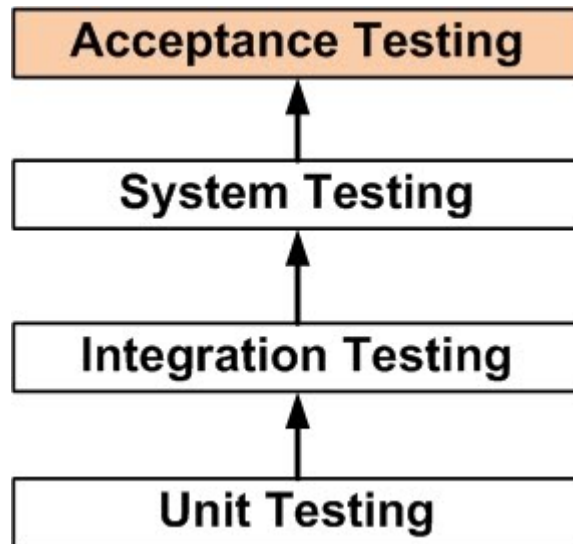




Acceptance Testing

ACCEPTANCE TESTING is a level of software testing where a system is tested for acceptability. The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery.



Definition by ISTQB

- **acceptance testing:** Formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system.

Analogy

During the process of manufacturing a ballpoint pen, the cap, the body, the tail and clip, the ink cartridge and the ballpoint are produced separately and unit tested separately. When two or more units are ready, they are assembled and Integration Testing is performed. When the complete pen is integrated, System Testing is performed. Once System Testing is complete, Acceptance Testing is performed so as to confirm that the ballpoint pen is ready to be made available to the end-users.

Method

Usually, Black Box Testing (<http://softwaretestingfundamentals.com/black-box-testing/>) method is used in Acceptance Testing. Testing does not normally follow a strict procedure and is not scripted but is rather ad-hoc (<http://softwaretestingfundamentals.com/ad-hoc-testing/>).

Tasks

- Acceptance Test Plan
 - Prepare
 - Review
 - Rework
 - Baseline
- Acceptance Test Cases/Checklist
 - Prepare
 - Review
 - Rework
 - Baseline
- Acceptance Test
 - Perform

^

When is it performed?

Acceptance Testing is the fourth and last level of software testing

(<http://softwaretestingfundamentals.com/software-testing-levels/>) performed after System Testing

(<http://softwaretestingfundamentals.com/system-testing/>) and before making the system available for actual use.

Who performs it?

- *Internal Acceptance Testing* (Also known as Alpha Testing) is performed by members of the organization that developed the software but who are not directly involved in the project (Development or Testing). Usually, it is the members of Product Management, Sales and/or Customer Support.
- *External Acceptance Testing* is performed by people who are not employees of the organization that developed the software.
 - *Customer Acceptance Testing* is performed by the customers of the organization that developed the software. They are the ones who asked the organization to develop the software. [This is in the case of the software not being owned by the organization that developed it.]
 - *User Acceptance Testing* (Also known as Beta Testing) is performed by the end users of the software. They can be the customers themselves or the customers' customers.

Recommend 21

Share

G+ Share

Tweet

StumbleUpon

32
Shares

◀ Software Testing Exercises (<http://softwaretestingfundamentals.com/software-testing-exercises/>)

System Testing ▶ (<http://softwaretestingfundamentals.com/system-testing/>)

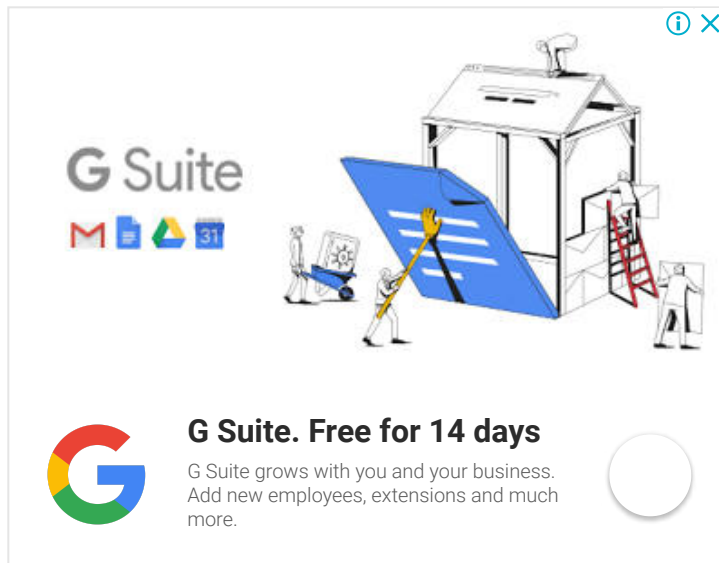


Adjustable Scrum/Kanban Board

with Backlog, Sprint Planner, Agile charts and more.



Search...



G Suite. Free for 14 days

G Suite grows with you and your business. Add new employees, extensions and much more.

Enter your email address:

Subscribe

If you do not find the verification email, check your spam folder

Delivered by FeedBurner (<https://feedburner.google.com>)

Categories

📁 Artifacts (<http://softwaretestingfundamentals.com/category/artifacts/>)

📁 Basics (<http://softwaretestingfundamentals.com/category/basics/>)

📁 Career (<http://softwaretestingfundamentals.com/category/career/>)



📁 [Defects \(http://softwaretestingfundamentals.com/category/defects/\)](http://softwaretestingfundamentals.com/category/defects/)

📁 [Levels \(http://softwaretestingfundamentals.com/category/levels/\)](http://softwaretestingfundamentals.com/category/levels/)

📁 [Methods \(http://softwaretestingfundamentals.com/category/methods/\)](http://softwaretestingfundamentals.com/category/methods/)

📁 [Metrics \(http://softwaretestingfundamentals.com/category/metrics/\)](http://softwaretestingfundamentals.com/category/metrics/)

📁 [Resources \(http://softwaretestingfundamentals.com/category/resources/\)](http://softwaretestingfundamentals.com/category/resources/)

📁 [Types \(http://softwaretestingfundamentals.com/category/types/\)](http://softwaretestingfundamentals.com/category/types/)

STF

[Disclaimer \(http://softwaretestingfundamentals.com/disclaimer/\)](http://softwaretestingfundamentals.com/disclaimer/)

[Privacy Policy \(http://softwaretestingfundamentals.com/privacy-policy/\)](http://softwaretestingfundamentals.com/privacy-policy/)

[Contact \(http://softwaretestingfundamentals.com/contact/\)](http://softwaretestingfundamentals.com/contact/)

Social

[Facebook Page \(https://www.facebook.com/softwareTestingFundamentals/\)](https://www.facebook.com/softwareTestingFundamentals/)

[Facebook Group \(https://www.facebook.com/groups/259049327448031/\)](https://www.facebook.com/groups/259049327448031/)

[Twitter \(https://twitter.com/stestingf\)](https://twitter.com/stestingf)



Ponder

Weinberg's Second Law: If builders built buildings the way programmers wrote programs, then the first woodpecker that came along would have destroyed civilization.

Copyright - 2018 STF | Theme by Colorlib (<http://colorlib.com/wp/>) Powered by WordPress (<http://wordpress.org/>)