

Monografia I

Acadêmico: Alefe Variani

Atividade 2

b) Entrega das etapas: Objetivo geral, Objetivos específicos, Justificativa e a estrutura geral da revisão bibliográfica;

## **2. OBJETIVOS**

### **2.1 Objetivo geral**

Fazer um levantamento sobre tecnologias e práticas que utilizam os conceitos do DevOps, analisar às melhorias que o DevOps pode trazer para o ambiente de desenvolvimento de *software* e aplicá-la em uma situação real, para verificar se a abordagem realmente traz benefícios aos envolvidos no processo de desenvolvimento de *software*.

### **2.2 Objetivos específicos**

- Conhecer os processos do desenvolvimento de *software*.
- Entender conceitos da metodologia que se aplicam ao ambiente DevOps.
- Aplicar conceitos do DevOps no CRS – Unochapecó.
- Melhorar o ambiente de desenvolvimento e monitoramento da aplicação através do uso cultura do DevOps.
- Aplicar uma proposta em um ambiente desenvolvimento real, utilizando conceitos e práticas do DevOps.
- Analisar o comportamento, vantagens e desvantagens que a proposta traz ao ambiente de desenvolvimento.

### 3. JUSTIFICAVA

Entregar um *software* em produção se torna cada dia um processo mais difícil, ciclos longos para entregas e para testes, divisão entre equipes desenvolvimento e operações, aplicações sem monitoramento e ambientes de desenvolvimento e produção diferentes, são fatores para este problema. Mesmo equipes ágeis produzindo *software* entregáveis ao final de cada iteração, ainda sofre-se para chegar em produção quando encontram estas barreiras. (SATO, 2013).

Embora tenham surgido várias metodologias ágeis de desenvolvimento de *software*, e muitas empresas aderiram à uma metodologia ágil, desenvolvendo produtos mais rapidamente, as empresas ainda encontram problemas de monitoramento de suas aplicações, dificuldade em manter o ambiente de desenvolvimento idêntico ao ambiente de produção e realizar *deploy* de maneira fácil e com menos chances de erros no ambiente de produção. (SATO, 2013).

Sendo assim, o DevOps é um movimento cultural e profissional que tenta quebrar esta barreira, mostrando que desenvolvedores e engenheiros de sistema têm muito o que aprender uns com os outros (SATO, 2013). Segundo Sete (2015) este movimento se propõem a resolver alguns itens, como:

- Medos de mudanças, quando uma versão de *software* finalmente é entregue, equipes de operações acreditam que *software* e ambientes são coisas frágeis e vulneráveis, que qualquer mínima interferência é suficiente para quebrar a aplicação e tirá-la do ar. Criando assim um processo caro e burocrático para gestão de mudanças e então um longo e doloroso processo deve ser percorrido para que seja possível atualizar versões das aplicações ou mesmo corrigir erros.
- Liberações de versões (*Deployments*) arriscadas, desenvolvedores na maioria de vezes não sabem como os códigos se comportam no ambiente de produção, se está funcionando como o esperado, se vai suportar o volume de acesso. Estas respostas dos pontos citados demoram para chegar ao desenvolvedor ou às vezes nem chegam, acontecendo assim uma nova liberação de versão para o ambiente de produção, aguardando ou observando se ele vai continuar de pé.
- “Funciona na minha máquina” expressão bastante comum, são problemas que se manifestam apenas no ambiente de produção. Estes problemas normalmente são identificados pela equipe de operações, quando os próprios usuários descobrem e reportam os problemas. Após análise, o problema é informado aos desenvolvedores, que verificam e acabam dizendo “na minha máquina funciona”. Entretanto não são analisadas as diferenças entre o ambiente de desenvolvimento do ambiente de

produção e os vários fatores importantes diante disso, SO (sistema operacional), versões e arquivos de configuração são alguns deles.

- Silos<sup>1</sup>, geralmente nas empresas de TI as equipes são divididas por desenvolvedores e operações ou outras equipes que existam. Isso causa desperdício, podendo levar à uma cultura de os problemas serem transferidos entre as equipes. É comum que estes silos não estejam localizados fisicamente dentro do mesmo escritório, na mesma cidade ou ainda no mesmo país, tornado uma mentalidade entre os times de desconfiança, suspeita, criando um ambiente não agradável entre as áreas.
- Foco e Visão, existe claramente um déficit de visão compartilhada entre os responsáveis por construir e operar aplicações. Enquanto as equipes de desenvolvedores se convencem que os códigos estão ótimos, as equipes de operações se isolam, cuidando dos ambientes e combatendo tudo e todos que podem de alguma forma colocar em risco sua estabilidade.

Kevin Behr, autor e diretor de área tecnológica da *Praxis Flow*, explica "Você não pode comprar DevOps numa caixa" (RACKSPACE), Peter Drucker representa bem o maior desafio de uma iniciativa de DevOps. "Infelizmente, não se pode fazer download da cultura. Mudança de comportamento e foco nas pessoas é o que vai mudar a cultura das organizações". O grande ponto do DevOps é o modelo mental que preconiza que todos estão do mesmo lado, tentando a mesma coisa: entregar continuamente *software* de valor, capaz de transformar a vida das pessoas. (SETE, 2015).

---

1 Silos – são considerados os vários departamentos que a empresa de TI pode ter, onde cada um mantém suas funcionalidades, sem haver interações umas com as outras.

## ESTRUTURA GERAL DA REVISÃO BIBLIOGRÁFICA

### 4. REVISÃO BIBLIOGRÁFICA

#### 4.1 Apresentação dos diferentes Ambientes de desenvolvimento de *software*

##### 4.1.1 Ambiente de Desenvolvimento

##### 4.1.2 Ambiente de Produção

##### 4.1.3 Ambiente de Operações

##### 4.1.4 Considerações finais

#### 4.2 Monitoramento de aplicações

##### 4.2.1 Considerações finais

#### 4.3 Problemas entre equipes de Desenvolvimento e equipes de Operações

##### 4.3.1 Equipes de Desenvolvimento

##### 4.3.2 Equipes de Operações

##### 4.3.3 Problemas entre as Equipes

##### 4.3.4 Considerações finais

#### 4.4 DevOps

##### 4.4.1 Introdução

##### 4.4.2 Como surgiu

##### 4.4.3 Conceitos

##### 4.4.4 Ferramentas

##### 4.4.4.1 Gerenciamento do ambiente de Desenvolvimento

##### 4.4.4.2 Gerenciamento de configurações

##### 4.4.4.3 Gerenciamento das configurações da Aplicação

##### 4.4.4.4 Monitoramento da Aplicação

##### 4.4.5 Considerações finais