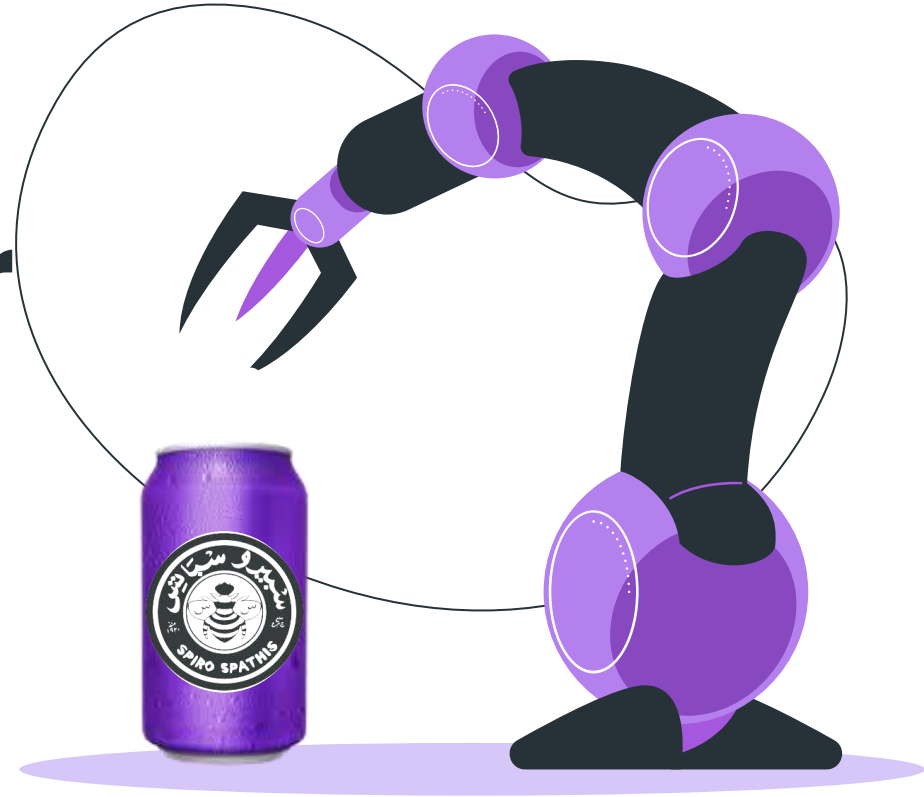


Biomedical Wastes Handler Robot



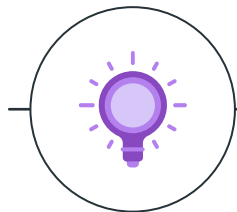
Agenda

<u>Problem definition</u>	Removing Hazardous materials Safely
<u>Proposed Methodology</u>	Automating the process through the proposed pipeline.
<u>Simulation</u>	Testing the implementation on a simulated environment
<u>Challenges</u>	Compatibility challenges
<u>Real life implementation</u>	Implementation using turtlebot3 as our testbed in scaled map
<u>Results and Conclusion</u>	What we found at the end of this journey

Project Timeline

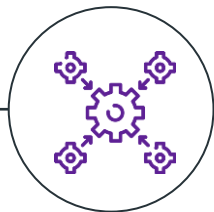
Ideation

Plan the project and divide the tasks



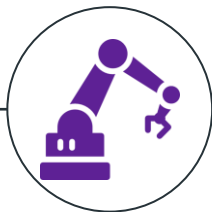
Simulation

Simulate each module



Integration

Integrate the environment



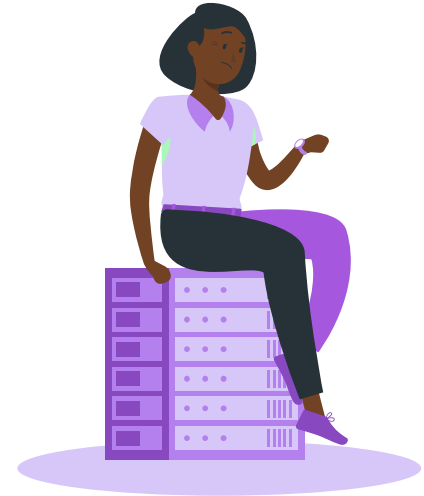
Real Life implementation

Using turtlebot3 as our testbed in scaled map

Problem definition

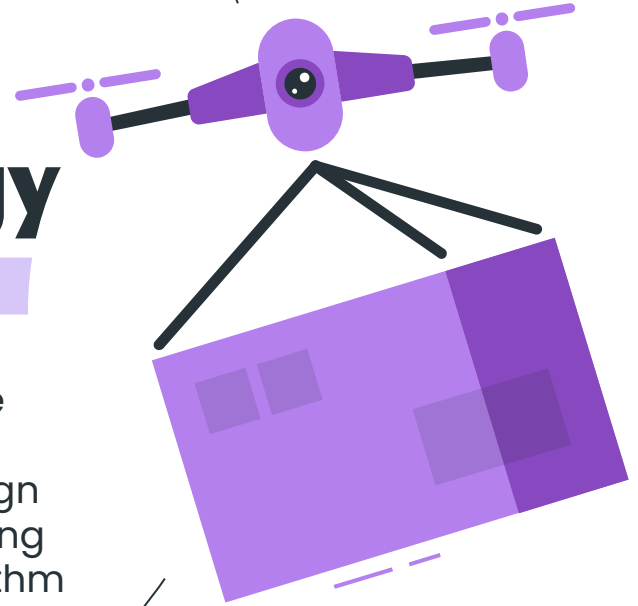
Moving Biomedical Wastes has:

- High risk to human health
- Emergency response time
- Unknown conditions
- Complex and confined spaces
- Efficiency and optimization



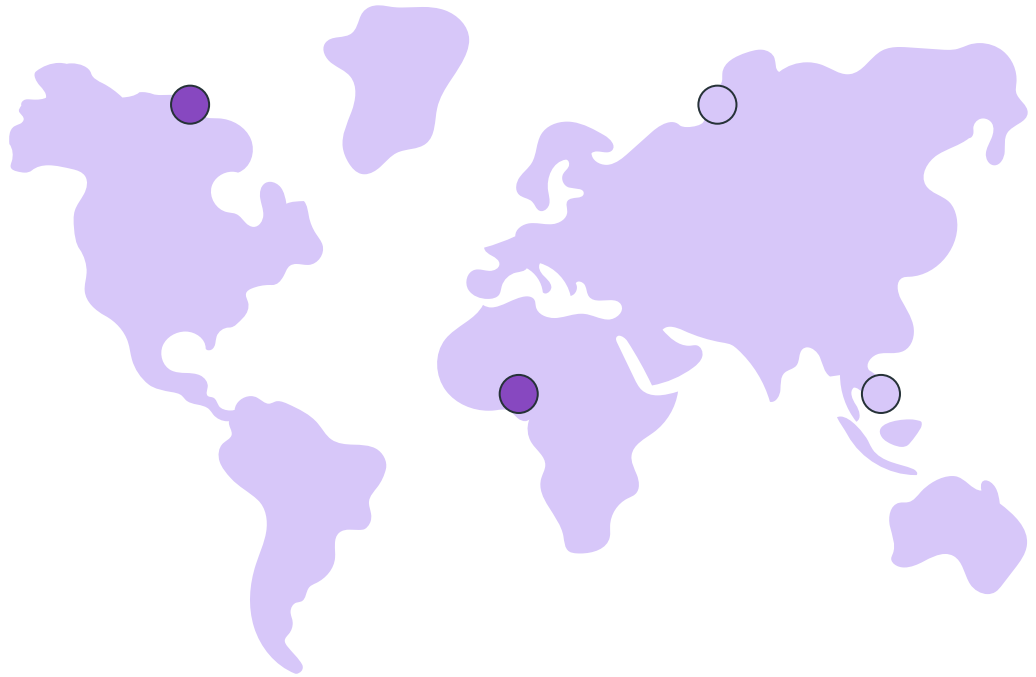
Proposed Methodology

Develop a comprehensive map Implement SLAM for simultaneous localization and mapping to determine the robot's position. Use the A* algorithm for optimal path planning, considering obstacle avoidance. Design container detection using a camera and OpenCV using digital servoing algorithm Implement the DWA algorithm for dynamic obstacle avoidance. Test and evaluate the system's performance in a simulated environment before real-world deployment.

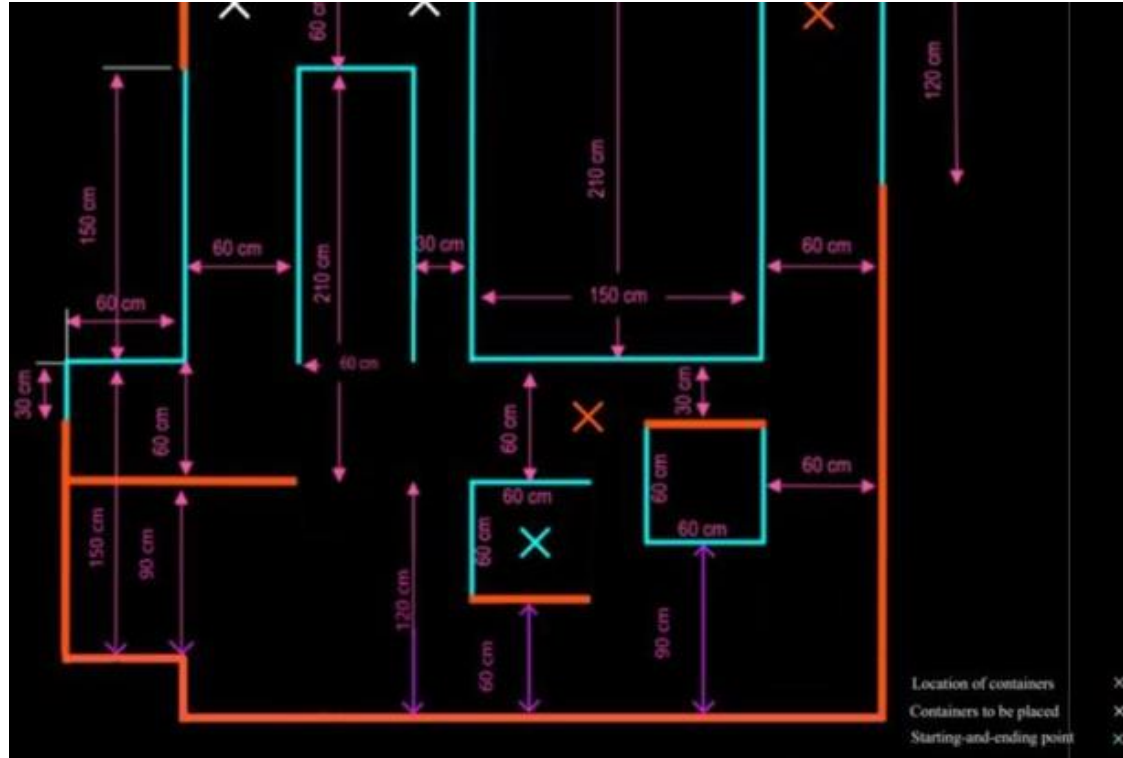


Workspace map

The robot should move from point to point on a map to pick some hazardous materials and place it in its designed place while avoiding dynamic obstacles and narrow paths. The map have keep out zones to be restricted areas that should be considered in the navigation system



MAP



Project Mile stones

1

Navigation

2

Keep out
zones

3

Visual
Servoing

4

Pick and
Place

5

Nodes
Integration

Navigation



Goal detection

Using A* algorithm to plan the way into the goal's points.



Obstacle avoidance

Using navigation 2 to do Slamming online while avoiding dynamic obstacles



Way points planning

Planning sub stations as a series of points to cover a path

Keep out zones



Red lines detection

Doing image processing to detect red lines or red rectangles and getting the dimensions and the positions of them.

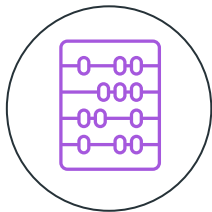
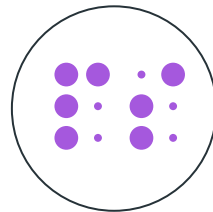


Image to map conversion

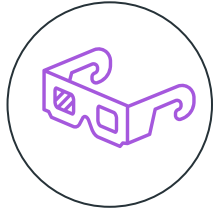
Converting the maps from image bird view into 2d maps.



Cost map update

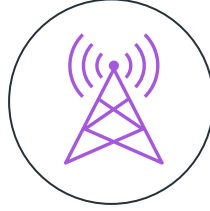
Publishing the red lines as dynamic obstacles by updating the cost map using keep out zones filters.

Visual servoing



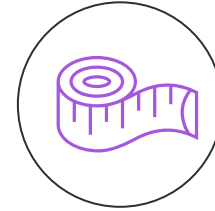
Detecting mask

Detecting Blue colored rectangles with specific aspect ratio. to initialize the servoing task. The mask focuses on a specified ROI.



Topic publishing

Publishing the info on a topic to handle the robots movement to maintain the can in the targeted area



Stopping distance

Pooling on the aspect ration of the detected object as an indication of the distance to it to stop and pick it.

Pick and Place



Controlling the angles

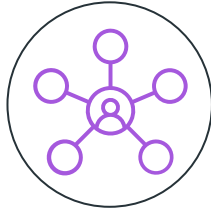
Controlling the angles of the robotic manipulator's by publishing the angles to the motors and the degree of how the gripper is opened.



Pick and place

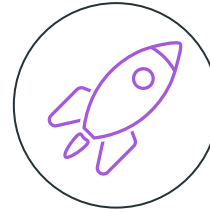
Planning the pick and place angle points to keep the gripper parallel to the object to pick and place it smoothly.

Nodes integration



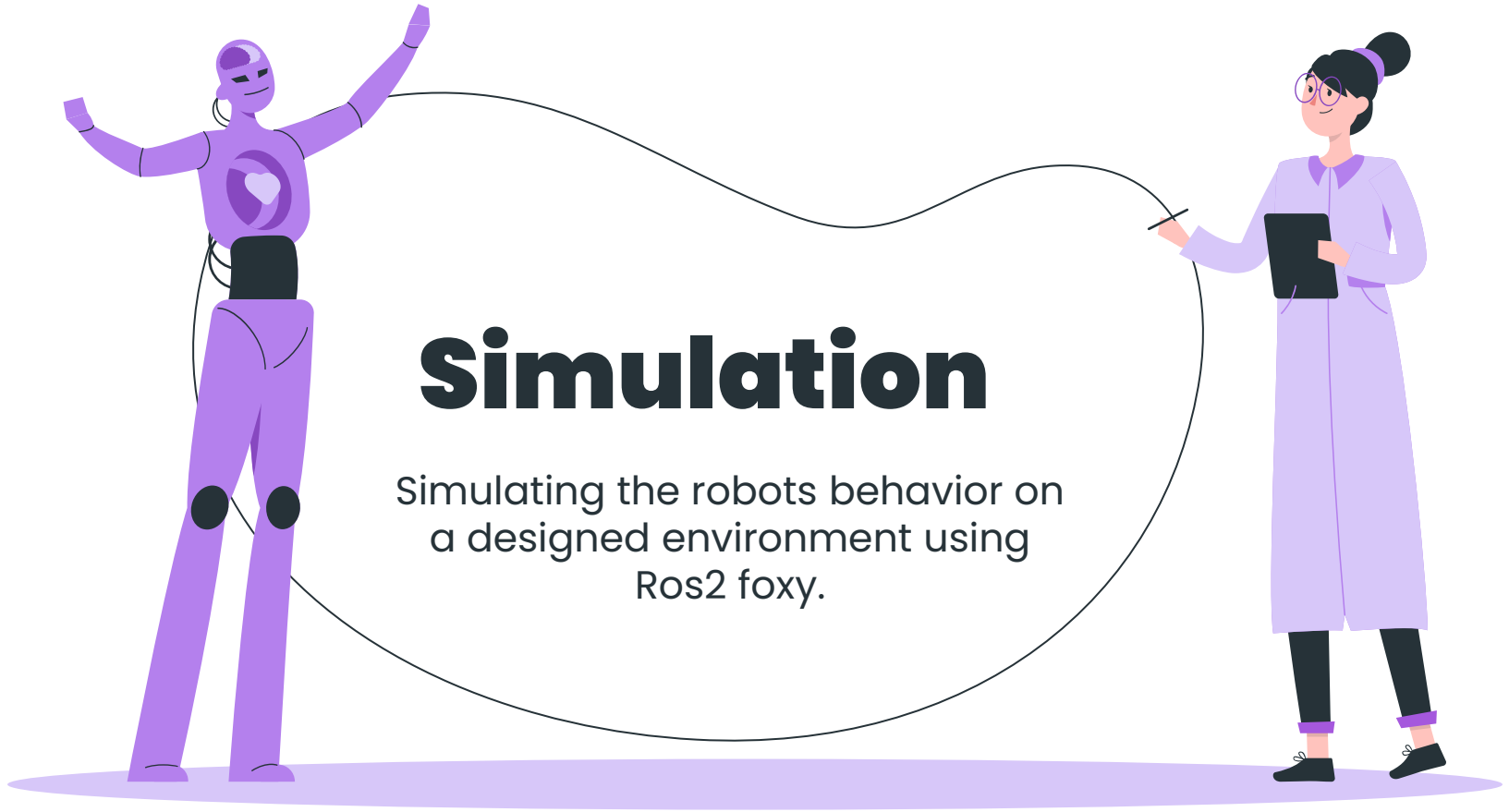
Handling topics

Using the integration
between different
modules through Ros2
topics and actions



Launch files

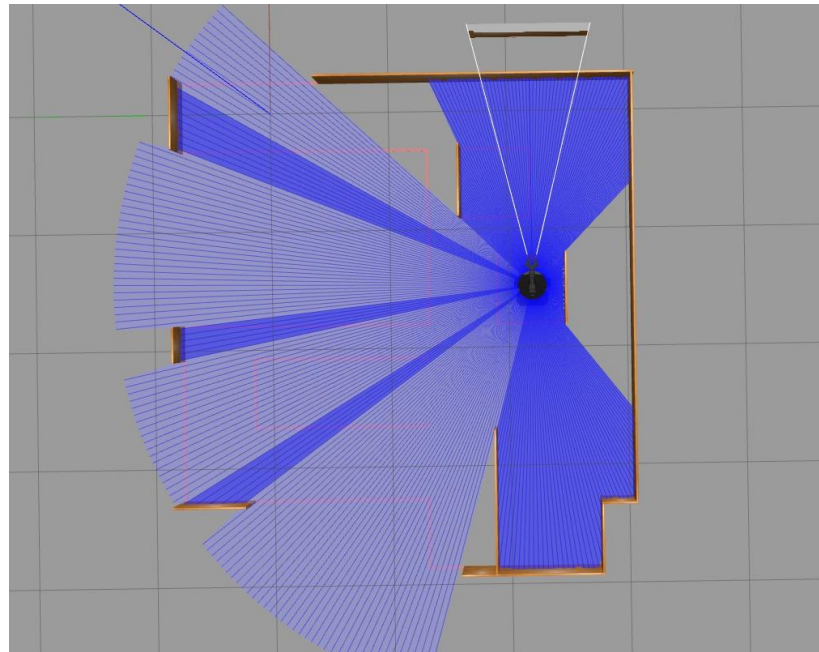
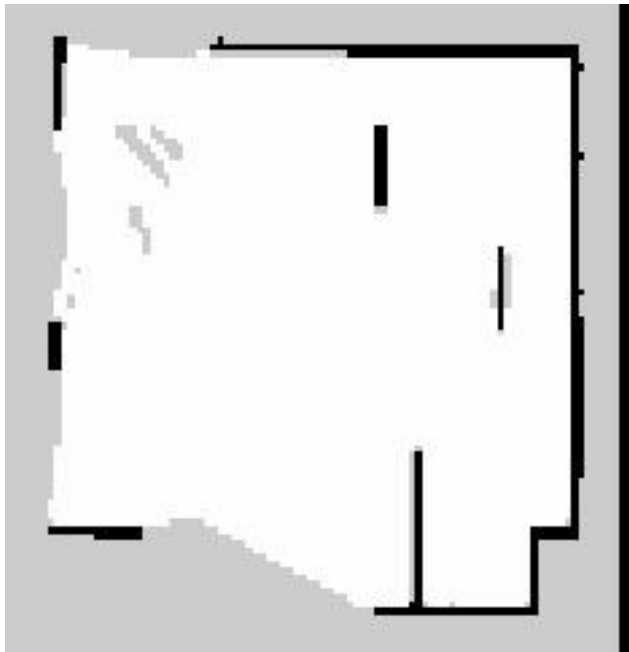
Using launch
files to start
different nodes
and scripts



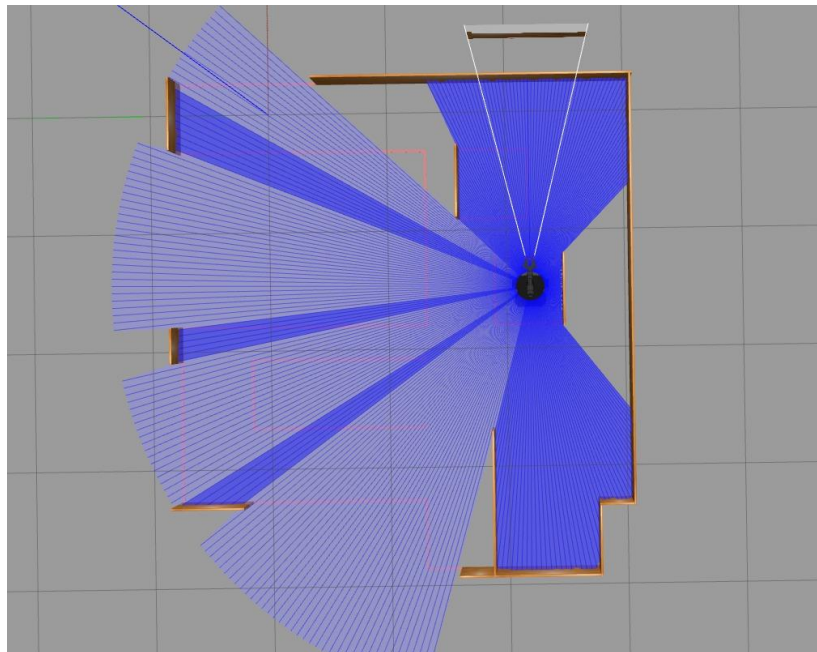
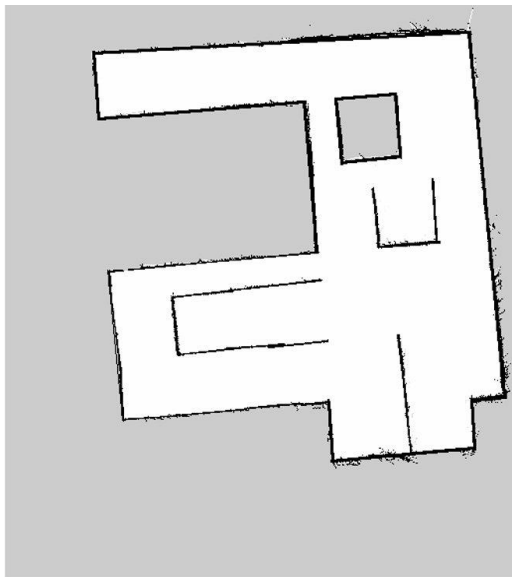
Simulation

Simulating the robots behavior on
a designed environment using
Ros2 foxy.

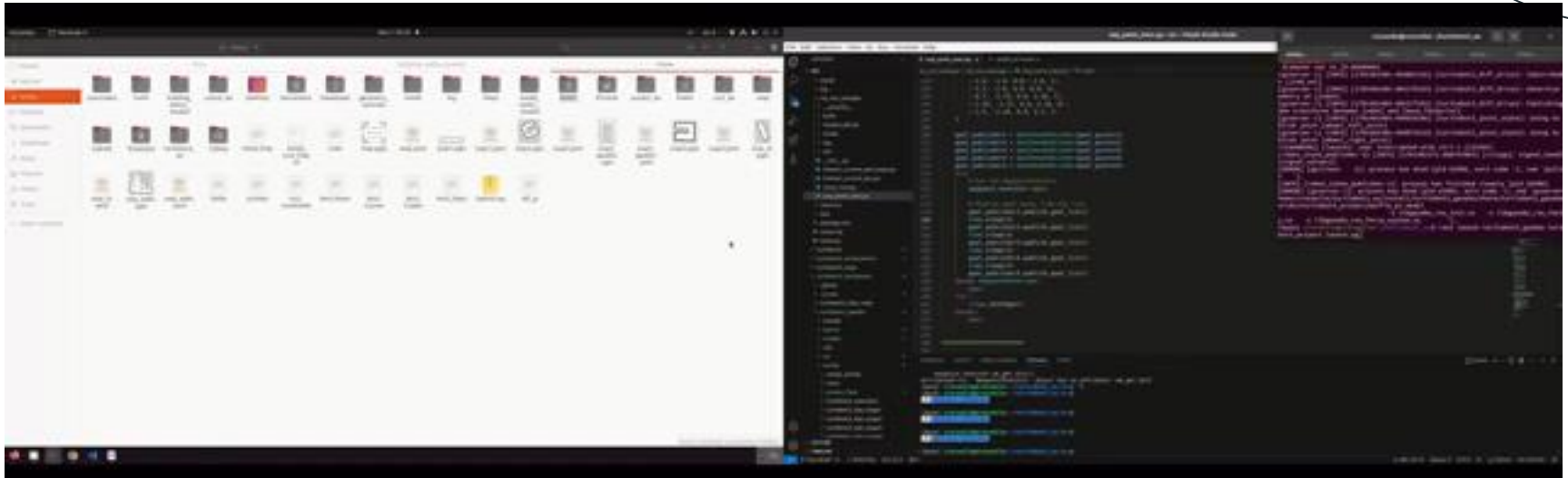
MAP



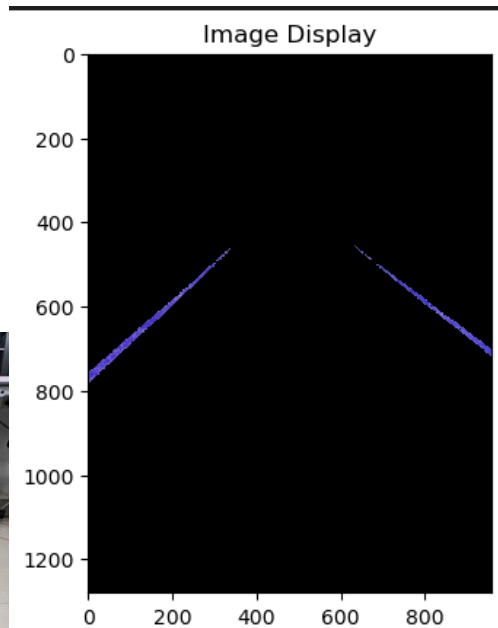
MAP



Way Point Executer

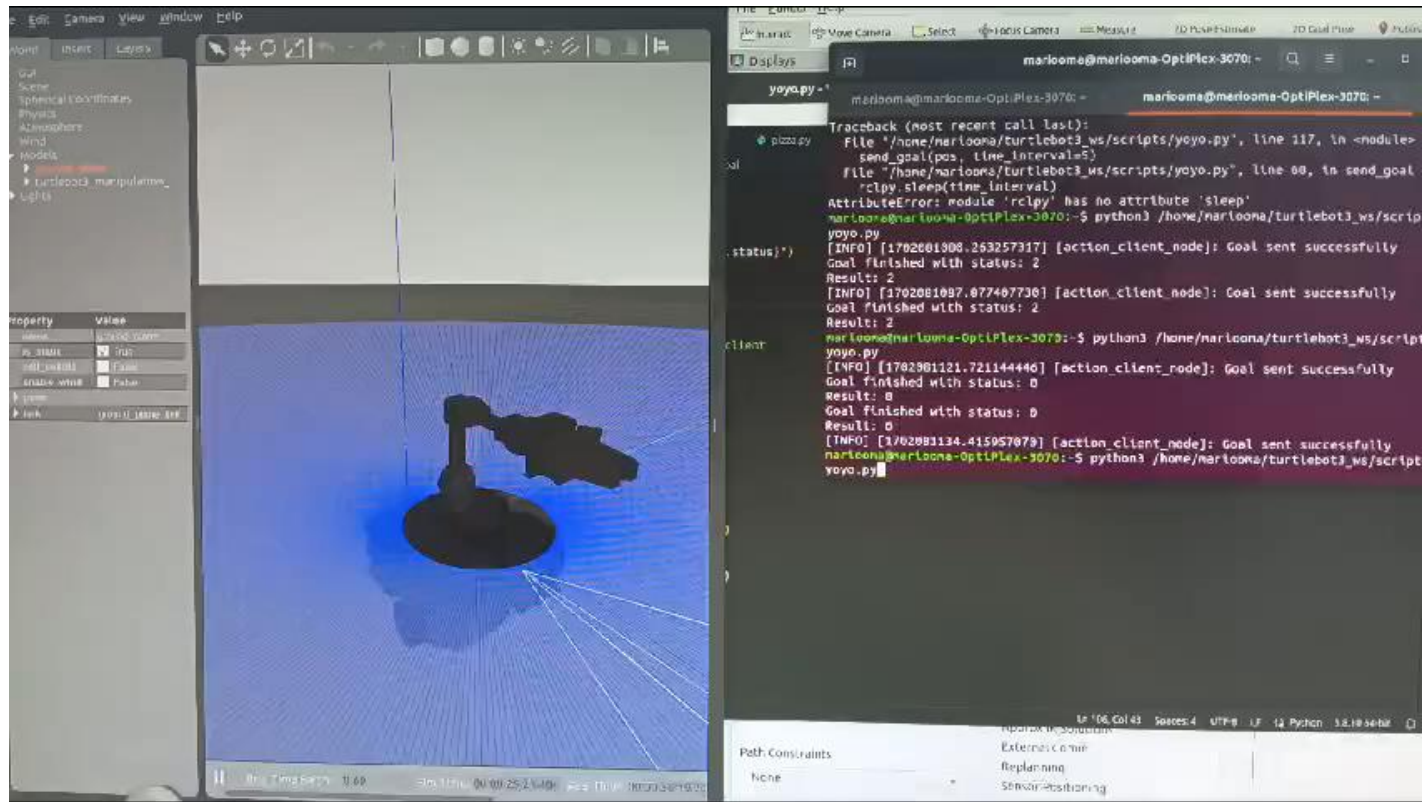


Red line detection

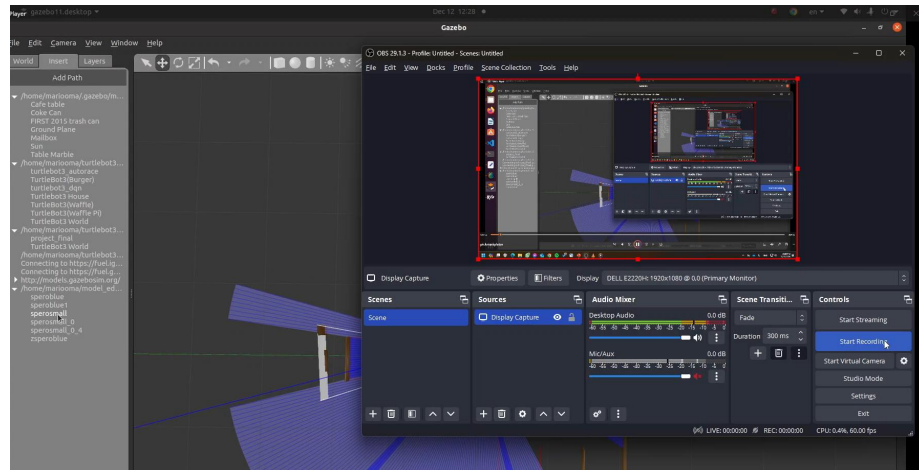
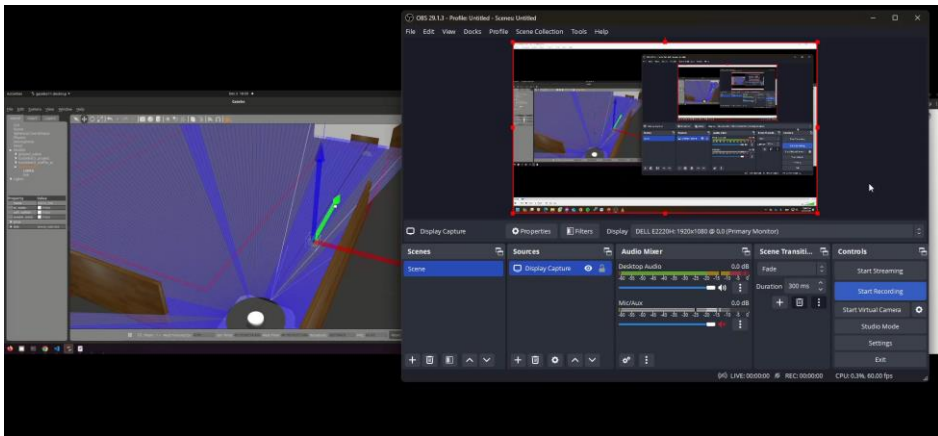
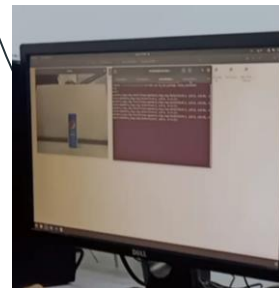


```
... [[841 623]]  
      [[842 623]]]  
rect  
[[841 620]]  
      [[840 620]]  
      [[839 621]]  
      [[838 621]]  
      [[840 621]]]  
[[303 499]]  
      [[ 0 757]]  
      [[303 500]]  
      [[ 0 785]]]  
rect  
[[679 496]]  
      [[959 720]]  
...  
      [[552 105]]  
      [[552 102]]]  
rect
```

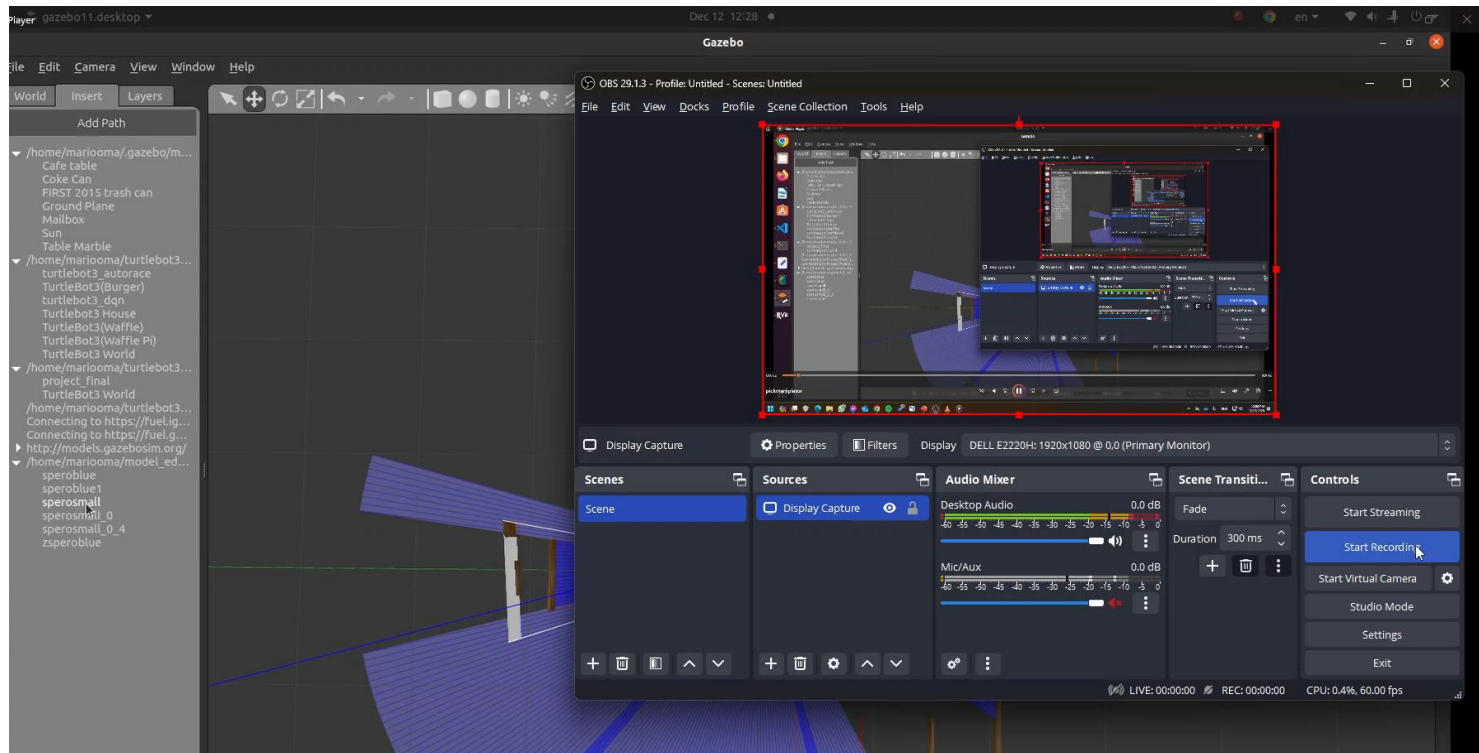
Pick and place



Visual Servoing



Pick the container





Challenges

- ⚠ Ros foxy issues
- 🗺 Cost map
- 🕒 Time Management

General challenges



ROS Foxy Fitzroy is one of the ROS 2 distributions. Issues may arise due to bugs, compatibility problems, or conflicts with other software packages.



Real-time updates of cost maps.



Development timeline may get conflicts.

Navigation Challenges



Nav2 package
simple commander
not supported in foxy



Waypoints not met in
simulation requiring
inflation radius



Waypoints topic in
real life differs from
simulation

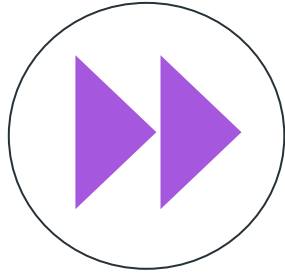


Tried SLAM toolbox
but failed

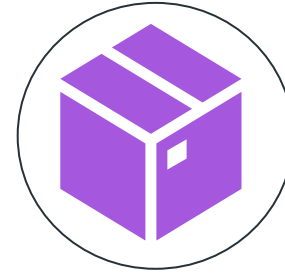


Map rotating and not
giving exact location

Pick and Place challenges

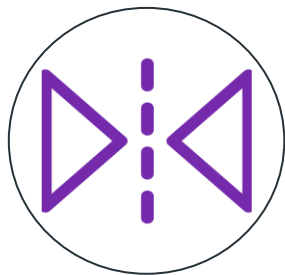


Moveit2 package not
supported in Foxy

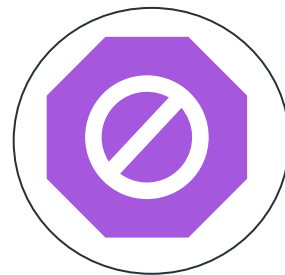


Packages/Dependencies
errors

Visual Servoing Challenges

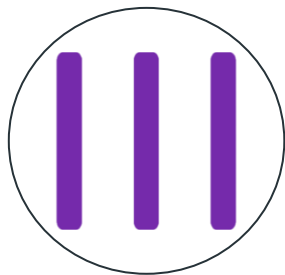


Reflections in can preventing
color masking for visual
servoing



Determining stop point for
picking up the can

Redlines Challenges

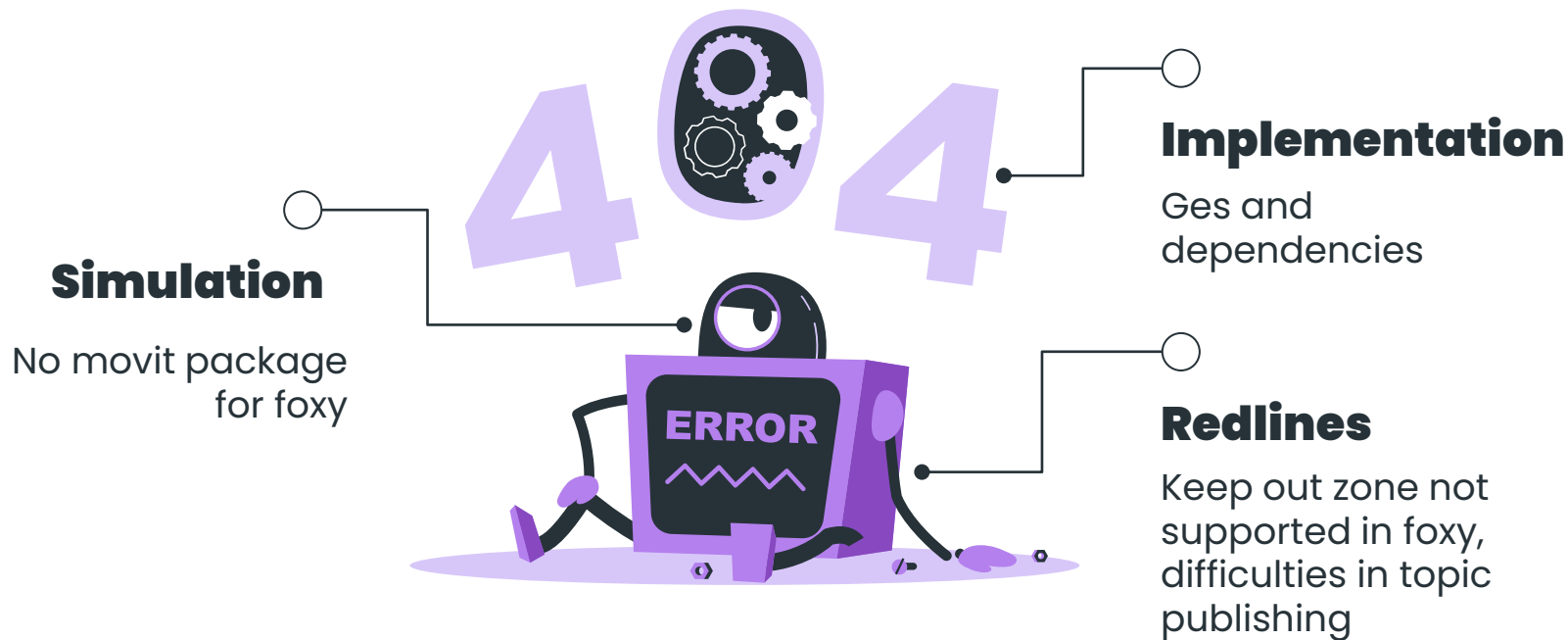


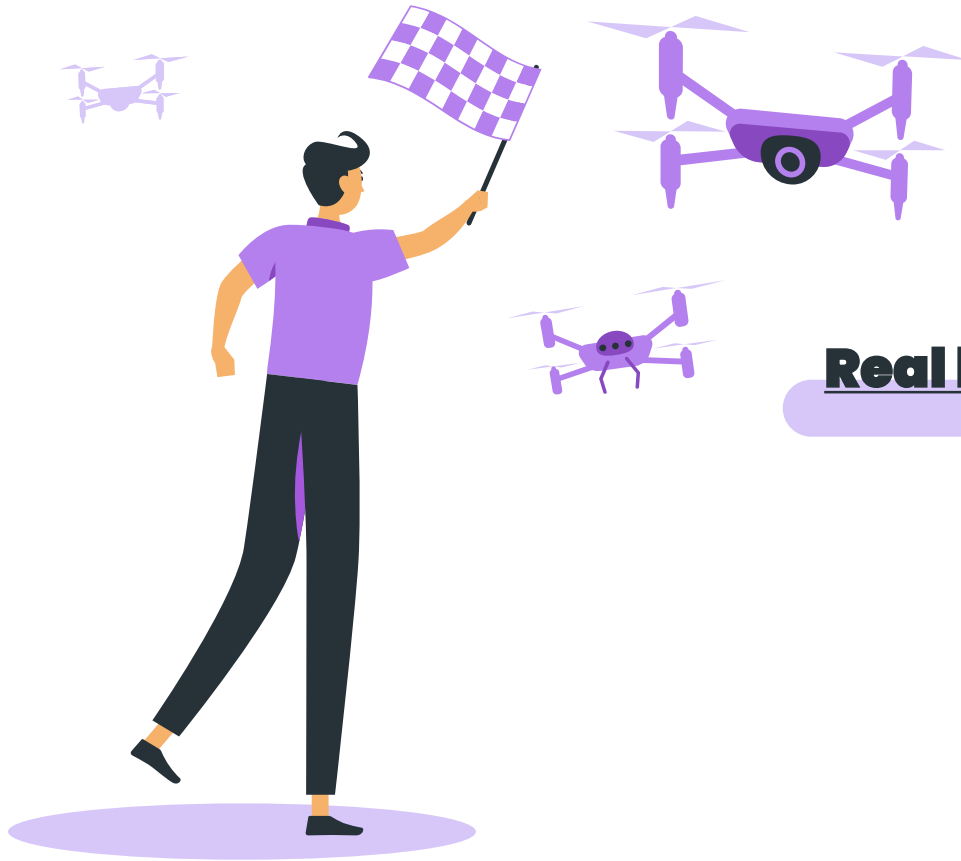
Redlines detected
but not published



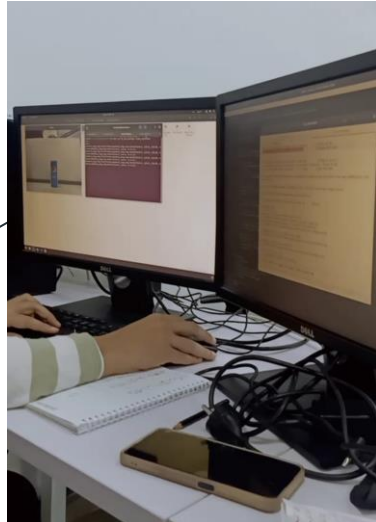
Keepout zone not
supported in foxy

Anatomy Of The Errors





Real life implementation



Color picker

Enter RGB hex code (#):

or

Enter red color (R):

Enter green color (G):

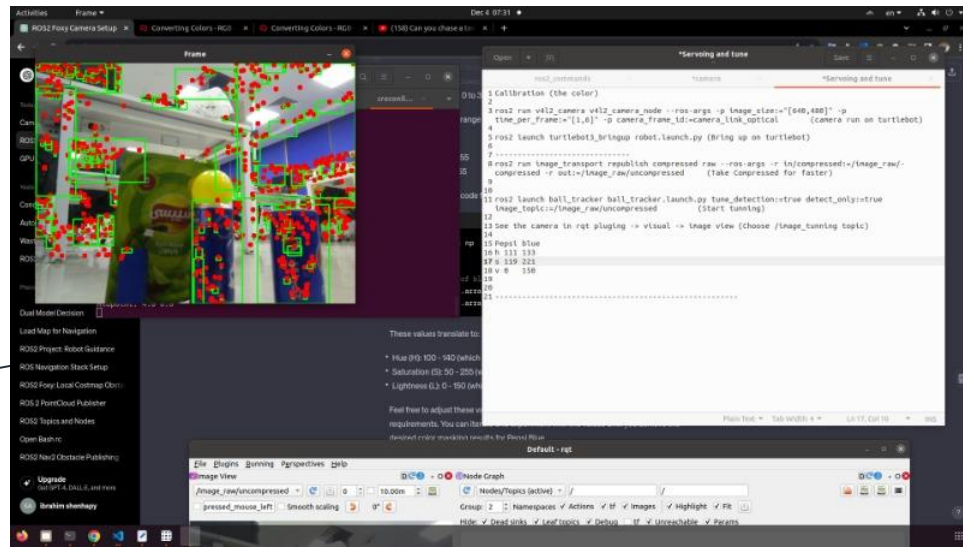
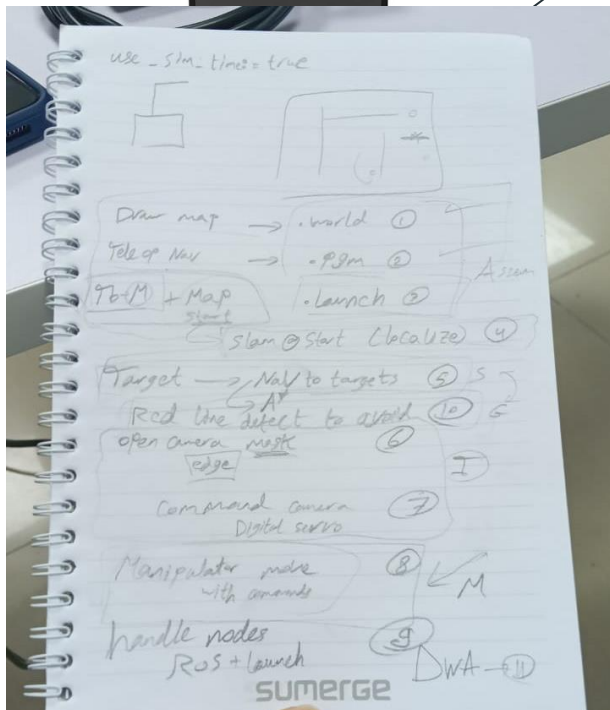
Enter blue color (B):

Hue (H): °

Saturation (S): %

Value (V): %

Color preview:



Path Follow



Pick and place





References

- <https://wiki.ros.org/Documentation>
- <https://emanual.robotis.com/docs/en/platform/turtlebot3/overview/>
- A Concise Introduction to Robot Programming with ROS2 – Taylor and Francis group
- A. R. C, A. P. M, R. K. C and A. Mohanarathinam, "Clinical Waste Storage System with Contamination Prevention Mechanism using UV and Arduino Microcontroller" 2023 International Conference on Inventive Computation Technologies (ICICT), Lalitpur, Nepal, 2023, pp. 1585-1591, doi: 10.1109/ICICT57646.2023.10134053.
- A. Yildirim, H. Reefke, and E. Aktas, "Mobile Robot Automation in Warehouses: A Framework for Decision Making and Integration," Palgrave Studies in Logistics and Supply Chain Management, 1st ed., Palgrave Macmillan Cham, 2023. [Online]. Available: <https://doi.org/10.1007/978-3-031-12307-8>
- A.-T. Nguyen and C.-T. Vu, "Obstacle Avoidance for Autonomous Mobile Robots Based on Mapping Method," in Faculty on Mechanical Engineering, Hanoi University of Industry, Hanoi, Vietnam, 2021. Faculty on Mechanical Engineering, Hanoi University of Industry, Hanoi, Vietnam, 2021. Available: <https://tinyurl.com/5azymbw9>



Thanks!



Thanks Again...

