

Testing Process Document

for *Earth*

Alex Egan

Abstract

This document explains when testing should be performed, what testing should be done, who is responsible for it and who is responsible for any issues.

Contents

1	Process	1
1.1	Introduction	1
1.2	Local Level Testing	1
1.2.1	Summary	1
1.3	Subgroup Level Testing	1
1.3.1	Summary	2
1.4	Root Level Testing	2
1.4.1	Summary	2

1 Process

1.1 Introduction

Testing needs to be performed at 3 points in the process of working on Earth. These are:

1. Local level
2. Subgroup level
3. Root level

Local level testing refers to the testing performed by an individual group member while modifying or adding code to the project.

Subgroup level testing refers to the testing performed when accepting a group member's code into the subgroup repository.

Root level testing refers to the testing performed when accepting a subgroup's code into the root repository.

For clarification on the repository structure, see the Repository Process Document.

1.2 Local Level Testing

When a group member has modified or added to the Earth project, they should perform testing so that their changes or new functionality work as intended. This can be done by hand as sanity checks while work is being done. When the piece of work is complete, test cases should be written if they are required. To determine whether new test cases are required, the rcov tool can be used to check if the new code is covered by existing test cases. Should new test cases be required, these should be added based on the sanity checks used previously and thorough testing such as bounds conditions checks and specific cases that could cause issues.

Once these test cases are complete and the new code passes them and all the other test cases that were already implemented, the latest subgroup changes should be merged with the local repository. This can be done in a temporary branch. The new test cases should be run against this combined codebase.

Once this local level of testing has been performed and is satisfied, code can be marked to be escalated up the repository hierarchy into the subgroup repository.

1.2.1 Summary

Below is a summary of how a successful addition of code would be gone about. If the test cases fail or do not cover the new code, they must be revised by the group member who wrote the changes and test cases. The process must then be repeated.

1. Work on code
2. Test code while working
3. Run test cases
4. Check test case code coverage
5. Write test cases
6. Run test cases
7. Check test case code coverage
8. Mark code for escalation

1.3 Subgroup Level Testing

Once a group member is satisfied their code is ready to be merged with the subgroup repository, the git leader of the subgroup should be notified as per the Repository Process. The git leader should merge the new code with the subgroup repository and check that all test cases

are successful and that there is sufficient code coverage. This can be done in a temporary branch.

If the testing is unsuccessful, the git leader may attempt to diagnose what the issue is. Due to the nature of this Process, any issues should be caused by the new code being introduced. The git leader should then inform the group member that their code is not working sufficiently and give them any information that they gleaned as to what the exact issue is. The group member must then revise it.

If the testing is successful, the code from the group member can be pushed into the subgroup repository and the other subgroup members can pull this code as per the Repository Process.

1.3.1 Summary

Below is a summary of how a successful addition of code would be gone about. If the test cases fail or do not cover the new code, they must be revised by the group member who wrote the changes and test cases. The process must then be repeated.

1. Code from subgroup member is marked for escalation
2. Merge code with current subgroup repository code
3. Run test cases
4. Check test case code coverage
5. Accept code into subgroup repository
6. Mark code for escalation

1.4 Root Level Testing

Once a subgroup has a new code they deem to be working correctly, this status can be indicated to the git leader, or leaders, currently in charge of the main, root, group repository as per the Repository Process. The git leaders then merge the subgroup's code with the root code and run the test cases and check the code coverage. This can be done in a temporary branch.

If the testing is unsuccessful, the git leader may attempt to diagnose what the issue is. Due to the nature of this Process, any issues should be caused by the new code being introduced. The git leader should then inform the subgroup that their code is not working sufficiently and give them any information that they gleaned as to what the exact issue is. The subgroup must then revise it.

If the testing is successful, the code from the subgroup can be pushed into the root repository and the other subgroups can pull this code as per the Repository Process.

1.4.1 Summary

Below is a summary of how a successful addition of code would be gone about. If the test cases fail or do not cover the new code, they must be revised by the subgroup who wrote the changes and test cases. The process must then be repeated.

1. Code from subgroup is marked for escalation
2. Merge code with current root repository code
3. Run test cases
4. Check test case code coverage
5. Accept code into root repository