



**Compreender os principais conceitos de
desenvolvimento**



Aula 1 – Principais tipos de linguagem de programação

Aula 2 – Principais tipos de aplicações

Aula 3 – Conceitos de front-end e back-end

Aula 4 – Metodologias de desenvolvimento

Aula 5 – Conceitos básicos de banco de dados

Aula 01

Principais tipos de linguagem de programação



Linguagem de programação

Linguagem de Programação

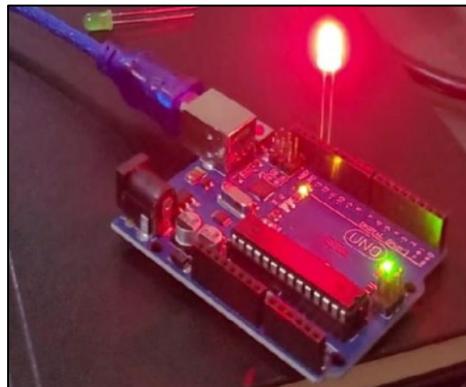
A linguagem de programação é uma ferramenta que segue um conjunto de instruções, permitindo que sejam escritas ações que construam programas de computador que controlam aspectos físicos e lógicos de máquinas.

Como assim?

É uma linguagem de comunicação entre o programador e o computador, onde é possível especificar:

- os dados em que o software deve receber;
- como esses dados serão tratados, transmitidos e armazenados;
- ações que o software deve executar, de acordo com as necessidades específicas.

Aspectos físicos



```
>>> import math
>>> a = math.fsum([1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
>>> b = math.fsum((10, 20, 30, 40, 50))
>>> c = math.fsum({1.1, 2.2, 3.3, 4.4, 5.5})
>>> print(a)
55.0
>>> print(b)
150.0
>>> print(c)
16.5
```

Aspectos lógicos

Linguagem de Programação



Resumindo:

A linguagem de programação é um sistema de comunicação, composto por vários símbolos, regras sintáticas e semânticas que permitem que o programador e a máquina se comuniquem.

Exemplo de regras sintáticas (Javascript)

```
if (condition_1) {  
    // statements  
} else if (condition_2) {  
    // statements  
} else {  
    // statements  
}
```

Exemplo de Semântica (HTML)

```
1  <!DOCTYPE html>  
2  <html lang="en">  
3  <head>  
4      <meta charset="UTF-8">  
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">  
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">  
7      <title>PRIMER RETO</title>  
8  </head>  
9  <body>  
10     <section>  
11         <h1>Platzi </h1>  
12         <p>Tiene una comunidad</p>  
13         <ul>  
14             <li>Maravillosa</li>  
15             <li>Increible</li>  
16         </ul>  
17     </section>  
18     <footer>Iniciando en el Desarrollo WEB</footer>  
19  </body>  
20  </html>
```

Linguagem de programação x Linguagem de computação

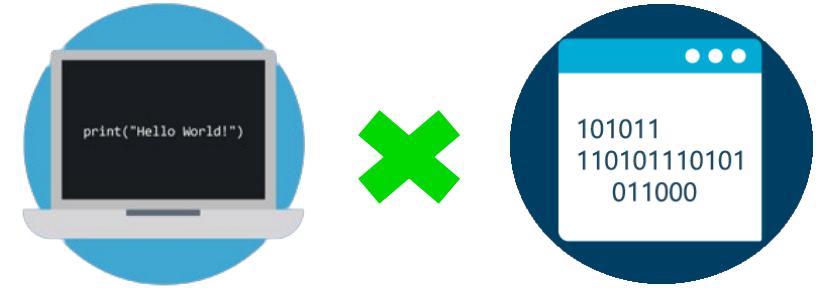
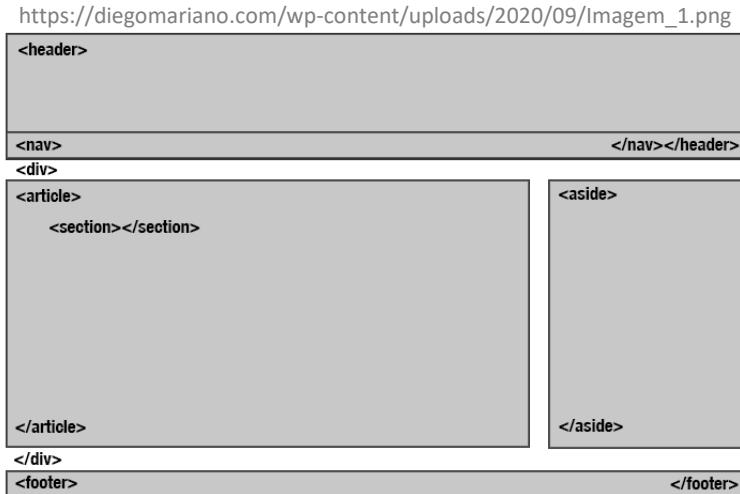
A linguagem de programação segue um conjunto de regras e instruções que serão interpretadas pelo programador.

A linguagem de computação inclui outras linguagens que formatam um texto, mas não é exatamente uma programação em si mesma.

Logo, nem todas as linguagens computacionais são de programação, mas todas as linguagens de programação são computacionais.

Exemplo de linguagem computacional que não são de programação:

HTML (Hypertext Markup Language):

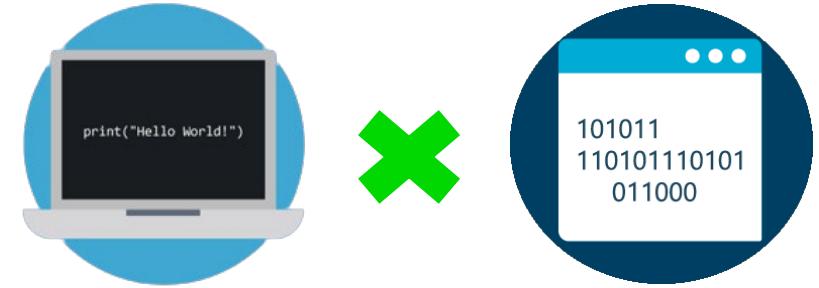




Tipos de linguagem de programação

Tipos de linguagem de programação

As linguagens de programação são classificadas em dois tipos principais: linguagem de baixo nível e linguagem de alto nível.



Linguagem de programação de baixo nível

É a linguagem que se comunica diretamente com a máquina.

É a forma de interpretação entre hardware e software.

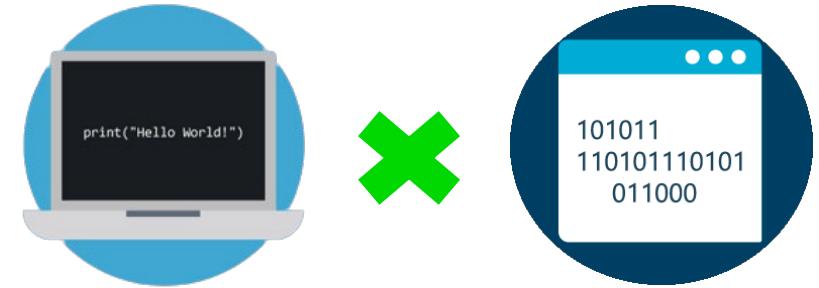
Tem como objetivo, exercer controle sobre o equipamento e sua estrutura física. Para a sua utilização é fundamental que o programador entenda de hardware.

Tipos de linguagem de programação

Linguagem de máquina

É a linguagem mais primitiva e é uma coleção de dígitos ou bits binários (0 e 1) que o computador lê e interpreta e é dessa forma que os computadores entendem.

Exemplo: 10110110



Linguagem Assembly

A linguagem Assembly surgiu para aproximar a linguagem de máquina da usada por seres humanos.

Um programa desenvolvido em Assembly é armazenado como texto e é composto por instruções que comandam as ações de um microprocessador.

Exemplo: MOV AL, 25 ("MOV" é um comando a ser executado que atribui o valor 25 ao registro "AL")

Tipos de linguagem de programação

Linguagem de programação de alto nível

Estas linguagens usam instruções mais fáceis, permitindo que você escreva códigos usando linguagem humana, traduzindo em seguida para a máquina, usando por tradutores ou compiladores.



A linguagem de alto nível facilita o uso de comandos e instruções dado à máquina. Enquanto o programador escreve seus códigos no idioma conhecido, a máquina os recebe traduzidos para a linguagem de máquina.

Tipos de linguagem de programação

Tradutor

Traduz programas escritos em linguagem de programação para linguagem de máquina e executam à medida que são traduzidos.



Compilador

Permite a tradução de um programa inteiro de uma única vez, agilizando o processo e pode ser armazenado para uso futuro sem necessidade de nova tradução.

Tipos de linguagem de programação

As vantagens das linguagens de alto nível são:

- Redução do tempo de programação;
- Atarefa a ser executada é entendida mais facilmente;
- Oprogramador não necessita atuar na operação interna da máquina.





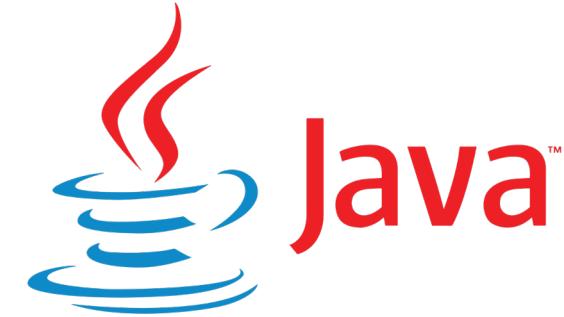
Linguagens mais utilizadas

Linguagens de programação mais usadas:

Java

Muitas empresas têm sistemas desenvolvidos em Java, a busca por profissionais que saibam a linguagem é considerável, tem uma grande comunidade de desenvolvedores, e é uma linguagem portável e escalável.

É a linguagem base dos sistemas Android.



```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello, World!");  
    }  
}
```

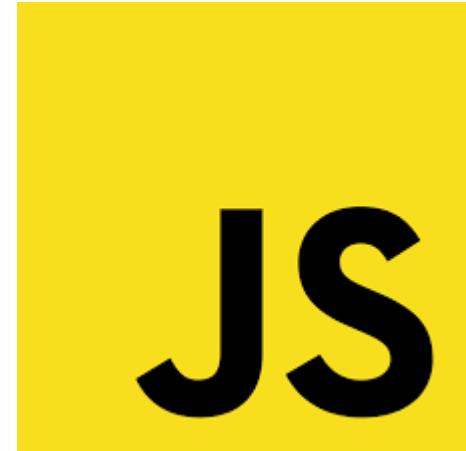
```
Hello, World!  
Press any key to continue . . .
```

Linguagens de programação mais usadas:

JavaScript

Usada tanto no front como no back-end, é uma linguagem versátil.

É muito usada para desenvolvimento web e o programador certamente terá contato com ela no mercado de trabalho.



```
console.log("Hello, World!");
```

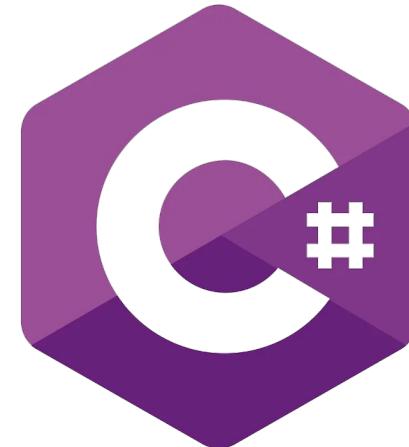
Linguagens de programação mais usadas:

C#

Assim como o Java, é portável e escalável.

Desenvolvida pela Microsoft, é uma linguagem muito conhecida e profundamente usada no desenvolvimento de jogos.

É construída baseada nas linguagens C++, Java e Object Pascal.



```
using System;

class Program
{
    static void Main()
    {
        Console.WriteLine("Hello, World!");
    }
}
```

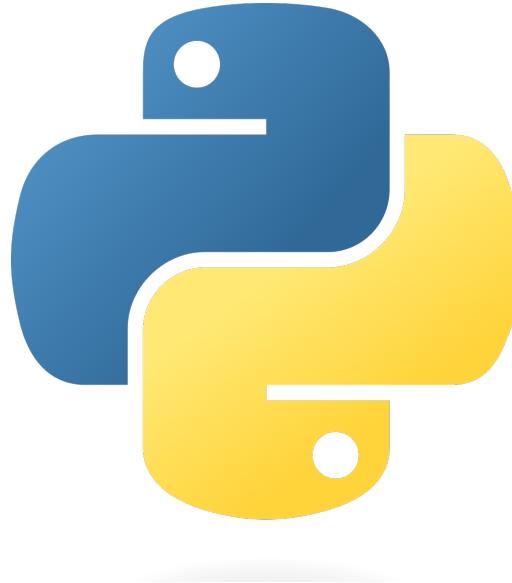
Linguagens de programação mais usadas:

Python

Uma das linguagens mais fáceis de aprender.

Tem sintaxe simples, direta e legível.

É muito utilizada nas áreas de Machine Learning e Ciência de Dados.



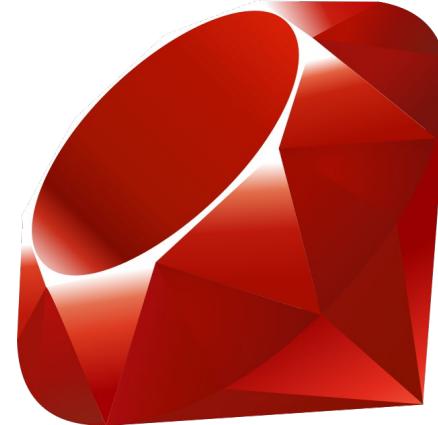
```
print("Hello, World!")
```

Linguagens de programação mais usadas:

Ruby

Sintaxe simples e fácil, tem como objetivo usar as melhores práticas da programação funcional e imperativa.

Muito popular, usada na criação do Airbnb, Twitter e GitHub.



```
puts "Hello, World!"
```

Linguagens de programação mais usadas:

A linguagem de baixo nível está próxima das linguagens de máquina, enquanto a linguagem de alto nível está mais próxima da compreensão e da linguagem humana.





Aula 02

Principais tipo de aplicações

Programação

Antes de falar sobre os tipos de aplicações, vamos entender como eles ganham vida. Compiladores e interpretadores são nossos aliados nessa jornada.



Fonte: vecteezy – construindo software

Disponível em <https://pt.vecteezy.com/arte-vetorial/9160120-cloud-server-web-hosting-development-time>
Acesso em 01 Dezembro 2023

Compilador

Transforma código fonte em linguagem de máquina, identifica erros, e gera um arquivo executável para o computador.

Etapas de compilação

Análise: Examina o código fonte em busca de erros e comprehende a estrutura da linguagem.

Síntese: Traduz o código para linguagem binária, criando um arquivo executável.

Exemplos de Softwares:

- GCC (GNU Compiler Collection).
- Visual Studio Compiler.

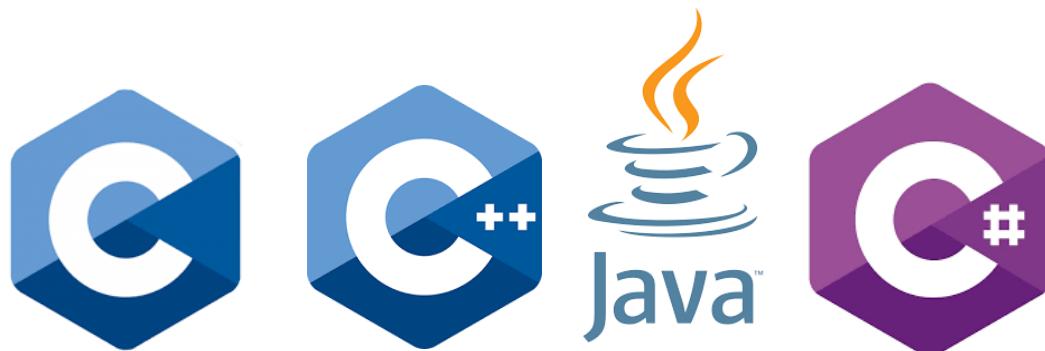
Linguagens compiladas

Linguagens compiladas são transformadas diretamente em código de máquina, executável pelo processador.

Essa abordagem as torna mais rápidas e eficientes em comparação com linguagens interpretadas. Além disso, oferecem aos desenvolvedores maior controle sobre aspectos do hardware, como gerenciamento de memória e uso da CPU.

Exemplos de Linguagens Compiladas:

1. C
2. C++
3. Java
4. C#



Definição:

- Programa que traduz e executa instruções uma a uma, sem converter todo o código para linguagem de máquina de uma vez.

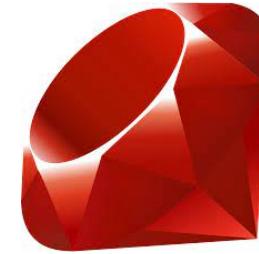
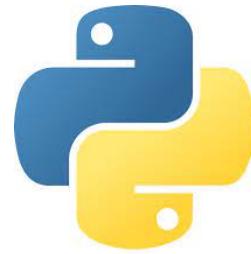
Exemplos de Linguagens Interpretadas :

1. Python

2. JavaScript

3. Ruby

4. PHP





Aplicações

O que é uma aplicação

Uma aplicação converte tarefas tediosas em processos dinâmicos, eficientes e digitais, simplificando e aprimorando atividades cotidianas.



Fonte: freepik – Lendo mapa
Disponível https://br.freepik.com/fotos-premium/turista-asiatica-segurando-o-mapa-da-cidade-de-turismo-na-cidade-de-osaka-japao-andando-no-mercado-fervilhante-pela-manha-enormes-lanternas-vermelhas-penduradas-no-telhado-dentro-de-casa-na-area-comercial_32316099.htm#from_view=detail_author
Acesso em 01 Dezembro 2023



O que é uma aplicação



Fonte: TREETOP – Locadora
Disponível https://cdn.acritica.net/upload/dn_arquivo/2021/07/screenshot-27_4.jpg
Acesso em 01 Dezembro 2023



Fonte: Netflix – Netflix
Disponível <https://www.netflix.com/>
Acesso em 01 Dezembro 2023



Fonte: Guia da carreira – Presencial
Disponível <https://www.guiadacarreira.com.br/blog/empresas-podem-exigir-o-retorno-do-trabalho-presencial>
Acesso em 01 Dezembro 2023



Fonte: Medium– HomeOffice
Disponível <https://medium.com/@rogersdepelle/home-office-a-arte-de-trabalhar-de-pijama-parte-i-fe9d7abb7952>
Acesso em 01 Dezembro 2023

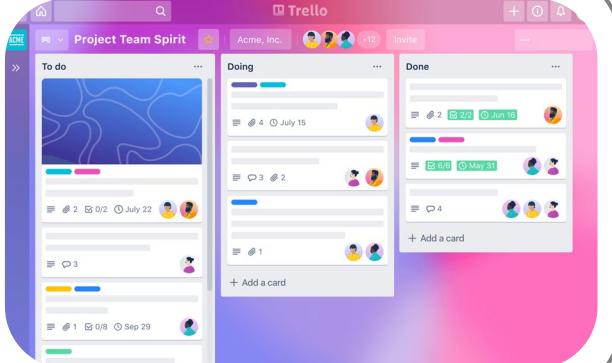


Fonte: Valeinox– caixa de correio
Disponível <https://www.valeinox.com.br/produtos/caixa-de-correio-inox-modelo-l-para-grade-horizontal/>
Acesso em 01 Dezembro 2023

O que é uma aplicação



Fonte: Freepik – Postit
Disponível https://www.freepik.com/premium-ai-image/creative-brainstorming-colorful-adhesive-notes-modern-office-organization-generated-by-ai_48025817.htm
Acesso em 01 Dezembro 2023



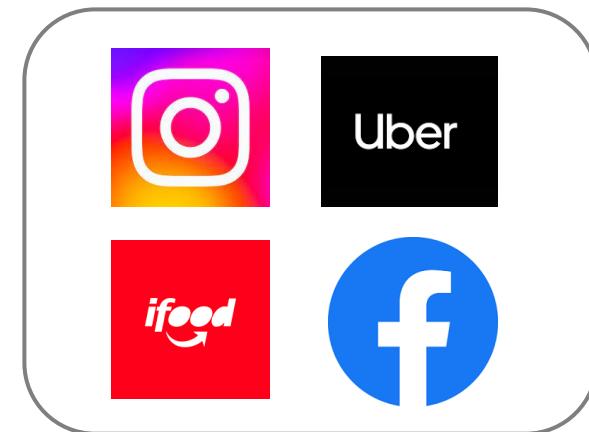
Fonte: Freepik – Trello
Disponível <https://trello.com/tour>
Acesso em 01 Dezembro 2023



Fonte: Freepik – Trello
Disponível <https://oglobo.globo.com/economia/desinformacao-erros-no-sistema-aumentam-filas-nas-agencias-da-caixa-1-24398865>
Acesso em 04 Dezembro 2023



Fonte: Imagem criada por Inteligência Artificial no Bing.



Aplicações WEB

Definição:

- Acessíveis através de navegadores web.

Linguagens de Programação:

- JavaScript: Para interatividade do lado do cliente ou até mesmo o servidor.
- HTML/CSS: Para estrutura e estilo.



Aplicações Mobile

Definição:

- Aplicações projetadas para dispositivos móveis, proporcionando mobilidade e acessibilidade.

Linguagens de Programação:

- Swift (iOS): Desenvolvimento para dispositivos Apple.
- Kotlin/Java (Android): Desenvolvimento para dispositivos Android.
- React-native para desenvolvimento para ambos os sistemas operacionais.



Aplicações Desktop

Definição:

- Aplicações instaladas e executadas localmente em computadores.

Linguagens de Programação:

- Java: Multiplataforma.
- C# (C Sharp): Microsoft Windows.
- Python: Multiplataforma.



Fonte: Imagem criada por Inteligência Artificial no Bing.

Aplicações Híbridas

Definição:

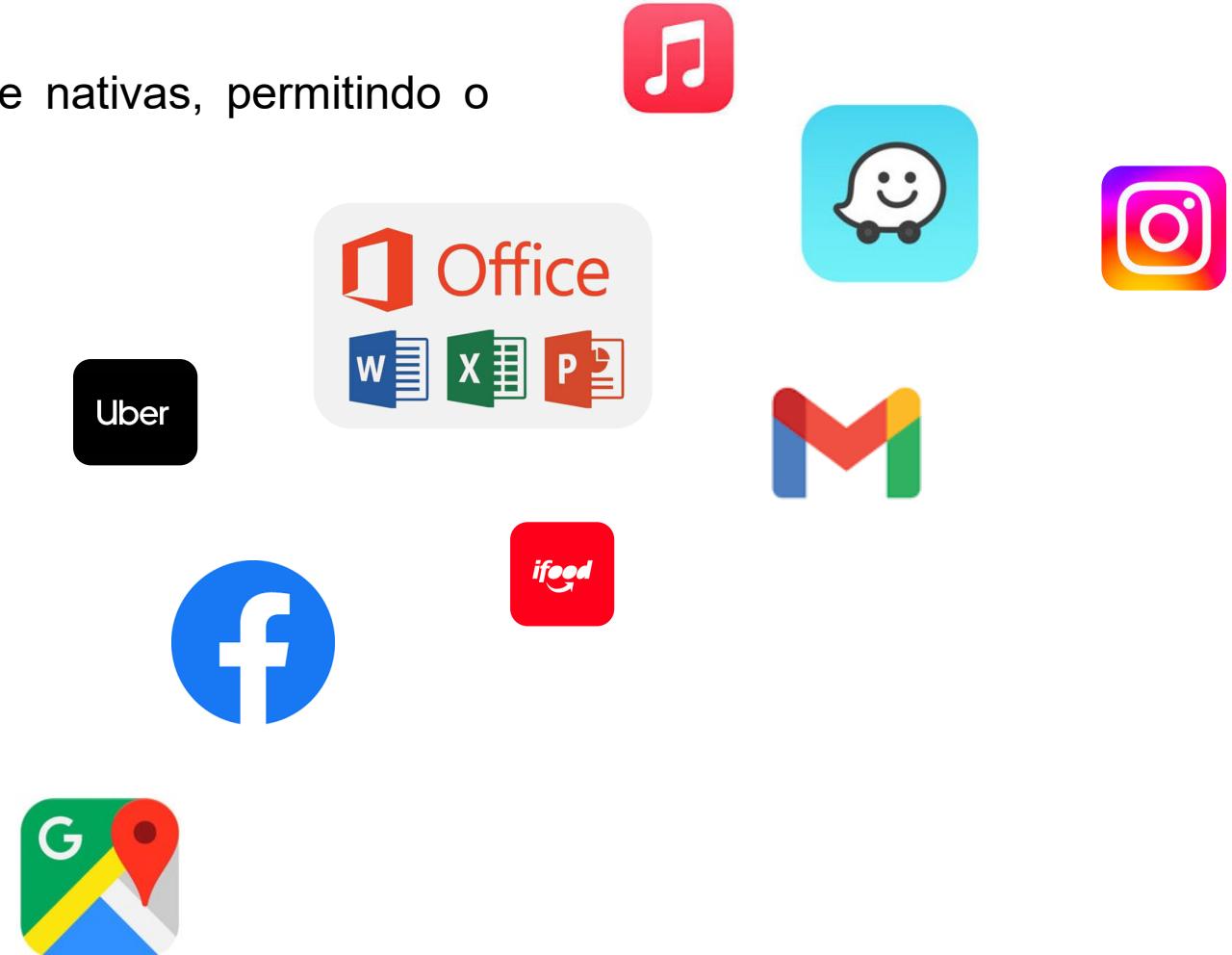
- Combinam elementos de aplicações web e nativas, permitindo o uso em múltiplas plataformas.

Linguagens de Programação:

- JavaScript: Para lógica de negócios.
- HTML/CSS: Para estrutura e estilo.
- Frameworks como React Native ou Ionic.

Importância:

- Desenvolvimento mais rápido.
- Código reutilizável entre plataformas.
- Manutenção simplificada.



Aplicações Nativas

Definição:

- Desenvolvidas para uma plataforma específica (iOS, Android, Windows), otimizando o desempenho e a experiência do usuário.



Linguagens de Programação:

- Swift (iOS): Desenvolvimento para dispositivos Apple.
- Kotlin/Java (Android): Desenvolvimento para dispositivos Android.
- C# (UWP - Windows): Desenvolvimento para Windows.



Importância:

- Muitas das vezes já vem instalado no dispositivo.
- Oferecem integração profunda com os sistemas operacionais, explorando ao máximo os recursos e proporcionando uma experiência nativa.

Aplicações Nuvem (Cloud)

Definição:

- Hospedadas em servidores remotos e acessadas pela internet, proporcionando flexibilidade e escalabilidade.



Linguagens de Programação:

- Variedade de Linguagens: Depende da infraestrutura da nuvem escolhida.
- Exemplos comuns: Java, Python, Node.js.



Microsoft 365



Importância:

- Transformam a forma como armazenamos, processamos e compartilhamos informações, proporcionando maior flexibilidade e eficiência.



Aplicações Inteligência Artificial (IA)

Definição:

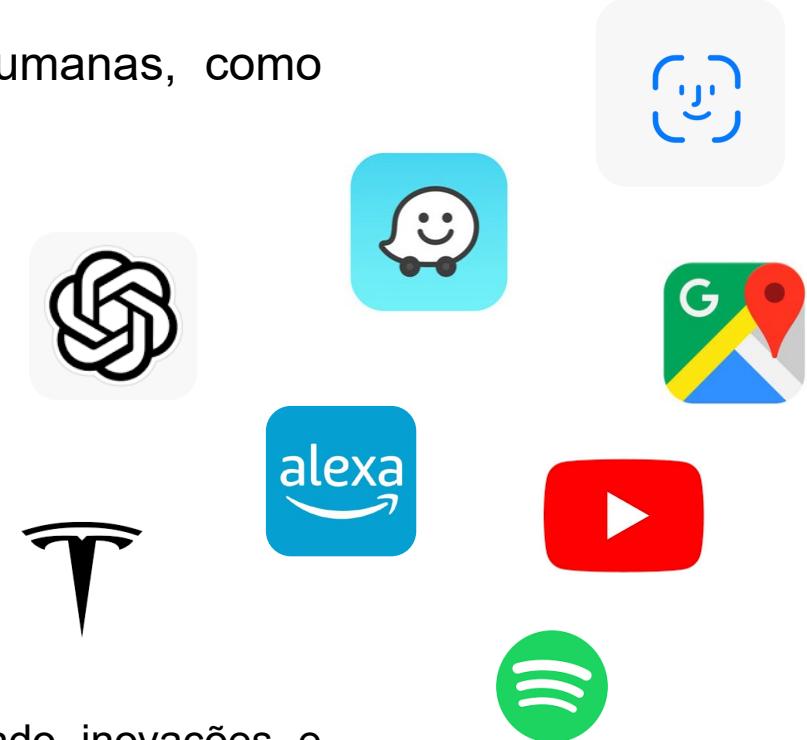
- Utilizam algoritmos inteligentes para imitar funções cognitivas humanas, como aprendizado, raciocínio e resolução de problemas.

Linguagens de Programação:

- Python: Amplamente utilizado para desenvolvimento de IA.
- R: Para análise estatística e aprendizado de máquina.

Importância:

- Transformam setores como saúde, comércio e transporte, proporcionando inovações e eficiências significativas.



Aplicações de Realidade Virtual (RV) e Realidade Aumentada (RA)

Definição:

Realidade Virtual (RV): Cria ambientes virtuais imersivos.

Realidade Aumentada (RA): Sobreposição de elementos virtuais no mundo real.

Linguagens de Programação:

- Unity (C#): Amplamente usado para desenvolvimento de RV e RA.
- Java, Swift, Kotlin: Dependendo da plataforma móvel.
- Blueprint: Programação por blocos (Unreal Engine)

Importância:

- Transformam a maneira como interagimos com ambientes digitais e físicos, oferecendo experiências envolventes e práticas.

Aplicações de Realidade Virtual (RV)



imagem criada pelo Instituto SENAI de Tecnologia



Fonte: Horizon Workrooms – Meta
Disponível <https://about.fb.com/news/2021/08/introducing-horizon-workrooms-remote-collaboration-reimagined/>
Acesso em 04 Dezembro 2023



Fonte: Ocen property– Usando Quest
https://www.windowscentral.com/sites/wpcentral.com/files/styles/large_wm_brw/public/field/image/2019/03/oculus-rift-s-side-russell-two-controllers.jpg?itok=96r23h7i
Acesso em 04 Dezembro 2023

Aplicações de Realidade Aumentada (RA)



Fonte:Microsoft – Dynamics 365 Guides with HoloLens 2

Disponível
<https://blogs.windows.com/windowsexperience/2023/04/13/microsoft-brings-windows-11-to-hololens-2/>
 Acesso em 04 Dezembro 2023



Fonte:Microsoft – Dynamics 365 Guides with HoloLens

Disponível <https://microsoft-business-applications.hcltech.com/services/mixed-reality/>
 Acesso em 05 Dezembro 2023



Fonte: tecnolife – Hololens 2
<https://www.tecnolife.com.br/philips/2019/2/microsoft-lanca-hololens-2-em-parceria-com-a-philips/>

Acesso em 04 Dezembro 2023



Fonte: techcrunch – Dynamics 365 applications

Disponível
<https://techcrunch.com/2018/09/18/microsoft-is-putting-hololens-to-work-in-dynamics-365/>
 Acesso em 04 Dezembro 2023



Aula 03

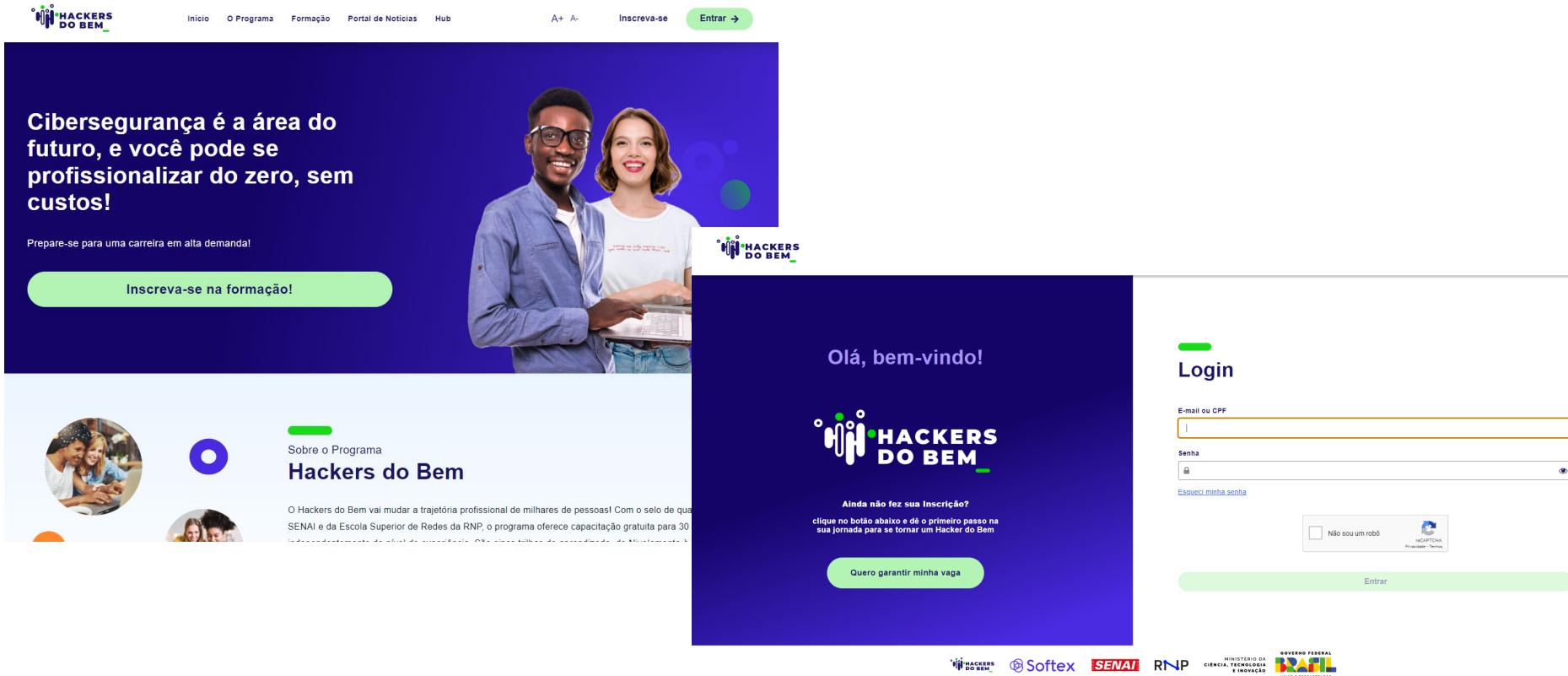
Conceitos de Front-end e Back-end



Front-end

Front-end

É uma área da programação que se dedica a criar a parte visual e interativa de um site, aplicativo ou software. É o que o usuário vê e usa quando acessa uma plataforma digital.



The screenshot displays the Hackers do Bem website's front-end design. At the top, there is a navigation bar with links for 'Início', 'O Programa', 'Formação', 'Portal de Notícias', 'Hub', font size controls ('A+', 'A-'), a registration button ('Inscreva-se'), and a login button ('Entrar'). The main header features a purple background with two people smiling and the text: 'Cibersegurança é a área do futuro, e você pode se profissionalizar do zero, sem custos!'. Below this, a call-to-action button says 'Inscreva-se na formação!'. To the left, there are circular profile pictures and a section titled 'Sobre o Programa Hackers do Bem' with a brief description. The central part of the page has a dark blue background with the 'HACKERS DO BEM' logo and a button to 'Quero garantir minha vaga'. On the right, there is a 'Login' form with fields for 'E-mail ou CPF', 'Senha', and 'Receber notícias', along with a CAPTCHA and an 'Entrar' button.

Desenvolvimento Front-end

Principais linguagens de programação:



Python



.js JavaScript



Haskell



HTML



CSS



GoLang



Swift



Kotlin



PHP

UX Design

O UX Design é a área do design de produto que vai além dos aspectos visuais, garantindo que a usabilidade seja simples e intuitiva ao usuário.

Contact Us

First Name Last Name

E-mail Business Phone

Company Name Job Title

Street 1

City State/Province

ZIP/Postal Code Country/Region

Topic

Contact Us

Feel like contacting us? Submit your queries here and we will get back to you as soon as possible.

Velocity Web, LLC
3245 Rose Avenue, Avondale, LA 70094

Send us a Message

Name

Email

Phone

Preferred method of communication

Email Phone

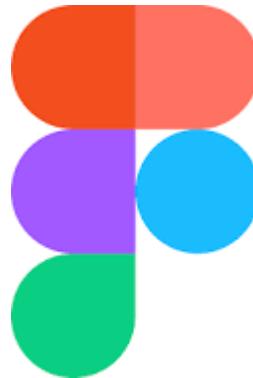
Message

Submit

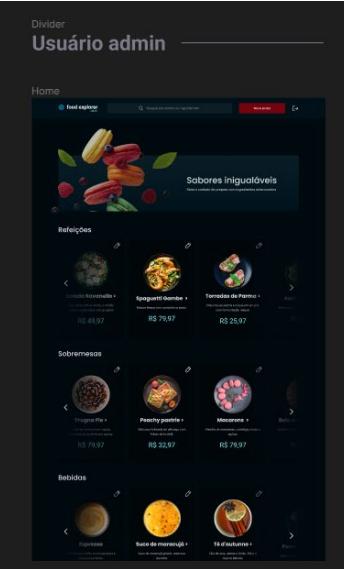
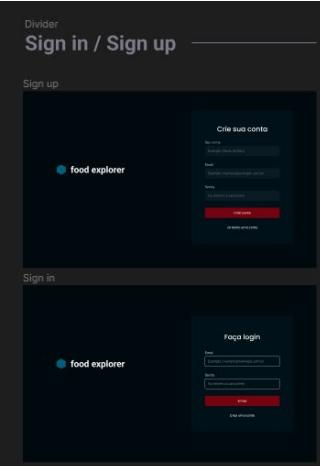
Protótipos

O Figma é uma plataforma online de criação de interfaces, wireframes e protótipos. Seu papel é oferecer recursos de design de telas para aplicações variadas, permitindo que times de Design trabalhem em conjunto no mesmo projeto remotamente e simultaneamente.

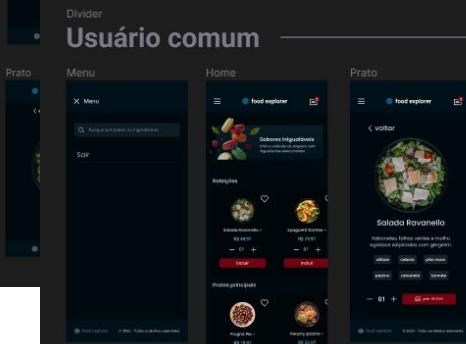
Figma:



Protótipos



Desktop



Mobile



Assets

Inspecionando



<https://conteudo.hackersdobem.org.br/>

div.container 1320 × 85

Cibersegurança é a área do futuro, e você pode se profissionalizar do zero, sem custos!

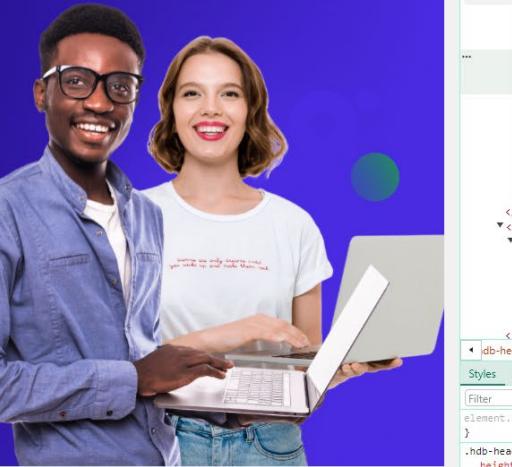
Prepare-se para uma carreira em alta demanda!

Inscreve-se na formação!

Sobre o Programa

Hackers do Bem

O Hackers do Bem vai mudar a trajetória profissional de milhares de pessoas! Com o selo de qualidade do SENAI e da Escola Superior de Redes da PNP, o programa oferece capacitação gratuita para 30 mil alunos.



Hackers do Bem

INÍCIO

O Programa

Formação

Portal de Notícias

Hub

A+ A-

Inscreve-se

Entrar →

```
<!DOCTYPE html>
<html lang="en">
  <head> ...
    <script>You need to enable JavaScript to run this app.</script>
  </head>
  <body> ...
    <div id="root"> ...
      <header class="hdb-header">
        <div class="container">
          <nav class="hdb-header__nav"> ...
            <a title="Página inicial" aria-current="page" href="/" class="hdb-header__logo active"> ...
              
            ...
          </a>
          <ul class="hdb-header__menu" ...> ...
          <ul class="hdb-header__accessibility-controls" ...> ...
          <ul class="hdb-header__auth-buttons" ...> ...
          <div class="hdb-dropdown-menu" ...> ...
        </nav>
      </div>
    </header>
    <main>
      <div class="hdb-home">
        <div class="hdb-banner" ...> ...
        <div class="hdb-blue-gradient-container" ...> ...
        <div class="hdb-home__gradient" ...> ...
        <div class="hdb-blue-gradient-container" ...> ...
      </div>
    </main>
    <div class="hdb-container nav.hdb-header__nav a.hdb-header__logo.active img" ...> ...
      <img alt="Logo Hackers do Bem" data-bbox="855 185 955 275"/>
    </div>
  </body>

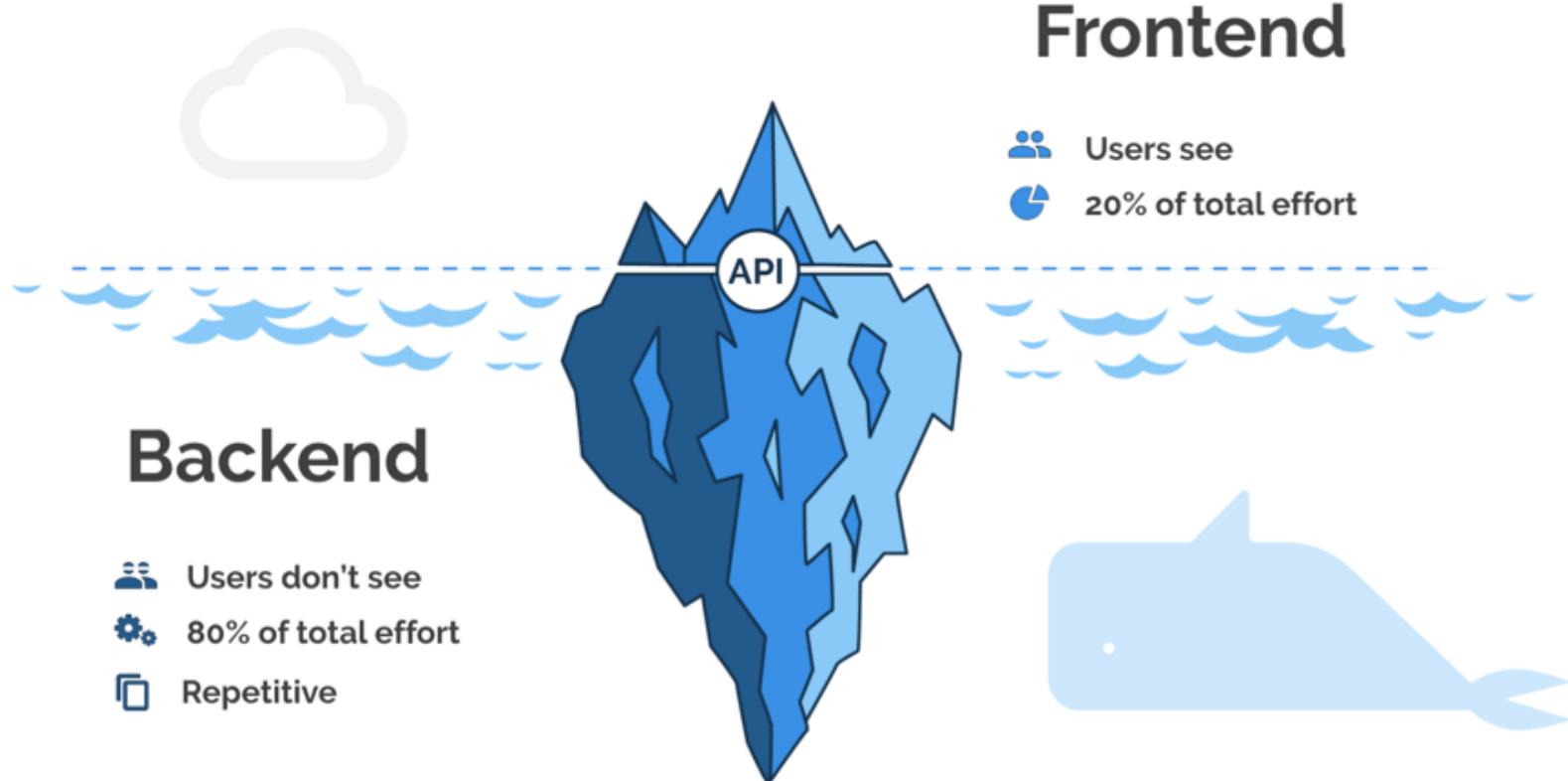
```



Back-end

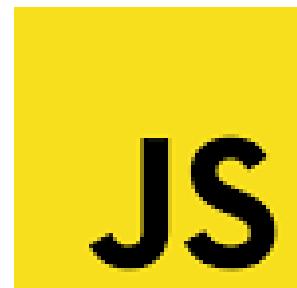
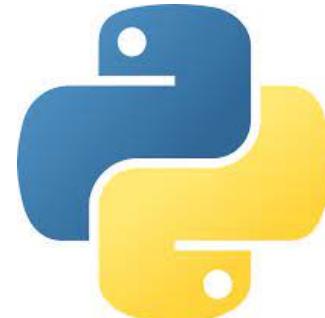
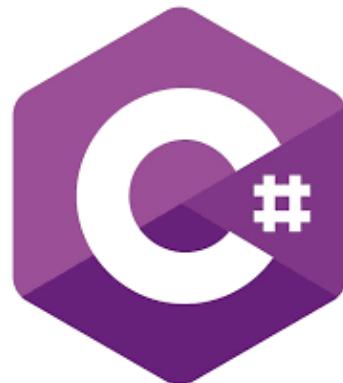
Back-end

É o código que conecta a internet com o banco de dados, gerencia as conexões dos usuários e alimenta a aplicações.



Desenvolvimento Back-end

Principais linguagens de programação:



Desenvolvimento Back-end

Exemplos de Funcionalidades:

- Gestão, Interação ou manutenção de bancos de dados
- Desenvolvimento ou Integração de APIs
- Segurança da informação
- Trabalho conjunto com profissionais de UX e UI
- Teste de projetos criados
- Criação de bibliotecas
- Planejamento, criação, implementação e manutenção da estrutura que forma a base de um site

Desenvolvimento Back-end

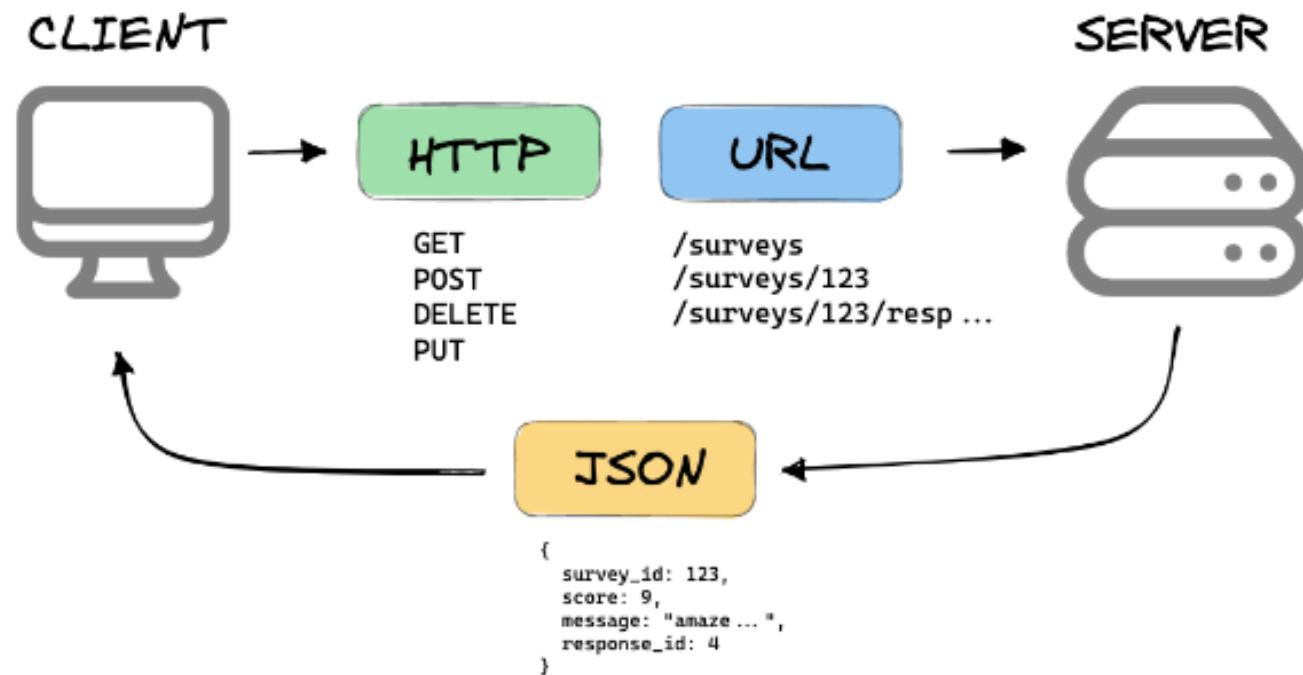
Continuando Exemplos:

- Criação de bancos de dados nos servidores e atualização
- Instalação de APIs
- Cuidado da performance do site, estrutura e lógica
- Acesso ao banco de dados para guardar e resgatar as informações do sistema
- Fornecimento de dados para o Front-end exibir na tela do usuário
- Comunicação com sistemas de terceiros
- Planejamento, programação, teste e manutenção da estrutura de códigos que faz a interface entre um site, o servidor e o banco de dados

API

Application Programming Interface (Interface de Programação de Aplicação).

É um conjunto de padrões, instruções e rotinas de programação que permitem o acesso às funcionalidades de um aplicativo por um outro software. Dessa forma, é possível fazer com que dois sistemas diferentes “conversem” entre si e compartilhem dados, funções e serviços.



Requisições HTTP

As requisições HTTP são mensagens enviadas pelo cliente para iniciar uma ação no servidor.

Os verbos HTTP mais utilizados são:

- *GET* - Solicita dados de um recurso.
- *POST* - Envia dados para serem processados por um recurso.
- *PATCH* - Modifica partes de um recurso existente.
- *PUT* - Atualiza um recurso.
- *DELETE* - Remove um recurso.

Análise de riscos

É uma atividade que visa identificar, avaliar e priorizar os possíveis riscos que podem afetar a segurança de uma aplicação, tais como:

- ameaças,
- vulnerabilidades,
- impactos e probabilidades.

A partir da análise de riscos, é possível definir as estratégias e as ações de mitigação, prevenção e resposta aos riscos, bem como os recursos necessários para implementá-las.

Testes de Segurança

Criptografia

Atualização

Autenticação

Testes de segurança

São processos que verificam e validam se uma aplicação atende aos requisitos e aos padrões de segurança estabelecidos.

bem como se ela está livre de **fallas, erros e vulnerabilidades** que possam comprometer o seu funcionamento e a sua segurança.

Os testes de segurança devem ser realizados em todas as fases do ciclo de vida do desenvolvimento de software, desde a concepção até a manutenção, e devem abranger aspectos como **funcionalidade, usabilidade, performance, confiabilidade, compatibilidade e segurança**.

Atualização e conhecimento das ameaças

É importante também estar atualizado sobre as mais recentes ameaças e técnicas de ataque que podem afetar a segurança de uma aplicação, bem como sobre as melhores práticas e as soluções disponíveis para combatê-las.

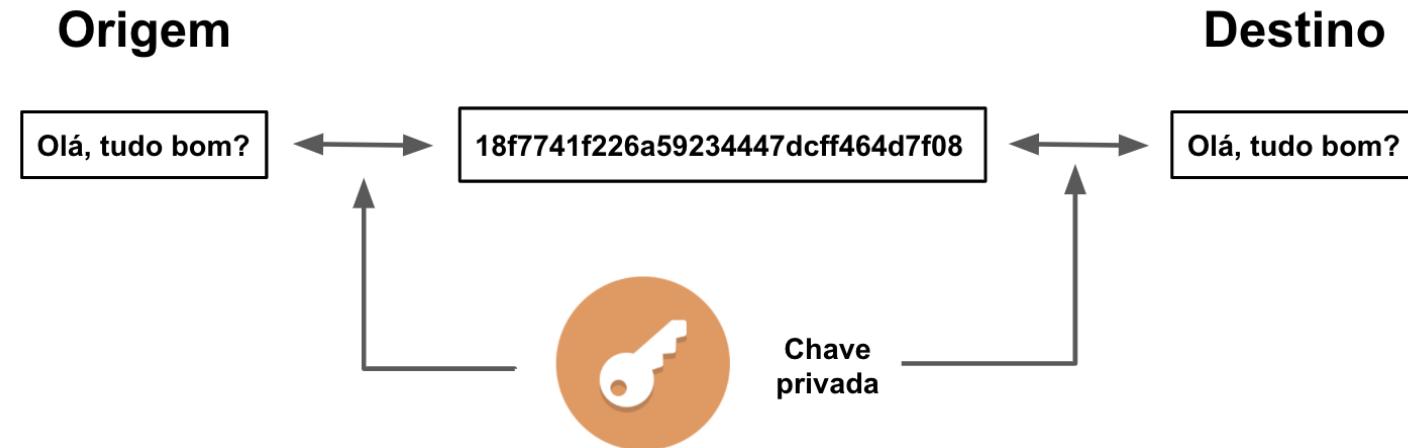
Isso permite antecipar e prevenir possíveis ataques, bem como reagir e se recuperar de forma rápida e eficaz em caso de incidentes.



Criptografia de dados

É um método que transforma os dados em uma forma ilegível para quem não possui a chave de acesso, impedindo que eles sejam acessados, modificados ou roubados por pessoas não autorizadas.

A criptografia de dados pode ser aplicada tanto aos dados em trânsito, como aos dados em repouso, aumentando a confidencialidade e a integridade das informações.

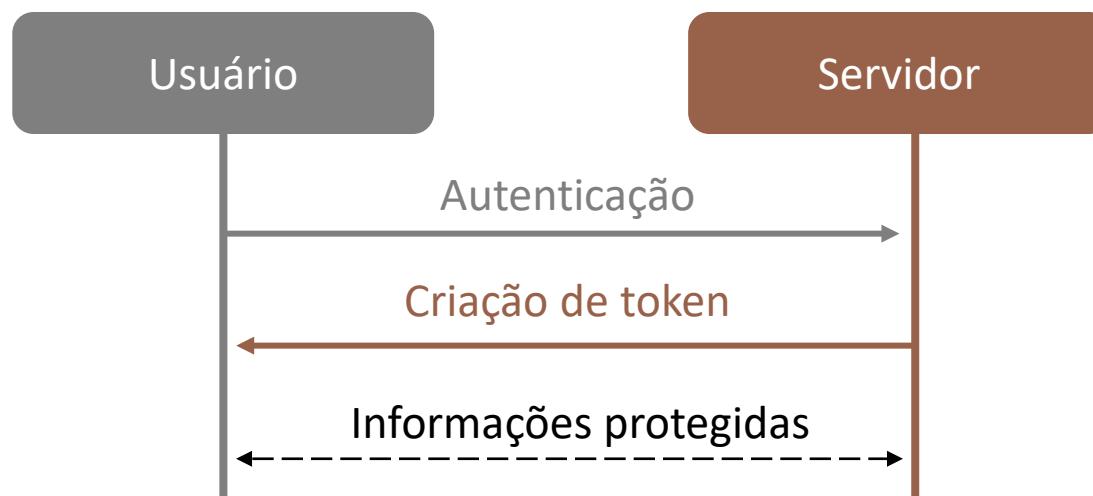


Autenticação e autorização de usuários

São mecanismos que controlam o acesso dos usuários aos recursos e às funcionalidades de uma aplicação, verificando a sua identidade e o seu nível de permissão.

A autenticação e a autorização de usuários podem ser implementadas por meio de diferentes métodos, como senhas, tokens, biometria, certificados digitais, entre outros.

Esses mecanismos garantem que apenas os usuários legítimos possam acessar e manipular os dados e os recursos da aplicação, evitando o acesso indevido ou malicioso.



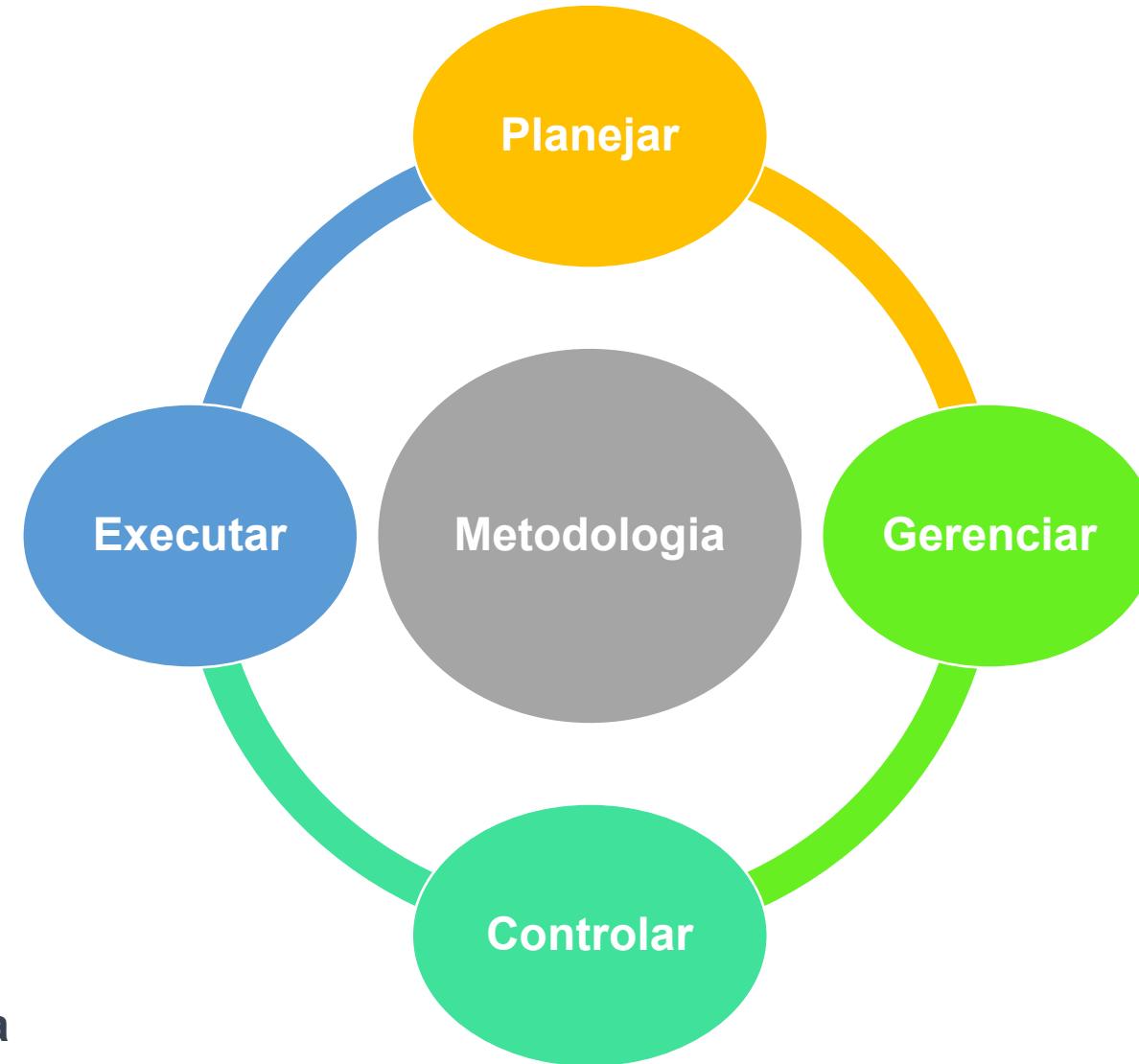


Aula 04

Metodologias de Desenvolvimento

Importância das Metodologias de Desenvolvimento

- Mais Organização
- Melhor Comunicação
- Qualidade Garantida
- Menos Surpresas
- Se Adapta às Mudanças
- Trabalho em Equipe
- Cumprimento de Prazos
- Clientes Felizes
- Sempre Melhorando
- Resultados que Valem a Pena



Metodologias de Desenvolvimento

Cascata



Desenvolvimento sequencial em que as fases são executadas em ordem fixa.



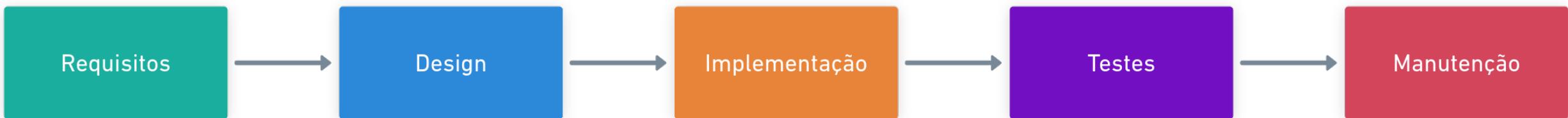
Uma fase deve ser concluída para que outra seja iniciada.

Metodologias de Desenvolvimento

Indicado Para:

- Projetos bem definidos desde o início, ou seja, requisitos claros e estáveis desde o começo do projeto, facilitando a progressão em etapas previsíveis.

Funcionamento:



1. Requisitos - Define claramente o que o sistema precisa fazer.
2. Design - Planeja a arquitetura e estrutura do sistema.
3. Implementação - Codificação do sistema com base nos requisitos e design.
4. Testes - Verifica se o sistema atende aos requisitos especificados.
5. Manutenção - Correções e melhorias após a entrega.

Desenvolvimento Ágil



Abordagem iterativa e colaborativa que enfatiza a entrega contínua de software funcional.



Adaptação a mudanças e a satisfação do cliente.



Desenvolvimento Ágil

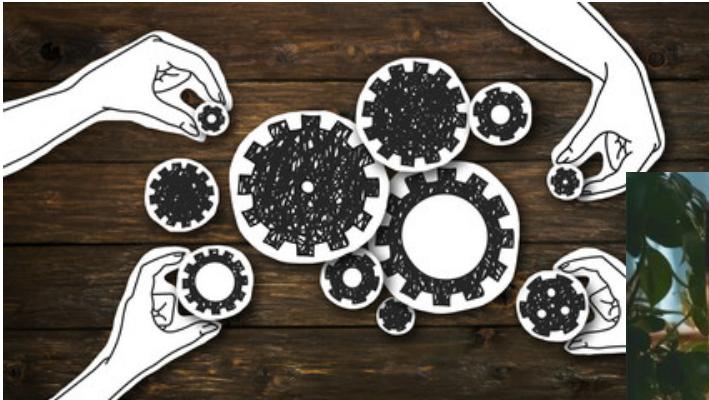
Desenvolvimento Ágil

Indicado Para:

- Projetos que exigem flexibilidade: Ideal para ambientes onde os requisitos podem mudar ou evoluir.

Princípios Ágeis:

Indivíduos e interações acima de processos e ferramentas



Responder a mudanças acima de seguir um plano



Software em funcionamento acima de documentação abrangente

Scrum

Características :

- Divide o projeto em iterações chamadas sprints.

Ciclo de Vida do Scrum:

1. Planejamento do Sprint:

Seleção de tarefas para o sprint.

2. Daily Scrum:

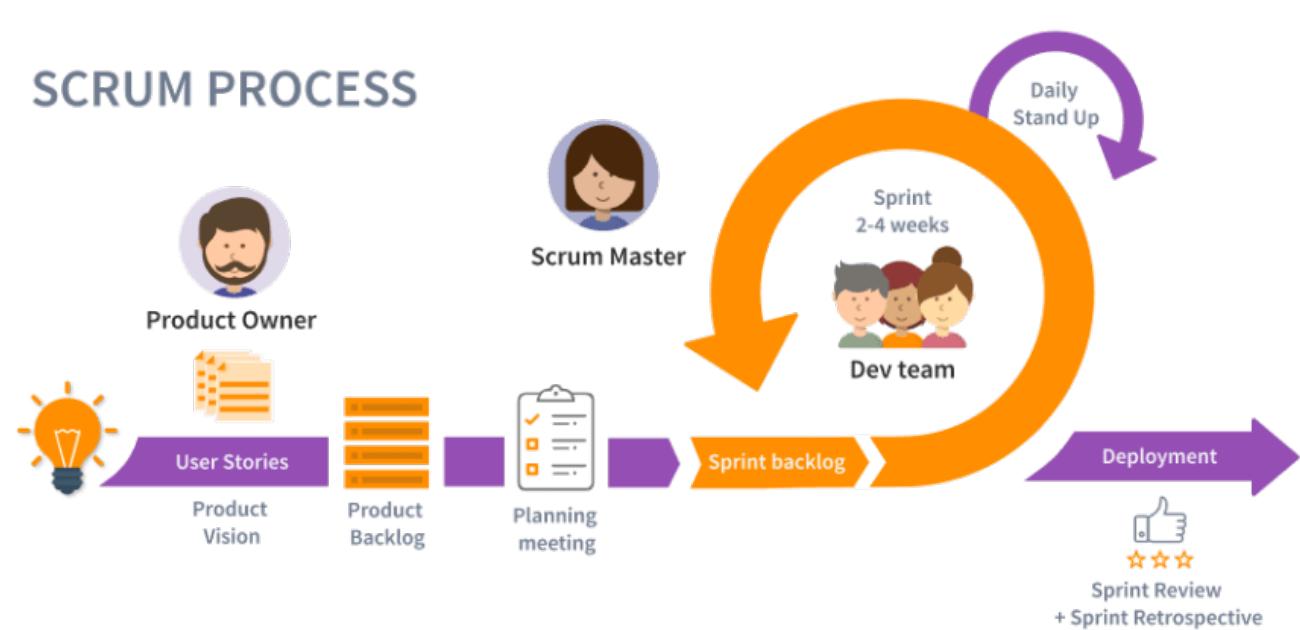
Reuniões diárias para atualizações rápidas.

3. Revisão do Sprint:

Demonstração do que foi feito no sprint.

4. Retrospectiva do Sprint:

Avaliação para melhorias contínuas.

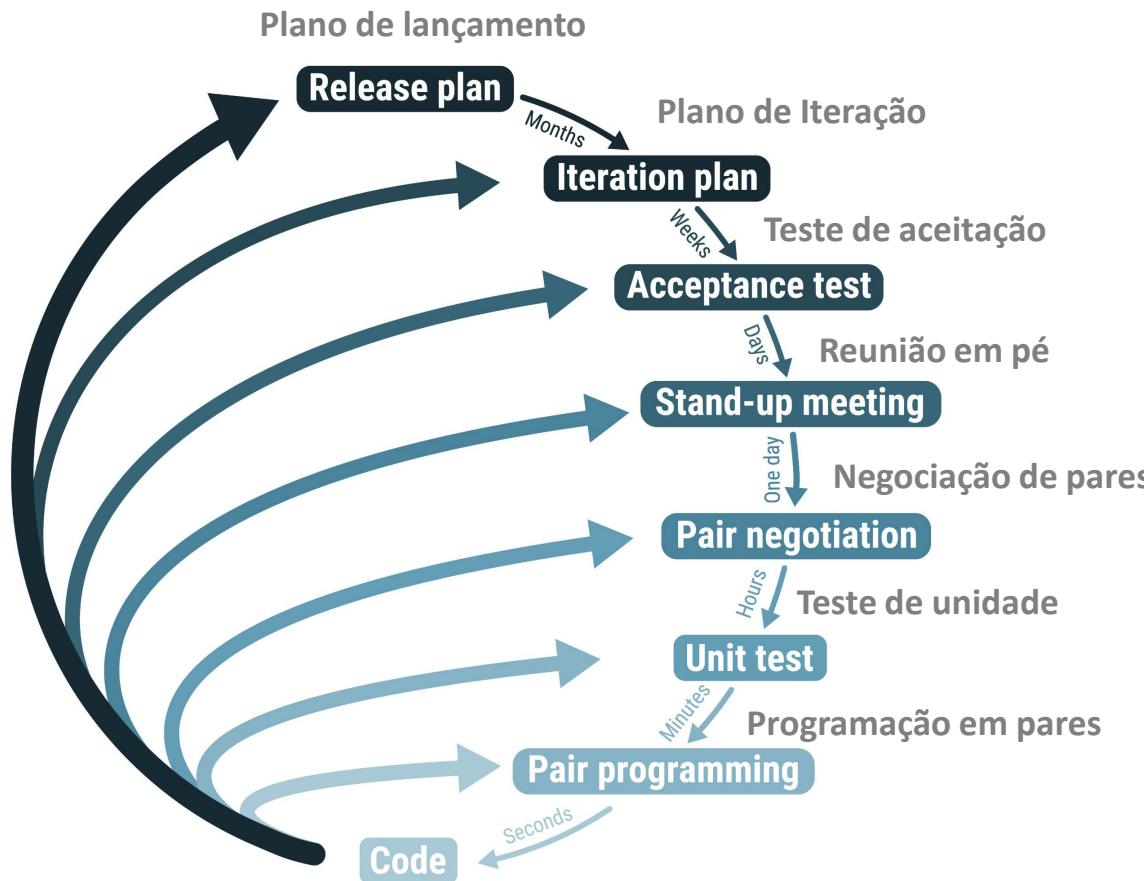


Fonte: Usenmobile – Lendo mapa
 Disponível <https://usenmobile.com.br/metodologia-scrum-desenvolvimento/>
 Acesso em 08 Dezembro 2023

Metodologia Extreme Programming (XP)

Características:

- Ênfase em técnicas que melhoram a qualidade do código e a colaboração na equipe.



Ciclo de Vida do XP:

- 1. Planejamento:**
Estabelece as metas do cliente.
- 2. Pequenos Lançamentos:**
Entregas frequentes de pequenas funcionalidades.
- 3. Feedback do Cliente:**
Avaliação contínua para ajustes.
- 4. Retrospectiva:**
Avaliação interna da equipe para melhorias.

Kanban

Características :

- Ênfase na Visualização do Fluxo de Trabalho, Representado por um quadro Kanban.

Ciclo de Vida no Kanban:

1. Adicionar Tarefas ao Quadro:

Identificar e adicionar tarefas à lista "A fazer".

2. Mover Tarefas para "Em Progresso":

Quando a equipe começa a trabalhar nelas.

3. Mover Tarefas para "Concluído":

Quando as tarefas são finalizadas.



Fonte: Imagem retirada de Adobe Stock

Desenvolvimento Espiral



Abordagem iterativa e de gestão de riscos.

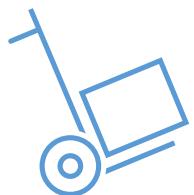


Une etapas do desenvolvimento em cascata com análises e avaliações periódicas.

Desenvolvimento Incremental



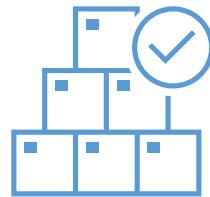
Abordagem que divide o projeto em partes independentes e funcionais.



Entrega funcionalidades adicionais em cada incremento.

Metodologias de Desenvolvimento

Desenvolvimento Lean



Busca a eficiência eliminando desperdícios.



Prioriza a entrega de valor ao cliente.

Entrega de Valor ao Cliente:

- Identificação das Necessidades do Cliente;
- Eliminação de Desperdícios;
- Eficiência Operacional;
- Rápida Resposta a Mudanças;
- Feedback Contínuo.

Metodologias de Desenvolvimento

Desenvolvimento Test-Driven



Envolve a criação de testes automatizados antes do código de produção.



Os testes são para garantir que o código atenda aos requisitos especificados pelos clientes.

Passos Principais do TDD:

- Escrever Teste
- Falhar o Teste
- Implementar Funcionalidade
- Executar Testes
- Refatorar o Código



Aula 05

Conceitos de Banco de Dados

Introdução a Banco de Dados

O que são Dados?

Dados são informações ou fatos que têm significado em um contexto específico.

Exemplo: nome e número de telefone de pessoas conhecidas.

E os Bancos de Dados?

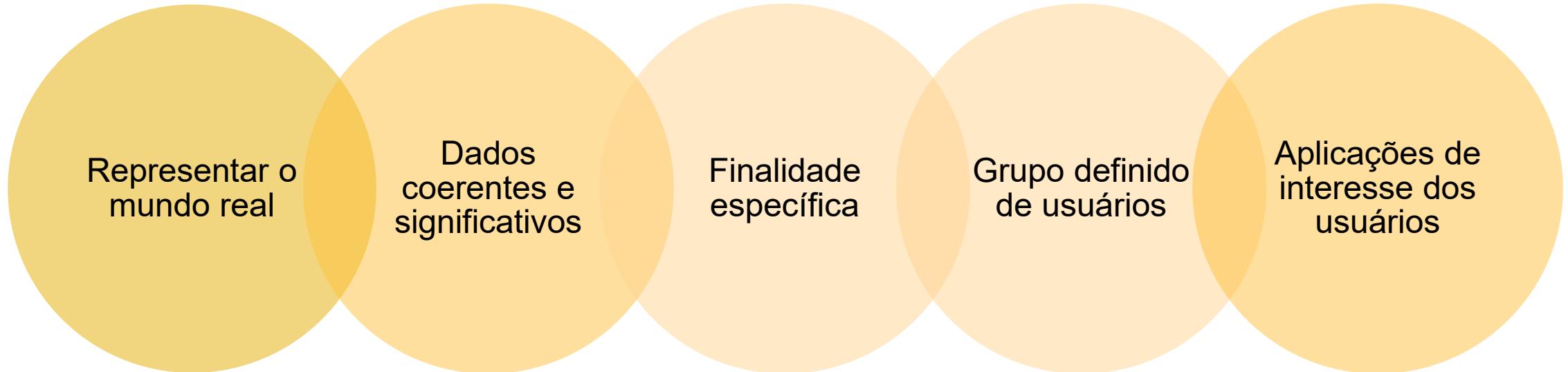
Bancos de dados são conjuntos organizados de dados armazenados em um sistema de computador.

Vamos conhecer 3 conceitos básicos:

- Propriedades
- Tamanho e Complexidade
- SGBD

Banco de Dados

Propriedades



Banco de Dados

Tamanho e complexidade

Bancos de dados podem variar significativamente em tamanho e complexidade.

Exemplos Ilustrativos:

1. Lista de Contatos no Celular:

- Banco de dados pequeno, organizado e personalizado.

2. Amazon:

- Mais de 20 milhões de itens, 15 milhões de visitantes diários, gerenciados por 200 servidores e atualizados por uma equipe substancial.

| Nome | Sobrenome | Email | Empresa | Celular | Cidade |
|----------|-----------|-----------------------|-----------|-----------------|----------------|
| Éber | Silva | ebersilv@br.ibm.com | IBM | (55) 99999-1234 | São Paulo |
| Bianca | Costa | bianca@gmail.com | IBM | (55) 99999-4321 | São Paulo |
| Bruno | Oliveira | bruno@ibm.com | Google | (55) 99999-9876 | São Paulo |
| Cibelle | Siqueira | cibelle@mediamath.com | MediaMath | (55) 99999-0987 | Campinas |
| Leonardo | Mello | leonardo@ibm.com | IBM | (55) 99999-0123 | Rio de Janeiro |
| Rafael | Chaves | rafael@hcl.com | HCL | (55) 99999-0789 | São Paulo |
| Ronaldo | Gonçalves | ronaldo@ibm.com | IBM | (55) 99999-4567 | São Paulo |

Fonte: Medium – **Contact Lists** - Disponível <https://medium.com/markethinkers/o-que-s%C3%A3o-contact-lists-e-como-us%C3%A1-las-ae6384c8f589/> - Acesso em 11 Dezembro 2023



Fonte: AWS – AWS Achieves
Disponível <https://aws.amazon.com/pt/blogs/publicsector/aws-achieves-full-empanelment-for-the-delivery-of-cloud-services-by-indias-ministry-of-electronics-and-information-technology/>
Acesso em 11 Dezembro 2023

Banco de Dados

SGBD – Sistema Gerenciador de Banco de Dados

O SGBD é um sistema de software projetado para criar, gerenciar e manipular bancos de dados.

Facilitam o processo de:

1. Definição:

Define a estrutura e as propriedades do banco de dados.

2. Construção:

Cria o banco de dados e tabelas com base nas definições estabelecidas.

3. Manipulação:

Realiza operações como seleção, inserção, atualização e exclusão de dados.

4. Compartilhamento:

Facilita o acesso e compartilhamento eficiente de dados entre usuários e várias aplicações.



Fonte: AWS – AWS Achieves

Disponível <https://leonardofonseca.com.br/2021/03/11/sistemas-gerenciadores-de-banco-de-dados-sgbd/>
Acesso em 11 Dezembro 2023

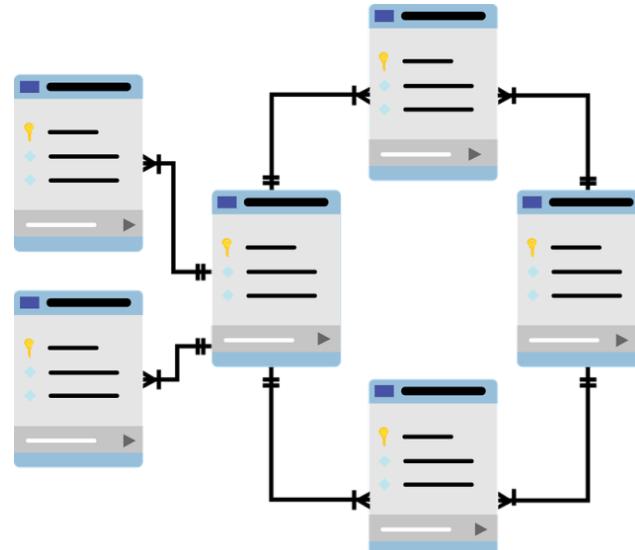


Tipos de bancos de dados

Tipos de Banco de Dados

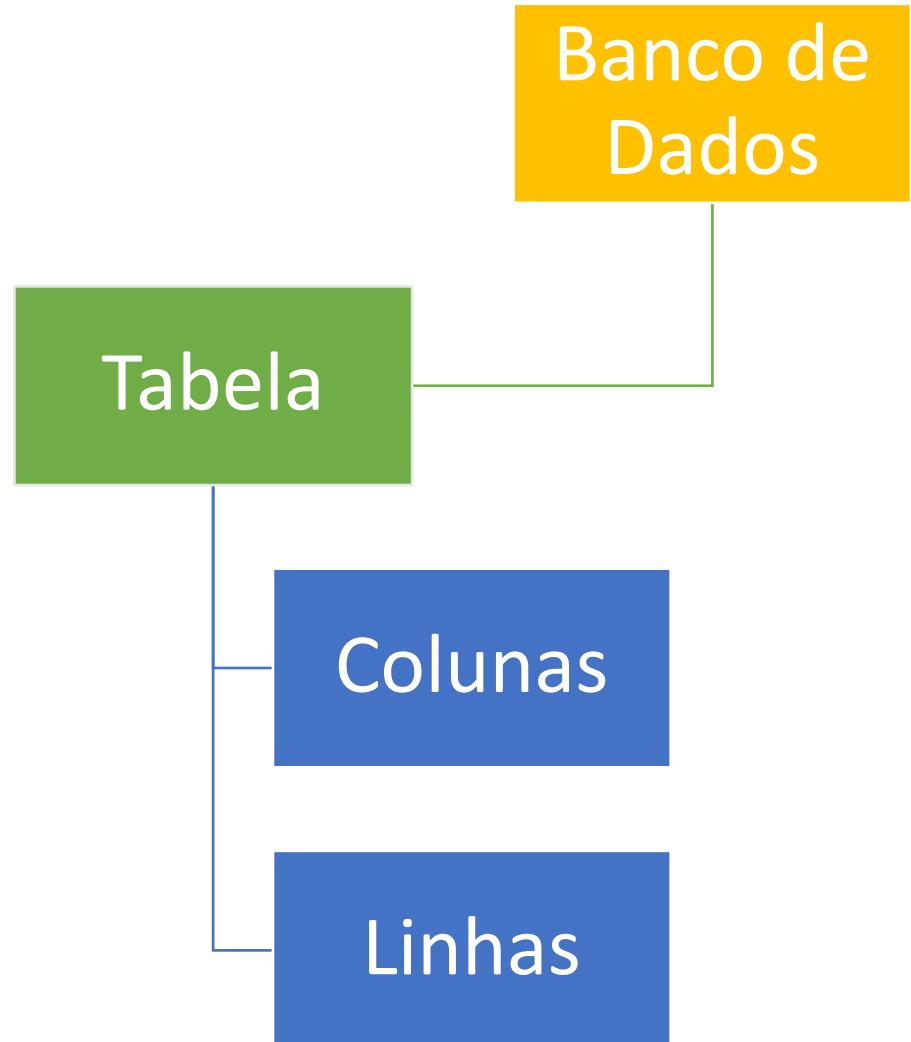
Bancos de Dados Relacionais:

- Organizam dados em tabelas com linhas e colunas.
- Facilitam o relacionamento entre diferentes conjuntos de dados.
- Exemplos: MySQL, PostgreSQL, SQLite.



Fonte: leonardofonseca – Bancos Relacionais
Disponível <https://leonardofonseca.com.br/2021/04/26/relacionamento-em-um-modelo-relacional-parte-1/Acesso em 11 Dezembro 2023>

Banco de Dados Relacionais



Banco de Dados Relacionais

Outros conceitos importantes

Chave Primária

Aplicada em um ou mais campos para tornar os dados únicos na tabela.

Chave Primária Simples → Um único campo que armazena uma informação única dentro da tabela.

Chave Primária Composta → Dois ou mais campos que, juntos, tornam as informações únicas dentro da tabela.

Chave primária 

| Tabela de Cliente | | |
|-------------------|---------------------|--------|
| Chave Primária | Código do Cliente | número |
| | Nome do Cliente | texto |
| | Telefone do Cliente | número |

Banco de Dados Relacionais

Outros conceitos importantes

Relacionamentos

Caracterizado pelo vínculo entre tabelas através de uma chave estrangeira. A chave estrangeira é vinculada a chave primária da tabela principal.

Veja um exemplo de como isso acontece:

| Tabela de Cliente | | |
|---|---------------------|--------|
|  | Código do Cliente | número |
|  | Nome do Cliente | texto |
|  | Telefone do Cliente | número |

Chave primária

| Tabela de Pedido | | |
|---|-------------------|--------|
|  | Código do Pedido | número |
|  | Código do Cliente | número |
| | Valor do Pedido | moeda |
| | Data do Pedido | data |

Chave estrangeira

Tipos de Banco de Dados

Bancos de Dados Não Relacionais (NoSQL):

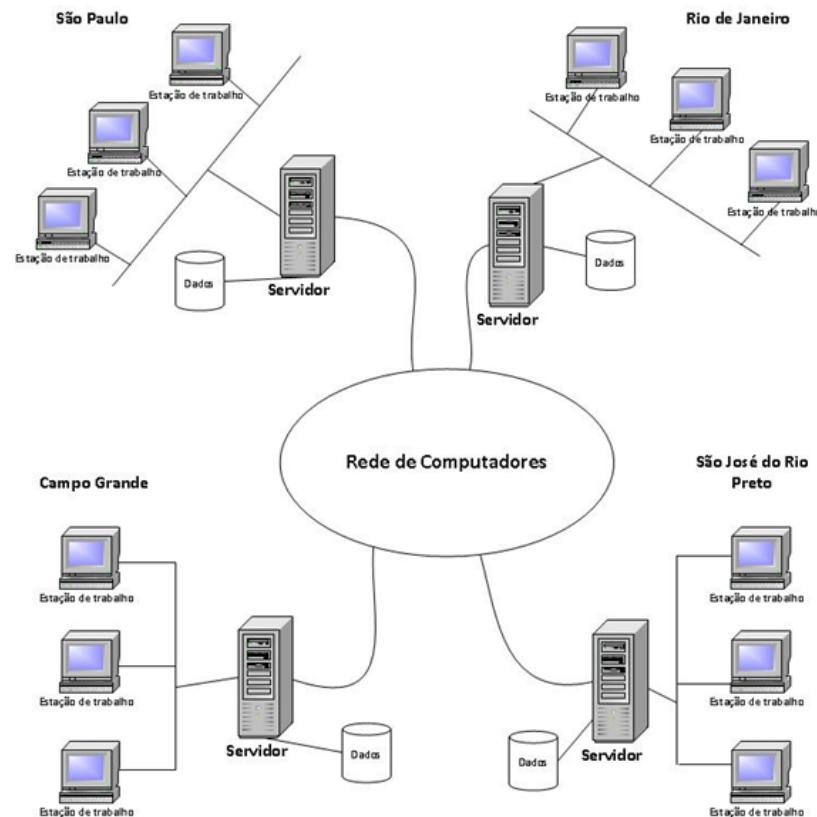
- Lidam com grandes volumes de dados distribuídos.
- Estruturas de dados menos rígidas, ideais para dados não tabulares.
- Exemplos: MongoDB (orientado a documentos), Cassandra, Redis (chave-valor).

| Key | Document |
|------|--|
| 1001 | <pre>{ "CustomerID": 99, "OrderItems": [{ "ProductID": 2010, "Quantity": 2, "Cost": 520 }, { "ProductID": 4365, "Quantity": 1, "Cost": 18 }], "OrderDate": "04/01/2017" }</pre> |
| 1002 | <pre>{ "CustomerID": 220, "OrderItems": [{ "ProductID": 1285, "Quantity": 1, "Cost": 120 }], "OrderDate": "05/08/2017" }</pre> |

Tipos de Banco de Dados

Bancos de Dados Distribuídos:

- Espalhados fisicamente em múltiplos locais.
- Fornecem alta disponibilidade e escalabilidade.
- Exemplos: Cassandra, Couchbase.



Tipos de Banco de Dados

Data Warehouses:

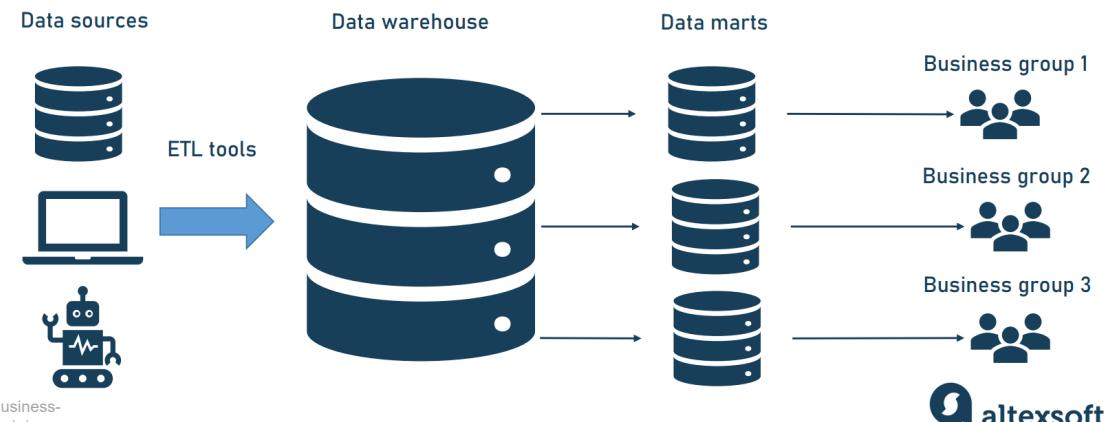
- Otimizados para análise e relatórios de grandes volumes de dados.
- Principalmente usados em business intelligence.
- Exemplos: *Amazon Redshift, Google BigQuery*.

Uso Comum:

Aplicações em que a análise extensiva de dados históricos é essencial para tomada de decisões estratégicas.

Considerações Importantes:

- Cada tipo de banco de dados tem vantagens e desvantagens.
- A escolha depende das necessidades específicas da aplicação e do tipo de dados a serem gerenciados.



Banco de Dados Relacionais

SQL (Structured Query Language): Linguagem utilizada para interagir com bancos de dados.

Funcionalidades Principais:

- Criação de Tabelas: Define a estrutura das tabelas no banco de dados.

Exemplo: `CREATE TABLE NOME_TABELA.`

- Consulta de Registros: Recupera dados específicos de tabelas.

Exemplo: `SELECT * FROM NOME_TABELA.`

- *Para atualizar registros:*

`UPDATE NOME-TABELA SET NOME-CAMPO=VALOR`

- Para inserir registros:

`INSERT INTO NOME-TABELA (CAMPO1, CAMPO2, CAMPO3)`

`VALUES (VALOR-CAMPO1, VALOR-CAMPO2, VALOR-CAMPO3)`



Conclusão

Neste curso, você irá conheceu o principais tipos de linguagem de programação, suas características e as diferenças entre Java e Python. Conheceu, também, o que são aplicações e como são utilizadas na web e por meio de inteligência artificial.

Por fim, aprendeu o que são metodologias de desenvolvimento e as características das metodologias ágeis. E, também, aprendeu o que são banco de dados e as caraterísticas do SQL.



Fonte: Adobe Stock

Referências



OKUYAMA, F. Y.; MILETTO E. M.; NICOLAO, M. **Desenvolvimento de Software**: conceitos básicos. [S. I.]: Bookman, 2014.

VELLOSO, F. F. **Informática**: conceitos básicos. Elsevier Brasil, 2014.

WAZLAWICK, R. **Engenharia de software**: conceitos e práticas. Elsevier Editora Ltda., 2019.

