

REQUIREMENT FILE

◦ Requirements

Course:

CSA2001 - Fundamentals of AI and ML

Project-Based Learning – Autonomous Delivery Agent

Project Title:

Design and Implementation of an Autonomous Delivery Agent in a 2D Grid City

Objective:

To design an intelligent autonomous agent capable of navigating a **2D grid-based city** to deliver packages efficiently. The agent must plan optimal routes considering **dynamic obstacles, terrain costs, and time/fuel constraints** using various AI search algorithms.

Functional Requirements:

1. **Environment Model:** ◦ Represent static obstacles (walls, buildings, etc.). ◦ Incorporate varying terrain movement costs (e.g., road, grass, water).
 - Include dynamic obstacles like moving vehicles.
2. **Rational Agent:**

REQUIREMENT FILE

- The agent must make rational decisions to minimize delivery time and fuel usage.
 - 3. Search Algorithms Implementation:**
 - **Uninformed Search:** BFS or Uniform-cost Search
 - **Informed Search:** A* (with admissible heuristic)
 - **Local Search:** Hill-climbing with random restarts or Simulated Annealing
 - 4. Dynamic Replanning:**
 - Replan when new obstacles appear or when paths are blocked dynamically.
 - 5. Algorithm Comparison:**
 - Evaluate and compare each algorithm's performance on different maps.
 - Metrics: path cost, number of nodes expanded, time taken.
 - 6. Analysis Report:**
 - Provide detailed analysis describing when and why certain algorithms perform better.
-

Required Deliverables:

- 1. Source Code (Well-documented):**
 - Preferably in Python.
 - Include CLI to run each planner.
 - Must include:
 - Logging of dynamic replanning
 - Modular structure with comments and docstrings

REQUIREMENT FILE

2. Test Maps (Minimum 4):

- Small ◦ Medium ◦ Large ◦ One with dynamic moving obstacles ◦ Maps must be in .txt or grid file format.

3. Short Report (Max 6 pages):

- Environment model
- Agent and algorithm design
- Heuristics used
- Experimental results with tables and plots
- Analysis and conclusion

Technical Constraints & Assumptions:

- Grid cells have **integer movement costs** ≥ 1 • Moving obstacles:
 - Move **deterministically** based on a known schedule (for A*, BFS) ◦ Or move **unpredictably** (for local search testing)
- Agent can move in **4 directions**: up, down, left, right (diagonal optional – mention in report)
- Code must be **testable and reproducible** (include dependencies and instructions)

Tools & Libraries:

- Programming Language: **Python 3.8+** • Libraries:
 - numpy, matplotlib, pandas, pygame (for visualization), tqdm
- CLI Interface: argparse

REQUIREMENT FILE

- Version Control: Git (with README and setup instructions)