جامعة بيرزيت
BIRZEIT UNIVERSITY
Faculty of Engineering and Technology كلية الهندسة والتكنولوجيا
Electrical and Computer Engineering Department دائرة الهندسة الكهربائية وهندسة الحاسوب

ARTIFICIAL INTELLIGENCE

ENCS3340

Project#1:

Optimization Strategies for Local Package Delivery Operations

Prepared by:

Jenan Mohammad - 1210345

Taleen Bayatneh-1211305

Instructor: Dr. Samah Alaydi

Section: 3

May 2025

Introduction

This project simulates and optimizes the operations of a local package delivery shop. Each package is characterized by x,y coordinates, weight, and priority. The goal is to assign packages to vehicles such that the total traveled distance is minimized. The project implements two optimization techniques: Simulated Annealing (SA) and Genetic Algorithm (GA), each capable of handling user-defined inputs.

Problem Formulation

Delivering packages efficiently using a fleet of delivery vehicles. Each package has a specific location, a weight, and a priority level (with 1 being the highest priority). Each vehicle has a limited weight capacitym while the shop location is (0,0). The goal is to assign packages to vehicles and determine the route each vehicle should take while:

- Keeping the total distance traveled by all vehicles as short as possible.
- Making sure no vehicle exceeds its weight limit.
- Giving preference to high-priority packages, especially when resources are limited.

Inputs:

1. A list of packages, each with: (x, y) coordinates (destination), weight (kg), and a priority.
2. A list of vehicles, each with a weight capacity (kg).

Output:

Which packages are delivered by which vehicles.

The optimal route each vehicle should follow (always starting and ending at the shop located at point (0,0)).

Optimization Goal:

Reduce the total distance all vehicles travel using the Euclidean formula to measure straight-line distance between points.

$$\sum_{vehicles} = distance from shop to each package and back$$

Constraints:

- No vehicle should carry more than it can handle.
- Try to include higher-priority packages first — unless doing so would result in significantly longer routes or constraint violations.

Heuristics Used

GA: simulates natural selection. A population of solutions is generated and evolved through selection, crossover, and mutation. The fittest individuals survive and combine, aiming to find an optimal solution with minimal route distance.

-> **Smart Starting Point:** The initial population isn't completely random — we generate it using a greedy approach that assigns packages to vehicles based on priority and available capacity.

-> **Fitness Evaluation:** The quality of each solution (or "individual") is measured by the total distance all vehicles travel to deliver their assigned packages. The less the distance, the better the solution.

-> **Selection Process:** We use a roulette wheel selection method — this means better solutions (those with shorter distances) are more likely to be chosen as parents, but weaker solutions still have a chance, keeping diversity in the population.

-> **Crossover Technique:** During crossover, we swap entire routes between two vehicles from different parent solutions. This helps to mix delivery assignments and explore new combinations.

-> **Mutation Operation:** To introduce variation, we occasionally reassign a random package to a different vehicle (if capacity allows), shaking up the solution to escape local optima.

SA: a probabilistic optimization method. The solution starts with a random assignment of packages to vehicles and gradually evolves by exploring neighbor solutions. Acceptance of worse solutions is controlled by a temperature schedule that decreases over time, enabling the algorithm to escape local optima.

-> **Initial Solution:** We don't start randomly — we begin from a greedy solution, where high-priority packages are placed first without exceeding vehicle limits.

-> **Generating Neighbors:** We explore nearby solutions by:

-> Swapping packages between two vehicles.

-> Changing the delivery order within a single vehicle's route.

-> **Controlled Acceptance of Worse Solutions:** If a new solution is worse (longer route), we might still accept it ,but the chance depends on the current temperature (higher temperature = more flexibility). As the algorithm cools down, it becomes more selective.

Constraint Handling

The system ensures that no vehicle is assigned a total package weight exceeding its capacity. If a package cannot be assigned due to weight, a warning is printed. Vehicle routes are limited to feasible assignments only.

Parameter Tuning

The parameters used in both algorithms were selected based on testing and known best practices. In the Genetic Algorithm, a population size of 20, 50 generations, and a mutation rate of 10% were used, providing a balance between performance and speed. For Simulated Annealing, we used an initial temperature of 1000 with a cooling rate of 0.95 and 100 iterations per temperature. These settings align closely with our tuning insights and produced reliable results across different scenarios.

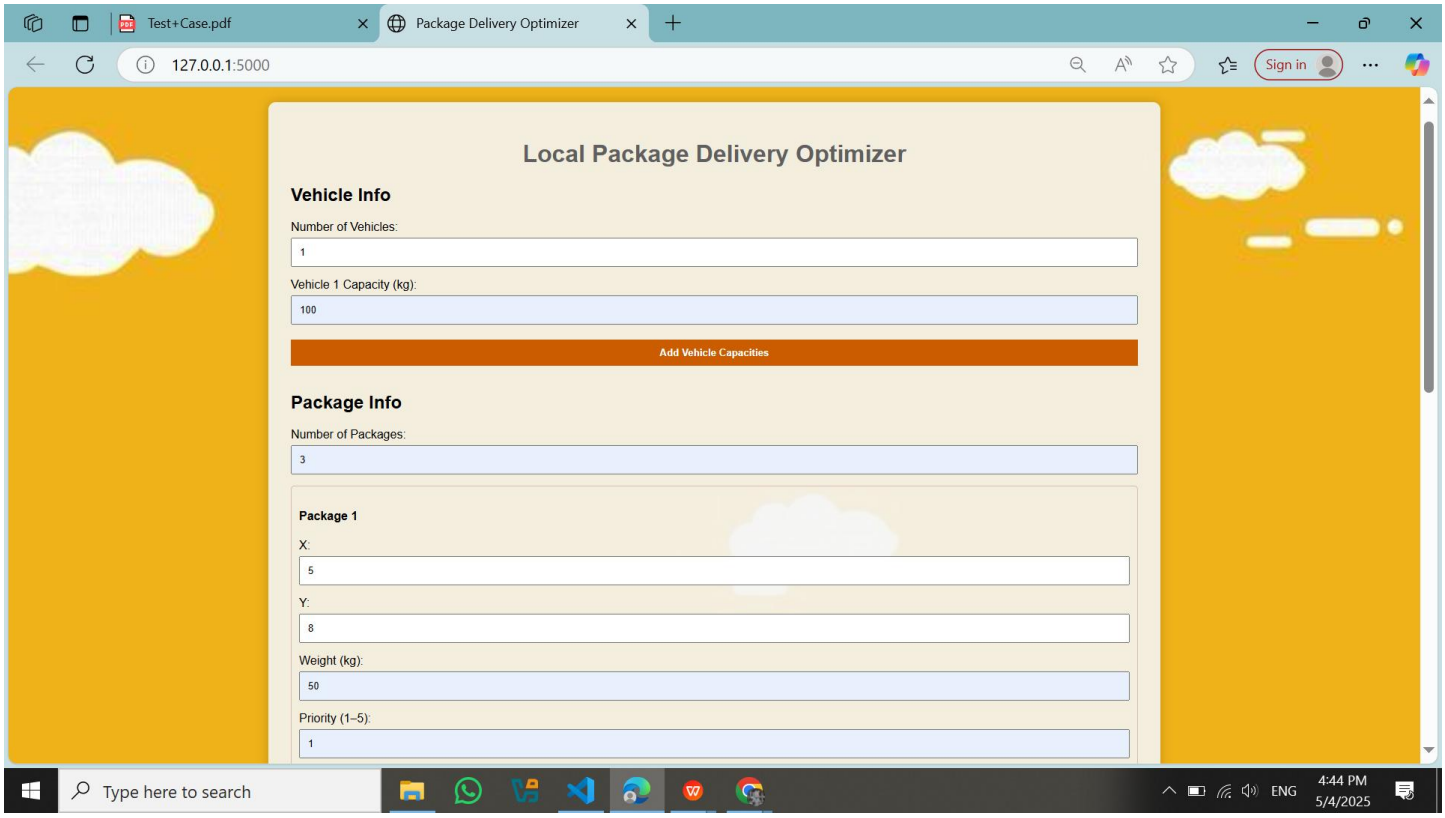
Test Cases

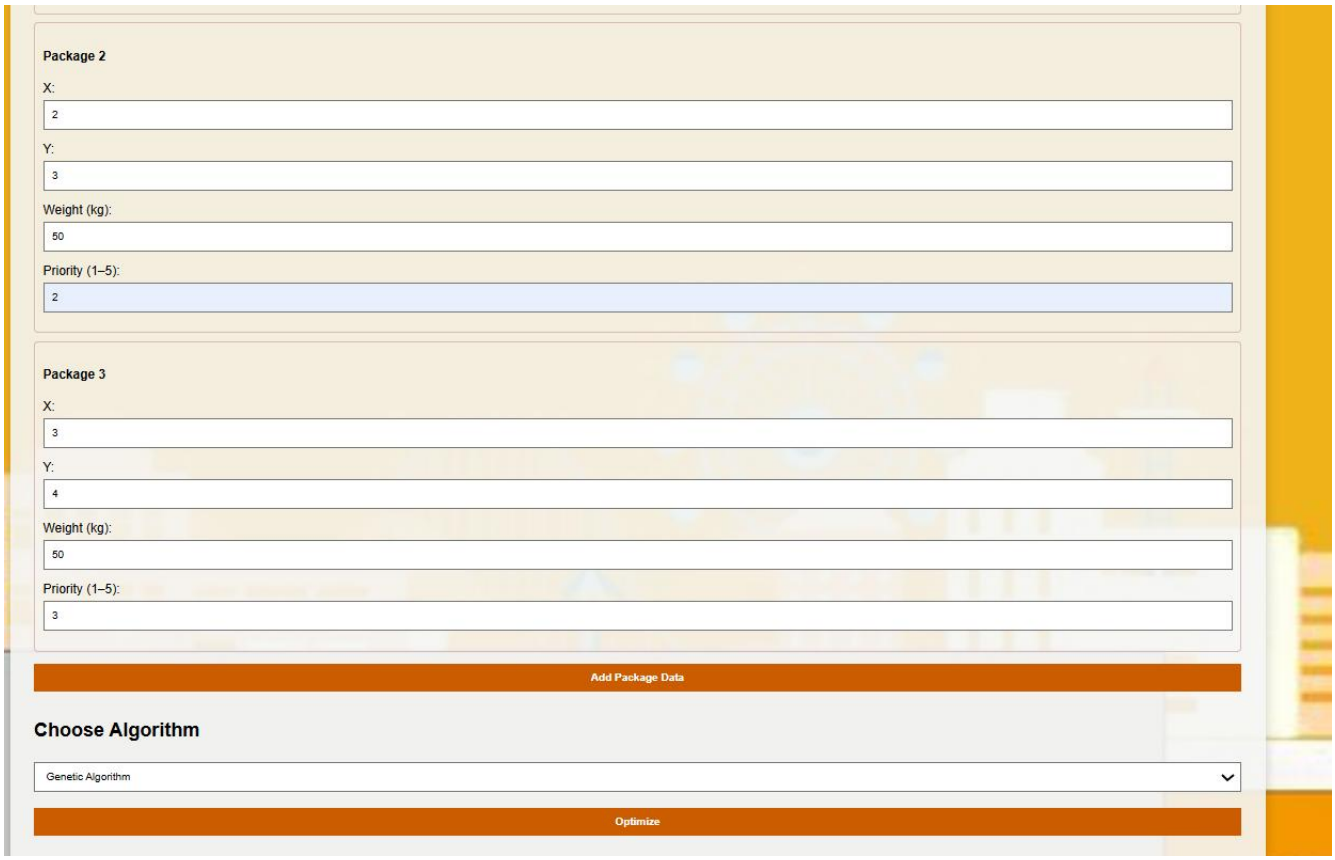
Priority Handling Test

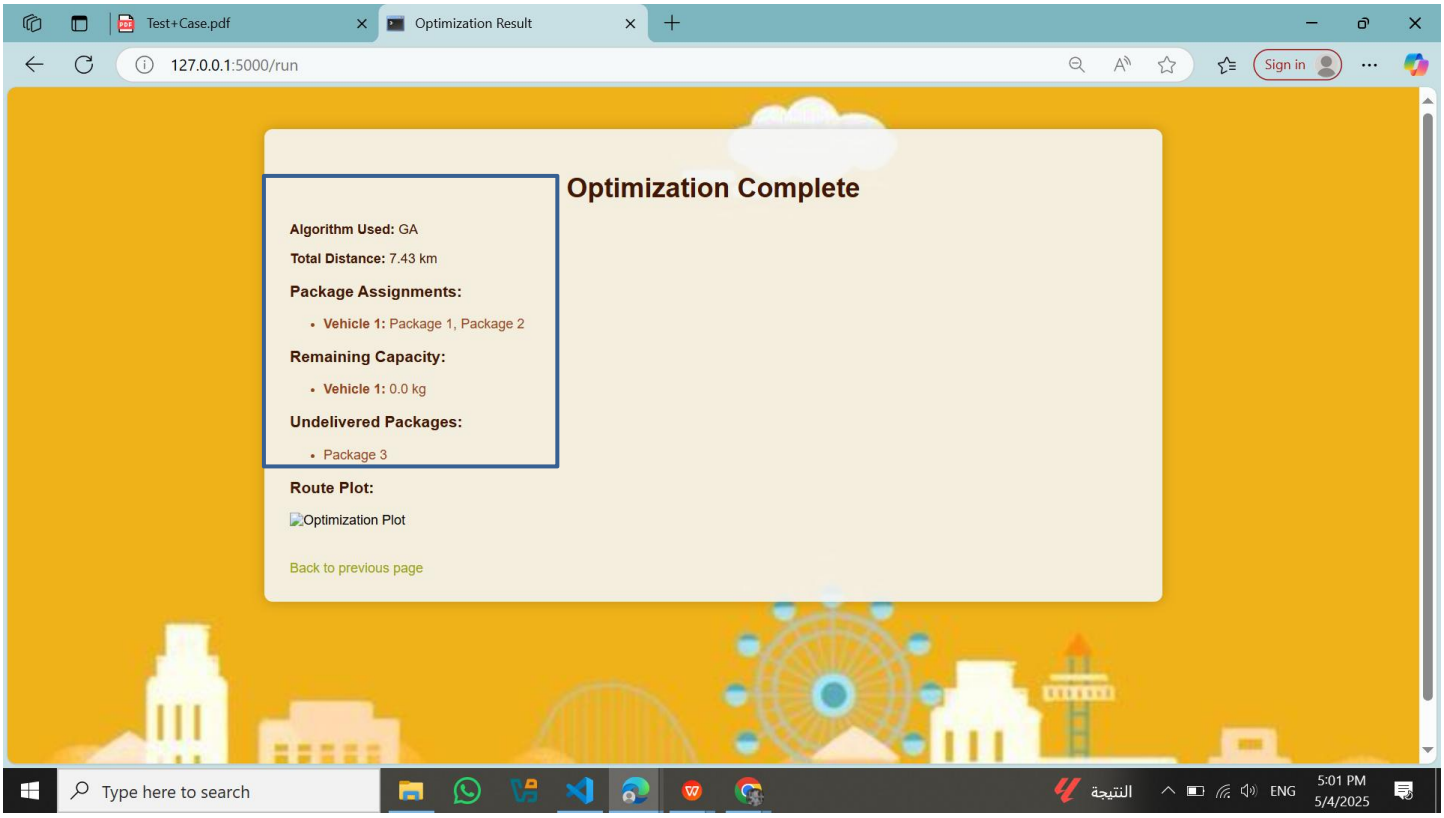
• Input: o Vehicles: 1 vehicle with a capacity of 100 kg.

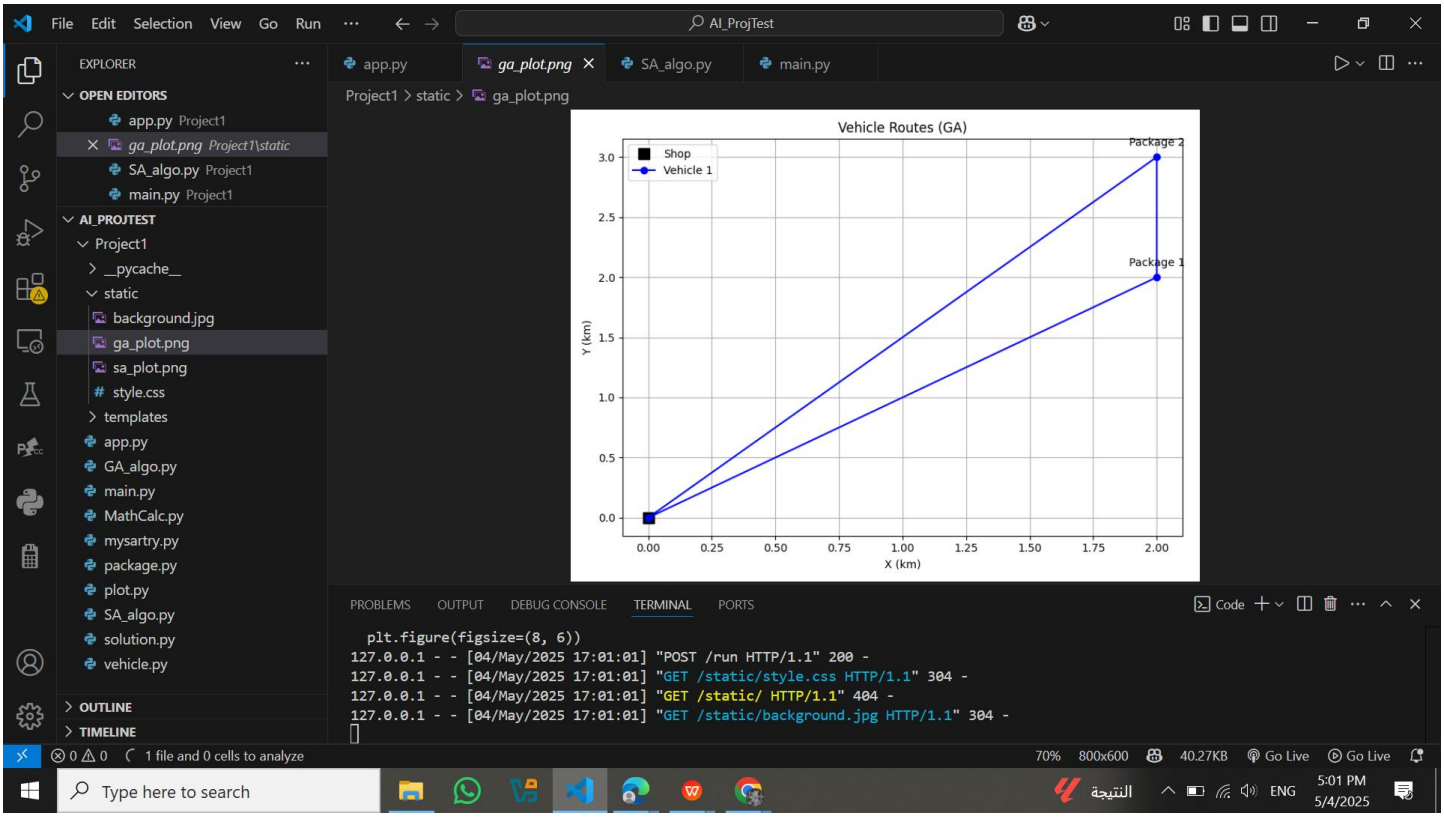
o Packages: ▪ Package 1,2,3= 50 kg(weight) Priority: 1, 2, 3 respectively

• Expected Outcome: o Packages 1 and 2 are selected for delivery due to higher priority. o Package 3 is deferred or unassigned due to capacity constraints.




Package 2
X:
2
Y:
3
Weight (kg):
50
Priority (1–5):
2
Package 3
X:
3
Y:
4
Weight (kg):
50
Priority (1–5):
3
Add Package Data
Choose Algorithm
Genetic Algorithm
Optimize


Test+Case.pdf
Optimization Result
127.0.0.1:5000/run
Sign in
Optimization Complete
Algorithm Used: GA
Total Distance: 7.43 km
Package Assignments:
Vehicle 1: Package 1, Package 2
Remaining Capacity:
Vehicle 1: 0.0 kg
Undelivered Packages:
Package 3
Route Plot:
Optimization Plot
Back to previous page
Type here to search
النتيجة
ENG
5:01 PM
5/4/2025



Discussion:

In this case, 1 delivery vehicle with a 100 kg and 3 packages, each weight :50 kg. Obviously, the vehicle can only carry two at most , so the system had to make a choice. Using the Genetic Algorithm, the optimizer selected package 1 and package 2 for delivery, fully maxing out the vehicle's capacity. It left out package 3, which makes sense since adding it would exceed the weight limit. Interestingly, Package 1 had the highest priority, so it's reassuring to see the algorithm took that into account when making the decision.

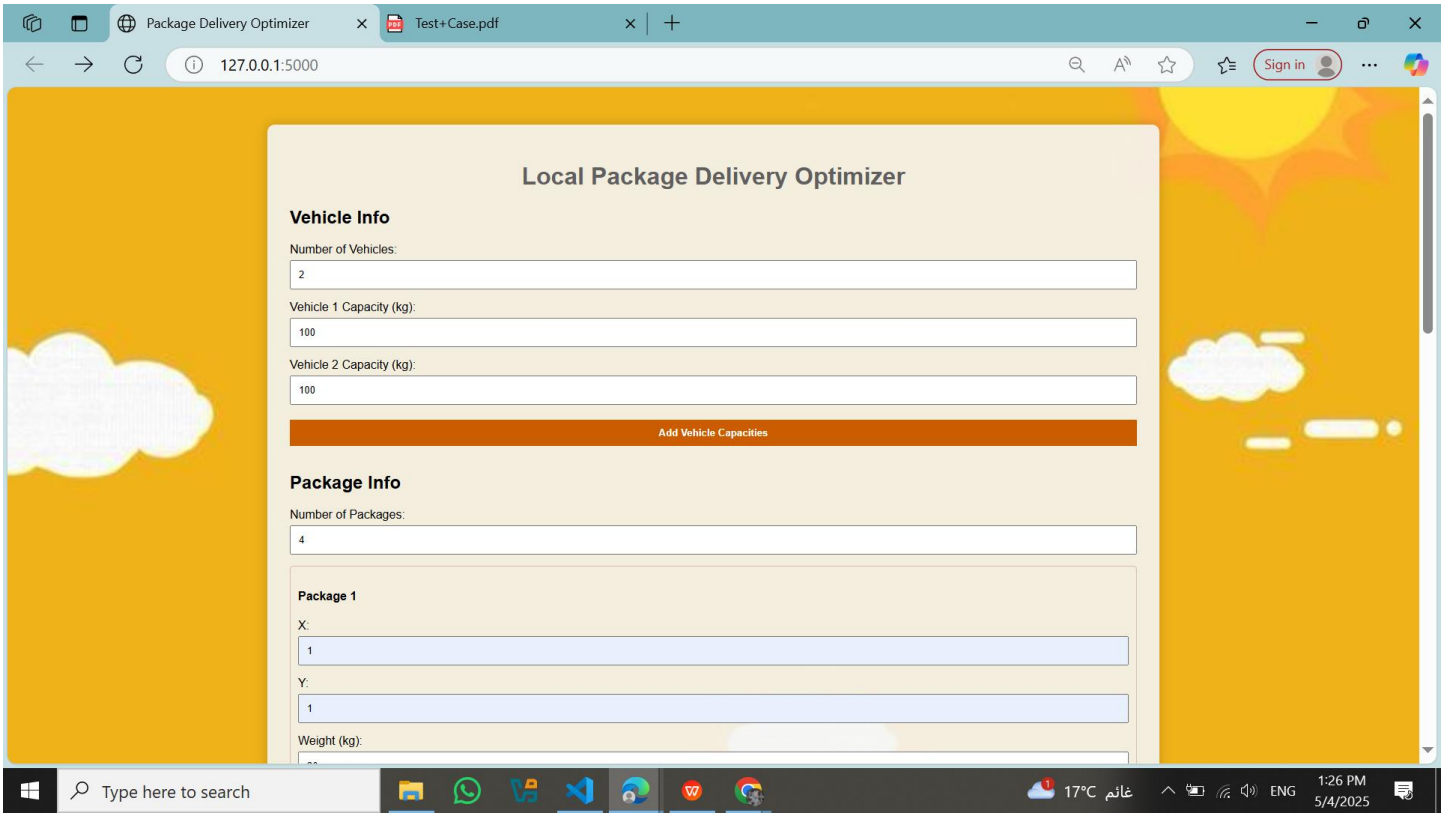
Basic Feasibility Test

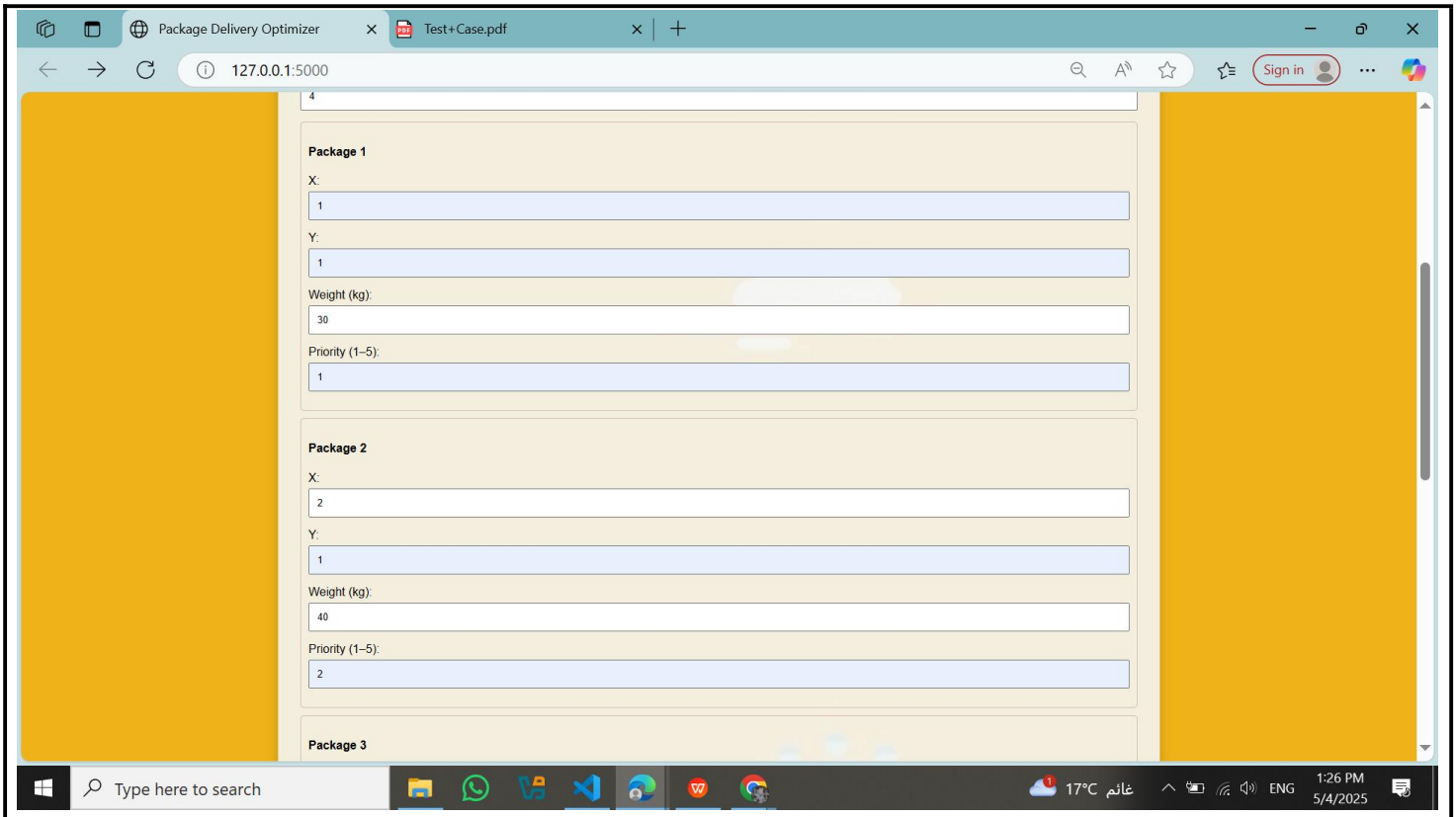
• Input: o Vehicles: 2 vehicles, each with a capacity of 100 kg.

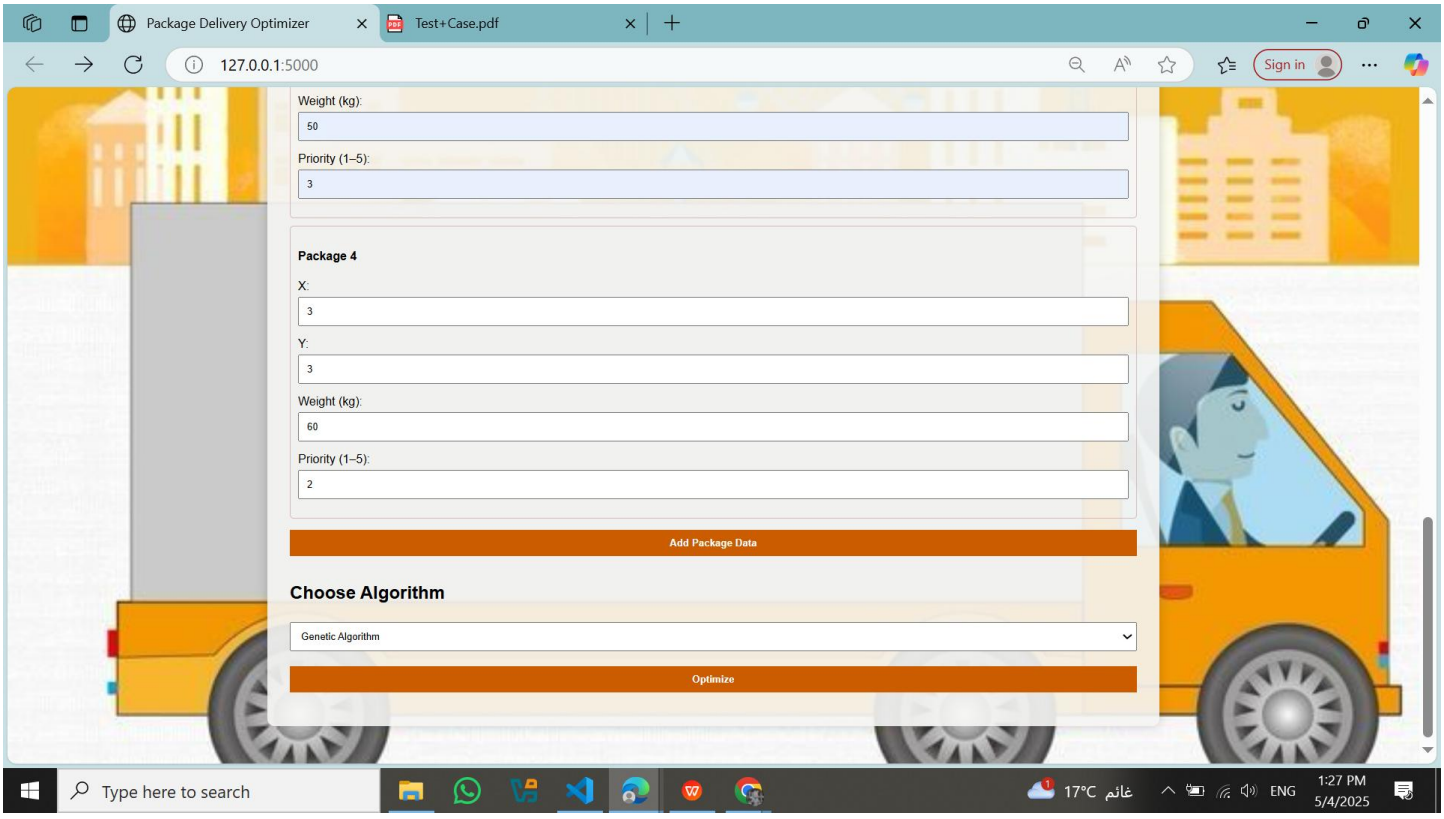
o Packages: 4 packages with the following weights: 30 kg, 40 kg, 50 kg, and 60 kg.

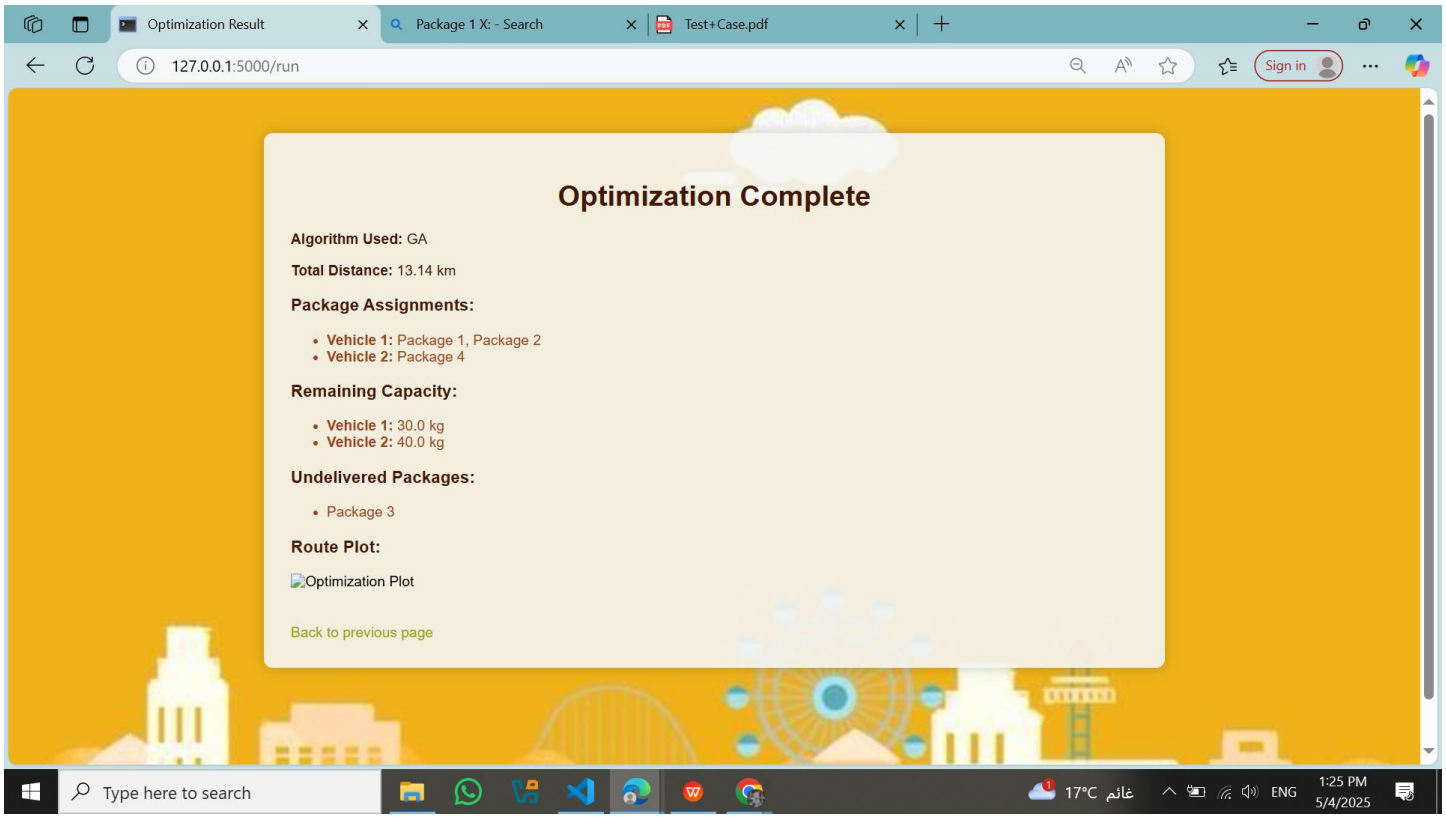
• Expected Outcome: Packages are distributed among vehicles such that no vehicle carries more than 100 kg.

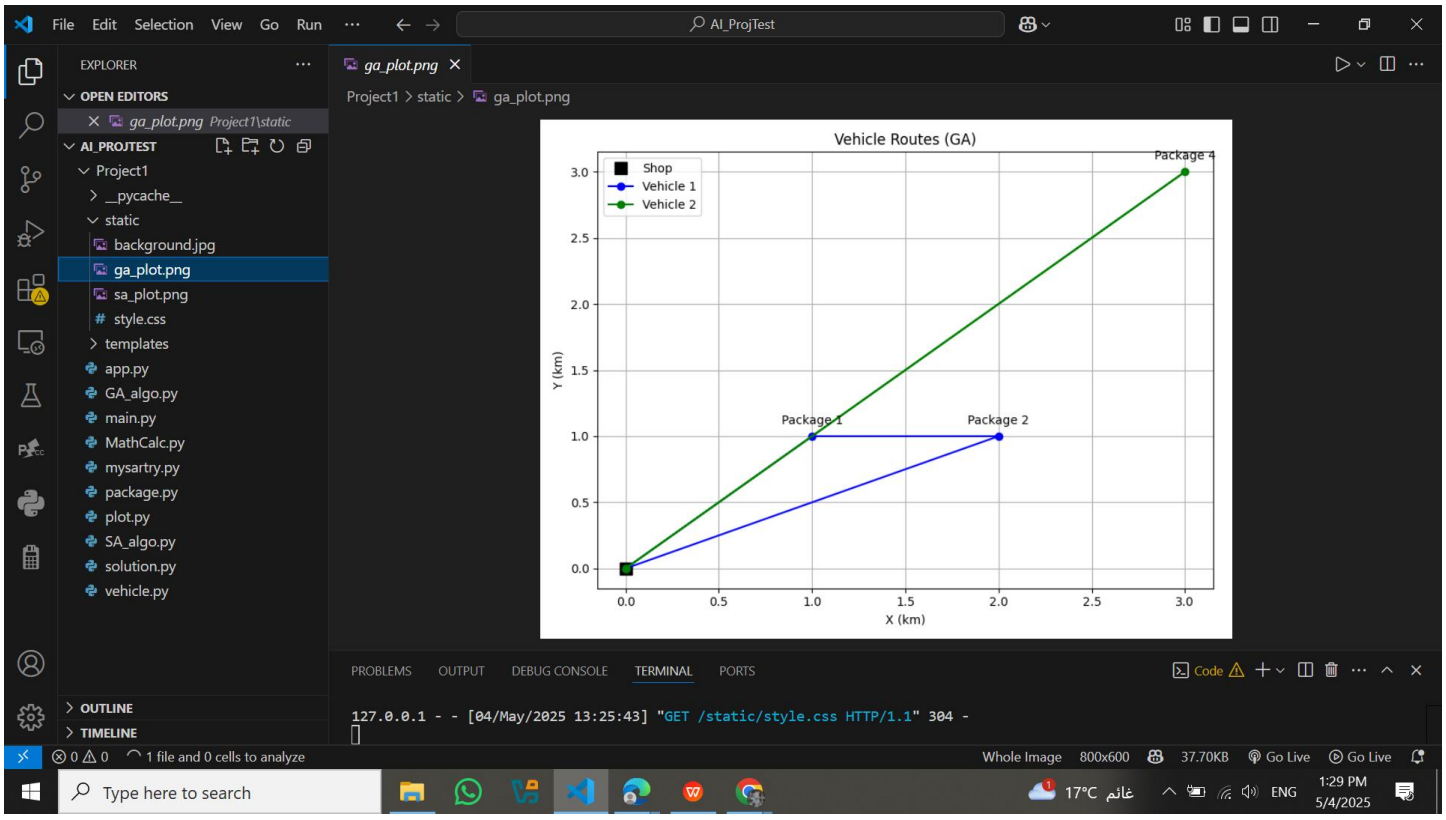
o All packages are assigned to vehicles.




Package Delivery Optimizer
Test+Case.pdf
127.0.0.1:5000
Sign in
4
Package 1
X:
1
Y:
1
Weight (kg):
30
Priority (1–5):
1
Package 2
X:
2
Y:
1
Weight (kg):
40
Priority (1–5):
2
Package 3
Type here to search
17°C غائم
ENG
1:26 PM
5/4/2025


Package Delivery Optimizer
Test+Case.pdf
127.0.0.1:5000
Sign in
Weight (kg):
50
Priority (1–5):
3
Package 4
X:
3
Y:
3
Weight (kg):
60
Priority (1–5):
2
Add Package Data
Choose Algorithm
Genetic Algorithm
Optimize
Type here to search
17°C غائم
ENG
1:27 PM
5/4/2025


Optimization Result
Package 1 X: - Search
Test+Case.pdf
127.0.0.1:5000/run
Sign in
Optimization Complete
Algorithm Used: GA
Total Distance: 13.14 km
Package Assignments:
Vehicle 1: Package 1, Package 2
Vehicle 2: Package 4
Remaining Capacity:
Vehicle 1: 30.0 kg
Vehicle 2: 40.0 kg
Undelivered Packages:
Package 3
Route Plot:
Optimization Plot
Back to previous page
Type here to search
17°C غائم
ENG
1:25 PM
5/4/2025


File Edit Selection View Go Run
AI_ProjTest
EXPLORER
OPEN EDITORS
ga_plot.png Project1\static
AI_PROJTEST
Project1
__pycache__
static
background.jpg
ga_plot.png
sa_plot.png
style.css
templates
app.py
GA_algo.py
main.py
MathCalc.py
mysartry.py
package.py
plot.py
SA_algo.py
solution.py
vehicle.py
ga_plot.png
Project1 > static > ga_plot.png
Vehicle Routes (GA)
Shop
Vehicle 1
Vehicle 2
Package 1
Package 2
Package 4
Y (km)
X (km)
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
127.0.0.1 - - [04/May/2025 13:25:43] "GET /static/style.css HTTP/1.1" 304 -
OUTLINE
TIMELINE
1 file and 0 cells to analyze
Whole Image 800x600 37.70KB
Type here to search
17°C غائم
ENG
1:29 PM
5/4/2025

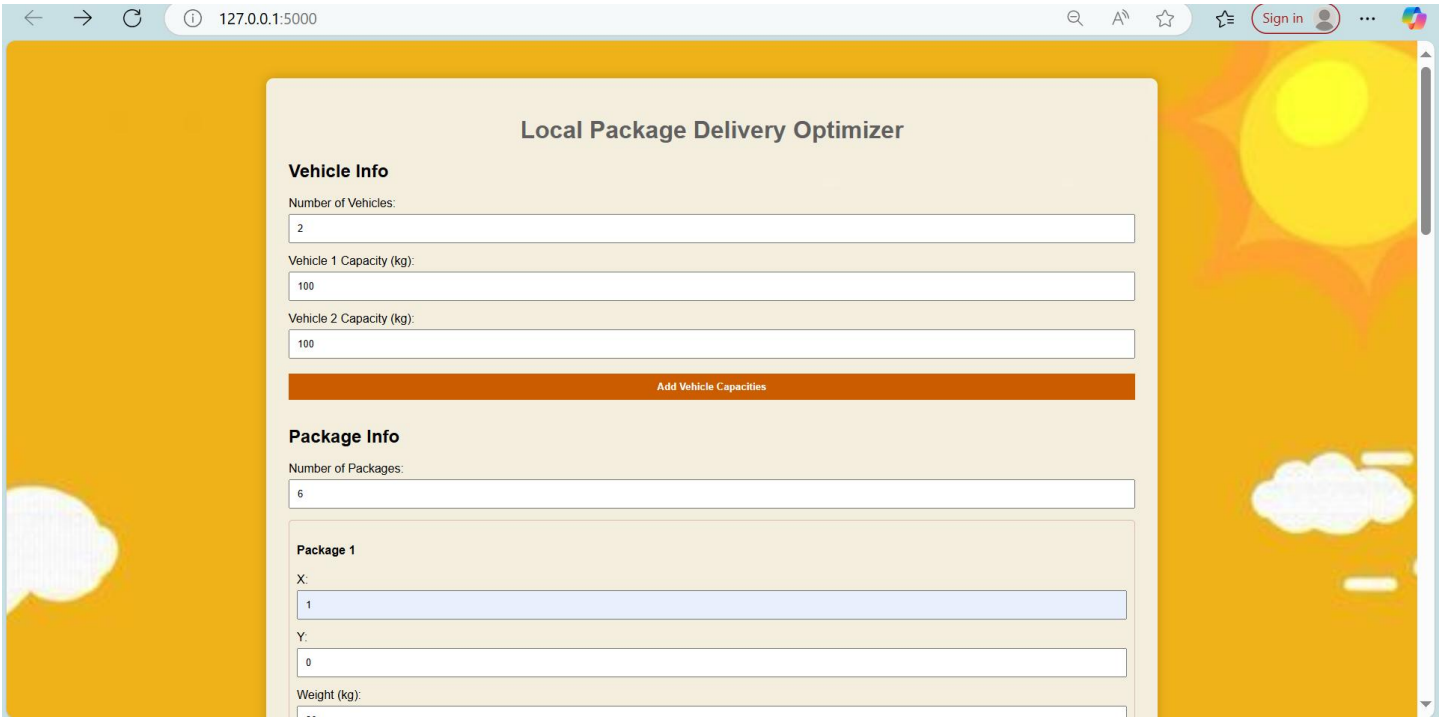
Distance Optimization Test

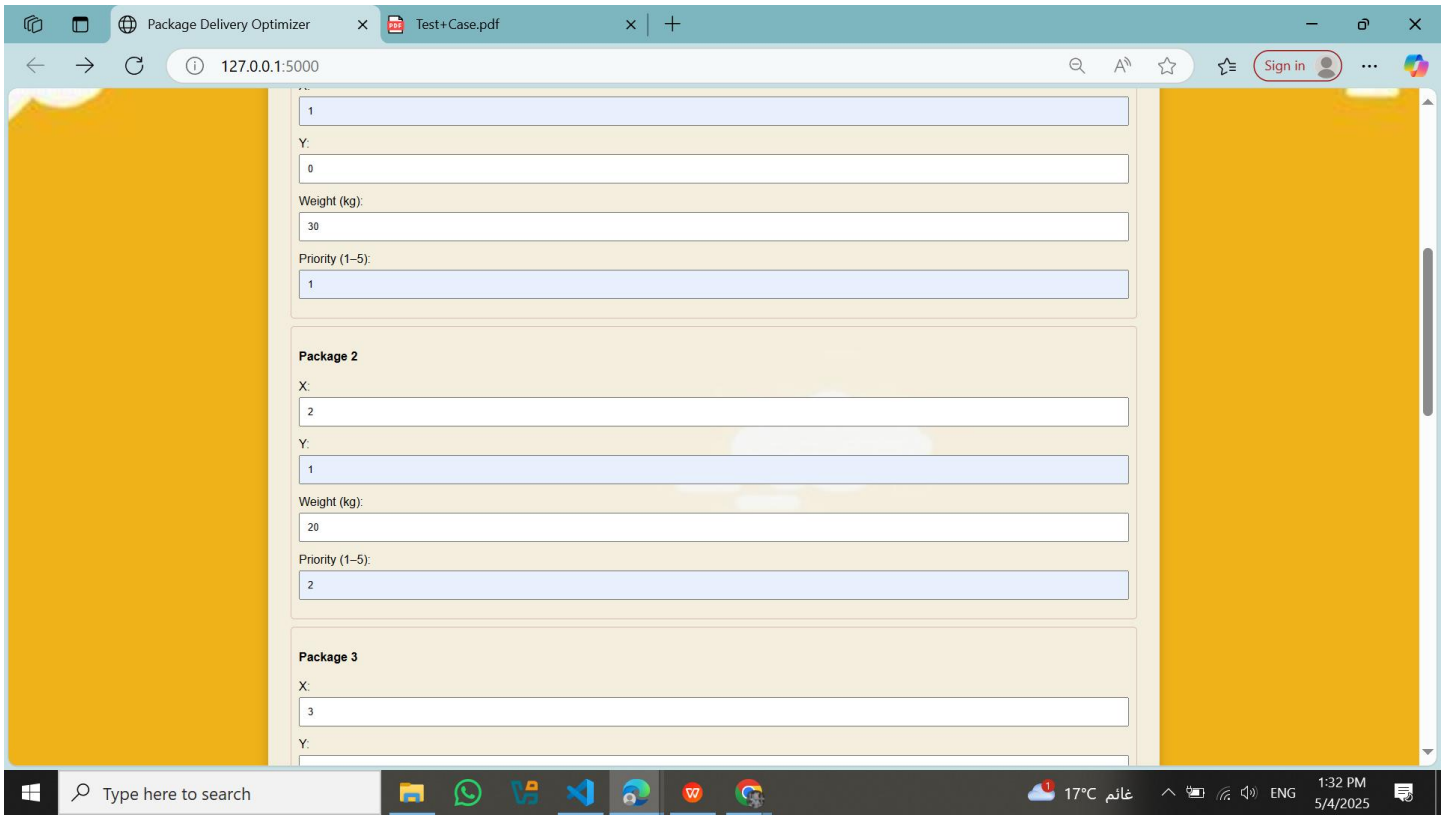
• Objective: Ensure that the system minimizes the total distance travelled by all vehicles.

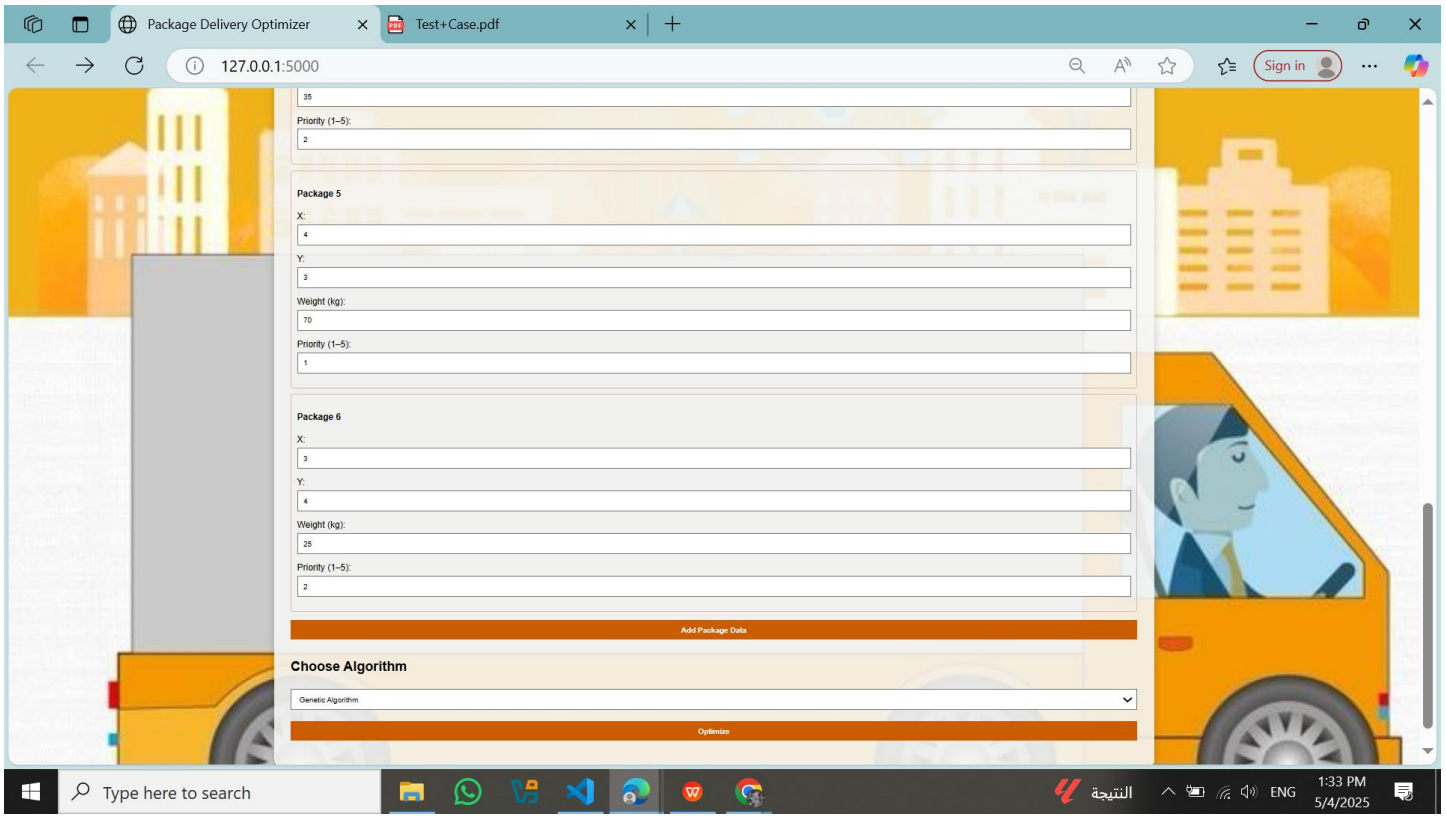
• Input: o Vehicles: 2 vehicles, each with a capacity of 100 kg.

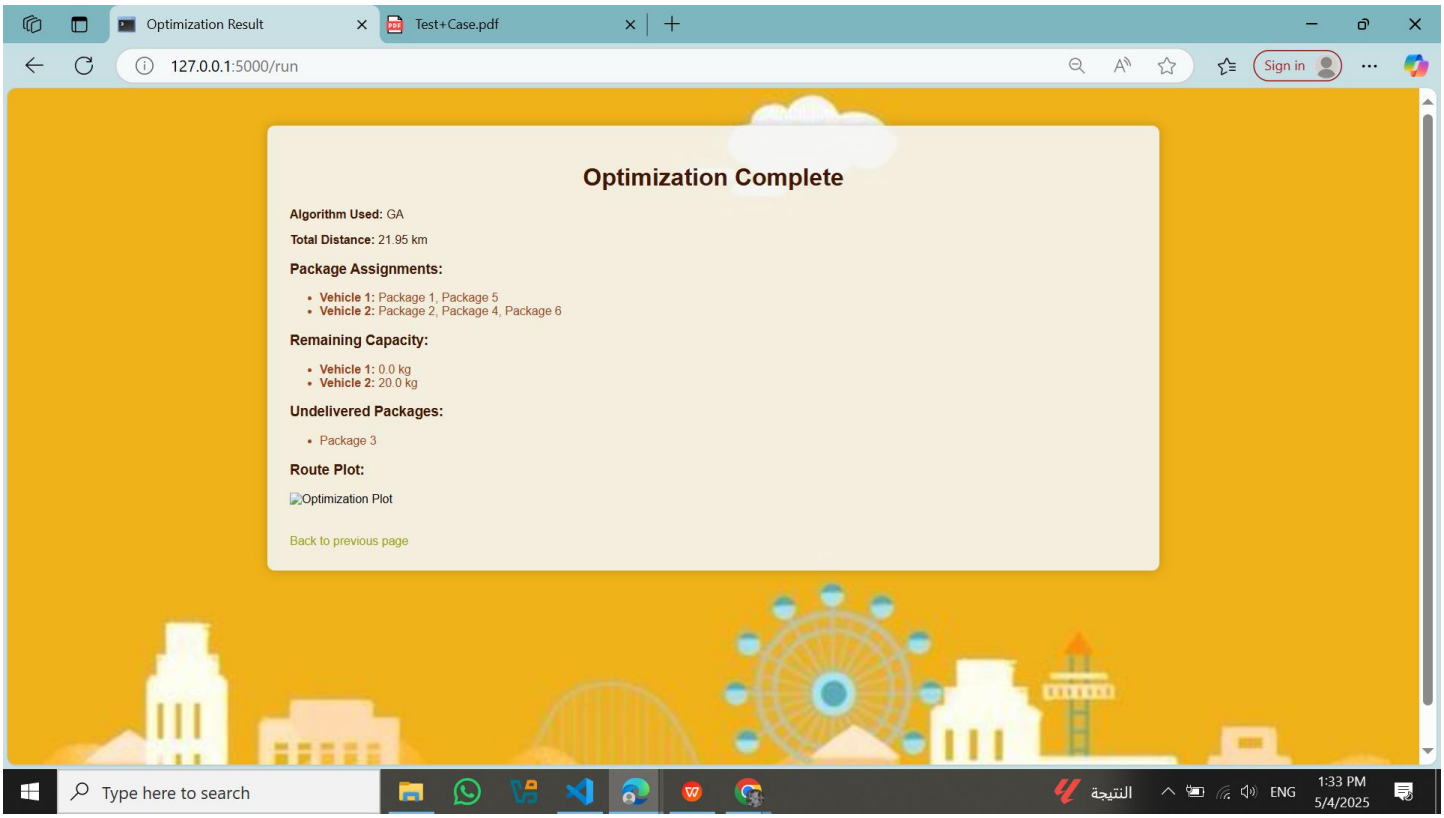
o Packages: 6 packages located at varying distances from the depot.

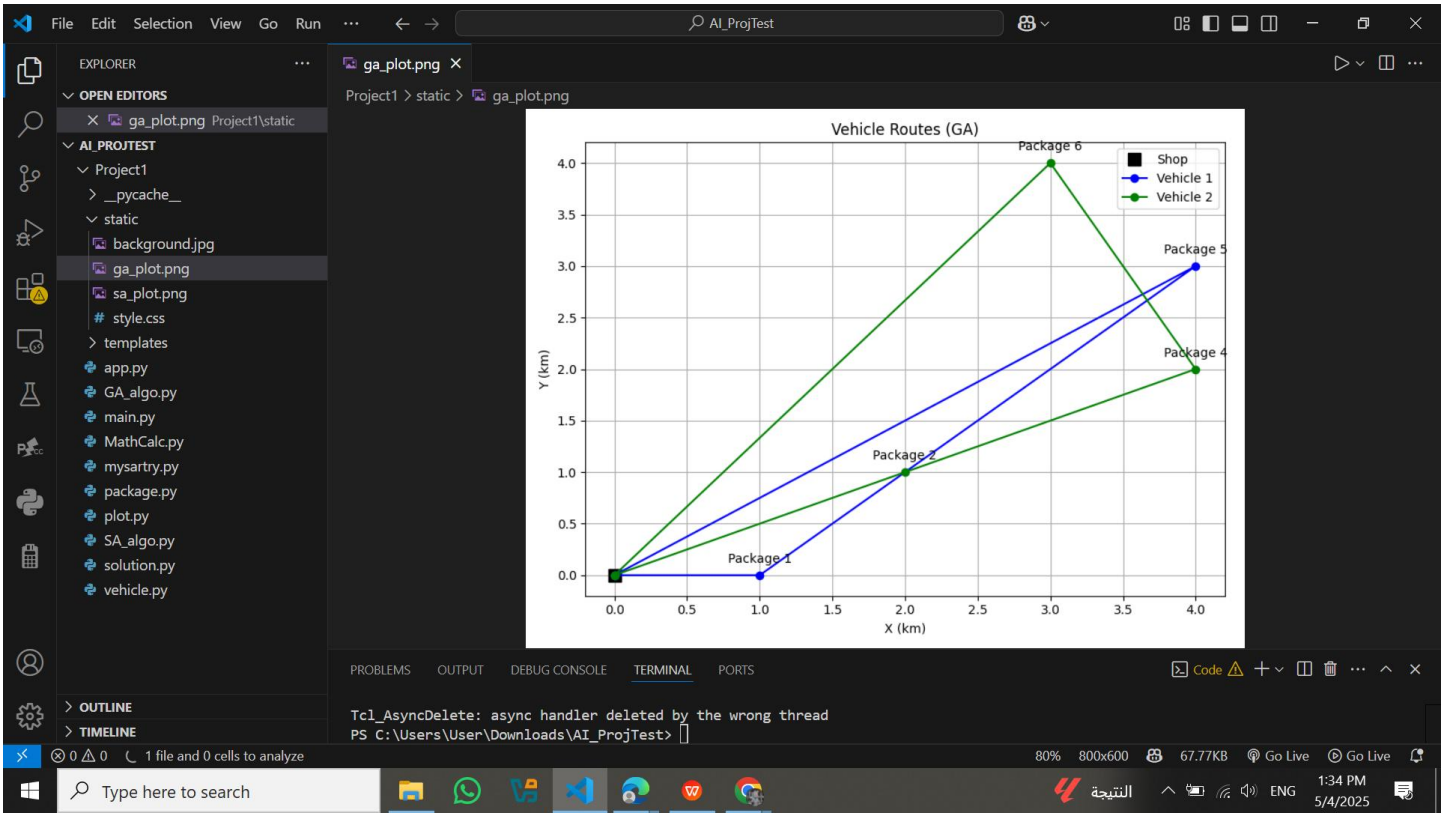
• Expected Outcome: Packages are assigned and routed to minimize the combined distance travelled by both vehicles.






Package Delivery Optimizer
Test+Case.pdf
127.0.0.1:5000
Sign in
35
Priority (1–5):
2
Package 5
X:
4
Y:
3
Weight (kg):
70
Priority (1–5):
1
Package 6
X:
3
Y:
4
Weight (kg):
25
Priority (1–5):
2
Add Package Data
Choose Algorithm
Genetic Algorithm
Optimize
Type here to search
النتيجة
ENG
1:33 PM
5/4/2025


Optimization Result
Test+Case.pdf
127.0.0.1:5000/run
Sign in
Optimization Complete
Algorithm Used: GA
Total Distance: 21.95 km
Package Assignments:
Vehicle 1: Package 1, Package 5
Vehicle 2: Package 2, Package 4, Package 6
Remaining Capacity:
Vehicle 1: 0.0 kg
Vehicle 2: 20.0 kg
Undelivered Packages:
Package 3
Route Plot:
Optimization Plot
Back to previous page
Type here to search
النتيجة
ENG
1:33 PM
5/4/2025



Discussion about this case :

In this test, the Genetic Algorithm handled 6 packages using 2 delivery vehicles. It packed 5 of them based on the weight and priority, fully loading one vehicle and nearly filling the other. Only one package(P#3) couldn't be delivered due to limited space. The total route covered was around 21.95 km, showing the algorithm's ability to make efficient delivery decisions while respecting capacity limits.

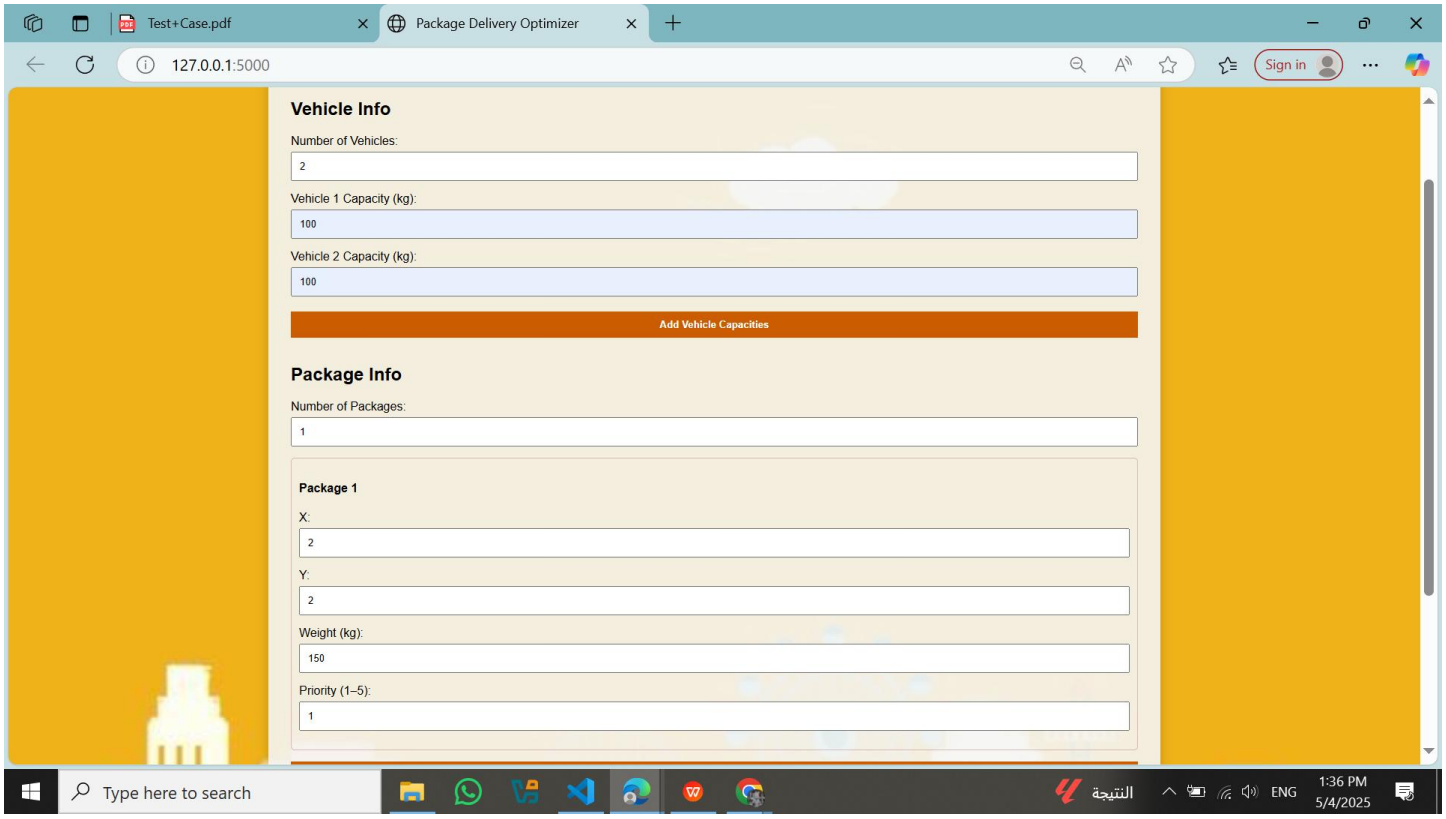
Edge Case - Overcapacity Package

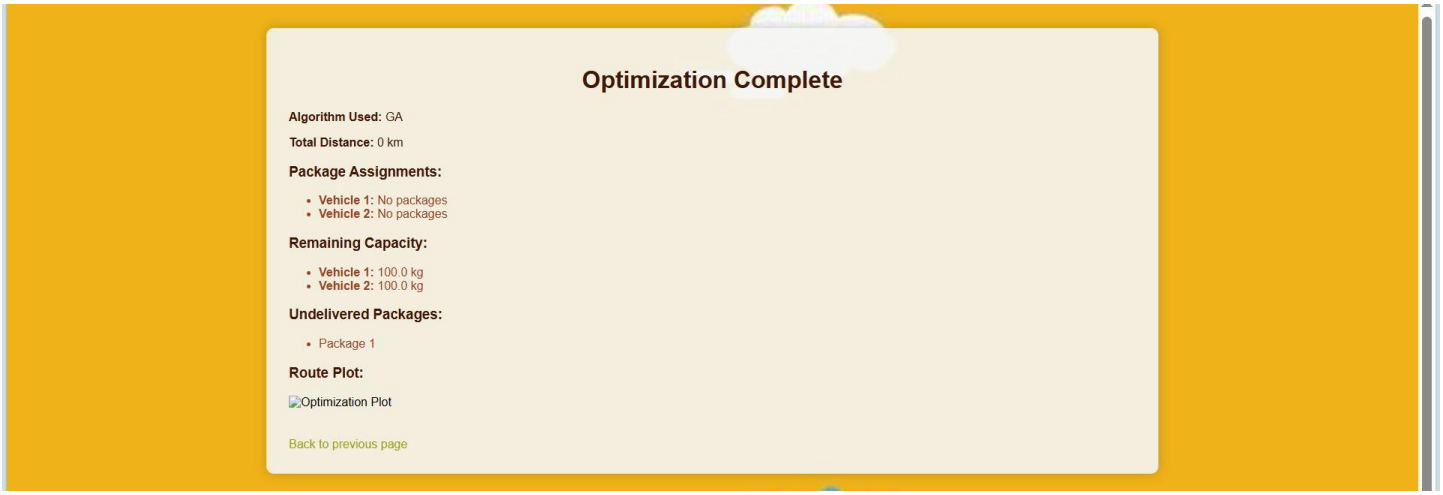
• Objective: Test the system's behaviour when a package exceeds the capacity of all available vehicles.

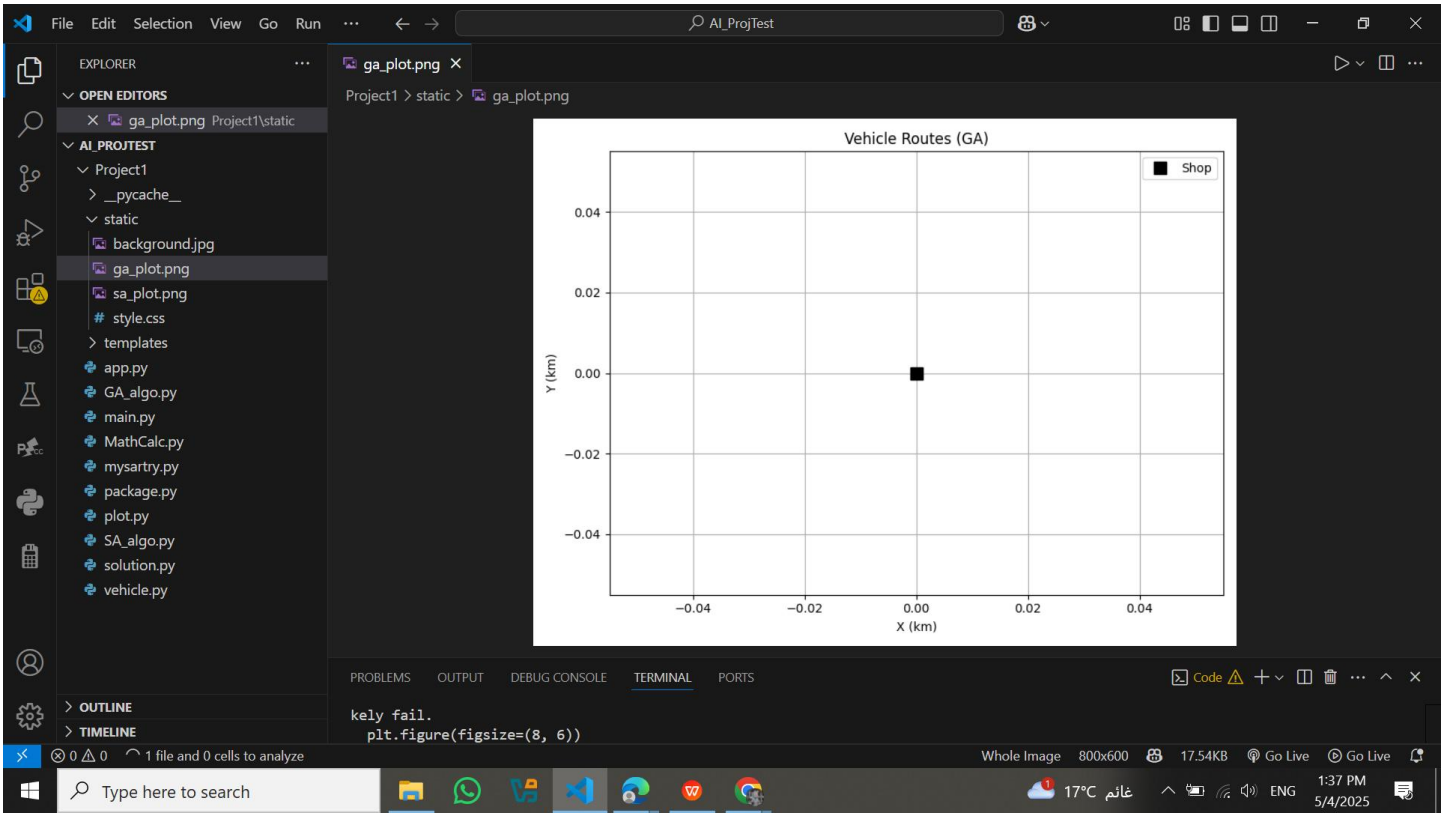
• Input: o Vehicles: 2 vehicles, each with a capacity of 100 kg. o Packages: 1 package weighing 150 kg.

• Expected Outcome: o The system identifies that the package cannot be delivered due to weight constraints.

o Appropriate handling or notification is provided.







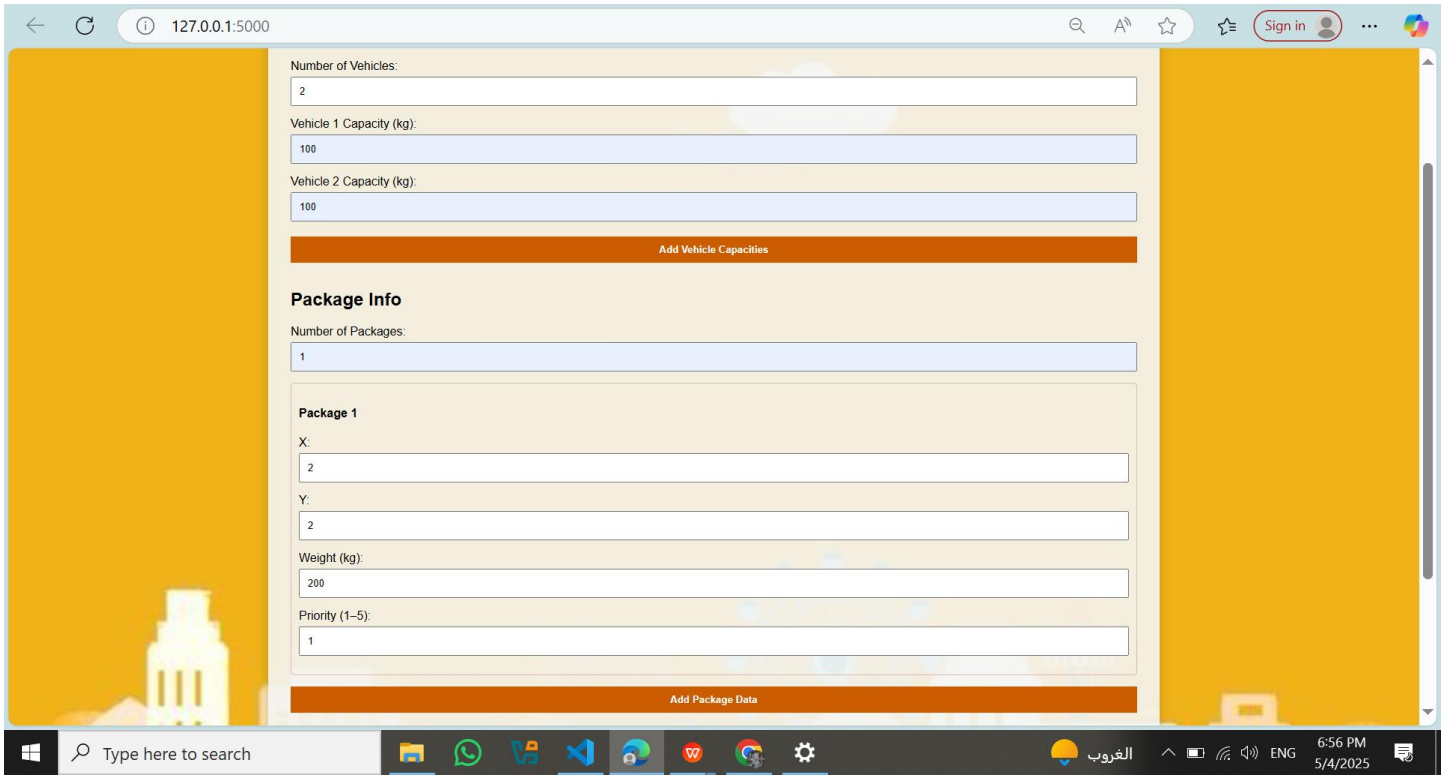
Discussion about this case :

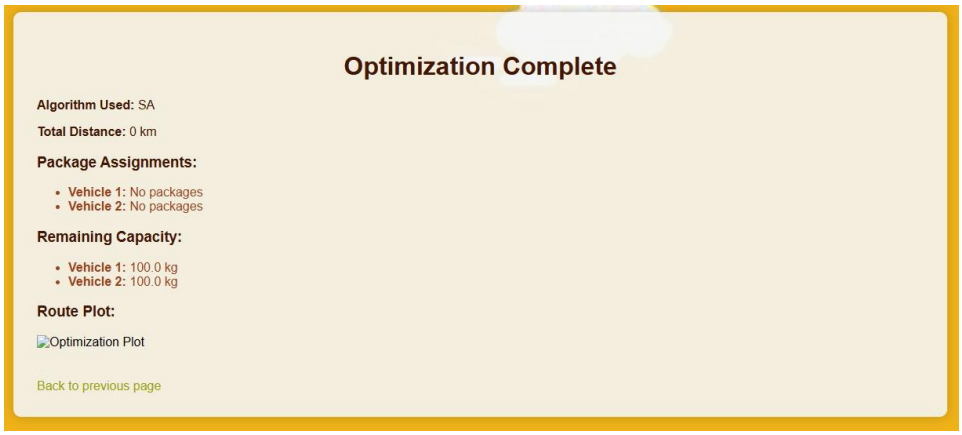
A single package with weight of 150 kg was tested against 2 vehicles, each with a capacity of 100 kg. Since the package exceeded the capacity of both vehicles,our system has rejected the delivery, leaving the package undelivered. The optimization concluded with no distance traveled and confirmed that all vehicles remained unused which is a proper and expected handling of capacity overflow by the Genetic Algorithm.

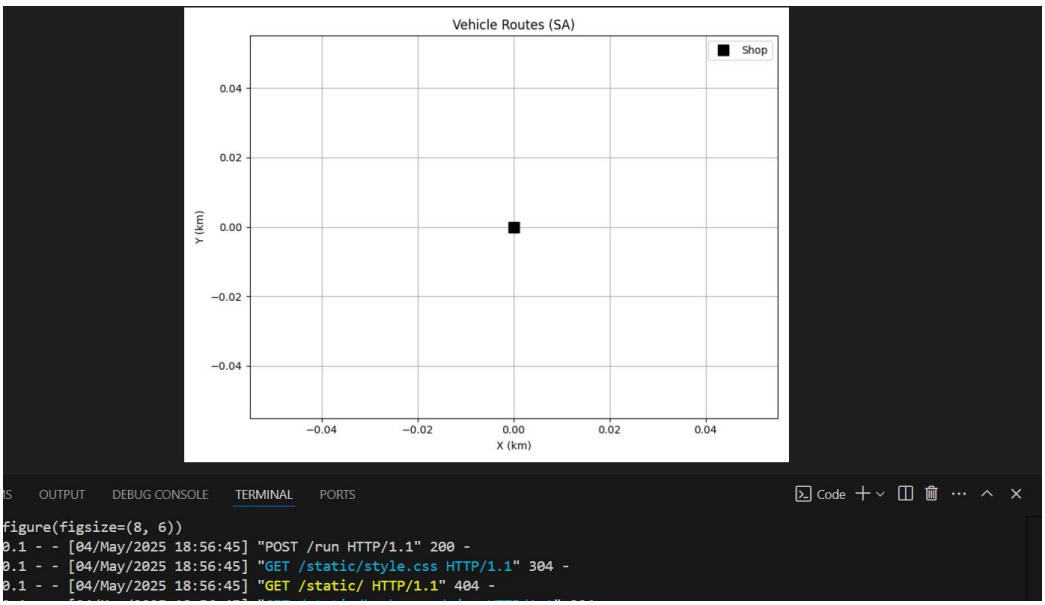
- **All the results are consistent with our expectations.**

SA_Testing cases

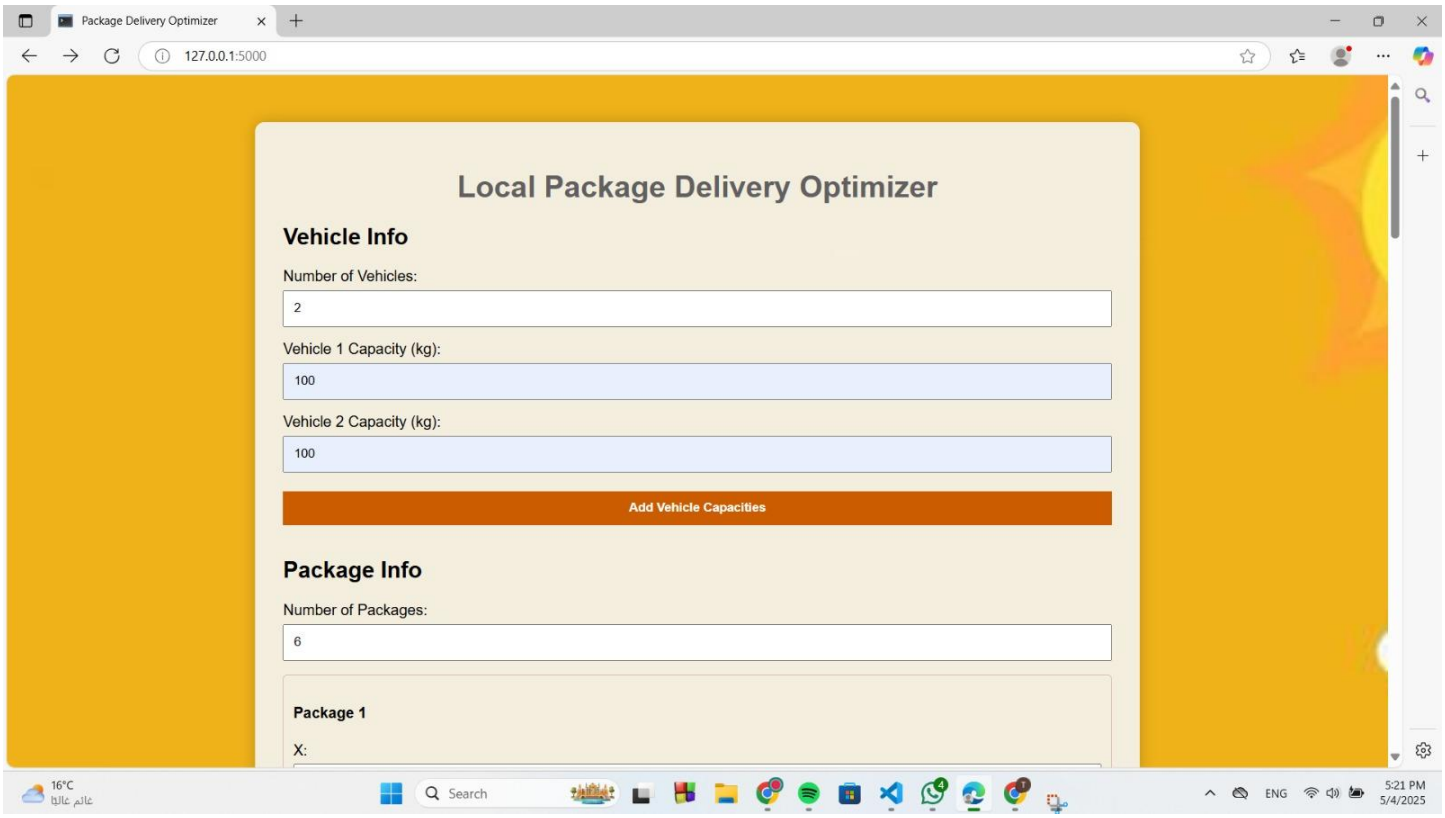
1 package exceeds the vehicle's capacity

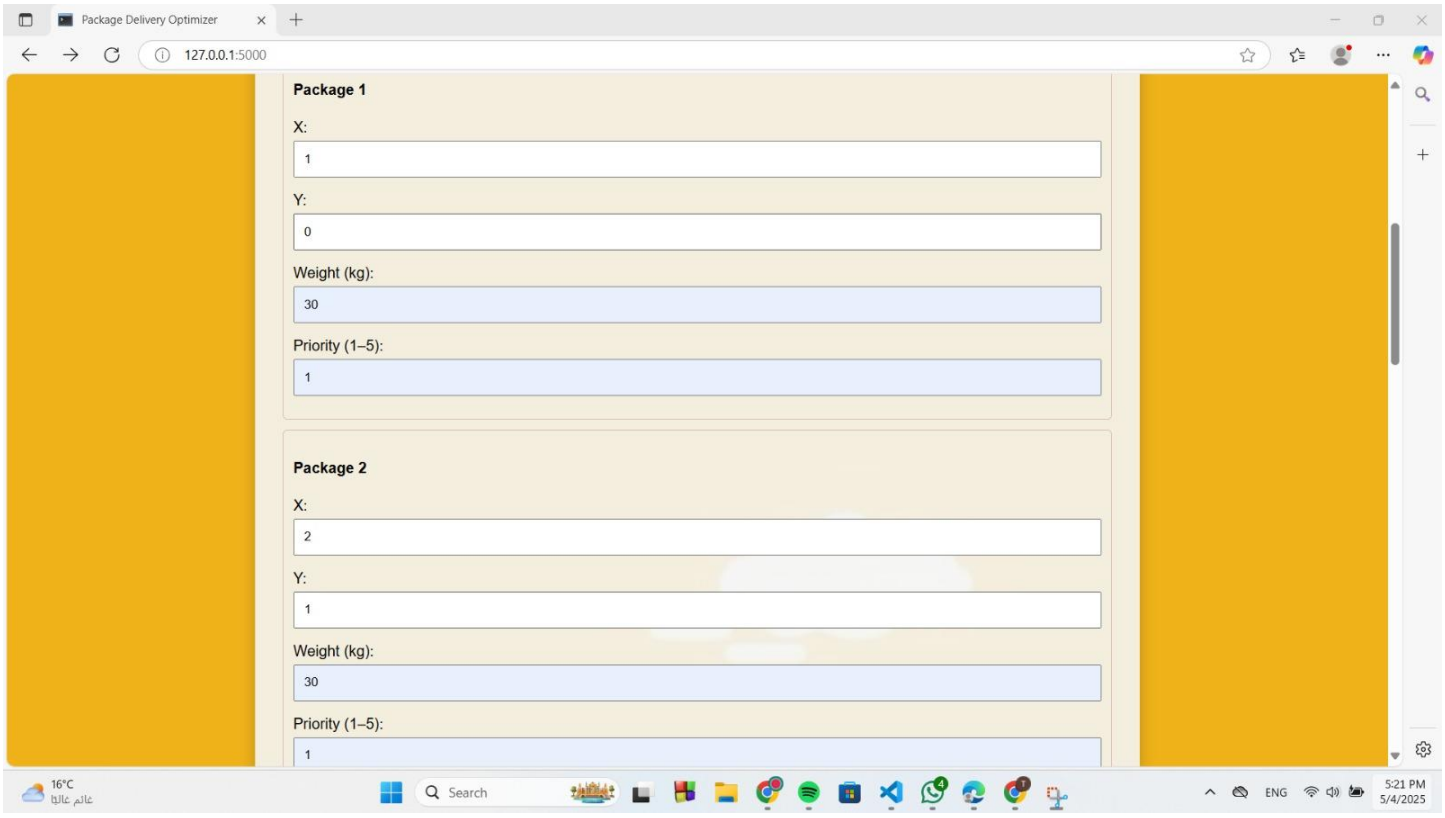


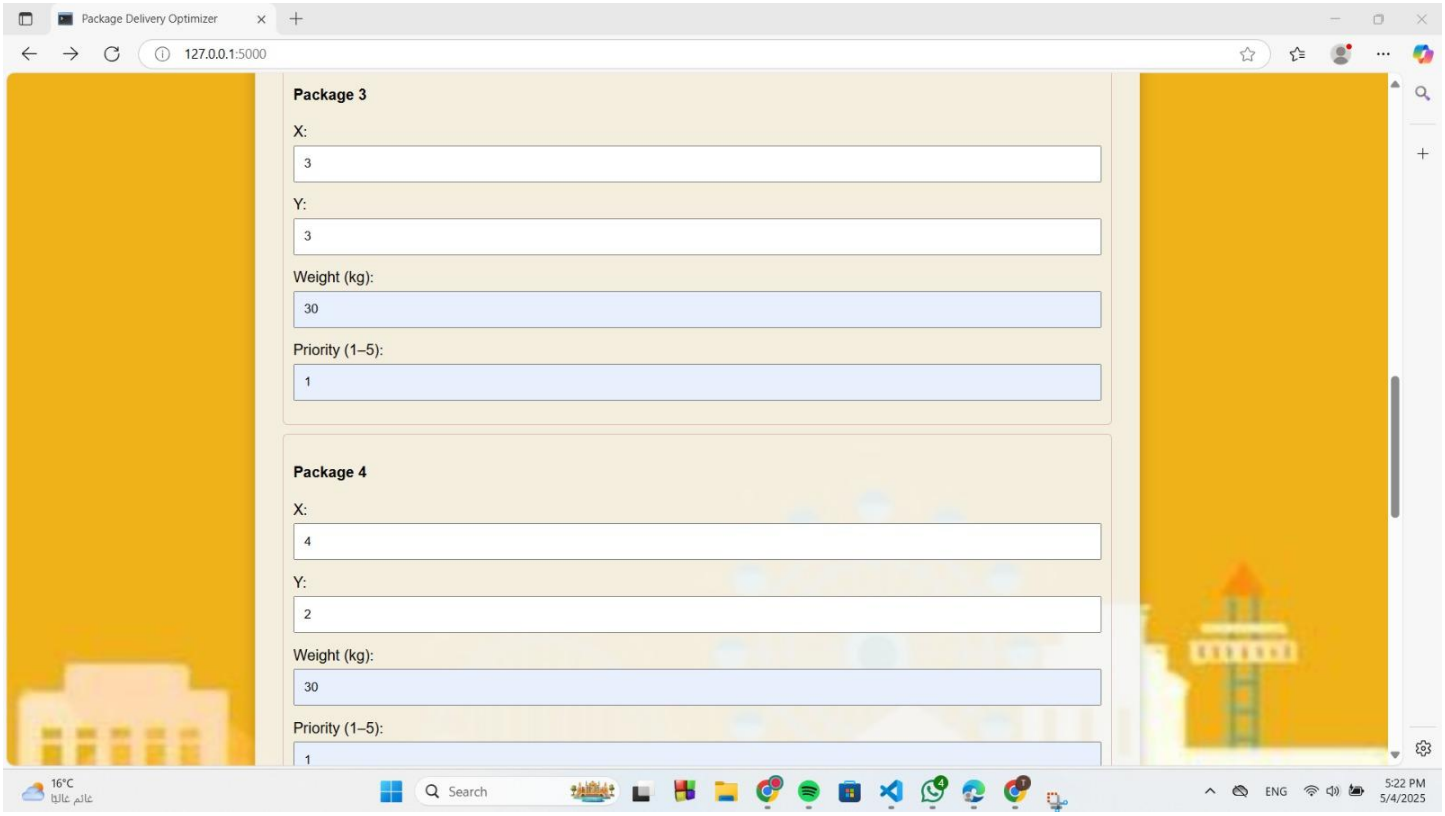


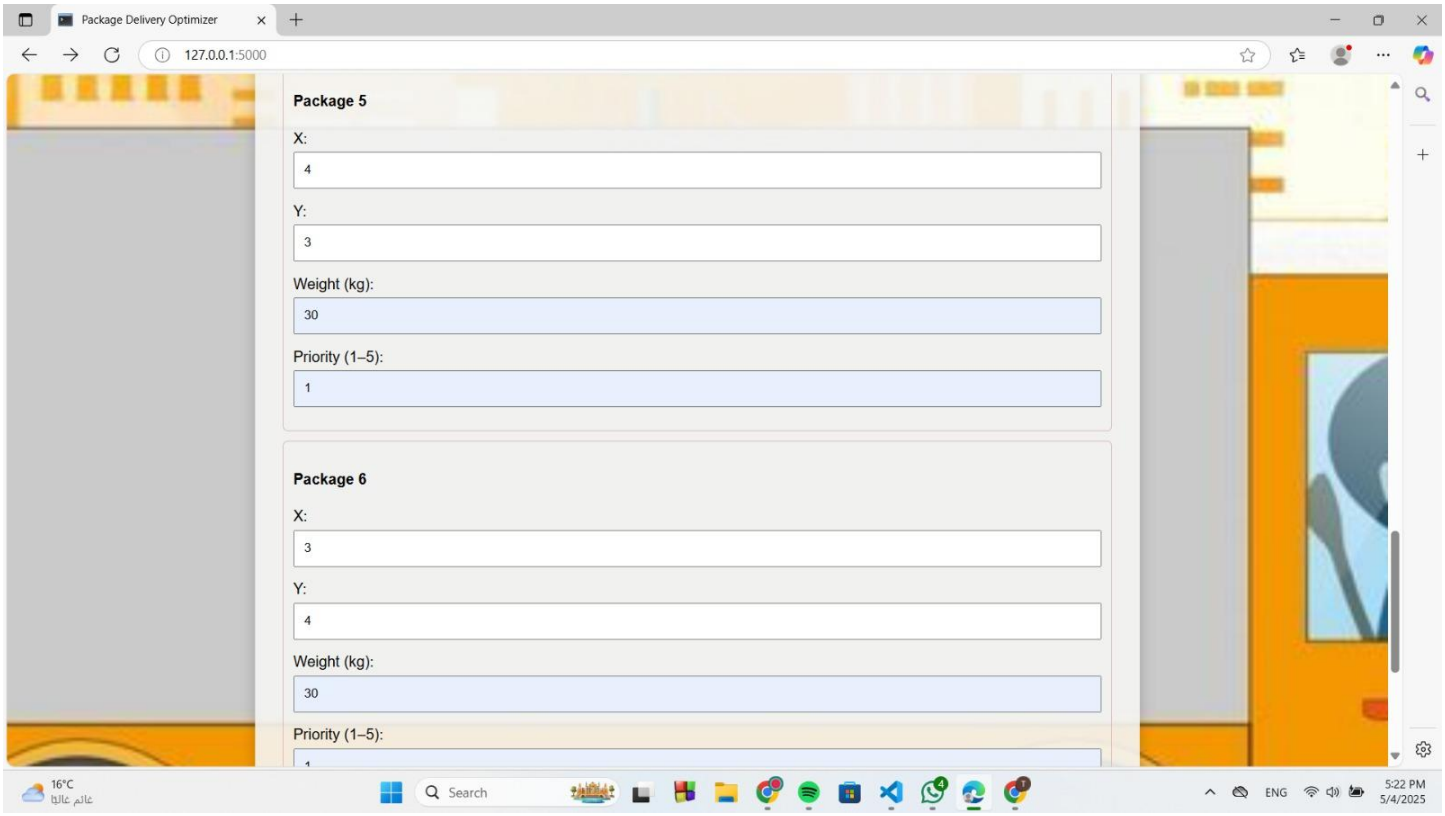


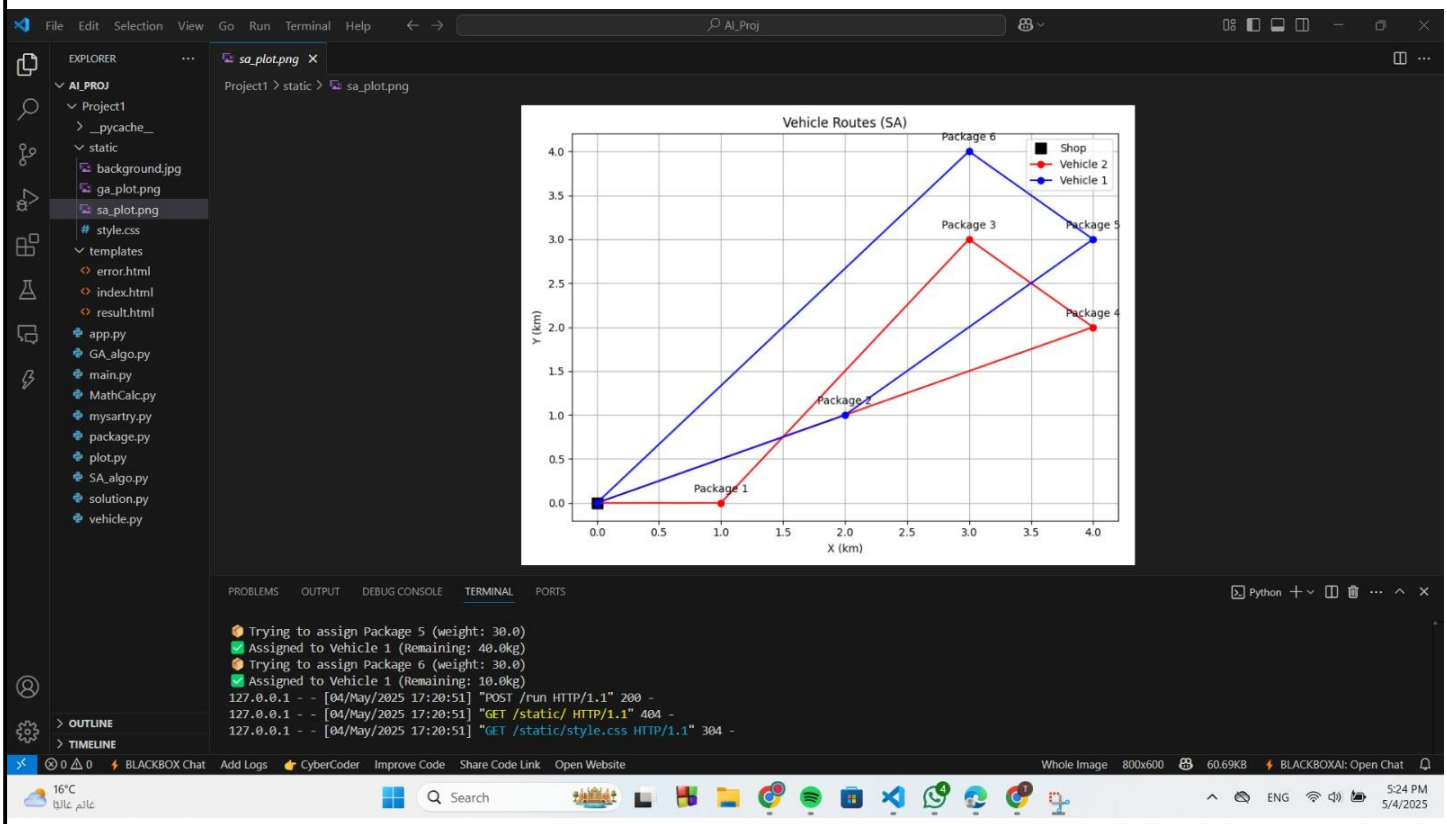
-> 2 vehicles& 6 packages with same weight and priority but different locations:






Package Delivery Optimizer
127.0.0.1:5000
Package 3
X:
3
Y:
3
Weight (kg):
30
Priority (1–5):
1
Package 4
X:
4
Y:
2
Weight (kg):
30
Priority (1–5):
1
16°C
Search
ENG
5:22 PM
5/4/2025


Package Delivery Optimizer
127.0.0.1:5000
Package 5
X:
4
Y:
3
Weight (kg):
30
Priority (1–5):
1
Package 6
X:
3
Y:
4
Weight (kg):
30
Priority (1–5):
16°C
Search
ENG
5:22 PM
5/4/2025



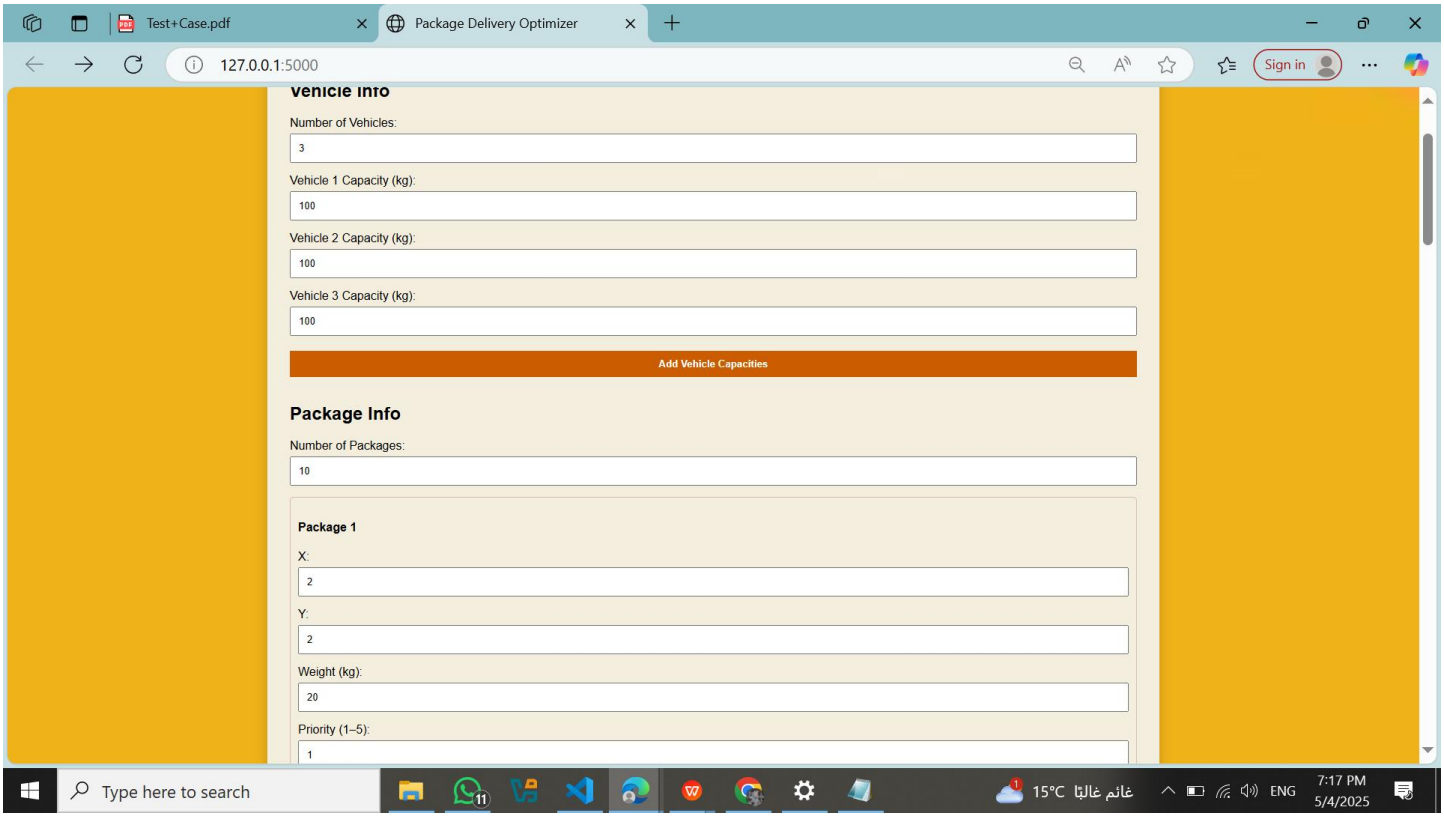
Discussion about this case :

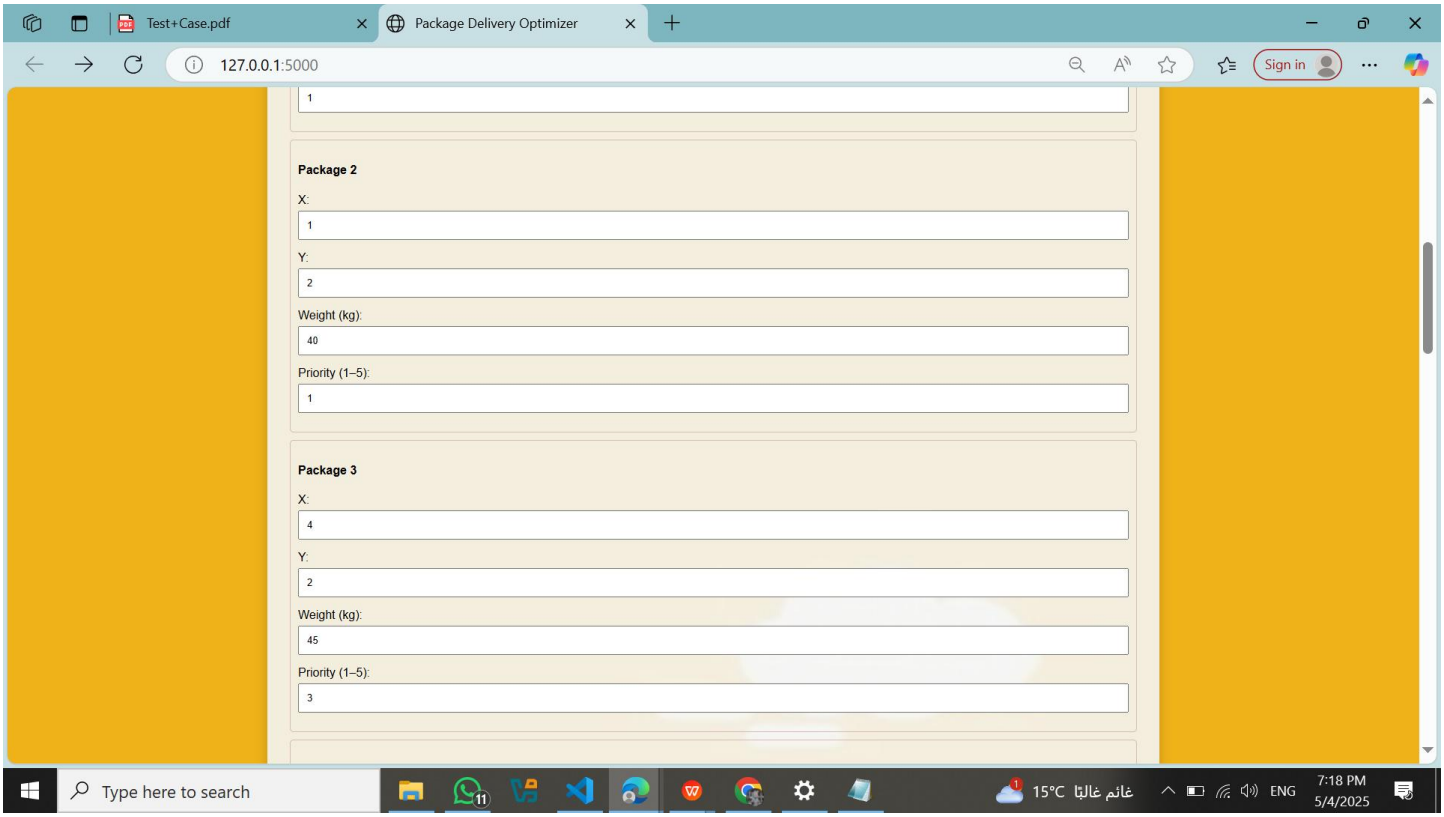
Using this algorithm “Simulated Annealing” (SA) , the system efficiently assigned all 6 packages across 2 vehicles, each with a capacity of 100 kg. Each package weighed 30 kg and had equal priority (1), totaling 180 kg. Our optimizer successfully utilized both vehicles without exceeding capacity limits, distributing packages in a way that minimized total travel distance. The plotted result confirms that all delivery points were reached efficiently, demonstrating the SA algorithm’s effectiveness in handling balanced workloads and uniform priorities.

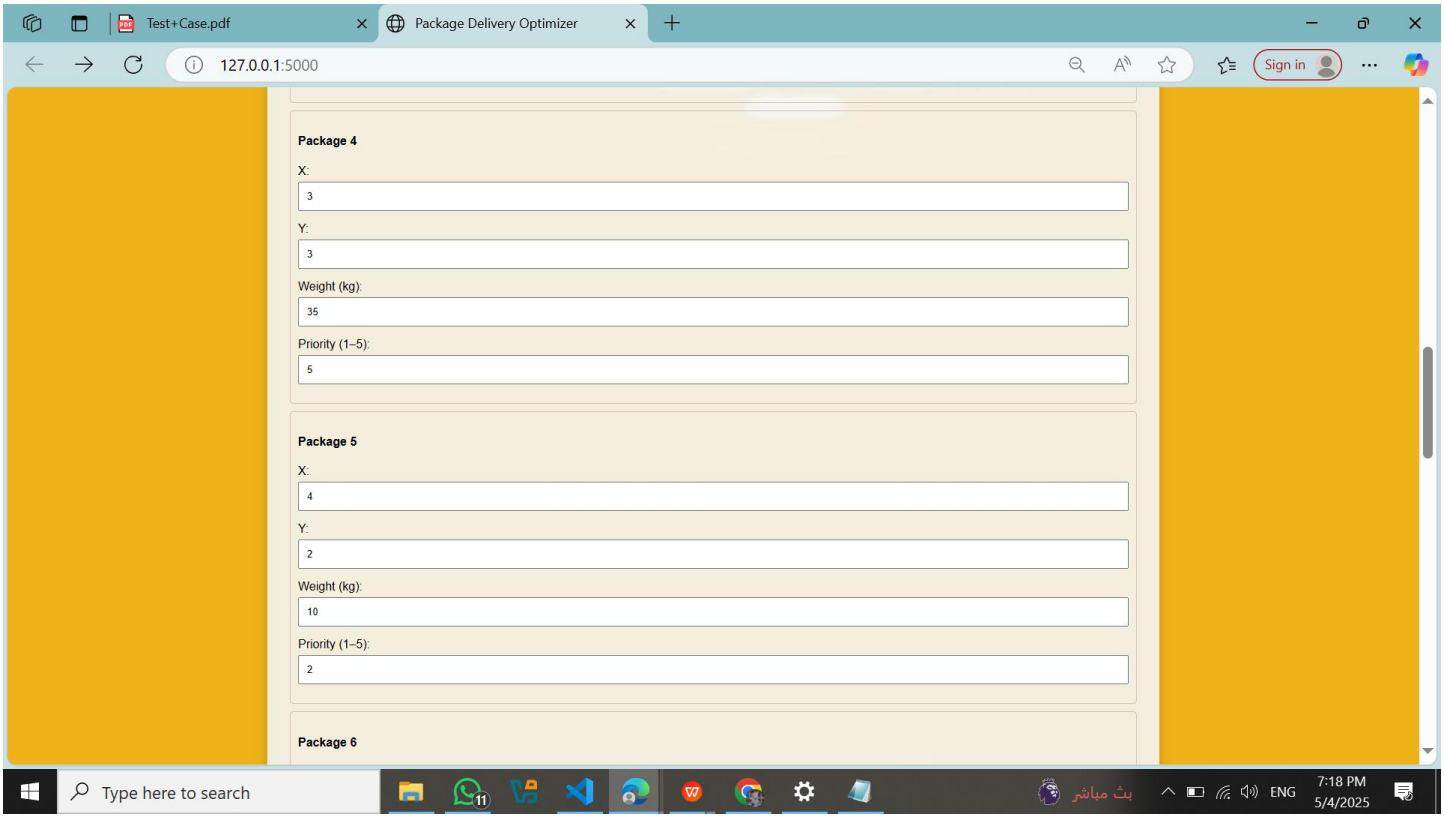
Simulated Annealing vs. Genetic Algorithm Comparison

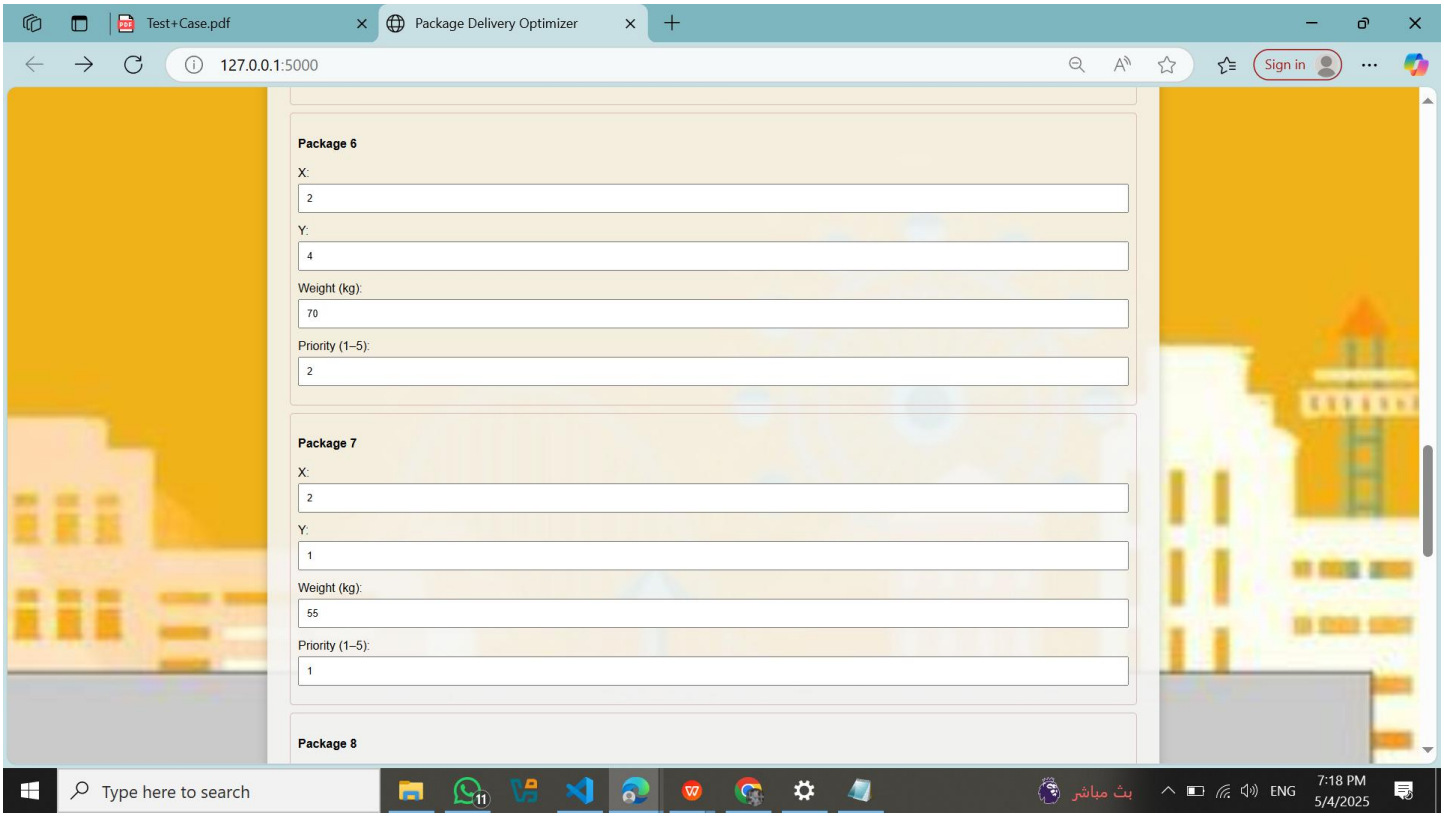
• Input: o Vehicles: 3 vehicles, each with a capacity of 100 kg.

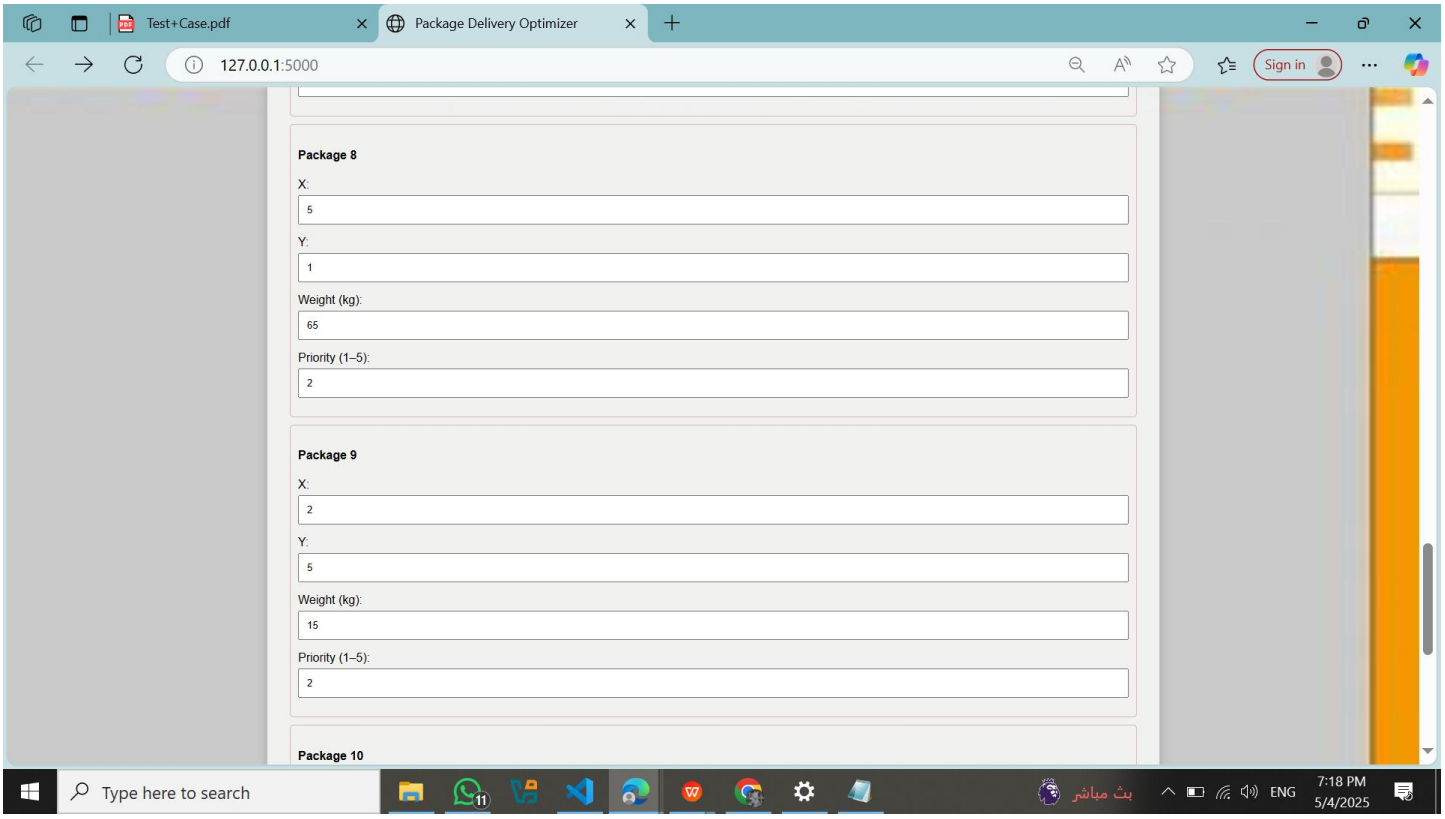
o Packages: 10 packages with varying weights and priorities

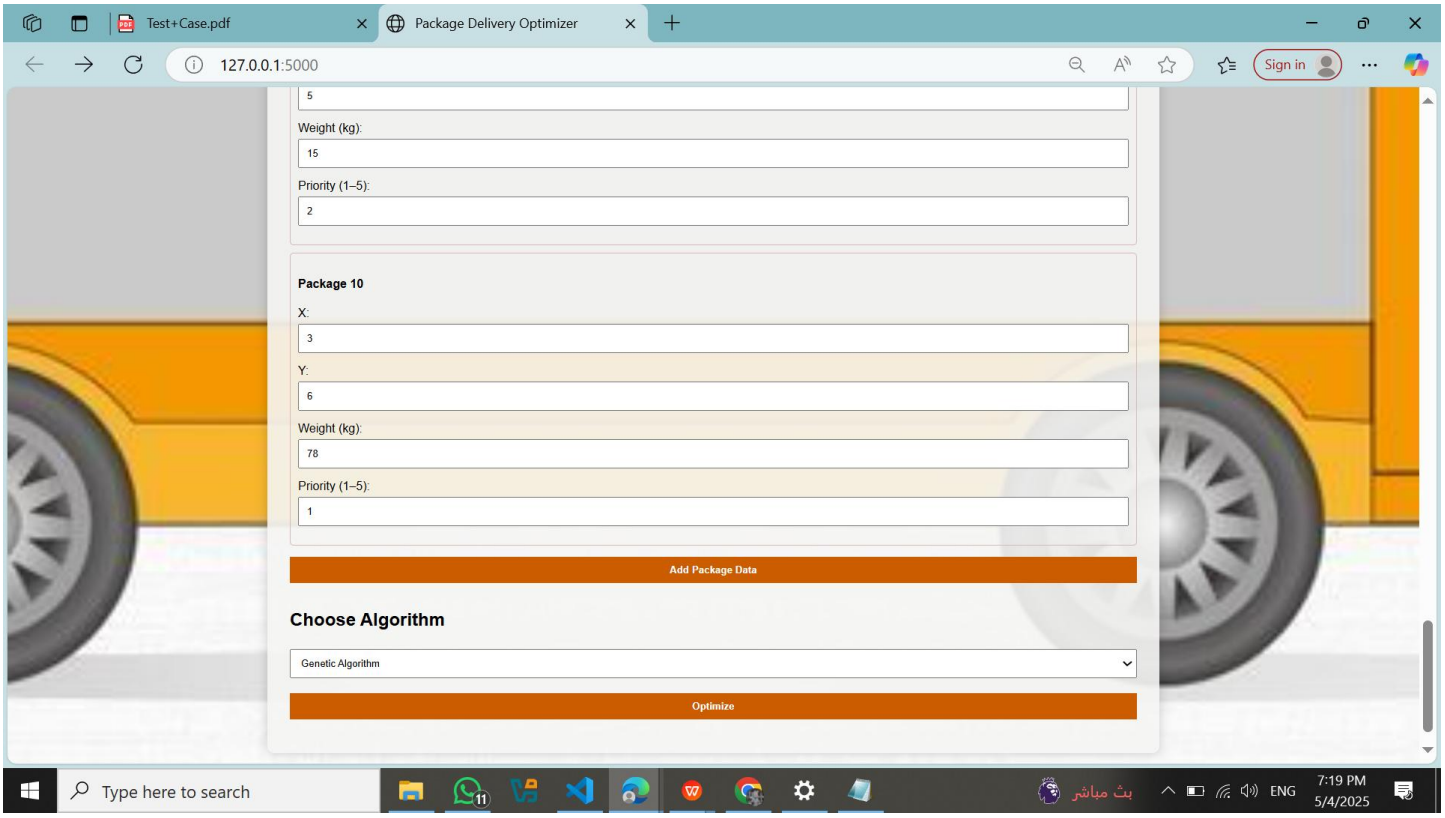


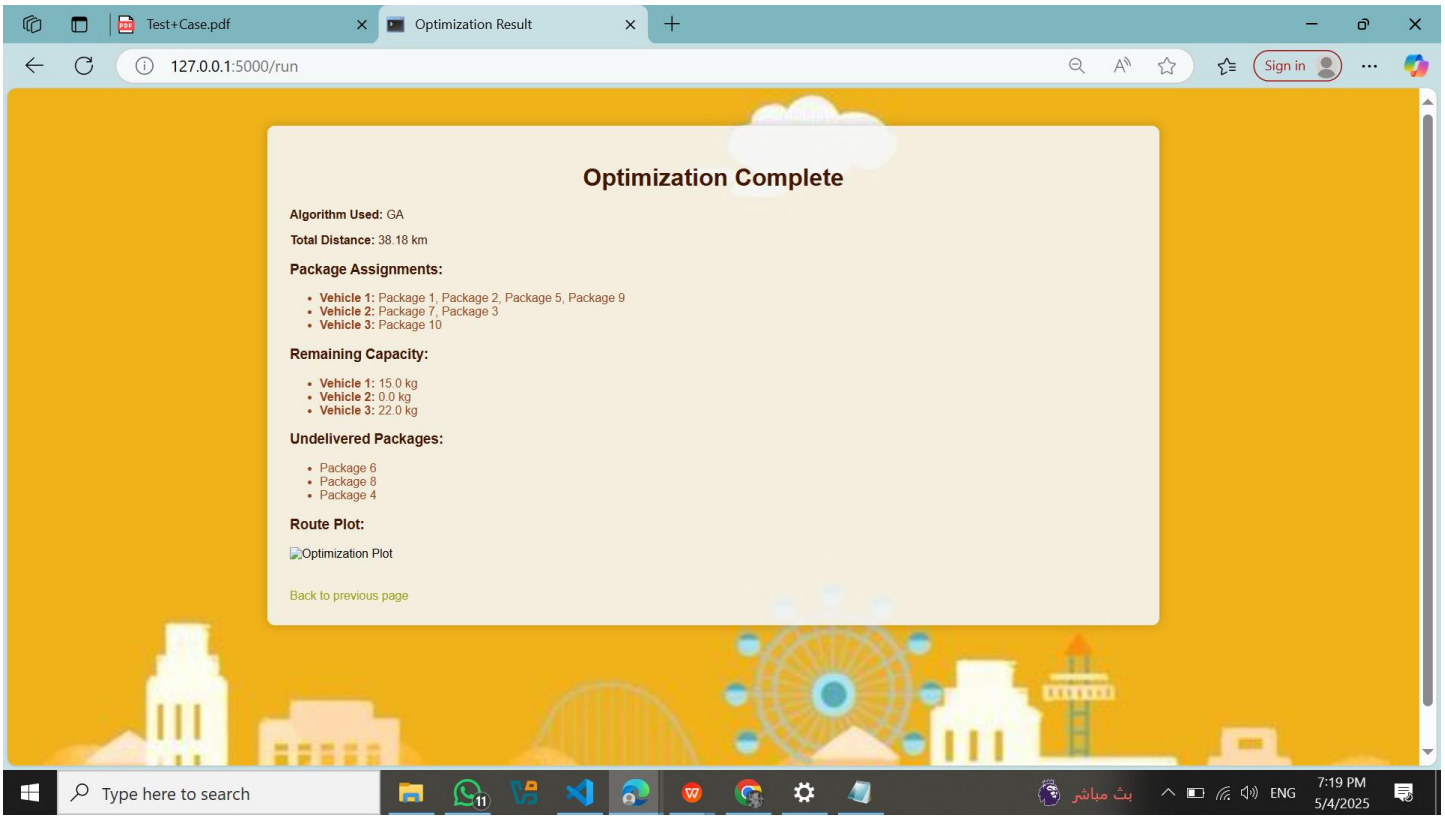


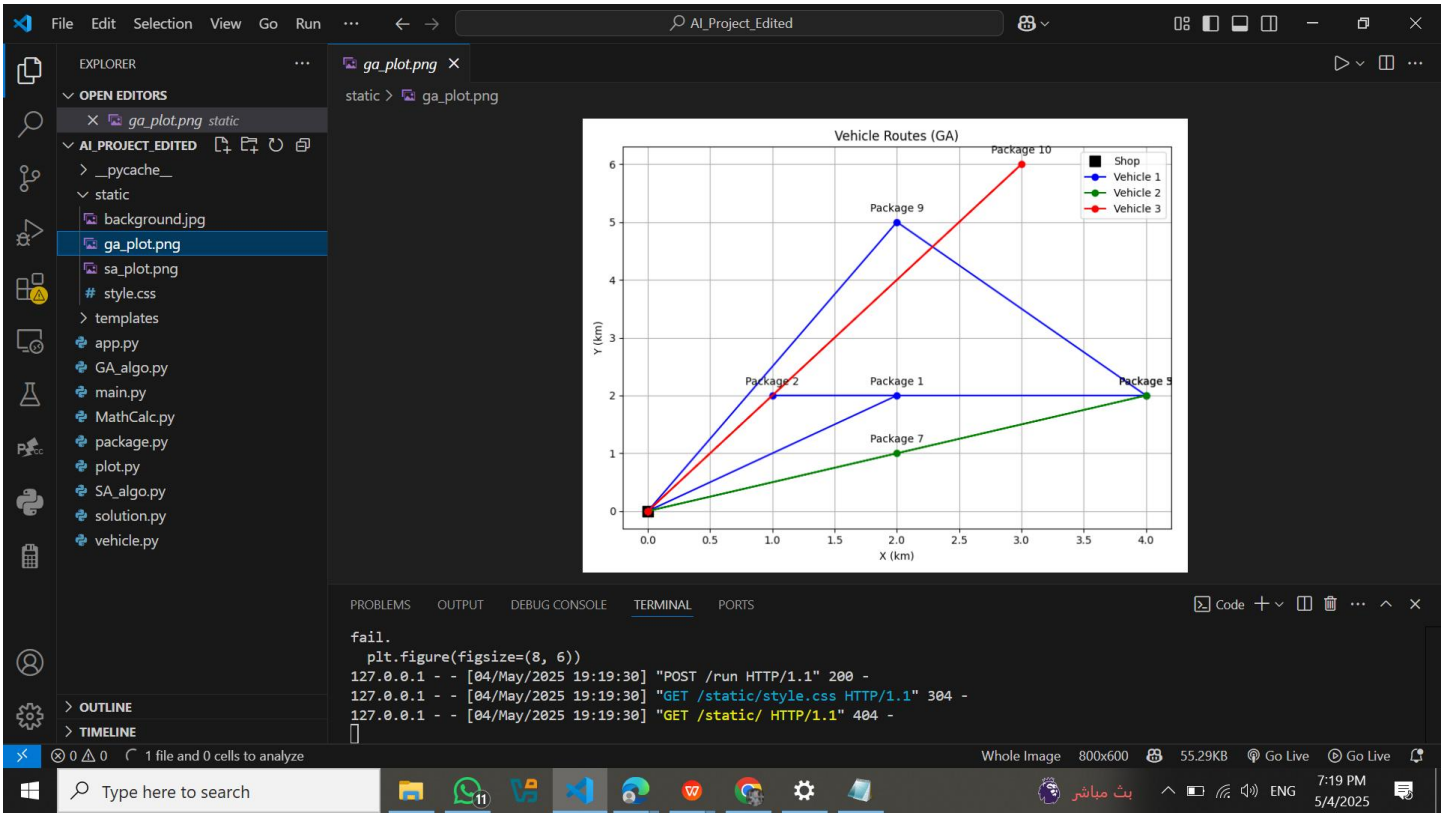

Test+Case.pdf
Package Delivery Optimizer
127.0.0.1:5000
Sign in
Package 4
X:
3
Y:
3
Weight (kg):
35
Priority (1–5):
5
Package 5
X:
4
Y:
2
Weight (kg):
10
Priority (1–5):
2
Package 6
Type here to search
ENG
7:18 PM
5/4/2025

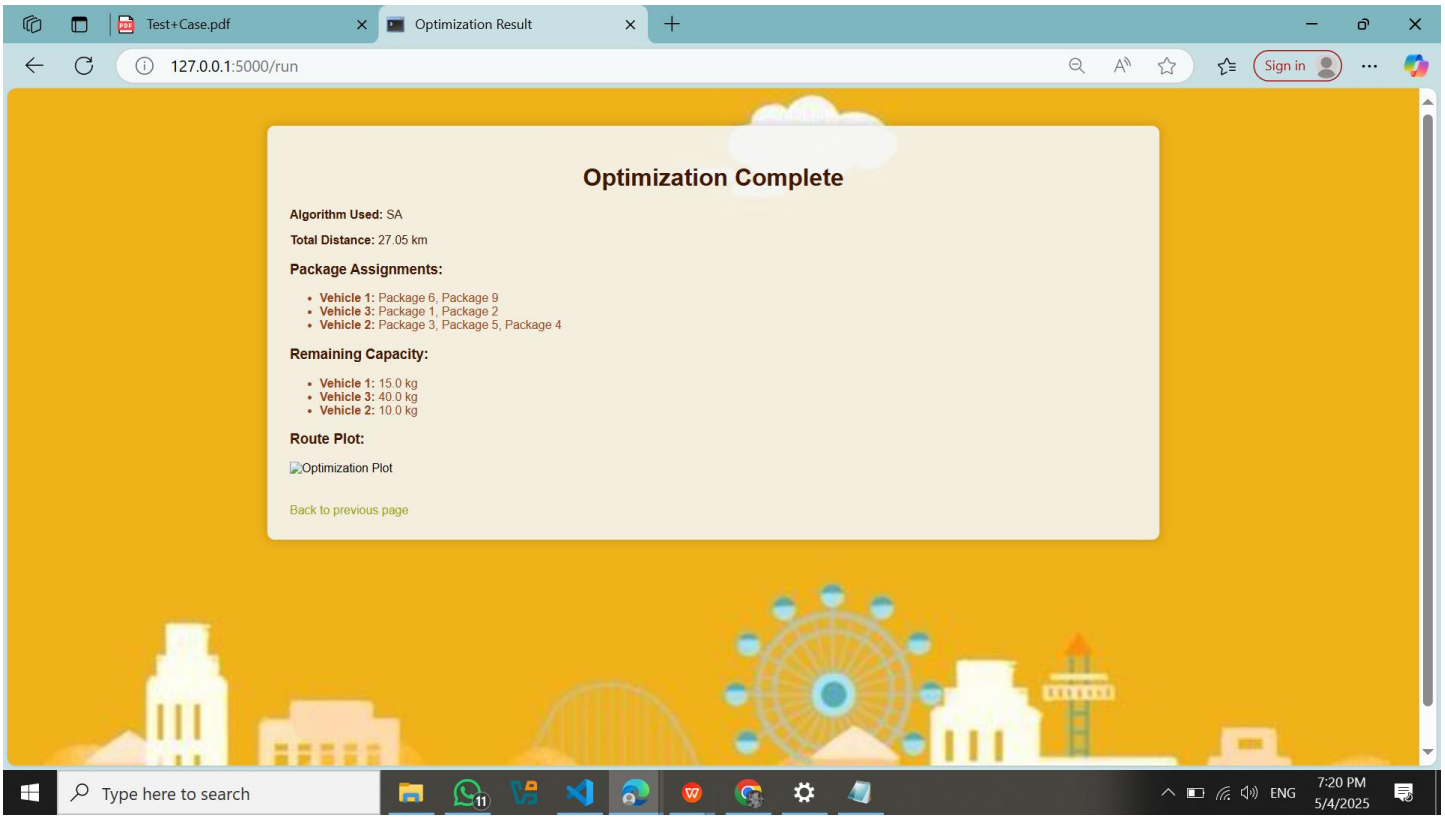

Test+Case.pdf
Package Delivery Optimizer
127.0.0.1:5000
Sign in
Package 6
X:
2
Y:
4
Weight (kg):
70
Priority (1–5):
2
Package 7
X:
2
Y:
1
Weight (kg):
55
Priority (1–5):
1
Package 8
Type here to search
ENG
7:18 PM
5/4/2025

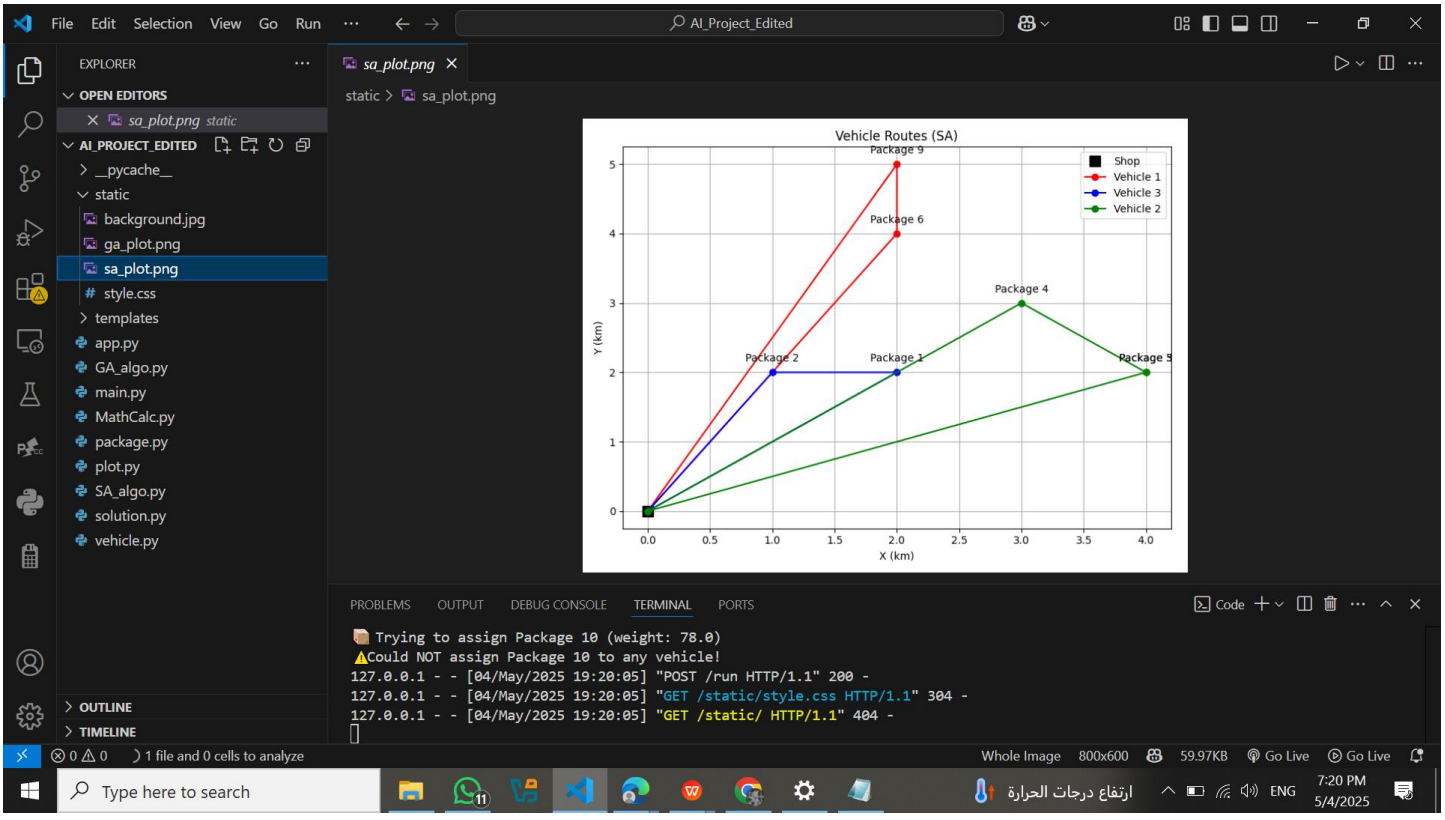

Test+Case.pdf
Package Delivery Optimizer
127.0.0.1:5000
Sign in
Package 8
X:
5
Y:
1
Weight (kg):
65
Priority (1–5):
2
Package 9
X:
2
Y:
5
Weight (kg):
15
Priority (1–5):
2
Package 10
Type here to search
ENG
7:18 PM
5/4/2025


Test+Case.pdf
Package Delivery Optimizer
127.0.0.1:5000
Sign in
5
Weight (kg):
15
Priority (1–5):
2
Package 10
X:
3
Y:
6
Weight (kg):
78
Priority (1–5):
1
Add Package Data
Choose Algorithm
Genetic Algorithm
Optimize
Type here to search
ENG
7:19 PM
5/4/2025


Test+Case.pdf
Optimization Result
127.0.0.1:5000/run
Sign in
Optimization Complete
Algorithm Used: GA
Total Distance: 38.18 km
Package Assignments:
Vehicle 1: Package 1, Package 2, Package 5, Package 9
Vehicle 2: Package 7, Package 3
Vehicle 3: Package 10
Remaining Capacity:
Vehicle 1: 15.0 kg
Vehicle 2: 0.0 kg
Vehicle 3: 22.0 kg
Undelivered Packages:
Package 6
Package 8
Package 4
Route Plot:
Optimization Plot
Back to previous page
Type here to search
ENG
7:19 PM
5/4/2025


File Edit Selection View Go Run
AI_Project_Edited
EXPLORER
OPEN EDITORS
ga_plot.png static
AI_PROJECT_EDITED
__pycache__
static
background.jpg
ga_plot.png
sa_plot.png
style.css
templates
app.py
GA_algo.py
main.py
MathCalc.py
package.py
plot.py
SA_algo.py
solution.py
vehicle.py
OUTLINE
TIMELINE
ga_plot.png
static > ga_plot.png
Vehicle Routes (GA)
Shop
Vehicle 1
Vehicle 2
Vehicle 3
Package 10
Package 9
Package 2
Package 1
Package 7
Y (km)
X (km)
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
fail.
plt.figure(figsize=(8, 6))
127.0.0.1 - - [04/May/2025 19:19:30] "POST /run HTTP/1.1" 200 -
127.0.0.1 - - [04/May/2025 19:19:30] "GET /static/style.css HTTP/1.1" 304 -
127.0.0.1 - - [04/May/2025 19:19:30] "GET /static/ HTTP/1.1" 404 -
1 file and 0 cells to analyze
Whole Image 800x600 55.29KB
Type here to search
ENG
7:19 PM
5/4/2025

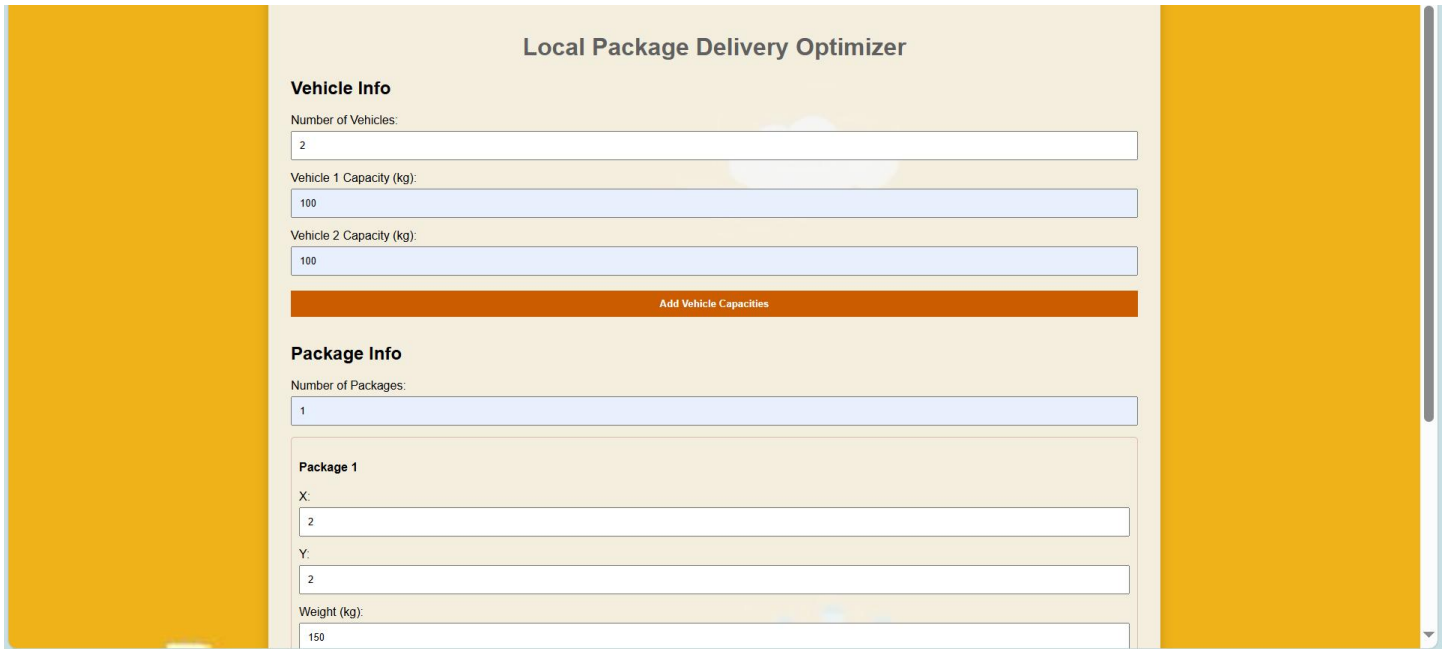

Test+Case.pdf
Optimization Result
127.0.0.1:5000/run
Sign in
Optimization Complete
Algorithm Used: SA
Total Distance: 27.05 km
Package Assignments:
Vehicle 1: Package 6, Package 9
Vehicle 3: Package 1, Package 2
Vehicle 2: Package 3, Package 5, Package 4
Remaining Capacity:
Vehicle 1: 15.0 kg
Vehicle 3: 40.0 kg
Vehicle 2: 10.0 kg
Route Plot:
Optimization Plot
Back to previous page
Type here to search
ENG
7:20 PM
5/4/2025

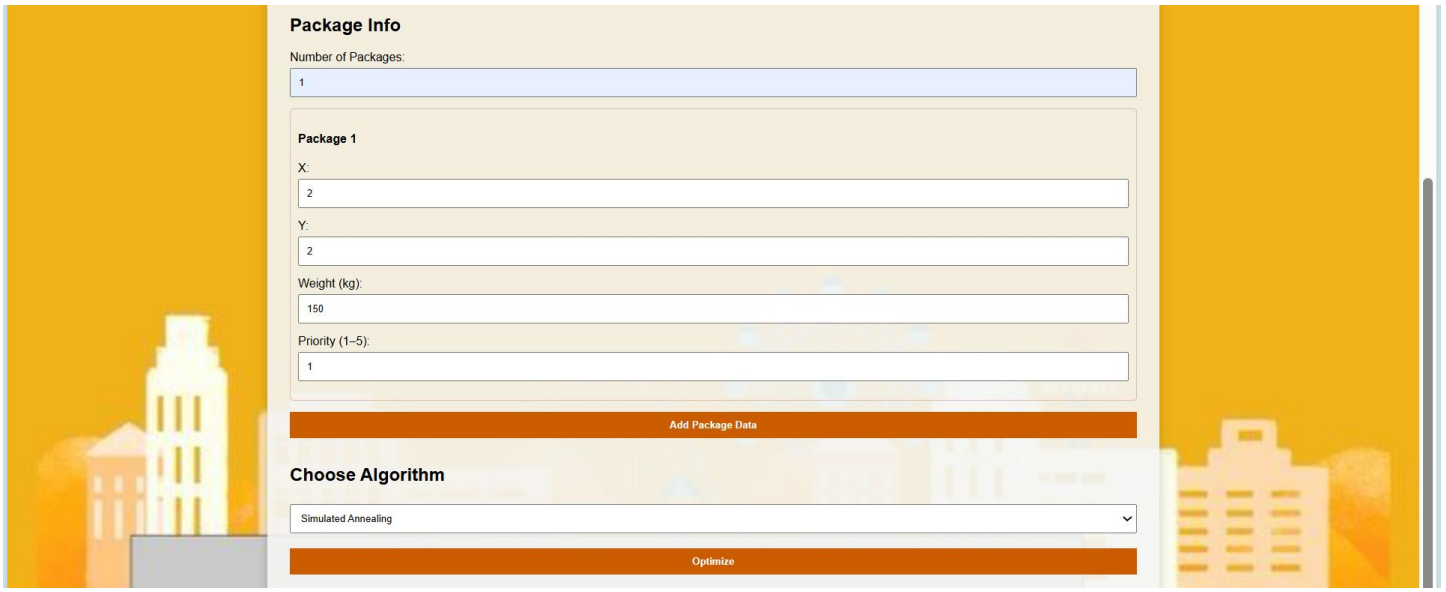

File Edit Selection View Go Run
AI_Project_Edited
EXPLORER
OPEN EDITORS
sa_plot.png static
AI_PROJECT_EDITED
__pycache__
static
background.jpg
ga_plot.png
sa_plot.png
style.css
templates
app.py
GA_algo.py
main.py
MathCalc.py
package.py
plot.py
SA_algo.py
solution.py
vehicle.py
sa_plot.png
static > sa_plot.png
Vehicle Routes (SA)
Package 9
Package 6
Package 4
Package 2
Package 1
Package 5
Shop
Vehicle 1
Vehicle 3
Vehicle 2
Y (km)
X (km)
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Trying to assign Package 10 (weight: 78.0)
Could NOT assign Package 10 to any vehicle!
127.0.0.1 - - [04/May/2025 19:20:05] "POST /run HTTP/1.1" 200 -
127.0.0.1 - - [04/May/2025 19:20:05] "GET /static/style.css HTTP/1.1" 304 -
127.0.0.1 - - [04/May/2025 19:20:05] "GET /static/ HTTP/1.1" 404 -
OUTLINE
TIMELINE
1 file and 0 cells to analyze
Whole Image 800x600 59.97KB Go Live
Type here to search
ارتفاع درجات الحرارة
ENG
7:20 PM
5/4/2025

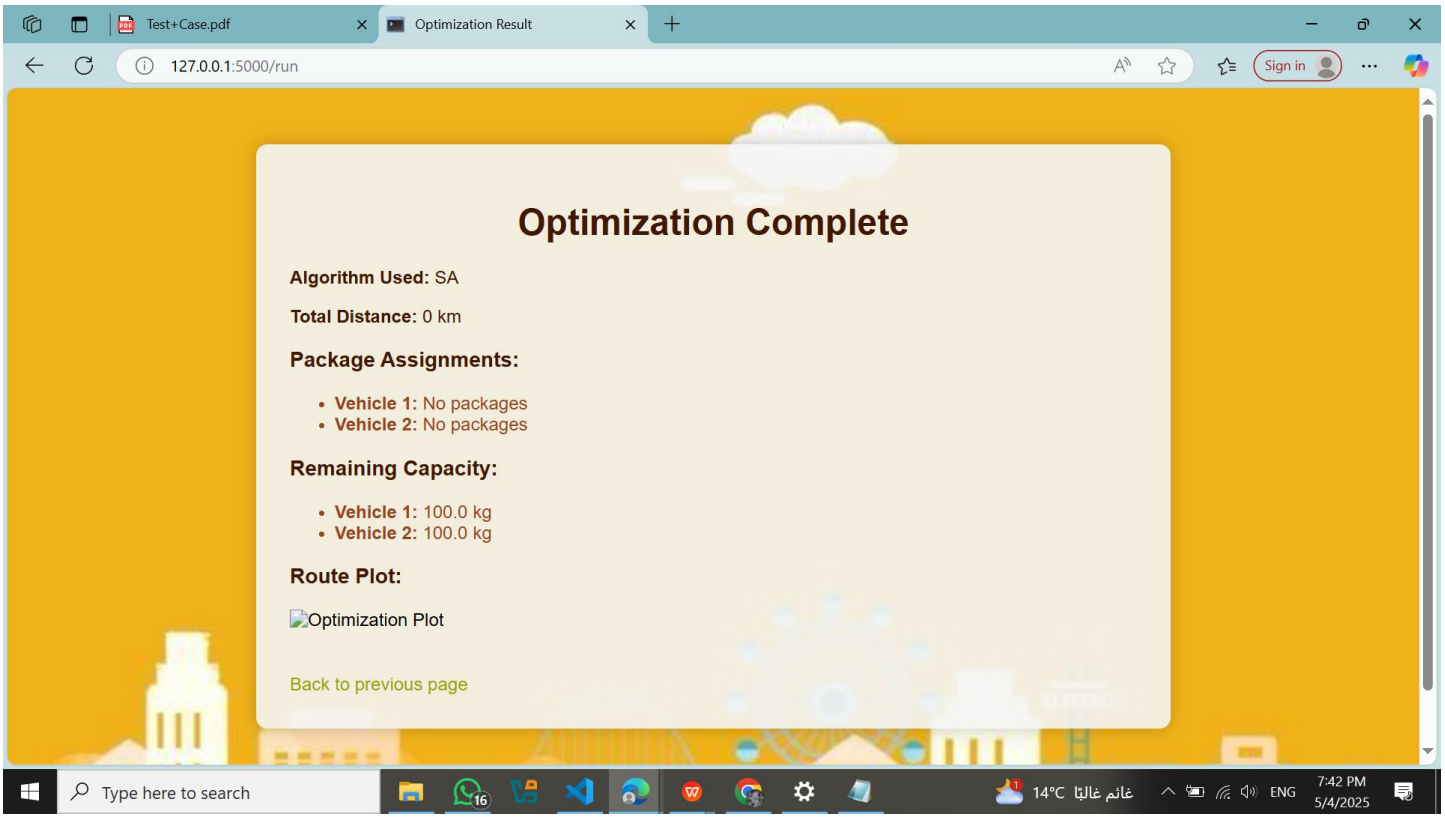
Discussion about this case :

In this delivery optimization task, both Genetic Algorithm (GA) and Simulated Annealing (SA) successfully generated valid solutions while respecting all vehicle capacity constraints. However, their performance varied notably. The SA algorithm achieved a lower total delivery distance of 27.05 km, compared to GA's 38.18 km, indicating a more efficient route overall. Additionally, SA managed to deliver more packages(only package#10 was left unassigned) due to its high weight (78 kg) . In contrast, GA left 3 packages (4, 6, and 8) undelivered, despite all vehicles having the same "100 kg" capacity. This suggests that SA was better at distributing packages to maximize capacity usage and coverage. On the other hand, GA algorithm still provided a decent solution, and its visual route was more evenly spread across vehicles. But overall, in this case, SA outperformed GA in both delivery efficiency and package coverage, especially when dealing with weight priority trade offs.

Edge Case - Overcapacity Package (SA algo)







Discussion:

In this case, we tested how the system handles a situation where a package is too heavy for any available vehicle to carry ,for example: a 150 kg package with 2 vehicles, each limited to 100 kg. As expected, our system didn't assign the package to any vehicle and reported a total travel distance of zero. This means the algorithm respected the weight constraints and avoided forcing an impossible delivery.