix} \cos(\text{roll}) & 0 & \text{sen}(\text{roll}) \\ 0 & 1 & 0 \\ -\text{sen}(\text{roll}) & 0 & \cos(\text{roll}) \end{pmatrix}$$

Donde cada una de las columnas es la proyección de cada uno de los ejes del sistema P y girada confirmando el nuevo sistema P'.

Ahora para el cálculo del ángulo *pitch* tomamos un nuevo sistema de ejes coordenados llamado S de donde partiremos a realizar el cálculo de la matriz correspondiente a dicho ángulo y pasarlo al sistema S'.

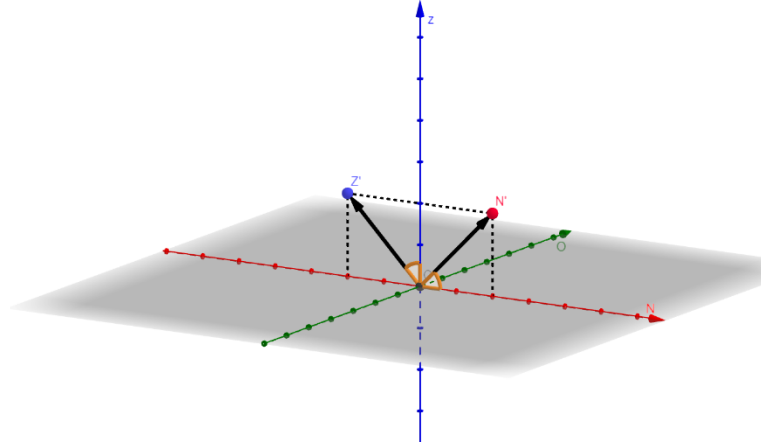


Figura 4 Sistema S' proyectado en S

Una vez realizada la proyección de los vectores obtenemos la matriz de rotación asociada al ángulo pitch como se muestra a continuación:

$$R_{Pitch} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\text{pitch}) & -\text{sen}(\text{pitch}) \\ 0 & \text{sen}(\text{pitch}) & \cos(\text{pitch}) \end{pmatrix}$$

Cabe mencionar que el cálculo de las matrices de rotación se lo realizó de manera separada ya que durante el desarrollo pudimos constatar que llegamos a la misma solución obteniendo las matrices por separado que tomando como base el sistema P' y posteriormente sobre ese sistema aplicar la proyección y obtener el sistema S'.

Una vez obtenidas estas matrices para el panel procedemos a realizar la multiplicación de cada una de estas matrices de la siguiente manera:

Primera operación: Matriz de rotación R_{Roll} y la matriz correspondiente a un vector cualquiera del plano P:

$$\begin{pmatrix} \cos(\text{roll}) & 0 & \text{sen}(\text{roll}) \\ 0 & 1 & 0 \\ -\text{sen}(\text{roll}) & 0 & \cos(\text{roll}) \end{pmatrix} * \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} X\cos(\text{roll}) + Z\text{sen}(\text{roll}) \\ y \\ -X\cos(\text{roll}) + Z\cos(\text{roll}) \end{pmatrix}$$

Tomamos la matriz resultado y la multiplicamos por la matriz de R_{pitch} para unificar el sistema en uno solo en los que se dependa de los ángulos *pitch* y *roll*.

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\text{pitch}) & -\text{sen}(\text{pitch}) \\ 0 & \text{sen}(\text{pitch}) & \cos(\text{pitch}) \end{pmatrix} * \begin{pmatrix} X\cos(\text{roll}) + Z\text{sen}(\text{roll}) \\ y \\ -X\cos(\text{roll}) + Z\cos(\text{roll}) \end{pmatrix} \\ = \begin{pmatrix} X\cos(\text{roll}) + Z\text{sen}(\text{roll}) \\ X\text{sen}(\text{roll})\text{sen}(\text{pitch}) + Y\cos(\text{pitch}) - Z\text{sen}(\text{pitch})\cos(\text{roll}) \\ -X\text{sen}(\text{roll})\cos(\text{pitch}) + Y\text{sen}(\text{pitch}) + Z\cos(\text{roll})\cos(\text{pitch}) \end{pmatrix}$$

Una vez ya obtenida la matriz resultante de multiplicar cada una de las matrices procedemos a remplazar los valores del vector director $z = (0,0,1)$. Esto se realiza ya que durante el calculo de las matrices de rotación pudimos observar que el eje que esta involucrado dentro de todos los giros es el vector de las *zetas* donde por tal motivo elegimos a este como nuestro vector director en el movimiento del panel. Además, esto posee sentido ya que lo que queremos es tener perpendicularidad con la dirección en la que se encuentra el sol para maximizar la producción de energía y el único eje con dirección al cielo es dicho vector.

$$Z^*M_{\text{resultante}} = \begin{pmatrix} \text{sen}(\text{roll}) \\ -\text{sen}(\text{pitch})\cos(\text{roll}) \\ \cos(\text{roll})\cos(\text{pitch}) \end{pmatrix}$$

Una vez obtenido como resultado la matriz de rotación con respecto al vector director z procedemos a calcular la matriz de rotación con respecto al sol.

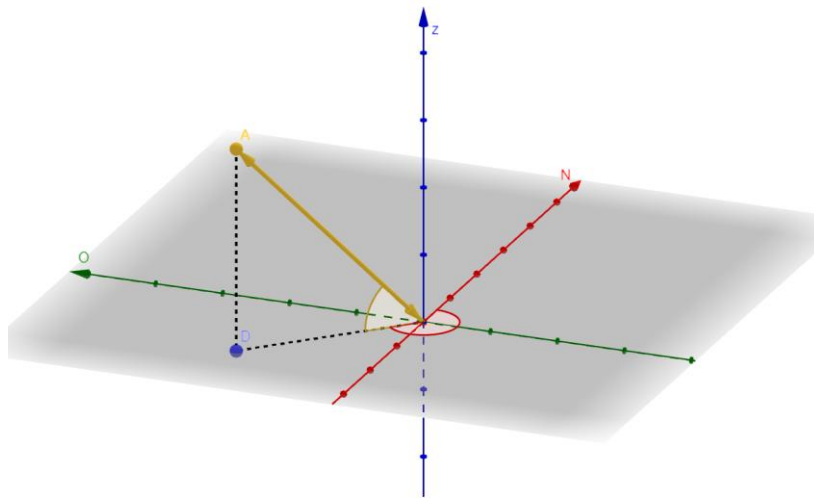


Figura 5. Sistema original

$$M_{sol} = \begin{pmatrix} \cos(el) \sin(az) \\ \cos(az) \cos(el) \\ \sin(el) \end{pmatrix}$$

Una vez obtenido la matriz de rotación en base al sistema original considerando los ángulos de elevación y azimutal lo igualamos al sistema de rotación del panel solar para obtener el valor de los ángulos de pitch y roll en base al comportamiento de los ángulos azimutal y elevación.

$$\begin{pmatrix} sen(roll) \\ -sen(pich)cos(roll) \\ cos(roll)cos(pitch) \end{pmatrix} = \begin{pmatrix} cos(el)sen(az) \\ cos(az)cos(el) \\ sen(el) \end{pmatrix}$$

De donde finalmente tenemos que el ángulo roll y pitch son:

$$\text{Roll} = \sin^{-1}(\cos(el)\sin(az))$$

$$\text{Pitch} = \sin^{-1}\left(\frac{-\cos(el)\cos(az)}{\cos(roll)}\right)$$

3.3 Diagrama de flujo

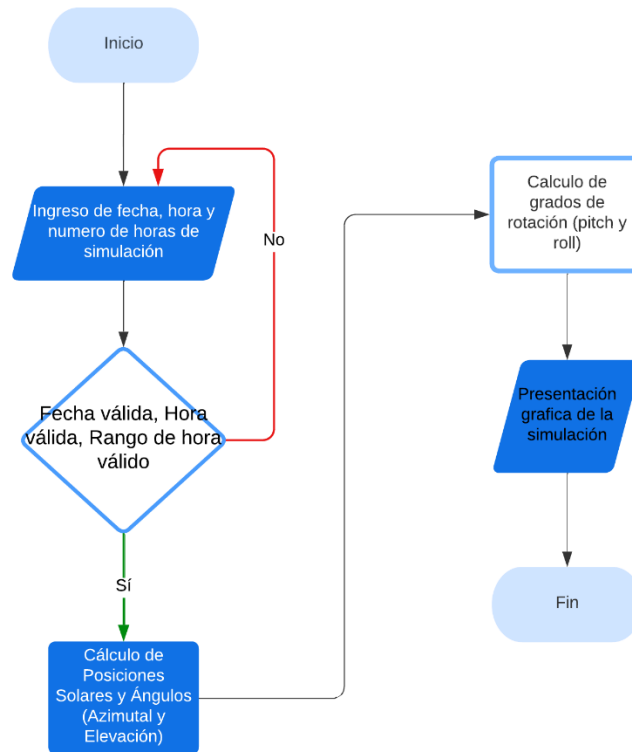


Figura 6. Diagrama de Flujo

4. Análisis y Resultados

4.1. Validaciones

1. Cuando el usuario deja los campos vacíos, se despliega un error el cual alerta al usuario que debe ingresar todos los campos requeridos



Figura 7. Validación 1

2. Cuando el usuario ingresa mal un campo, por ejemplo, si se ingresa una hora que no existe o en otro formato se despliega un mensaje de error que dice “Fecha u hora incorrecta. Intente de nuevo.”



Figura 8. validación 2

3. Supongamos que un usuario ingresa como hora las 4:00 sin embargo se sabe que el sol a estas horas aun no es visible en la localización que planteamos anteriormente, por lo tanto, el programa muestra una alerta en la que se observa “El sol no es visible a estas horas. Intente en un rango de 6:00 a 18:00.”



Figura 9. validación 3

4. Si un usuario ingresa una hora, como, por ejemplo, las 16:00 y coloca el tiempo de simulación de 3 horas, la simulación únicamente se hará por dos horas, ya que el rango donde es visible es sol esta definido entre las 6:00 y 18:00 horas, esto desplegará un mensaje para el usuario como el siguiente: “La simulación se hará por dos horas dado que el sol a las 18 horas estará oculto.”



Figura 10. validación 4

4.2. Pruebas de simulación

Prueba 1

Para la primera prueba se ingresa como datos, la fecha del día actual, hora 6:00 (apenas el sol es visible), tiempo de simulación 12 horas, se escogen estos datos ya que así se

puede observar la simulación de un día completo es decir de 6:00 horas a 18:00 horas. Una vez iniciada la simulación se puede comprobar como es el movimiento del sol y del panel a lo largo del día, también se puede observar la trayectoria de estos, además del rayo de sol.

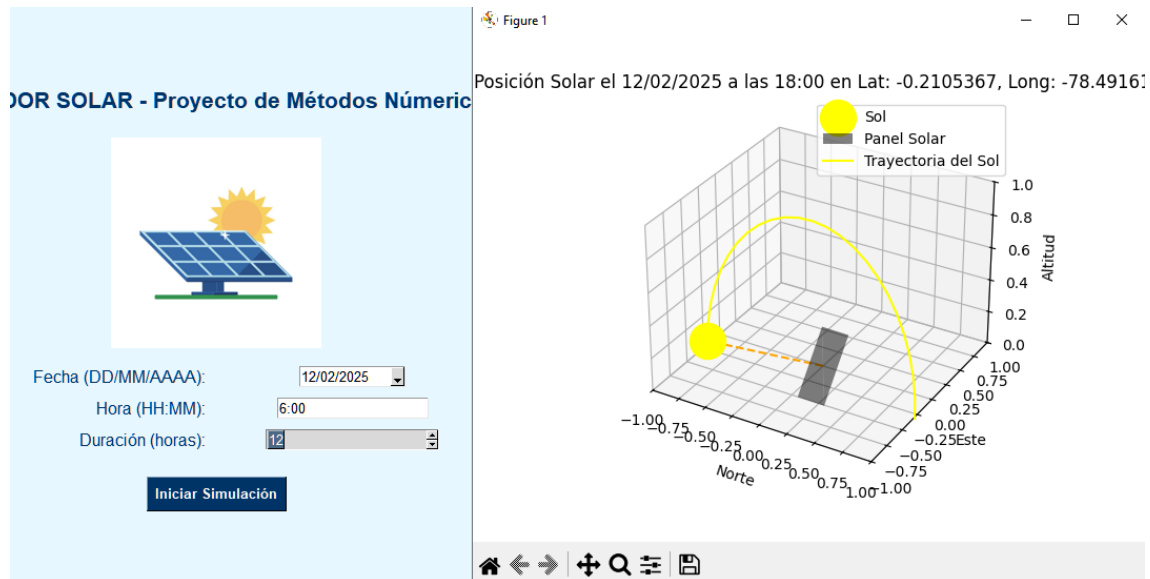


Figura 11 Prueba 1

Prueba 2

Para esta prueba se tiene como datos, fecha del día actual, hora 6:00, y un tiempo de simulación de 6 horas. Con esto se espera que el sol este en su punto mas alto o aproximado cuando la simulación acabe, con ello se comprueba que la simulación se esté realizando de manera correcta.

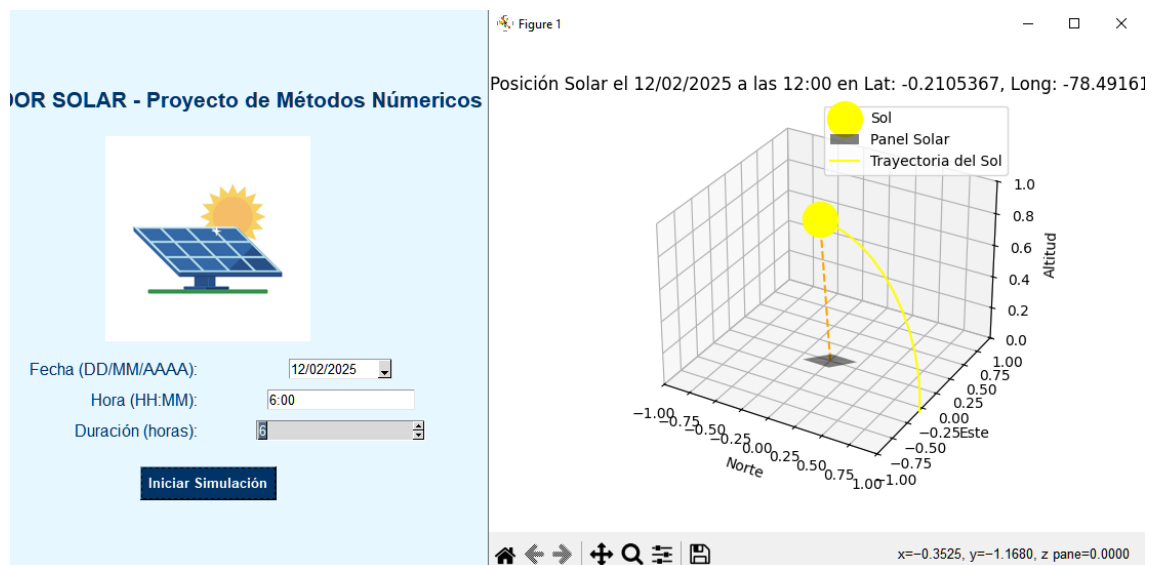


Figura 12. Prueba 2

Prueba 3

En este caso se comprueba la cuarta validación, para ello se ingresa las 16:00 horas y se coloca el tiempo de simulación de 3 horas, entonces se comprueba que la simulación únicamente se realiza por dos horas, dado que el sol ya no es visible.



Figura 13. Prueba 3

5. Conclusiones

- Se logró desarrollar un modelo matemático preciso para el cálculo de los ángulos pitch y roll, permitiendo que el panel solar mantenga una orientación óptima en función de la posición del sol. El uso de matrices de rotación resultó ser una estrategia eficaz para el ajuste dinámico del panel.
- El programa desarrollado en Python cumple con los objetivos planteados, logrando simular en tiempo real la orientación del panel solar. La integración de una interfaz gráfica interactiva facilita el ingreso de parámetros y la visualización de la trayectoria solar.
- A través de diferentes pruebas y validaciones, se comprobó que el sistema responde correctamente a variaciones en la fecha, hora y duración de la simulación, asegurando que el panel se mantenga perpendicular a la luz solar en todo momento.
- Se identificaron restricciones en la visibilidad del sol según la ubicación geográfica y el horario, lo que se reflejó en la simulación mediante mensajes de advertencia al usuario. Esto refuerza la importancia de contar con umbrales de validación en sistemas automatizados.

6. Recomendaciones

- Aunque la simulación está basada en un seguidor solar de la Escuela Politécnica Nacional, sería interesante incluir la opción de ingresar coordenadas para probar cómo varía la orientación del panel en diferentes partes del mundo.
- Se pueden explorar algoritmos más avanzados para mejorar la precisión en el cálculo de los ángulos de orientación, considerando factores como la refracción atmosférica y variaciones estacionales.
- Incorporar datos en tiempo real sobre nubosidad o intensidad de la radiación solar permitiría una simulación más realista y útil para aplicaciones prácticas.
- Incorporar datos en tiempo real sobre nubosidad o intensidad de la radiación solar permitiría una simulación más realista y útil para aplicaciones prácticas.
- Permitir la generación de reportes o gráficos exportables facilitaría el análisis posterior de las simulaciones realizadas.
- Validar los resultados de la simulación con mediciones reales de un seguidor solar físico ayudaría a evaluar la precisión del modelo implementado.