Integração Contínua de Código

Amauri P. Sousa

Centro Universitário De Anápolis - (UNIEVANGÉLICA))

Bacharelado Em Engenharia De Computação

{amauri.sousa}@aluno.unievangelica.edu.br.

Abstract. *Continuous integration is a DevOps software development practice where developers often bundle their code changes into a central repository. After that, builds and tests are performed. Continuous integration typically refers to the stage of creation or integration of the software release process, as well as originating an automation component (eg, a CI or authoring service) and a cultural component (eg, learning to integrate with frequency). The main goals of continuous integration are to find and investigate bugs faster, improve software quality and reduce the time it takes to validate and release new software updates.*

Keywords*: integration, Continuous, DevOps, repository, software.*

Resumo. *A integração contínua é uma prática de desenvolvimento de software de em que os desenvolvedores, com frequência, juntam suas alterações de código em um repositório central. Depois disso, criações e testes são executados. Geralmente, a integração contínua se refere ao estágio de criação ou integração do processo de lançamento de software, além de originar um componente de automação (ex.: uma CI ou serviço de criação) e um componente cultural (ex.: aprender a integrar com frequência). Os principais objetivos da integração contínua são encontrar e investigar bugs mais rapidamente, melhorar a qualidade do software e reduzir o tempo que leva para validar e lançar novas atualizações de software.*

Palavras-chaves: *integração, contínua, desenvolvedores, repositório, software.*

1. Introdução

A integração contínua é um termo originado na metodologia ágil XP e utilizado em diversas metodologias, consistindo em algo simples: o desenvolvedor integra o código alterado e/ou desenvolvido ao projeto principal na mesma frequência com que as funcionalidades são desenvolvidas, sendo feito muitas vezes ao dia ao invés de apenas uma vez. O objetivo principal de utilizar a integração contínua é verificar se as alterações ou novas funcionalidades não criaram novos defeitos no projeto já existente. A prática da integração contínua pode ser feita através de processos manuais ou automatizados, utilizando ferramentas como o Jenkins, Hudson entre outros.Para a utilização contínua, é importante que a equipe possa trabalhar em grupo, para isso é importante que se tenha um sistema de controle centralizado de versão.Como parte crucial no processo de desenvolvimento e, para o sucesso da integração contínua, o controle de versão deve ser utilizado. Sendo que um dos objetivos do controle de versão é o trabalho colaborativo, em que diversos desenvolvedores trabalham em conjunto e compartilhando dados. O controle de versão é essencial para metodologias como o XP, em que a equipe deve trabalhar em conjunto constantemente.

2. Referencial Teórico

Desenvolvimento de software

O desenvolvimento é um processo composto por várias tarefas para se chegar ao produto final: o software. De acordo com Lima (2012, p. 3):

Nesse contexto, o processo de software é composto por diversas atividades que resultam no produto final. Entretanto, esse processo é flexível ao trabalho à ser realizado, como Pressman (2011, p. 40) enfatiza:

> "Integração Contínua é uma prática de desenvolvimento de software onde os membros de um time integram seu trabalho frequentemente, geralmente cada pessoa integra pelo menos diariamente – podendo haver múltiplas integrações por dia. Cada integração é verificada por um build automatizado (incluindo testes) para detectar erros de integração o mais rápido possível. Muitos times acham que essa abordagem leva a uma significante redução nos problemas de integração e permite que um time desenvolva software coeso mais
>
> rapidamente." - Martin Fowler

3. Método de Pesquisa

O trabalho iniciou-se de um estudo bibliográfico, através de uma pesquisa exploratória a respeito da Integração contínua de código, o controle de versão e integração Contínua, ferramentas para integração Contínua de Código. Quais os pontos positivos e negativos da mesma.

A natureza de pesquisa do presente trabalho é a de resumo de assunto, especificamente uma revisão sistemática da literatura. O objetivo é exploratório. Os procedimentos técnicos utilizados foram os de pesquisa bibliográfica.

Esta revisão da literatura investigou [...] durante o período de 2014 até 2019. As questões de pesquisa são: [...]. Um total de [...] artigos foram analisados e comparados.

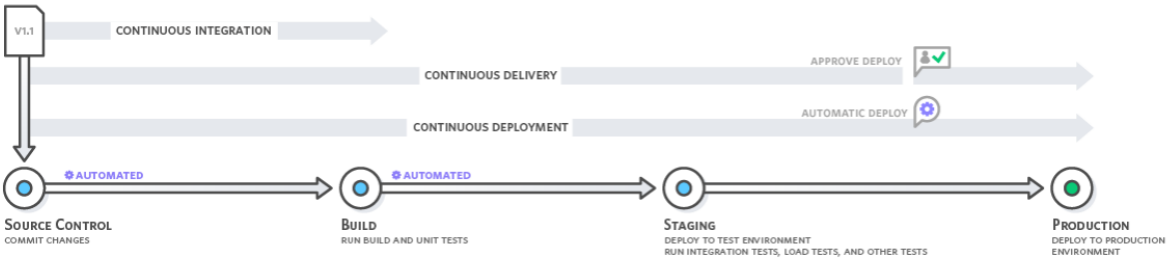
Utilizamos string query [...].

Delimitação: [...]26.

4. Abordagem Proposta / Estudo na FTT

No passado, os desenvolvedores de uma equipe podem trabalhar isoladamente por um longo período e só juntar suas alterações à ramificação mestre quando concluísse em seu trabalho. Dessa forma, a junção das alterações de códigos era difícil e demorada, além de resultar no acúmulo de erros sem correção por longos períodos. Estes fatores dificultavam uma distribuição de atualizações rápida para os clientes.

5. Figuras e Legendas



A integração contínua é referente aos estágios de criação e teste de unidade do processo de lançamento de software. Cada revisão confirmada aciona criação e teste automatizados.

Figura 1.

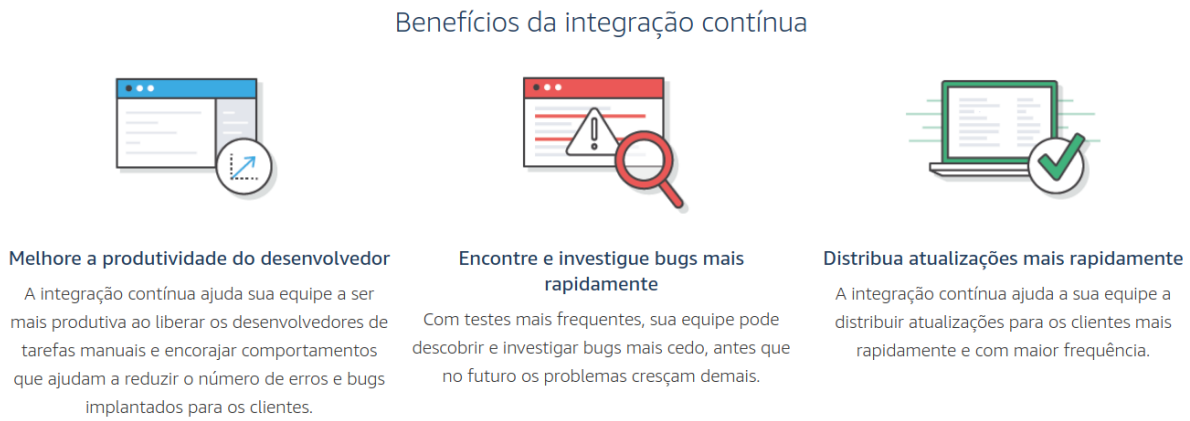


Figura 2. Configure um fluxo de trabalho de integração contínua com o AWS CodePipeline, que permite criar um fluxo de trabalho que compila código no AWS CodeBuild todas as vezes que você confirma uma alteração.

6. Resultados e Discussões

Entre os benefícios da utilização de um sistema de controle de versão, temos a possibilidade de restaurar versões anteriores do sistema;comparar códigos;gestão de alteração.É possível utilizar o sistema de controle de versão para gerenciar códigos, documentos, scripts de teste entre outros. Além disso, é possível criar uma linha alternativa para o desenvolvimento, o chamado branches.O controle de versão funcionará de forma básica da seguinte forma:O desenvolvedor faz o seu código, efetua um build antes de integrar seu código com a base

principal; Após realizar o build, o sistema deve ser integrado a base do sistema de controle de versão através de sincronização; Este processo deve ser feito frequentemente, evitando-se assim o acúmulo de codificação para a integração ao repositório. Algumas metodologias ditam que o desenvolvedor só pode considerar como pronto o trabalho quando o trabalho estiver sincronizado e então o desenvolvedor realizar um build na máquina de integração e após todos os testes serem executados com sucesso.

7. Considerações Finais

A integração contínua com ferramentas automatizadas traz diversos benefícios, como vimos. Primeiro é que o trabalho em equipe gera menos erros, reduz riscos, pois como o sistema é integrado continuamente e rapidamente, os erros também são detectados na mesma velocidade.Os bugs neste formato, não se cumulam, não gerando um problema geral no sistema, pois ao invés de demorar um dia ou uma semana, o erro é detectado em horas. Pois mesmo o desenvolvedor testando na codificação, ao integrar, erros podem surgir e quanto antes os erros forem detectados, antes serão corrigidas.

Referências

LIMA, Alessandra. Processo de Desenvolvimento de Software – PDS. Instituto Nacional de Tecnologias da Informação; Coordenação-Geral de Planejamento, Orçamento e Administração Coordenação de Desenvolvimento, Infraestrutura e Suporte – CODIS; Processo de Desenvolvimento de Software (PDS - ITI), Nov. 2012. Disponível em: < http://www.iti.gov.br/images/institucional/politicas/PDS_ITI.pdf>.

PRESSMAN, Roger S. Engenharia de Software: uma abordagem profissional. Tradução: Ariovaldo Griesi; Mario Moro Fecchio; revisão técnica Reginaldo Arakaki; Julio Arakaki; Renato Manzan de Andrade. 7 ed. Porto Alegre, AMGH, 2011.

https://www.devmedia.com.br/integracao-continua-uma-introducao-ao-assunto/28002

https://aws.amazon.com/pt/devops/continuous-integration/

https://www.opus-software.com.br/o-que-e-integracao-continua/

https://blog.rocketseat.com.br/integracao-continua-ci-do-zero/

https://www.desenvolvimentoagil.com.br/xp/praticas/integracao

http://demoiselle.sourceforge.net/process/ds/1.2.3-BETA1/ProcessoDemoisellePlugin/guidances/guidelines/integracaoContinua_41368505.html

https://pt.wikiversity.org/wiki/Integra%C3%A7%C3%A3o_Cont%C3%ADnua