

Nome: Raylander Marques Melo Matrícula: 494563

PodCast -

A melhora da performance da empresa que trabalha com DevOps melhora significativamente.

Característica de empresas importantes sobre devops, tipo integração contínua, entrega contínua e afins

Maturidade das empresas para realizar vários deployments por dia pois, é preciso de várias coisas para se chegar nesse ponto.

Sistemas que vão lhe ajudar a realizar deployment contínuos

O tempo de desenvolvimento daquele deployment não pode demorar muito tempo como meses para sair, não adianta ter vários deployments em um dia que demoraram meses sendo desenvolvidos.

Perceber que no pull Request não se deve focar tanto em estrutura de código ou em variável.

O tempo de revisão de código em um caso para a liberação de uma nova build.

DevOps é uma cultura conjunto de prática e princípios.

A questão da automação de devops como testes e afins.

É melhor usar uma aplicação que já está funcionando do que mudar e ter problemas, linguagem.

Não é tão legal usar muitas linguagens pois fica complicado de escalar.

A equipe deve ter maturidade na hora de desenvolver.

Contradição de devops é dizer que ir rápido com segurança, ou seja, nem tudo é segurança o importante é reduzir os riscos e ter velocidade de implementação.

Problemas das mudanças por achar que é a frequência de deployment, quanto mais difícil quebre o problema em problemas menores.

A contradição dos testes não garante que o sistema não quebre.

Não tem como garantir segurança que um sistema vai funcionar sempre, o importante é criar sistemas confiáveis.

Tempo e dinheiro não são infinitos.

A catedral e o bazar

Quem imaginaria que um sistema operacional iria aparecer com a iniciativa de desenvolvedores que com seu tempo livre destinaram para desenvolver o linux?

Achar que grandes software teria que ser criado como catedrais de forma isolada e criados com cuidado, pequenos grupos desenvolvendo e nenhuma beta sendo liberada antes da entrega do software em si.

A comparação do linux como uma bazar por ser um grande barulhento onde todos falam e tem opinião .

As dificuldades de envios de correios na internet na época.

Todo bom trabalho de software começa colocando o dedo na ferida de um programador.

Os programadores bons sabem o que escrever. Os grandes sabem o que re-escrever.

Planeje jogar algo fora; você irá, de qualquer maneira.

Esteja preparado para jogar alguma versão do seu software fora.

Se você tem a atitude certa, problemas interessantes irão encontrá-lo.

Quando você perde interesse em um programa, sua última obrigação a fazer com ele é entregá-lo a um sucessor competente.

Tratar seus usuários como codesenvolvedores é seu caminho mais fácil para uma melhora do código e depuração eficaz.

Libere cedo. Libere frequentemente. E ouça seus fregueses.

Dada uma base grande o suficiente de beta-testers e co-desenvolvedores, praticamente todo problema será caracterizado rapidamente e a solução será óbvia para alguém

O custo total de manter um programa amplamente utilizado é tipicamente 40 por cento ou mais o custo de desenvolvê-lo. Surpreendentemente este custo é muito afetado pelo número de usuários.

Mostre-me seu código e esconda suas estruturas de dados, e eu poderei continuar mistificado. Mostre-me suas estruturas de dados, e eu provavelmente não necessitarei do seu código; ele será óbvio

Se você tratar seus beta testers como seu recurso mais valioso, eles irão responder tornando-se seu mais valioso recurso

A melhor coisa depois de ter boas idéias é reconhecer boas idéias dos seus usuários

Freqüentemente, as soluções mais impressionantes e inovadoras surgem ao se perceber que o seu conceito do problema estava errado

A perfeição (em projetar) é alcançada não quando não há mais nada a adicionar, mas quando não há nada para jogar fora

Para resolver um problema interessante, comece achando um problema que é interessante para você