Peer review:

Design: 3	The classes of project are well structured. Use of external libraries is done well and the code is easy to understand. The SFML library is directly attached to the rest of the code which might make extending the project in the future harder. The simulation and visualization are not decoupled. The project utilizes inheritance.
Working practices: 5	Extensive use of feature branches. Worklog is well maintained and covers everyone's main responsibilities well. Testing is documented but no real unit tests were written for the project and all testing was done manually. This is partly due to the design of the project.
Implementation: 4	CMake works without problems. One of our team could not build because of required CMake version. The project also utilizes smart pointers to some extent. Memory management is not documented. Valgrind shows some memory leakage.
Features: 4	Basic features all work. GUI, building and passenger profiles are implemented and they work. So ⅗ additional features are implemented to some extent. Lanes don’t have limits for the amount of cars which is listed as a minimal requirement.

Teacher:

Design: 4	**Not many advanced C++ constructs are applied but some inheritance and algorithms are used. Standard C++ libraries and external libraries are used appropriately. Good class structure. There could be more documentation about the design choices and class relationships.**
Working practices: 4	**Good use of Git features (branches used extensively). Git commits are frequent and commit messages are informative enough. Some variation in work hours between team members but work was mostly distributed evenly. Division of work is well tracked and documented. There is some documentation about testing. Testing mostly done manually, no automated unit tests. Regular communication between group and advisor, meetings were mostly attended every week.**
Implementation: 3.5	**Building the software is easy, cmake works well. Project documentation has good instructions for building and using the software. Software works robustly and handles unexceptional user behavior. Smart pointers are used but there is some issues with memory management as some memory leakage originates from traffic-simulator-main.cpp line 89 where memory for a new Vehicle is allocated. No documentation about memory management.**
Features: 4	**Basic features mostly implemented, but cars can move on top of each other, roads don't seem to limit the number of cars that can be on the same road at the same time. Some additional features implemented: simple GUI, passenger profiles, building profiles, analysis tools can average over multiple days.**