

Tips on writing maintainable Test Cases

Don't just directly jump on to write test cases first. First try to make a proper plan on how you are going to organise or sub-divide the functions of your Test Application into test suites.

The following rules are written in context with vREST. Few thumb rules for making maintainable test suites should be

- Divide your Test Application into meaningful Test suites.
- Each test suite should be independent from other test suite.
- Each test suite should not have more than 50 test cases or at max, you can stretch it to 100 test cases. If the test suite is having more than 100 test cases then try to divide further into multiple test suites.
- Provide meaningful summary for each test case and also try to provide description for each test case to describe what a particular test case is validating.
- Use variables for hard coded values like base URL, port numbers or other environment specific things.
- Try to reduce dependency between test cases as low as possible.
 - Suppose you are validating the pagination functionality, and you want to validate whether API is returning correct records for page 2. Then do not create so many test cases to pre-fill the data, instead define a separate test case to restore the data before that test case.
- Automated test suite should be runnable from any machine where the test application can be installed. This exercise should resolve all environment related issues and hard coded values.
- ...