



# From Zero To Hero

An evolution of a software lifecycle in a small company





Quem sou

# Paulo Gomes

Especialista em Desenvolvimento de Software na Daitan Group, 14 anos na área de TI, 8 como desenvolvedor. Amante de Board Games, RPG e Pixelart.

## Auto Intitulado “Toolsmith”



Website

<https://paulushcgcj.github.io>



LinkedIn

[www.linkedin.com/in/paulushc](https://www.linkedin.com/in/paulushc)



StackOverflow

<https://stackoverflow.com/story/paulushc>



Medium

<https://medium.com/@paulushc>



GitHub

<https://github.com/paulushcgcj>



# Agenda

O que é DevOps

Use Case

Problemas, Processos e  
Pessoas

Q & A



# O que é DevOps

Um conjunto de **práticas** que automatizam os **processos** entre equipes de desenvolvimento de software e de TI para resolver **problemas** críticos rapidamente e melhorar o gerenciamento do trabalho não planejado.

“Website da Atlassian”





Keep **CALMS** and Carry On

Culture

Automation

Lean

Measurement

Sharing



# Use Case

Multiportal - A Zero to Hero





# Cenário

Segundo semestre de 2012

- 01 | Código legado em C
- 02 | Sem IDE
- 03 | Silos de Conhecimento
- 04 | Falta de comunicação Suporte - Desenvolvimento
- 05 | Monitor do Monitor do Serviço

```
a=2, cx[300]={1,2}, cy[300]={7,7}, t=1, mx  
a='a';
```

```
x < 18; x++) // LINHA VERTICAL ESQ
```

```
color(BLUE);  
cy(0,x);  
F ("%c",219);
```

```
x < 50; x++) // LINHA HORIZONTAL S
```

```
cy(x,0);  
F ("%c",219);
```

```
x <18; x++) // LINHA VERTICAL DIRE
```

```
cy(50,x);  
F ("%c",219);
```

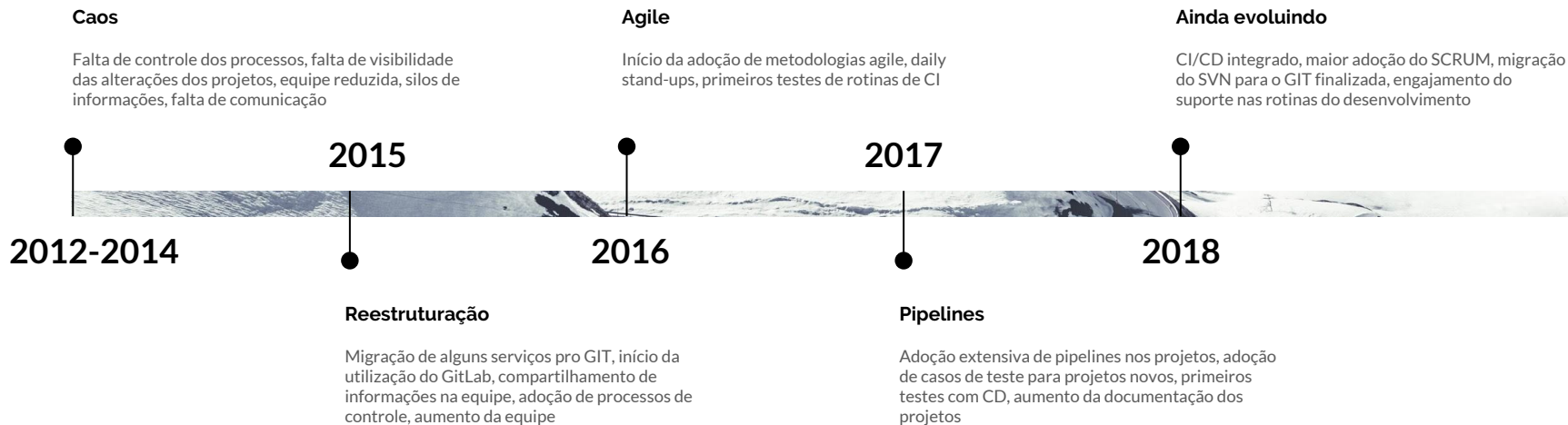
```
x<51; x++) // LINHA HORIZONTAL INFE
```

```
cy(x,18);  
F ("%c",219);
```





# Timeline Evolutivo





# Para processos - Ferramentas



## Git

Git permitiu que a equipe pudesse gerenciar o código de forma mais coesa e concisa, usando repositórios locais, e branches remotos.

## GitLab

GitLab nos deu mais visibilidade sobre as mudanças no código, e permite gerenciar issues, bases de conhecimento e MR facilmente.



## Automation

Com o stack GASD (Gitlab, Artifactory, Sonarqube, Docker), implementamos automação do nosso processo, rodando builds no docker por commit, checando o código com o sonarqube e deployando no artifactory.

# Para problemas - Soluções



## Adoção de boas práticas

Adotar algumas boas práticas no processo, como documentação, limpeza do código, testes e padronização.

## Diálogo entre equipes

A quebra da barreira entre equipes fez com que ambos os lados fossem beneficiados com uma visão mais ampla do processo como um todo.



## GSD\*

Admitir que há um problema e corrigi-lo, é a parte mais demorada, e às vezes impossível de ser feita sem um processo coeso e controlado. No final das contas, corrigir o problema é a única solução.

# Para pessoas - Boas práticas

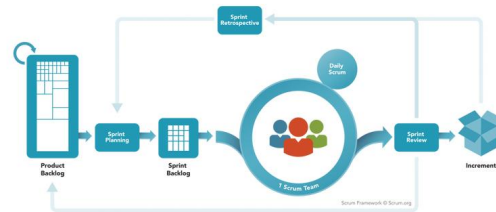


## Conversas diárias

Adoção de um processo ágil é complexa, então começamos com conversas diárias (Daily Stand-Up Meetings) para saber o que cada um está trabalhando e problemas enfrentados.

## Engajamento das equipes

Unir as equipes de desenvolvimento e suporte durante as reuniões de planejamento aumentou não só a visibilidade, mas a compreensão geral dos processos e serviços.



## Metodologias Ágeis

A continuação da adoção do SCRUM pela equipe aumentou o controle e definiu escopos tangíveis para a equipe, fazendo com que o trabalho fosse melhor distribuído e priorizado.

Moral da história

**Problemas** são resolvidos por **Pessoas**  
usando os devidos **Processos**.



**Chega de papo rapaz, quero numeros!!!!**



# Volume de Operações

Com uma base instalada com crescimento de cerca de 1TB por ano de dados, a demanda por processamento concorrente de todas as informações cresce exponencialmente

Base de Dispositivos Ativos

350K

Entre rastreadores, celulares e antenas satelitais.

Mensagens por Minuto

800K

Mensagens trocadas entre dispositivo <-> servidor

Dados por Mês

80GB

Entre dados de localização, telemetria e outros divididos em DB/NoSQL/Arquivos



# Dev + Ops

Com boa parte da plataforma baseada em Java, os serviços rodam em 13 ambientes 24/7 com SLA de 1 hora. Após todas as mudanças foi possível reduzir a indisponibilidade do sistema para menos de 30 minutos/mês

Desenvolvedores

11

Responsáveis por toda a stack de serviços

Serviços

70

Entre front-end e back-end, com média de 2 instâncias por ambiente.

Linhas de código

3M

Divididas entre Java, JavaScript, C, C++ e Bash



# Q & A







# Obrigado!

