**INTRODUÇÃO**

**ANTES** DA IMAGEM

* Não basta saber programar, tem que saber programar BEM.
* Não se escreve código para a MÁQUINA entender, se escreve para os DESENVOLVEDORES lerem.

**DEPOIS** DA IMAGEM

* Participar de PROJETO ANTIGO é dificil para os DESENVOLVEDORES pq sabem que o CÓDIGO é RUIM.

**JUSTIFICATIVA**

DESENVOLVEDORES

* Escrever código melhor
* Melhorar PROFISSIONALMENTE

EMPRESAS

* *Ler slide*

**PROBLEMA**

BUGS EVERYWHERE

* Em um NOVO PROJETO estima-se que ao final do desenvolvimento com **50 MIL linhas** de código existam **750 bugs**
* Esse número aumenta na MANUTENÇÃO: avalia-se que em **50 MIL** linhas de código encontrem-se **1.080 bugs**.

FINDING OUR WAY BAD CODE

* O TEMPO de FAMILIARIZAÇÃO com um sistema JÁ DESENVOLVIDO é muito ALTO.
* TEMPO NAVEGANDO pelo código, DEPURANDO...

**OBJETIVO**

*Ler slide*

**CÓDIGO RUIM CUSTA CARO**

* O Grande Replanejamento

**O QUE É CÓDIGO LIMPO?**

ANTES

* É uma ARTE
* Como se DEFINE ARTE?
* Caracteristicas OBJETIVAS e SUBJETIVAS

DEPOIS

* FAZER SORRIR
* Coberto por TESTES
* Aquele código que cada LINHA é aquilo mesmo que você ESPERAVA que fosse
* ELEGÂNCIA, AGRADABILIDADE, LEGIBILIDADE

**NOMES SIGNIFICATIVOS**

* Nomes são dificeis de se ACERTAR na PRIMEIRA VEZ.
* Deve existir uma MELHORA CONTÍNUA (refatoração).

**SOLID**

* **5 Princípios** de Programação Orientada a Objetos **IDENTIFICADOS** por **Robert Martin** em MEADOS de **2000**
* APLICANDO: sistema FÁCIL de MANTER e EXTENDER ao longo do tempo.

**REFERÊNCIA NULA**

* Conferência 2009 Londres

**Testes Unitários**

* A menor parte testável de um sistema
* Testa a unidade para verificar se ela está pronta para uso
* MUITOS NÃO TESTAM a unidade que implementaram, deixando o problema estourar mais pra FRENTE.

**MÉTODOS**

* 9 pessoas
* EXPLICAR o experimento e PROBLEMA

**RESULTADOS E DISCUSSÃO**

MANUTENIBILIDADE

* *Explicar o Gráfico*
* Alta manutenibilidade = TEMPO GASTO menor (MOSTRAR PRÓXIMO SLIDE)

TEMPO

* *Explicar o Gráfico*

QUANTIDADE DE LINHAS

* Ponto NEGATIVO no CÓDIGO LIMPO
* Teve mais linhas porque CONTINHA MAIS CLASSES

FACILIDADE DE LEITURA

* CÓDIGO LIMPO: 100% falaram BOA e EXECELENTE
* CÓDIGO CONVENCIONAL: 100% falaram PÉSSIMA

TAMANHO DE CLASSES e MÉTODOS

* Apesar do CÓDIGO LIMPO ter mais LINHAS, o MÉTODOS estavam PEQUENOS

COMENTÁRIOS

* Preferir EXPRESSIVIDADE pelo CÓDIGO
* *Mostrar PROJETO no VisualStudio*

**CONCLUSÃO**

ANTES DAS IMAGENS

* Apesar da PEQUENA AMOSTRA
* Apesar da FACILIDADE do PROBLEMA (Cálculo INSS)
* CÓDIGO LIMPO se DESTACOU

DEPOIS das IMAGENS

* Estratégias que visam o TREINAMENTO