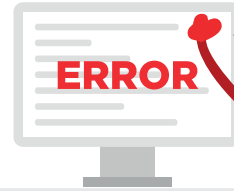




Testability



By Rob Meaney and Ash Winter

- 1 Define what you believe testability is. Share your definitions on [The Club](#).
- 2 Perform some testing on your application, then open your applications log files. Can you find the actions you performed in the logs?
- 3 Begin reading a book related to testability and share your learnings by Day 30.
- 4 Do you know what your top three customer impacting issues are? How could you find out?
- 5 What monitoring system is used for your application? Do the alerts it has configured reflect what you test?
- 6 Explore the output of your applications analytics, how can they help guide your testing?
- 7 Find out if your application has an operational manual or runbook? How can you test this?
- 8 Share a video about testability with your testing peers at your company.
- 9 In your next story kick off session ask, 'how are we going to test this?' Share the test ideas and techniques that are suggested.
- 10 Follow and share three people on LinkedIn/ Twitter who regularly talk about testability.
- 11 Get access to your source control and find the active branches for your application. Has anything changed that you didn't consider?
- 12 Decomposability is an important part of testability. Complete our circuit breakers testing exercise over on [The Club](#)!
- 13 Find out how test data is populated in your system. How could it be improved? You can watch [Techniques for Generating and Managing Test Data](#) by Omosé Ogala for some ideas to get you started.
- 14 Watch the [Testability Ask Me Anything](#) on the Dojo. Post any additional questions on [The Club](#).
- 15 Share a blog post that you found interesting related to testability. Don't forget to use the hashtag #30DaysOfTesting if you share on Twitter.
- 16 Conduct a show and tell of the latest features in your application for staff outside your immediate team. Capture their feedback and share with your team.

- 17 Pair with a developer to see if they can improve something that you find difficult or time consuming to test. We have a handy guide from Lisa Crispin on [Pairing with Developers: A Guide for Testers](#) that's worth a read.
 - 18 Note the top 3 challenges you have while testing and raise them at the team retrospective.
 - 19 Your dependencies can constrain your testability. Head over to [The Club](#) to visualise your applications boundaries.
 - 20 Think about what's currently stopping you from achieving higher testability. Share your findings on [The Club](#).
 - 21 Unit tests provide insight into an applications testability. Pair with a developer to explore some unit tests.
 - 22 How long does it take to set up a new test environment and start testing. Could this be faster?
 - 23 Share an article about application log levels and how they can be applied.
 - 24 What could you learn about your application's testability from being on call for support? This [eBook](#) could help you get the most out of taking support calls.
 - 25 Ask your team if there are any areas of the system they fear to change. How could you mitigate that fear?
 - 26 Relationships with other teams affect testability. Share your experiences on [The Club](#).
 - 27 Use source control history to find out which parts of your system change most often. Compare with your regression test coverage.
 - 28 Pair with an internal user or customer support person and explore your application together. Share your findings on [The Club](#).
 - 29 Do you know which components of your application respond the slowest? How could you find out?
 - 30 What book did you choose on Day 3 and what did you learn from it?
- Bonus:** Arrange a meeting with your team to take the team testability test over at <http://testabilityquestions.com/>

