

Documentação de Teste de Software: Maneiras de Registrar e Documentar os Testes de Software

A importância da documentação de testes de software relacionada à rastreabilidade e uma documentação completa do processo.

Elaborado por: Clara Beatriz Bonifácio

18 de janeiro de 2024

Registrando os Testes de Software: Relevância e Formas de Documentação

A documentação dos testes de software é uma etapa fundamental no processo de desenvolvimento de um projeto. Ela consiste em registrar e documentar todas as informações relevantes sobre os testes realizados, permitindo que a equipe de desenvolvimento e outras partes interessadas tenham acesso a informações importantes sobre o sistema testado.

A importância de registrar os testes de software está relacionada à rastreabilidade e documentação completa do processo. Ao registrar cada teste realizado, é possível acompanhar o histórico de execução e identificar possíveis falhas ou problemas recorrentes. Além disso, a documentação adequada dos testes permite que os resultados sejam compartilhados e analisados por outras equipes, como desenvolvedores e gerentes de projeto.

Métodos de Documentação de Testes de Software

Existem diferentes métodos de documentação de testes de software que podem ser utilizados, dependendo das necessidades e preferências da equipe. Alguns dos métodos mais comuns incluem:

→ Planilhas ou documentos de texto

- ◆ Essenciais para descrever detalhadamente os casos de testes, procedimentos, critérios de aceitação e resultados esperados.
- ◆ A padronização de templates pode ser adotada para garantir consistências na documentação.

→ Ferramentas de gerenciamento de testes

- ◆ O uso de plataformas dedicadas para gerenciamento de testes tais como: Jira, TestRail ou Xray, pode facilitar a organização, execução e monitoramento dos testes.
- ◆ Integração com ferramentas de desenvolvimento pode melhorar a colaboração entre equipes.

Independentemente do método utilizado, é essencial que a documentação de testes de software seja clara, concisa e completa. Isso garantirá que as informações

estejam disponíveis para toda a equipe e facilitem a análise e solução de problemas.

Ferramentas e Técnicas para Documentação de Teste de Software

A documentação de testes de software pode ser facilitada pelo uso de ferramentas e técnicas específicas. Essas ferramentas e técnicas ajudam a tornar o processo de documentação mais eficiente, preciso e organizado. Nesta seção, exploraremos algumas das principais ferramentas e técnicas disponíveis para a documentação de testes de software:

→ Sistema de gerenciamento de testes

- ◆ Além de mencionar ferramentas específicas, destaque a importância de registrar informações como status de execução, bugs encontrados e métricas de desempenho.

→ Captura de tela e Gravações de vídeo

- ◆ São ferramentas valiosas para ilustrar visualmente cenários de teste, facilitando a compreensão e reprodução de problemas.
- ◆ Podem ser úteis para documentar a interface do usuário e interações específicas.

→ Notas explicativas

- ◆ Inclua detalhes adicionais, explicações sobre decisões de teste, configurações específicas utilizadas e quaisquer outras observações que julgue serem relevantes para o escopo do projeto.

Ao utilizar essas ferramentas e técnicas de documentação de testes de software, a equipe pode garantir que todas as informações relevantes sejam registradas e documentadas de forma adequada. Isso facilita a rastreabilidade, a análise e a resolução de problemas durante o processo de desenvolvimento de software.

Boas Práticas na Documentação de Teste de Software: Como Garantir a Rastreabilidade e Coerência

A documentação de testes de software requer o uso de boas práticas para garantir que ela seja efetiva, precisa e de fácil compreensão para todas as partes envolvidas. Quando seguidas corretamente, essas boas práticas podem ajudar a garantir a rastreabilidade dos testes realizados e a coerência das informações registradas. Aqui estão algumas práticas recomendadas na documentação de teste de software:

→ Padronização de formatos

- ◆ Estabeleça um formato consistente para documentos de teste, facilitando a leitura e interpretação por diferentes membros da equipe.
- ◆ Considerar o uso de linguagem clara e concisa para evitar ambiguidades.

→ Detalhamento dos casos de teste

- ◆ Crie casos de teste detalhados, incluindo entradas, passos de execução, resultados esperados e critérios de aceitação.
- ◆ Utilize marcadores visuais para destacar pontos críticos ou complexos nos testes.

→ Uso de documentação visual

- ◆ Além de capturas de tela, considere a criação de diagramas ou fluxogramas para representar fluxo de testes complexos.
- ◆ Ferramentas como Draw.io podem ser úteis para criar documentação visual de forma colaborativa.

→ Revisão e validação

- ◆ Inclua um processo de validação sistemática da documentação de teste por diferentes membros da equipe
- ◆ A validação regular assegura que a documentação reflete com precisão os requisitos e as funcionalidades do software

Ao implementar essas boas práticas na documentação de testes de software, as equipes podem garantir a rastreabilidade dos testes, facilitar a compreensão das informações e promover a coerência entre os documentos.

Essa abordagem resultará em um processo de desenvolvimento mais eficiente, com menor margem de erro e maior qualidade nos resultados obtidos.

Benefícios da Documentação de Teste de Software: Como Aumentar a Eficiência e a Qualidade dos Testes

Investir na documentação de testes de software pode trazer uma série de benefícios para as equipes de desenvolvimento, para o projeto e por consequência, para a empresa como um todo. Esses benefícios incluem:

- Rastreabilidade
- Compartilhamento de informações
 - ◆ Além de desenvolvedores e gerentes de projeto, destaque a importância da documentação para equipes de QA, suporte técnico e stakeholders externos.
- Reprodutibilidade
 - ◆ Destaque como uma documentação detalhada permite a reprodução eficiente dos testes, facilitando a identificação e correção de problemas.
- Melhoria contínua
 - ◆ Enfatize que a análise regular da documentação pode levar a melhorias contínuas nos processos de teste e no software em desenvolvimento.

Exemplo prático referente a teste de software – Critérios de aceite descrito como especificação por exemplos

Uma documentação obrigatória para habilitar o entendimento dos requisitos, guia do desenvolvimento e validação pelo usuário final é o Critério de Aceite. Ela é criada geralmente em conjunto com o time e o cliente.

Este pode ser escrito de diversas formas. A mais recomendada para o entendimento por todo time é através da Especificação por Exemplos. Através dela podemos, de

uma só vez, informar como será o detalhe do requisito em termos de utilização mais o exemplo de dados que utilizaremos.

Uma forma comum é adotar a escrita dos critérios de aceite através do Gherkin - no qual Sua função é padronizar a forma de descrever especificações de cenários, baseado na regra de negócio - (Given, When, Then). Abaixo um exemplo:

- **Dado** que eu estou na página inicial da Adaptworks;
- **Quando** eu pesquisar pelo treinamento “Agile Testing”;
- **Então** o treinamento “Agile Testing” é apresentado;
- **E** o treinamento “Agile Testing Automation” é apresentado.
- **Mas**: No geral serve a mesma funcionalidade do “E”, porém é normalmente utilizado após uma validação negativa depois do “Então”;

Para saber mais sobre automatização de testes utilizando Gherkin, leia aqui 😊:

<https://blog.onedaytesting.com.br/gherkin/#:~:text=c%C3%B3digo%20e%20funcionalidades,-,0%20Gherkin,baseado%20na%20regra%20de%20neg%C3%B3cio.>

E para saber mais sobre BDD, acesse aqui 😊:

<https://blog.onedaytesting.com.br/bdd-introducao/>

Exemplo prático referente a teste de software – Plano para testes de performance

O plano para a execução dos testes de performance é um exemplo de uma documentação de teste evolutiva ao longo do projeto. Vamos pegar o exemplo onde precisamos executar testes de performance antes da entrega da release. A cada iteração iremos planejar quais serão os fluxos criados para o script de teste, coletar a massa de dados necessária, conversar com o pessoal de infraestrutura para ter um ambiente suportado ou mesmo o ambiente de produção, aplicação de monitores em diversas partes da arquitetura etc...

Isso é muito difícil de fazer em apenas uma iteração, onde dividimos todas as tarefas referente a esta ação em diversas partes.

Conclusão

A partir das colocações feitas acima, o processo de documentação de testes de software representa uma etapa crucial e de extrema relevância no ciclo de desenvolvimento de projetos, uma vez que esta etapa envolve o registro e a documentação abrangente de todas as informações relacionadas aos testes executados. Essa prática visa fornecer acesso a dados cruciais sobre o sistema testado, não apenas à equipe de desenvolvimento, mas também a outras partes interessadas.

Assim sendo, a relevância de documentar os testes de software está intrinsecamente ligada à rastreabilidade e à documentação completa do processo. Ao registrar cada teste, é possível estabelecer um histórico de execução, possibilitando a identificação de falhas ou problemas recorrentes que podem ser corrigidos com mais rapidez. A documentação apropriada dos testes, além disso, permite que os resultados sejam compartilhados e analisados por diferentes equipes, incluindo desenvolvedores, analistas, líderes técnicos e gerentes de projeto.

Existem diversos métodos de documentação de testes de software, cuja escolha depende das necessidades e preferências da equipe. Alguns métodos comuns incluem a elaboração de planilhas ou documentos de texto, a utilização de ferramentas de gerenciamento de testes e a adoção de planos padronizados. Independentemente do método escolhido, a clareza, concisão e abrangência da documentação são elementos essenciais para facilitar a análise e solução de problemas por toda a equipe.

No contexto das ferramentas e técnicas para documentação de testes, algumas abordagens específicas incluem o uso de sistemas de gerenciamento de testes, capturas de tela, gravação de vídeo e notas explicativas. Essas ferramentas e

técnicas visam tornar o processo de documentação mais eficiente, preciso e organizado, facilitando a rastreabilidade, análise e resolução de problemas durante o desenvolvimento de software.

A implementação de melhores práticas na documentação de testes é crucial para assegurar sua efetividade, precisão e compreensibilidade por todas as partes envolvidas. Práticas recomendadas incluem a padronização de formatos, o detalhamento minucioso dos casos de teste, a utilização de documentação visual e a realização de revisões e validações sistemáticas.

Logo, investir na documentação de testes de software proporciona diversos benefícios às equipes de desenvolvimento, ao projeto e à empresa como um todo. Estes benefícios, tais como descritos anteriormente, incluem a rastreabilidade eficiente, o compartilhamento de informações entre equipes, a reprodutibilidade dos testes e a possibilidade de melhorias contínuas.

Em síntese, a documentação de testes é uma prática essencial para garantir eficiência e qualidade nos testes realizados, promovendo maior eficiência, comunicação aprimorada e oportunidades para aprimoramento constante no processo de desenvolvimento de software. Portanto, é imperativo investir na documentação de testes como parte integral de projetos orientados à qualidade.