**Meets Specifications**

Dear Learner,  
I must say this submission was indeed enjoyable to review. I appreciate and commend the efforts and hard work put into this piece. Congratulationsfor making it pass this stage of learning with us and I wish that this spirit is carried forward in subsequent projects. You should be proud of yourself because success is no accident. It is hard work, perseverance, learning, studying, sacrifice and most of all, love of what you are doing or learning to do. Please keep practicing on these projects and I wish you all the best.

Thanks for your great effort!

Nice work!!

**More In-depth Knowledge**

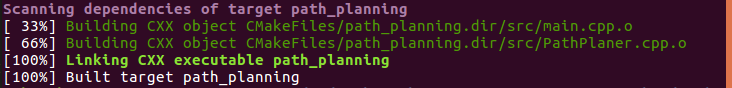
* [Path planning in environments of different complexity](https://www.mathworks.com/help/robotics/examples/path-planning-in-environments-of-different-complexity.html)
* [Robot Motion Planning](http://ais.informatik.uni-freiburg.de/teaching/ss11/robotics/slides/18-robot-motion-planning.pdf)
* [Robotics Path Planning](http://ais.informatik.uni-freiburg.de/teaching/ss10/robotics/slides/16-pathplanning.pdf)
* [Path Planning module in Robotics](http://www.coppeliarobotics.com/helpFiles/en/pathPlanningModule.htm)
* [Robo Realm Path Planning](http://www.roborealm.com/help/Path_Planning.php)
* [Difference between path planning and motion planning](https://robotics.stackexchange.com/questions/8302/what-is-the-difference-between-path-planning-and-motion-planning)
* [Excellent tutorial on a robot path planning](https://www.robotshop.com/letsmakerobots/excellent-tutorial-a-robot-path-planning)
* [Path Planning and Collision Avoidance](http://ais.informatik.uni-freiburg.de/teaching/ss10/robotics/slides/16-pathplanning.pdf)
* [Safe Motion Planning for Autonomous Driving](http://wesscholar.wesleyan.edu/cgi/viewcontent.cgi?article=1856&context=etd_hon_theses)
* [Local and Global Path Generation for Autonomous Vehicles Using Splines](http://www.scielo.org.co/pdf/inge/v21n2/v21n2a05.pdf)

**Compilation**

Code must compile without errors with cmake and make.

Given that we've made CMakeLists.txt as general as possible, it's recommend that you do not change it unless you can guarantee that your changes will still compile on any platform.

On compiling your project with cmake && make, there were no errors. Good job avoiding all syntax errors.

[](https://udacity-reviews-uploads.s3.us-west-2.amazonaws.com/_attachments/58227/1560433400/1.png)

**More In-depth Knowledge**

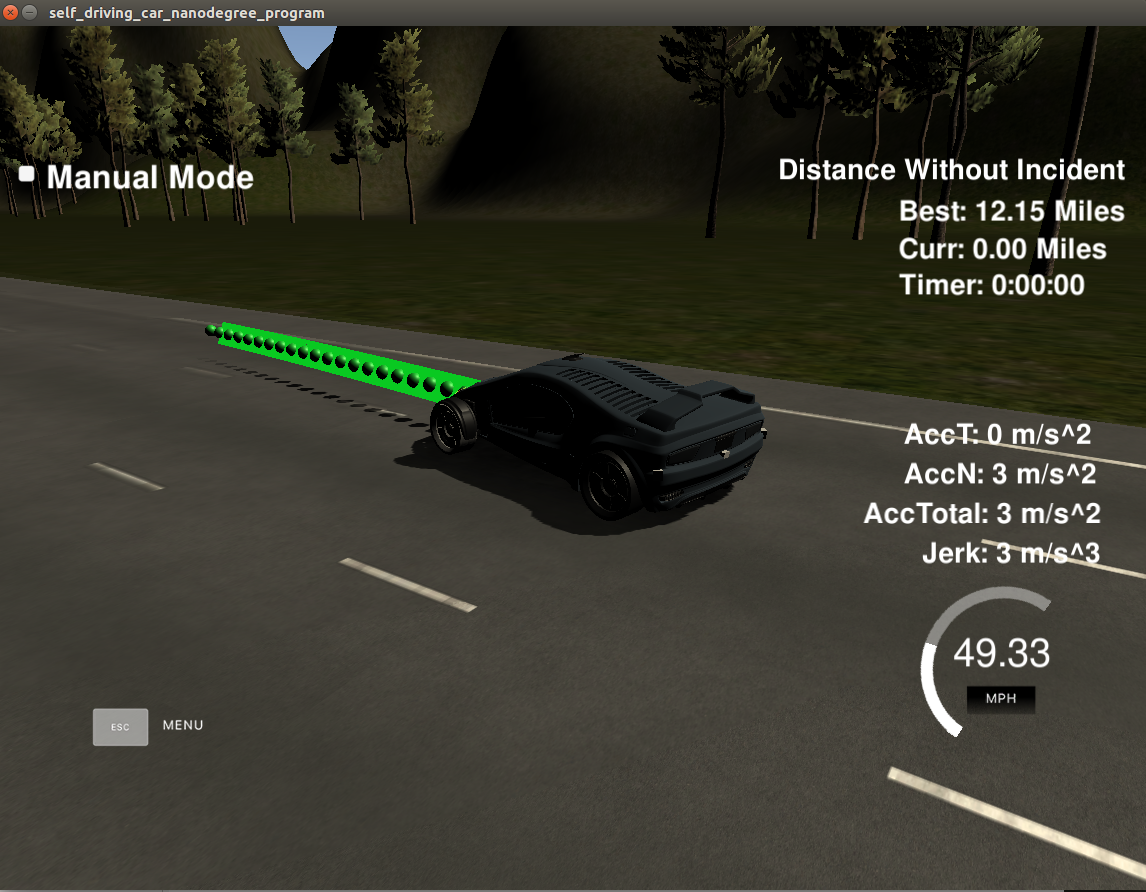
To know more about CMAKE and MAKE, please check some resources below:

* [CMake Script Debugger provided by the VisualGDB tool](https://visualgdb.com/)
* [Difference between Makefile and cmake to compile the code](https://stackoverflow.com/questions/25789644/difference-between-using-makefile-and-cmake-to-compile-the-code)
* [Cmake FAQS](https://cmake.org/Wiki/CMake_FAQ).
* [Using make and writing Makefiles](https://www.cs.swarthmore.edu/~newhall/unixhelp/howto_makefiles.html).
* [Youtube set of tutorials on using make and writing Makefile](https://www.youtube.com/watch?v=aw9wHbFTnAQ).
* [MakeFiles](https://www.cs.umd.edu/class/fall2002/cmsc214/Tutorial/makefile.html).

**Valid Trajectories**

The top right screen of the simulator shows the current/best miles driven without incident. Incidents include exceeding acceleration/jerk/speed, collision, and driving outside of the lanes. Each incident case is also listed below in more detail.

The current and best miles covered by the ego car without exceeding the set acceleration, jerk, and speed limits, and without collisions with other cars are displayed throughout the simulation. Also, the car was able to cover more than 12 miles without incidents as expected. Well done!

[](https://udacity-reviews-uploads.s3.us-west-2.amazonaws.com/_attachments/58227/1560433449/2.png)

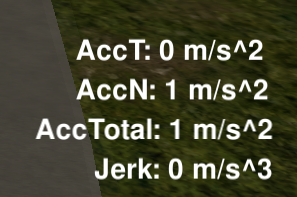
The car doesn't drive faster than the speed limit. Also the car isn't driving much slower than speed limit unless obstructed by traffic.

Nice work here! This speed of the car is wisely adjusted based on the traffic (how close the ego car is to a car that is in front). You did very well by setting the speed of the car such that it does not exceed the 50MPH speed limit. Excellent work!

[](https://udacity-reviews-uploads.s3.us-west-2.amazonaws.com/_attachments/58227/1560433547/4.png)

The car does not exceed a total acceleration of 10 m/s^2 and a jerk of 10 m/s^3.

Good job here! At no point during the simulation did the ego car exceed the 10m/s^2 acceleration and 10m/s^3 jerk limits.

[](https://udacity-reviews-uploads.s3.us-west-2.amazonaws.com/_attachments/58227/1560433597/3.png)

The car must not come into contact with any of the other cars on the road.

The car didn't get in contact with other cars during the simulation.

**More In-depth Knowledge**

Here are a few links with useful information on collision avoidance.

* [Path Planning for Collision Avoidance Maneuver](https://www.researchgate.net/publication/267596342_Path_Planning_for_Collision_Avoidance_Maneuver)
* [Optimal Trajectory Planning for Glass-Handing Robot Based on Execution Time Acceleration and Jerk](https://www.hindawi.com/journals/jr/2016/9329131/)
* [This discussion on StackExchange can be of interest Which trajectory planning algorithm for minimizing jerk.](https://robotics.stackexchange.com/questions/8555/which-trajectory-planning-algorithm-for-minimizing-jerk)

The car doesn't spend more than a 3 second length out side the lane lanes during changing lanes, and every other time the car stays inside one of the 3 lanes on the right hand side of the road.

Indeed, the car was always in one of the three lanes and doesn’t spend more than the recommended time (3 seconds) when crossing to a different lane. This is very well implemented!

The car is able to smoothly change lanes when it makes sense to do so, such as when behind a slower moving car and an adjacent lane is clear of other traffic.

Nice job here! The lane changing code allows the car to change lanes only when there is a free lane and the distance to cars in that lane is safe. As such, the car changes lanes smoothly without collisions or exceeding the jerk limit of 10ms^3. This was well thought out! Keep it up!

**Reflection**

The code model for generating paths is described in detail. This can be part of the README or a separate doc labeled "Model Documentation".

Great Job!!