

# ALFA CENTAURE

Juan Pablo Agudelo Barajas, Víctor Daniel Alvarez Valencia, Diego Iván Perea Montealegre  
[victor.alvarez@uao.edu.co](mailto:victor.alvarez@uao.edu.co), [diego.perea@uao.edu.co](mailto:diego.perea@uao.edu.co), [juan\\_pab.agudelo@uao.edu.co](mailto:juan_pab.agudelo@uao.edu.co),

Departamento de Automática y Electrónica  
Universidad Autónoma de Occidente  
Santiago de Cali, Valle del Cauca

*Todos los días estamos expuestos a los rayos UVB y UVA, e incluso si queremos escapar de la intensidad del sol en lugares abiertos como centros comerciales o parques a través de sombrillas de mesa. Esto no siempre será posible ya que con el paso de las horas el sol completa su trayectoria, haciendo disfuncional en pocas horas hasta una sombrilla de mesa estática, esto es perjudicial para la piel ya que provoca un rápido envejecimiento y cáncer de piel. Por lo tanto, la idea de solucionar este problema se implementó con el desarrollo de una sombrilla robot que sigue la trayectoria del sol, de manera que brinda sombra en todas las direcciones con respecto al movimiento del sol.*

**Palabras clave**— sombrilla;sol;robot;rayos;rviz;gazebo

## I. INTRODUCCIÓN

Durante el desarrollo de la robótica, este ha superado grandes desafíos que varios años atrás se creían imposibles de superar, pero una de las cosas de la que la robótica está más orgullosa es el hecho de poder cambiar las cosas cotidianas para ser transformadas en objetos con múltiples funcionalidades que abren nuevos campos en la comodidad, estudio, salud entre otros.

Con esta finalidad se busca implementar un diseño robótico, para una situación de la vida diaria, un problema que ocurre que no muchas personas buscan solucionar o no le dan importancia, y es que este tipo de problemas son los que más consecuencias generan debido al mismo abandono o desinterés de las personas para su solución, así en este documento se busca el análisis, el estudio y la solución de un problema cotidiano aplicando todos los conocimientos vistos en la asignatura de Robótica, concluyendo con un robot que pueda satisfacer las necesidades de las personas y finalizando de que los conocimientos adquiridos fueron aplicados correctamente.

Para esto se realiza el análisis de distintos entornos donde se presenten un problema esto mayormente inspirado en las situaciones que ocurren en el campus de la Universidad Autónoma de Occidente, para posteriormente realizar un estudio más cuidadoso enfocados a un mercado local, generando así una solución con aplicación en robótica

## II. MARCO TEÓRICO

### A. Luz solar

La luz solar consiste de un fenómeno de espectro de rayos de diferente longitud de onda, así la luz visible tiene una longitud de onda de 400 a 700 nm, así la luz ultravioleta o UV tiene una longitud incluso más diminuta llegando a ser de 280 a 400 nm y por último se tiene la luz infrarroja invisible tiene un longitud de onda mayor a 700 nm a 1 nm, por lo que las longitudes de onda más largas, la luz visible e infrarroja son capaces de penetrar más profundamente en la piel. En resumen en el sentido más amplio de la palabra la luz solar es el espectro total de la radiación electromagnética proveniente del sol, esto ocurre durante las horas en la que el sol apunta directamente a la tierra en el ciclo conocido como día.

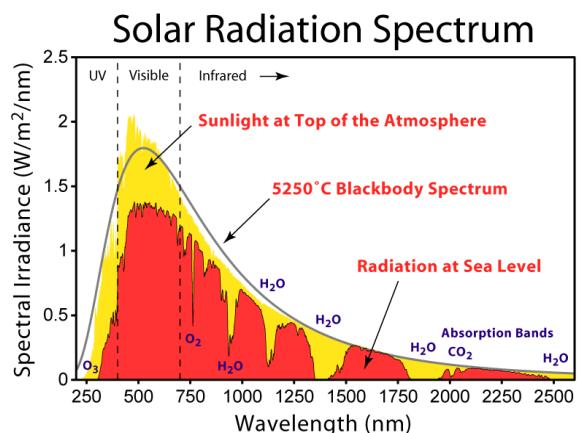
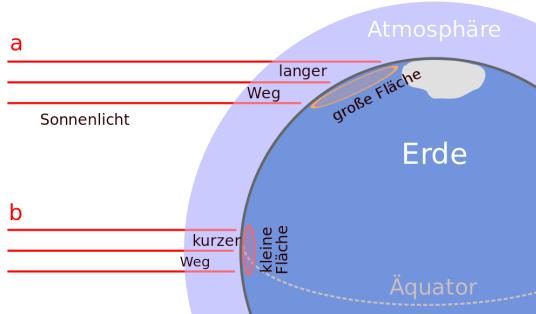


Imagen 1. Espectro de radiación solar

Esta luz solar genera a su vez radiación térmica, que es producida directamente por la radiación del sol es diferente del incremento de la temperatura atmosférica debido al calentamiento radiactivo de la atmósfera por la radiación solar.



**Imagen 2. Reflexión de la luz solar en la tierra**

La luz solar de manera directa puede proporcionar alrededor de 93 lumens de iluminación por vatio de potencia electromagnética, incluyendo infrarrojo, visible y ultravioleta, eso genera un factor fundamental en el proceso de fotosíntesis, tan importante para la vida.

#### B. Sombrilla



**Imagen 3. Sombrillas tradicionales**

La sombrilla es un objeto de uso manual que varía en tamaños siendo para una persona que es el tamaño más pequeño o uno de gran tamaño que abarca a múltiples personas. Una sombrilla se diferencia de un paraguas puesto que esta suele ser más grande, así su mayor uso es de protegernos contra el sol como contra la lluvia, siendo el elemento más representativo en lugares con espacios abiertos como playas o en sitios donde haya una mayor presencia del sol.

Este objeto se dice que es de origen chino, ya que se relata en pinturas que remontan a unos 4.000 años atrás, tanto en textos escritos como en pinturas, siendo así la

civilización oriental como el mayor potenciador de la cultura del uso de la sombrillas, representando inicialmente una simbología social, siendo así que en el siglo IX era usada de manera exclusiva en las clases altas.

Dentro del mundo de las sombrillas estas se encuentran fabricadas de distintos materiales resistentes, siendo el material principal el plástico o de lona, mientras que su estructura consta de infinita cantidad de materiales que dependen del lugar y del uso que se le desea dar al objeto.

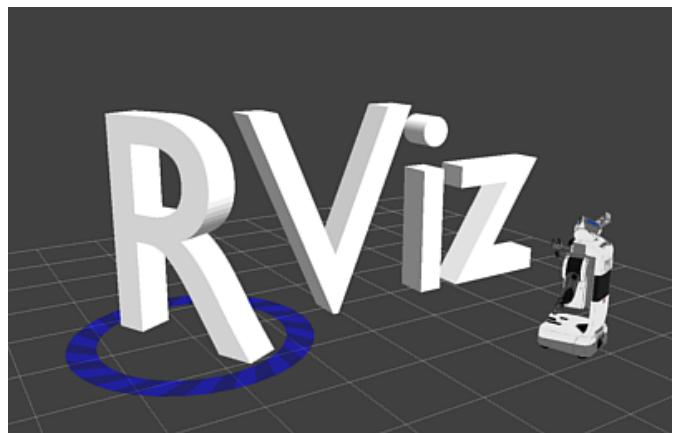
#### C. Gazebo



**Imagen 4. Logo de Gazebo**

Gazebo es un simulador de múltiples robots con una orientación de programación abierta, siendo capaz de simular una serie de robots, sensores y objetos generando un mundo tridimensional, siendo el punto más fuerte de este simulador, la capacidad de poder generar movimiento y comunicación de los datos, pudiendo así conectar en la vida real diferentes sensores e instrumentos para poder ver su comportamiento en el entorno de simulación.

#### D. Rviz



**Imagen 5. Interfaz de RViz**

Rviz es una interfaz gráfica de ROS que permite la visualización de varios tipos de información, desde modelos físicos hasta la generación de datos de manera física, en este trabajo los complementos más importantes serán:

- Modelo de Robot: Este complemento permite la visualización del modelo del robot de acuerdo con su descripción del modelo URDF que se le proporcione.

- TF: Este complemento permite la visualización de la posición y orientación de todos los marcos que componen la jerarquía de TF.

### **III. OBJETIVO**

El objetivo principal de este documento es la capacidad de identificar y analizar problemas de ingeniería generando una solución a partir de los conocimientos en el campo de la robótica.

Ahora las competencias que se desea que el estudiante desarrolle son:

- Habilidad para identificar, formular y resolver problemas complejos de ingeniería aplicando principios de ingeniería, ciencias y matemáticas
- Habilidad para adquirir y aplicar nuevo conocimiento cuando es requerido, usando estrategias apropiadas de aprendizaje

### **IV. DESCRIPCIÓN DEL PROBLEMA**

La universidad autónoma de occidente se caracteriza por poseer varias zonas verdes, estas están acompañadas por mesas y sombrillas para el sol, pero se analizado que tras el transcurso del día a medida de que el sol va cruzando el horizonte, la mayoría de las sombrillas se vuelven inútiles puesto que no tiene el ángulo para poder cubrir la mesa, dejando solo con sombra en ocasiones una o incluso ninguna silla, pero esto qué tipo de impacto puede llegar a tener?

**MOTIVACIÓN:** Todos tendremos que exponernos al sol, de hecho nos exponemos diariamente ya sea para ir a estudiar, trabajar, pasear o simplemente para regar las plantas o sacar el perro, constantemente estamos siendo bombardeados por esta luz solar y esto no significa que sea malo, de hecho todos necesitamos exponernos al sol en alguna medida, ya que gracias a esta luz somos capaces de fabricar vitamina D que ayuda al organismo a absorber el calcio.

Pero la luz del sol también puede ser peligrosa, ya que el exceso de exposición de los rayos ultravioletas pueden generar lesiones en la piel, ojos, debilitar el sistema inmunológico e incluso causar cáncer de piel. Además de esto durante los últimos años se ha generado una tendencia en el incremento del ingreso de los rayos ultravioleta al planeta tierra, ocasionado por diferentes factores como la contaminación, esto provoca que incluso los jóvenes de 20 años para abajo pueden desarrollar cáncer de piel. Las personas de piel más blanca son las más susceptibles de quemarse bajo el sol, ya que el pigmento de la piel se vuelve más oscura apareciendo arrugas y manchas, aumentando las posibilidades de un cáncer de piel. Por esta razón, en los sitios públicos como Centros Comerciales, Parques, restaurantes, centros culturales etc. Poseen lugares que

ayudan al resguardo de las personas ante los rayos solares, uno de los métodos más usados son las sombrillas de mesa ya que son relativamente económicas y cubren entre 2 a 4 personas, pero como se mencionó anteriormente, esta solución no es 100% efectiva generando aún altos riesgos de sufrir insolaciones o quemaduras, al estar bajo de sombrillas que no cumplan con su principal objetivo de diseño.

#### **¿Qué beneficios se espera de este proyecto?**

Se desea obtener un espacio abierto al aire libre el cual se encuentre protegido de los rayos UV cubierto con una sombra la cual no debe desplazarse durante el día sin importar la posición del sol durante este. Además de proporcionar una sensación de seguridad y tranquilidad en este mismo espacio para cualquier usuario a la hora de usarlo.

#### **¿Cuál es el alcance o restricción del problema?**

Por cuestiones de tiempo y recursos, el resultado esperado para este proyecto es un prototipo funcional a escala del mecanismo capaz de cumplir con las especificaciones dadas, usando como base de su funcionamiento la aplicación de los conocimientos adquiridos a lo largo de la carrera, focalizando en los procesos de diseño y robótica. Al momento de establecer las demás limitantes del proyecto se contaba con cierta idea basada en una no muy rigurosa observación, por lo que dejaba el enfoque y desarrollo del proyecto en un concepto muy subjetivo; razón por la cual se hizo uso de herramientas como la encuesta a diferentes personas consideradas como potencial usuario del producto:

Se realizó una encuesta a un total de 10 personas, a continuación se muestra la información recopilada de dichas encuestas:

#### **1. ¿Qué es lo que más le incomoda de una sombrilla de mesa?**

El calor que hace aún estando en la sombra, y que cuando hay fuertes vientos pero el día está soleado, es muy molesto ya que la sombrilla comienza a moverse a todos lados aun teniendo los soportes instalados.

#### **2. ¿Tiene alguna dificultad a la hora de ensamblar una sombrilla de mesa?**

Si hay veces no es claro el plano de que tubos hacen parte de la base y cuáles de las puntas además de que algunos tornillos no coinciden y toca hacer presión para que cuadren.

#### **3. ¿La sombrilla de mesa te parecen peligrosas?**

¡Si! me parece peligrosa principalmente para los niños,

ya que ellos se ponen a jugar y en ocasiones golpean la sombrilla y esta cae hay mismo golpeandolos, o en otra ocasiones uno quiere acomodar la sombrilla para que le de mas sombra y el seguro se suelta y se cae.

#### 4. ¿Cada cuanto guarda la sombrilla de mesa?

La guardo cada vez que hay fuertes lluvias o hay mucho viento, pero es molesto ya que es difícil de guardar y de volver a armar así que no he vuelto a sacarla a pesar de que estamos en época de pocas lluvias.

#### 5. ¿Si tuvieras la oportunidad de modificar un aspecto o característica de una sombrilla que harías?

Me gustaría que pudieran ser frescas y que no hagan calor a pesar de estar bajo la sombra y también me gustaría que ella sola se guardará o se desplegará, eso me ahorraría una o dos horas de mi tiempo. Para la definición de las restricciones en base a la idea que se tenía previamente y con la información recopilada de los usuarios, es posible aterrizar el proyecto a una base más sentada, donde decimos que el mecanismo previamente mencionado debe:

- Seguir al sol en todo momento.
- Ser resistente a los golpes.
- Funciona bajo la exposición a condiciones ambientales.

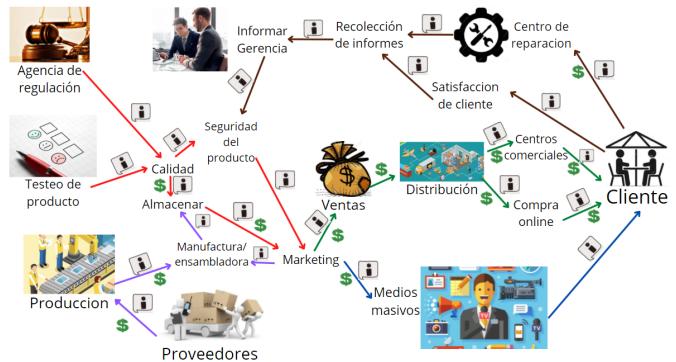
PLANTEAMIENTO DE LA MISIÓN: Proyecto-Sombrilla Inteligente	
Descripción del producto	Sombrilla con capacidad de generar una sombra a todos los asientos independientemente de la posición del sol.
Propuesta de valor	-Fácil configuración. -Regulación de la altura. -Resistente a climas atípicos. -Batería de larga duración. -Fácil de guardar. -Posición inteligente.
Objetivos de mercado	-Precios competitivos que permitan generar volúmenes de venta -Ser líderes y generar un nuevo segmento en la venta de sombrillas inteligentes -Aceptación de clientes y usuarios a través de buenas reseñas
Mercado primario	-Entes que requieran el uso de sombrillas de mesa en el exterior (Centros comerciales, restaurantes, casas campesinas, centros vacacionales etc) alrededor de la ciudad de Cali.
Premisas y restricciones	- Debe ser resistente a golpes - La lluvia no debe afectar el sistema - Debe verse bien en su entorno - Duración de carga de 1 semana - Debe permitir su uso conectado a un tomacorriente - Fácil mantenimiento - Debe seguir el sol en todo momento.
Agentes implicados	-Restaurante, centros recreativos y potenciales compradores. -Proveedores de piezas. -Cadena de producción. -Marketing y ventas. -Productos con funciones similares.

**Imagen 6. Planteamiento de la misión**

- Ser estética en el entorno.
- Contar con una batería duradera.
- Tener una comunicación con el usuario.

#### Agentes involucrados en el desarrollo del proyecto:

Ahora los agentes implicados son aquellos que van a ser afectados por el producto (en este caso tomamos todos aquellos que posean un beneficio de este producto). Teniendo esto en cuenta el mejor método para poder catalogar y conocer los agentes implicados es a través del Customer Value Chain Analysis o CVCA, el cual dio el siguiente mapa:



**Imagen 7. Agentes involucrados**

De este CVCA observamos entonces que los agentes involucrados, van a hacer:

- |  |   |
|--|---|
| <ul style="list-style-type: none"> <li>- Producción</li> <li>- Proveedores</li> <li>- Cliente</li> </ul> | <ul style="list-style-type: none"> <li>- Medios masivos</li> <li>- Marketing</li> <li>- Gerencia</li> </ul> |
|--|---|

En el mapa poseemos más agentes involucrados, pero se consideran estos como los más importantes ya que hacen parte de una línea simple de producción y venta, esto se decidió ya que se busca generar el mejor impacto en el mercado por lo que es oportuno apostar por la línea que pueda asegurar una ruta corta pero efectiva para llegar al cliente. Ahora, dentro de estos agentes involucrados podemos definir que el propio Cliente puede llegar a ser el más importante, pero esto se vuelve relativo al ver que cada uno depende de cada agente, es cierto que uno será más relevante que otro pero para abarcar a cada uno, es necesario primero entender el cliente, luego entender la producción, proveedores y por último el marketing, así tenemos una línea simple que busca que el producto sea eficaz al usuario, a la vez que sea fácil de producir para generar grandes cantidades.

#### Potenciales impactos del producto desarrollado:

Ahora que se realizó su respectiva investigación, de todos los aspectos que conllevan la creación física de esta solución, podemos resumirlo en una misión que estará dividida en varias secciones así como se muestra a continuación:

¿Cómo podemos asegurar el éxito de este producto?

El éxito de este producto también se asegura observando diferentes soluciones comerciales ya existentes en el mercado, así vemos el auge que este producto está generando en dicho mercado. Analizando dichos productos encontramos que los más representativos son:

**ShadeCraft Robotics:** Es una empresa de tecnología y robótica con sede en Pasadena, California, la misión

principal de esta empresa es la mejora en la calidad de la vida humana al aire libre, esto a través del diseño industrial, la tecnología innovadora y la robótica.

Hizo su debut con su primer producto conocido como Sunflower en CES 2017, el cual cambió para siempre el



panorama al aire libre esto a través de la introducción de la automatización y tecnología en los paraguas para exteriores existentes, siendo esta la

pionera en generar tecnología IOT de manera significativa al paisaje al aire libre. Siendo apoyada *Imagen 8. ShadeCraft umbrella* con creces, siendo ganadora de premios a la innovación y la fomentación de tecnologías renovables.

**FelsoGlass:** Es una empresa dedicada exclusivamente



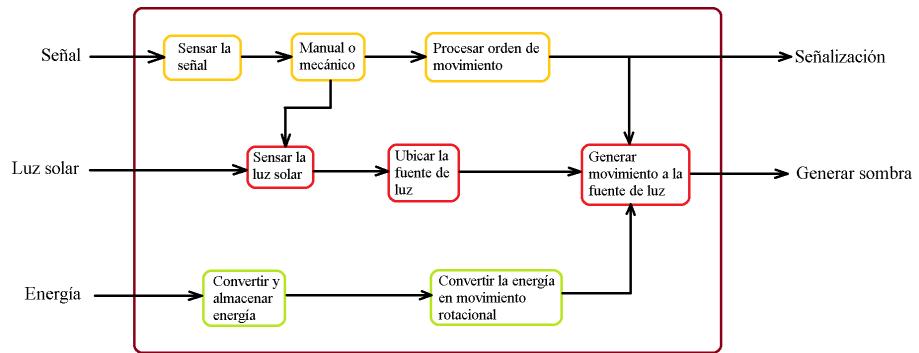
al asesoramiento técnico y comercial, venta e instalación de cerramientos con cortinas de cristal, correderas deslizantes, estructuras portantes fijas, techos móviles, pérgolas. Así el objetivo

*Imagen 9. FelsoGlass umbrella* principal que buscamos es el equilibrio entre lo estético y lo funcional, ofreciendo productos orientados a crear espacios seguros, que aporten bienestar y confort y que además contribuyan a la eficiencia energética, siendo su producto más característico la instalación de techos móviles, que al detectar la lluvia se cierran automáticamente, de igual manera al detectar cierta cantidad de luz esto se abrirá de manera automática dando paso a una mayor ventilación, esto depende también del usuario y de sus preferencias. Esta empresa es líder en Europa en techos inteligentes y domótica.

Como se puede observar hay varios productos que generaron un gran impacto, los cuales tiene como característica el uso de cierto nivel de inteligencia para la detección de la luz solar, o la automatización de un objeto común como lo son las sombrillas o techos, por esto afirmamos que el producto a desarrollar en este trabajo puede generar este impacto ya que entrará en un mercado que se considera nuevo, en el que cada objeto puede traer un grado innovación impactando al sector al que es dirigido.

## V. DISEÑO DE CONCEPTOS Y SOLUCIÓN

Después de realizar varias investigaciones a través de la casa de calidad y la búsqueda de necesidades y requerimientos, se llega a la fase de generación de conceptos, para realizar el concepto se debe de crear



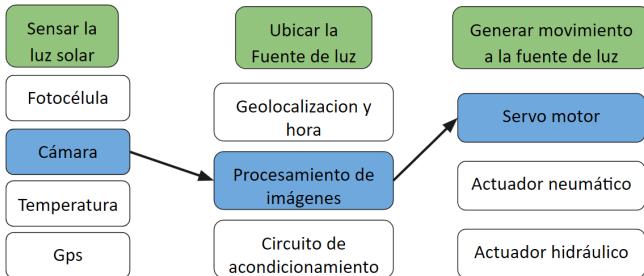
una caja transparente la cual nos permitirá poder observar el funcionamiento de manera general del proyecto, generando la siguiente caja transparente:

**Imagen 10. Caja transparente de funciones**

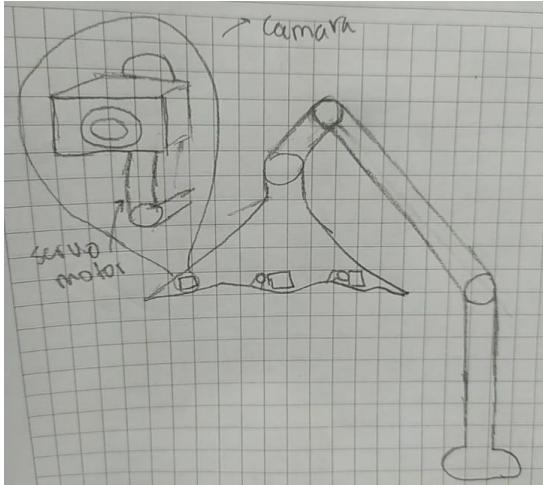
Ahora como se puede observar tenemos varios cuadros de subfunciones, donde cada función es vital para la generación del objetivo final, que en este caso es la Generación de sombra y la señalización de una orden. Ahora la linea mas importante en este caso será la que recibe como entrada la luz solar, esta se toma como la principal ya que a partir de este se va a generar la sombra que es el punto final, por lo que para realizar esta línea también es necesario desarrollar las otras 2 líneas, por lo que con base a esta línea generamos una lluvia de ideas:

Tomando como enfoque la adquisición y manipulación de la luz solar se dió solución a los subproblemas que surgieron y en base a esto fue posible generar una tabla de combinación de conceptos con 13 distintas posibles soluciones. Se tuvo en cuenta que algunas soluciones a los subproblemas no tienen manera de relacionarse con las soluciones de los otros subproblemas, siendo por ejemplo el caso de la propuesta de solución GPS que sólo se puede asociar con la propuesta de solución de Geolocalización y hora.

Concepto 1:



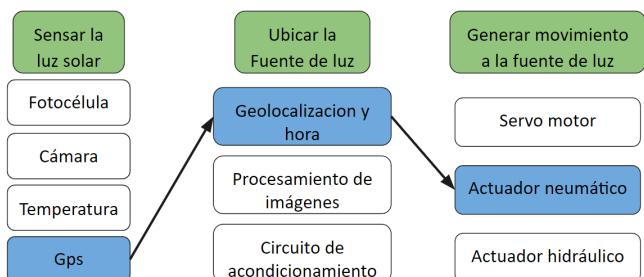
**Imagen 11. Concepto 1 de ideación**



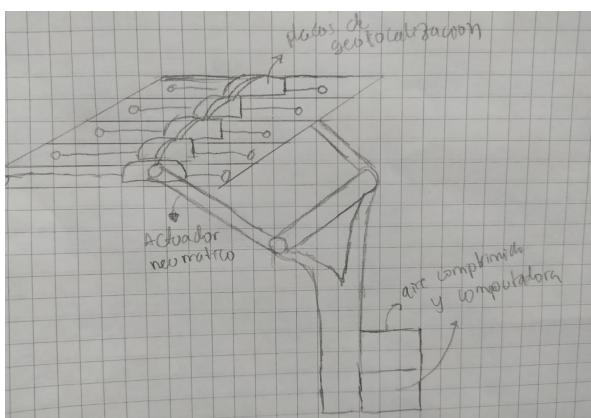
**Imagen 12. Boceto del concepto 1 de ideación**

Este concepto consiste en implementar cámaras para un procesamiento de imágenes, esto con el fin de establecer la posición del sol, para posteriormente mandar la orden a un servomotor, generando movimiento en la sombrilla a la posición detectada.

#### Concepto 2:



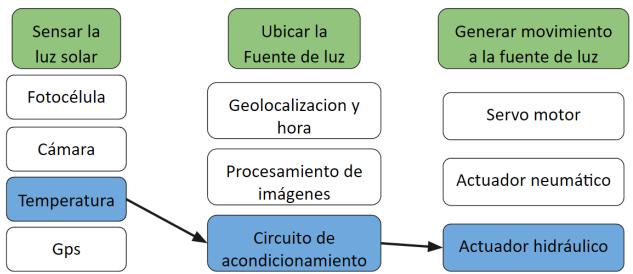
**Imagen 13. Concepto 2 de ideación**



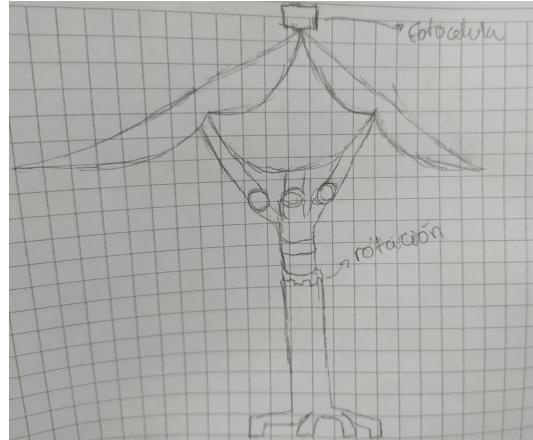
**Imagen 14. Boceto del concepto 2 de ideación**

Este concepto consiste en hacer uso de un gps de zona horaria para la ubicación del sol, con esto posteriormente hacer uso de actuadores neumáticos para la creación de movimiento.

#### Concepto 3:



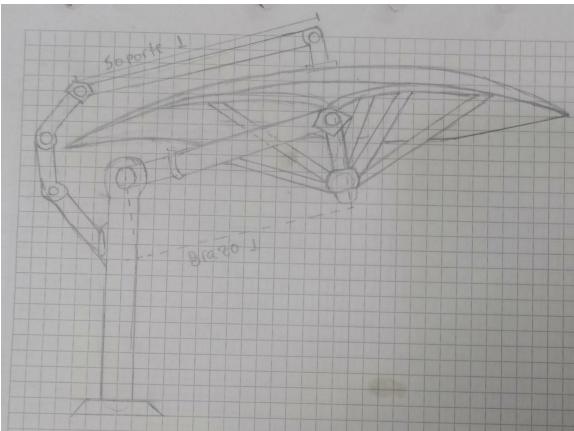
**Imagen 15. Concepto 3 de ideación**



**Imagen 16. Boceto del concepto 3 de ideación**

Este tercer concepto consta de realizar a través de un sensor de temperatura una aproximación de la mayor fuente de luz, con esto posteriormente generar una señal a un circuito de acondicionamiento, llevando a un actuador hidráulico ya sea una serie de pistones para la generación de movimiento.

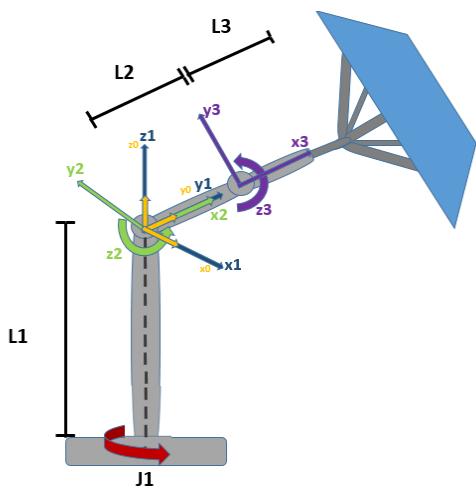
Selección de solución: La selección del concepto se realizó a través de un proceso de tamizaje de dos niveles, así el resultado de esta selección se decidió que el mejor concepto a desarrollar es un amalgama de conceptos, así el desarrollo final es la forma del concepto 3 pero combinado con el funcionamiento de concepto 2 para la parte de captación y procesamiento de información y por último se hará uso de un servomotor para el concepto 3, obteniendo lo siguiente:



**Imagen 17. Boceto del concepto final de ideación**

Así tenemos el diseño conceptual a desarrollar, es el de una sombrilla con seguimiento de luz solar a través de la zona horario de la región en la que la sombrilla se encuentre ubicada, después a través de un circuito de acondicionamiento y de geolocalización genere la señal para que un sistema de servomotores generen el movimiento y la dirección de la sombrilla.

Desarrollo del concepto: Ahora que se definió el concepto que se va a trabajar, es necesario ser más precisos en sus dimensiones y principalmente en la cantidad de articulaciones, por eso se realizó el siguiente diseño:



En este se observa entonces los grados de libertad que va a tener el robot, esto es necesario para poder generar dos matrices importantes para nuestro proyecto, el cual es la cinemática

**Imagen 18.**

**Análisis de las articulaciones del sistema** directa y la inversa del robot. Con estas dos definiciones podemos lograr controlar el robot a futuro pero primero es necesario calcular dichas matrices:

Lo primero entonces es el cálculo de la cinemática directa (todos estos cálculos están referenciados con la Imagen 18)

i	ai-1	ai-1	di	Θi
1	0	0	0	Θ1
2	90°	0	0	Θ2

3	0	L <sub>2</sub>	0	Θ <sub>3</sub>
---	---	----------------	---	----------------

**Tabla 1. Cinemática directa**

Por lo que la cinemática directa dado en sus matrices independientes son:

$${}^0T = \begin{bmatrix} c1 & -s1 & 0 & 0 \\ s1 & c1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^1T = \begin{bmatrix} c2 & -s2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s2 & c2 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2T = \begin{bmatrix} c3 & -s3 & 0 & L_2 \\ s3 & c3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Para tener una posición concreta respecto a las variables que se muestran en la tabla 1 es necesario realizar la cinemática inversa por lo que se procede realizar la inversa con matrices de transformación :

$${}^3T = {}^0T {}^1T {}^2T$$

$$({}^0T)^{-1}T = {}^1T {}^2T$$

$$\begin{bmatrix} c1 & s1 & 0 & 0 \\ -s1 & c1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & O_x & a_x & P_x \\ n_y & O_y & a_y & P_y \\ n_z & O_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^1T {}^2T$$

Eligiendo el elemento (3,4)

$$-P_x s1 + P_y c1 = 0$$

$$\theta_1 = \arctan2\left(\frac{P_y}{P_x}\right)$$

$$({}^1T)^{-1}({}^0T)^{-1}T = {}^2T$$

$$\begin{bmatrix} c2 & 0 & s2 & 0 \\ -s2 & 0 & c2 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c1 & s1 & 0 & 0 \\ -s1 & c1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & O_x & a_x & P_x \\ n_y & O_y & a_y & P_y \\ n_z & O_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^2T$$

Eligiendo el elemento (2,4)

$$-P_x c1 s2 - P_y s1 s2 + P_z c2 = 0$$

$$s2(-P_x c1 - P_y s1) + P_z c2 = 0$$

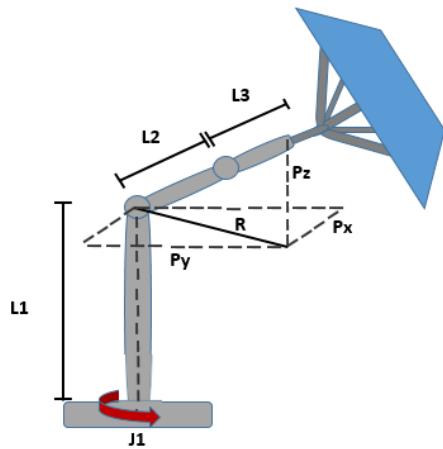
$$\theta_2 = \arctan2 - \frac{(-P_x c1 - P_y s1)}{P_z}$$

Eliriendo el elemento (1,4)

$$P_x c1 c2 + P_y c2 s1 + P_z s2 = L_2$$

$$c2(P_x c1 + P_y s1) + P_z s2 = L_2$$

Para  $\theta_3$  se aplicó el método geométrico



$$R^2 = P_y^2 + P_x^2$$

$$R^2 + P_z^2 = L_2^2 + L_3^2 + 2 * L_2^2 * L_3^2 \cos q_3$$

Despejar  $\cos q_3$  y reemplazar R

$$\cos q_3 = \frac{P_x^2 + P_y^2 + P_z^2 - L_2^2 - L_3^2}{2 * L_2^2 * L_3^2}$$

Poner la ecuación con término de seno

$$\sin q_3 = \pm \sqrt{1 - \cos^2 q_3}$$

Por lo que al obtener en términos de seno y coseno tendríamos :

$$\theta_3 = \arctan2 \frac{\pm \sqrt{1 - \cos^2 q_3}}{\cos q_3}$$

Al hallar la inversa de diferentes manera hace que un método no simplemente halla todos los ángulos respectivos, debido a que cada robot tiene una estructura que hace que en sí un método sea aplicable , pero este no resulte dando todos los ángulos necesarias para el resultado de la cinemática inversa.

Después de definir el concepto que vamos a usar (principalmente la forma y diseño de este) se procede a realizar un modelo en 3d, esto a través del simulador SolidWorks



Ahora se realizó varias mejoras con respecto al dibujo, lo primero es la parte inferior de la sombrilla, así como se puede observar la base es más ancha además de contar con dos bases, una será la rotacional y la segunda servirá como medio de agarre de toda la sombrilla. Además se eliminó el brazo adicional que se tenía en el dibujo ya que se considera que es innecesario,

**Imagen 19. Diseño 3D del robot** un solo brazo le da la suficiente fuerza y estabilidad a la sombrilla.Otro punto a destacar es que en el modelado no se realizó la sombrilla entera (con tela) ya que lo que nos interesa en estos modelados son las articulaciones que tendrá, así podemos definir los grados de libertad.

Después de realizar el modelado, podemos hacer una vista explosiva para observar las diferentes articulaciones creadas con más detalle.



Con la vista explosiva, encontramos 3 grados de libertad distintos, uno es el que conecta base rotacional con la base fija, la segunda es la que conecta el brazo con el tronco, y la tercera es la unión entre la estructura de la sombrilla y el brazo.

Teniendo así que los grados de libertad que maneja nuestro proyecto

será un total de 3. **Imagen 20. Diseño explosionado en SolidWorks del modelo inicial.**

Si el lector de este proyecto desea ver como se genera la vista de explosion, en el siguiente link se muestra la simulacion de dicha vista:  
<https://drive.google.com/file/d/1Hm76viQlzUkRinFxrnF0uvHikxgFJceC/view?resourcekey>

## VI. Visualización de la solución usando RVIZ

Después de tener nuestro modelo en solidwork a través de una herramienta que este simulador posee, generamos el urdf de dicho modelo:

```
<?xml version="1.0" encoding="UTF-8 ?>
<robot
  name="robotmilagro">
  <link
    name="base2_Link">
    <inertial>
      <origin
        xyz="0 0 0" />
      <mass
        value="37.3619805208222" />
    </inertial>
    <visual>
      <origin
        ixz="1.07878719190163"
        ixy="0.0421503067964398"
        izx="0.0134509247170474"
        iyy="1.0784469370143"
        izy="0.00131074568973874"
        izz="2.0532044687573" />
    </visual>
    <collision>
      <origin
        xyz="0 0 0"
        rpy="0 0 0" />
      <geometry>
        <nest>
          <mesh
            filename="package://robot_milagro/meshes/base_Link.STL" />
        </geometry>
      </material>
    </collision>
  </link>
  <link
    name="Joint_Link1">
    <inertial>
      <origin
        xyz="0 0 0"
        rpy="0 0 0" />
    <geometry>
      <mesh
        filename="package://robot_milagro/meshes/base_Link.STL" />
    </geometry>
  </link>
  </robot>
```

*Si desea ver completo el código ingrese a nuestro Git:*

[https://github.com/victorAlvarezValencia/ProyectoSombillaInteligente/blob/main/robot\\_milagro/urdf/robotmilagro.urdf](https://github.com/victorAlvarezValencia/ProyectoSombillaInteligente/blob/main/robot_milagro/urdf/robotmilagro.urdf)

Así tenemos la descripción de nuestro robot, esto acompañado con los meshes que contiene el material del robot, podemos generar una visualización 3d de este, pero para eso es necesario generar un archivo para poder hacer una reconstrucción del URDF, por eso es necesario la creación de un archivo Launch:

```
1 # Author: Addison Sears-Collins
2 # Date: August 27, 2021
3 # Description: Launch a basic mobile robot URDF file using Rviz.
4 # https://automaticaddison.com
5
6 import os
7 from launch import LaunchDescription
8 from launch.actions import DeclareLaunchArgument
9 from launch.conditions import IfCondition, UnlessCondition
10 from launch.substitutions import Command, LaunchConfiguration
11 from launch_ros.actions import Node
12 from launch_ros.substitutions import FindPackageShare
13
14 def generate_launch_description():
15
16     # Set the path to different files and folders.
17     pkg_share = FindPackageShare(package='robot_milagro').find('robot_milagro')
18     default_launch_dir = os.path.join(pkg_share, 'launch')
19     default_model_path = os.path.join(pkg_share, 'models/robotmilagro.urdf')
20     robot_name_in_urdf = 'robotmilagro'
21     default_rviz_config_path = os.path.join(pkg_share, 'rviz/urdf_config.rviz')
22
23     # Launch configuration variables specific to visualization
24
25     gui = LaunchConfiguration('gui')
26     model = LaunchConfiguration('model')
27     rviz_config_file = LaunchConfiguration('rviz_config_file')
28     use_robot_state_pub = LaunchConfiguration('use_robot_state_pub')
29     use_rviz = LaunchConfiguration('use_rviz')
30     use_sim_time = LaunchConfiguration('use_sim_time')
31
32     # Declare the launch arguments
33     declare_model_path_cmd = DeclareLaunchArgument(
34         name='model',
35         default_value=default_model_path,
36         description='Absolute path to robot urdf file')
37
38     declare_rviz_config_file_cmd = DeclareLaunchArgument(
39         name='rviz_config_file',
40         default_value=default_rviz_config_path,
41         description='Full path to the RVIZ config file to use')
42
43     declare_use_joint_state_publisher_cmd = DeclareLaunchArgument(
44         name='gui',
45         default_value='True',
46         description='Flag to enable joint_state_publisher.gui')
```

*Si desea ver completo el código ingrese a nuestro Git:*

[https://github.com/victorAlvarezValencia/ProyectoSombillaInteligente/blob/main/robot\\_milagro/launch/robotpurplea\\_display.launch.py](https://github.com/victorAlvarezValencia/ProyectoSombillaInteligente/blob/main/robot_milagro/launch/robotpurplea_display.launch.py)

En este archivo launch se invoca el Joint state publishes, que es el encargado de llamar a todos los joints del robot, después está el robot description, que es

la descripción del robot y luego la transformation publisher que generará las transformaciones con ayuda TFT para convertir las entidades y geometrías del URDF a una visualización física, con esto generamos el siguiente modelo:



Como se observa la conversión del modelo 3d de SolidWorks fue exitosa. Esto gracias a que se generó sus respectivos ejes y articulación de manera correcta en la extensión de exportación de Urdf en el SolidWorks.

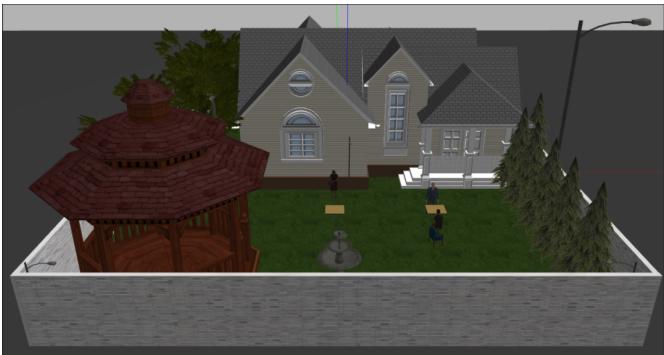
**Imagen 21. Modelo de la sombrilla simulado en RViz**

Un dato adicional que se puede observar es el hecho de que los colores cambian, este cambio se realizó en el urdf, además de este también se realizó los límites de movimiento de los Joints, así evitamos que estas atravesen por todo el cuerpo sólido de la sombrilla, generando un entorno más realista. Si desea ver el movimiento de las articulaciones las cuales consta de Joint\_Link1 que corresponde a un movimiento rotacional de 360 grados o una articulacion continua, Joint\_Link2, que sera el movimeinto del brazo robotico, el cual corresponde al de una articulacion rotacional y por ultimo el Joint\_Link3 que corresponde a una articulacion rotaciona y este se encuentra en la union entre la estructura de la sombrilla y el brazo robotico, todo esto lo puede visualizar en movimiento en Rviz en el siguiente video:

<https://drive.google.com/file/d/16QPIN1F-y4Hyw9tn7hkRJccvrU11JNj2/view?resourcekey>

## VII. Visualización en el simulador Gazebo

Ya se visualizó nuestro robot en rviz, esto significa de que el código y la programación son correctos, además se calculó la cinemática directa e inverso, estas componentes son requeridas en el simulador Gazebo puesto que aquí inicia la simulación real del funcionamiento del robot, pero antes es necesario generar un entorno simulado en el que podamos observar una mundo que se acerque al entorno en el que trabajaría la sombrilla, para esto se diseñó el siguiente mundo:



Como se observa en la anterior figura el entorno simulado será una casa con un jardín, ademas de esto es una casa de un estrato 4 para arriba esto se debe a que aquí se simula los compradores naturales, que en su mayoría serán personas de estos indoles, ahora para los compradores en potencia que serían centros comerciales, o restaurantes el mundo generado es el siguiente:



Como se observa este mundo, es mucho más cargado por lo que para una mayor fluidez en la simulación se usará el primer mundo. Continuando con la simulación es necesario generar nuestra sombrilla en este entorno creado, para esto se programa un Launch que permite conectar con un servidor de Gazebo para posteriormente generar el spawn del robot, esto obviamente implica usar la misma configuración de rviz para generar todas las articulaciones y dinámica del robot:

*Si desea ver completo el código ingrese a nuestro Git:*

```
1 import os
2 from launch import LaunchDescription
3 from launch.actions import DeclareLaunchArgument, ExecuteProcess, IncludeLaunchDescription
4 from launch.conditions import IfCondition, UnlessCondition
5 from launch.launch_description_sources import PythonLaunchDescriptionSource
6 from launch.substitutions import Command, LaunchConfiguration, PythonExpression
7 from launch.actions import SubprocessAction
8 from launch_ros.actions import Node
9 from launch_ros.substitutions import FindPackageShare
10
11 def generate_launch_description():
12     gazebo_models_path = "models"
13     package_name = "robotmilagro"
14     robot_name_in_model = "robotmilagro"
15     urdf_config_file_path = "rviz/urdf_gazebo.gazebo_config.rviz"
16     urdf_file_path = "urdf/robot/milagro_victor/milagro.urdf"
17     world_file_path = "worlds/mundo.world"
18
19     # Since we want to spawn the robot
20     spawn_x_val = "-10.0"
21     spawn_y_val = "0.0"
22     spawn_z_val = "0.0"
23     spawn_urdf_val = "-1.59"
24
25 ##### You do not need to change anything below this line #####
26
27     # Set the path to different files and folders.
28     package_share = FindPackageShare(package="robotmilagro").find("robotmilagro")
29     pkg_share = FindPackageShare(package="gazebo_ros").find("gazebo_ros")
30     default_urdf_model_path = os.path.join(pkg_share, "urdf_file_path")
31     default_urdf_file_path = os.path.join(pkg_share, "urdf_file_path", "rviz_config_file_path")
32     world_path = os.path.join(pkg_share, "worlds")
33     gazebo_models_path = os.path.join(pkg_share, "gazebo_models_path")
34     os.environ["GAZEBO_MODEL_PATH"] = gazebo_models_path
35
36     # Launch configuration variables specific to simulation
37     ld = LaunchConfiguration("ld")
38     ld.add_action(declare_namespace("headless"))
39     namespace = LaunchConfiguration("namespace")
40     rviz_config_file = LaunchConfiguration("rviz_config_file")
41     use_rviz = LaunchConfiguration("use_rviz")
42     use_namespace = LaunchConfiguration("use_namespace")
43     use_robot_state_pub = LaunchConfiguration("use_robot_state_pub")
44     use_sim_time = LaunchConfiguration("use_sim_time")
45     use_stm_time = LaunchConfiguration("use_stm_time")
46     use_simulator = LaunchConfiguration("use_simulator")
47     world = LaunchConfiguration("world")
```

[https://github.com/victorAlvarezValencia/ProyectoSombillaInteligente/blob/main/robot\\_milagro/launch/3\\_gazebo\\_controller.launch.py](https://github.com/victorAlvarezValencia/ProyectoSombillaInteligente/blob/main/robot_milagro/launch/3_gazebo_controller.launch.py)

Después de diseñar el launch ya podemos traer nuestro robot al simulador de Gazebo:



Como se observa las dimensiones del robot parecen algo exageradas en comparación con las personas, pero esto se debe a que la distancia adicional permite que se proyectó una mayor sombra, además de que debido al factor de seguridad es obligatorio crear cierta altura entre la sombrilla que es la parte móvil. Esto puede prevenir que las personas sean golpeadas a la hora que la sombrilla genere sus trayectorias.

**Cómo así que trayectorias?** Y si en este punto hemos investigado y formulado un problema de ingeniería, ademas se a desarrollado una solucion, pero aun no podemos decir que esta solución está completa, ya que en este punto de la simulación la sombrilla solo esta moviéndose a traves de la inercia de sus Joints, esto es un elemento que agrega Gazebo. Pero no es la respuesta que se desea buscar ya que recordando el objetivo planteado es necesario que la solución pueda hacer un seguimiento de la luz solar.

Desafortunadamente en el curso de robótica no tenemos los conocimientos suficientes para hacer uso de sensores reales para ser conectados a ROS y posteriormente a Gazebo para generar dicha interacción en cambio, lo que podemos realizar es una serie de trayectorias ya definidas, con el fin de comprobar la funcionalidad de la sombrilla, para esto es necesario crear un nodo suscriptor en gazebo, para que así cuando se activó un nodo publicador, este inmediatamente se inscriba para recibir el comando de la trayectoria, con esto en mente es necesario realizar los siguientes cambios en el Launch de Gazebo:

```
157 # Launch the robot
158 spawn_entity_cmd = Node(
159     package="gazebo_ros",
160     executable="spawn_entity.py",
161     arguments=[ "entity", "robot_description",
162                 '-x', spawn_x_val,
163                 '-y', spawn_y_val,
164                 '-z', spawn_z_val,
165                 '-spawn_yaw_val'],
166     output="screen")
167 declare_use_joint_state_broadcaster_cmd = ExecuteProcess(
168     cmd=["ros2", "control", "/load_controller", "--set-state", "joint_state_broadcaster"],
169     output="screen")
170 declare_use_joint_state_trajectory_controller_cmd = ExecuteProcess(
171     cmd=["ros2", "control", "/load_controller", "--set-state", "start", "joint_trajectory_controller"],
172     output="screen")
173
174 ld = LaunchConfiguration("ld")
175 ld.add_action(declare_use_joint_state_publisher_cmd)
176 ld.add_action(declare_use_namespace_cmd)
177 ld.add_action(declare_use_urdf_file_cmd)
178 ld.add_action(declare_simulator_cmd)
179 ld.add_action(declare_use_urdf_file_path_cmd)
180 ld.add_action(declare_use_robot_state_pub_cmd)
181 ld.add_action(declare_use_rviz_cmd)
182 ld.add_action(declare_use_sim_time_cmd)
183 ld.add_action(declare_use_simulator_cmd)
184 ld.add_action(declare_use_world_cmd)
185 ld.add_action(declare_use_joint_state_broadcaster_cmd)
186 ld.add_action(declare_use_joint_state_trajectory_controller_cmd)
```

*Si desea ver completo el código ingrese a nuestro Git:*

[https://github.com/victorAlvarezValencia/ProyectoSombillaInteligente/blob/main/robot\\_milagro/launch/3\\_gazebo\\_controller.launch.py](https://github.com/victorAlvarezValencia/ProyectoSombillaInteligente/blob/main/robot_milagro/launch/3_gazebo_controller.launch.py)

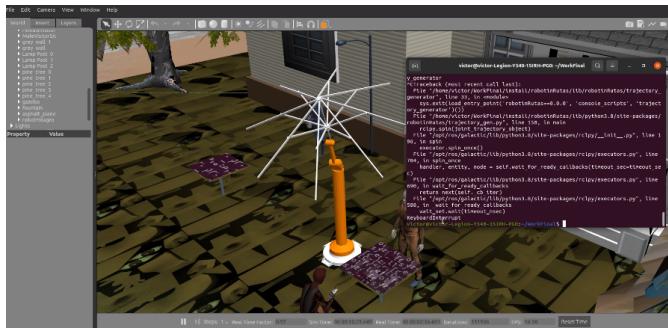
Las trayectorias de la sombrilla nos ayudan al movimiento en donde se sitúa la sombrilla y donde esta se posiciona frente diferentes ángulos respectos al sol , en donde estas trayectoria son dadas por la cinemática inversa en el que dando los valores de Px,Py y Pz obtenemos los valores respectivos de los grados de libertad de la sombrilla en donde son  $\theta_1, \theta_2$  y  $\theta_3$  , que a su vez se generan diferentes trayectoria dependiendo del valor asignado en Px,Py y Pz, como se muestra en el siguiente código:

```

13 class Trajectory_publisher(Node):
14
15     def __init__(self):
16         super().__init__('trajectory_publisher_node')
17         publish_topic = '/joint_trajectory_controller/joint_trajectory'
18         self.trajectory_publisher = self.create_publisher(JointTrajectory,publish_topic, 10)
19         timer_period = 7
20         self.timer = self.create_timer(timer_period, self.timer_callback)
21         Callangulos = self.angulosInv()
22         θ1= Callangulos[0]
23         θ2 = Callangulos[1]
24         θ3 = Callangulos[2]
25         θ11 = Callangulos[3]
26         θ21 = Callangulos[4]
27         θ31 = Callangulos[5]
28         θ12= Callangulos[6]
29         θ22 = Callangulos[7]
30         θ32 = Callangulos[8]
31         self.joints = ['joint1','joint2','joint3']
32         self.goal_positions = [θ1,θ2,θ3]
33         self.goal_positions1 = [θ11,θ21,θ31]
34         self.goal_positions2 = [θ12,θ22,θ32]
35
36     def angulosInv(self):
37         Px=6
38         Py=-1
39         Pz=4
40         l2=5
41         l3=5
42         θ1=arctan2(Py,Px)
43         θ2=arctan2(Px*cos(θ1)+Py*sin(θ1),Pz)
44         cosq3=(Px**2+Py**2+Pz**2-l2**2-l3**2)/(2*l2**2*l3**2)
45         sinq3=sqrt(1-cosq3**2)
46         θ3=arctan2(sinq3,cosq3)
47
48         Px1=-5
49         Py1=-1
50         Pz1=4
51         l21=5
52         l31=5
53         θ11=arctan2(Py1,Px1)
54         θ21=arctan2(Px1*cos(θ11)+Py1*sin(θ11),Pz1)
55         cosq3=(Px1**2+Py1**2+Pz1**2-l21**2-l31**2)/(2*l21**2*l31**2)
56         sinq3=sqrt(1-cosq3**2)
57         θ31=arctan2(sinq3,cosq3)

```

[https://github.com/victorAlvarezValencia/ProyectoSombillaInteligente/blob/main/robot\\_milagro/launch/3\\_gazebo\\_controller.launch.py](https://github.com/victorAlvarezValencia/ProyectoSombillaInteligente/blob/main/robot_milagro/launch/3_gazebo_controller.launch.py)

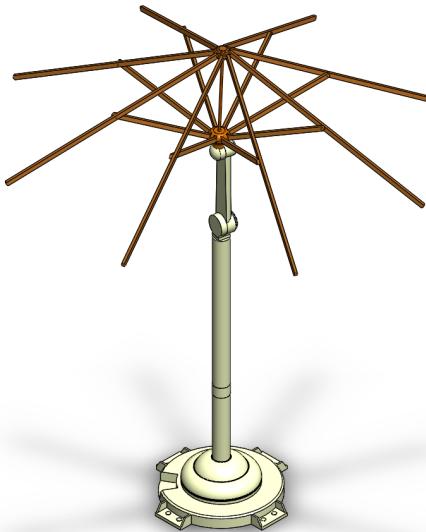


### VIII. Construcción de un prototipo

Después de hacer sus respectivas investigaciones, y llegando a este punto en el que se logró generar varias rutas que simulan el funcionamiento de la sombrilla, se tiene la certeza y la confianza de que se puede llegar a la construcción de un modelo físico de este brazo, ahora dicho modelo fisico tiene como objetivo el demostrar que el diseño y forma del robot son el camino correcto para la construcción de un prototipo final, para esto se realizó una serie de procedimientos para su construcción:

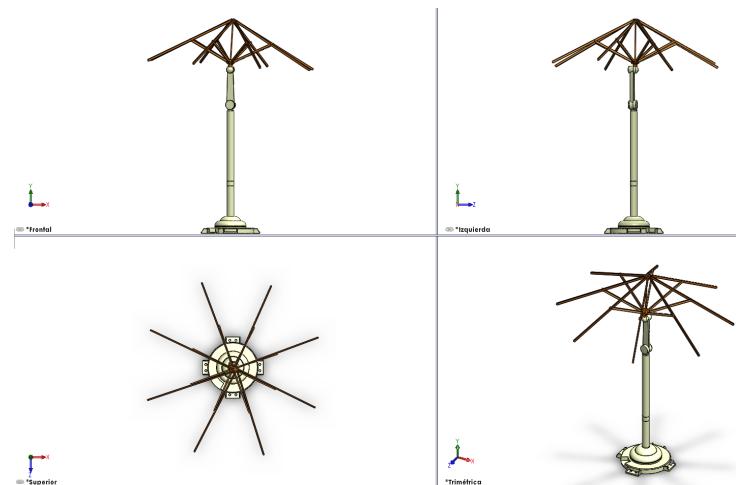
Como se mencionó anteriormente el objetivo de este prototipo fue con el fin de comprobar que el diseño y el concepto de funcionalidad escogidos sean los correctos o en caso contrario requieren ser modificados, recordando que hasta la versión de venta, todo proyecto puede estar en constante cambio y mejoras, con esto en mente se comienza con el diseño detallado:

Mejoras En solidworks: Comenzando se realizó un nuevo diseño del solid, este diseño se intentara ser implementado en el prototipo físico, ahora como se observa el nuevo diseño se centra en la estética y funcionalidad, teniendo así que la sombrilla está seccionada en dos partes, permitiendo que sólo una sea la rotacional, esto brinda una mayor facilidad al mover una menor cantidad de peso, además se elimina la forma del brazo dejando solo un tubo que servirá como brazo robótico:



**Imagen 15.**  
**Diseño 3D**  
**sombrilla**

Con este nuevo diseño, hallamos varios beneficios aparte de una mayor facilidad para el movimiento, también permite que futuras mejoras puedan ser implementadas además de que el intercambio de piezas sea mucho más sencillo a través de una configuración modular, con esto obtenemos las nuevas vistas de la sombrilla:



**Imagen 16. Vistas del diseño 3D de la sombrilla**

Ahora como poseemos un diseño de solidWork más complejo, es hora de trasladar esta información a un formato real, para esto primero observamos el peso de la sombrilla y los momentos de inercia (El peso está con

base a una aleación de aluminio además los puntos de inercia son necesarios para el modelo urdf en caso de nuevas simulaciones).

#### Propiedades de masa de EnsambleSombrillaPrueba2

Configuración: Predeterminado

Sistema de coordenadas: -- predeterminado --

Masa = 69039.46 gramos

Volumen = 71738.71 centímetros cúbicos

Área de superficie = 40372.35 centímetros cuadrados

Centro de masa: ( centímetros )

X = -3.49

Y = 52.87

Z = 81.56

Ejes principales de inercia y momentos principales de inercia: ( gramos \* centímetros cuadrados )

Medido desde el centro de masa.

$I_x = (0.00, 1.00, 0.00)$   $P_x = 42086186.53$

$I_y = (0.68, 0.00, 0.73)$   $P_y = 527742847.50$

$I_z = (0.73, 0.00, -0.68)$   $P_z = 528593465.20$

Momentos de inercia: ( gramos \* centímetros cuadrados )

Obtenidos en el centro de masa y alineados con el sistema de coordenadas de resultados.

$L_{xx} = 528195676.26$   $L_{yy} = 298519.69$   $L_{zz} = 425806.02$

$L_{yx} = 298519.69$   $L_{xy} = 42096998.64$   $L_{yz} = 2272633.62$

$L_{zx} = 425806.02$   $L_{zy} = 2272633.62$   $L_{xz} = 528129824.33$

Momentos de inercia: ( gramos \* centímetros cuadrados )

Medido desde el sistema de coordenadas de salida.

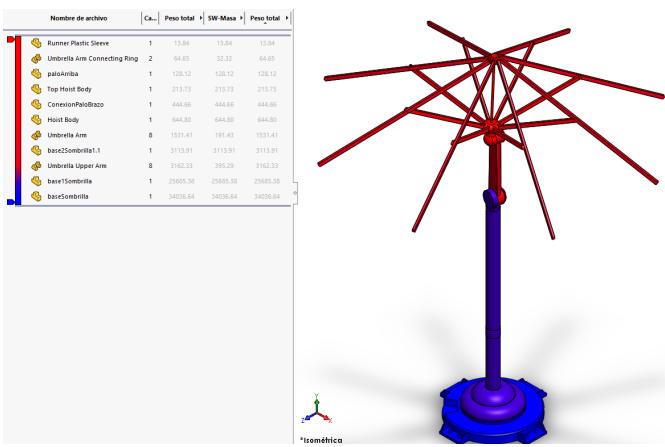
$I_{xx} = 1180443129.69$   $I_{yy} = -12433115.80$   $I_{zz} = -19215631.75$

$I_{yx} = -12433115.80$   $I_{xy} = 502212197.42$   $I_{yz} = 299976123.11$

$I_{zx} = -19215631.75$   $I_{zy} = 299976123.11$   $I_{xz} = 721942056.78$

**Imagen 17. Propiedades del material de la sombrilla**

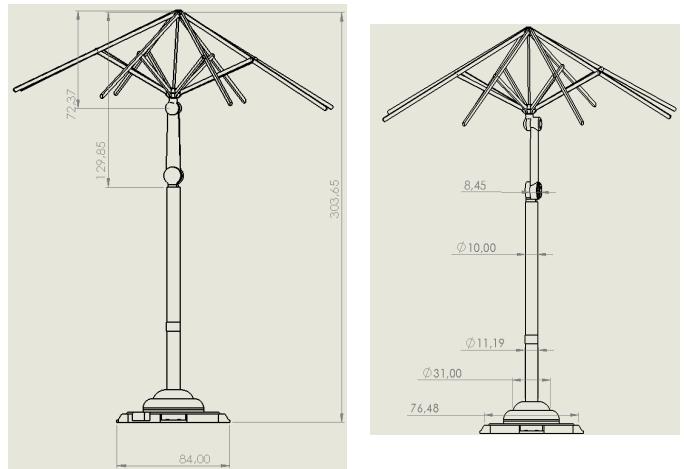
Como observamos de la siguiente tabla es que el peso de la sombrilla es algo elevado, además de su volumen es elevado, pero considerando las dimensiones del producto, se llega a entender de que es normal, aun así para asegurar de que la masa esté distribuida de tal manera que la parte más liviana sea la parte del brazo y la sombrilla, hacemos un estudio de distribución de masas en el simulador SolidWorks:



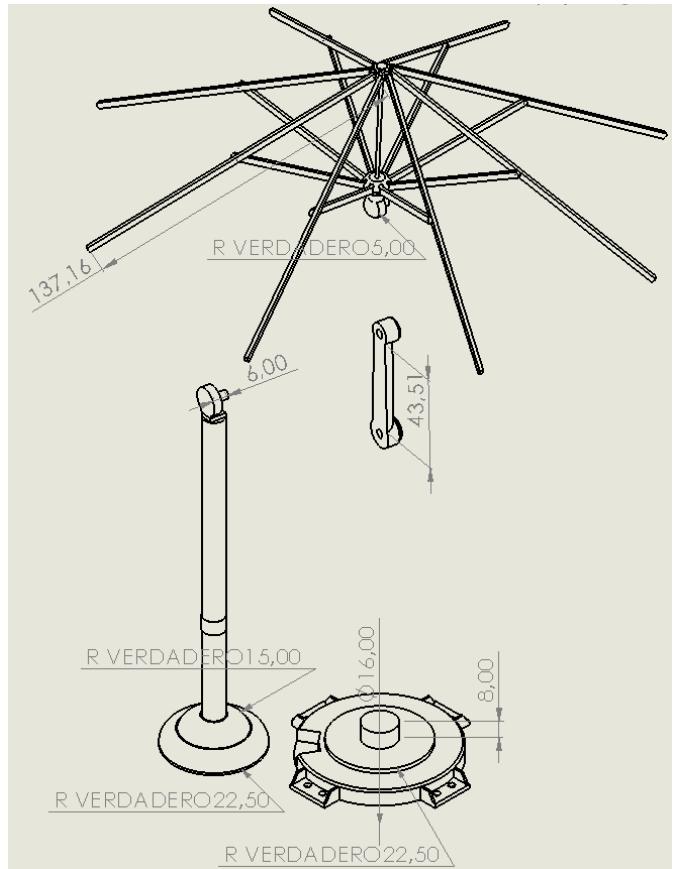
**Imagen 18. Análisis de masa en SolidWorks**

Como se puede observar el simulador resalta que el mayor peso y masa se encuentran distribuidos tanto en la base como el tronco principal de la sombrilla, y en caso contrario es la sombrilla la parte mas liviana, ademas de esto recordando que el tronco se encuentra seccionado en dos partes, permite entonces que la masa que se va a mover sea menor a la que se muestra en el solid, observando así que su eficiencia es mayor. Ahora que nos aseguramos de que la distribución es correcta, de que la tolerancia no generan un mal movimiento de las articulaciones, y de que el diseño nuevo permite una

mayor libertad de movimiento, es hora de la generación de planos:



**Imagen 20. Plano con medidas del modelo en SolidWorks**



**Imagen 21. Plano con medidas del modelo en SolidWorks**

Aquí tenemos todas las medidas necesarias para el diseño, que a la hora de ser compradas con las de una sombrilla normal, se concluye de que son perfectamente similares, esto es necesario ya que se debe de asegurar de que la sombrilla entre al mercado con similitudes de competidores, para así empatizar con nuevos clientes.

Montaje Físico: Despues de obtener los planos y dimensiones de la sombrilla, es necesario comenzar a

realizar un prototipo más específico, para esto se encontró tres grandes problemas:

**-Dimensiones:** Debido a las dimensiones de la sombrilla, su manipulación es compleja y costosa con el presupuesto que se cuenta, por ende para solucionar esto se redujo y se hizo un prototipo de escala 1:2 aproximadamente, ya que si se hace por el contrario muy pequeño, no se estaría demostrando la funcionalidad.

**-Mecanismos:** Los mecanismos como motores resultaron ser más difíciles de obtener debido a sus costos, para esto se realizó una alternativa que se verá más adelante.

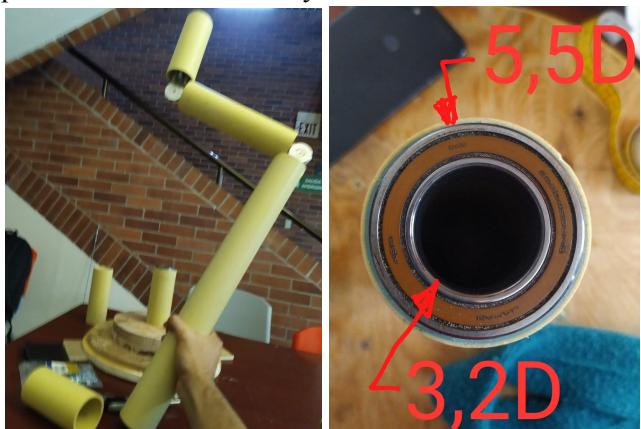
**-Material:** Debido a las dimensiones y a la forma de la sombrilla, materiales como madera, o icopor no son capaces de soportar el peso y permitir movimiento, por ende para la arquitectura del prototipo se hace uso de un material poco común para el prototipado.

Después de solucionar estos problemas que surgieron a la hora de comenzar con la implementación física, se inicia con la base y estructura de la sombrilla, para eso se hace uso de tubos de PVC o de cañería, ya que son resistentes, tienen dimensiones definidas además de que venden múltiples accesorios como codos, o balineras que traen sus mismas dimensiones, por lo que se inicia con su construcción:



**Imagen 22. Proceso de construcción del prototipo**

Esta es la base de la sombrilla, para ser lo más ecológico posible con el medio ambiente, se reutilizaron tablas de cama, para la construcción de la base, posteriormente se construyó el resto del robot:



**Imagen 23. Construcción de la estructura externa del sistema**

Aquí se comienza a desarrollar las primeras funcionalidades, que en este caso será el movimiento rotacional del tronco, para esto se hacen uso de dos

tubos, uno el más pequeño contiene un rodamiento, que permite rotar sobre su propio eje, posteriormente se ingresa un tubo, del tamaño del diámetro interno del rodamiento:



**Imagen 24. Medidas internas y externas del tubo principal**

Después con un círculo de madera en el segundo tubo de PVC, unimos ambas secciones, teniendo así la estructura de la sombrilla:



**Imagen 25. Vista del primer prototipo presentado**

Como se observa la altura de la sombrilla está en un término medio, de no ser tan pequeña para que sea inútil, ni de ser de las dimensiones completas de una sombrilla real, por lo que nos permite una mayor movilidad y facilidad a la hora de experimentar con los motores, aun así esta estructura no es lo suficientemente sólida para sostenerse, por lo que los siguientes pasos es integrar soportes ortogonales, con frenos, para generar algo de resistencia, además de limitar el movimiento a través del uso de ejes de madera:



### **Imagen 26. Vista del modelo físico en el laboratorio de prototipado articulación final**

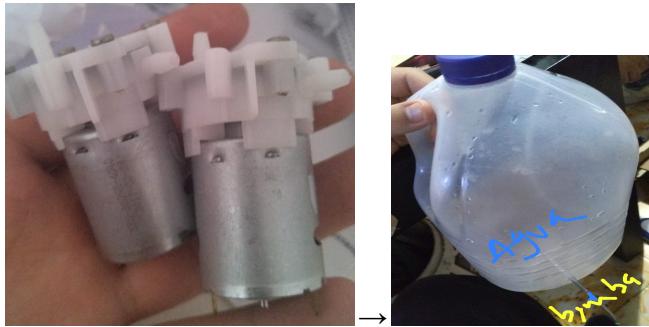
Con esto tenemos la arquitectura de la sombrilla completa y funcional, ahora el siguiente paso es:

**Funcionalidad:** En el concepto se definió que se haría uso de **SERVOMOTORES O MOTORREDUCTORES**, lastimosamente afrontamos la realidad de que el peso de la sombrilla es demasiado grande para ser levantados por los motores, descartando la idea inicialmente. Así nos devolvimos a la generación de conceptos, y analizamos que el segundo concepto que estuvo cerca de ganar, era la implementación de la hidráulica para el movimiento de la sombrilla, y aunque fue descartada por su difícil manipulación y poca precisión en comparación a los servomotores, se decidió por generar una nueva alternativa, que consta en combinar la hidráulica con los motores, así se realizaron las primeras pruebas:



**Imagen 27. Jeringa usada para la manipulación de la orientación**

Con el uso de pequeñas jeringas en ambos extremos se determinó de que si es posible mover el brazo con estas, pero el tamaño de las jeringas y la presión no eran suficientes todavía, ademas de que se tenía una gran limitante y era el largo de las jeringas, para eso se introduce el concepto de **MOTORES DE BOMBA**, que son básicamente motores que se modificaron y se colocaron una aspa interna, para posteriormente ser envueltos en una cámara de plástico en el que se transporta el agua:



**Imagen 28. Motores y agua usada para el movimiento**

Con estas motobombas, se conectaron a una sonda y posteriormente se colocó a una jeringa de mayor

tamaño, esperando que el concepto se aplique: El principio es simple, con la motobomba, se extraerá agua de un depósito, esta agua del depósito debe de ser igual o mayor a la cantidad que puede almacenar la jeringa a la que va conectada, así al ser arrastrada se aplicará el principio de pascal, llegando a una superficie pequeña de jeringa, donde se aplicara una mayor tensión superficial, generando un gran torque y así poder mover cierta cantidad de Kg, la cual para levantar el brazo de la sombrilla, esta aguja debe de tener la capacidad de levantar alrededor de 10 kg.



**Imagen 29. Modelo físico final sin la sombrilla**

Con un concepto definido y con los materiales necesarios, se aplicó la jeringa, de manera externa, esto en forma de un soporte triangular, debido a que se consideró el punto de masa del brazo, para así que la aplicación de la fuerza se distribuya uniformemente, y oponga una menor resistencia, así obtenemos el resultado que puede

observar a mano izquierda de este párrafo. Concluyendo que este es el método más efectivo para generar el movimiento necesario, en este prototipo.

### **Imagen 30. Modelo final**

Ahora lo último que queda es añadir la otra jeringa y la sombrilla obteniendo el resultado a mano derecha. Ahora si desea ver el funcionamiento de la sombrilla, puede observar desde los link de abajo, recuerde que debe de abrirlo con la cuenta institucional de la Universidad autónoma de occidente.



link1:

<https://drive.google.com/file/d/1LOqOrnShmTcgKryFd>

[F5QKeLhHeyZSGuB/view?usp=sharing](https://drive.google.com/file/d/16xiCEN-63z9H-2MDXTPXgvaKHn2ILkQ/view?usp=sharing),  
Link2:<https://drive.google.com/file/d/16xiCEN-63z9H-2MDXTPXgvaKHn2ILkQ/view?usp=sharing>

## IX. ¿Trabajo independiente?

Durante el desarrollo de este proyecto, se consultaron en varias ocasiones diferentes fuentes, que fueron usadas para poder completar o agregar cosas extras en el proyecto.

De hecho una de las fuentes más usadas fueron de parte de The Construct una plataforma de cursos virtuales, en el que aprendimos muchas cosas y datos, pero lo que más usamos fueron las siguientes entradas:

“How to use Xacro files with Gazebo in ROS2”

“Spawn a URDF model in Gazebo with ROS2”

En estos títulos nos explicaron el procedimiento para realizar un Xacro con Gazebo incluyendo el mundo y la modificación de diferentes elementos como la dinámica, choques la generación de archivo xacro en un archivo de lanzamiento de simulación entre otros.

Con esta información se logró añadir el archivo de lanzamiento de la simulación de Gazebo o Launch junto con el urdf en la misma carpeta lo que permitió de que no fuera necesario estar moviéndose en dos carpetas distintas si se deseaba hacer una configuración ya sea en el urdf o en el Launch de Gazebo facilitando enormemente la organización de las carpetas del proyecto.

La segunda fuente que más se usó fue el mismo repositorio de ROS2, al inicio cuando se generó varios problemas con respecto al URDF nuestra mayor ayuda fue ROS2 Documentation: Foxy en el apartado de:

### URDF Tutorials

Y la primera pregunta fue, ¿porque Foxy cuando se está trabajando en Galaxy? La respuesta es sencilla, ya que foxy tiene mayor apartado de información que Galaxy esto se debe al tiempo que lleva publicado cada uno, siendo Foxy el más viejo, además nuestro problema radicaba en la compresión del URDF por lo que estos documentos fueron de gran utilidad para entender el funcionamiento y capacidad que tiene este archivo, así se logró configurar los límites de movimiento de la sombrilla, limitando el movimiento del brazo y de la parte superior de la sombrilla, además de poder añadir inercia y dinámicas, puesto que en el exportUrdf de Solidwork no se genera ningún movimiento dinámico ni limitación de links.

Otros aspecto pero que parece irrelevante pero de gran utilidad fue que en estos documentos también se logró conocer la combinación para diferentes colores de la

sombrilla y aunque no fueron muchos los colores que logramos encontrar, por lo menos los que conseguimos le da un toque estético diferente a la sombrilla.

La tercera fuente de información fueron videos acerca de cómo exportar modelos de SolidWork en formato URDF, para este apartado las principales fuentes de consulta fueron videos de youtube, por lo que en la bibliografía se dejará algunos de los que más se usó. Para descargar la extension de URDF se usó una biblioteca llamada ROS.org, en el que dentro de ella se encontró un artículo de una extensión para Solid que exportaba en URDF



About | Support | Discussion Forum | Service Status | Q&A ans

Documentation

Browse Software

News

[sw\\_urdf\\_exporter](#)

### SolidWorks to URDF Exporter

The SolidWorks to URDF exporter is a SolidWorks add-in that allows for the convenient export of SW Parts and Assemblies into a URDF file. The exporter will create a ROS-like package that contains a directory for meshes, textures and robots (urdf files). For single SolidWorks parts, the part exporter will pull the material properties and create a single link in the URDF. For assemblies, the exporter will build the links and create a tree based on the SW assembly hierarchy. The exporter can automatically determine the proper joint type, joint transforms, and axes.

Download  
Installer

If the above provided download fails to work on your system, please install by source before reporting an issue. If installing by source does work, please submit an [Update Installer Request](#)

#### 1. SolidWorks Version Compatibility

There is a known STL export bug with SolidWorks 2018 that exists up to Service Pack 4 that renders this add-in unusable. If you are using 2018, please update to service pack 5 or use SolidWorks 2019 or later. 2017 and below may also work

#### 2. Some Important Items

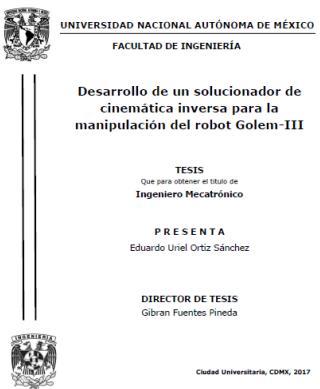
Development on this plugin as of recently has come from the generous donations of several ROS community members. Without this support or from pull requests from communities members, development would not be able to continue. We also appreciate any bugs or feature requests so that we can work on them when the resources become available.

This project is not dependent on ROS and can be used for exporting SolidWorks files for any URDF needs. The URDF will include rospack URI file locations ('package:/'), so you will need to change those for non-ROS systems (some Gazebo systems can handle 'package' URLs).

Con ayuda de la guía que ofrece la página se logró colocar dicha extensión URDF para el programa de SolidWork, sin esta página hubiera sido imposible poder trasladar nuestro modelo utilizado a URDF al ser usuarios nuevos en este mundo de ROS.

Por último una fuente que se usó pero desafortunadamente no se aplicó fue un libro con el nombre de “ROS Robotics By Example”, segunda edición en este libro aprendimos acerca de los nodos y los topics de Ros, aprendiendo configuración de nodos, esto nos permitió aplicar y configurar el nodo necesario para realizar la trayectoria, también queríamos incluir más funciones pero el libro maneja ROS y no ROS2 por lo que hubo muchos comandos que no pudimos ejecutar y no logramos encontrar su versión en ROS2.

Además de estas fuente se busco tesis relacionadas con el tema como lo fue la tesis de “Desarrollo de un solucionador de cinemática inversa para la manipulación del robot Golem-III” del ingeniero mecatrónico Eduardo Uriel Ortiz Sánchez, en la universidad nacional autónoma de México[1]



En que esta tesis nos ayudó en la hora de comprender la cinemática directa e inversa del robot y como era los pasos a hallar , por consiguiente también en el aspecto de relación de uso de ros y gazebo en funcionamiento en código como funcionamiento físico y cómo se conectaban entre sí aquellos dispositivos .

Sin embargo también se consultó diferentes formas y métodos de realizar matrices ya conocidas en clase como lo fue la jacobiana una forma interesante de hallar la directa e inversa cinemática.[2]

### Jacobiana inversa

- **Inversión simbólica de la matriz Jacobiana:**
  - Gran complejidad: matriz 6x6 de funciones trigonométricas.
- **Evaluación e Inversión numérica de la matriz Jacobiana:**
  - Necesidad de recálculo continuo.
  - En ocasiones  $J$  no es cuadrada . Matriz pseudoinversa ( $J J^T$ ) $^{-1}$ .
  - En ocasiones el determinante de  $J$  es nulo: configuraciones singulares.

$$q_1 = f_1(x, y, z, \alpha, \beta, \gamma) \quad \begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix} = J^{-1} \cdot \begin{bmatrix} x \\ \vdots \\ z \end{bmatrix} \quad J^{-1} = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \dots & \frac{\partial f_1}{\partial \gamma} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x} & \dots & \frac{\partial f_n}{\partial \gamma} \end{bmatrix}$$

## X. Discussion

Este proyecto se realizó teniendo en cuenta varios criterios según la rúbrica de evaluación de la asignatura de Robótica, dichos criterios son:

1. Aplicación de los principios básicos de ingeniería para resolver problemas complejos de ingeniería.
2. Rigurosidad en el uso de los principios de ciencia y matemática para resolver problemas complejos de ingeniería
3. Credibilidad y actualización de las busquedas de información
4. Pertinencia en la aplicación de información

Ahora los logros principales que se obtuvieron de este proyecto son:

- Formulación de problemas de ingeniería, innovación y rediseño de productos: Se logró analizar de forma clara y precisa un problema

para posteriormente ser transformado en un problema de ingeniería, además de un proceso de rediseño de dicho producto, siendo en este caso el rediseño de una sombrilla con el fin de cumplir el mismo objetivo de una sombrilla común, pero siguiendo diferentes rutas, así cumplimos con el 1 criterio de evaluación.

- Análisis cinemático de un robot desde cero y cálculos de diseño: Durante la resolución del problema se realizó diferentes cálculos matemáticos como la cinemática directa e inversa, además de varios cálculos simples para la creación de un modelo físico, por lo que desde el punto de vista de los autores de este artículo, podemos afirmar que se aplicaron distintos principios de ciencia y matemática.
- Búsqueda de diversas fuentes confiables: Durante el desarrollo del proyecto se tuvo que recurrir a diversas fuentes, por lo que durante el proceso siempre se verifica que dichas fuentes tengan una manera de mostrar su credibilidad con respecto a la información mostrada, por eso las fuentes observadas en este artículo son de carácter profesional, respaldado por estudios demostrados, entidades entre otros métodos que certifica la calidad de la información.
- Uso de información recolectada: Este punto es uno de los logros que mas se uso para el trabajo, puesto que en varios momentos del proyecto fue necesario recurrir a la manipulación de distintos proyectos de tercero junto con la información que brindan para poder aplicarse en el caso que estamos manejando, ejemplos claros tenemos la aplicación de los conocimientos ofrecidos por un trabajo para la creación de modelos de cinemática directa e inversa. O la modificación de archivos URDF o archivos tipo Launch, para adaptarse a las necesidades que teníamos.

Así podemos confirmar que durante todo el desarrollo de este proyecto se cumplieron con los criterios de evaluación además de la integración de conocimiento, para dar la solución más realista y efectiva posible.

## XI. Conclusiones

- Se planteó la solución para la problemática de manera correcta gracias al manejo adecuado de herramientas como la guía Gemma que resulta ser muy práctica para el desarrollo de soluciones automatizadas, ya que garantiza el cumplimiento de estándares de funcionamiento y permite un diseño estructurado y flexible que brinda equilibrio entre las necesidades específicas y generales del proyecto.
- El análisis permitió resaltar la importancia de las marchas de verificación, ya que al momento

de realizar la validación del funcionamiento completo del proceso, se puede observar cada movimiento de manera pausada, lo que permite analizar y descartar fallos en algún punto.

- Al utilizar diferentes herramientas de desarrollo de robótica como lo fue rviz y gazebo nos ayudan a mejorar aplicar del funcionamiento del simulador a como se reflejaba en un aspecto de la vida real.

## Referencias

Actividades	SEMANAS									
	8	9	10	11	12	13	14	15	16	17
Definicion de modelo a usar										
Modificacion y correccion del modelo				■	■					
Generar nuevo Launch del modelo					■				■	
Modificacion del Launch						■	■			
Comprimir Carpeta de proyecto					■					■
Subir carpeta de proyecto a Git					■				■	
Generacion de trayectoria						■		■		
Modificacion de multiples trayectorias							■	■		
Implementacion de cinematica en el codigo								■	■	
Trayectoria generada a traves de cinematica									■	
Actualizacion de los archivos Git										
Video Pitch										■
Entrega de Proyecto										■

[1]Ortiz Sánchez, Eduardo Uriel. (2017). "Desarrollo de un solucionador de cinemática inversa para la manipulación del robot Golem-III". (Tesis de Licenciatura). Universidad Nacional Autónoma de México, México. Recuperado de <https://repositorio.unam.mx/contenidos/113572>

[2]Icaro.eii.us.es, 2022. [Online]. Available: [http://icaro.eii.us.es/descargas/Tema%20\\_4\\_%20parte\\_4\\_y\\_ultima.pdf](http://icaro.eii.us.es/descargas/Tema%20_4_%20parte_4_y_ultima.pdf)

[3]"Ejercicios Resueltos Cinematica Inversa\_matlab", 1library.co, 2022. [Online]. Available: <https://1library.co/document/y9r4vrjy-ejercicios-resueltos-cinematica-inversa-matlab.html>.

[4]"UNIDAD 02: FUNDAMENTOS MATEMÁTICOS Y FÍSICOS. (2da parte) - ppt video online descargar", Slideplayer.es, 2022. [Online]. Available: <https://slideplayer.es/slide/2261797/>. [5]"DR. José Antonio Garrido Natarén - ppt descargar", Slideplayer.es, 2022. [Online]. Available: <https://slideplayer.es/amp/12140206/>.

[6]D. Milanés Hermosilla and A. Castilla Pérez, "Generación de trayectorias para el brazo robótico (ArmX)", Scielo.sld.cu, 2022. [Online]. Available: [http://scielo.sld.cu/scielo.php?script=sci\\_arttext&pid=S1815-59282016000300006](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1815-59282016000300006).

[7]Eudim.uta.cl, 2022. [Online]. Available: [http://www.eudim.uta.cl/rmendozag/courses/fundamentos\\_robotica/lectures/cinematica\\_inversa.pdf](http://www.eudim.uta.cl/rmendozag/courses/fundamentos_robotica/lectures/cinematica_inversa.pdf).

## Anexos

En este apartado se dejará un cronograma en el que se demuestre las actividades y logros alcanzados, iniciando en la semana 8 y finalizando en la semana 12