Project 3 Documentation

# UML Diagram

# How to Use

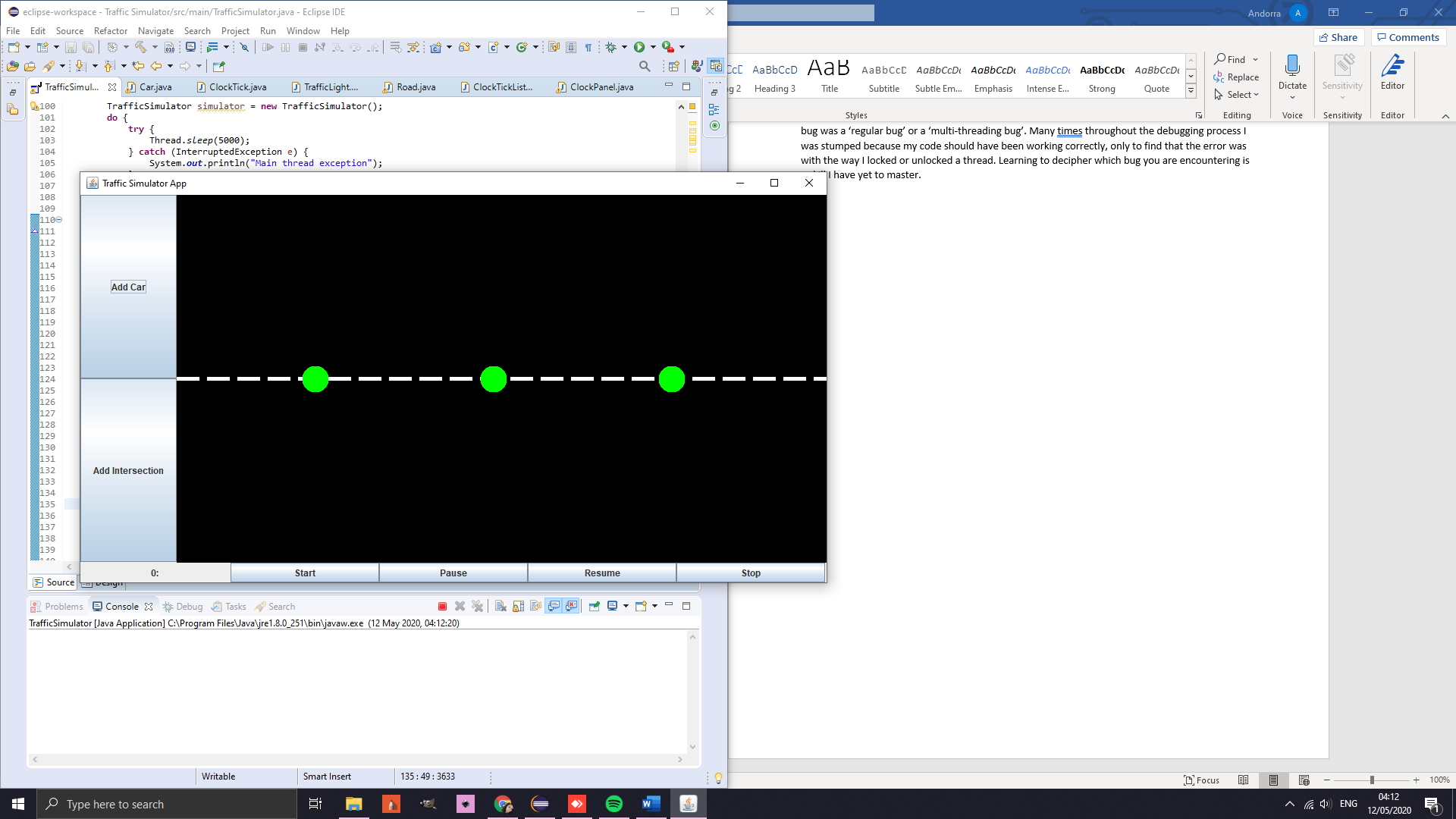
The main method is in the file “TrafficSimulator.java”.

To start the simulation, click the start button. To add a car or intersection, press the appropriate add button (add car or add light). If you add a light, you will have to adjust the window to see it. The cars will move across the screen changing position when the corresponding traffic light is green or yellow and stop when red. To pause the simulation, click the pause button, and to resume, the resume button. When you wish to end the program, press stop and exit the program.

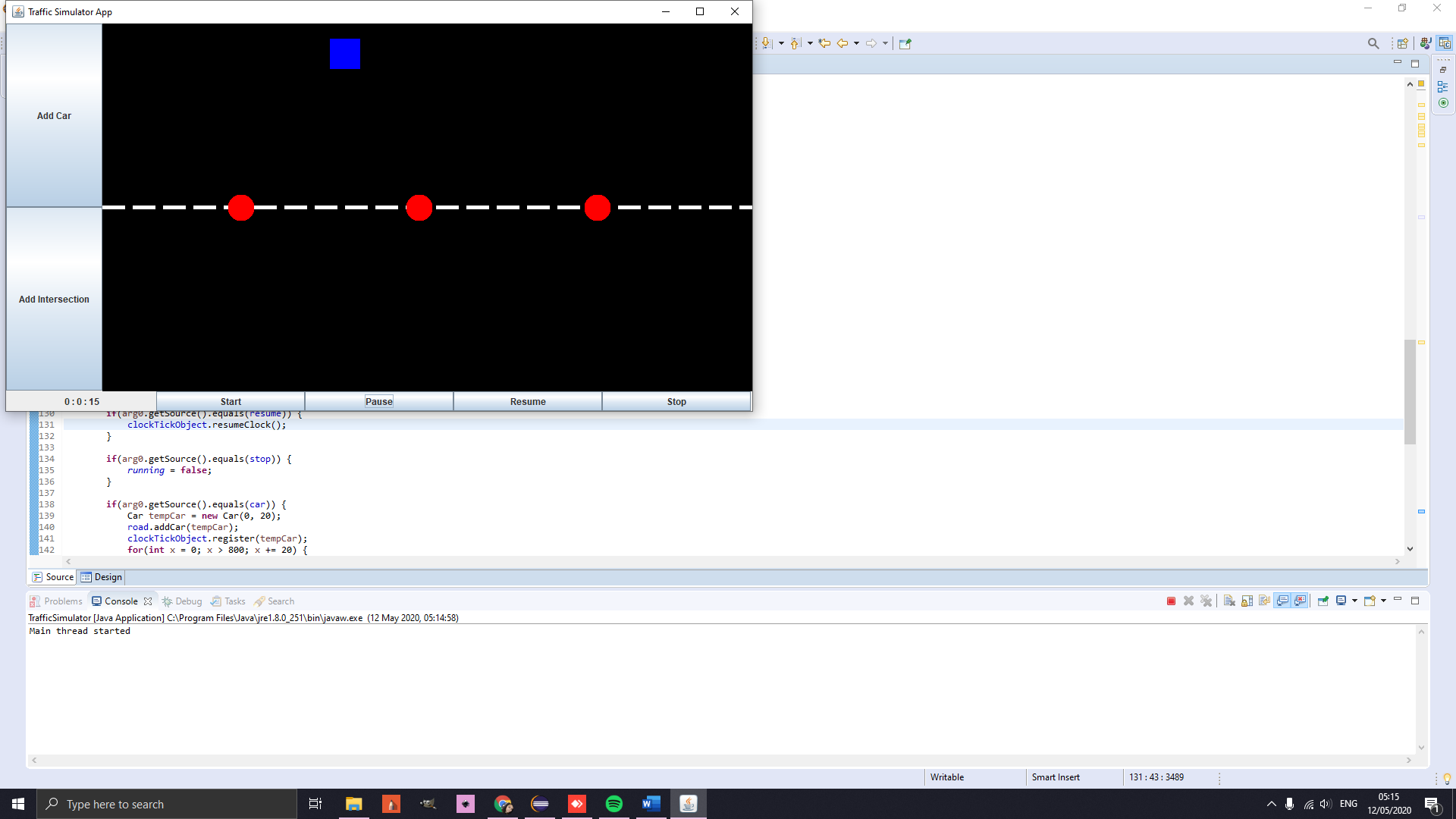
# Testing Plan and Screenshots

The steps of my testing plan

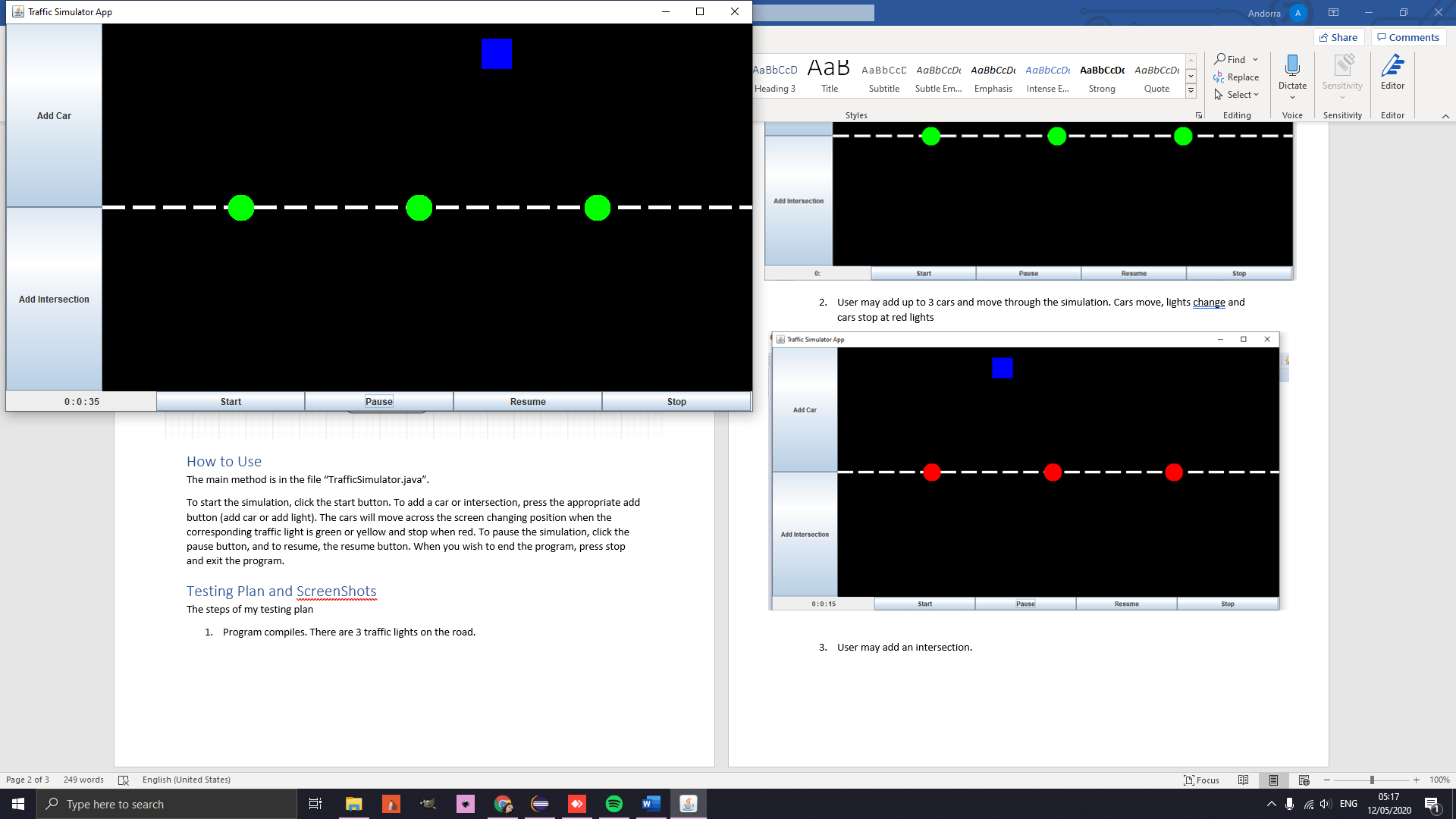
1. Program compiles. There are 3 traffic lights on the road.



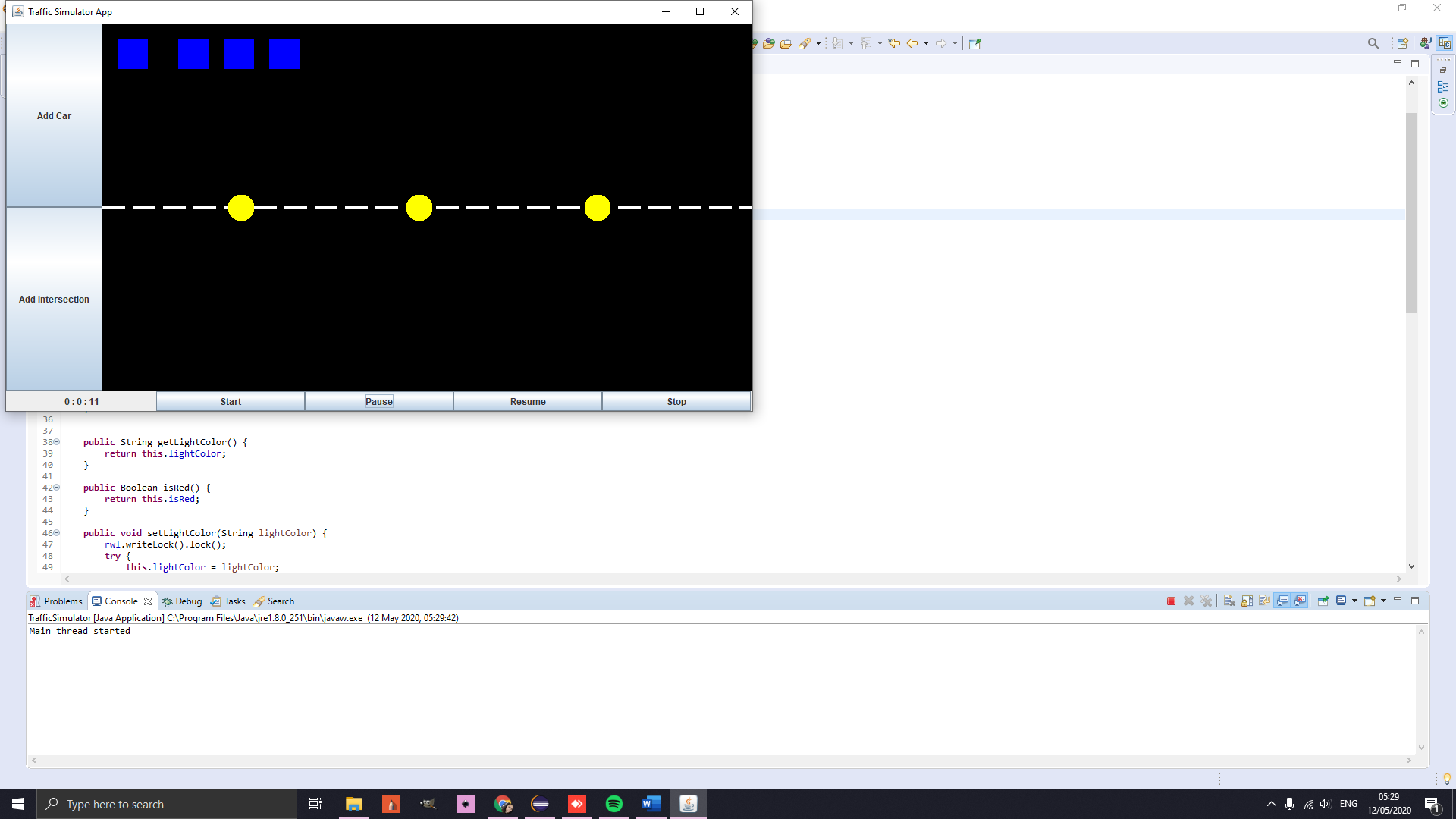
1. User may add up to 3 cars and move through the simulation. Cars move, lights change and cars stop at red lights

Stopping

Going

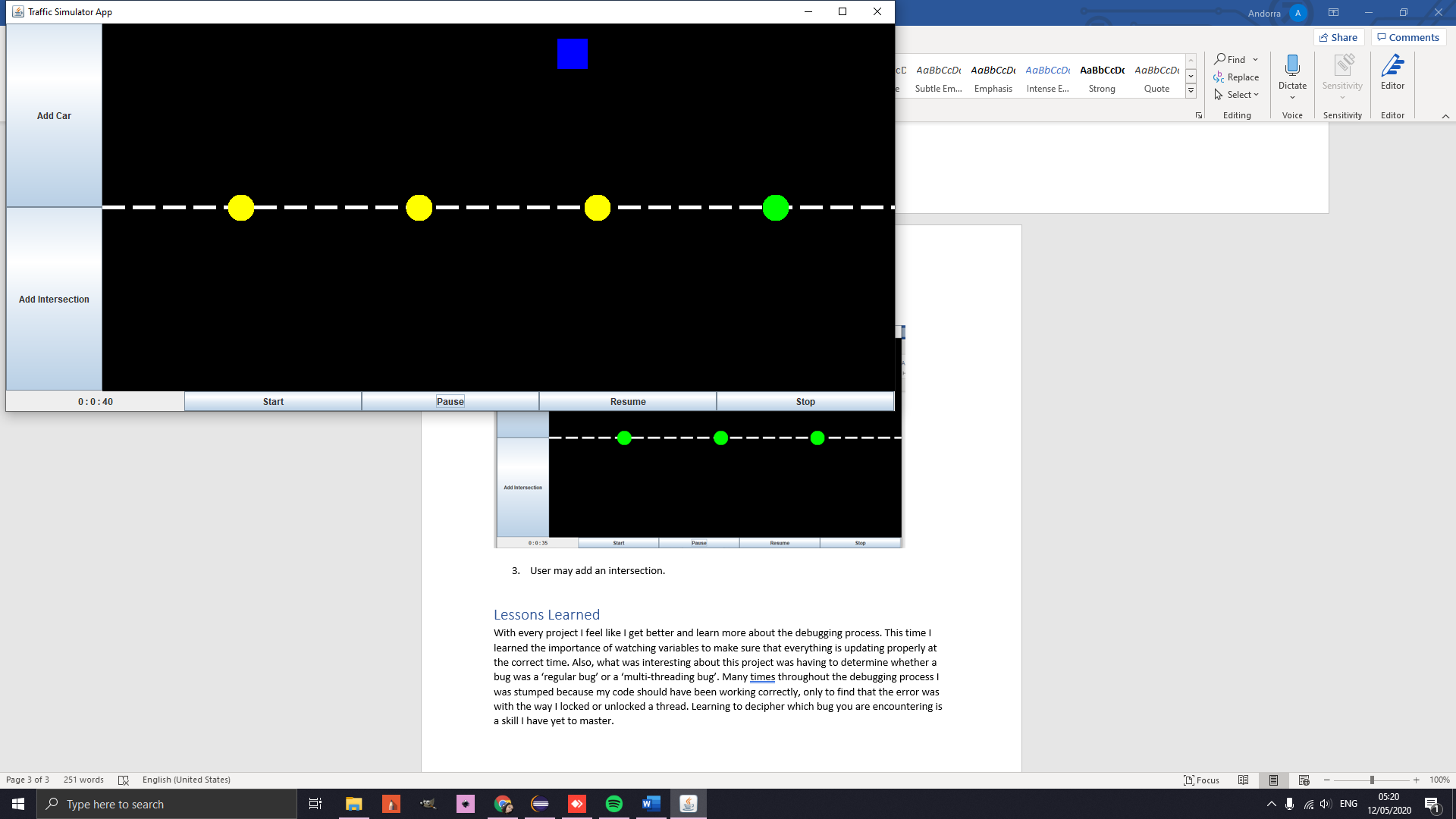


Clicking add car button 4 times



1. User may add an intersection.

(clicked the add intersection button)



# Lessons Learned

With every project I feel like I get better and learn more about the debugging process. This time I learned the importance of watching variables to make sure that everything is updating properly at the correct time. Also, what was interesting about this project was having to determine whether a bug was a ‘regular bug’ or a ‘multi-threading bug’. Many times throughout the debugging process I was stumped because my code should have been working correctly, only to find that the error was with the way I locked or unlocked a thread. Learning to decipher which bug you are encountering is a skill I have yet to master.