

 ${\color{blue} \textbf{Dashboard Calendar Progress Projects Activities More}}$





← Project review - ROS. Team01

Type of project	Group
Ouration	30 min
Passed Peer Reviews	0/2

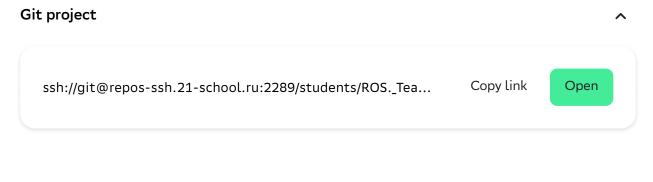
Git project

Team 1 (TL: aarchiba-python-ds)

About

Main part

Feedback





About

Introduction

The methodology of School 21 makes sense only if peer-to-peer reviews are done seriou sly. Please read all guidelines carefully before starting the review.

- Please, stay courteous, polite, respectful and constructive in all communications during this review.
- Highlight possible malfunctions of the work done by the person and take the time to discuss and debate it.
- Keep in mind that sometimes there can be differences in interpretation of the tasks a nd the scope of features. Please, stay open-minded to the vision of the other.
- If you have not finished the project yet, it is compulsory to read the entire instruction before starting the review.

Guidelines

- Evaluate only the files that are in src folder on the GIT repository of the student or group.
- Ensure to start reviewing a group project only when the team is present in full.
- Use special flags in the checklist to report, for example, an "empty work" if repositor y does not contain the work of the student (or group) in the src folder of the develop br

anch, or "cheat" in case of cheating or if the student (or group) are unable to explain their work at any time during review as well as if one of the points below is not met. How ever, except for cheating cases, you are encouraged to continue reviewing the project to identify the problems that caused the situation in order to avoid them at the next review.

- Doublecheck that the GIT repository is the one corresponding to the student or the group.
- Meticulously check that nothing malicious has been used to mislead you.
- In controversial cases, remember that the checklist determines only the general order of the check. The final decision on project evaluation remains with the reviewer.

Main part

Mandatory part

- You need to run the ROS metapackage from the contributor repository in the Gazebo AutoRace scene from the official repository https://github.com/ROBOTIS-GIT/turtlebot3_simulations branch melodic-devel. All participants are tested on one Gazebo AutoRace stage.

The result of your check of the performance of the race should be a table of scores for missions, total scores, completion time.

Assessment of the assignment is carried out according to the rules set forth in the doc ument rush01.

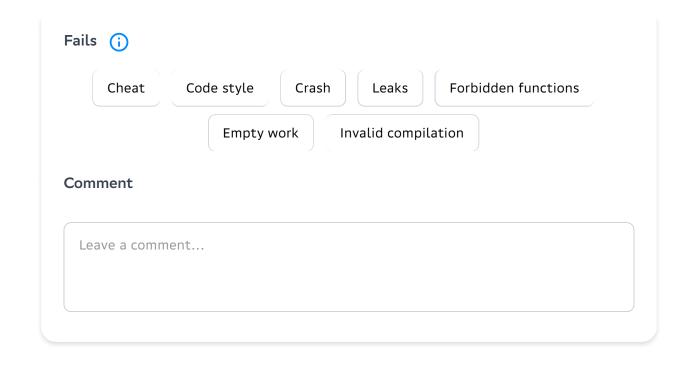


Report

- Does the repository contain a presentation?
- Does the presentation have flowcharts?
- Is it clear from the presentation how the algorithm works?
- Are there comprehensive comments in the source code of the ROS package?



Feedback



✓ Review