

PROYECTO FINAL GUNDAM RX78



Integrantes:

Gloris Cedeño
Azael Batista
Iván Saavedra
Samuel Olmedo
Elena Linares
Jorge Ruíz

Deyka Jordan
Isaac Perdomo
Mario Villarreal
Franger Lezcano
Alexis García

Contenido

Investigaciones Preliminares	3
GUNDAM RX78	3
Historia.....	3
Connotación en la Cultura Japonesa	3
Gundam Challenge.....	3
Objetivo del Proyecto	4
Bases Del Proyecto.....	4
Definición de Metodología de Trabajo	5
Definición de Roles	5
Creación de Sprint Backlog	5
Definición de Stakeholders	6
Implementación del Código Base y Desarrollo del Proyecto.....	8
Paquetes de ROS para robots de GUNDAM	8
Configuración del Ambiente de Trabajo	8
Instalación de Mesh y URDF	9
Visualizar modelo URDF.....	9
Mapa de Panamá	10
Ejecución del Gazebo	11
Para controlar los ángulos de las articulaciones.....	11
joint_trajectory_client_csv.py	13
Video de YouTube:.....	13
Referencias.....	14

Investigaciones Preliminares

GUNDAM RX78

Historia

El RX-78-2 Gundam (en japonés: RX-78-2 ガンダム, Hepburn: Āru Ekkusu Nanajū Hachi no Ni Gandamu) es un robot (mecha) tripulado ficticio, introducido en 1979 en la serie de anime Mobile Suit Gundam de Yoshiyuki Tomino y Sunrise. [1]

El RX-78-2 fue el segundo de los tres mobile suits prototipos Gundam construidos en Side 7 en el año UC 0079, como parte del Proyecto V de la Federación Terrestre. Construido para complementar a la unidad de largo alcance Guntank y la unidad de apoyo de alcance medio Guncannon, el Gundam fue diseñado para ser una máquina de corto alcance, altamente móvil con avanzado armamento más allá de los usados por las máquinas Zeón. [2]

Connotación en la Cultura Japonesa

Se han realizado investigaciones que revelan cómo el personaje RX-78-2 Gundam se presenta como un héroe en el contexto de la cultura japonesa presentada en el anime Mobile Suit Gundam.

Los resultados de este estudio son que el personaje RX-78-2 es una encarnación o una representación de un héroe japonés del pasado, a saber, el samurái que fue revivido por el director y el diseñador en una historia ficticia en forma de un robot gigante; en el que hay principios de vida que hasta el día de hoy los japoneses siguen. Además, hace connotación al espíritu de un samurái o lo que se llama el espíritu de Bushido para seguir luchando incansablemente en defensa de la justicia, principio que se muestra en la figura del personaje ficticio RX -78-2 Gundam como héroe de Japón. [3]

Gundam Challenge

Durante el año 2009 apareció una estatua de GUNDAM de tamaño natural. La vista de la estatua con los pies firmemente plantados en el suelo capturó los corazones de muchas personas y representó la posibilidad de que los sueños se hicieran realidad.

En el año 2019, GUNDAM, que celebra su 40 aniversario conmoverá Japón y conmoverá al mundo.

Se creó un desafío para mover el GUNDAM de tamaño real de 18 m de altura.

Objetivo del Proyecto

El objetivo es encontrar una forma que permita a Gundam tener un combate, manteniendo la estabilidad del mismo.

Bases Del Proyecto

Sistema Operativo: Linux, Ubuntu, Sistema Operativo Robótico.

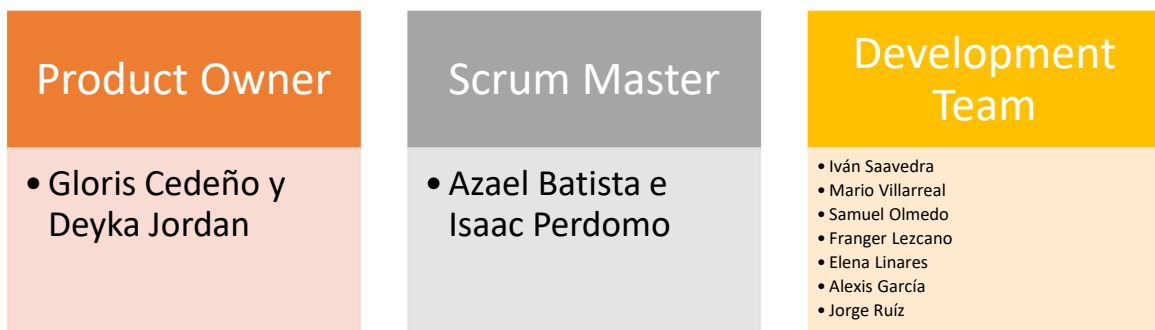
Robot Modelo: https://github.com/gundam-global-challenge/gundam_robot

Referencias: https://github.com/HumaRobotics/darwin_gazebo

Definición de Metodología de Trabajo

En este espacio nos reunimos para organizar la forma en la que nos dividiremos la carga de trabajo, eligiendo la metodología de desarrollo SCRUM, ya que esto nos permite realizar un desarrollo ágil en el tiempo de entrega estimado, cumpliendo con el siguiente diagrama de trabajo:

Definición de Roles



Creación de Sprint Backlog

Como siguiente punto desarrollamos las tareas de las que estaría compuesta el Backlog.

Backlog:

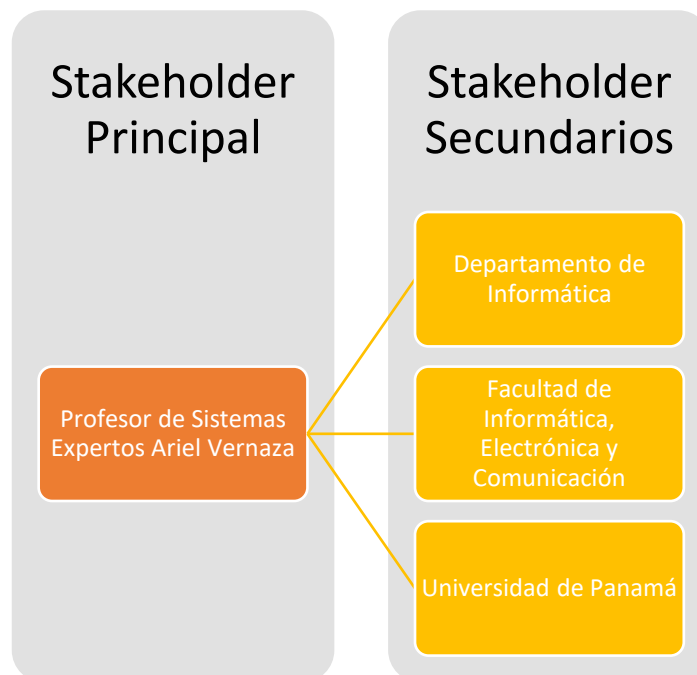
- Estudiar la información provista por el Stakeholder Ariel Vernaza
- Realizar Investigación Preliminar
- Definir Metodología de Trabajo
- Configurar el ambiente de trabajo
- Desarrollar el Código de Programación para ROS
- Probar el código en una Simulación
- Grabar un video de la simulación de GUNDAM RX78
- Editar el video de la simulación
- Subir a YouTube el video de la simulación

- Crear el Reporte de Avances del Semestral
- Subir al Github el código de programación

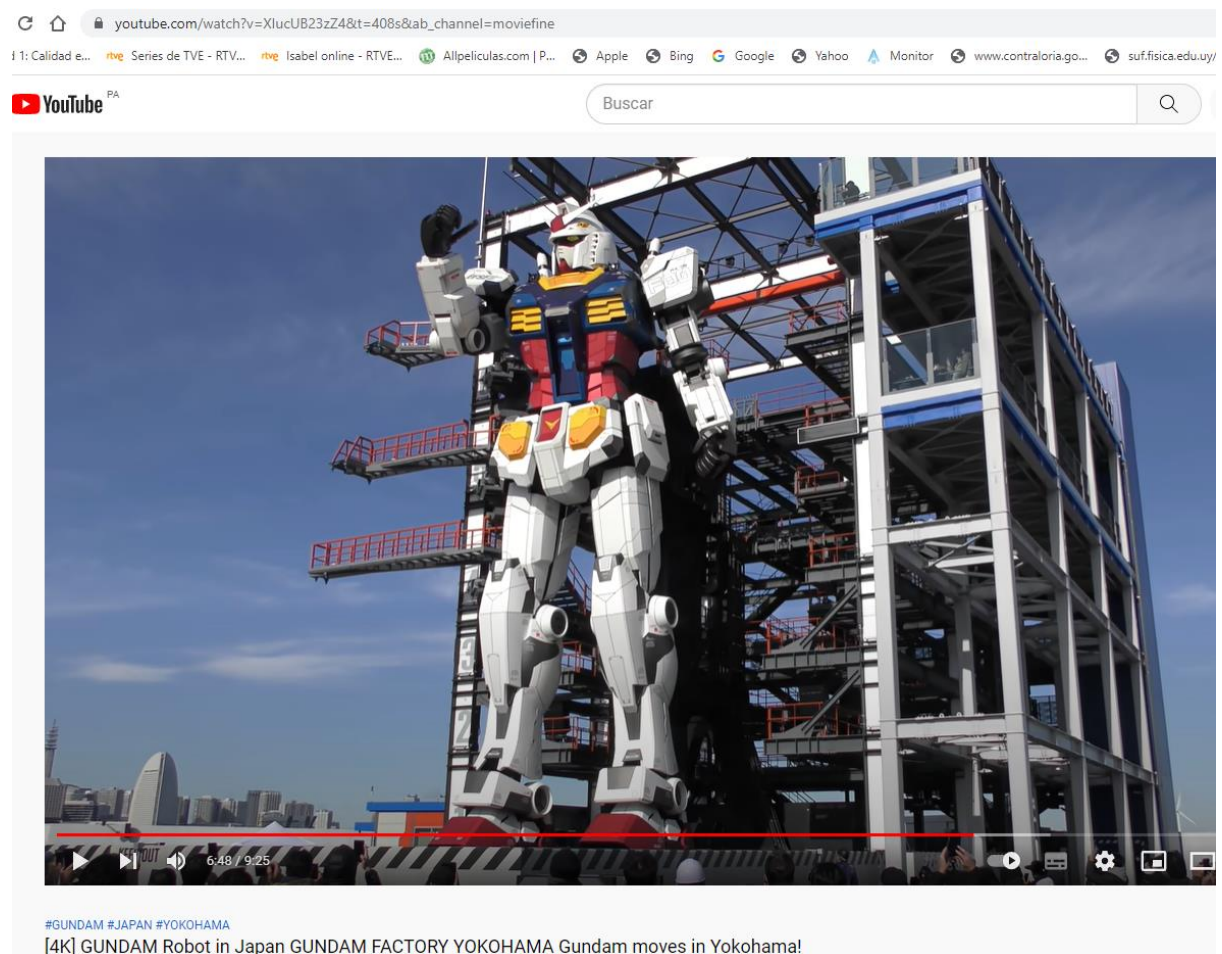


Definición de Stakeholders

Además, definimos nuestros Stakeholders.



Una vez de acuerdo con la metodología comenzamos a desarrollar el backlog, leyendo toda la información provista por nuestro Stakeholder Principal Ariel Vernaza.



Ahora que hemos terminado con la lectura procedemos a realizar la investigación preliminar que se puede observar en este informe.

Procedemos a configurar el ambiente de trabajo Gazebo y a ejecutar el simulador como se puede observar en la sección de desarrollo del proyecto.

Implementación del Código Base y Desarrollo del Proyecto

El sistema operativo utilizado es Ubuntu 18.04. Es necesario tener ROS instalado, con el entorno de trabajo configurado. Los pasos para la instalación y configuración pueden verse en: <http://wiki.ros.org/melodic/Installation/Ubuntu>

Paquetes de ROS para robots de GUNDAM

original: https://github.com/gundam-global-challenge/gundam_robot.git



Configuración del Ambiente de Trabajo

```
$ mkdir -p catkin_ws/src
$ cd catkin_ws
$ wstool init src
$ wstool merge -t src https://raw.githubusercontent.com/gundam-global-challenge/gundam_robot/.gundam.rosinstall
$ wstool update -t src
$ source /opt/ros/$ROS_DISTRO/setup.bash
$ rosdep install -y -r --from-paths src --ignore-src
$ catkin build
$ source devel/setup.bash
```


Instalación de Mesh y URDF

Primero, descargamos el archivo Gundam Collada (p. ej., GGC_TestModel_rx78_20170112.DAE) en el directorio gundam_rx78_description.

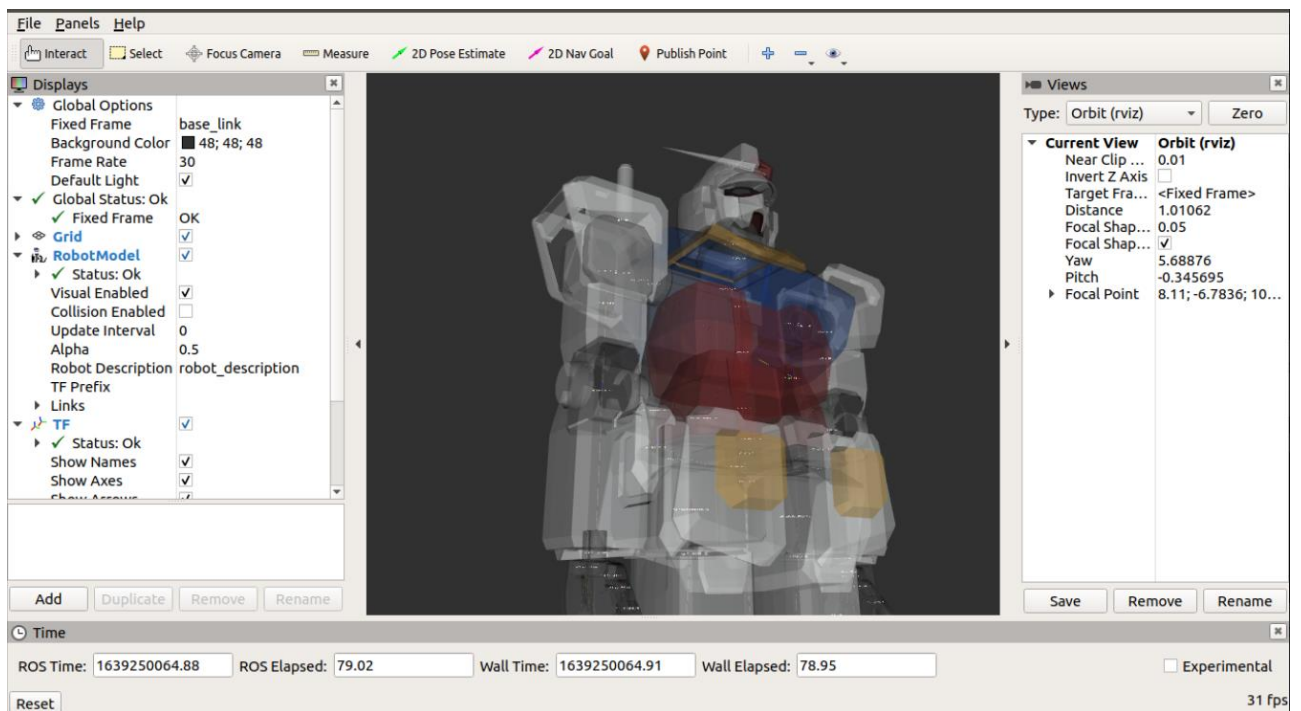
Luego, ejecutamos el archivo `./scripts/dae_to_urdf.py` con el nombre del archivo descargado como argumento. Esto creó archivos de malla en el directorio `meshes/` y creó el archivo URDF en el directorio `urdf/`.

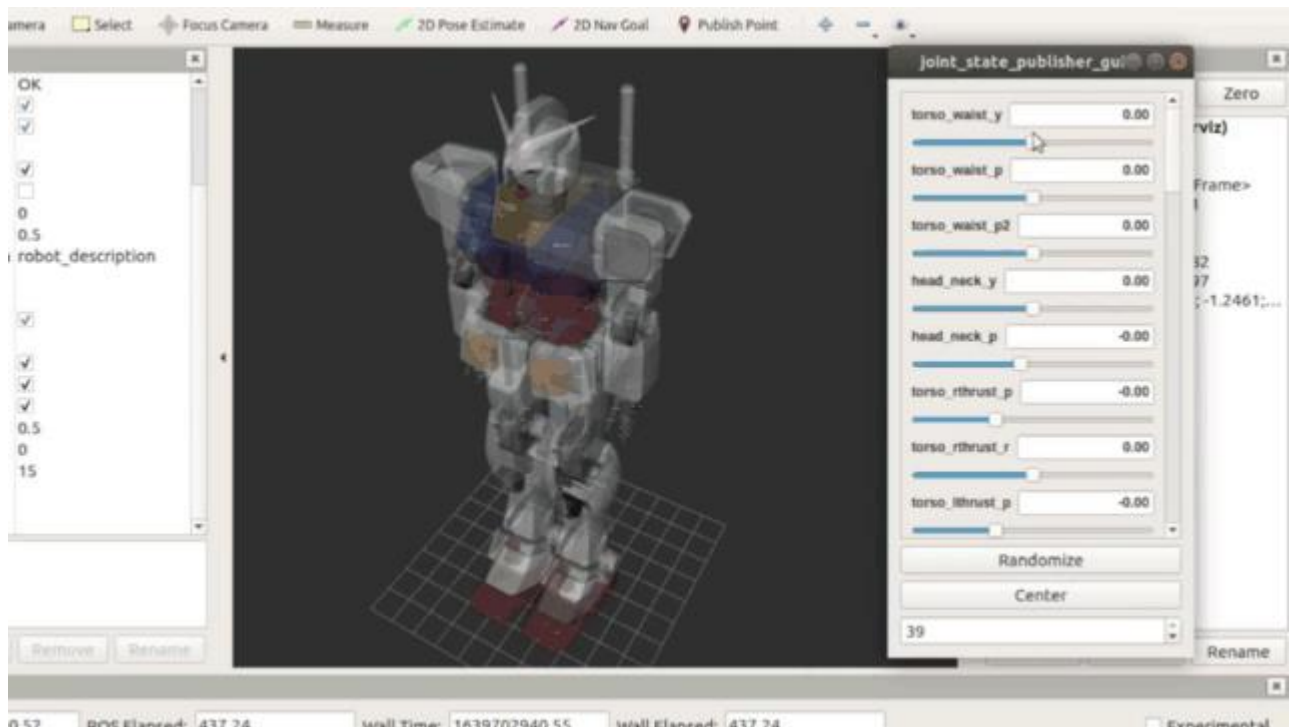
```
$ roscd gundam_rx78_description
$ python ./scripts/ggc_dae_to_urdf.py GGC_TestModel_rx78_20170112.DAE --
write_mesh
$ mv urdf/GGC_TestModel_rx78_20170112.urdf urdf/gundam_rx78.urdf
```

Visualizar modelo URDF

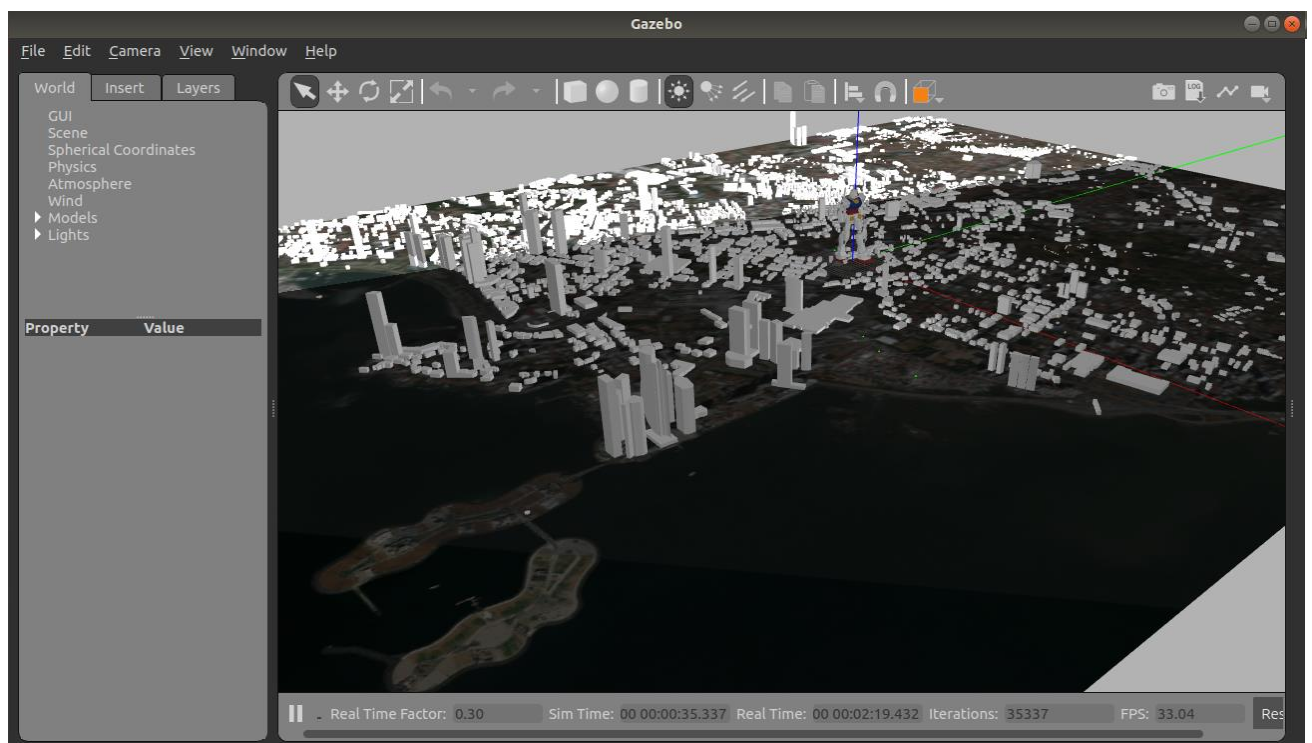
Archivo URDF en rviz, puede usar el archivo `display.launch`.

```
$ roslaunch gundam_rx78_description display.launch
```





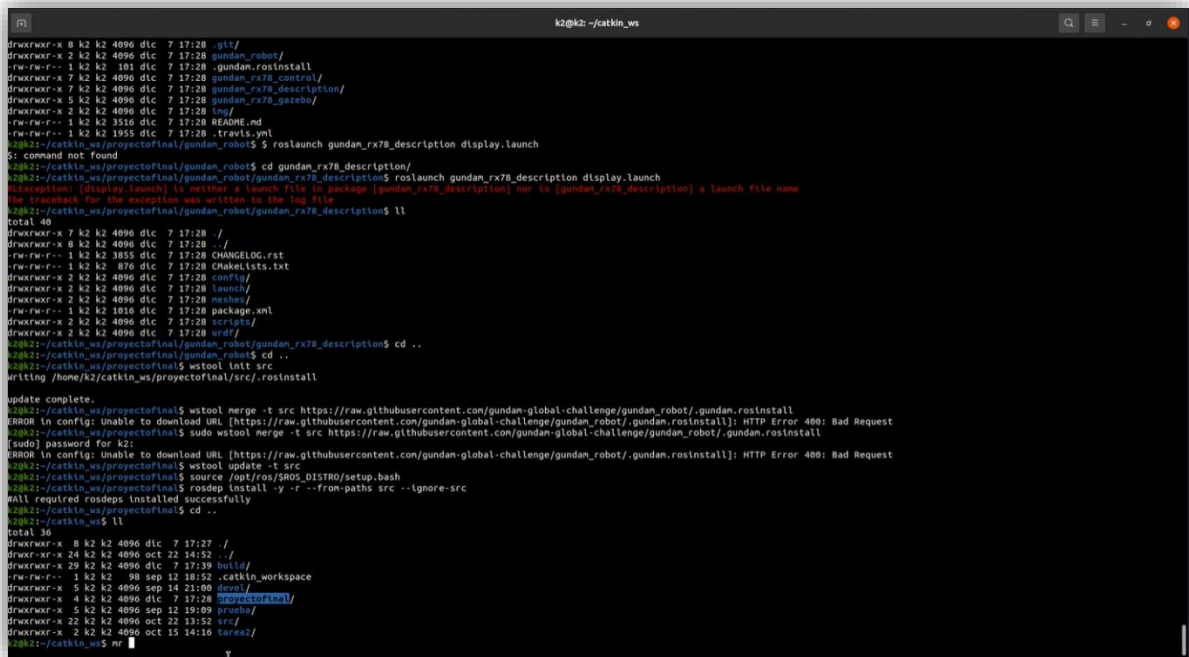
Mapa de Panamá



Ejecución del Gazebo

En esta sección nos disponemos a instalar el ambiente de trabajo Gazebo en nuestro sistema operativo a través de la terminal.

```
$ roslaunch gundam_rx78_gazebo gundam_rx78_world.launch
```



Para controlar los ángulos de las articulaciones

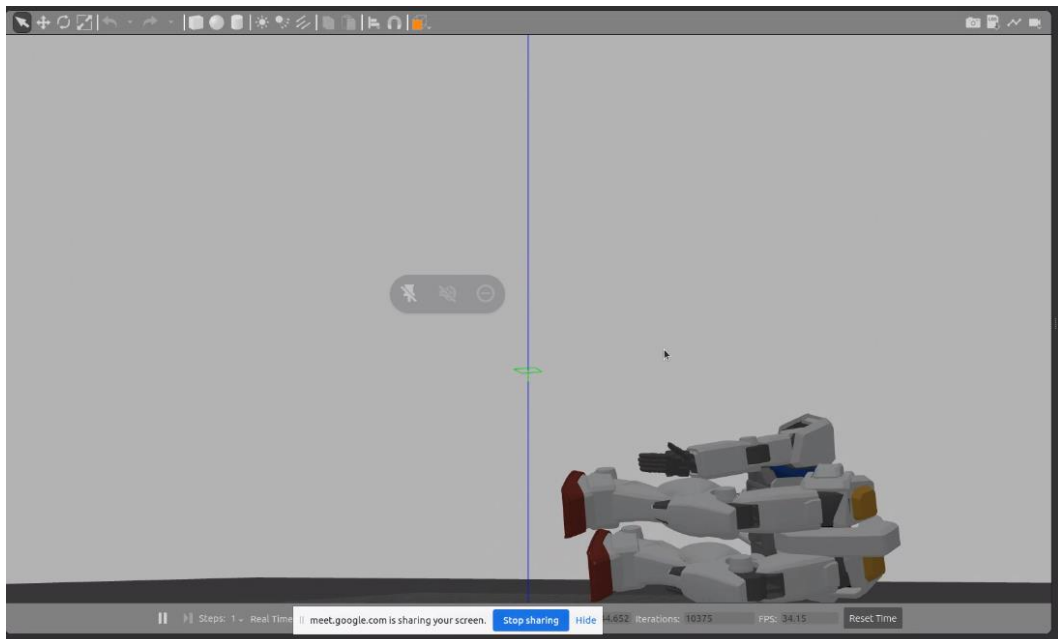
Puede ejecutar un patrón de caminata similar a un "robot" en la simulación

```
$ roslaunch gundam_rx78_gazebo gundam_rx78_walk.launch
# paso
$ roslaunch gundam_rx78_control joint_trajectory_client_csv.py `rospack find
gundam_rx78_control`/sample/csv/step.csv
# Caminar hacia adelante
$ roslaunch gundam_rx78_control joint_trajectory_client_csv.py `rospack find
gundam_rx78_control`/sample/csv/walk-forward.csv
# caminar hacia atrás
$ roslaunch gundam_rx78_control joint_trajectory_client_csv.py `rospack find
gundam_rx78_control`/sample/csv/walk-backward.csv
# caminar a la derecha
$ roslaunch gundam_rx78_control joint_trajectory_client_csv.py `rospack find
gundam_rx78_control`/sample/csv/walk-to-right.csv
# caminar a la izquierda
$ roslaunch gundam_rx78_control joint_trajectory_client_csv.py `rospack find
gundam_rx78_control`/sample/csv/walk-to-left.csv
```

```
# Girar a la derecha
$ rosrund gundam_rx78_control joint_trajectory_client_csv.py `rospack find
gundam_rx78_control`/sample/csv/turn-right.csv
# Girar a la izquierda
$ rosrund gundam_rx78_control joint_trajectory_client_csv.py `rospack find
gundam_rx78_control`/sample/csv/turn-left.csv
# Levantar los brazos y caminar
$ rosrund gundam_rx78_control joint_trajectory_client_csv.py `rospack find
gundam_rx78_control`/sample/csv/up.csv
```

Realizamos las primeras pruebas con el robot. Pero el gran reto que se tuvo fue evitar que el robot cayese al vacío, apenas se abre el entorno.

Simulación:



Comandos:

```
bash: rosrund: command not found
$ rosrund gundam_rx78_control joint_trajectory_client_csv.py `rospack find
gundam_rx78_control`/sample/csv/turn-right.csv
# Girar a la izquierda
$ rosrund gundam_rx78_control joint_trajectory_client_csv.py `rospack find
gundam_rx78_control`/sample/csv/turn-left.csv
# Levantar los brazos y caminar
$ rosrund gundam_rx78_control joint_trajectory_client_csv.py `rospack find
gundam_rx78_control`/sample/csv/up.csv
```

Tengamos en cuenta que actualmente tenemos varias limitaciones en esta simulación, solo tenemos un controlador de posición etc.

También se pueden encontrar archivos de control de movimiento de muestra en el directorio *gundam_rx78_control/sample*.

[joint_trajectory_client_csv.py](#)

Usando el control de gundam con *joint_trajectory_client_csv.py* e ingresando un archivo .csv

Este utilizará los patrones indicados en el archivo para simular el movimiento del gundam según los valores de los ángulos de cada componente.

```
# mover la parte superior del cuerpo  
$ rosrund gundam_rx78_control joint_trajectory_client_example.py
```



Video de YouTube:

<https://youtu.be/hVrQR7I3-CA>

Referencias

- [1] «Wikipedia.com,» 9 Noviembre 2022. [En línea]. Available: [https://en.wikipedia.org/wiki/Gundam_\(fictional_robot\)](https://en.wikipedia.org/wiki/Gundam_(fictional_robot)). [Último acceso: 2022 Diciembre 2022].
- [2] G. Fandom, «gundam.fandom.com,» [En línea]. Available: https://gundam.fandom.com/es/wiki/RX-78-2_Gundam#Historia. [Último acceso: 2022].
- [3] W. S. Pramono, «Kajian Semiotika Gundam RX-78-2 Sebagai Hero Dalam Anime Mobile Suit Gundam (1979-1980,» de *UPT Perpustakaan ISI Yogyakarta*, 2017.