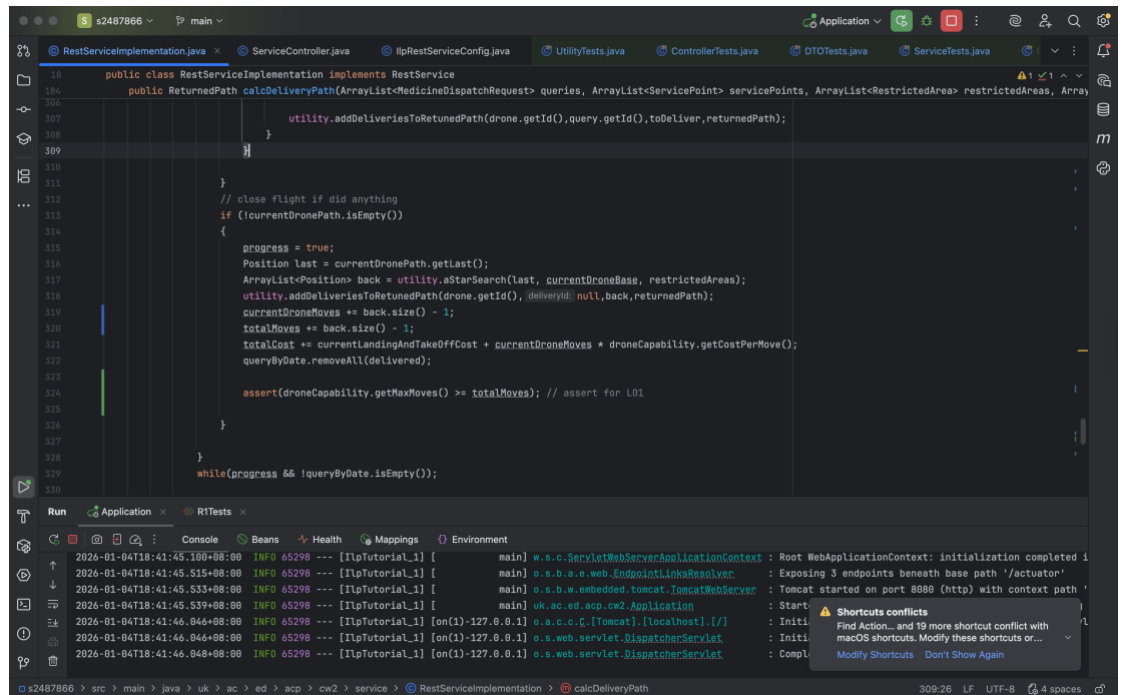


## Testing Log (Bugs Found)

1. The method for calculating the moves was wrong.
  - a) Problem: The old method was trying to use the length of the path array (i.e. how many positions in the list)
  - b) Result: This results in the calculated move being more than the actual
    - i. The real move should be the number of positions minus 1
    - ii. If a point is 495 moves away, the algorithm would return totalMoves = 993, which is wrong. Should be 901
  - c) Screenshots for old code and fixes
    - i. Old code



The screenshot shows an IDE with the following components:

- Editor:** Displays the `RestServiceImplementation.java` file. The `calcDeliveryPath` method is visible, which calculates the delivery path and moves. The code includes comments and logic for handling drone paths, restricted areas, and move calculations. A comment at line 124 states: `assert(droneCapability.getMaxMoves() >= totalMoves); // assert for L01`.
- Run Console:** Shows the output of the application. The logs indicate that the application started successfully on port 8080. The logs include the following information:
  - 2026-01-04T18:41:45.100+08:00 INFO 65298 --- [IIPTutorial\_1] [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed
  - 2026-01-04T18:41:45.515+08:00 INFO 65298 --- [IIPTutorial\_1] [main] o.s.b.a.e.web.EndpointLinksResolver : Exposing 3 endpoints beneath base path '/actuator'
  - 2026-01-04T18:41:45.533+08:00 INFO 65298 --- [IIPTutorial\_1] [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8080 (http) with context path ''
  - 2026-01-04T18:41:45.539+08:00 INFO 65298 --- [IIPTutorial\_1] [main] uk.ac.ed.acp.cw2.Application : Start
  - 2026-01-04T18:41:46.046+08:00 INFO 65298 --- [IIPTutorial\_1] [on(1)-127.0.0.1] o.a.c.c.g.[Tomcat].[localhost].[/] : Initi
  - 2026-01-04T18:41:46.046+08:00 INFO 65298 --- [IIPTutorial\_1] [on(1)-127.0.0.1] o.s.web.servlet.DispatcherServlet : Initi
  - 2026-01-04T18:41:46.048+08:00 INFO 65298 --- [IIPTutorial\_1] [on(1)-127.0.0.1] o.s.web.servlet.DispatcherServlet : Compl

- ii. Fixed c

```

18 public class RestServiceImplementation implements RestService
19 {
20     public ReturnedPath calcDeliveryPath(ArrayList<MedicineDispatchRequest> queries, ArrayList<ServicePoint> servicePoints, ArrayList<RestrictedArea> restrictedAreas, ArrayList<Position> toDeliver)
21     {
22         start = droneBase;
23     }
24     else
25     {
26         start = currentDronePath.getLast();
27     }
28     end = query.getDelivery();
29
30     ArrayList<Position> toDeliver = utility.aStarSearch(start, end, restrictedAreas);
31     // For hover
32     toDeliver.add(toDeliver.getLast());
33     int movesTo = toDeliver.size()-1;
34
35     ArrayList<Position> toBase = utility.aStarSearch(end, droneBase, restrictedAreas);
36     int movesBack = toBase.size()-1;
37
38     if (toDeliver.isEmpty() || toBase.isEmpty())
39     {
40         // no valid path, treat as cannot deliver this query
41         continue;
42     }
43
44     int estimatedCurrentDroneMoves = currentDroneMoves + (movesTo + movesBack);
45     int estimatedCurrentNumberOfDeliveries = currentNumberOfDeliveries + 1;
46 }

```

2. The code for adding the hover point was wrong
  - a) Problem: The old method is trying to add the last point from the toDeliver list without checking if that is empty or null
  - b) Result: This results in a NullPointerException
    - i. If the delivery point is completely blocked, i.e. no path can be found, then an error will be returned
  - c) Screenshots for old code and fixes
    - i. Old code

```

18 public class RestServiceImplementation implements RestService
19 {
20     public ReturnedPath calcDeliveryPath(ArrayList<MedicineDispatchRequest> queries, ArrayList<ServicePoint> servicePoints, ArrayList<RestrictedArea> restrictedAreas, ArrayList<Position> toDeliver)
21     {
22         start = droneBase;
23     }
24     else
25     {
26         start = currentDronePath.getLast();
27     }
28     end = query.getDelivery();
29
30     ArrayList<Position> toDeliver = utility.aStarSearch(start, end, restrictedAreas);
31     // For hover
32     // toDeliver.add(toDeliver.getLast());
33     int movesTo = toDeliver.size()-1;
34
35     ArrayList<Position> toBase = utility.aStarSearch(end, droneBase, restrictedAreas);
36     int movesBack = toBase.size()-1;
37
38     if (toDeliver.isEmpty() || toBase.isEmpty())
39     {
40         // no valid path, treat as cannot deliver this query
41         continue;
42     }
43
44     int estimatedCurrentDroneMoves = currentDroneMoves + (movesTo + movesBack);
45     int estimatedCurrentNumberOfDeliveries = currentNumberOfDeliveries + 1;
46     double estimatedCurrentFlightCost = currentLandingAndTakeOffCost + estimatedCurrentDroneMoves * droneCapability.g
47     currentCostPerDelivery = estimatedCurrentFlightCost / estimatedCurrentNumberOfDeliveries;
48 }

```

- ii. Fixed code

The screenshot shows an IDE with the following components:

- Project Explorer:** Shows a project structure with packages like `uk.ac.ed.acp.cw2`, `configuration`, `controller`, `data`, `handler`, `service`, `utility`, `validation`, `Application`, `resources`, `test`, `target`, and `tests`.
- Editor:** Displays the `RestServiceImpl.java` file. The code includes a `calcDeliveryPath` method that uses A\* search to find a path from a base to a destination, avoiding restricted areas. It calculates estimated current drone moves, number of deliveries, flight cost, and cost per delivery.
- Run Console:** Shows the output of running the application. It indicates that 5 tests passed (P1Test, P2Test, P3Test, P4Test, P5Test) and displays a ASCII art logo of a robot.

3. The A star search cut-out moves were set too high
  - a) Problem: If there are no possible paths, it may take a long time for the algorithm to return a JSON object with no deliveries
  - b) Fix: Changed from 100000000 to 1000000