Deivaprakash K

-Robotics and Automation student (Final Year).

Bio:

A passionate student with budding interest in **Autonomous robots.**Eagerly searching for a opportunity to expand my basic knowledge into profound understanding of creating self-sufficient robotic systems.

Skills :

- Robot operating system 2 (ROS2).
- Gazebo , Rviz.
- Python programming.
- Machine Learning (Have done a virtual internship for 4 months in Feynn Labs.)

Intern Task .

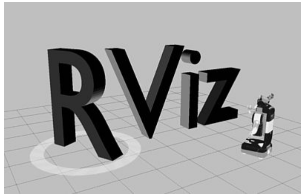
Problem Statement:

- To design and simulate a butler robot that navigates through the restaurant to receive orders and to deliver the food.

Solution approach:

- Selection of simulator.
- Environment creation.
- Selection of robot for Butler purpose.
- Loading the robot into our environment.
- Mapping the environment
- Navigation of the robot in our environment.
- Implementation(logic)as per the requirement.

Selection of simulator:
(GAZEBO,Rviz)



Gazebo:

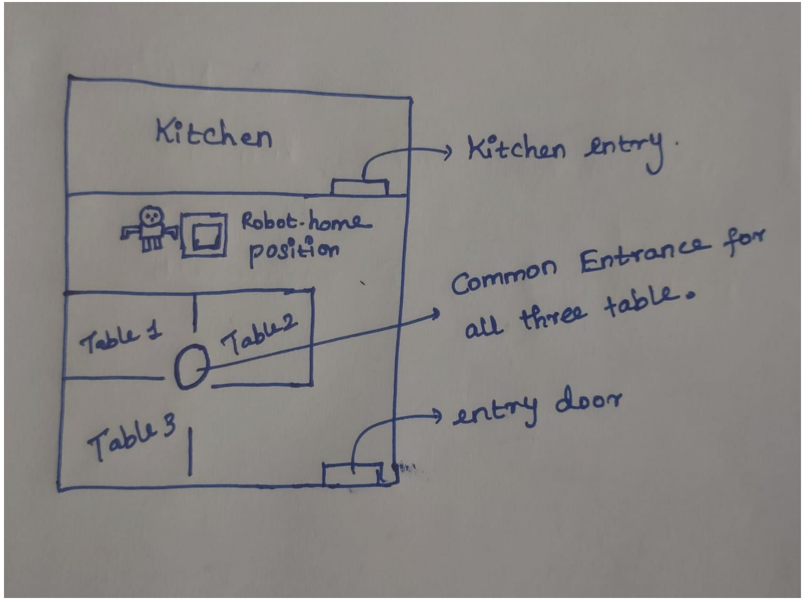
- Gazebo is an open-source 3D robotics simulator.
- With Gazebo, we can create a virtual “world”, and load simulated versions of our robots into it.

Rviz:(ROS visualisation)

- Rviz is a 3D visualisation software tool for robots.
- The purpose of Rviz is to enable you to visualise the state of a robot.

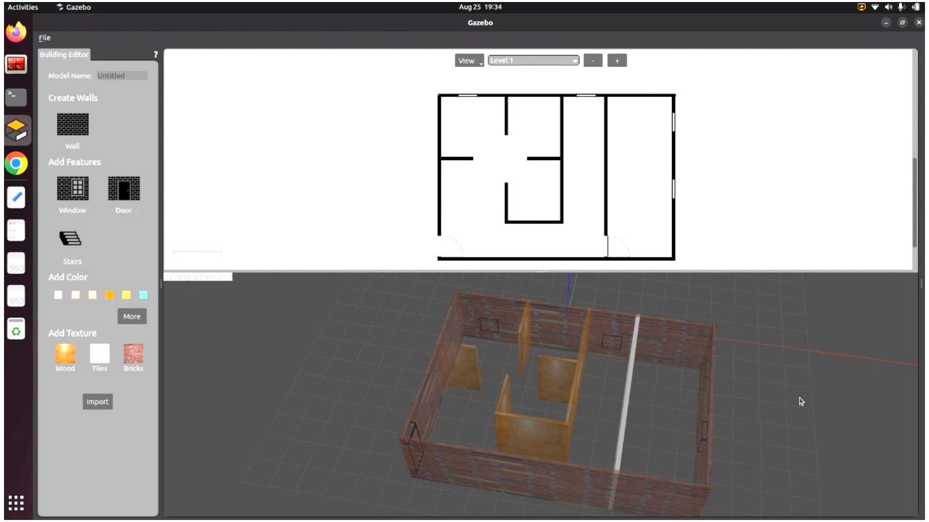
Environment Creation(World):

- For creating an environment, I just imagined a restaurant setup in my mind and roughly sketched on a piece of paper.

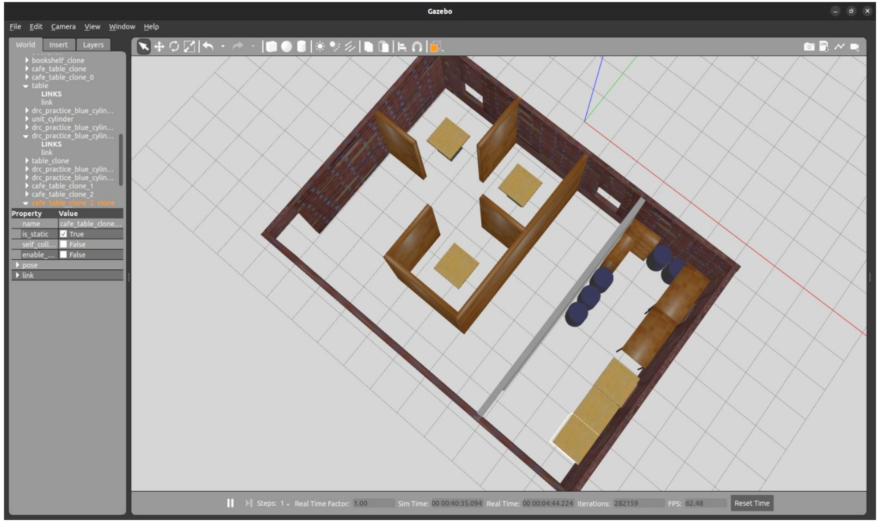


Our customised environment (world file) :

- Created a basic Layout structure in **Gazebo simulator** with the reference of my basic sketch.



Model Name: Untitled



bookshelf_clone
cafe_table_clone
cafe_table_clone_0
table
LINKS
link
drc_practice_blue_cylin...
unit_cylinder
drc_practice_blue_cylin...
drc_practice_blue_cylin...
LINKS
link
table_clone
drc_practice_blue_cylin...
drc_practice_blue_cylin...
cafe_table_clone_1
cafe_table_clone_2
cafe_table_clone_2_clone

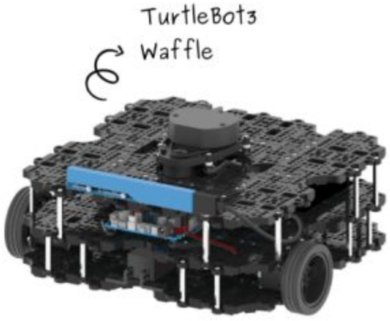
Property	Value
name	cafe_table_clone...
is_static	☑ True
self_coll...	□ False
enable_...	□ False
pose	
link	

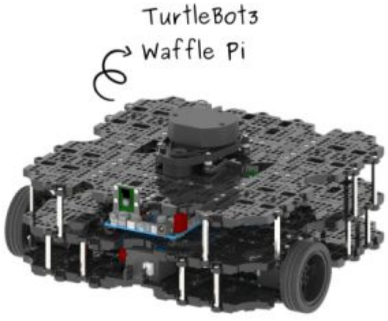
Steps: 1 Real Time Factor: 1.00 Sim Time: 00 00:40:35.094 Real Time: 00 00:04:44.224 Iterations: 282159 FPS: 62.48

Selection of a robot for Butler purpose:

- As per my assigned task,I have selected Turtlebot 3 Robot.
- Since it has the largest community service for better troubleshooting purposes.

TURTLEBOT3

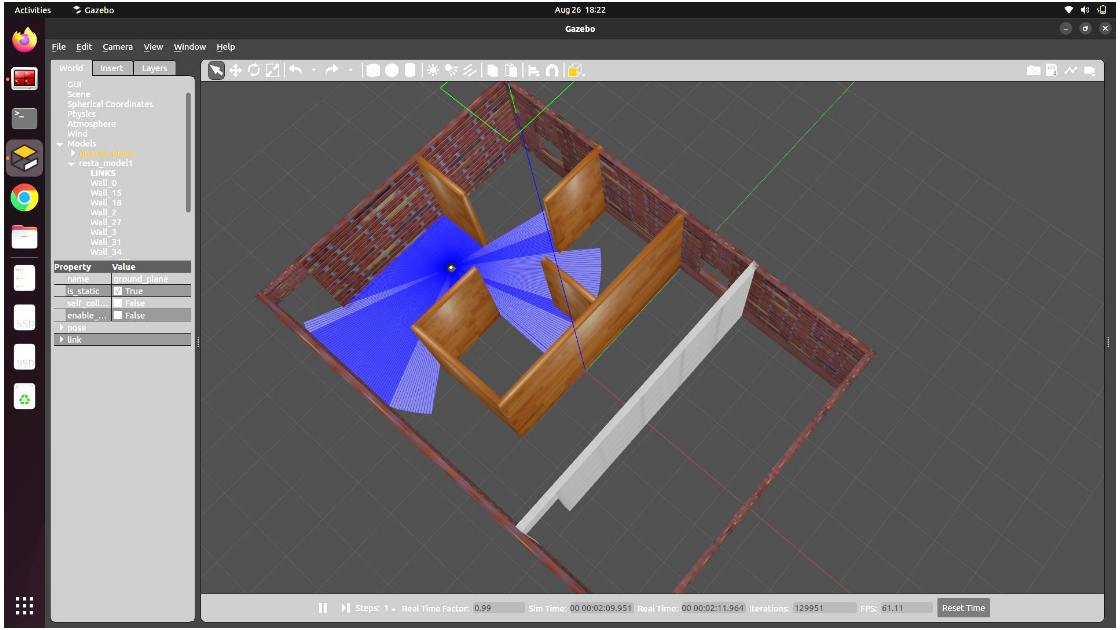




Loading the robot into our environment :

- For Loading the turtlebot 3 robot , we can clone the necessary executables from the official github page of turtlebot.

- Then we can choose the model type of turtlebot and load them into our gazebo world.



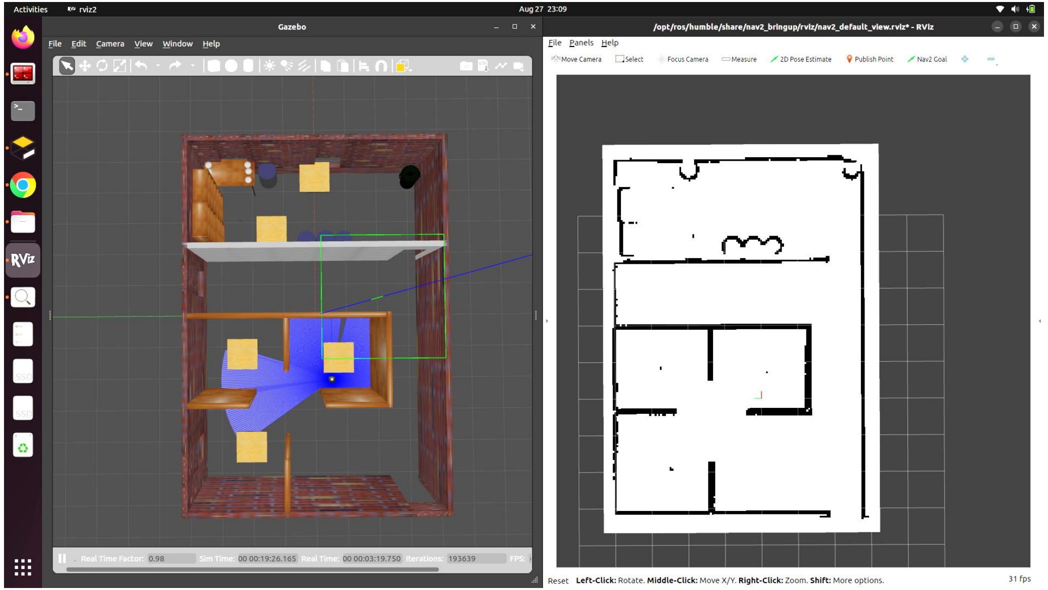
Property	Value
name	ground_plane
is_static	☑ True
self_coll...	□ False
enable_...	□ False

Steps: 1 Real Time Factor: 0.99 Sim Time: 00 00:02:09.951 Real Time: 00 00:02:11.964 Iterations: 129951 FPS: 61.11

Mapping the environment :

- For mapping an environment , there are several tools such as gmapping ,SLAM toolbox,etc…

- I have chosen the **CARTOGRAPHER** tool for mapping the world. Which provides real-time **Simultaneous localization and mapping (SLAM)** in 2D and 3D across.



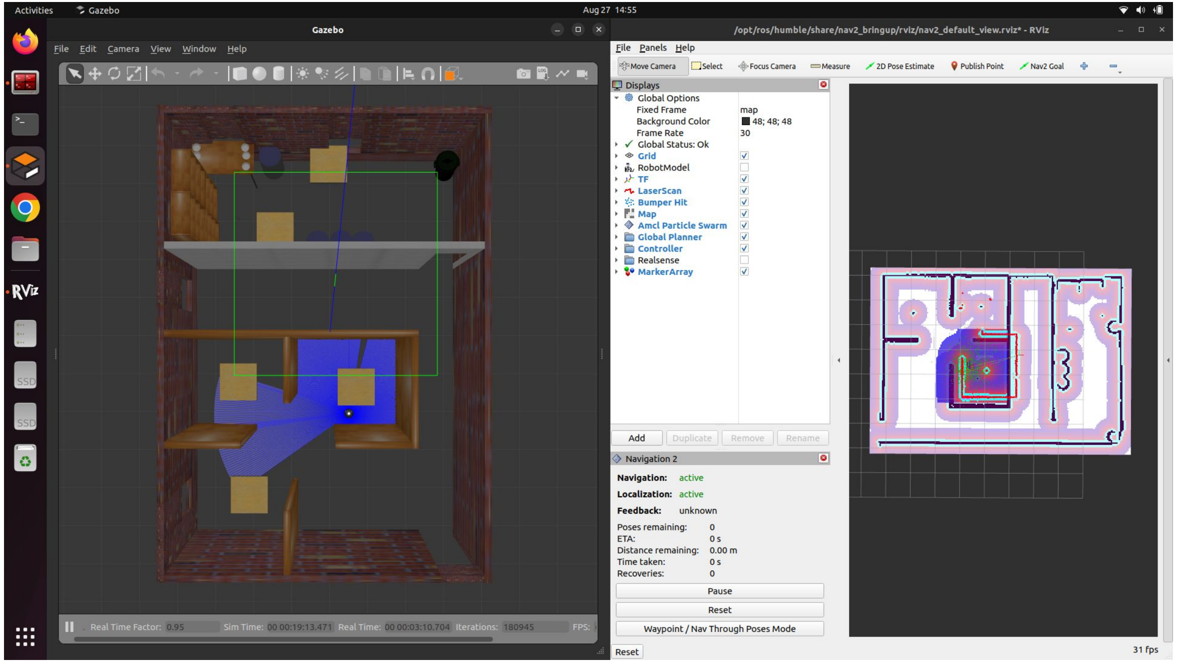
Real Time Factor: 0.98 Sim Time: 00 00:19:26.165 Real Time: 00 00:03:19.750 Iterations: 193639 FPS:

Reset **Left-Click:** Rotate. **Middle-Click:** Move X/Y. **Right-Click:** Zoom. **Shift:** More options.

Navigation of the robot in our environment :

- For navigating the loaded robot in our environment , we can use the navigation 2 package by loading our map into it.

- For this purpose , **Rviz (Ros/Robot Visualizer)** is used.



Real Time Factor: 0.95 Sim Time: 00 00:19:13.471 Real Time: 00 00:03:10.704 Iterations: 180945 FPS:

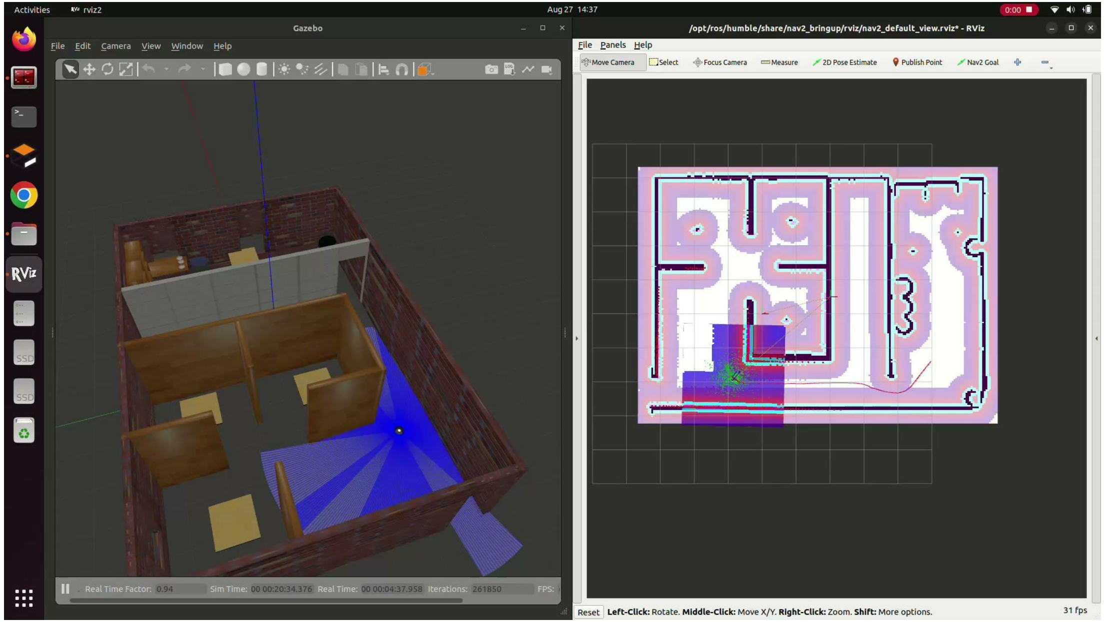
Global Options
Fixed Frame map
Background Color 48; 48; 48
Frame Rate 30
✓ Global Status: Ok
Grid ☑
RobotModel □
TF ☑
LaserScan ☑
Bumper Hit ☑
Map ☑
Amcl Particle Swarm ☑
Global Planner ☑
Controller ☑
Realsense □
MarkerArray ☑

Navigation: active
Localization: active
Feedback: unknown
Poses remaining: 0
ETA: 0 s
Distance remaining: 0.00 m
Time taken: 0 s
Recoveries: 0

Implementation as per the requirement :

- In Rviz , we can provide the destination point of the robot to navigate autonomously that is **Nav2 goal.**

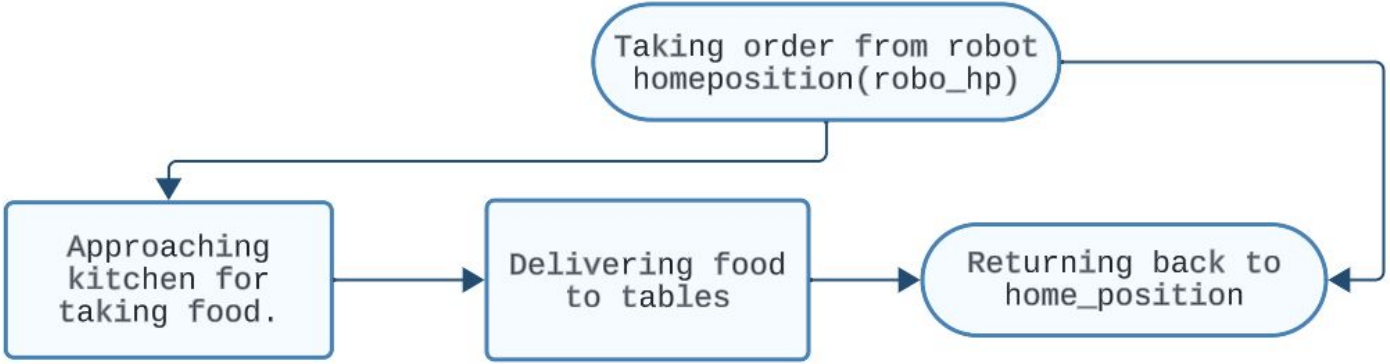
- Even for complete autonomous navigation of several destination points , we can use **waypoint/nav through poses mode.**


Activities
rviz2
Aug 27 14:37
Gazebo
File Edit Camera View Window Help
Real Time Factor: 0.94
Sim Time: 00 00:20:34.376
Real Time: 00 00:04:37.958
Iterations: 261850
FPS:
/opt/ros/humble/share/nav2_bringup/rviz/nav2_default_view.rviz* - RViz
File Panels Help
Move Camera
Select
Focus Camera
Measure
2D Pose Estimate
Publish Point
Nav2 Goal
Reset
Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click: Zoom. Shift: More options.
31 fps

Implementing logics for each criteria:

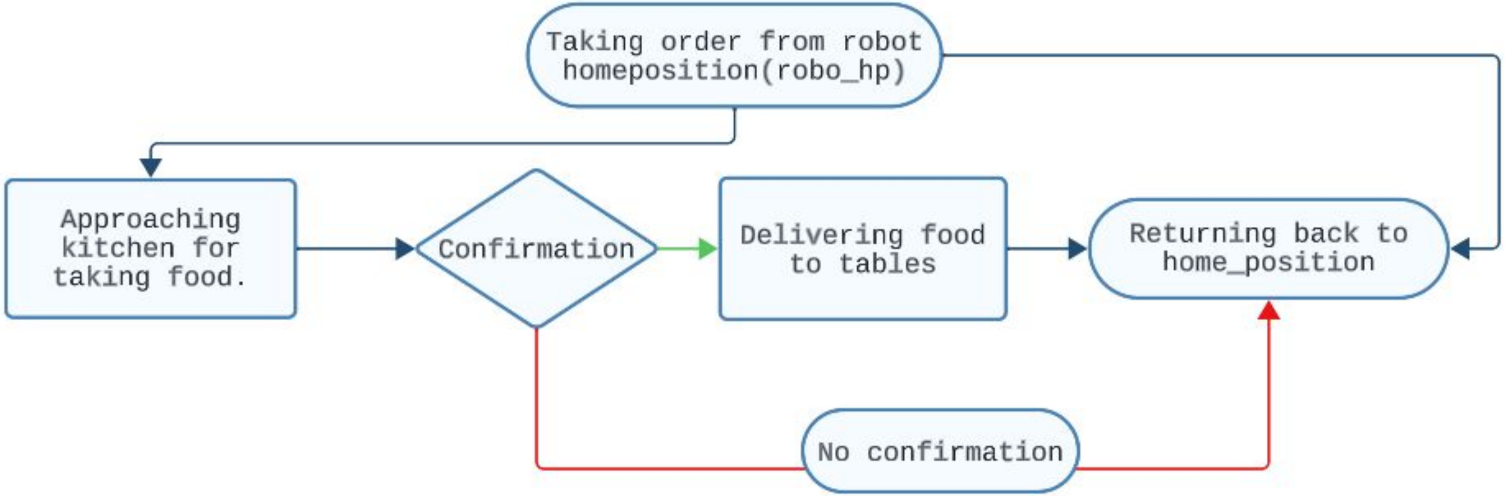
Criteria 1 (Single order)and 5(Multiple orders) :

- When orders received with table number,
- Robot_Homeposition → kitchen → Table for food delivery. (Requires no confirmation)


Taking order from robot homeposition(robo_hp)
Approaching kitchen for taking food.
Delivering food to tables
Returning back to home_position

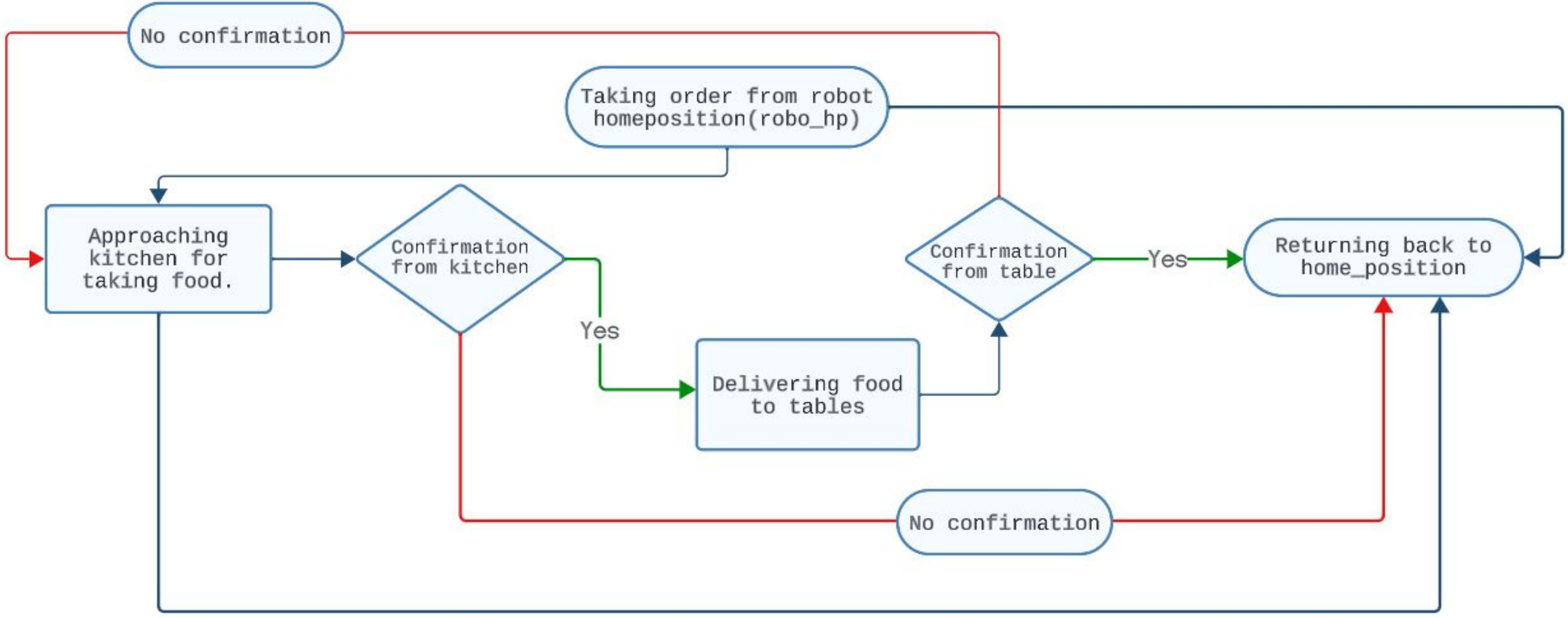
Criteria 2(Single order):

- When orders received with table number,
- Robot_Homeposition → kitchen → Table for food delivery. (Requires confirmation)


Taking order from robot homeposition(robo_hp)
Approaching kitchen for taking food.
Confirmation
Delivering food to tables
Returning back to home_position
No confirmation

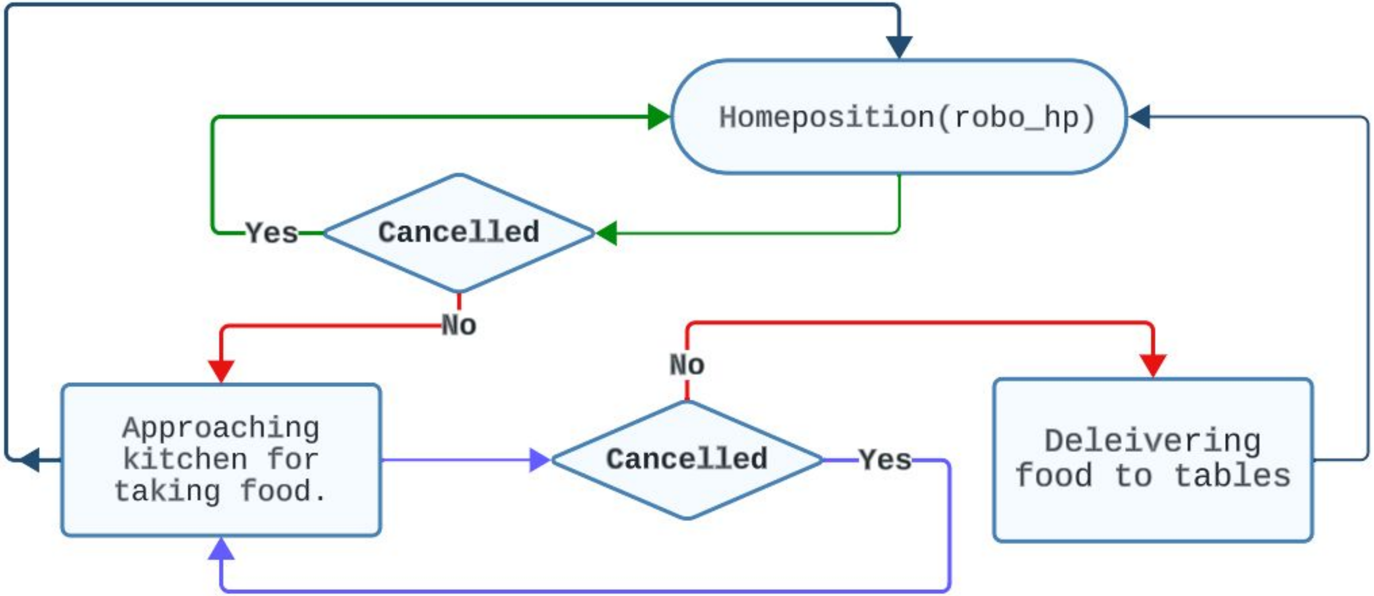
Criteria 3(Single order) and criteria 6(Multiple orders) :

- Robo_hp → kitchen (no confirmation) →Return back to robo_hp.
- Robo_hp → kitchen (confirmation) →Table (confirmed) → robo_hp.
- Robo_hp → kitchen (confirmation) →Table (no confirmed) → kitchen →robo_hp.


No confirmation
Taking order from robot homeposition(robo_hp)
Approaching kitchen for taking food.
Confirmation from kitchen
Yes
Delivering food to tables
Confirmation from table
Yes
Returning back to home_position
No confirmation

Criteria 4(Single order) and 7(multiple orders) :

- Order placed →Robo_hp → kitchen→**Cancelled** →return to robo_hp.
- Order placed →Robo_hp →kitchen →Table of order →**Cancelled** →kitchen →robo_hp.


Homeposition(robo_hp)
Yes
Cancelled
No
No
Approaching kitchen for taking food.
Cancelled
Yes
Deleivering food to tables

THANK YOU