FP1: Please look in camFusion_Student.cpp in function matchBoundingBoxes() from line 236 where, I have been implemented matching boxes through getting all the boxes in previous and current frame at the same matched key point and then sum every box from the previous with the current in nested loop to check the freq of every box from previous with current and then once we finish the matching, we get the highest freq between previous and current .

FP2: please look in camFusion_Student.cpp in function 199-232 computeTTCLidar() where, I calculated the time of collision using the previous lidar data from previous frame and current frame, every lidar frame we filtered the lidar data with specific range in y where we set scanned desired area for car in x unlimited, y(2) with any point in this range for previous or current lidar frame will be saved in vector. I used to make sort and take the min captured x in prev and curr not mean or median for safety issue since the minimum will give the earliest estimated time of collision. Then I used the data I get it and I put in the formal.

FP3: please look in camFusion_Student.cpp in function 132-163 clusterKptMatchesWithROI() where, I filtered the matches key points in order to keep the more accurate key points, I followed the way of checking which matches points fall in ROI and then I got the distance of every matches point with the prevframe with the same match point through using distance attribute and once I got the distance of all matches points, I got the mean and with 0.7of the mean I make it as standard for filtering the best fit matches points

FP4: please look in camFusion_Student.cpp in function 168-196 computeTTCCamera(), where I applied the methodology that I learned from the previous lectures. The way was following getting matched key in previous frame and start get the distance between this matched key point and others neighbor matched keypoint, the same thing for the current frame with the same matched keypoint and the same with neighbor keypoint the process is repeating form the original matched keypoint to matches in ROI in both frames in every time we get the difference distance of two matched key points in prev and curr and we divide it to get the Ratio the process is repeating and we save the distraction in vector, then we median of this vector and use the formal of TTC using Camera

FP5: When the car moves constantly and the distance between the ego and front car with range of 7- 8 meters, the difference in the previous and current lidar time is approx. 1 second and close to be considered, but when the car deaccelerating with the decreased distance between ego and front car the time seems that is not make sense, especially the time must be less than the detected when car was moving constantly and distance was more!. this was happened in frame 3 which the time detected was 16.384 and frame 4 which the time was 14.076 the difference is higher than average as well as time must be lesser. The error is because the formal which is used in the TTC which consider only the vehicle in constant velocity. For other lidar TTC it's close the be proper with TTC camera as well as the time can be considered.
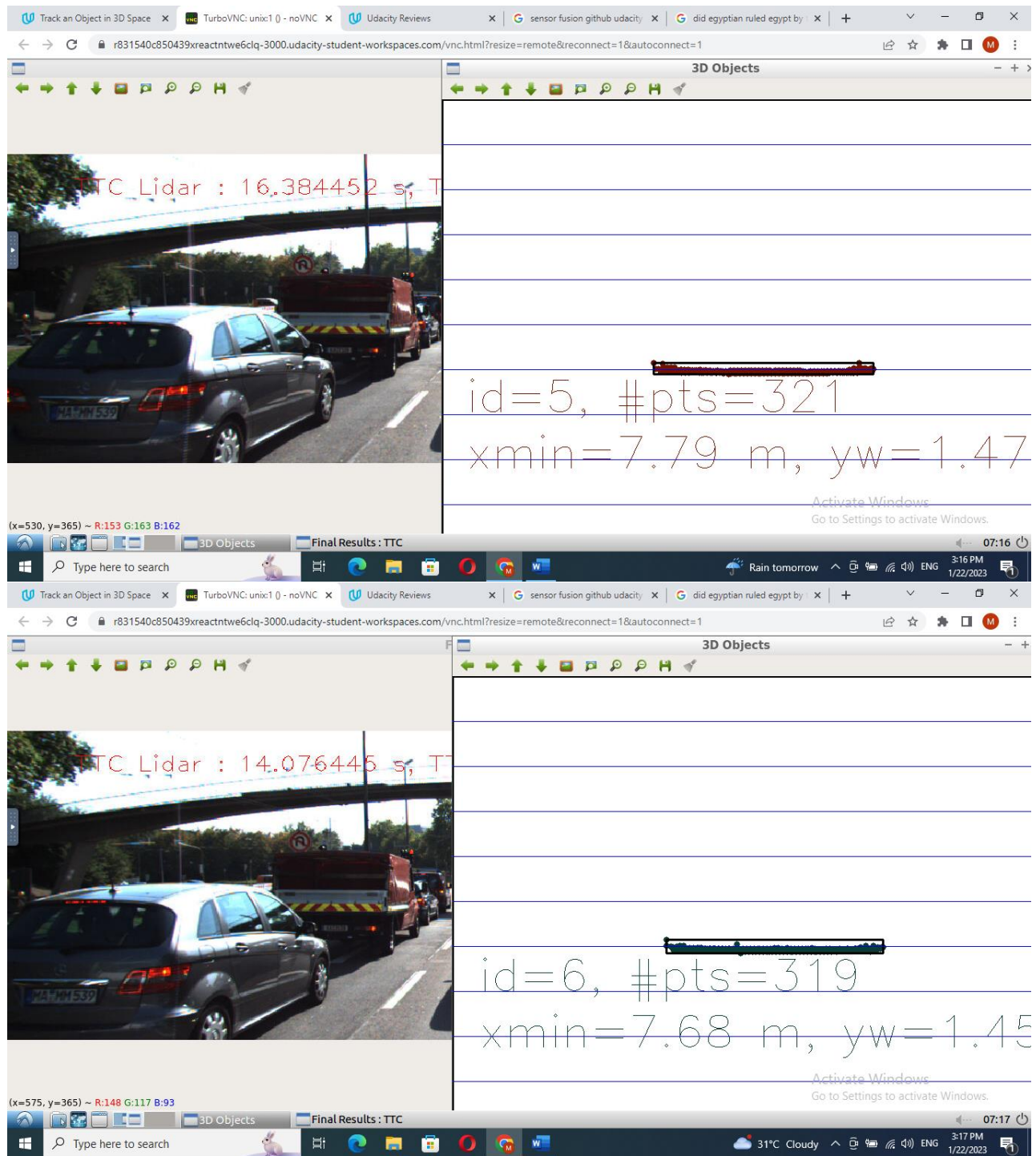
FP6: Please take Look on README.xlsx it contains the full explanation under Evaluation 2.