



GCMES

DÍVIDA TÉCNICA

Prof. Dr. Rosana T. Vaccare Braga

Estagiário PAE: Karlos Oliveira

Baseado no material elaborado por Gustavo Avelar e Katia Damaceno

AGENDA

- Conceituação da Dívida Técnica – DT
 - Impacto da dívida técnica
 - Custo da dívida técnica
 - Tipos de dívida técnica
 - Consequências da dívida técnica
 - Estratégias para gerenciar a dívida técnica
-

DÍVIDA TÉCNICA OU DÉBITO TÉCNICO?

**VOU ELIMINAR SEU DÉBITO TÉCNICO DE
UMA VEZ POR TODAS...**



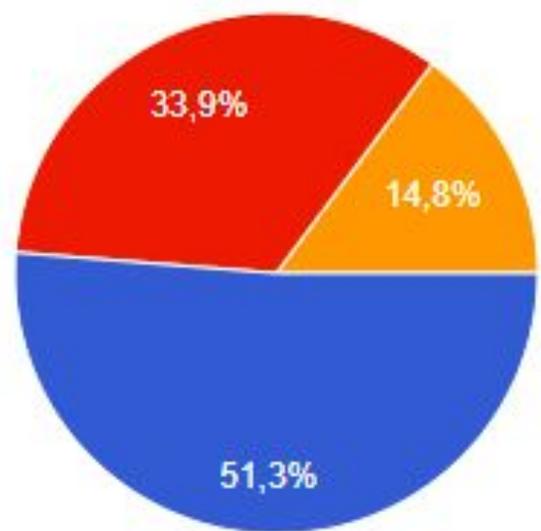
**NÃO EXISTE DÉBITO TÉCNICO EM
SOFTWARE!!!**

VOCÊ JÁ SABE O QUE É DÍVIDA TÉCNICA???



O termo dívida técnica é comum para você?

236 respostas



- Sim
- Não
- Talvez. Conheço débito técnico ou outro termo similar

SINTOMAS

“Não precisa se preocupar com documentação agora”

“Aquela parte do código só o Carlos consegue mexer”

“Para agora está bom, depois a gente melhora”



“Eu sei que quando eu mexo naquele código, todo o resto quebra”

“Estamos com o prazo apertado, faça como puder”

“Se eu parar para projetar não entrego”

Você já terminou o software?

Não, eu ainda estou pagando a dívida técnica do último programador que você apressou.

Eu não sei o que isso significa. Bem, isso explica muita coisa



COMO SURTIU O TERMO DÍVIDA TÉCNICA?

- Metáfora da Dívida Técnica por Cunningham (OOPSLA 92)



COMO SURTIU O TERMO DÍVIDA TÉCNICA?

- Metáfora da Dívida Técnica por Cunningham (OOPSLA 92)
- “...Quando a **qualidade do software é comprometida**, incorre-se numa dívida. Uma pequena dívida acelera o desenvolvimento até que seja paga com reescrita de software. O **problema** é quando a **dívida não é paga**. Cada minuto que o software é mantido na inconformidade, **juros** são acrescidos na forma de reescrita...”

É O GAP ENTRE FATORES COMO FALTA DE TEMPO E QUALIDADE DO SOFTWARE



CUSTO DA DÍVIDA TÉCNICA

- Segundo Gartner: Previsão de US\$ 1 Tri de gastos.

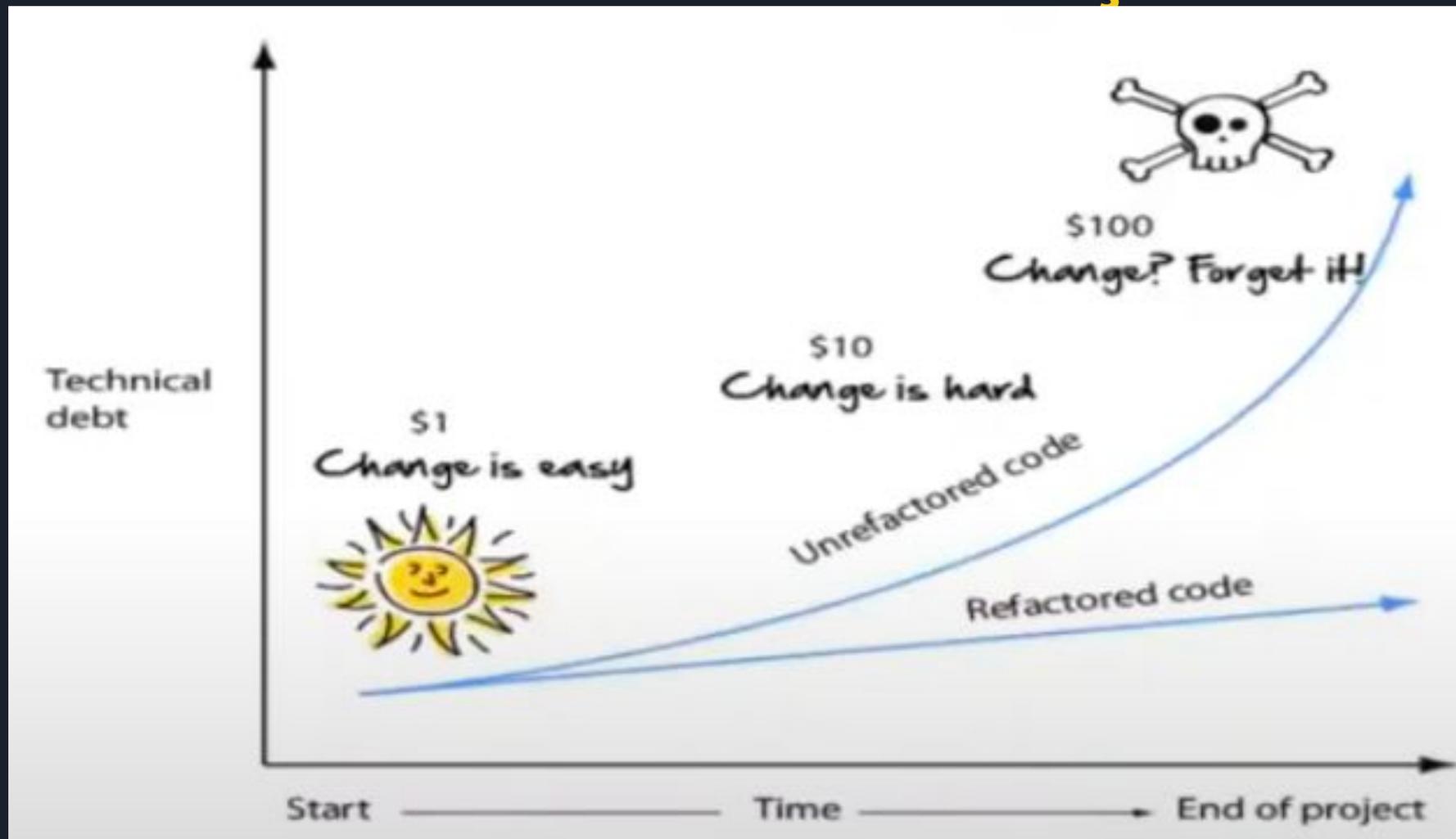


CUSTO DA DÍVIDA TÉCNICA

CURTIS ET AL., (2012)

- 745 aplicações de 160 organizações, em torno de 365 milhões de linhas de código.
 - Custo adicional em juros de **\$3,61** por linha de código
 - Aplicações com 300.000 mil linhas de código podem conter mais de \$1 milhão de dólares em juros
-

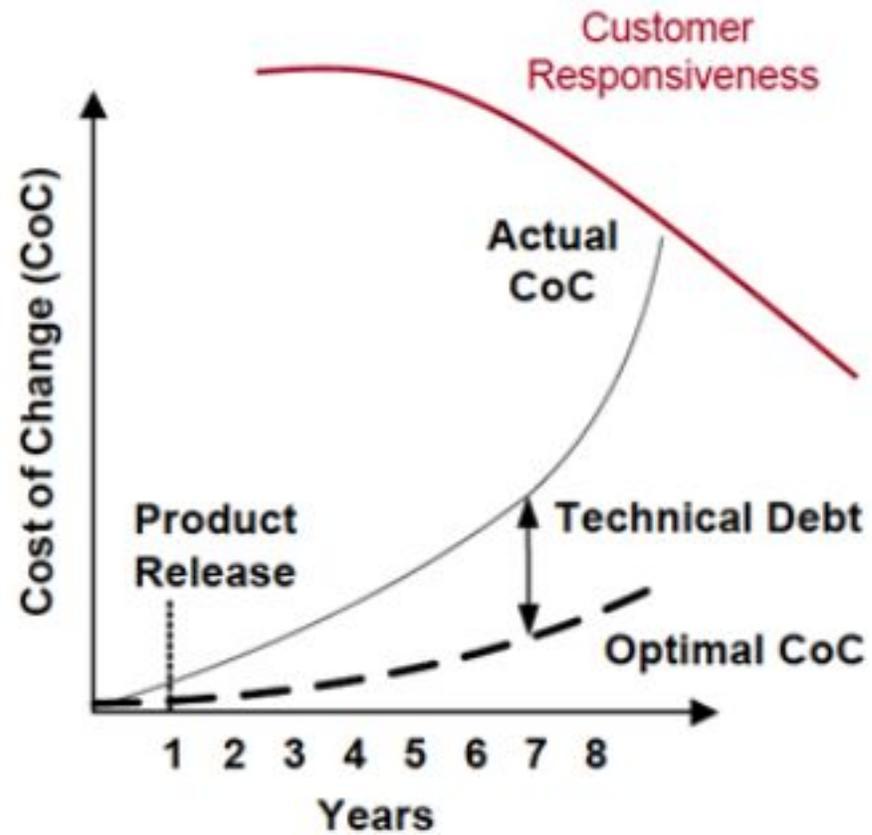
CUSTO DA MUDANÇA



IMPACTOS DA DÍVIDA TÉCNICA

- Produtividade da equipe
 - Alto índice de DT são 1,6 vezes menos produtivos
 - Negócios
 - Aplicação
-

IMPACTO NOS NEGÓCIOS



Technical Debt Chart from Jim Highsmith

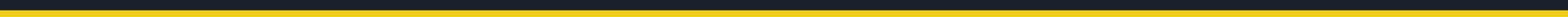
**ENTÃO, TODO PROBLEMA NO
SOFTWARE É UMA DÍVIDA TÉCNICA?**



DÍVIDA TÉCNICA

- Relacionada diretamente com

Artefatos ou código imaturos



EXEMPLOS DE DÍVIDA TÉCNICA

- Gambiarras
 - Classes ou scripts que realizam muitas funções
 - Nomenclatura e organização
 - Escolha inadequada de tecnologia, ferramenta ou plataforma ou mudança de versão
-

CAUSAS DA DÍVIDA TÉCNICA

Definição

insuficiente

Pressão do Negócio
(Político/Legal)

Alto Acoplamento

Ausência de Suítes/
kit de Testes

Falta de empenho/
motivação

**Lacuna de
Entendimento sobre DT**

Liderança fraca em
Tecnologia

Desenvolvimento Paralelo

Falta de documentação

Postergação da Refatoração

Especificação de “último minuto”

TIPOS BÁSICOS DE DÍVIDA TÉCNICA (STEVE MCCONNELL)

1. SEM QUERER (NÃO INTENCIONAL)

O primeiro tipo de DT é do tipo incorrido **involuntariamente**. Essa DT é o resultado não estratégico de se fazer um trabalho ruim.

Ex.: Aquisição de uma arquitetura de software com DT

2. INTENCIONAL

O segundo tipo de DT é do tipo incorrido **intencionalmente**. Ocorre geralmente, quando uma organização toma uma decisão consciente de otimizar o presente e não o futuro, mesmo que isso ocorra em detrimento da qualidade.

Ex.: Preciso lançar o produto com DT para não perder o time-to-marketing.

CLASSIFICAÇÃO DE DÍVIDA TÉCNICA DADA POR MARTIN FOWLER

	Imprudente	Prudente
Deliberado (intencional)	<p>Não há tempo “Não temos tempo para projetar”</p>	<p>Eu sei “Vamos lidar com as consequências”</p>
Inadvertido (sem querer)	<p>Bagunça “O que estamos fazendo?”</p>	<p>Tudo Ok... mas poderia ser melhor “Agora sabemos como devíamos ter feito isso”</p>

**BOM DESIGN E CÓDIGO LIMPO FAZEM
COM QUE VOCÊ VÁ MAIS RÁPIDO**

MARTIN FOWLER

CLASSIFICAÇÕES DE DT (LI, AVGERIOU E LIANG (2015))

- **Requisitos:** refere-se à distância entre a especificação de requisitos ideal e a aplicação efetiva do sistema, com base em pressupostos de domínio e suas restrições (ERNST, 2012).
- **Arquitetural:** é causada por decisões de arquitetura que fazem concessões em alguns aspectos internos de qualidade, tais como manutenção.
- **Projeto:** refere-se a atalhos técnicos que são tomados no projeto.
- **Código:** representa o código mal escrito que viola as melhores práticas de codificação ou regras de codificação. Exemplos incluem a duplicação de código e código excessivamente complexo.
- **Teste:** refere-se a atalhos tomados em testes. Um exemplo é a falta de testes (por exemplo, testes unitários, testes de integração e testes de aceitação).
- **Construção:** refere-se a falhas em um sistema de software no seu sistema de construção, ou em seu processo de construção que fazem a compilação excessivamente complexa e difícil.
- **Documentação:** diz respeito à documentação insuficiente, incompleta ou desatualizada em qualquer aspecto do desenvolvimento de software. Exemplos incluem documentação da arquitetura desatualizada e a falta de comentários de código.
- **Infraestrutura:** refere-se a uma configuração sub-ótima dos processos relacionados com o desenvolvimento, tecnologias, ferramentas de apoio, entre outros fatores. Tal configuração sub-ótima afeta negativamente a capacidade da equipe para produzir um produto de qualidade.
- **Controle de Versão:** refere-se aos problemas no versionamento do código fonte, como contribuições de código desnecessárias.
- **Defeito:** representa defeitos, erros ou falhas encontradas em sistemas de software.

PARA GERENCIAR É NECESSÁRIO CONHECER...

Conceitos Relacionados	Definição	Autores
Razão	Motivação que levou à decisão de comprometer a qualidade do sistema.	(BOB, 2009; FOWLER, 2009; MCCONNELL, 2007; GUO, 2009)
Benefício	Consequências em curto prazo da contração da dívida.	(BOB, 2009; CUNNINGHAM, 1992; FOWLER, 2009; GUO, 2009; MCCONNELL, 2007)
Resultado	Consequência em longo prazo da contração da dívida.	(GUO; SEAMAN, 2011)
Principal	Valor que indica quanto custaria para pagar a dívida no momento.	(BOB, 2009; CUNNINGHAM, 1992; FOWLER, 2009; GUO, 2009; MCCONNELL, 2007)
Juros	Possível penalidade em decorrência da dívida (aumento de esforço ou impacto na produtividade de manutenção).	(BOB, 2009; CUNNINGHAM, 1992; FOWLER, 2009; GUO, 2009; MCCONNELL, 2007)
Ganho ou valor rendido	Valor gerado em resultado dos benefícios alcançados.	(BOB, 2009; FOWLER, 2009)
Rentabilidade	Relação de custo/benefício. Considerando o custo para pagar o principal e o juros, e o valor ganho.	(GUO; SEAMAN, 2011)

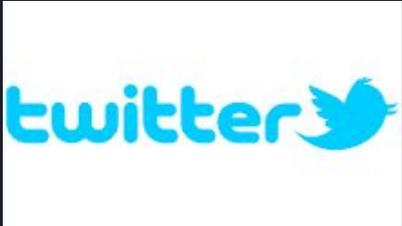
CONSEQUÊNCIA EM CASES REAIS DE DÍVIDA TÉCNICA

- Os seguintes cases mostram o quanto o negligenciamento técnico afeta a existência de muitas empresas.



- Apesar de terem se tornado negócios de sucesso, tiveram sua existência comprometida e contornaram a situação pagando um alto preço.

CONSEQUÊNCIA EM CASES REAIS DE DÍVIDA TÉCNICA



- Inicialmente, construiu o **front-end** de sua plataforma utilizando **Ruby on Rails**, sentiu dificuldades em melhorar o desempenho das pesquisas e adicionar recursos.
- Logo, o substituiu por um servidor Java, que segundo a empresa, **possibilitou a redução de três vezes na latência** da pesquisa.

CONSEQUÊNCIA EM CASES REAIS DE DÍVIDA TÉCNICA



- Krishna Gade, na época Gestor de Engenharia da rede social, publicou em um anúncio no blog do Twitter:

“Com o tempo, também acumulamos dívidas técnicas significativas em nossa base de códigos Ruby, dificultando a adição de recursos e a confiabilidade do nosso mecanismo de pesquisa.”

CONSEQUÊNCIA EM CASES REAIS DE DÍVIDA TÉCNICA



- Desde o início, o Instagram foi uma vítima da dívida técnica.
 - Ao lançar seu aplicativo para iPhone, em outubro de 2010, a operação foi realizada em um único servidor em Los Angeles. Em seis horas, a operação de back-end estava completamente sobrecarregada.
- Porém, logo após um ataque de tráfego quase travar o servidor, o Instagram precisou realizar uma mudança, **em apenas três dias**, para um banco de dados hospedado no Amazon Elastic Compute Cloud (EC2).

CONSEQUÊNCIA EM CASES REAIS DE DÍVIDA TÉCNICA



- Mike Krieger, co-founder do aplicativo, comparou essa transferência a **uma cirurgia de coração aberto**.
- Krieger e o também co-founder, Kevin Systrom, imediatamente viram que seria necessário refazer tudo rapidamente. Principalmente, se houvesse um grande fluxo de pessoas no aplicativo no fim de semana, que na época, se aproximava de **40.000 usuários** – hoje, são 1 bilhão de usuários ativos.
- Hoje o Instagram tem uma política preventiva de DT.

CONSEQUÊNCIA EM CASES REAIS DE DÍVIDA TÉCNICA

The Uber logo, consisting of the word "Uber" in white text on a black square background.

- Em 2016, 57 milhões de passageiros e motoristas do Uber tiveram seus dados violados, como o número de cartão de crédito e outras credenciais.

CONSEQUÊNCIA EM CASES REAIS DE DÍVIDA TÉCNICA

Uber

- A Causa? Dívida técnica
- A empresa cresceu rapidamente e isso levou a decisões de codificação que:
 1. economizaria tempo,
 2. aumentaria sua eficiência momentânea
 3. possibilitaria alterações rapidamente no software.

CONSEQUÊNCIA EM CASES REAIS DE DÍVIDA TÉCNICA

Uber

- Erros se agregaram:
 - Principalmente pelo código ter sido postado no popular GitHub, em texto simples e nada criptografado, possibilitando o ataque.
 - Além disso, não havia sido ativada a autenticação em dois fatores.

CONSEQUÊNCIA EM CASES REAIS DE DÍVIDA TÉCNICA

Uber

- chave de acesso codificada estava em texto simples, os atacantes a usaram para acessar os bancos de dados de software, onde os dados do cliente eram armazenados.



CONSEQUÊNCIA EM CASES REAIS DE DÍVIDA TÉCNICA

Uber

- Conta GitHub invadida e 57 milhões de usuários tiveram dados roubados



Uber

CONSEQUÊNCIA EM CASES REAIS DE DÍVIDA TÉCNICA

- Como comentado pela CSO, a respeito do caso:

“*A realidade é que não é a segurança que gera dinheiro para uma empresa. Mas a falta de segurança pode custar muito à empresa, além de danos institucionais e à reputação. A resposta para essa dívida técnica é incorporar as práticas recomendadas, bem como as ferramentas de segurança adequadas. [...] São as medidas preventivas adotadas que podem eliminar a dívida técnica e manter os negócios e seus clientes seguros.*”

ESTRATÉGIAS PARA GERENCIAR A DT



ESTRATÉGIAS PARA GERENCIAR A DT

- Passo 1

- Inspeção seu código

- Quanto de dívida técnica tem acumulado?

FERRAMENTAS QUE PODEM VERIFICAR A QUALIDADE DO CÓDIGO E ENTÃO, VOCÊ PODE IDENTIFICAR A DT



sonarqube

The logo for Sonarqube features the word "sonarqube" in a bold, black, sans-serif font. To the right of the text is a blue icon consisting of three curved lines that resemble a sonar wave or a signal.

sonarlint

The logo for Sonarlint features the word "sonarlint" in a bold, black, sans-serif font. Below the letters "i" and "n" is a red wavy line that resembles a sonar wave or a signal.

<https://www.sonarlint.org>

ESTRATÉGIAS PARA GERENCIAR A DT

- Passo 2
 - Parar de acumular a DT
 - Começar a pagar a DT

Recomendações

**DÍVIDA TÉCNICA NÃO É QUESTÃO DE SE, MAS
QUANDO E QUANTO...**

**AGIR RÁPIDO REDUZ OS IMPACTOS NO
PROJETO!!**

Recomendações

**UMA DÍVIDA TÉCNICA PODE TER
TRAZER ALGUNS BENEFÍCIOS DESDE
QUE SEJA GERENCIADA COM
RESPONSABILIDADE E CONTROLE...**

Recomendações

- **Planejamento e estratégia:** defina o máximo de detalhes antes do início do projeto.
 - **Conscientização e conhecimento:** todo o time deve conhecer o conceito de Dívida Técnica
 - **Resista à economia de tempo:** “barato sai caro”
 - **Nunca mais ignore a dívida técnica**
-

Recomendações

- **Gerencie e Monitore sua Dívida Técnica**
 - **Crie mecanismos nos seus times para:**
 - **Identificar**
 - **Gerenciar**
 - **Controlar**
 - **Pagar**
-

REFERÊNCIAS

Curtis, B., Sappidi, J., & Szyrkarski, A. (2012). Estimating the size, cost, and types of Technical Debt. 2012 Third International Workshop on Managing Technical Debt (MTD). doi:10.1109/mtd.2012.6226000

Cunningham, W. (1992). The WyCash portfolio management system. ACM SIGPLAN OOPS Messenger, 4(2), 29–30. doi:10.1145/157710.157715

Seaman, Carolyn & Guo, Yuepu. (2011). Measuring and Monitoring Technical Debt. Advances in Computers. 82. 25-46. 10.1016/B978-0-12-385512-1.00002-5.

Li, Z., Avgeriou, P., & Liang, P. (2015). A systematic mapping study on technical debt and its management. Journal of Systems and Software, 101, 193-220.

<https://martinfowler.com/bliki/TechnicalDebt.html> - Technical Debt - Martin Fowler