

Testing and Quality Assurance

What is the importance of testing and quality assurance?

Quality assurance (QA) focuses on measuring and examining the quality of design, development, enhancements and maintenance of processes and products to make sure they conform to requirements, standards, and procedures set forth in a project. Testing, on the other hand, may keep an eye on the processes and often owns them, but is far more concerned looking at ways to break systems. Testers are to observe what systems do and to report on the level of quality as well as any serious issues they encounter.

How to incorporate testing and QA into your software project?

When trying to bring QA into your work you can follow these five phases;

Assessment - what needs to be tested and documented

Testing assessment should consider the following:

1. New project - One of the keys for a new project is to define how software should function and how you will test it in the requirements and design phases of the project. Also consider the impact that this system or module might have on business processes, and existing software and processes. Make sure that you plan to test business processes and ensure that they work with the new software. Ensure that there are plans to test documentation, business processes, help processes and all aspects of the new system or module.
2. Enhancement or change - Someone who knows the system and business process well, will need to do an assessment to determine what documentation, business processes and computer functions will be impacted by the enhancement or change.

Planning

In the Planning phase, the team defines specific system requirements and develops strategies around the information architecture (static content and information flows) and the business functions that will be addressed.

Testing planning should include:

1. Development of a test strategy that covers the following:
 - a. Levels of testing to include (Unit testing by developers, system testing by testing professionals, customer testing; ensuring that real users can perform their job functions and the system produces the correct results, load testing / capacity testing to determine what types of transaction loads the system can handle).
 - b. Who will perform each level of testing
 - c. How will testing be documented
 - d. How will test failures be handled
 - e. Will any testing levels be automated, if so, what tools will be used
2. Test cases to include - Plan what test cases need to be executed. These should be based upon requirements and should include testing boundary and error situations.
3. Environments for executing tests. A team should have one or more dedicated environments that can be used to execute test cases.

Design / Development

During the Design phase, the team identifies all of the necessary system components based on the requirements identified during the Assessment and Planning phases. The team then creates detailed design specifications for each component and for the associated physical data requirements. During the Development phase, the team constructs the components specified during the Design Phase.

Testing design and development should include:

1. Development of test cases and testing processes
2. Initial informal run through of test cases as the system is being developed

Testing

Execute the testing plan that has been developed. Ensure that there is a process for recording testing failures and re-testing.

Implementation

In the Implementation phase, the team focuses on testing and review of all aspects of the system. The team will also develop system documentation and a training or market test plan in preparation for system launch.

Resources and links

http://www.etestinghub.com/what_is_softwaretesting.php

Gherkin is also a "language" or a formatted process that is being used to define features or test cases. The DHIS2 team is capturing their feature files in GitHub. [Example feature file](#)