## Stage 1: PoC (Proof of Concept)

## ****Overview****

### ****Objective****

Develop a ROS-based system that **tracks and visualizes** a robot’s trail on the map while simulating **dosage accumulation** (e.g., UV disinfection, radiation, or any exposure-based activity) as the robot moves.

### ****System Components****

**ROS Node**: map\_trail\_persistence

**Input**:

/move\_base/global\_costmap/costmap – the robot’s global map

TF – for robot's current position (map → base\_link)

**Output**:

/map\_modified – occupancy grid with trail overlay

### ****Process Loop Summary****

At a fixed rate (1 Hz):

**Read the robot’s position** using TF.

**Define a circular footprint** based on the robot’s radius.

For every cell within the footprint:

**Accumulate a dosage value** proportional to time.

**Assign a trail color** based on the dosage level

**Draw the updated trail** on a copy of the occupancy grid.

**Publish the new map** to /map\_modified.

### ****Key Concepts****

| **Concept** | **Purpose** |
| --- | --- |
| **Dosage Simulation** | Tracks how long the robot stays in an area and accumulates value |
| **Trail Coloring** | Encodes dosage levels visually using negative values on the map |
| **Persistence** | Keeps a memory of the robot’s past paths across map updates |
| **Robot Footprint** | Defines the circular region around the robot affecting the dosage map |

### ****Trail Color Codes****

| **Trail Segment** | **Condition (dosage ≥)** | **Grid Value** |
| --- | --- | --- |
| TRAIL\_COLOR1 | 0 | -2 |
| TRAIL\_COLOR2 | 25% of threshold | -50 |
| TRAIL\_COLOR3 | 50% of threshold | -100 |
| TRAIL\_COLOR4 | 75% of threshold | -150 |

### ⚙️ ****Adjustable Parameters****

disinfect\_type: sets dosage threshold (0 = 70, 1 = 80, 2 = 90, 3 = 100)

dosage\_value: how much dosage to accumulate per second

robot\_radius: size of the robot footprint on the map

### ****Stage 1 Deliverables****

Working ROS node: map\_trail\_persistence.py

Publishes /map\_modified for Rviz visualization

Logs dosage value and threshold on launch

Uses threading to separate TF processing and publishing

### ****Test it using:****

rosrun your\_package map\_trail\_persistence.py

Ensure /move\_base/global\_costmap/costmap and TF are active.

Let me know when you’re ready for:

**Stage 2**: Visualization in RViz or logging dose statistics

**Stage 3**: Saving coverage maps or creating a report UI

or a **launch file** for running everything cleanly.

Want this as a PDF or Markdown for documentation?