

Este é um post de teste para verificar uploads de imagens de todos os tipos

Contents

Este é um post de teste para verificar uploads de imagens de todos os tipos

O que é teste manual?

O teste manual é uma técnica para testar o software que é realizada usando as funções e os recursos de um aplicativo. Em testes de software manuais, um testador testa o software seguindo casos de teste predefinidos. Neste teste, os testadores fazem casos de teste para os códigos, testam o software e fornecem o relatório final sobre esse software. Testes manuais são demorados porque são feitos por humanos, e há uma chance de erros humanos.

Cada novo aplicativo deve ser testado manualmente antes que seu teste possa ser automatizado. O teste manual requer mais esforço do que o teste de automação, mas é necessário para verificar a viabilidade da automação. Não há necessidade de conhecimento de qualquer ferramenta de teste em testes manuais.

Diferentemente dos testes automatizados, os testes manuais dependem de casos de teste predefinidos e podem ser demorados e sujeitos a erros humanos. Para dominar as nuances do teste manual e aprimorar suas habilidades de teste de software, considere explorar o [Guia Completo para Teste e Automação de Software por GeeksforGeeks](#). Este curso fornece percepções abrangentes sobre técnicas de teste manual, ajudando você a fornecer software mais confiável e de alta qualidade.

Tipos de testes manuais

Há diferentes [métodos para implementar o teste manual](#), mas ele é amplamente classificado em três tipos de teste manual:

1. Teste da caixa preta





Na técnica [Black Box Testing](#), o testador ou o analista de controle de qualidade só verificará a funcionalidade do módulo específico ou método específico ou, às vezes, todo o aplicativo fornecendo os diferentes casos de teste manualmente. Aqui, o testador fornecerá a entrada para o aplicativo e o testará manualmente.

Se retornar a saída esperada, o testador prosseguirá com outro conjunto de entradas e relatará todos os resultados à equipe. Se a entrada fornecida manualmente pelo usuário falhar durante o teste, ele/ela relatará esse problema à equipe de desenvolvimento.

Essas técnicas englobam testes funcionais e não funcionais.

HERE
IS A
SAMPLE



- [Ensaaios funcionais](#): Neste teste, o engenheiro de teste verifica se os recursos do aplicativo funcionam de acordo com os requisitos especificados. O testador verifica se o sistema faz o que deve ser feito com a conclusão adequada dos requisitos, com base nos casos de teste. Os testes funcionais podem ser feitos manualmente ou de forma automatizada, dependendo das necessidades do projeto.

Share Your Experiences

Basics

SDLC Models

Types of Testing

Types of Manual

White Box Techniques

Black Box Techniques

Types of Black Box

Types of Functional

QA in System Design

Manual testing is an important part of software development. Unlike automated testing, it involves a person actively using the software to find bugs and issues. This hands-on approach helps ensure the software works as intended and meets user needs.

Table of Content

- What is Manual Testing
- Types of Manual Testing
- Steps in Manual Testing
- Differences between Manual Testing and Automation Testing
- Tools Used for Manual Testing
- Characteristics of Manual Testing
- Why Need Manual Testing
- Advantages of Manual Testing
- Disadvantages of Manual Testing

What is Manual Testing

Manual testing is a technique to test the software that is carried out using the functions and features of an application. In manual software testing, a tester tests the software by following predefined test cases. In this testing, testers make test cases for the codes, test the software, and give the final report about that software. Manual testing is time-consuming because it is done by humans, and there is a chance of human errors.

Every new application must be manually tested before its testing can be automated. Manual Testing requires more effort than automation testing but is necessary to check automation feasibility. There is no requirement for knowledge of any testing tool in manual testing.

Unlike automated testing, manual testing relies on predefined test cases and can be time-consuming and prone to human error. To master the nuances of manual testing and enhance your software testing skills, consider exploring the [Comprehensive Guide to Software Testing & Automation by GeeksforGeeks](#). This course provides comprehensive insights into manual testing techniques, helping you deliver more reliable and high-quality software.

Types of Manual Testing

There are different methods to implement manual testing, but it is broadly classified into three types of manual testing:

White Box Testing

Black Box Testing

Grey Box Testing

1. Black Box Testing

In **Black Box Testing** technique, the tester or the QA analyst will only check the functionality of the particular module or particular method or sometimes the entire application by providing the different test cases manually. Here, the tester will give the input for the application and test it manually.

If it returns the expected output, then the tester will proceed with another set of inputs and report all the results to the team. If the input given by the user manually is failed during the testing, then he/she will report this issue to the development team.

These techniques encompass both functional and non-functional testing.

- Functional testing:** In this testing Test engineer checks whether the features of the application work according to the specified requirements. The tester verifies that the system does what it is supposed to do with proper requirement completion, based on the test cases. Functional testing can be done manually or automated, depending on the project needs.
- Non-functional testing:** It will focuses on the software all start to end performance, usability, and other quality aspects that do not directly relate to specific functions. This includes testing the application speed, reliability, and user experience to re-ensure it meets to the expected requirements.
- Regression testing:** This testing is done after any code changes, updates, or bug fixes to verify that the new code does not have negatively impact the existing features of the software which are already working fine. Since new code might be having bugs or conflict with old code, regression testing involves re-running previous test cases to check that everything still works as expected. This type of testing is crucial for maintaining the stability of the software after updates.

2. White Box Testing

In **White Box Testing** technique, the person will check the internal structure of the system like designs, coding, etc., manually. Here, the development team will review the entire coding part line by line to ensure the correctness of the code. If he/she finds any dissimilarities or errors in the code, they will correct or fix the errors in the coding or designs. Here, the process is entirely carried out manually and the process is efficient since the checking code or design is manually checked by humans.

Here are some key types of tests commonly used in white box testing:

Upcoming Courses

Web Full Stack...

System Design Train...

Generative AI Trainin...

Teacher Jobs

Download Jobs

- **Ensaio não funcional:** Ele se concentra no software, com todos os aspectos de desempenho, usabilidade e outros aspectos de qualidade do início ao fim que não se relacionam diretamente a funções específicas. Isso inclui testar a velocidade do aplicativo, a confiabilidade e a experiência do usuário para garantir que ele atenda aos requisitos esperados.
- **Ensaio de regressão:** Este teste é feito após qualquer alteração de código, atualização ou correção de bug para verificar se o novo código não tem um impacto negativo nos recursos existentes do software que já estão funcionando bem. Já que o novo código pode estar tendo bugs ou conflito com o código antigo, o teste de regressão envolve a repetição de casos de teste anteriores para verificar se tudo ainda funciona como esperado. Esse tipo de teste é crucial para manter a estabilidade do software após as atualizações.

Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.