

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS



Engenharia de Software

Marcus Brasileiro Passos

Goiânia, 10 de junho de 2023

Sumário

1. Introdução	3
2. Teste de Software.....	3
2.1. Postman	3
2.2. Selenium	3
2.3. OpenVAS.....	4
3. Configuração de Software	4
3.1. Git.....	4
3.2. Docker	4
3.3. Puppet.....	4
4. Projeto de Software	4
4.1. Visual Studio	5
4.2. Lucidchart	5
4.3. DBeaver.....	5
5. Conclusão.....	5
6. Bibliografia.....	6

1. Introdução

Na Engenharia de Software temos diversas fases, que podem variar dependendo do modelo ou metodologia utilizada, mas em geral as fases mais comuns são levantamentos de requisitos, análise, projeto, codificação, teste, implantação e manutenção. Como solicitado foi realizado uma pesquisa nas fases de teste de software, configuração de software e projeto de software, essas fases são de extrema importância para garantir uma qualidade satisfatória e garantir o ciclo de vida de um software. Abaixo terá alguns exemplos de ferramentas que podem ser utilizadas de acordo com a necessidade, plataforma e/ou linguagem utilizada no desenvolvimento do Software.

2. Teste de Software

Na fase de Teste de Software é onde devem ser escolhidas algumas das ferramentas que vão garantir a integridade, segurança, integração (se necessário), qualidade, usabilidade e desempenho de um software. Devem ser selecionadas algumas ferramentas para serem utilizadas conforme cada necessidade do projeto como, linguagem, plataforma (móvel, web e etc.), integrações entre outros.

2.1. Postman

Nos dias atuais existem muitos softwares com integrações via API, uma ferramenta muito utilizada para testes de API é o Postman, ela permite que os desenvolvedores testem e desenvolvam facilmente suas APIs, sem precisar escrever código. Com o Postman, os desenvolvedores podem enviar solicitações HTTP (como GET, POST, PUT e DELETE) para uma API e verificar as respostas retornadas. Em resumo, o Postman é uma ferramenta muito útil para desenvolvedores que trabalham com APIs, pois ele permite testar e desenvolver facilmente suas APIs, sem precisar escrever código, além de oferecer recursos para automatizar testes, colaboração, documentação e outros.

2.2. Selenium

Uma ótima ferramenta para realizar testes de performance é o Selenium, ela é um framework gratuito que pode ser utilizada para realizar testes de performance em diferentes navegadores e plataformas, como Windows, Mac, e Linux. Ela também é compatível com as principais linguagens utilizadas atualmente como Java, C#, PHP, Python entre outras.

2.3. OpenVAS

Partindo para o teste de segurança existem diversas ferramentas para auxiliar nessa importante missão. Um deles é o Scanner de Vulnerabilidade OpenVAS, uma ferramenta de análise de vulnerabilidade verificando qualquer serviço que tenham portas abertas, configurações incorretas e vulnerabilidades nas instalações existentes exibindo um relatório automatizado enviado via e-mail.

3. Configuração de Software

O uso dos sistemas de Gerência de Configuração é fundamental para prover controle sobre os artefatos produzidos e modificados por diferentes recursos desde o planejamento e levantamento de requisitos até a construção e entrega do produto.

3.1. Git

Uma boa ferramenta e indispensável é uma ferramenta de controle de versões. O Git é uma ferramenta amplamente utilizada, que permitem varias pessoas trabalhem em um projeto simultaneamente, mantendo um histórico de alterações e facilitando no controle de alterações de códigos fontes. No Git é possível o usuário executar diversos comandos como Init para criar um repositório novo, Clone para clonar códigos de um repositório, add onde adiciona um arquivo alterado a um staging área, push onde envia arquivos de um repositório local a um repositório remoto, pull é responsável por executar o comando contrário do push, trazendo o arquivo de um repositório remoto para o repositório local, o merge serve para unir arquivos alterados ao arquivo original de um projeto e por fim o comando log que apresenta um histórico de commits de um usuário ou arquivo.

3.2. Docker

Também temos o Docker, uma ferramenta tem como vantagens a velocidade, economia e segurança que permite virtualização de contêiner que permite empacotar aplicativos e suas dependências em contêineres isolados. Isso simplifica a implantação e o gerenciamento de aplicativo em diferentes ambientes. Essa rotina é muito útil, pois, podem gerar ambientes diferentes com diversos SGBD sem interferências entre eles. O Docker elimina muitos problemas de dependências e retrabalho além de padronizar os ambientes para todos desenvolvedores da empresa independentes se são ambientes simples ou complexos.

3.3. Puppet

O Puppet é uma aplicação de automação de TI escrita em Ruby, oferecendo várias opções de solução, incluindo o Puppet Enterprise. Ele é escalável e projetado para automatizar infraestruturas híbridas em larga escala. O Puppet normalmente é executado como uma solução baseada em agente, exigindo software extra nos dispositivos gerenciados, mas também oferece recursos sem agente. Ele segue a programação declarativa, permitindo que o usuário defina o estado desejado das máquinas gerenciadas e automatize as etapas para alcançar esse estado.

4. Projeto de Software

Um bom projeto de software busca otimizar o uso de recursos, minimizar erros e riscos, e entregar um produto final que atenda às expectativas dos clientes. Além disso, considera-se também a manutenção contínua do software, com a aplicação de atualizações e correções de bugs ao longo do tempo. Em resumo, o projeto de software é a disciplina que envolve o planejamento e a implementação de um produto de software, seguindo metodologias e práticas para garantir a entrega de soluções eficientes e de alta qualidade.

4.1. Visual Studio

O Visual Studio é uma das IDEs mais populares para o desenvolvimento de software. Ela fornece um ambiente integrado multiplataforma permitindo o desenvolvedor desenvolver software para Windows, Mac, Android, IOS e nuvem. O Visual Studio oferece suporte a várias linguagens como C++, C#, F#, JavaScript, TypeScript, Python. Ela é muito utilizada para criação de API e app móvel. Essa IDE é completa, permite o desenvolvedor depurar seu código, testar seu código, compilar o aplicativo e até mesmo o controle de versão com o Git dentro da própria IDE.

4.2. Lucidchart

Ferramentas para a criação de diagramas e UMLs também são bem necessárias no projeto de software, para isso pode ser utilizada a ferramenta Lucidchart. Ela permite que vários usuários atualizem e comentem simultaneamente o mesmo diagrama, contém diversos modelos de diagramas e permite importar diagramas de outras plataformas como diagrams.net, Visio, Gliffy e OmniGraffle.

4.3. DBeaver

Essa é uma ótima ferramenta para gerir bases de dados, é uma ferramenta multiplataforma direcionada para desenvolvedores e DBA. Ela permite fazer ER diagramas e tem um suporte a uma vasta opção de conexão com SGBD SQL e NoSQL como MySQL, PostgreSQL, SQLite, Oracle, Db2, Firebird, SQLServer, MongoDB, Apache Cassandra, Amazon DynamoDB entre outros. Essa com certeza é uma ferramenta indispensável no projeto de software, uma vez que te dar um amplo suporte quando necessário conectar a um banco de dados.

5. Conclusão

Este trabalho abordou as fases de teste de software, configuração de software e projeto de software, destacando a importância de cada uma delas para garantir a qualidade e o ciclo de vida de um software.

Na fase de teste de software, foram apresentadas algumas ferramentas essenciais, como o Postman para testes de API, o Selenium para testes de performance e o OpenVAS para testes de segurança. Essas ferramentas permitem verificar a integridade, segurança, qualidade e desempenho de um software, garantindo uma experiência satisfatória aos usuários.

A configuração de software foi abordada destacando a importância do controle de versões e foram apresentadas duas ferramentas relevantes nesse contexto. O Git é

amplamente utilizado para controle de versões, permitindo que várias pessoas trabalhem em um projeto simultaneamente e mantendo um histórico de alterações. O Docker, por sua vez, facilita a virtualização de contêineres, simplificando a implantação e o gerenciamento de aplicativos em diferentes ambientes.

No projeto de software, foi ressaltada a importância de otimizar o uso de recursos, minimizar erros e riscos, e entregar um produto final que atenda às expectativas dos clientes. O Visual Studio foi mencionado como uma IDE popular e versátil, oferecendo suporte a várias linguagens e plataformas. Além disso, a ferramenta Lucidchart foi apresentada como uma opção para criar diagramas e UMLs, facilitando a visualização e comunicação do projeto. Por fim, a ferramenta DBeaver foi destacada como uma excelente opção para gerenciar bases de dados, oferecendo suporte a uma ampla variedade de SGBDs SQL e NoSQL.

6. Bibliografia

<https://www.devmedia.com.br/testando-apis-web-com-o-postman/37264>

<https://hnz.com.br/selenium-saiba-o-que-e-suas-vantagens-e-como-utilizar/>

<https://cryptoid.com.br/criptografia-identificacao-digital-id-biometria/as-10-melhores-ferramentas-de-verificacao-de-vulnerabilidades-para-testes-de-penetracao-2019/#:~:text=Scanner%20de%20Vulnerabilidade%20OpenVAS&text=Esses%20scanners%20pr%20ocurar%C3%A3o%20um%20endereço%C3%A7o,e%20vulnerabilidades%20nas%20instala%C3%A7%C3%B5es%20existentes>

<https://kenzie.com.br/blog/o-que-e-git/#:~:text=Outro%20grande%20benef%C3%ADcio%20do%20GIT,o%20hist%C3%B3rico%20de%20seu%20trabalho>

<https://logap.com.br/blog/o-que-e-docker/#:~:text=Principais%20vantagens%20do%20Docker&text=Economia%3A%20gr%C3%A7as%20%C3%A0%20caracter%C3%ADstica%20mencionada,n%C3%ADvel%20de%20seguran%C3%A7a%20da%20ferramenta>

<https://www.redhat.com/pt-br/topics/automation/ansible-vs-puppet#:~:text=Uma%20das%20principais%20vantagens%20do,infraestruturas%20h%C3%ADbridas%20em%20larga%20escala>

<https://learn.microsoft.com/pt-br/visualstudio/get-started/visual-studio-ide?view=vs-2022>

<https://www.lucidchart.com/blog/pt/como-usar-os-recursos-basicos-do-lucidchart>

<https://www.lucidchart.com/pages/pt/lucidchart-x-drawio>

<https://pplware.sapo.pt/software/dbeaver-nunca-tao-facil-gerir-bases-dados/>