Lessons Learned

I spent a lot of time at the beginning of this project trying to figure out how to display the cars. I had never worked on anything GUI in Java that involved moving objects before. This led me to read a lot about how game engines keep a consistent frame rate and ended up With a SimpulationPanel object that would try to repaint itself at 60fps to save on CPU time. I applied the same kind of thinking to the threads which updated the car location to get something like a “game tick” going. I had hoped to be able to add some actual images of cars and traffic lights, but getting the thread logic to work around my “game loop” ended up being a lot more time consuming than I thought.

Build and Run the program

After extracting the project, you can install any necessary dependencies by running:

```
$ mvn install
```

In the project’s root. Then build the executable jar with:

```
$ mvn clean package
```

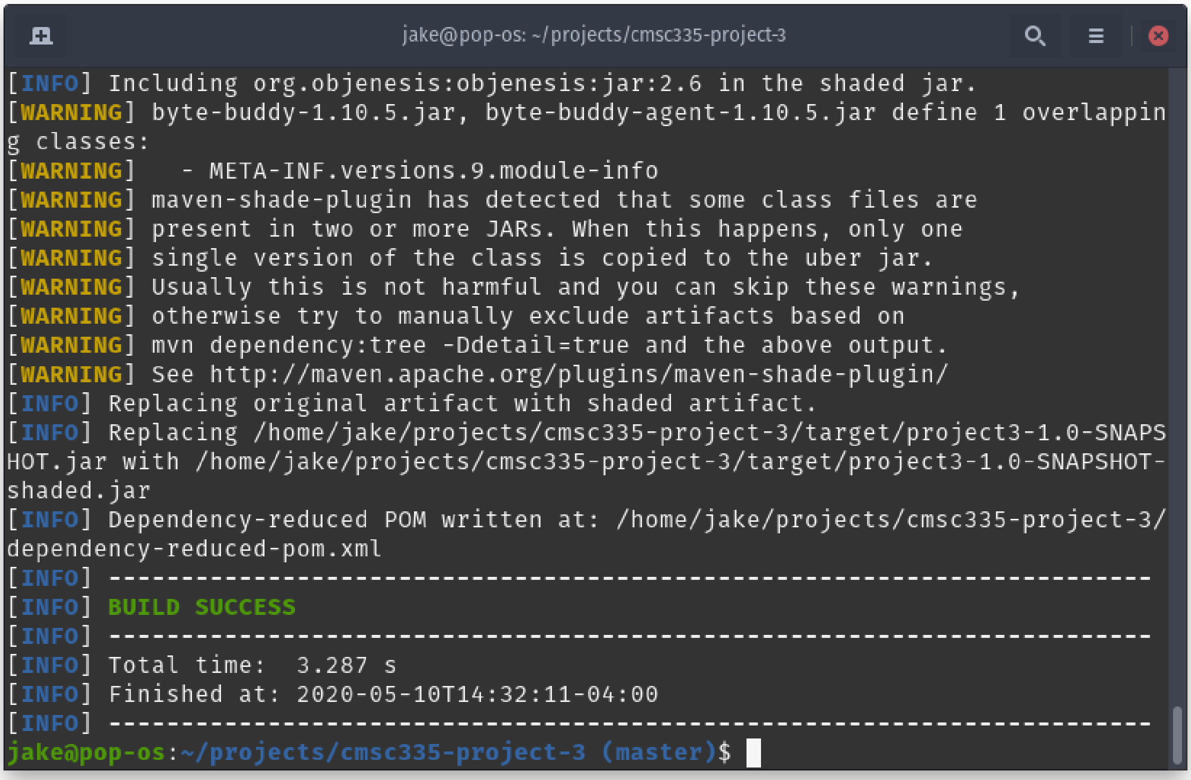
An executable will now be generated in the target/ directory, but you can also run the project by running:

```
$ mvn exec:java
```

Test Plan

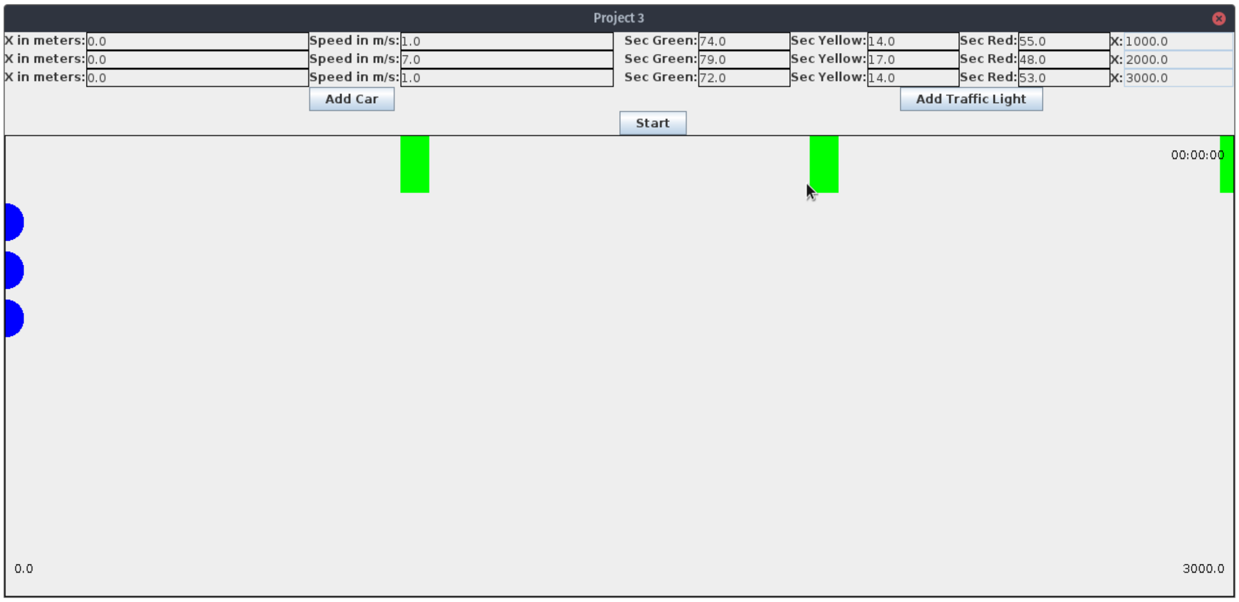
Program Builds and Runs Without Errors

I use maven to build the project and it completes without error:



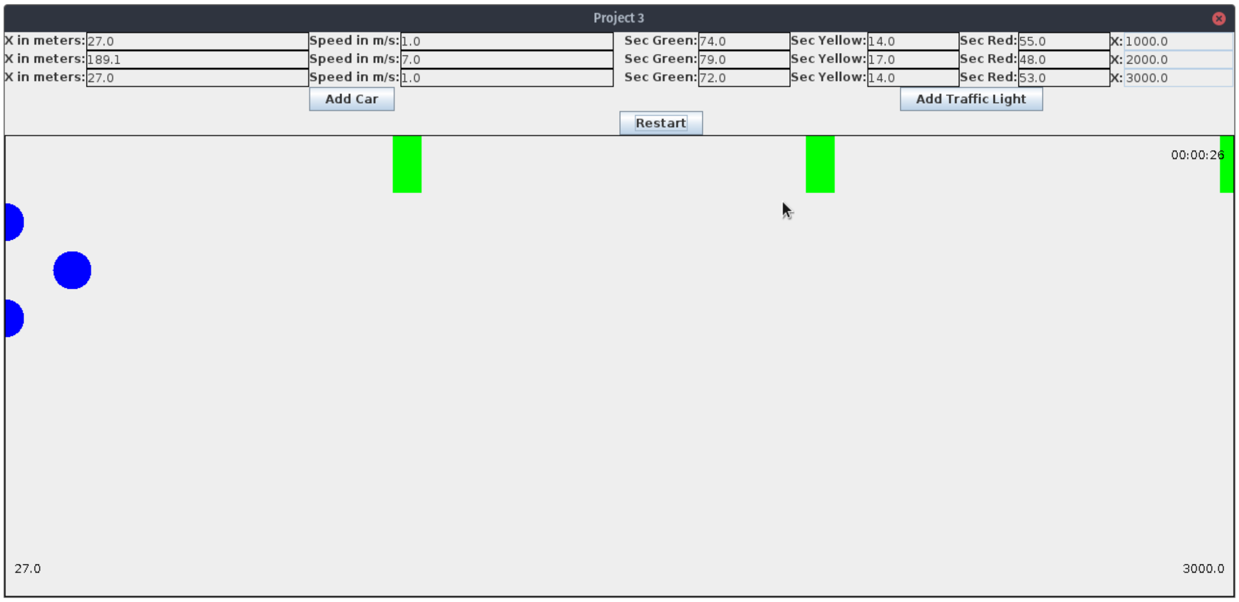
```
[INFO] Including org.objenesis:objenesis:jar:2.6 in the shaded jar.
[WARNING] byte-buddy-1.10.5.jar, byte-buddy-agent-1.10.5.jar define 1 overlappin
g classes:
[WARNING]   - META-INF.versions.9.module-info
[WARNING] maven-shade-plugin has detected that some class files are
[WARNING] present in two or more JARs. When this happens, only one
[WARNING] single version of the class is copied to the uber jar.
[WARNING] Usually this is not harmful and you can skip these warnings,
[WARNING] otherwise try to manually exclude artifacts based on
[WARNING] mvn dependency:tree -Ddetail=true and the above output.
[WARNING] See http://maven.apache.org/plugins/maven-shade-plugin/
[INFO] Replacing original artifact with shaded artifact.
[INFO] Replacing /home/jake/projects/cmsc335-project-3/target/project3-1.0-SNAPS
HOT.jar with /home/jake/projects/cmsc335-project-3/target/project3-1.0-SNAPSHOT-
shaded.jar
[INFO] Dependency-reduced POM written at: /home/jake/projects/cmsc335-project-3/
dependency-reduced-pom.xml
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  3.287 s
[INFO] Finished at: 2020-05-10T14:32:11-04:00
[INFO] ------------------------------------------------------------------------
jake@pop-os:~/projects/cmsc335-project-3 (master)$ 
```

When the program first launches, three cars will be created at x = 0 with random speeds ranging from 1 m/s to 10 m/s and three lights will be placed at 1000m, 2000m, and 3000m respectively:



X in meters: 0.0	Speed in m/s: 1.0	Sec Green: 74.0	Sec Yellow: 14.0	Sec Red: 55.0	X: 1000.0
X in meters: 0.0	Speed in m/s: 7.0	Sec Green: 79.0	Sec Yellow: 17.0	Sec Red: 48.0	X: 2000.0
X in meters: 0.0	Speed in m/s: 1.0	Sec Green: 72.0	Sec Yellow: 14.0	Sec Red: 53.0	X: 3000.0

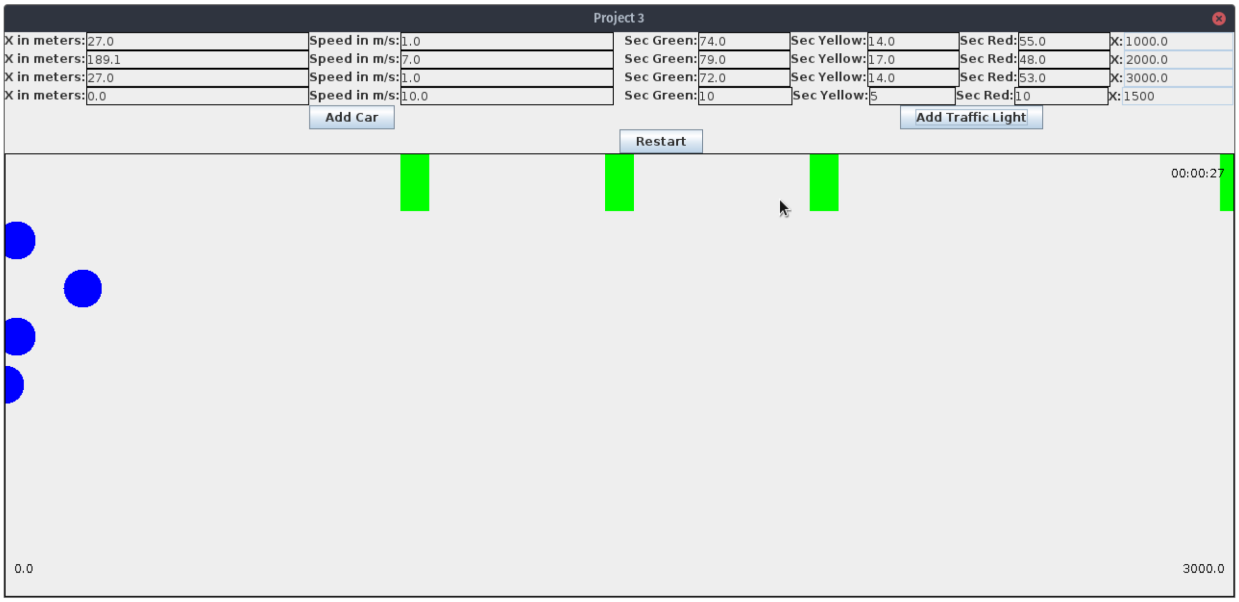
Once the user clicks start the cars will begin to move. The main drawing area live updates three strings: there is a timer in the upper right, while the bottom left indicates the leftmost axis' location in meters. The bottom right displays the rightmost axis location in meters. As cars and lights drift further or closer, the distance represented on the screen will change so that all entities are always visible:



Project 3

X in meters:	27.0	Speed in m/s:	1.0	Sec Green:	74.0	Sec Yellow:	14.0	Sec Red:	55.0	X:	1000.0
X in meters:	189.1	Speed in m/s:	7.0	Sec Green:	79.0	Sec Yellow:	17.0	Sec Red:	48.0	X:	2000.0
X in meters:	27.0	Speed in m/s:	1.0	Sec Green:	72.0	Sec Yellow:	14.0	Sec Red:	53.0	X:	3000.0

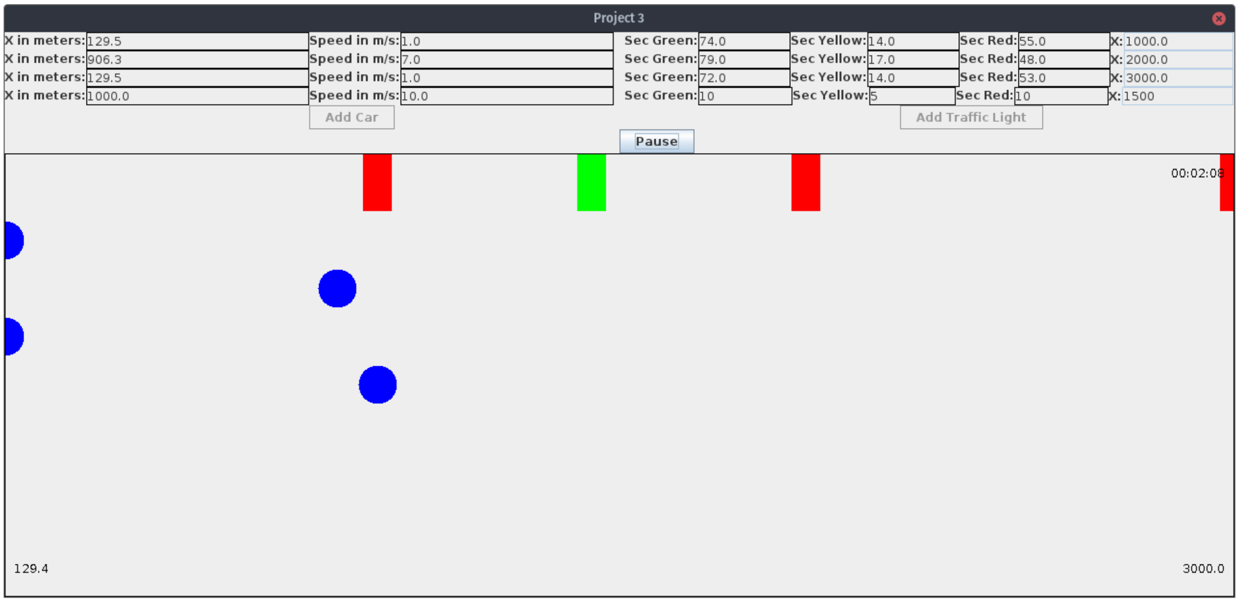
The user can pause the simulation at any time to add more cars or lights:



Project 3

X in meters:	27.0	Speed in m/s:	1.0	Sec Green:	74.0	Sec Yellow:	14.0	Sec Red:	55.0	X:	1000.0
X in meters:	189.1	Speed in m/s:	7.0	Sec Green:	79.0	Sec Yellow:	17.0	Sec Red:	48.0	X:	2000.0
X in meters:	27.0	Speed in m/s:	1.0	Sec Green:	72.0	Sec Yellow:	14.0	Sec Red:	53.0	X:	3000.0
X in meters:	0.0	Speed in m/s:	10.0	Sec Green:	10	Sec Yellow:	5	Sec Red:	10	X:	1500

When a car reaches a red light, it will stop until the light turns green. The location of the car and the light is measured by their center point:



Project 3

X in meters:	129.5	Speed in m/s:	1.0	Sec Green:	74.0	Sec Yellow:	14.0	Sec Red:	55.0	X:	1000.0
X in meters:	906.3	Speed in m/s:	7.0	Sec Green:	79.0	Sec Yellow:	17.0	Sec Red:	48.0	X:	2000.0
X in meters:	129.5	Speed in m/s:	1.0	Sec Green:	72.0	Sec Yellow:	14.0	Sec Red:	53.0	X:	3000.0
X in meters:	1000.0	Speed in m/s:	10.0	Sec Green:	10	Sec Yellow:	5	Sec Red:	10	X:	1500