

Use-Case Essentials

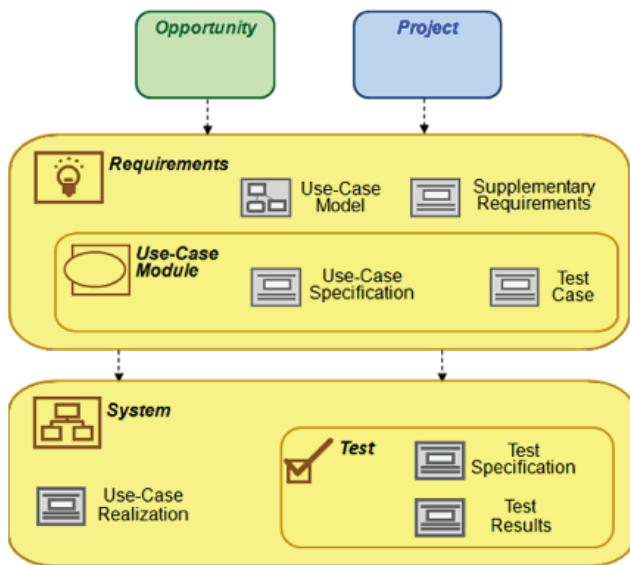
An agile, scalable approach to requirements management, development and system testing

Use this practice to capture requirements in an accessible form that can be used to drive the development of the software.

This practice allows teams to:

- Work with customers to capture the truly essential requirements.
- Work together more effectively to quickly develop a useable solution.
- Identify and deliver the value expected from the system.
- Establish the correct level of requirements detail to support their needs and the needs of their customers.
- Prioritize and sub-set requirements to identify a minimal solution and drive iterative development.
- Use a systematic approach to ensure the correct design, implementation and verification of requirements.

Things to Produce



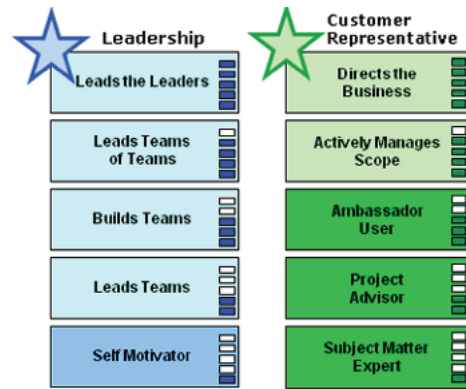
This practice involves the production of a number of requirements, design and test artifacts:

- A use-case based specification of the requirements, scenarios and test cases.
- The realization of the use cases to drive the development of the software.
- The generation of tests and test results to test the resulting system and record the results of the testing.

Key Competencies

This practice requires the team to be skilled in software requirements capture, design, coding, integration and testing.

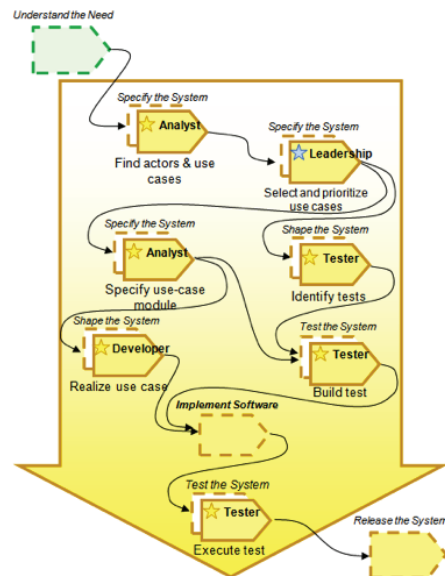
Analyst	Developer	Tester
Applies Complementary Techniques	Shapes Enterprise Systems	Leads Large-Scale Testing Efforts
Adapts Techniques	Shapes Systems	Plans Testing
Facilitates Discussions	Designs Interfaces & Interactions	Leads Testing
Builds Models	Designs Internals of Elements	Specifies Test Cases
Clearly Describes What's Needed	Writes Good Code	Executes Tests



The most important of these are the skills of the customer representative and the analyst as without these the wrong thing may well be developed and tested.

Things to do

The practice starts by finding actors and use cases, and selecting and prioritizing the use cases to be developed.



It continues by specifying the use cases and their tests, and then implementing software to meet the tests. It concludes by executing the tests and tracking progress in terms of verified, working software.