



SÃO  
PAULO  
TECH  
SCHOOL

# Tópicos da Aula

- Apresentação do Professor;
- Regras Gerais;
- Ementário (Conteúdo da Disciplina);
- Introdução;
- Entender o que é e pra que serve Engenharia de Software;
- UX vs UI;
- Atividade em grupo;

# Engenharia de Software

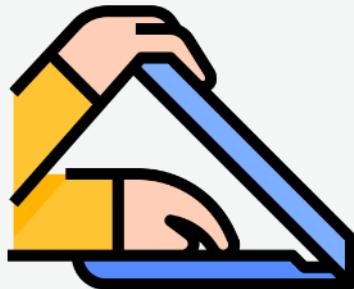
## Aula 1

Introdução a Engenharia de Software

**Fábio Figueiredo**

[fabio.figueiredo@sptech.school](mailto:fabio.figueiredo@sptech.school)

# Regras básicas da sala de aula



- 1. Notebooks Fechados:** Aguarde a liberação do professor;
- 2. Celulares em modo silencioso e guardado**, para não tirar sua atenção
  - Se, caso haja uma situação urgente e você precisar **atender ao celular, peça licença para sair da sala** e atenda fora da aula.
- 3. Proibido usar Fones de ouvido:** São liberados apenas com autorização do professor.
- 4. Foco total no aprendizado**, pois nosso tempo em sala de aula é precioso.
  - Venham sempre com o **conteúdo da aula passada em mente** e as atividades realizadas.
  - Tenham caderno e caneta;
  - **Evitem faltas e procure ir além** daquilo que lhe foi proposto.
  - **Capricho, apresentação e profundidade** no assunto serão observados.
    - “**frequentar as aulas** e demais atividades curriculares aplicando a **máxima diligência no seu aproveitamento**” (Direitos e deveres dos membros do corpo discente - Manual do aluno, p. 31)



# Regras básicas da sala de aula



As aulas podem e devem ser divertidas! Mas:

- **Devemos respeitar uns aos outros** – cuidado com as brincadeiras.
  - “observar e cumprir o regime escolar e disciplinar e comportar-se, dentro e fora da Faculdade, **de acordo com princípios éticos condizentes**” (Direitos e deveres dos membros do corpo discente - Manual do aluno, p. 31)

# Boas práticas no Projeto

## COMPROMISSO



COM VOCÊ:  
**ARRISQUE**, NÃO  
TENHA MEDO DE  
ERRAR



COM OS  
PROFESSORES:  
ORGANIZE A **ROTINA**  
PARA OS ESTUDOS

COM OS COLEGAS:  
**PARTICIPAÇÃO**  
**ATIVA** E PRESENTE



COM O PROJETO:  
**RESPEITO** E  
FLEXIBILIDADE



# Boas práticas no Projeto

Reações **defensivas** não levam ao envolvimento verdadeiro!

Transforme cada problema e cada dificuldade em uma **OPORTUNIDADE** de aprendizado e crescimento.

## EVITE:

- Justificativas e Desculpas
- Transferir a culpa
- Se conformar com o que sabe
- Se comparar com o outro

# Dica: **Como ter sucesso** (*Maiores índices de aprovações*)

## Comprometimento

- Não ter faltas e atrasos. Estar presente (*Não fazer 2 coisas ao mesmo tempo*)
- Fazer o combinado cumprindo os prazos

## Atitudes Esperadas:

- **Professionalismo**: Entender que não é mais ensino médio (*Atitude, comportamento, etc.*)
- **Não estar aqui só pelo** estágio ou pelo diploma
- Não ficar escondido: precisa **experimentar**
- **Trabalhar** em grupo e **participar** na aula
- **Não ser superficial** ou “achar que sabe”
- **Não se enganar** utilizando de “cola”
- Assumir a responsabilidade: Não colocar a culpa em outra coisa. **Não se vitimizar**.



## Intervalo

**Atenção:** Atrasados deverão aguardar autorização para entrar na sala.

## Break

> Pausas durante a aula.

**Obs:** Permanecer no andar, casos específicos me procurar.



AVALIAÇÕES

# Avaliações da Disciplina

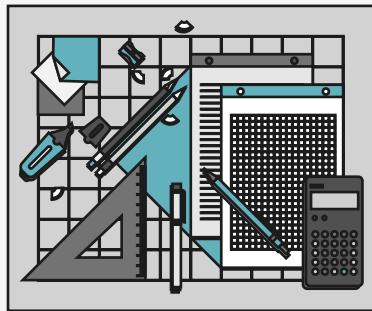
## Avaliação Continuada

- 3 continuadas, sendo 1 por sprint
- Conjunto de atividades que formam a nota de uma continuada
- Representa **40%** da nota final, ou seja, cada nota de continuada vale no máximo 1,33

## Avaliação Integrada

- 1 prova múltipla escolha
- 1 prova dissertativa
- Realizadas no mesmo dia no final do semestre
- Questões interdisciplinares
- Representa **60%** da nota final

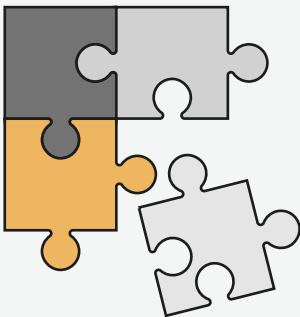
# Formato Continuadas



## Atividades semanais

Entre 2 a 4 **entregas individuais** (atividades)

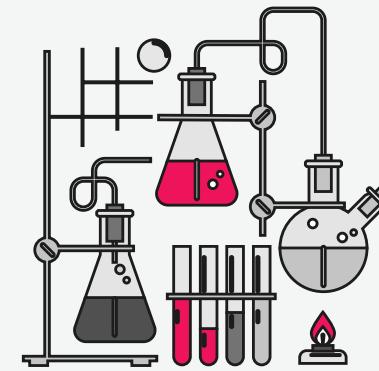
As entregas representam **30%** da nota final da continuada.



## Integradinha (Moodle)

Prova feita no Moodle com **questões interdisciplinares**

A integradinha representa **30%** da nota final da continuada.



## Avaliação Prática

Provas práticas devem ser entregues via Moodle

A prova prática representa **40%** da nota final da continuada.



## Conheça Xampson

- Esse foi o desempenho do Xampson na Sprint 1:
- Entregou **somente um de dois** projetos das atividades semanais devido à má gestão do tempo.
- Correu atrás do “preju” e tirou **8,5 na integradinha**.
- Caiu na zona de conforto e não praticou o suficiente, obteve 6 na prova prática.

**Portanto a nota final de Xampson é...**

# Nota final (Simulação)

	<b>Atividades Semanais (2 atividades)</b>	<b>Integradinha (Moodle)</b>	<b>Avaliação Prática</b>
Peso	<b>30%</b>	<b>30%</b>	<b>40%</b>
Atividade 1	10	8,5	6
Atividade 2	0	--	--
Atividade 3	--	--	--
Atividade 4	--	--	--
Nota final Atividade	<b>1,5</b>	<b>2,55</b>	<b>2,4</b>
<b>Nota final continuada 1</b>		<b>6,45</b>	

## Dicas:

Acreditamos que a nota é reflexo da jornada, portanto:

- São poucas aulas antes de cada prova;
- Tá com dúvida? Pergunte!
- Participe e contribua para uma aula mais rica e com mais exemplos;
- Não deixe para última hora;
- Peça ajuda o quanto antes;
- Colabore com as aulas;
- **Tenha foco e aproveite a jornada!**



# CONTEÚDO DO SEMESTRE





A 3D rendering of a road intersection. The roads are grey with white dashed lines, and the surrounding area is covered in green grass and small trees. The perspective is from an elevated position, looking down at the intersection.

SERÃO MUITOS ASSUNTOS:

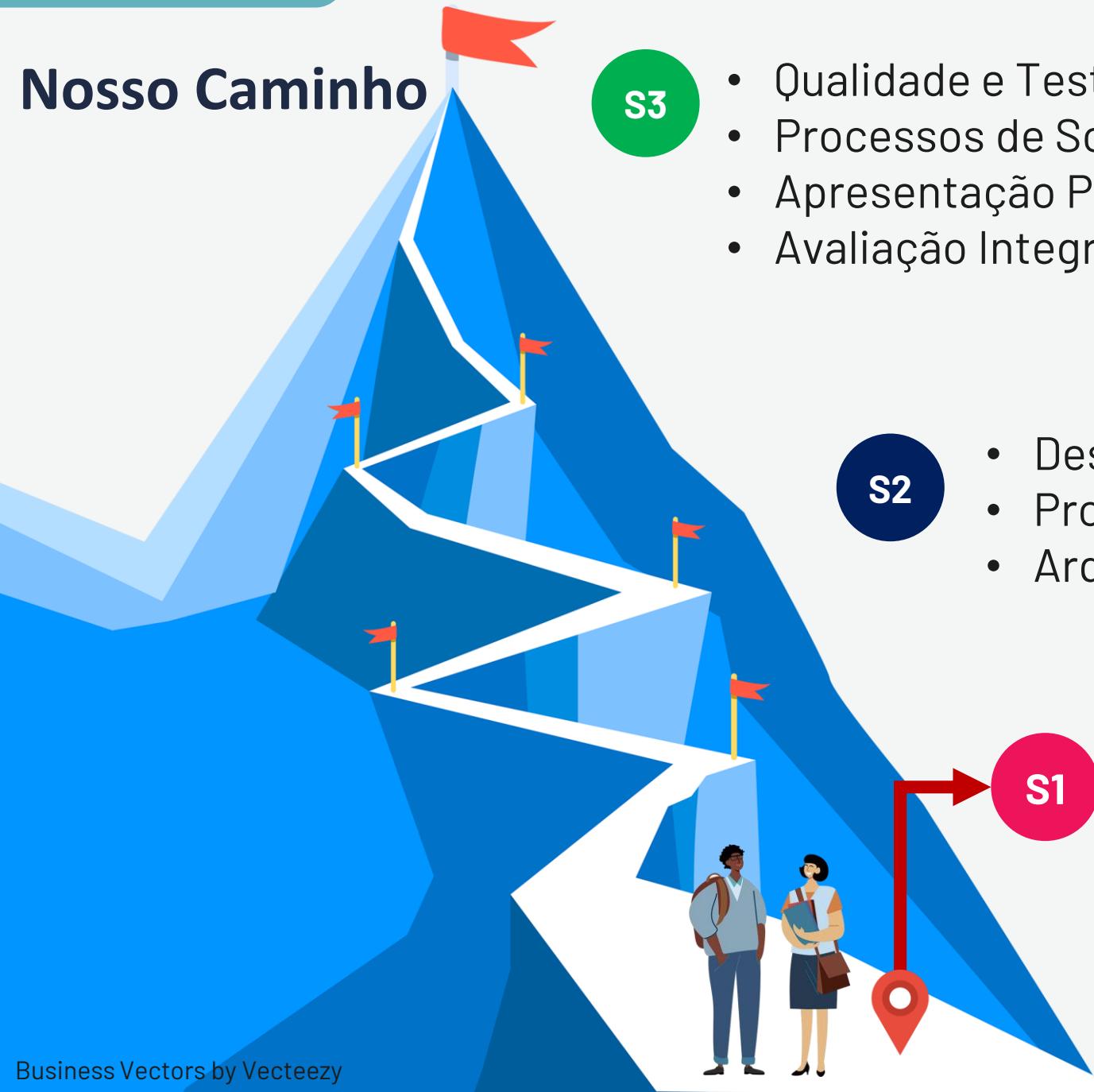
UI / UX PERSONAS

FATORES HUMANOS

ARQUITETURA DE SW

QUALIDADE, PROC...

# Nosso Caminho



S3

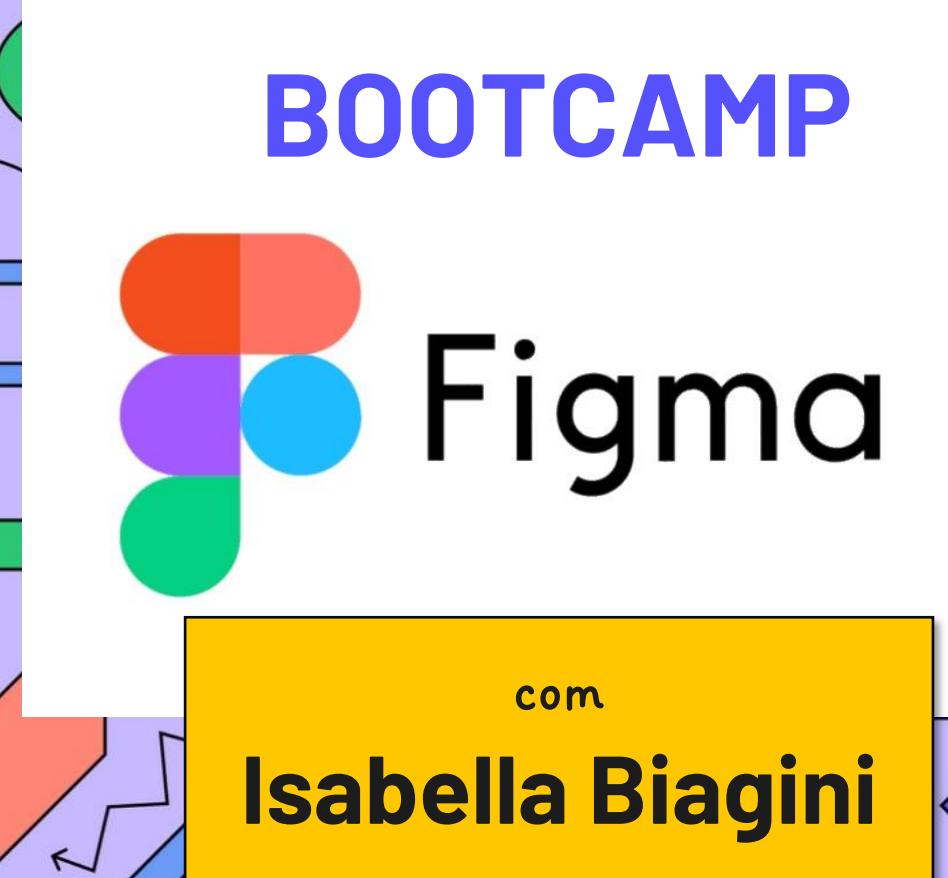
- Qualidade e Testes
- Processos de Software
- Apresentação PI
- Avaliação Integrada

S2

- Design de Interfaces Web
- Projeto de Software
- Arquitetura de Software

S1

- **Introdução a Engenharia de Software**
- **Conceitos de UI e UX**
- Fatores Humanos
- Personas
- Design de Interface Básico
- **Bootcamp de Figma**



# BOOTCAMP



com  
**Isabella Biagini**



## PRÉ REQUISITOS

- Veja os vídeos disponíveis no Youtube da SPTech;
- Bootcamp 100% mão na massa;
- Respeite o processo;
- Tire suas dúvidas.

ACESSE:



[\*\*> Playlist - SPTech Tips - Figma - YouTube\*\*](#)



# BIBLIOGRAFIA

- Engenharia de Software 8º Edição / **Ian Sommerville**
- Engenharia de Software 6º Edição / **Roger Pressman**

## Adicional:

- Code Complete
- SWEBOK
- Entre MUITOS outros que vou informar em tempo de aula.



# O QUE É ENGENHARIA DE SOFTWARE?

# O QUE É ENGENHARIA DE SOFTWARE

- ISO/IEC/IEEE Systems and Software Engineering  
“A aplicação sistemática, disciplinada e quantificáveis abordagens para o desenvolvimento, operação e manutenção de software”
- É uma disciplina de engenharia relacionada a todos os aspectos de produção de software (**Sommerville**)

# O QUE É ENGENHARIA DE SOFTWARE

- Engenharia de software engloba processos, métodos e ferramentas que possibilitam a construção de sistemas complexos baseados em computador dentro do prazo e com qualidade. (**Pressman**)
- en·ge·nha·ri·a (engenho + -aria) substantivo feminino
  - 1. Conjunto de técnicas e métodos para aplicar o conhecimento técnico e científico na planificação, criação e manutenção de estruturas, máquinas e sistemas para benefício do ser humano.
  - 2. Ciência ou arte da construção (ex.: engenharia mecânica, engenharia militar, engenharia naval)

# ETAPAS PARA DESENVOLVIMENTO DE UM SOFTWARE

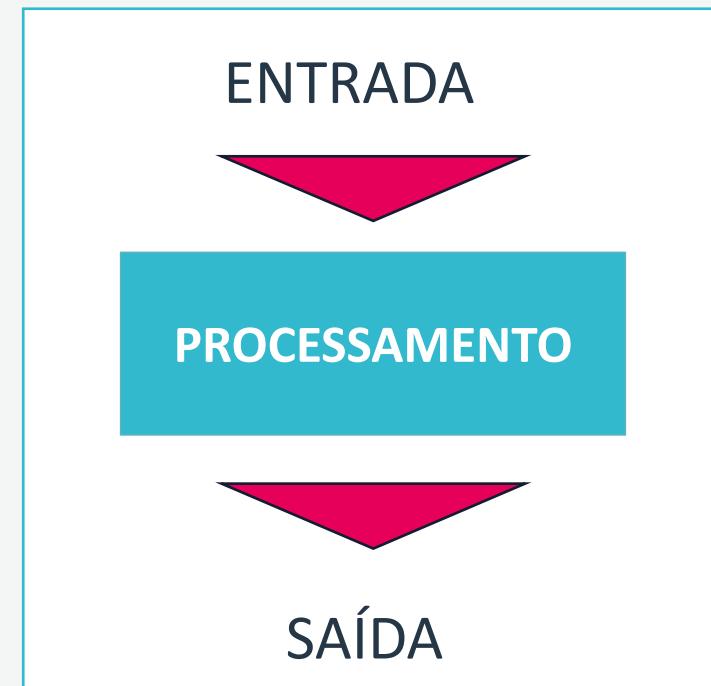
- Ideação;
- Levantamento e Análise de Requisitos;
- Wireframe e Prototipação;
- Documentação Técnica;
  - Arquitetura do Software;
  - Especificação Técnica;
  - ...
- Desenvolvimento;
  - Versionamento de código;
  - ...
- Qualidade - Testes;
  - Unitários;
  - Performance;
  - Aceite do Usuário;
- Implantação;
- Distribuição;
- Manutenção;
- Evolução...

## A NATUREZA DO SOFTWARE

Natureza do software envolve: sua flexibilidade, adaptabilidade, evolução e as dependências de hardware.

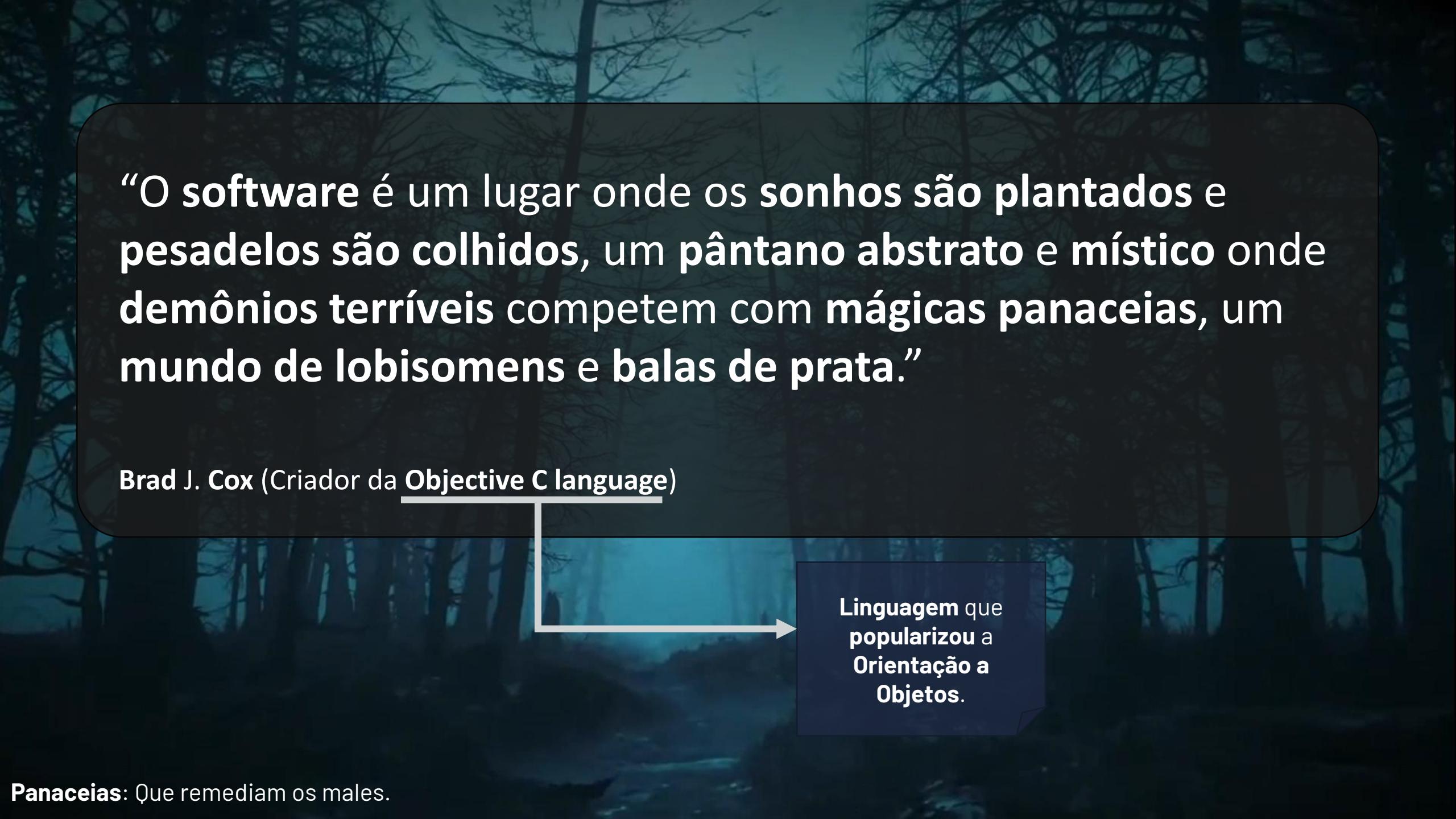
**Software é tanto o produto quanto o meio que distribui o produto.**

> O software distribui o produto  
mais importante destes tempos, a  
**INFORMAÇÃO.**



**“O software é um lugar onde os sonhos são plantados e pesadelos são colhidos, um pântano abstrato e místico onde demônios terríveis competem com mágicas panaceias, um mundo de lobisomens e balas de prata.”**

Brad J. Cox (Criador da Objective C language)

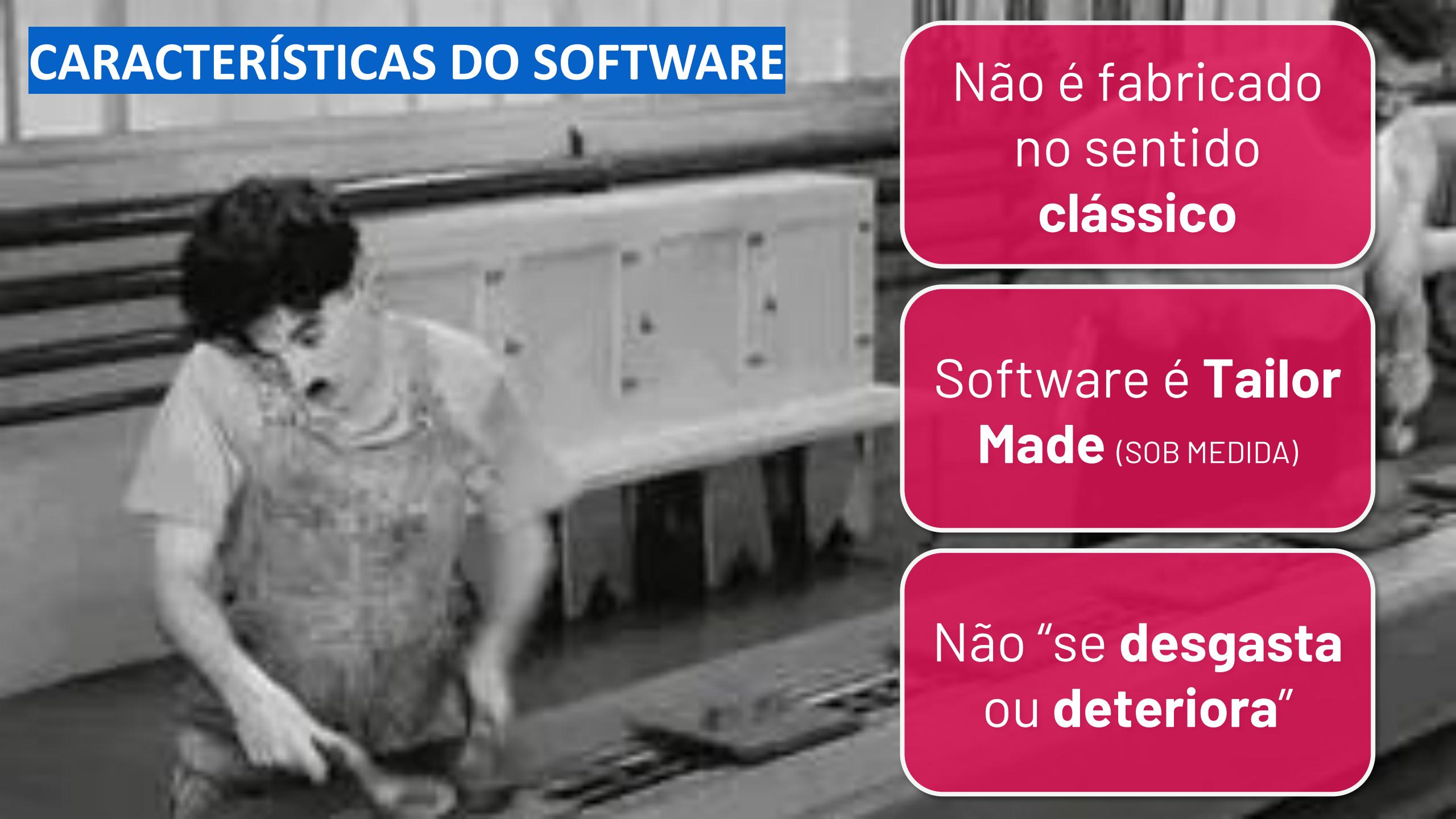


Linguagem que popularizou a Orientação a Objetos.

## NOSSO OBJETIVO NO SEMESTRE

**Aprender/Ensinar** processos, métodos e  
ferramentas para construção e manutenção de  
**Softwares Profissionais.**

# CARACTERÍSTICAS DO SOFTWARE



Não é fabricado  
no sentido  
**clássico**

Software é **Tailor  
Made** (SOB MEDIDA)

Não “se **desgasta**  
ou **deteriora**”

# PROFISSIONAL VS AMADOR VS ARTESANAL



# PROFISSIONAL VS AMADOR VS ARTESANAL

## Amador

1. Que ou o que ama; que ou o que gosta muito de algo ou de alguém. = AMANTE, APRECIADOR.
2. Que ou aquele que, por gosto e não por profissão, exerce qualquer ofício ou arte.
3. Que ou o que revela inexperiência em algum assunto ou .atividade.
4. Que é praticado ou exercido por gosto e não profissionalmente (ex.: desporto amador; teatro amador).

Feito por **pessoas** que **não são profissionais** na área, geralmente **NÃO** é **projeto** ou **desenvolvido** com base em **princípios e técnicas estabelecidas**, e pode **não atender** a **padrões de qualidade e desempenho**.

# PROFISSIONAL VS AMADOR VS ARTESANAL

## Artesanal

1. Relativo a artesanato.
2. Que é fabricado por um artesão.
3. Que é feito sem recursos a meios sofisticados ou a técnicas industriais elaboradas.

**Feito à mão** por uma pessoa ou um pequeno grupo de pessoas. Geralmente é feito **com materiais de alta qualidade e atenção aos detalhes** e **pode ser personalizado** para **atender às necessidades específicas** do usuário.

# PROFISSIONAL VS AMADOR VS ARTESANAL

## Professional

1. Que se relaciona com uma profissão.
2. Pessoa que faz por profissão (ex: Profissionais de Futebol, Profissionais em E-Sports).
3. Que revela profissionalismo (conhecimento).

**Feito por profissionais com experiência e conhecimento na área. Projetado e desenvolvido com base em princípios e técnicas estabelecidas, e atende a padrões de qualidade e desempenho.**

# ENGENHARIA DE SOFTWARE NÃO É PARA AMADORES

---

NOSSO OBJETIVO É O  
SOFTWARE **PROFISSIONAL!**

Englobará todos os processos para a **construção de um software com QUALIDADE.**





# Atividade Rápida

**Em Grupo:**

**1. Para que serve a Engenharia de Software?**

**2. Procurem:**

- 1 Falha de Software;
- 1 Tipo de problema que pode ser evitado com Engenharia de Software;

# GRANDES FALHAS DE SOFTWARE



WEEKLY WORLD

**NEWS**

THE COMPUTER CRASH OF  
THE MILLENNIUM!

JANUARY 1, 2000

IS  
THIS  
THE  
END?

THE DAY THE EARTH  
STANDS STILL!



ALL BANKS WILL FAIL!

FOOD SUPPLIES  
WILL BE DEPLETED!

THE STOCK MARKET  
WILL CRASH!

ELECTRICITY  
WILL BE CUT OFF!

VEHICLES USING  
COMPUTER CHIPS  
WILL STOP DEAD!

TELEPHONES WILL  
CEASE TO FUNCTION!

DEVASTATING WORLDWIDE  
DEPRESSION!



523  
14014 18259 1

WEEKLY WORLD

**NEWS**

THE  
DOOMSDAY  
COUNTDOWN  
FINAL DAYS

ALSO INSIDE

ARMAGEDDON

YEAR 2000 COMPUTER BUG  
will turn machine against man!

Hundreds  
of planes will  
fall out of  
the sky!

Cars will  
stop dead  
in their  
tracks!

Nuclear  
missiles  
will launch  
themselves!

# GRANDES FALHAS DE SOFTWARE

O ano era 1996...



# GRANDES FALHAS DE SOFTWARE

Em 1999...



# GRANDES FALHAS DE SOFTWARE

## Bug do Milênio (1999)

**Custo:** \$600 bilhões

**Desastre:** O desastre de um homem é a fortuna de outro, como demonstra o Bug do Milênio. Empresas gastaram bilhões com programadores para corrigir uma falha no software legado. Embora nenhuma falha significativa tenha ocorrida, a preparação para o Bug do Milênio teve um custo significativo e impacto no tempo em todas as indústrias que usam a tecnologia computacional.

**Causa:** Para economizar espaço de armazenamento de computador, softwares legados muitas vezes armazenavam anos para datas com números de dois dígitos, como 99 para 1999. Esses softwares também interpretavam 00 para significar 1900, em vez de 2000, por isso, quando o ano de 2000 veio, bugs apareceriam.

# GRANDES FALHAS DE SOFTWARE

## 1. Problemas no Mariner (1962)

**Custo:** 18,5 milhões dólares

**Desastre:** Mariner, um foguete com uma sonda espacial para Vênus, foi desviado de seu percurso de voo logo após o lançamento. O controle da missão destruiu o foguete 293 segundos após a decolagem.

**Causa:** Um programador, ao passar para o computador uma fórmula que haviam lhe entregado escrita manualmente, se esqueceu de uma barra. Sem ela, o software tratava variações normais de velocidade como se fossem sérios problemas, causando falhas por tentativas de correções que acabaram por enviar o foguete fora do curso.

## 2. Hartford Coliseu Desmorona (1978)

**Custo:** 70 milhões de dólares, além de outros danos de 20 milhões para a economia local

**Desastre:** Poucas horas depois de milhares de fãs deixarem o Coliseu Hartford, o teto de treliça de aço desabou sob o peso da neve molhada.

**Causa:** O programador do software CAD, utilizado para projetar o coliseu, incorretamente assumiu que o suporte do telhado de aço enfrentaria apenas compressão natural. Mas quando um dos suportes inesperadamente recebeu um bloco de neve, este desencadeou uma reação em cadeia que derrubou o telhado de outras seções como dominós.

# GRANDES FALHAS DE SOFTWARE

## 3. CIA distribui gás aos soviéticos (1982)

**Custo:** Milhões de dólares e danos significativos a economia soviética

**Desastre:** O software de controle se descontrolou e produziu uma intensa pressão no gasoduto Trans-Siberian, resultando na maior explosão não-nuclear da história.

**Causa:** Agentes da CIA supostamente plantaram um bug em um sistema de computador canadense comprado pelos soviéticos para controlar seus gasodutos. A compra fazia parte de um plano estratégico soviético para roubar ou obter secretamente tecnologia sensível dos EUA. Quando a CIA descobriu a compra, eles sabotaram o software para que passasse pela inspeção soviética, mas falhasse na operação.

## 4. 3ª Guerra Mundial (Quase!)(1983)

**Custo:** Quase toda a humanidade

**Desastre:** O sistema de alerta precoce soviético falsamente indicou que os Estados Unidos tinham lançado cinco mísseis balísticos. Felizmente, o oficial de serviço soviético tinha uma “sensação esquisita no estômago” e fundamentalmente, se os EUA estavam realmente atacando, eles lançariam mais de cinco mísseis, por isso ele relatou o aparente ataque como um alarme falso.

**Causa:** Um bug no software soviético falhou ao detectar reflexos solares como falsos mísseis.

# GRANDES FALHAS DE SOFTWARE

## 5. Máquina medicinal mata (1985)

**Custo:** Três mortos e três seriamente feridos

**Desastre:** A máquina de radiação canadense Therac-25 irradiou doses letais em pacientes.

**Causa:** Por causa de um bug sutil chamado de "condição de corrida", um técnico accidentalmente configurou o Therac-25 de modo que o feixe de elétrons seria como um fogo de alta potência.

## 6. Crash na Wall Street (1987)

**Custo:** \$500 bilhões em um dia

**Desastre:** Em 19 de outubro de 1987, o índice Dow Jones caiu 508 pontos, perdendo 22,6% de seu valor total. Esta foi a maior perda que Wall Street já sofreu em um único dia.

**Causa:** Um mercado em grande alta foi interrompido por uma série de investigações conduzidas pela SEC e por outras forças do mercado. Como os investidores fugiram de ações investigadas, um número muito grande de ordens de venda foram gerados pelos computadores, quebrando sistemas e deixando os investidores efetivamente cegos.

## 7. Linhas da AT&T “morrem” (1990)

**Custo:** 75 milhões de ligações perdidas e 200 reservas aéreas perdidas

**Desastre:** Um switch dos 114 centros de switches da AT&T sofreu um problema mecânico que fez com que todo o seu centro fosse desligado. Quando o seu centro voltou a ativa, enviou uma mensagem aos outros, o que causou o desligamento dos outros centros e deixou a empresa parada por 9 horas.

**Causa:** Uma única linha de código em uma atualização de software implementada para acelerar chamadas causou um efeito cascata que desligou a rede.

# GRANDES FALHAS DE SOFTWARE

## 8. Patriot Acaba com Soldados (1991)

**Custo:** 28 soldados mortos e 100 feridos.

**Desastre:** Durante a primeira Guerra do Golfo, um sistema (Patriot) americano de mísseis na Arábia Saudita falhou ao interceptar um míssel vindo do Iraque. O míssel destruiu acampamentos americanos.

**Causa:** Um erro de arredondamento no software calculou incorretamente o tempo, fazendo com que o sistema Patriot ignorasse os mísseis Scud de entrada.

## 9. Pentium Falha em uma Divisão Longa (1993)

**Custo:** \$475 milhões e a credibilidade de uma empresa

**Desastre:** O altamente promovido Pentium, da Intel, ocasionalmente cometeu erros ao dividir números de ponto flutuante em um intervalo específico. Por exemplo, dividindo  $4195835.0/3145727.0$  obteve 1,33374 ao invés de 1,33382, um erro de 0,006%. Embora o bug afetasse apenas alguns usuários, se tornou um pesadelo nas relações públicas. Com uma estimativa de 5 milhões de chips defeituosos em circulação, a Intel se ofereceu para substituir os chips Pentium apenas para os consumidores que poderiam provar que eles precisavam de alta precisão. Contudo a Intel acabou substituindo os chips de qualquer um que reclamou.

**Causa:** O divisor na unidade de ponto flutuante do Pentium tinha uma tabela de divisão falha, faltando cerca de cinco mil entradas, resultando nestes erros de arredondamento.

# GRANDES FALHAS DE SOFTWARE

## 10. Ariane Rocket Goes Boom (1996)

**Custo:** \$500 milhões

**Desastre:** Ariane 5, o mais novo foguete da Europa não-tripulado, fora intencionalmente destruído segundos após seu lançamento em seu vôo inaugural. Também foram destruídos quatro satélites científicos para estudar como o campo magnético da Terra interage com os ventos solares.

**Causa:** O desligamento ocorreu quando o computador de orientação tentou converter a velocidade do foguete de 64-bits para um formato de 16 bits. O número era muito grande, o que resultou em erro de estouro. Quando o sistema de orientação desligou, o controle passou para uma unidade idêntica redundante, que também falhou porque nele estava correndo o mesmo algoritmo.



<https://www.youtube.com/watch?v=1id0IMa4wPw>

# OUTRA FALHA DE SOFTWARE

São Caetano do Sul, 05 de Novembro de 2019.  
COMUNICADO VENDAS GMB KAL 2019



À Rede de Concessionárias Chevrolet  
Att.: Srs. Titular/Operador/Gerente de Vendas e Pós Vendas  
Ref.: Onix Plus MY20 – URGENTE

**IMPORTANTE**  
**Onix Plus MY20 – URGENTE**

A General Motors informa que com o intuito de proteger nossos clientes, tomamos a iniciativa de anunciar um recall, e a suspensão de entrega do modelo Onix Plus aos clientes.

Em condições muito específicas de pressão atmosférica, temperatura ambiente, umidade relativa do ar e composição do combustível, condições combinadas pouco prováveis, o software de gerenciamento do motor pode, eventualmente, apresentar uma falha, com risco de danos ao motor e potencial incêndio. Esta condição é precedida de um alerta visual no painel de instrumentos (luz do motor).

Uma versão atualizada do software de gerenciamento do motor será disponibilizada à Rede de Concessionárias, eliminando o risco.

A fim de minimizar o impacto aos nossos clientes estamos disponibilizando um veículo alugado pelo tempo necessário.

Atenciosamente,

Kleusner Lopes  
Diretor de Vendas

Carlos R. Meinert  
Diretor de Pós Vendas

Leia mais em:

<https://quatrorodas.abril.com.br/noticias/exclusivo-incendio-do-onix-plus-e-causado-por-quebra-do-bloco-do-motor/>

Nada nesta mensagem tem a intenção de ser uma assinatura eletrônica a menos que uma declaração específica do contrário seja incluída nela. Observação de confidencialidade: Esta mensagem é destinada somente à pessoa ou entidade para a qual é endereçada. Pode conter material confidencial e/ou privilegiado. Qualquer revisão, transmissão, divulgação ou outro uso ou qualquer ação tomada por confiança nesta mensagem por pessoas ou entidades que não sejam o destinatário pretendido é proibida e pode ser ilegal. Se você recebeu esta mensagem por engano, entre em contato com o remetente e apague-a do computador.



# TESLA



<https://youtu.be/OLTecQHRTXw>

# IMATURIDADE DE SOFTWARE

## Sintomas da Imaturidade de Software

- 1. As pessoas não recebem o treinamento necessário;**
- 2. Os projetos não são claros** (Não sabe se é manutenção, quem é responsável...);
- 3. As ferramentas não são suficientes** (cada um usa o que quer, não há ferramenta...);
- 4. Os procedimentos e padrões não existem ou existem só no papel (burocrático);**

## Sintomas da Imaturidade de Software

5. **Os compromissos não são realistas** (política, falta de competência, consequência da falta de processos...);
6. **Forças caóticas** (Magias, Balas de prata, Gurus (infalíveis), Heróis (benévolos));
7. **Problemas de escala** (Falta de documentação, ausência de desenho, complexidade desnecessária);
8. **Problemas de comunicação** (Exemplo das bolinhas...mais para frente);

# Perguntas e Respostas - Glossário

O que é Software?	Programas de computador e documentação associada.
Engenharia de Software vs Engenharia de Sistemas	Engenharia de Sistemas engloba todos os aspectos, incluindo hardware e engenharia de processos. Eng. de sistema contém engenharia de software.
O que é um processo de Software?	Um conjunto de atividades cujo objetivo é o desenvolvimento ou evolução de um software. (Especificação, desenvolvimento, validação e evolução).
O que é um modelo de processo de Software?	Representação simplificada do processo de software. Ex: WorkFlow, Fluxo de Dados, Papel/Ação, etc. Modelo de cascata, modelo interativo, modelo ágil, etc.
Qual é o custo da Engenharia de Software	60% desenvolvimento, 40% Teste. Adiante, mais informações.
O que é CASE (computer-aided software engineering)?	“Engenharia de Software auxiliada por computador”. Automação do processo de requisitos, modelagem de sistema, depuração e teste. As vezes até gera o código.

# Perguntas e Respostas - Glossário

O que é o Artefato?	Conjunto de programas, conteúdo (dados), modelos, documentos, o software.
Quais são os desafios-chave da engenharia de software?	Redução do tempo de entrega e desenvolvimento de software digno de confiança.
O que são métodos de engenharia de software?	Abordagens estruturadas para desenvolvimento de software que incluem modelo de sistema, notação, regras, recomendações de projeto e guias de processo.

# Mais definições que iremos repetir muito...

**prag·má·ti·co**

adjetivo

1. Relativo à pragmática ou ao pragmatismo.
2. Que tem motivações relacionadas com a .ação ou com a eficiência. = PRÁTICO  
adjetivo e substantivo masculino
3. Que ou quem revela um sentido prático e sabe ou quer agir com eficácia.

**prag·ma·tis·mo**

(inglês pragmatism)

substantivo masculino

Doutrina que toma por critério da verdade o valor prático e se opõe ao intelectualismo.

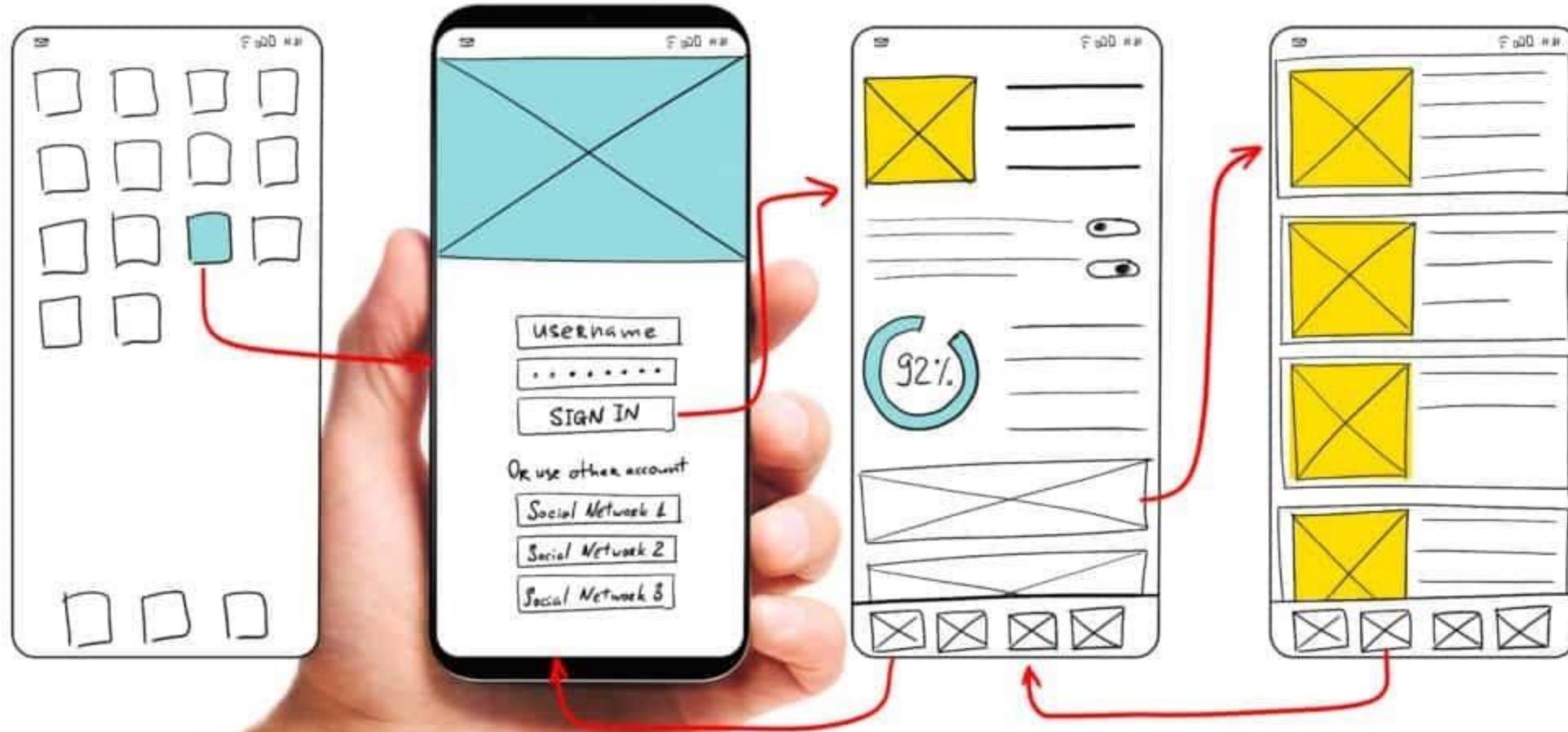
# Kahoot!

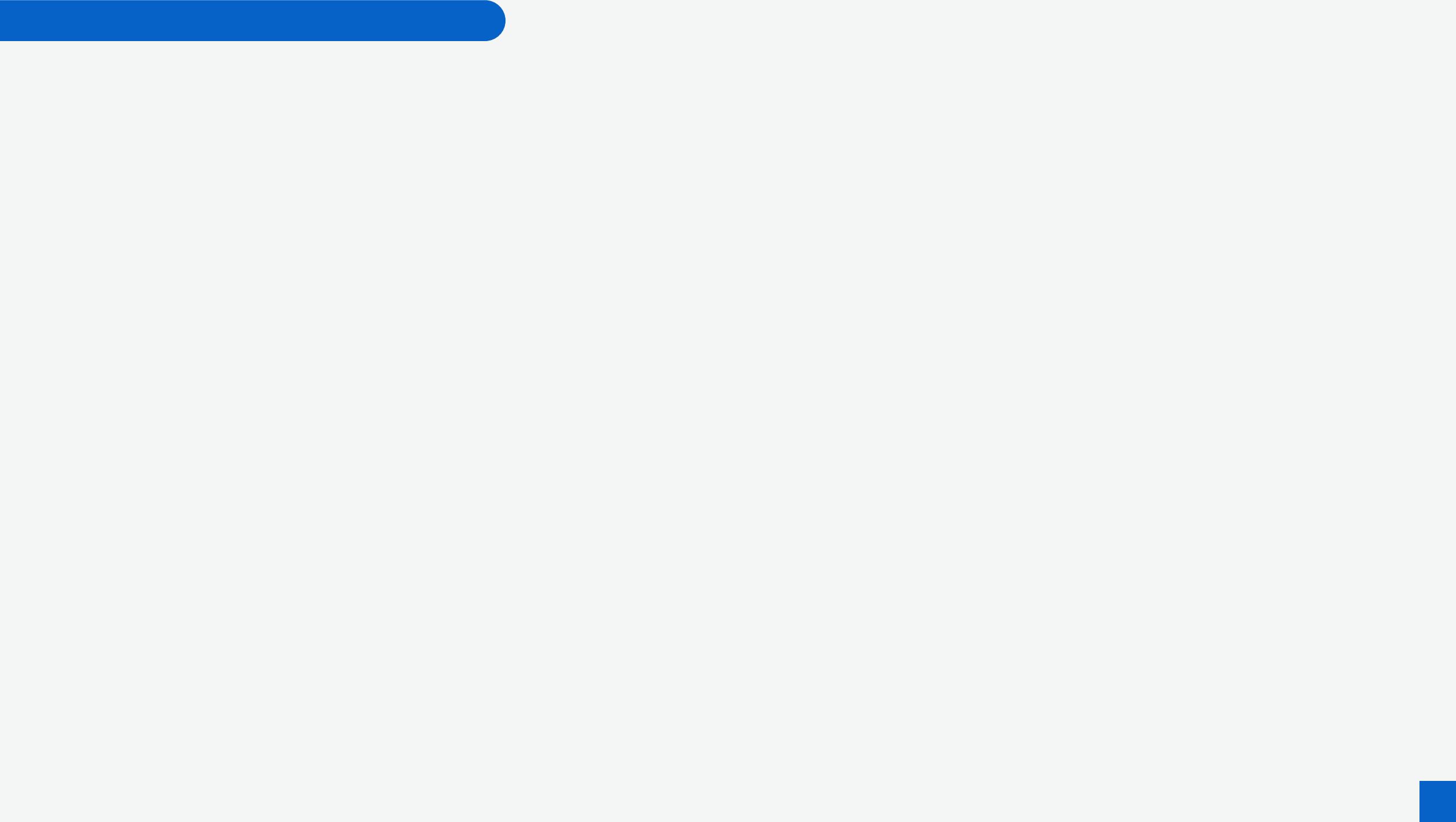


**U I  
U X D E S I G N**



# UX Design



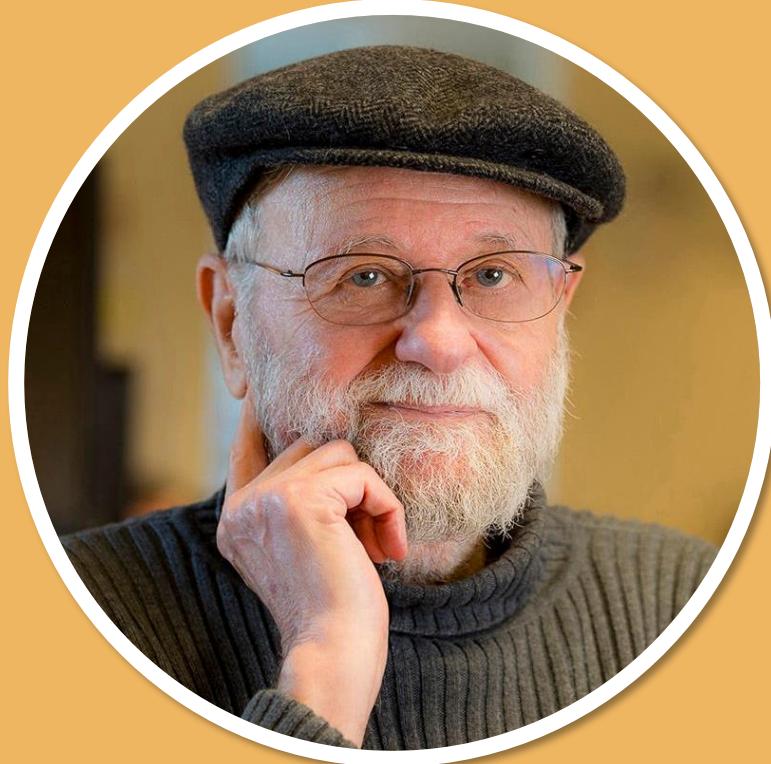


# UX trata de **PESSOAS!**

- Não está relacionado **somente** com **sites, aplicativos e programas**;
- Não está relacionado **somente** com **tecnologia**;
- **UX** é como o **ser humano** se **relaciona** com o **mundo**.



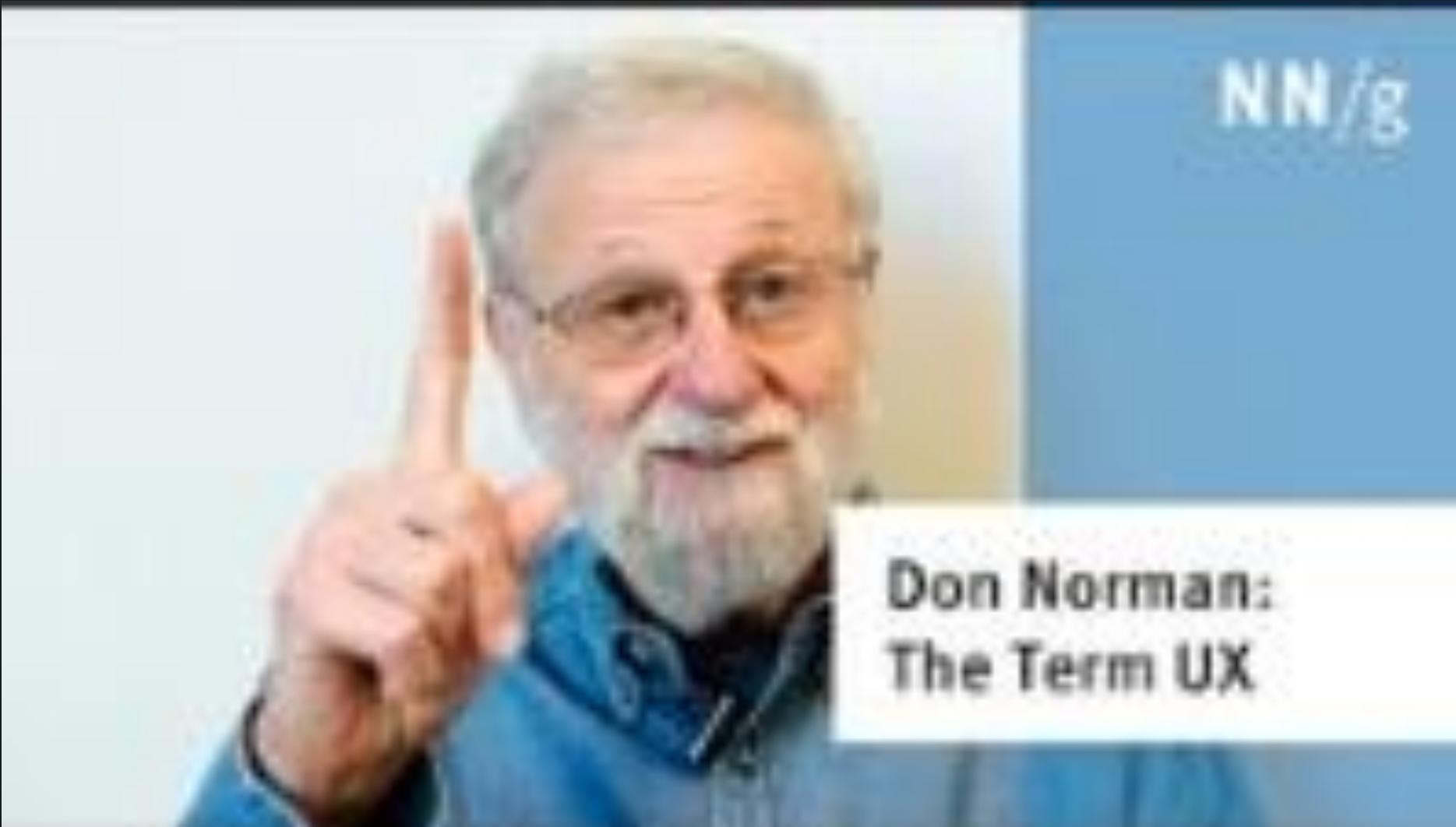
# Jakob Nielsen, Don Norman e o NN/g



NN/g

Nielsen Norman Group

# Donald Norman - O INVENTOR DO TERMO UX



<https://www.youtube.com/watch?v=9BdtGjoIN4E>

# UX aplicada à sua vida!

A Experiência do Usuário está relacionada **com o usuário**, e não com quem faz o projeto.



# Evolução dos PCs da Apple



# Principais Princípios de UX

- 1. Usabilidade – Fáceis de usar, intuitivos e eficientes;**
2. Simplicidade – Livre de complexidades desnecessárias;
3. Consistência – Para que os usuários “prevejam” como usar;
4. Feedback claro – Respostas claras e imediatas aos usuários;
5. Hierarquia visual – Organização e destaque dos itens importantes;
6. Legibilidade – Conteúdo de fácil compreensão;
7. Acessibilidade – Design acessível a todos os usuários (personas);
8. Foco no usuário – Design centrado no usuário;
9. Feedback e Iteração – Avaliação, ajustes e melhorias;
10. Estética – Design visual e atrativo;

# Principais Princípios de UX

1. Usabilidade - Fáceis de usar, intuitivos e eficientes;
- 2. Simplicidade - Livre de complexidades desnecessárias;**
3. Consistência - Para que os usuários “prevejam” como usar;
4. Feedback claro - Respostas claras e imediatas aos usuários;
5. Hierarquia visual - Organização e destaque dos itens importantes;
6. Legibilidade - Conteúdo de fácil compreensão;
7. Acessibilidade - Design acessível a todos os usuários (personas);
8. Foco no usuário - Design centrado no usuário;
9. Feedback e Iteração - Avaliação, ajustes e melhorias;
10. Estética - Design visual e atrativo;

# Principais Princípios de UX

1. Usabilidade - Fáceis de usar, intuitivos e eficientes;
2. Simplicidade - Livre de complexidades desnecessárias;
- 3. Consistência – Para que os usuários “prevejam” como usar;**
4. Feedback claro - Respostas claras e imediatas aos usuários;
5. Hierarquia visual - Organização e destaque dos itens importantes;
6. Legibilidade - Conteúdo de fácil compreensão;
7. Acessibilidade - Design acessível a todos os usuários (personas);
8. Foco no usuário - Design centrado no usuário;
9. Feedback e Iteração - Avaliação, ajustes e melhorias;
10. Estética - Design visual e atrativo;

# Principais Princípios de UX

1. Usabilidade - Fáceis de usar, intuitivos e eficientes;
2. Simplicidade - Livre de complexidades desnecessárias;
3. Consistência - Para que os usuários “prevejam” como usar;
- 4. Feedback claro - Respostas claras e imediatas aos usuários;**
5. Hierarquia visual - Organização e destaque dos itens importantes;
6. Legibilidade - Conteúdo de fácil compreensão;
7. Acessibilidade - Design acessível a todos os usuários (personas);
8. Foco no usuário - Design centrado no usuário;
9. Feedback e Iteração - Avaliação, ajustes e melhorias;
10. Estética - Design visual e atrativo;

# Principais Princípios de UX

1. Usabilidade - Fáceis de usar, intuitivos e eficientes;
2. Simplicidade - Livre de complexidades desnecessárias;
3. Consistência - Para que os usuários “prevejam” como usar;
4. Feedback claro - Respostas claras e imediatas aos usuários;
- 5. Hierarquia visual – Organização e destaque dos itens importantes;**
6. Legibilidade - Conteúdo de fácil compreensão;
7. Acessibilidade - Design acessível a todos os usuários (personas);
8. Foco no usuário - Design centrado no usuário;
9. Feedback e Iteração - Avaliação, ajustes e melhorias;
10. Estética - Design visual e atrativo;

# Principais Princípios de UX

1. Usabilidade - Fáceis de usar, intuitivos e eficientes;
2. Simplicidade - Livre de complexidades desnecessárias;
3. Consistência - Para que os usuários “prevejam” como usar;
4. Feedback claro - Respostas claras e imediatas aos usuários;
5. Hierarquia visual - Organização e destaque dos itens importantes;
- 6. Legibilidade - Conteúdo de fácil compreensão;**
7. Acessibilidade - Design acessível a todos os usuários (personas);
8. Foco no usuário - Design centrado no usuário;
9. Feedback e Iteração - Avaliação, ajustes e melhorias;
10. Estética - Design visual e atrativo;

# Principais Princípios de UX

1. Usabilidade - Fáceis de usar, intuitivos e eficientes;
2. Simplicidade - Livre de complexidades desnecessárias;
3. Consistência - Para que os usuários “prevejam” como usar;
4. Feedback claro - Respostas claras e imediatas aos usuários;
5. Hierarquia visual - Organização e destaque dos itens importantes;
6. Legibilidade - Conteúdo de fácil compreensão;
- 7. Acessibilidade - Design acessível a todos os usuários (personas);**
8. Foco no usuário - Design centrado no usuário;
9. Feedback e Iteração - Avaliação, ajustes e melhorias;
10. Estética - Design visual e atrativo;

# Principais Princípios de UX

1. Usabilidade - Fáceis de usar, intuitivos e eficientes;
2. Simplicidade - Livre de complexidades desnecessárias;
3. Consistência - Para que os usuários “prevejam” como usar;
4. Feedback claro - Respostas claras e imediatas aos usuários;
5. Hierarquia visual - Organização e destaque dos itens importantes;
6. Legibilidade - Conteúdo de fácil compreensão;
7. Acessibilidade - Design acessível a todos os usuários (personas);
- 8. Foco no usuário – Design centrado no usuário;**
9. Feedback e Iteração - Avaliação, ajustes e melhorias;
10. Estética - Design visual e atrativo;

# Principais Princípios de UX

1. Usabilidade - Fáceis de usar, intuitivos e eficientes;
2. Simplicidade - Livre de complexidades desnecessárias;
3. Consistência - Para que os usuários “prevejam” como usar;
4. Feedback claro - Respostas claras e imediatas aos usuários;
5. Hierarquia visual - Organização e destaque dos itens importantes;
6. Legibilidade - Conteúdo de fácil compreensão;
7. Acessibilidade - Design acessível a todos os usuários (personas);
8. Foco no usuário - Design centrado no usuário;
- 9. Feedback e Iteração - Avaliação, ajustes e melhorias;**
10. Estética - Design visual e atrativo;

# Principais Princípios de UX

1. Usabilidade - Fáceis de usar, intuitivos e eficientes;
2. Simplicidade - Livre de complexidades desnecessárias;
3. Consistência - Para que os usuários “prevejam” como usar;
4. Feedback claro - Respostas claras e imediatas aos usuários;
5. Hierarquia visual - Organização e destaque dos itens importantes;
6. Legibilidade - Conteúdo de fácil compreensão;
7. Acessibilidade - Design acessível a todos os usuários (personas);
8. Foco no usuário - Design centrado no usuário;
9. Feedback e Iteração - Avaliação, ajustes e melhorias;
- 10. Estética - Design visual e atrativo;**

## Consciência de Mundo

As coisas tem um **motivo**, as vezes você não tem todas as informações, mas **através** do seu **conhecimento prévio**, você **consegue utilizar...**

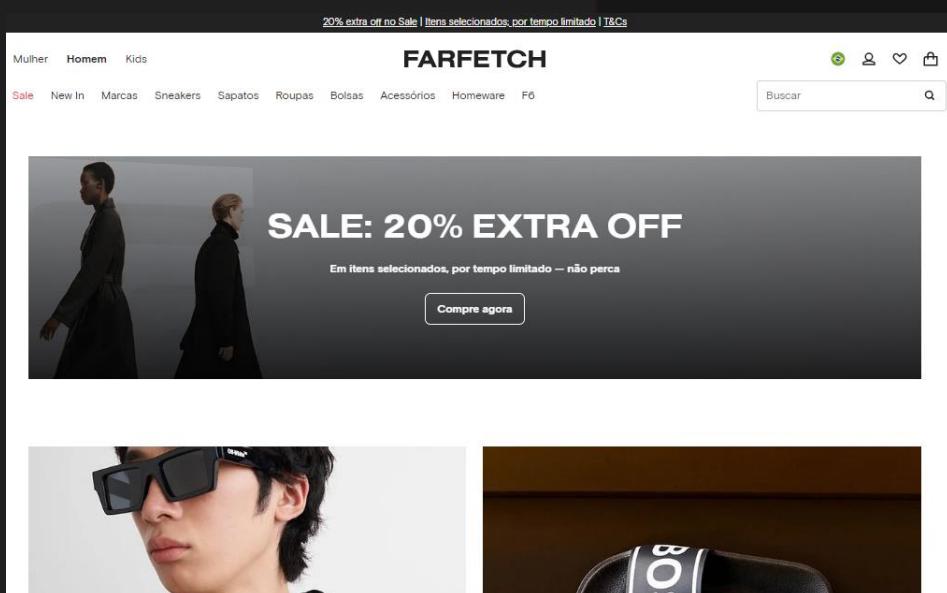
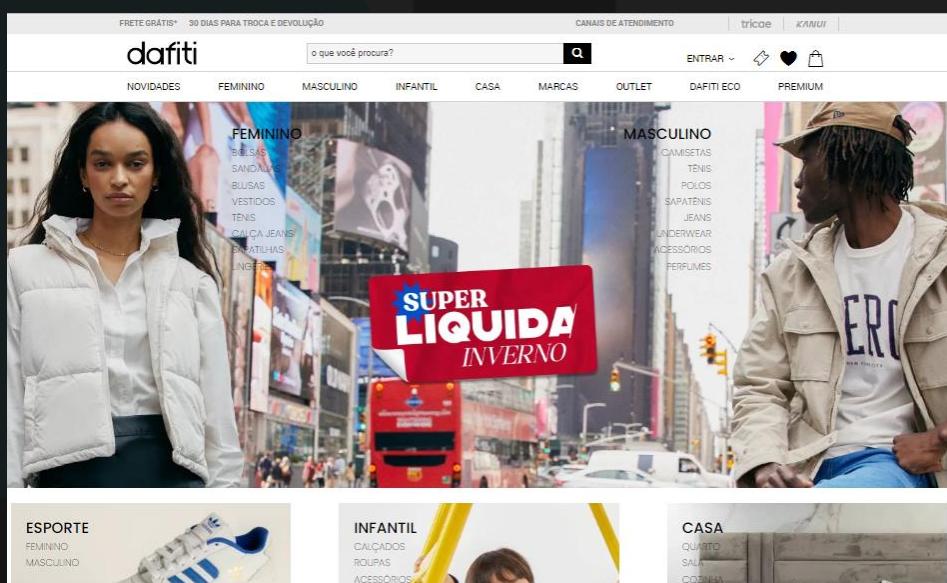
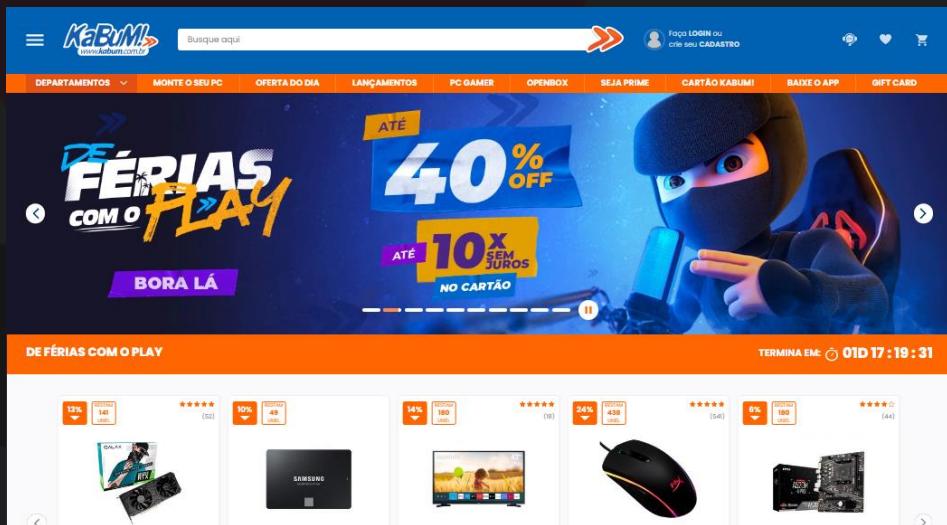


# Convenções

Onde ficam os **pratos** e **copos**? E os **talheres**?



# **Convenções - Funciona da mesma forma nas Interfaces**



# Convenções - Funciona da mesma forma nas Interfaces

The screenshot shows the AliExpress mobile website interface in Arabic. At the top, there are promotional banners for 'الحصول على التطبيق' (Get the app), 'الأمان والخصوصية' (Safety and Privacy) with a checkmark icon, and 'دفع آمن' (Safe payment) with a dollar sign icon. Below the banner is the AliExpress logo and a search bar with a shopping cart icon showing '0' items.

The main navigation menu includes tabs for 'Free Shipping', 'Supermarket', 'DO BRASIL', and 'Plus'. On the left side, there's a welcome message 'مرحباً بك في AliExpress' (Welcome to AliExpress) with ' تسجيل الدخول' (Log in) and ' تسجيل' (Sign up) buttons. A yellow banner below it promotes 'Free Shipping Via chartered flights to Brazil'.

The central part of the screen features a large promotional image for 'الألعاب والهوايات' (Toys and Hobbies) with a 30% discount offer. To the right, there's a sidebar titled 'المفاسد' (Categories) listing various product categories with corresponding icons:

- أزياء نسائية عصرية (Women's modern clothing)
- أزياء رجالية عصرية (Men's modern clothing)
- الهواتف و الاتصالات (Telephones and communications)
- الكمبيوتر، المكتب & الأمان (Computer, office & security)
- إلكترونيك المسهولة (Easy-to-use electronics)
- الحلويات & الساعات (Sweets & watches)
- مسكرمات المنزل، الحيوانات الأليفة ... (Home decorations, pets ...)
- الحقائب & الأحذية (Bags & shoes)
- الألعاب ، الأطفال والرضع (Games, children and infants)
- المرح و رياضات الأماكن المفتوحة (Fun and outdoor sports)

At the bottom, there are several small promotional cards for products like '17KM' and 'Get a Bonus Product or ...'.

**VAMOS ANALISAR ALGUMAS  
IMAGENS...**

# IDENTIFIQUE OS ASPECTOS DE USABILIDADE



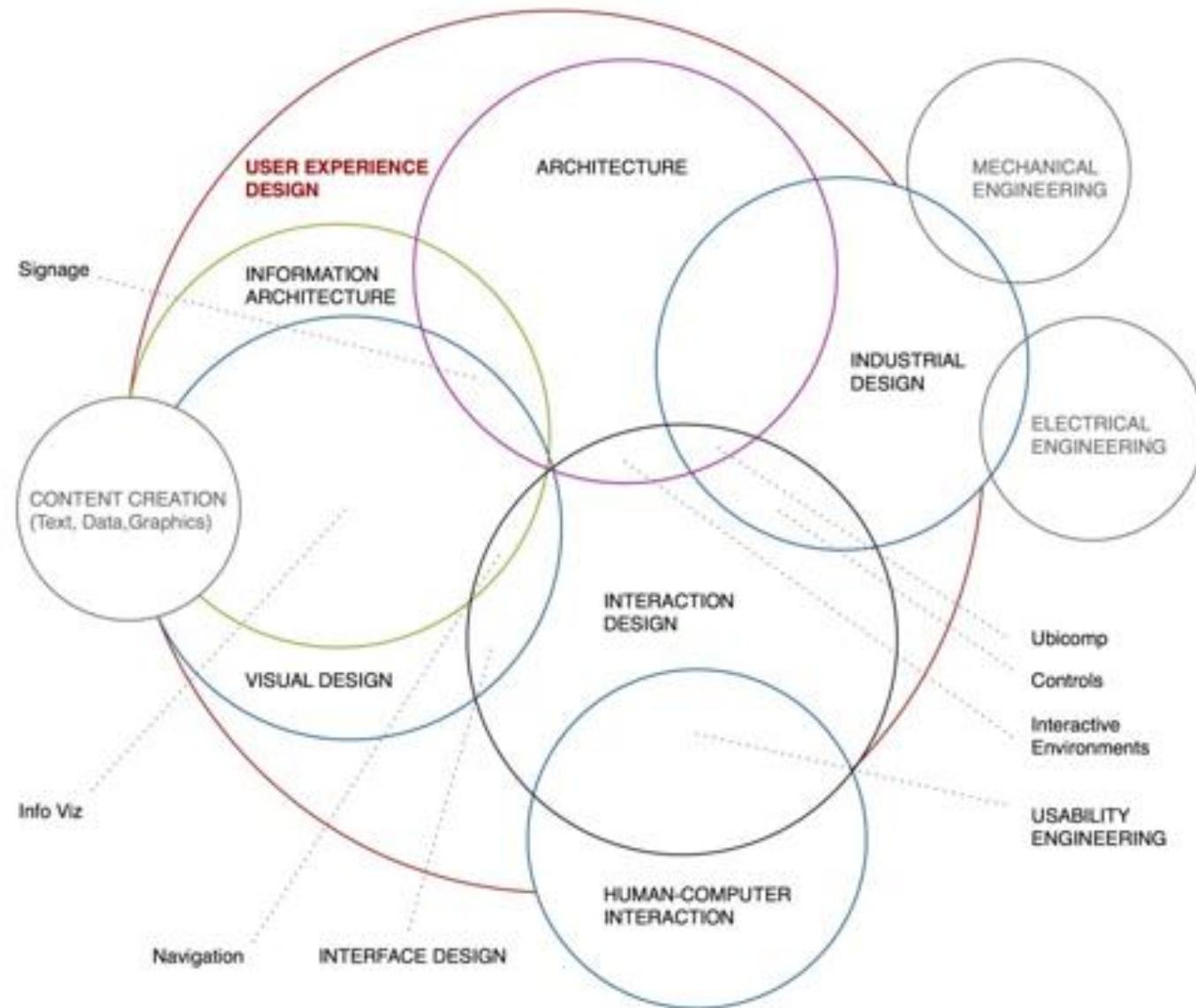
# IDENTIFIQUE OS ASPECTOS DE USABILIDADE



# IDENTIFIQUE OS ASPECTOS DE USABILIDADE



# VISÃO GERAL - UX



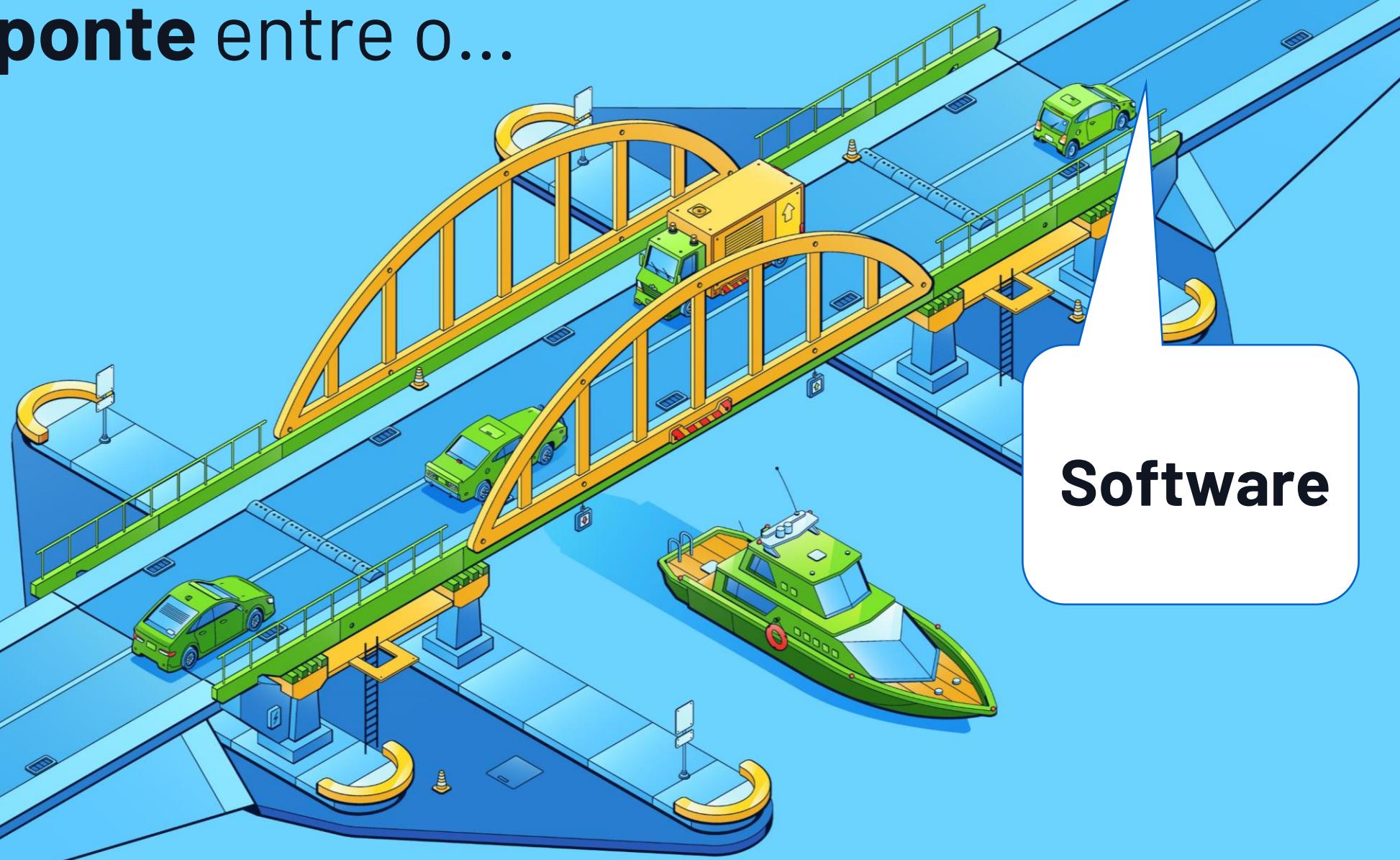
# UI está relacionado com o Visual



# UI e UX é a ponte entre o...

Usuário

Software



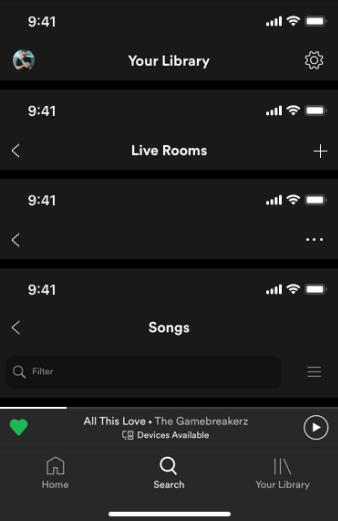


UI Kit

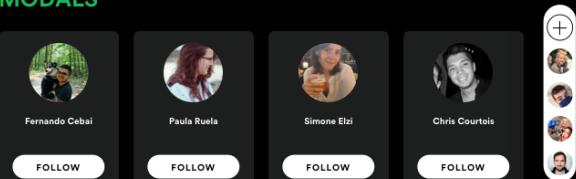
## COLOR PALETTE



## NAVIGATION



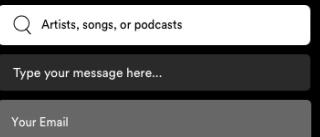
## MODALS



## BUTTONS



## INPUTS



## IMAGES



## TYPOGRAPHY

The quick brown fox jumps over the lazy dog

Circular Bold, 21pt

The quick brown fox jumps over the lazy dog

Circular Bold, 15pt

The quick brown fox jumps over the lazy dog

Circular Book, 13pt

The quick brown fox jumps over the lazy dog

Circular Bold, 10pt

The quick brown fox jumps over the lazy dog

Circular Book, 10pt

The quick brown fox jumps over the lazy dog.

The quick brown fox jumps over the lazy dog.

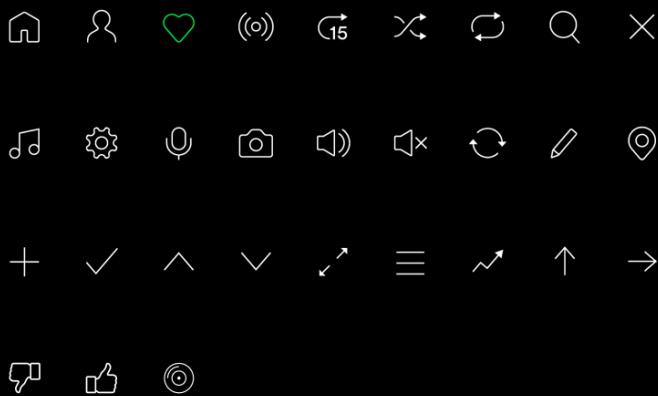
The quick brown fox jumps over the lazy dog.

The quick brown fox jumps over the lazy dog.

This is an example of a paragraph.

Circular Book, 13pt, Line 20, Aligned Center

## ICONS



Home

Search

Your Library

Playlists Podcasts & Shows Albums

Liked Songs Matt Suda: JAMS Matt Suda: Collection Favorite Playlists

This Is Linkin Park When I Close My Eyes How To Be Human Chelsea Cutler Artist Pharmacy This Is Galantis EDM Songs Pop Songs Rock Songs This Is Tate McRae This Is NOTD This Is Selena Gomez ARIZONA Boy In Space Pop Rising This Is ARIZONA New Music Friday Dance This Is Chelsea Cutler This Is Boy In Space Lost Linkin Park

Home Search

Your Library

Playlists By you By Spotify

Liked Songs Matt Suda: JAMS Matt Suda: Collection Favorite Playlists Local Files This Is Chelsea Cutler Spotify This Is Boy In Space Spotify This Is ARIZONA Spotify Pop Rising Spotify New Music Friday Dance Spotify This Is Linkin Park If I Hadn't Met You Chelsea Cutler

Home Search

Your Library

Playlists Podcasts & Shows Albums Artists Downloaded

Liked Songs Matt Suda: JAMS Matt Suda: Collection Favorite Playlists

ARIZONA Boy In Space Chelsea Cutler When I Close My Eyes

THIS IS Linkin Park THIS IS ARIZONA THIS IS Chelsea Cutler THIS IS Boy In Space

Moving On ARIZONA

Recent searches

Chelsea Cutler Quinn XCII

Browse all

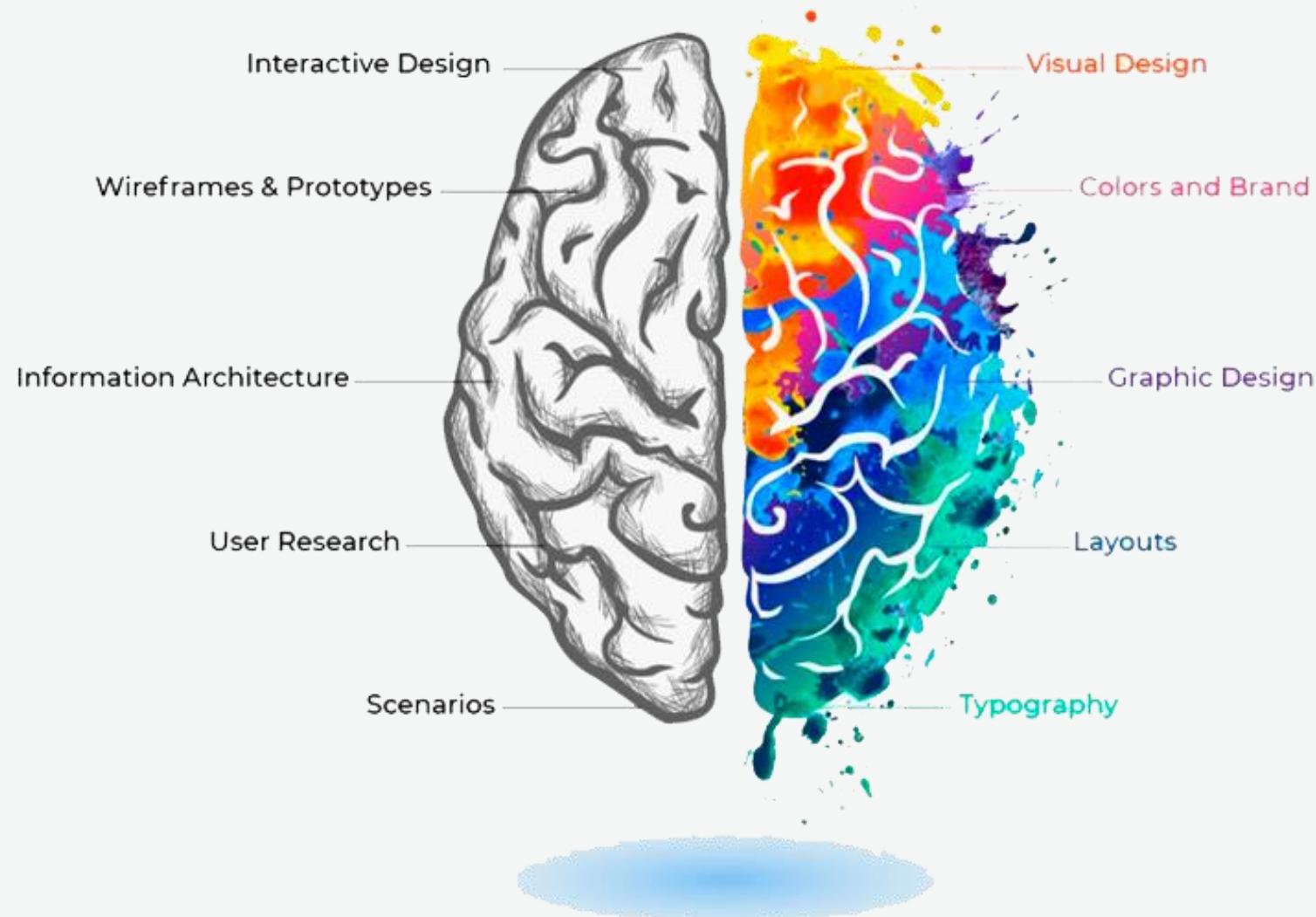
Podcasts Audiobooks

Hip-Hop Pop

Rock Dance/Electro

# UX vs UI

# UX vs UI



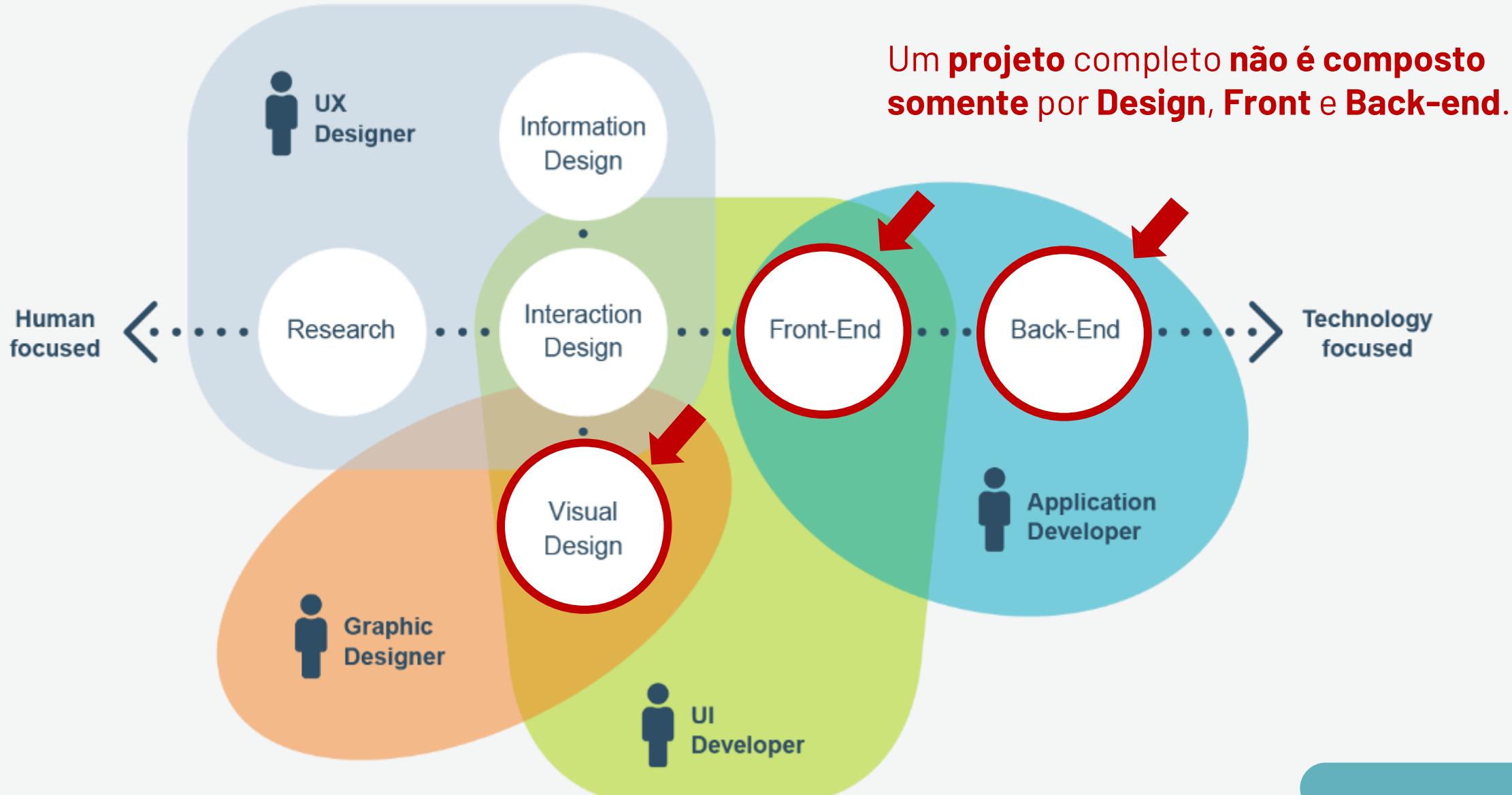
- Cores
  - Ilustrações / Ícones
  - Fotografia
  - Tipografia (fontes)
  - Design Visual / Gráfico
- 
- Psicologia
  - Testes de Usabilidade
  - Design de Interação
  - Estratégia de Conteúdo
  - Protótipos e **Wireframes**
  - Arquitetura da Informação
  - Pesquisa com usuários
  - Definição de Personas



UI

UX

# UX vs UI



# UX vs UI

## Formulário de Cadastro

### UX

- Qual o objetivo desse cadastro?
- Quem vai preencher esse cadastro?
- Deveria ser um formulário de cadastro?
- Qual a melhor forma desse usuário preencher esse cadastro?

### UI

- Qual tipografia deverá ser utilizada?
- Quais as cores?
- Qual o formato dos inputs?
- Qual será a diagramação nos diferentes dispositivos?

# UX vs UI

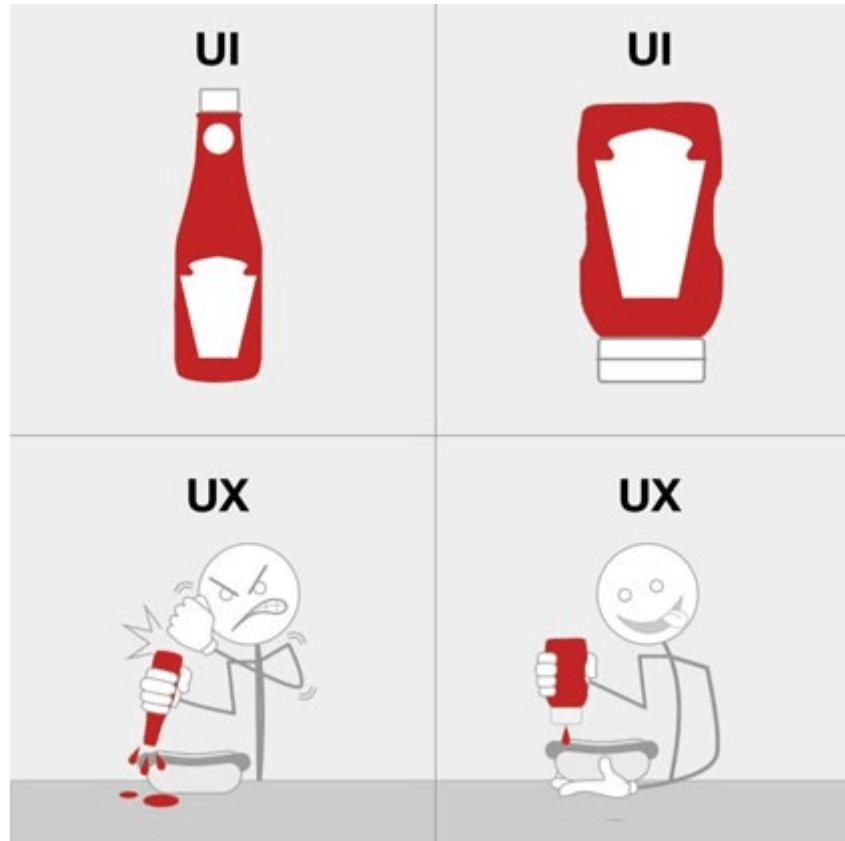


# UX vs UI

UI



UX





Frase dessa Sprint:

**Talvez sua opinião seja  
importante.**



Palavra-chave dessa Sprint:

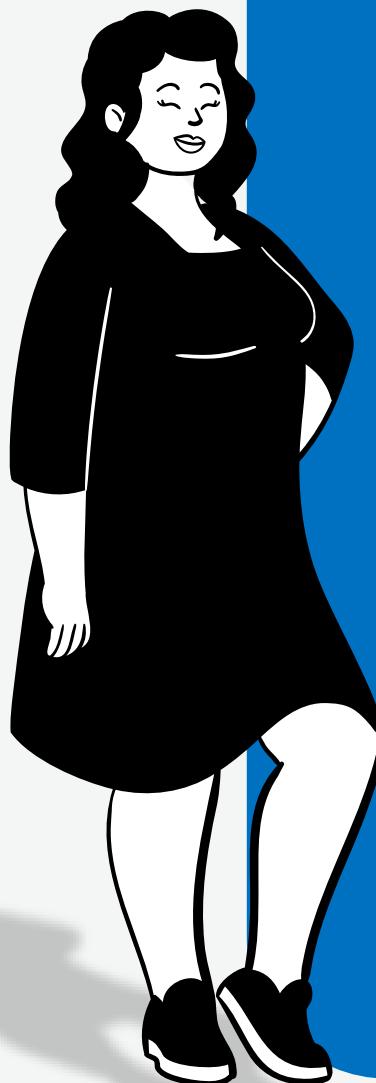
**EMPATIA!**

# ATIVIDADE

# METAS DE INTERFACE

**NOSSA!**  
Que legal.

- Útil
- Satisfatório
- Agradável
- Atraente
- Prazeroso
- Emocionante/Excitante
- Interessante
- Humanizado
- Motivador
- Desafiador
- Social
- Divertido
- Recompensador



**CREDO!**



- Não cumpre /Não resolve
- Tedioso
- Faz você se sentir culpado
- Infantil
- Desprazeroso
- Faz você se sentir estúpido
- Paternalizar (Te trata como uma criança)
- Forçosamente bonito
- Artificial / Falso
- Inconsistente

**Agradeço  
a sua atenção!**

**Fábio Figueiredo**

[fabio.figueiredo@sptech.school](mailto:fabio.figueiredo@sptech.school)

SÃO  
PAULO  
TECH  
SCHOOL