

**Avalone Silva Cabrera**  
**Christian Martins Teixeira**  
**Gustavo Gutierrez**  
**João Henrique de Oliveira**  
**Leonardo Henrique dos Santos**

**HandTalk**

**MOGI MIRIM**

**2023**

## Sumário

1	Introdução .....	3
2	2.1.2 Mockup.....	4
	2.1.3 DIAGRAMA .....	6
2.3	Plataformas de desenvolvimento .....	8
	2.3.1 Plataforma Web.....	12
	2.3.2 Plataforma Desktop.....	19
	2.3.2 Plataforma Mobile .....	25
2.4	Banco de dados .....	30
	2.4.1 Modelagem de Dados .....	30
	2.4.2 SGBDS.....	42
2.4	Tecnologias Diversas .....	47
	2.5.1 Bootstrap.....	47
3	Desenvolvimento.....	53
3.1	Levantamento de Requisitos.....	53
	3.1.1 Entrevista/coleta de dados .....	53
4	Referências .....	62

## 1 Introdução

Com a evolução da tecnologia, aumentam as possibilidades de desenvolvimento de soluções que ajudem a melhorar a qualidade de vida das pessoas. Nesse sentido, desenvolver um software que reconheça a língua de sinais é um passo importante para tornar as coisas mais acessíveis e inclusivas.

A comunicação é uma das principais barreiras enfrentadas por esse segmento da população. Muitas vezes, a falta de compreensão da língua de sinais por parte do público leva à exclusão e isolamento dos surdos em diversas situações cotidianas, como reuniões de trabalho, eventos sociais e até mesmo consultas médicas.

Utilizar a tecnologia para facilitar a comunicação entre surdos e ouvintes pode ser uma importante solução para superar essas barreiras. Ao desenvolver um software que reconheça a língua de sinais, gestos e símbolos, onde podem ser traduzidos em texto, tornando a comunicação mais fácil e eficiente.

Para desenvolver este software, são necessárias certas linguagens e bibliotecas para aprendizado de máquina para reconhecer e interpretar gestos e símbolos. Deve-se ressaltar que o desenvolvimento desse tipo de tecnologia não é simples, pois envolve a análise de múltiplas variáveis, como velocidade do movimento, posição da mão e expressão facial do usuário.

Nesse contexto, propor um projeto abrangente de desenvolvimento desse software é importante para promover a inclusão e acessibilidade de pessoas surdas. Além disso, o projeto visa construir uma análise dos surdos do mundo, suas necessidades, desafios e oportunidades. Essa análise pode ajudar a desenvolver políticas públicas mais efetivas para promover a inclusão e acessibilidade de pessoas surdas em todo o mundo.

## 2 2.1.2 Mockup

O Mock-up é um modelo produzido mais para a produção do trabalho. Ele serve para mudar ideias em funcionalidades e ajudar um freguês a expandir e entender o quanto ele necessita.

Normalmente, temos uma ideia genérica do quão queremos, porém não temos certeza quão aos detalhes da produção, quanto por exemplo a posição de botões e de conteúdo, ou sobre quanto as páginas e funcionalidades irão interagir entre si.

Além disso, pode viver a espera de que um programador/desenvolvedor ou web designer contratado irá pensar nessas estruturas, em que época muitas vezes ele pretende levemente exibir um trabalho, de contrato com instruções pré-definidas. Essa dessemelhança de expectativas muitas vezes acaba em disputa.



Fonte: <https://blog.workana.com/pt/sin-categorizar/a-importancia-do-mock-up-nos-projetos-de-ti/>

Ao se trabalhar com um mock-up, o freguês expõe o profissional ao que ele tem em mente e consegue perceber melhor o coeficiente de dificuldade do trabalho. Isto

também facilita a ligação e a produção do projeto depois da contratação. (Gabrielle Lopes, 2021)

O mock-up também permite que seu fabricante seja testemunhado por um empenho demasiado menor. Imagine que você descobre que os seus clientes não se adaptaram com saúde a uma peça do seu programa levando ao encerramento do trabalho. O valor e época para alterações nessa fase é enorme e isto pode evitar com uma fase de testes por meio do mock-up.

Dessa forma, você pode iniciar uma parte da programação neste instante sabendo que seu produto tem boa usabilidade e que foi admitido pelo seu freguês final. A maior peça dos gerentes de projetos trabalha dessa forma. (Lopes, 2021)

### **Como fazer seu mock-up:**

Agora sabemos que a consideração do mock-up, é essencial para saber que irão desenvolvê-lo. Há muitas ferramentas que possibilitaram quão você “brinque” com as suas ideias e simule as funcionalidades. Balsamiq e Pencil são bons exemplos.

Seu mock-up não terá planificação ou integração com fundamento de dados ou redes sociais, senão pode se igualar a vida isso tudo.

O caso de haver simplicidade ainda permite que você o crie, próprio sem saber programar, ou que pague alguém para fazê-lo por um custo mais baixo. De qualquer forma, é essencial que o freguês se envolva o máximo possível nesta etapa, pois só ele sabe o que realmente ele deseja.

Também aconselhamos que você não se preocupe com o layout nesse momento, deixe isso para um profissional de web layout. A ideia agora é armar a armação do seu projeto, não os detalhes.

Também não se preocupe se você não conceber seu produto final 100%. Faça o quanto você puder, e certamente esse material ajudará o profissional contratado a saber de forma melhor, também irá reduzir seus riscos de problema.

Esse caso também possibilitará que você mude de ideia quantas vezes quiser, envolvendo pouco que não te agrada. Muitas vezes, isso não é provável quanto o pedaço de programação que foi iniciada. (Lopes, 2021)

### 2.1.3 DIAGRAMA

#### CASO DE USO:

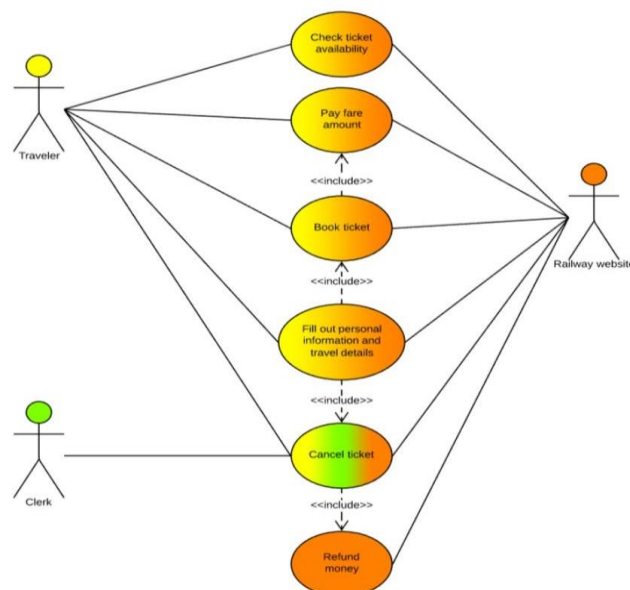
Bom nós iremos começar com o caso de uso que vai auxiliar você no levantamento dos seus requisitos funcionais do seu sistema, ele irá descrever o seu conjunto de funcionalidade do sistema e sua interação com os seus elementos externos entre si.

Cenários: quando nós falamos sobre o caso de uso, nós temos que ter em mente o conceito de um cenário, que no caso seriam as instâncias de uso.

O cenário poderá ser compreendido com a sequência dos passos que irá descrever uma interação entre o usuário do sistema.

#### Quando usar o diagrama de caso de uso:

Este diagrama de caso de uso não oferece muitos detalhes e não espere, por exemplo, que ele mostre uma ordem em que os seus passos serão executados.



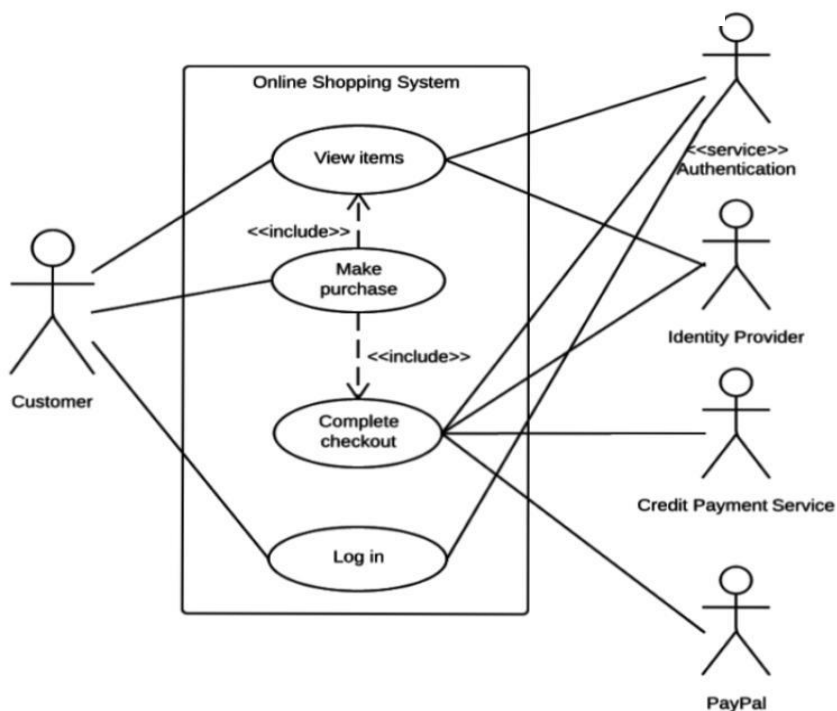
Fonte: <https://www.lucidchart.com/pages/pt/diagrama-de-caso-de-uso-uml#:~:text=um%20diagrama%20UML-,O%20que%20%C3%A9%20diagrama%20de%20caso%20de%20uso%3F,de%20s%C3%ADmbolos%20e%20conectores%20especializados>

Em vez disso, esse diagrama de caso será adequado e trará uma visão geral deste relacionamento entre um caso de uso como, atores e sistema.

Geralmente os profissionais desta área recomendam usar o diagrama de uso para você poder completar um caso de uso descrito em um único texto.

O diagrama de caso de uso também é ideal para:

- Poder representar as suas metas de uma interação feita no sistema do usuário.
- Define e organiza o requisito funcional no sistema.
- Especifica o contexto e o seu requisito do seu próprio sistema.
- Modela o seu fluxo básico de um evento no seu caso de uso.



Fonte: <https://www.lucidchart.com/pages/pt/diagrama-de-caso-de-uso-uml#:~:text=um%20diagrama%20UML-,O%20que%20%C3%A9%20diagrama%20de%20caso%20de%20uso%3F,d e%20s%C3%ADmbolos%20e%20conectores%20especializados>

## 2.3 Plataformas de desenvolvimento

“Plataforma de Desenvolvimento” é um termo usado para denominar plataformas que ajudam o usuário a construir e implementar de forma eficaz aplicativos personalizados com menos codificações do que o desenvolvimento tradicional.

Criado em 1970 por Winston Royce o modelo de desenvolvimento tradicional também conhecido como modelo cascata foi baseado em ordens de padrões derivados de diversos segmentos de engenharias dos anos 70.



Fonte: <https://www.devmedia.com.br/processo-de-teste-agil-x-tradicional/36854#:~:text=Metodologia%20tradicional,-Nesse%20contexto%20temos&text=Foi%20chamado%20de%20desenvolvimento%20tradicional,pequeno%2C%20m%C3%A9dio%20e%20grande%20porte>

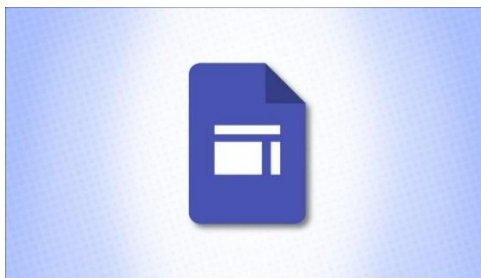
Foi chamado de desenvolvimento tradicional, pois é a base para diversos modelos utilizados há décadas pela indústria de software e é considerado um modelo rígido de pouca flexibilidade, adaptabilidade e versatilidade, sendo utilizado em projetos de pequeno, médio e grande porte.

### Plataforma de desenvolvimento de baixo código

Por meio de interações e fragmentos de códigos pré-construídos ou blocos, uma plataforma de baixo código permite que pessoas com pouco conhecimento sobre programação consigam desenvolver um aplicativo ou site de maneira especificada com uma interface visual de arrastar e soltar.

Dois sites que ganharam seu espaço entre plataformas de baixo código são o Google Sites e o Wix.com





Fonte: <https://maisgeek.com/como-usar-o-google-sites>

O Google Sites é uma ferramenta de criação de página da web incluída como parte do conjunto gratuito de editores de documentos do Google com base na web oferecido pelo Google. O “Google Docs., Google Drawings, Google Forms e Google Keep, Google Sheets, Google Slides.”



Fonte: <https://blog.ajasolucoesdigitais.com.br/a-verdade-sobre-wix/>

Wix.com foi fundada em 2006, em Israel, por Avishai Abrahami, Nadav Abrahami e Giora Kaplan. Consiste em uma plataforma online de criação e edição de sites, que permite aos usuários criar sites em HTML5 e sites Mobile sem necessidade de conhecimento de conhecimento prévio em programação ou design.

Além disso, a plataforma elimina a necessidade de criar estruturas, vincular bancos de dados e outras tarefas normalmente incluídas no desenvolvimento tradicional.

E torna o desenvolvimento mais fácil para que mesmo pessoas sem habilidades de programação possam desenvolver aplicativos por conta própria.

Com plataformas de baixo código, suas equipes de TI podem desenvolver aplicativos com mais rapidez e menos erros do que a codificação tradicional. Elas não precisam gastar meses construindo um aplicativo do zero, pois essas

plataformas fornecem componentes padronizados, como formulários, modelos de relatório e fragmentos de código prontos para uso.

Recursos a serem considerados em uma plataforma de baixo código

### **Modelagem visual**

Basicamente a modelagem visual é o uso de notações gráficas e textuais semanticamente ricas para capturar o design de um software.

A interface de modelagem visual de plataformas de baixo código permite que você veja a aparência do seu aplicativo conforme você o constrói.

### **Arrastar e soltar**

É um processo aonde a formatação e programação de um software é padronizada e facilitada para que não-programadores tenham acesso ao desenvolvimento de um website.

### **Segurança**

A parte de segurança está presente em todos os pilares de um processo de desenvolvimento de software mantendo sua privacidade e projetos privados de forma segura e confiável.

### **Implantação de vários dispositivos**

Hoje em dia a diversidade de marcas tecnológicas está presente no mercado popular então um software de fácil acessibilidade entre diversas máquinas como Windows, Mac e Linux é essencial. É essa questão também se encaixa nos dispositivos mobile como Android, IOS e Windows Phone.

### **Escalabilidade**

Conforme seus projetos vão crescendo funções vão sendo atribuídas a usuários o tornando administradores, especialistas e dessa forma seu projeto tende a crescer cada vez mais de forma segura e controlada.

### **Desenvolvimento mais rápido**

Negócios em rápida evolução exigem tecnologias que acompanhe as tendências de mercado que estão em constante mudança. E plataformas de desenvolvimento de baixo código pode ser essencial nessas situações. A vantagem que o desenvolvimento de baixo código oferece é sua capacidade de impactar significativamente a entrega de um novo software.

Os aplicativos desenvolvidos com plataformas de baixo código ajudam as organizações a se tornarem mais ágeis.

### **Produtividade aprimorada**

Tarefas manuais demoradas como atualizar os dados sempre que houver uma mudança dando assim mais tempo para o desenvolvedor se concentrar na codificação onde a engenhosidade humana é realmente necessária.

Como resultado, os projetos serão implementados com mais rapidez, eficiência e padrões mais elevados.

### **Gerenciamento de riscos**

Seus dados se tornam mais seguros permitindo o controle de todos os processos de gerenciamento de riscos em uma plataforma centralizada.

Simplificando assim a sua manutenção pois atualizar um aplicativo de baixo código para segurança de dados é fácil e economiza tempo.

### **Flexibilidade**

Uma plataforma de desenvolvimento de baixo código facilita a mudança de aplicativos e sua adaptação a novos requisitos.

Quer seja uma tarefa simples ou um fluxo de trabalho complexo, os usuários podem alcançá-lo sem escrever uma linha infinita de códigos. O baixo código fornece recursos que permitem aos usuários implementar novas estruturas ou modelos de sistema operacional rapidamente.

### **Experiência do cliente**

Aplicativos com um design agradável e criativos agradam seus clientes por ser visualmente atraentes, trazendo assim um bônus ao usuário final.

Também aumentando a compreensão e a capacidade de resposta do cliente diante ao seu aplicativo ou projeto.

### **2.3.1 Plataforma Web**

Assim como o nome já diz a plataforma web se baseia em uma relação harmônica de troca de informações entre quem produz com que consome podendo ser usada para fins acadêmicos, profissionais ou lazer.

## **Como criar uma plataforma web**

### **Principais passos**

Os passos a seguir não devem ser vistos como algo a se seguir à risca pois cada projeto tem sua particularidade como também sua finalidade e funcionalidades desejadas

### **Identifique os requisitos do projeto**

Identificar os requisitos do projeto não é nada mais do que manter constantemente certa interação com seu cliente afim de saber como ele idealiza e deseja sua web site por exemplo. Saber e identificar funcionalidades como a página de login ou até mesmo a barra de pesquisa.

### **Construir protótipo não funcional**

Também conhecido como wireframe, o protótipo pode ser definido como um esqueleto visual da plataforma, uma espécie de versão inicial ou então o termo “beta” a onde o protótipo oferece a base ao início do projeto.

### **Desenvolver a solução**

O Processo de desenvolvimento da solução é parcialmente dividida em 3 etapas

#### **Etapa 1: Elaborar as especificações das funções**

Implementar a interface de programação de aplicação (API) nessa etapa é fundamental por ter a possibilidade de integrar com outras plataformas, como mobile e desktop.

É função da API desempenhar toda a lógica do produto, assim como manter o domínio sobre os dados. É ela também que permite a integração com sistema de

terceiros, como ocorre quando há a necessidade de processamento de pagamentos ou do consumo de dados públicos.

O desenvolvimento de uma API pode ser aplicado por diversas linguagens de programação, mas Javascript e engine node são muito utilizadas e ganham preferência em diversas empresas como na Cubos. (Cubos.io)

## **Etapa 2: Elaborar e implantar banco de dados**

Planejar como os dados serão armazenados no banco e em como eles serão expostos ao usuário deve ser visto e revisto por conter informações privadas e pessoais. Também deve ser pensado no formato de otimização para que haja uma interação ágil entre o sistema e o banco de dados.

## **Etapa 3: Programar a Solução**

A partir do wireframe introduzir o layout por meio do front-end entram em cena para eu possam estruturar a navegabilidade da plataforma e fazer sua responsividade. Juntando assim as peças e atributos já prontos e dando um passo para o fim do projeto.

## **Testar e corrigir solução**

A melhor forma de encontrar bugs que passaram despercebidos no momento do desenvolvimento é por meio de testes e atribuições mantendo assim a integridade do projeto durante o desenvolvimento.

Assim como apps de games tem sua versão de teste conhecida como “versão beta” as plataformas web também tem sua versão de testes que são entregues a usuários que não fazem parte da equipe de desenvolvedores com a intenção de encontrar novos bugs ou falhas.

Simultaneamente à execução esses testes, os bugs serão corrigidos e as melhorias serão implementadas de acordo o feedback advindo dos testes.

## Publicar

Publicar sua plataforma da forma correta é essencial para que ela atinja o público que você desenvolvedor deseja atingir, mas mesmo depois de passar por todos os testes de correção a plataforma ainda pode apresentar bugs, portanto manter seu projeto sempre atualizado e em constante desenvolvimento é essencial.

## Front-end e Back-end

Duas formas de desenvolvimento que se completam como um quebra-cabeça, mas com duas funções totalmente diferentes uma da outra.

Front-end: Assim como o nome já diz (parte dianteira) a programação em front-end se baseia no design e projeção do seu projeto. Em um site por exemplo os códigos em front-end vão modelar editando as cores, a fonte, imagens entre outras funcionalidades



Fonte: <https://meunegocio.uol.com.br/blog/html-css-ou-javascript-qual-aprender-primeiro/>

## HTML

### -Surgimento

Criado na Suíça no ano de 1991 o HTML foi projetado para interligar instituições de pesquisa próximas e facilitar o compartilhamento de documentos com facilidade

### -Proposito

O HTML não é considerado uma linguagem de programação e sim uma linguagem de marcação que tem como proposito a modelagem e definições de uma

página na web. Seus códigos se baseiam em comandos para definirem o tipo de letra, qual o tamanho, cor, espaçamento, e vários outros aspectos do site ou página da web

### **-Exemplos**

“Alguns códigos em HTML e suas funções:

`<title>` Define o título da página.

`<script>` Permite adicionar funções em páginas com script, podendo assim adicionar códigos em Javascript (Este código permite que alguns sites em HTML tenham joguinhos ou animações, verificações de formulários antes do envio para o servidor, entre outras funcionalidades)

`<style>` Define formatação em CSS.”

(trecho pesquisado no site Infoescola <https://www.infoescola.com/informatica/html/> )

### **-Motivos para aprender**

HTML é considerada uma das linguagens mais reconhecidas no mercado de desenvolvimento de sites e mesmo tendo diversos comandos e funções é uma linguagem simplificada e de fácil entendimento.

## **CSS**

### **-Surgimento**

Criado pelo norueguês Håkon Wium Lie em outubro de 1994 com o intuito de facilitar a programação de sites onde se era considerada muito complexa

### **-Proposito**

Teve como proposito inicial facilitar a programação transformando códigos gigantescos em pequenas frases ou compostos por uma única palavra, mas com a notoriedade do HTML o CSS também conhecido como (Cascading Style Sheets ) ficou à disposição para os Web designers por ser uma linguagem que determinava o layout de uma página web

## -Exemplos



Fonte: [https://developer.mozilla.org/pt-BR/docs/Learn/Getting\\_started\\_with\\_the\\_web/CSS\\_basics](https://developer.mozilla.org/pt-BR/docs/Learn/Getting_started_with_the_web/CSS_basics)

P → Seletor abre colchetes para a apresentação das declarações e propriedades

{ } → Colchetes servem para caracterizar as propriedades e declarações

Color → Comando da propriedade para anteceder a o comando da cor

Red → valor da propriedade indica vermelha na seguinte declaração

## -Motivos para aprender

O CSS vem para facilitar a vida de nós programadores, trazendo assim um melhor manuseio de formatação e melhorando nossa forma de programar deixando-a mais eficiente e de forma eficaz

## Java Script

### -Surgimento

Criado pelo americano Brendan Eich desenvolvedor da empresa Netscape em meados de 1995 o JS é uma linguagem de programação utilizada inicialmente para implementar itens complexos a uma página da web.

### -Proposito

JavaScript é uma linguagem de programação utilizada não só para incrementar arquivos e complexidades a sua página web como também é utilizado para o desenvolvimento de aplicativos para a Web.



## -Exemplos

`<Script>` → Utilizado para iniciar a programação em JS

`</Script>` → Utilizado para finalizar a programação em JS

`//` → Indica um comentário

If (...) -> usado para comparar uma variável com uma string e verificar se a condição seja verdadeira (TRUE) ou se uma condição é falsa (FALSE)

Else (...) -> trabalha junta ao comando “if(...)” identificando condições falsas

## -Motivos para aprender

O Grande diferencial da linguagem JS é que ele permite ser desenvolvido dentro do código HTML, sendo assim conforme você avança no desenvolvimento do seu site basta colocar o comando “<script>” para iniciar a programação em JS e ao finalizar basta utilizar o código “</script>”.

Back-end: Assim como o próprio nome sugere, a programação em Back-end é baseada na parte por trás de um site ou aplicação, fazendo assim a ponte entre os dados abordados pelo navegador e o banco de dados na maioria dos casos.

Usuários comuns não tem acesso livre a Back-end, assim garantindo assim que não haja uma manipulação dos dados ou acesso direto ao banco de dados aonde este usuário em questão teria acesso a informações pessoais de clientes catalogados.

## Linguagens adeptas a programação em Back-end:



## **Node.js**

### **-Surgimento**

Lançado no ano de 2009 pelo pesquisador Ryan Dahl, a criação do node JS veio como uma necessidade pelo fato do Java Script não suportar uma sobrecarga de requisições, o que muitas das vezes geravam o bloqueio ao carregamento de arquivos ou processamentos pesados.

### **-Proposito**

Sua proposta inicial era facilitar o modo que usuários lidavam com os recursos da máquina e consumir apenas os recursos necessários de desenvolvimento. Assim acelerando o desempenho de suas aplicações.

Hoje você consegue desenvolver em node JS através de diversas plataformas como Windows, Mac e Linux possibilitando assim diversos tipos diferentes de aplicações Web em apenas um código Java script.

O Node.Js proporciona ao desenvolvedor diferentes possibilidades de desenvolvimento a serem feitos na Web, portanto, o Node.js é apreciado como uma plataforma colaborativa, ágil com as seguintes características:

Responde às requisições das aplicações;

Tecnologia gratuita e livre de licenças;

Front-end e Back-end;

Código aberto: possibilita uma comunidade ativa que contribui com o projeto;

Programação JavaScript: linguagem popular da web;

Performance: baixo consumo de memória e aproveitamento da CPU” (X-apps, 2019)

### **-Motivos para aprender**

Um dos motivos para se aprender Node.js é pelo seu fácil manuseio, facilitando assim sua interação e permitindo programar em Javascript no lado cliente e servidor.

Essa questão é muito significativa no mercado do desenvolvimento de projetos, pelo fato de que a maioria dos apps e sites são desenvolvidos por uma equipe e o node.js facilita a interação e a comunicação entre membros da equipe e apenas esse diferencial mudou a percepção do mercado sobre Node.js

A utilização do V8 JavaScript Engine e seu design assíncrono permitem uma otimização considerável com respostas imediatas ao seu projeto, concluindo assim projetos sofisticados sem necessariamente aumentar a complexidade do seu código.

### **2.3.2 Plataforma Desktop**

As aplicações desktop são aplicações/softwarewares que podem ser instalados e executados em um computador usado para realizar tarefas específicas. Algumas aplicações feitas para Desktop podem ser utilizadas por diversos usuários em um único ambiente de rede. Entretanto o desenvolvimento de aplicações web começou a tomar conta, substituir ou completar as aplicações de Desktop, que eram difíceis de serem vendidas, mantidas, compradas, atualizadas.

## **Python**

Python foi criado no final dos anos oitenta por Guido Van Rossum no Centro de Matemática e tecnologia da Informação (CWI), na Holanda. O nome da língua veio em referência a uma série de televisão chamada Monty Python do qual o criador gostava. Van Rossum é o principal autor de Python. Python é uma linguagem de programação interpretada da qual o objetivo é focado em trazer uma sintaxe que favoreça a legibilidade do código, além de ser “free”. Atualmente, a linguagem é gerenciada pela Python Software Foundation. Além da sintaxe simples, a linguagem de programação é também multi-paradigma, pois suporta orientação de objeto, programação imperativa e, em menor escala, programação funcional. Usa uma tipagem forte e dinâmica, o que permite o próprio interpretador entenda o tipo da variável sem que o usuário precise declarar. (Costa, 2023)

## O que é uma linguagem com tipagem forte e dinâmica?

Tipagem dinâmica significa que o tipo de dado que a variável irá receber pode mudar ao longo do código, já a linguagem com tipagem estática é justamente o contrário, uma vez que o valor for declarado como inteiro ele não pode mudar o tipo no decorrer do código, a não ser que haja uma intervenção do desenvolvedor, como por exemplo:

Tipagem dinâmica:

```
nome = "Gustavo"

nome = 1;

print(nome)

Resultado: 14
```

Fonte: Próprio Autor

Resultado: 14

Tipagem estática:

```
nome = "Gustavo"

nome = 1;

print(nome)

Resultado: 14
```

Fonte: Próprio Autor

Resultado: Erro, o valor "nome" já foi definido como String, linha 1.

Python além de ter uma tipagem dinâmica ele também é uma linguagem fortemente tipada, ou seja, caso o desenvolvedor, no código, queira realizar uma operação matemática entre dois valores com tipos distintos lhe resultará em erro ou exceções, portanto essas operações somente podem ser realizadas através de variáveis do mesmo tipo ou com a realização da conversão (casting) de valores. (Horn, 2022)

Exemplo:

```
1 num1 = 1
2
3 num2 = "2"
4
5 total = num1 + num2
6
7 print(total)|
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
/bin/python3 /home/gutierrez/Documentos/Programming/TCC.py
gutierrez@GUTIERREZ ~/Documentos/Programming
% /bin/python3 /home/gutierrez/Documentos/Programming/TCC.py
Traceback (most recent call last):
  File "/home/gutierrez/Documentos/Programming/TCC.py", line 5, in <module>
    total = num1 + num2
TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

Fonte: Próprio Autor

### Resultado: Erro

Para o compilador é impossível somar um número com uma variável de texto.

Python também é uma linguagem multiplataforma, podendo ser utilizada e desenvolvida em Windows, GNU/Linux, UNIX e MacOS/Apples, assim como outras linguagens, indo desde supercomputadores até telefones celulares. Além de ter alta portabilidade ela é bem poderosa tendo uma infinidade de bibliotecas onde você pode encontrar coisas desde criações de jogos até web development. E o melhor de tudo, Python é uma linguagem de programação gratuita, assim como as principais encontradas dentro do mercado. Python foi feita tendo base na linguagem ABC, parte da sintaxe derivada do C e compreensão de listas, funções anônimas e função map derivadas da linguagem Haskell.

### Principais características da linguagem:

1. Fácil de aprender
2. Fácil leitura e compreensão
3. Fácil manutenção

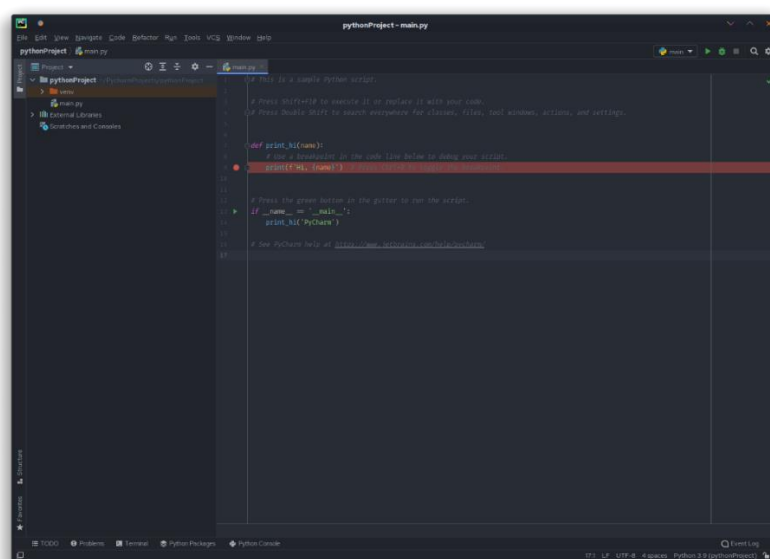
4. Multiplataforma
5. Modo interativo

### Curiosidades sobre a linguagem:

1. Fácil de ler, escrever e ser mantida;
2. StackOverflow considerou Python a linguagem de renome que mais cresce, e a linguagem que os programadores mais desejam aprender;
3. Eficiente para fazer protótipos;
4. Altamente escalável e poderosa;
5. Open-Source;
6. Não utiliza ponto e vírgula (;) para finalizar uma instrução;
7. Uma variável pode armazenar diferentes tipos de dados;
8. Não há chaves ({} ) para delimitar o início e final de um bloco de código.

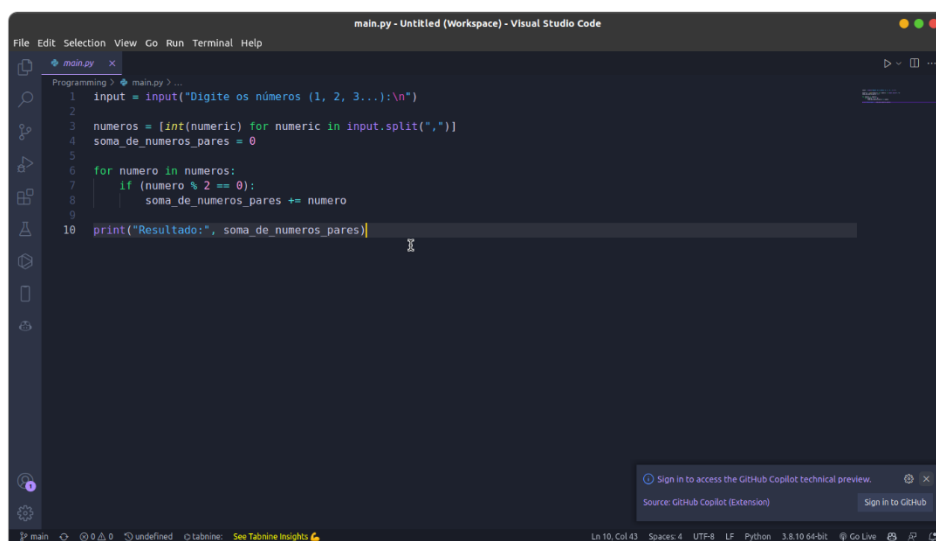
### Onde eu posso programar em Python?

Python pode ser instalado diretamente do site oficial da própria organização (python.org), assim como Java, Python pode ser desenvolvido dentro de ide's feitas especificamente para a linguagem como o JetBrains PyCharm:



Fonte: Próprio Autor

E é claro em editores de código como o VSCode:



Fonte: Próprio Autor

### Motivos para aprender Python:

1. Facilidade: Python além de ser uma linguagem altamente poderosa ela é bem simples. Utilizando uma sintaxe simples, ela permite realizar operações com muito mais facilidade do que linguagens complexas como Java ou C++;
2. Aumento de produtividade: Python é uma linguagem bastante intuitiva e consequentemente, irá gerar menos erros no código, tornando assim o processo mais rápido. Além disso, com um conhecimento prévio/básico de Python, você já consegue automatizar tarefas e processos que teria de fazer a mão.
3. Comunidade ampla e popularidade: Python é uma linguagem cuja comunidade é altamente ativa e consequentemente os materiais didáticos são mais amplos e acessíveis na internet. Sem contar que existem diversas bibliotecas comunitárias feitas para Python disponibilizadas gratuitamente por diversos usuários que possibilitam executar funções ainda mais elaboradas.

4. Utilização versátil: Python nos permite executar diversas operações integradas com outras linguagens de programação (cross-language), como Java, bibliotecas C e C++ e componentes .NET.
5. Automação: Além de tudo Python é uma ótima escolha para aqueles que querem automatizar tarefas no seu dia-a-dia. Uma vantagem é que os scripts de Python, pode ser usado por outros scripts, como se emprestassem funções para outros projetos, importando-as como bibliotecas.

Sintaxe Python:

Exemplo de programa que mostre a mensagem “Olá, mundo! ”

```
1 print ('Olá, Mundo!')
```

Fonte: Próprio Autor



### 2.3.2 Plataforma Mobile

Desenvolvimento mobile não se trata de apenas construir aplicativos para celulares e sim de fazer qualquer tipo de desenvolvimento para qualquer dispositivo móvel. As principais características desse tipo de desenvolvimento é a portabilidade, a conveniência e a praticidade.

#### Principais plataformas do desenvolvimento Mobile



Fonte: Imagem: [https://pt.m.wikipedia.org/wiki/Ficheiro:Apple\\_logo\\_black.svg](https://pt.m.wikipedia.org/wiki/Ficheiro:Apple_logo_black.svg)

### iOS

O iOS é um sistema operacional móvel criado e desenvolvido pela Apple. A cada ano é lançado uma nova versão do sistema operacional, que atualmente (2022) está na sua versão iOS 15.4.1. A empresa também não permite que o Sistema operacional seja utilizado em terceiros. A Apple foi a primeira empresa a introduzir o uso dos dedos para gerir a interface, descartando o uso de canetas para isso. Foi a primeira plataforma a trazer o desenvolvimento móvel para os dias atuais renovando a ideia do que é um dispositivo móvel e software.

#### O que é preciso para desenvolver em iOS?

Conhecimento básico em Swift

Swift é uma linguagem de programação desenvolvida pela Apple focada em desenvolvimento mobile com exclusividade do sistema operacional IOS. Com muitas características modernas, a linguagem de programação Swift é a linguagem de programação muito expressiva e flexível. Além disso aceita funções como objetos de

primeira classe, ou seja, você pode utilizar outras funções como parâmetros de outras funções.

### **Familiaridade com ides**

Como a Swift tem acesso a bibliotecas da Cocoa, interface de desenvolvimento de apps para computadores da Apple, é necessário ter conhecimento e experiência para trabalhar em plataformas e ferramentas como Xcode ou AppCode, por exemplo. Além de trabalhar com a Swift para o desenvolvimento de aplicações mobile iOS, você também pode desenvolver as mesmas em Objective-C. Uma linguagem orientada a objetos, também uma das linguagens mais antigas quando o assunto se trata de programação, estruturada e desenvolvida pela StepStone em 1980.

### **Ambiente de desenvolvimento IOS**

Além de algumas opções de ides para desenvolvimento de aplicações IOS a mais utilizada dentre os programadores de aplicativos IOS, o Xcode é a IDE mais utilizada, mas para ser utilizada é necessário que você possua a máquina específica para o sistema iOS, que no caso, podem ser dispositivos e modelos OS X, Macbook Pro Macbook Air, Macbook Mini e derivados.

### **Pontos negativos**

- Custos;
- A própria linguagem;
- Tempo.

Infelizmente para que seja possível o desenvolvimento de aplicações iOS com Swift e Xcode é necessário que você tenha ao menos um dispositivo da Apple como algum Macbook ou Macintosh (iMac), o que pode gerar alguma barreira socioeconômica e na maioria das vezes menos popular. Normalmente os custos para adquirir um Macbook por exemplo, pode variar de 5 a 9 mil reais dentre os mais antigos, enquanto os mais recentes ficam na faixa de 9 a 18 mil reais.



Fonte: [https://commons.wikimedia.org/wiki/File:Android\\_logo\\_2019.png](https://commons.wikimedia.org/wiki/File:Android_logo_2019.png)

## Android

A é o nome do sistema operacional baseado em Linux que pode ser utilizado em celulares, netbooks e tables. É desenvolvida pela Open Handset Alliance, um grupo organizado por várias empresas, dentre elas a Google.

A principal concorrente do iOS, o Android foi lançado em 2008, mas o Sistema Operacional mesmo começou muito antes disso, em outubro de 2003. Possui serviços bem completos com a marca Google, por exemplo, quando você compra algum produto do qual o sistema operacional é Android, todos os aplicativos são interconectados com outros sistemas da Google, além disso o dispositivo já vem com alguns apps pré-instalados como o Google Play, Google Play Filmes e outras funcionalidades exclusivas da google, facilitando a vida do usuário com a experiência de uso.

Além do fato de ser um sistema operacional open-source o Android traz muitas vantagens para seus utilizadores tais como: Armazenamento-extra, variedade de aparelhos, Atualizações rápidas, Integração com aplicativos Google, Backup fácil.

Onde desenvolver aplicações Android?

O Android Studio é uma das IDEs mais populares quando o assunto é desenvolvimento mobile para Android. Ela oferece muitas ferramentas que contém o SDK, debuggers e emuladores necessários para o desenvolvimento, além disso, outra vantagem de utilizar o Android Studio é que você pode utilizar em qualquer sistema

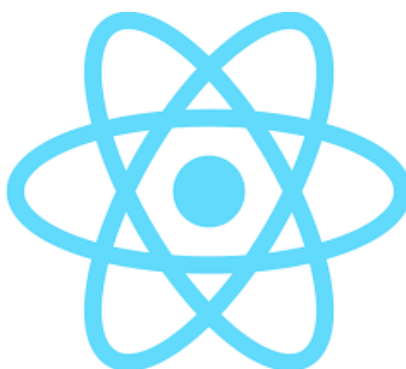
operacional, sendo Mac, Windows ou Linux. Junto à JetBrains o Android está desenvolvendo uma nova linguagem para aplicações em android chamado Kotlin.

**Vantagens:**

- Sistema operacional Open-Source;
- Funciona de forma leve junto aos aparelhos com mais hardware;
- Possui touch sensível em análise a outros aparelhos antigos;
- Aplicativos grátis na Android Market;
- Conexão à rede social;
- Conexão à internet;
- Aparelhos mais baratos nas operadoras.

**Desvantagens:**

- Vírus na loja Android Market;
- Problemas na atualização;
- Aplicativos funcionam para alguns aparelhos e para outros não funcionam;
- Falta de atualização do sistema operacional para o aparelho;
- Aplicativos pagos mais caros que em outras lojas.



Fonte: <https://reactnative.dev/>

**React Native**

React Native é uma biblioteca JavaScript criada pelo Facebook. É usada para desenvolver aplicativos para os sistemas Android e iOS de forma nativa. Criada em 2015 sobre a licença MIT, o React Native é um Framework de JavaScript para facilitar o desenvolvimento de aplicações mobile, facilitando muito a vida de muitos desenvolvedores que atuam na área. Com o React Native é possível criar aplicações

híbridas, ou seja, com um único código o desenvolvedor pode criar um app para diferentes plataformas, ela permite acessar a interface e os recursos nativos do Android e iOS utilizando o próprio JavaScript, o código é produzido de forma semelhante ao React para web e possui a base de conhecimento compartilhada entre o desenvolvimento mobile e front-end.

```
1  import React from 'react';
2  import { Text } from 'react-native';
3
4  export default function Ola() {
5    return (
6      <Text>Olá, mundo!</Text>
7    );
8  }
```

Fonte: Próprio Autor

### **Vantagens de utilizar React Native para o desenvolvimento de aplicações mobile:**

- Desenvolvimento Econômico;
- Entrega mais rápida de projetos de aplicativos;
- JavaScript;
- Requer equipes menores;
- Open-Source;
- Hot Reloading;
- Comunidade de desenvolvedores ativa;
- Excelente desempenho do aplicativo;
- Aparência Nativa;
- Design Modular.

**Desvantagens:**

- Desafios de depuração e compatibilidade;
- Requerimento de desenvolvedores nativos;
- Depende do Facebook;
- Gerenciamento de memória não é excelente;
- Problemas com licença e patente;
- Adoção lenta das funções mais recentes;
- Desafios de segurança de JavaScript;
- Desempenho vs. Problemas de recursos.

**2.4 Banco de dados**

Banco de dados de acordo com (Laundon, et al., 2014) é caracterizado como uma coleção de dados interligados de maneira organizada que se relacionam para dar mais eficiência durante uma pesquisa ou estudo científico, seu principal objetivo como o próprio nome diz é armazenar e salvar os dados utilizados.

No começo, cada entidade era um arquivo de dados que vinha com um software bem básico apenas para manipular os dados dessa entidade, nesses softwares era possível realizar operações de cadastro, alteração, exclusão e consulta de arquivos digitais

O primeiro projeto de banco de dados foi projetado pela IBM (International Business Machines Corporation), na década de 60, época na qual eles fizeram grande investimento para solucionar problemas desse banco de dados digital primitivo, nesse momento surgiram vários modelos de banco de dados

**2.4.1 Modelagem de Dados**

A modelagem de dados ou modelagem do banco de dados consiste em aplicar técnicas teóricas e práticas para a realização de um banco de dados consistente, não redundante e que se encaixe em qualquer tipo de Sistema Gerenciador de Banco de Dados. Ao realizar essa modelagem você criará uma representação visual dos dados conforme são usados (Instituto Federal Santa Catarina, 2015)

A modelagem de dados está dividida em 3 fases: Modelo Conceitual, Modelo Lógico e Modelo Físico, abaixo segue a definição de cada uma:

### Modelo Conceitual

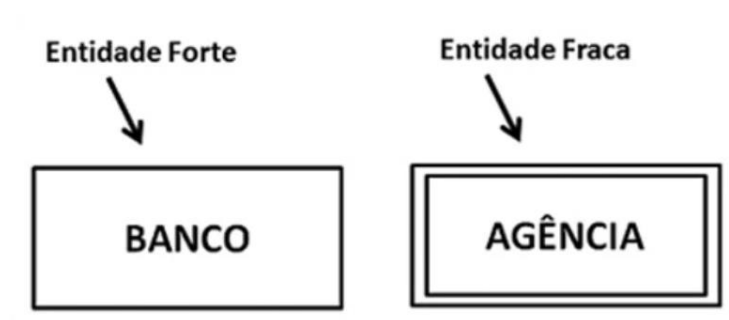
O Modelo Conceitual é o nível mais alto da abstração, seu foco é descrever as entidades envolvidas em um negócio, definindo seus atributos e o modo que se relaciona com outras entidades. O DER (Diagrama Entidade Relacionamento) é um exemplo de Modelo Conceitual criado em forma gráfica, na qual se deve identificar todas as entidades e relacionamentos de um determinado “universo”. **Fonte bibliográfica inválida especificada.**

Sua principal função é capturar os requisitos de informação e regras de negócio sob o ponto de vista do negócio, por isso o Modelo Conceitual deve ser feito na fase de levantamento de requisitos onde o cliente está inserido.

Abaixo estão os principais componentes do modelo conceitual e suas respectivas definições de acordo com (Lopes, 2022):

### Entidades

Uma entidade é a representação de um objeto que existe e é distinguível de outros, por exemplo entidades podem ser pessoas, lugares, objetos, profissões etc. As entidades geralmente são representadas por um Retângulo



Fonte:

[https://static.wixstatic.com/media/e523ba\\_51dbdff232d4e6c944f0da6b316e3e9.jpg/v1/fill/w\\_525,h\\_226,al\\_c,lg\\_1,q\\_90/e523ba\\_51dbdff232d4e6c944f0da6b316e3e9.webp](https://static.wixstatic.com/media/e523ba_51dbdff232d4e6c944f0da6b316e3e9.jpg/v1/fill/w_525,h_226,al_c,lg_1,q_90/e523ba_51dbdff232d4e6c944f0da6b316e3e9.webp)

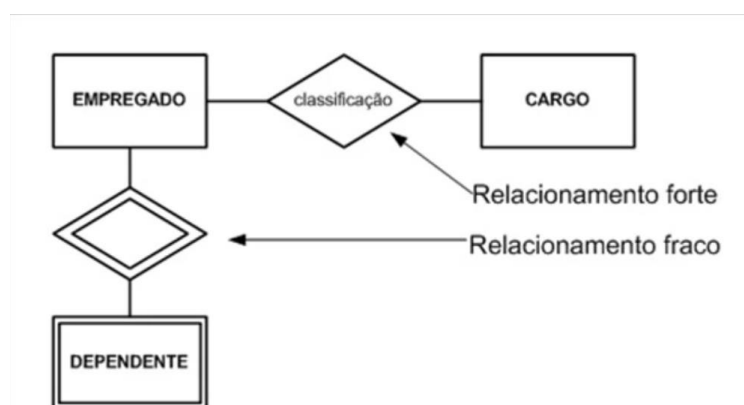
As entidades podem ser classificadas em dois tipos de Entidades: Entidades Fortes e Entidades Fracas

As entidades fortes são aquelas que são independentes, que não dependem de nenhuma outra entidade para a identificação para serem criadas, elas dependem apenas delas mesmas. Já a entidade fraca é aquela que depende da existência de outra entidade para ela existir/ se identificar, um exemplo pode ser um Dependente de um Funcionário de uma Empresa, pois só vai haver “Dependente” se tiver um funcionário

## Relacionamentos

Relacionamentos é a maneira que as entidades se ligam, elas geralmente são ações que associam/conectam as entidades. Assim como as entidades, os relacionamentos devem expressar um significado dentro do “universo” modelado.

Os relacionamentos podem ser representados através de um losango, eles também são divididos em 2 relacionamentos: Relacionamento forte x Relacionamento fraco



Fonte:

[https://static.wixstatic.com/media/e523ba\\_df6d9ff350be4d1696adaeb596d301cc.jpg/v1/fill/w\\_671,h\\_385,al\\_c,lg\\_1,q\\_90/e523ba\\_df6d9ff350be4d1696adaeb596d301cc.webp](https://static.wixstatic.com/media/e523ba_df6d9ff350be4d1696adaeb596d301cc.jpg/v1/fill/w_671,h_385,al_c,lg_1,q_90/e523ba_df6d9ff350be4d1696adaeb596d301cc.webp)

Relacionamento fraco irá existir quando houver uma entidade fraca, isso se deve pelo fato de tanto a entidade quanto o relacionamento dependem de outra entidade para sua “existência”, já o relacionamento forte ele não depende de entidade para existir, ele é apenas usado para conectar as entidades

Ao criarmos relacionamentos no modelo conceitual devemos aplicar 1 tipo de conceito na hora da criação do mesmo: a cardinalidade



## Cardinalidade

Cardinalidade corresponde a representação de quantas vezes um determinado elemento de uma entidade pode estar associado a um relacionamento e a outras entidades. Ela pode ser definida em cada um dos lados do relacionamento e é representada por um conjunto  $(x, y)$  onde  $x$  é a cardinalidade mínima e o  $y$  a máxima, porém em alguns mapeamentos é comum que apareça apenas cardinalidade máxima.

A cardinalidade mínima é utilizada para representar uma obrigatoriedade/opção do relacionamento. Enquanto isso a cardinalidade máxima define a quantidade máxima de vezes que o elemento da entidade pode estar associado no relacionamento.

O uso da cardinalidade é obrigatório, pois é a partir da cardinalidade que se refletem regras que devem ser representadas em um modelo conceitual e até mesmo no mapeamento do modelo lógico.

Existem 3 tipos de Cardinalidade:

0: Geralmente usado na cardinalidade mínima indica a opção do elemento da entidade no relacionamento;

1: Quando na cardinalidade mínima indica a obrigatoriedade de no mínimo 1 elemento da entidade no relacionamento, e para cardinalidade máxima indica que o máximo vai ser elementos da entidade que esse relacionamento vai ter é 1;

N: Geralmente usado na cardinalidade máxima indica que pode haver mais de 1 elemento da entidade no relacionamento;

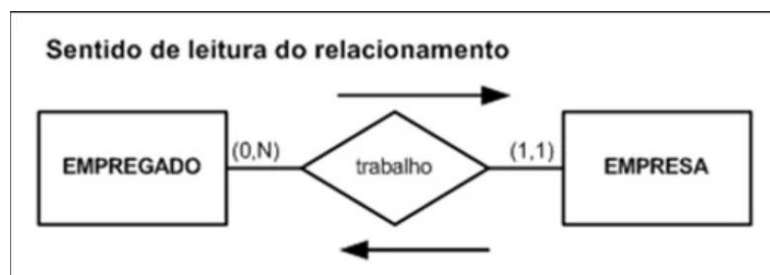
A partir desses 3 tipos de cardinalidade, existem 3 tipos de relacionamento

Relacionamento 1 para 1 (1:1): onde apenas 1 elemento da entidade A se relaciona apenas com 1 elemento da entidade B

Relacionamento 1 para vários (1:N): onde apenas 1 elemento da entidade A relaciona-se com vários elementos da entidade B

Relacionamento vários para vários (N:N): onde vários elementos da entidade A relaciona-se com vários elementos da entidade B

### Exemplo de Cardinalidade em um DER



Fonte:

[https://static.wixstatic.com/media/e523ba\\_bca6c4a57c1c440e9a3fa41f0b0c9518.jpg/v1/fill/w\\_525,h\\_186,al\\_c,lg\\_1,q\\_90/e523ba\\_bca6c4a57c1c440e9a3fa41f0b0c9518.webp](https://static.wixstatic.com/media/e523ba_bca6c4a57c1c440e9a3fa41f0b0c9518.jpg/v1/fill/w_525,h_186,al_c,lg_1,q_90/e523ba_bca6c4a57c1c440e9a3fa41f0b0c9518.webp)

A figura 55 explica que um empregado pode trabalhar e trabalha em apenas uma empresa e que uma empresa pode ter nenhum ou vários empregados, e mostra também o sentido da leitura da cardinalidade.

### Atributos

Um atributo pode ser considerado uma “característica” da entidade, eles são responsáveis por atribuírem informações que vão caracterizar a entidade, geralmente são representados por pequenos círculos que são ligados a entidade na qual eles pertencem.

Atributos podem ser divididos em 4 tipos:

#### Atributo Identificador

O atributo identificador, como o próprio nome diz é responsável por identificar uma entidade, geralmente são dados únicos como: Códigos ou Registros. O mesmo é representado por um círculo preenchido no Modelo Conceitual

#### Atributo Simples

O Atributo não identificado ou atributo simples, corresponde a maioria dos atributos, ele não possui nenhuma característica especial, exemplos de atributos simples são: nome, datas. Ele é representado por um círculo sem preenchimento no Modelo Conceitual

## Atributo Multivalorado

Os atributos multivalorados, são usados quando você pode ter vários itens para um mesmo campo, como é o caso do número telefone que uma pessoa pode ter mais de um número, esse atributo é recomendado apenas no Modelo Conceitual para abstrair os dados necessários. O Atributo Multivalorado é representado por um círculo com um asterisco do lado.

## Atributo Composto

Utilizamos o atributo composto, quando o atributo que estamos a usar pode ser subdividido em partes menores, um exemplo simples, um atributo endereço, pode ser subdividido em Rua, Cidade, Bairro etc. Igual ao Atributo multivalorado seu uso é recomendado apenas no Modelo Conceitual, e é representado por um atributo que recebe outros atributos “menores”.

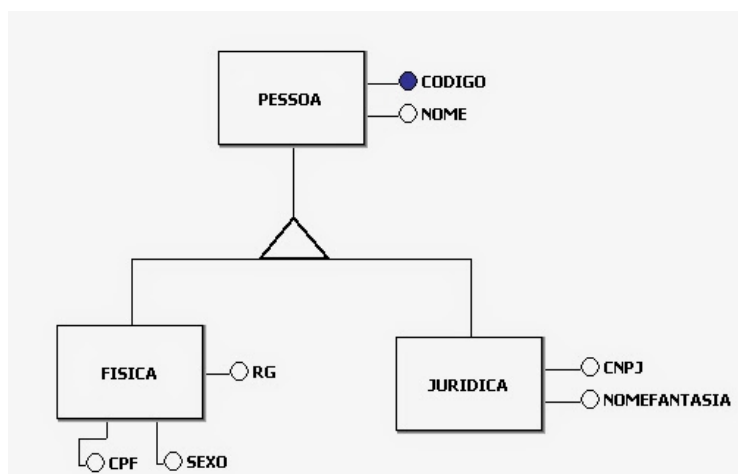


Fonte:

[https://static.wixstatic.com/media/e523ba\\_43d4d62fda874612abc4ddddd2ed960f.jpg/v1/fill/w\\_378,h\\_465,al\\_c,lg\\_1,q\\_90/e523ba\\_43d4d62fda874612abc4ddddd2ed960f.webp](https://static.wixstatic.com/media/e523ba_43d4d62fda874612abc4ddddd2ed960f.jpg/v1/fill/w_378,h_465,al_c,lg_1,q_90/e523ba_43d4d62fda874612abc4ddddd2ed960f.webp)

## Generalização e especialização

A Generalização e Especialização consiste em separar hierarquicamente entidades que terão subdivisões, dependendo do caso de uso podemos notar que uma entidade poderá ser subdividida, onde elas ainda serão a mesma entidade, porém com atributos e nomes diferentes, exemplo, em determinado caso de uso temos uma entidade pessoa e no próprio caso é dito que as pessoas ou são pessoas jurídicas, ou pessoas físicas, logo teríamos que dividir os “tipos” de pessoa



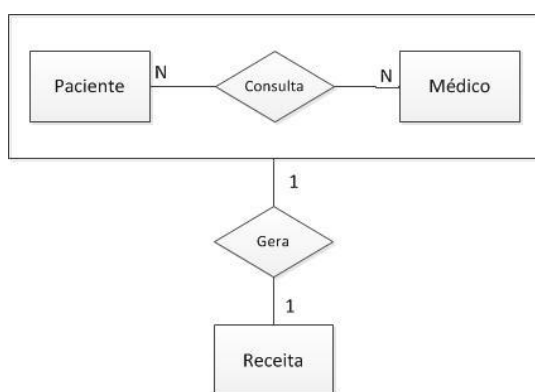
Fonte: [https://lh6.googleusercontent.com/-](https://lh6.googleusercontent.com/-6UrPk6Hpytg/U0IYZ5okm1I/AAAAAAAAANI/GLbESkpw2Pw/s504/genespetotal.jpg)

[6UrPk6Hpytg/U0IYZ5okm1I/AAAAAAAAANI/GLbESkpw2Pw/s504/genespetotal.jpg](https://lh6.googleusercontent.com/-6UrPk6Hpytg/U0IYZ5okm1I/AAAAAAAAANI/GLbESkpw2Pw/s504/genespetotal.jpg)

A figura 57 representa como é feita divisão, as pessoas físicas e jurídicas passaram a ter atributos próprios, porém continuam sendo uma pessoa e herdando os atributos daquela entidade “pai” Pessoa.

### Agregação

A Agregação é feita a partir de um relacionamento entre entidades, a partir desse relacionamento criamos uma espécie de “nova entidade”, que pode ser usada para outros relacionamentos, isso ocorre pois não podemos relacionar dois relacionamentos, então usamos a agregação para criar essa “exceção”.



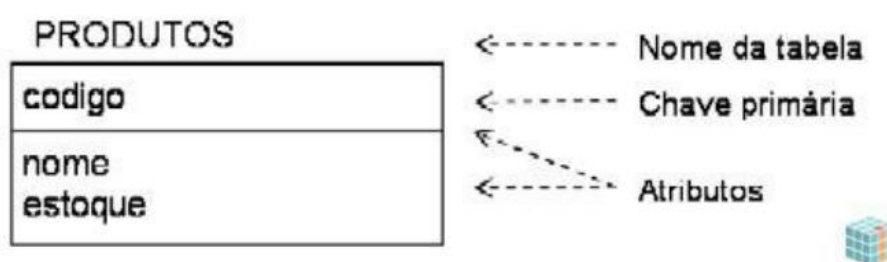
Fonte: <https://douglasgaspar.files.wordpress.com/2013/10/mer-8.jpg>

A Figura 58 nos mostra um exemplo prático, onde o relacionamento consulta entre Médico e Paciente, é utilizado para gerar uma receita, nesse caso a Consulta é uma agregação pois ela está utilizando uma relação para se relacionar com outra entidade.

### Modelo Lógico

A Modelagem lógica ajuda na compreensão visual das informações que serão armazenadas que serão usadas para cumprir as tarefas necessárias e suas funções. Para criarmos um modelo lógico é recomendável que tenhamos um modelo conceitual, pois a partir dele iremos introduzir os elementos associados, as definições e teremos um melhor contexto para nossa estrutura de dados, pois nessa fase é que iremos aplicar os conceitos das chaves primarias, estrangeiras, da cardinalidade e dos relacionamentos.

A principal função do modelo lógico é visualizar os elementos de dados, ver como eles se relacionam e detalhar e definir o tipo dos atributos. Ao fazermos o mapeamento para o modelo lógico já iremos utilizar “tabelas” para exibir os dados (Erwin Quest Software, 2022)



Fonte: [http://mz.pro.br/modelagem/conceitual\\_logico.pdf](http://mz.pro.br/modelagem/conceitual_logico.pdf)

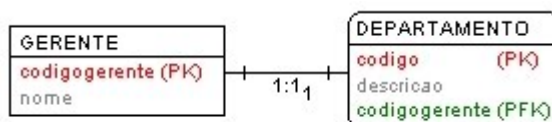
Na figura 59 podemos notar uma tabela, essa tabela nada mais é que uma entidade do Modelo Conceitual onde o nome da tabela, é o nome da entidade, a chave primaria é o atributo identificador, e os outros atributos são o restante dos atributos do modelo conceitual

Essa é uma conversão “simples” de apenas uma entidade, porém para fazermos o mapeamento completo devemos levar em conta a tudo que aquela entidade está relacionada e todo o “universo” do Modelo Conceitual.

Os principais relacionamentos que requerem “cuidados” na hora de serem mapeados são:

### **Relacionamento um para um (1:1)**

Onde a entidade dominada irá receber o atributo identificador ou chave primária da entidade dominante, como chave estrangeira que é responsável por interligar as tabelas

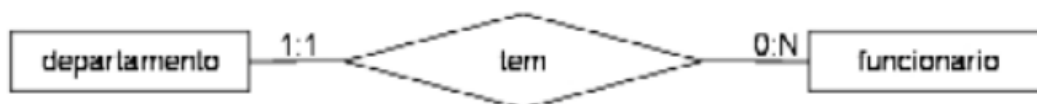


Fonte: [https://sites.google.com/site/uniplibancodedados1/\\_/rsrc/1348510710650/aulas/aula-7---tipos-de-relacionamento/aula\\_7\\_relacionamento3.bmp](https://sites.google.com/site/uniplibancodedados1/_/rsrc/1348510710650/aulas/aula-7---tipos-de-relacionamento/aula_7_relacionamento3.bmp)

Na figura 60 mostra o seguinte caso um gerente, gerencia um departamento, onde o gerente é a entidade dominante e departamento a entidade dominada, ao fazer o mapeamento pode se notar que o Departamento recebe o código do gerente.

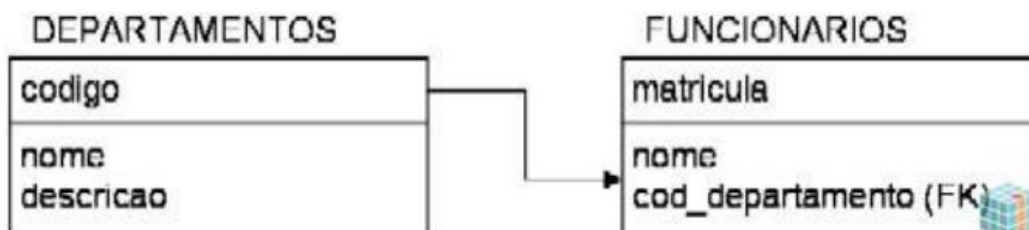
### Relacionamento um para muitos (1:N)

Para realizarmos o mapeamento desse caso primeiro precisamos identificar a entidade que recebe a cardinalidade de muitos, o N no caso e a Entidade com valor de 1



Fonte: [http://mz.pro.br/modelagem/conceitual\\_logico.pdf](http://mz.pro.br/modelagem/conceitual_logico.pdf)

Na figura 61 mostra um caso em que um departamento tem vários funcionários e um funcionário pertence a UM departamento, aqui ao realizarmos a conversão o funcionário irá receber a chave primária do departamento como chave estrangeira, igual mostra a figura abaixo



Fonte: [http://mz.pro.br/modelagem/conceitual\\_logico.pdf](http://mz.pro.br/modelagem/conceitual_logico.pdf)

Na figura 62 mostra as 2 tabelas já convertidas e mostra também que a tabela funcionário recebeu o código do departamento, que é a chave primária da entidade departamento, como sua chave estrangeira

### Relacionamento muitos para muitos (N:N)

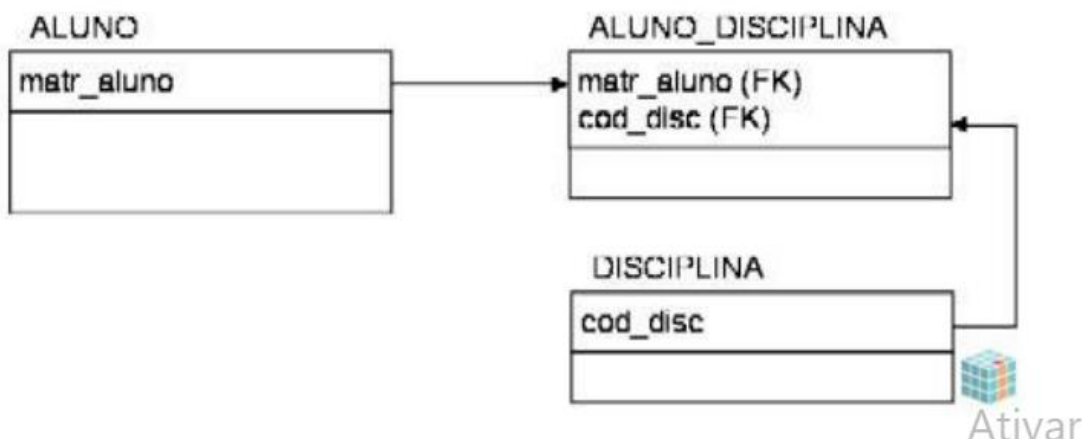
Nesse tipo de relacionamento iremos criar uma tabela para a relação entre as entidades, e pegar a chave primária de cada uma das entidades

Figura 1 - Exemplo Relacionamento N:N



[http://mz.pro.br/modelagem/conceitual\\_logico.pdf](http://mz.pro.br/modelagem/conceitual_logico.pdf)

Nesse caso o aluno pode cursar várias disciplinas e a disciplina “pertence” a vários alunos, logo a tabela aplicando o conceito de Relacionamento N para N, irá ficar da seguinte forma

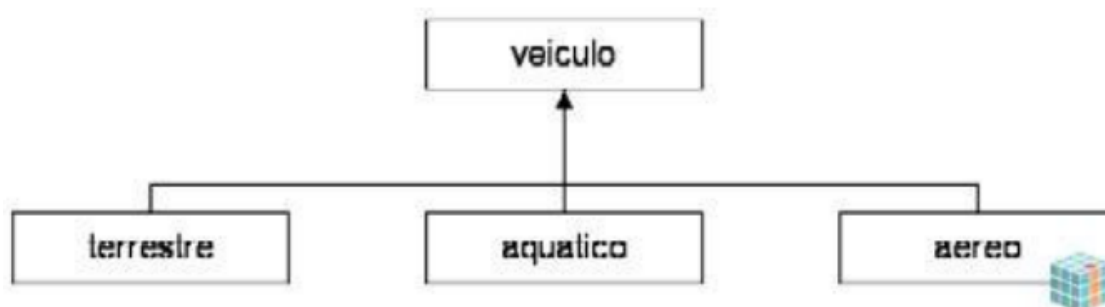


Fonte: [http://mz.pro.br/modelagem/conceitual\\_logico.pdf](http://mz.pro.br/modelagem/conceitual_logico.pdf)

Pode se notar que foi criada uma tabela recebendo as chaves primárias das tabelas aluno e disciplina como chave estrangeira.

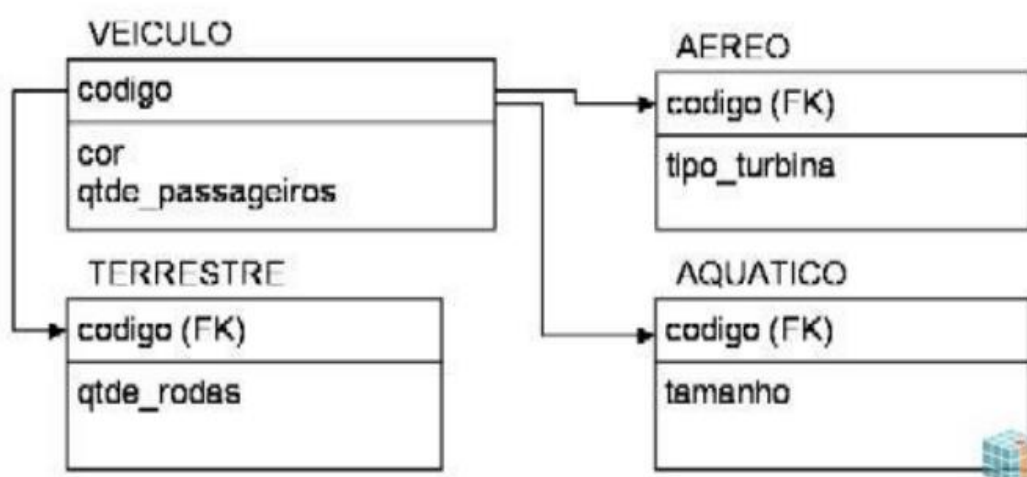
### Relacionamento Generalização/ Especialização

No caso de Generalização/ Especialização devemos criar uma tabela para cada entidade “filha”, e uma tabela para a entidade pai, e conseqüentemente as tabelas filhas irão receber a chave primária da tabela pai como chave primária e estrangeira.



Fonte: [http://mz.pro.br/modelagem/conceitual\\_logico.pdf](http://mz.pro.br/modelagem/conceitual_logico.pdf)

Nesse caso a entidade veículo é categorizada em 3 tipos: terrestre, aquático e aéreo. Para esse caso iremos mapear da seguinte maneira por se tratar de uma generalização e especialização



Fonte: [http://mz.pro.br/modelagem/conceitual\\_logico.pdf](http://mz.pro.br/modelagem/conceitual_logico.pdf)

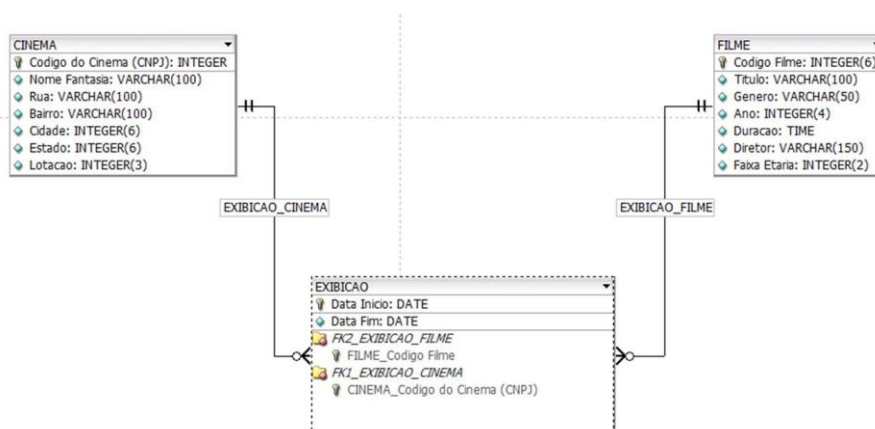
Conforme mostra a figura 66 todas as entidades filhas receberam uma tabela para cada, e a entidade pai também recebeu a tabela, todas as tabelas filhas receberam a chave primária da tabela veículo, como chave primária e estrangeira

## Modelo Físico

O Modelo físico é a última etapa da modelagem de dados, nesse modelo devemos ter uma linguagem de banco de dados definida e faremos a construção de tabelas, e definição dos tipos dos atributos no banco de dados a partir de comandos



SQL/ ou NOSQL dependendo do modelo escolhido ou até mesmo por interface gráfica caso estiver usando um SGBD (Sistema Gerenciador de Banco de Dados)



Fonte:

[https://livrodigital.uniasselvi.com.br/GTI11\\_principios\\_de\\_banco\\_de\\_dados/unidade2.html?topico=1](https://livrodigital.uniasselvi.com.br/GTI11_principios_de_banco_de_dados/unidade2.html?topico=1)

Como mostra a figura 67 nessa etapa da modelagem de dados que definimos o tipo dos atributos criados e seu limite de caracteres, são usados vários comandos para a criação dessas tabelas

Entre os principais comandos da Linguagem de Definição de tabelas e do CRUD (Linguagem de manipulação) estão:

Create: Usado para criação de tabelas e bancos de dados;

Primary Key: Utilizados para definição de chaves primárias;

Foreign Key: Utilizado para Definição de chave estrangeira;

Insert: Utilizado para Inserção de dados na tabela;

Update: Utilizado para atualizar dados de uma tabela;

Delete: Utilizado para exclusão de dados da tabela;

Select: Exibir Registros da tabela;

Alter: Alterar campos de uma tabela ou uma tabela;

```
CREATE TABLE turma (
    idturma INTEGER(4) NOT NULL AUTO_INCREMENT,
    capacidade INTEGER(2) NOT NULL,
    idProfessor INTEGER(4) NOT NULL,
    PRIMARY KEY (idturma),
    FOREIGN KEY (idProfessor) REFERENCES professor(idProfessor),
    UNIQUE KEY idturma (idturma)
)
```

Fonte: Próprio autor

A Figura x.x mostra o processo feito para a criação de uma tabela no modelo físico, onde como citado utilizamos o create table, para criamos uma tabela, e na hora de inserirmos os atributos da mesma, devemos definir seu nome, seu tipo, máximo de caracteres e Características existentes em alguns bancos de dados, nesse exemplo temos o NOT NULL que significa que o valor não pode ser nulo e o AUTO\_INCREMENT que adiciona um valor automaticamente caso ele não seja definido

### 2.4.2 SGBDS

Um SGBD ou Sistema Gerenciador de Banco de Dados, como o próprio nome diz é um software utilizado para o gerenciamento de Banco de Dados, nele é possível criar, modificar, eliminar, inserir dados e eliminar dados das bases de dados.

“O sistema de gerenciamento de bancos de dados (SGBD) é um conjunto de programas que gerenciam a estrutura do banco de dados e controlam o acesso aos dados armazenados. Até certo ponto, o banco de dados se assemelha a um arquivo eletrônico com conteúdo muito bem-organizado com a ajuda de um software poderoso, conhecido como sistema de gerenciamento de banco de dados. “

“O SGBD serve como intermediário entre o usuário e o banco de dados. Sua estrutura é armazenada como um conjunto de arquivos e o único modo de acessar os dados nesses arquivos é por meio do SGBD.” (Rob, et al., 2010 pp. 6, 7)

Os bancos de dados só começaram a ser operados por SGBDS (Sistemas Gerenciadores de Banco de Dados) a partir de 1970, quando o pesquisador Edgar

Frank “Ted” Codd da IBM(International Business Machines Corporation ) apresentou o Modelo Relacional no Artigo “A Relational Model of Data for Large Shared Data Banks”, nesse artigo explicava para técnicos sem experiência como armazenar e extrair grandes quantidades de dados de um banco de dados, a partir dessa ideia a própria IBM decidiu criar o primeiro projeto de Sistema Gerenciador de banco de dados chamado “System R”. Antes disso eram operados por sistemas de arquivos do sistema operacional utilizado na máquina.

Hoje em dia dentre os principais modelos de SGBDS estão os Modelos Relacionais (SQL – Standard Query Language) e Modelos Não Relacionais (NoSQL - Not Only Standard Query Language), abaixo seguem suas definições e exemplos:

### **Modelo Relacional (SQL)**

Como dito no anteriormente, o Modelo Relacional foi criado por Edgar Frank “Ted” Codd que trabalhava na IBM, por volta da década de 70 e consistia. Esse primeiro modelo relacional era operado pelo sistema chamado “System R” da IBM que utilizava a linguagem SQL (Standard Query Language) para a consulta do famoso System R, porém era utilizada apenas para o Sistema da IBM.

A Primeira versão padrão do SQL foi desenvolvida apenas em 1984 quase uma década após ter sido implementada no System R, pois era necessária uma padronização da linguagem, já que após ela ter sido utilizado no System R outros desenvolvedores estavam criando outras versões da linguagem.

Nos dias de hoje, a Linguagem SQL é uma das mais utilizadas no ramo de banco de dados, dentre os principais SGBDS que utilizam a linguagem estão: Oracle, SQL Server e MySQL

### **SGBD Oracle**

O Oracle SGBD foi criado no ano de 1977 por Lawrence Ellison, Bob Miner e Ed Oates, com o avanço das tecnologias de banco de dados nos anos 70 e 80, eles perceberam uma oportunidade de comercializar um Sistema que gerenciasse o banco de dados relacional na época, já que nenhuma empresa estava a comercializar um sistema desse tipo.

A Linguagem Utilizada pela Oracle é PL/SQL que é uma extensão da linguagem SQL padrão, além de ser utilizado para o SGBD ela é utilizada em muitos outros softwares da empresa.

Os principais fatores que levaram o SGBD da Oracle a ser tão popular são os fatos de que oferecem uma segurança avançada de várias camadas, implementação de criptografia, estabilidade, flexibilidade de gerenciamento, suporte entre outros (Andrade, 2022)

## MySQL

O MySQL é um SGBD Open-source que utiliza a linguagem SQL para criação e busca de banco de dados. O MySQL foi criado em 1994 na Suécia pelos finlandeses: David Axmark, Allan Larsson e Michael "Monty" Widenius, que futuramente vieram a criar a MySQL AB, desenvolvedora do SGBD, porém a empresa foi comprada pela Oracle no ano de 2010, passando a ser detentora do MySQL

De acordo com a **Fonte bibliográfica inválida especificada**, é um dos SGBDS mais utilizados no mercado de trabalho. Um dos motivos que levam o MySQL a ser um dos mais utilizados é pelo fato de ser Open-source (código aberto), por pode se conectar a outros bancos de dados como DB2 e Oracle, e por ter muitas funcionalidades, além de uma interface gráfica ótima e intuitiva. (Souza, 2020)

## SQL Server

O SQL Server é um SGBD que em 1988 foi criado pela Sybase, em parceria com a Microsoft, ele utiliza as Linguagens T-SQL e ANSI SQL que derivam da linguagem SQL padrão. Diferente do MySQL ele não é open-source, porém de acordo com **Fonte bibliográfica inválida especificada**, continua sendo um dos SGBDS mais utilizados do mercado.

Os principais motivos da popularidade do SQL Server se devem ao fato de possuírem facilidade na análise de dados, sua linguagem é flexível, possuem cloud storage

(armazenamento em nuvem), emitem relatórios com alta gestão de informação e têm ótima segurança e criptografia de dados. (Flexus, 2021)

## **Modelo Não Relacional (NoSQL)**

Os modelos de banco de dados NoSQL (Not Only Standard Query Language) de acordo com (Amazon, 2022) são criados para modelos de dados específicos e têm esquemas flexíveis para a criação de aplicativos modernos. Os bancos de dados NoSQL são amplamente reconhecidos por sua facilidade de desenvolvimento, funcionalidade e performance em escala.

O Modelo NoSQL possibilita uma variedade de modelos de dados, como chave-valor, documento e gráficos que são otimizados para performance. Nesse modelo o registro que seria salvo em uma tabela em um banco de dados relacional, vai ser salvo como um arquivo JSON (JavaScript Object Notation), esse arquivo irá conter todos seus atributos em um único documento permitindo desenvolvimento intuitivo e escalabilidade horizontal. (Amazon, 2022)

O principal motivo de utilizarem Banco de Dados NoSQL é o fato dele ter Flexibilidade, que é o fato de permitir um desenvolvimento rápido e iterativo, a escalabilidade que faz com sejam projetados para serem escalados horizontalmente usando clusters distribuídos de hardware, em vez de escalá-los verticalmente adicionando servidores caros e robustos e Alta Performance, que traz uma maior otimização dos dados utilizados, além de ser Altamente Funcional. (Amazon, 2022)

Entre os principais SGBDS NoSQL estão o: MongoDB, Redis e Cassandra

## **Mongo DB**

O MongoDB é um banco de dados orientado a documentos open-source lançado no ano de 2009 pela MongoDB Inc., seu código foi desenvolvido em C++, C, GO e JavaScript. O MongoDB é utilizado para grande quantidade de dados e que tenha bom funcionamento e performance com grandes volumes de dados. Ele oferece suporte para quase todas as linguagens de programação utilizadas atualmente.

Seu processo é mais fluido do que a de um banco de dados relacional, e seus elementos têm identificação única, os itens não são relacionados entre si e sua hierarquia é flexível e suas informações diferente de um banco de dados relacional que são armazenadas em tabelas, são armazenadas em documentos e coleções (que são subpartes do banco de dados, independentes).

As principais vantagens de usar o MongoDB são que não é necessário projetar um esquema do banco de dados para trabalhar com ele, fornece grande flexibilidade para os campos do documento, não requer injeções SQL e é facilmente integrável com Big Data Hadoop que é utilizado para armazenar dados e executar aplicações em clusters de hardwares comuns.

## **Redis**

Segundo a (Amazon, 2022) o Redis é um banco de dados de chave-valor mais conhecidos atualmente de código aberto e distribuído na memória, foi lançado em 2009 pela Redis Labs e é compatível com diversas linguagens de programação. Ele oferece um conjunto de estrutura versátil, permitindo a criação de várias aplicações personalizadas. Seu nome REDIS significa Remote Dictionary Server, ou traduzindo para português servidor de dicionário remoto.

“Por conta da sua velocidade e facilidade de uso, o Redis é uma escolha em alta demanda para aplicações web e móveis, como também de jogos, tecnologia de anúncios e IoT, que exigem o melhor desempenho do mercado. ” (Amazon, 2022)

Ainda de acordo com a (Amazon, 2022) as principais vantagens de usar o REDIS são desempenho extremamente rápido, versatilidade e fácil uso, replicação e persistência, além da compatibilidade com diversas linguagens de programação no mercado

## **Cassandra**

Cassandra é um banco de dados open-source não relacional e colunar, criado pelo Facebook em 2008, foi inspirado no DynamoDB da Amazon e no BigTable da Google. O banco de dados colunar utilizado na Cassandra é bem similar ao utilizado em um banco de dados relacional.

De acordo com o (DevMedia, 2022) Cassandra utiliza a Linguagem CQL (Cassandra Query Language) que é uma linguagem utilizada para facilitar a criação e manipulação de estruturas e realizar manipulações de dados, sua sintaxe é bem similar ao SQL.

Entre as principais vantagens do Cassandra é de que ele possui uma alta performance, é escalável, possui uma boa segurança e sua linguagem CQL é similar ao SQL que pode ser útil para usuários acostumados ao banco de dados relacional.

## **2.4 Tecnologias Diversas**

A tecnologia é um produto da ciência e da engenharia envolvendo uma série de instrumentos, métodos e técnicas projetados para resolver problemas. É a aplicação prática do conhecimento científico em vários campos de estudo.

### **2.5.1 Bootstrap**

Bootstrap é um framework para a linguagem CSS que foi desenvolvido no Twitter nos meados de 2010, com seu principal objetivo de padronizações de ferramentas da empresa. Desde então ele passou por diversas atualizações, e atualmente ela está na sua versão 4.4.

Este framework pode combinar com JavaScript e CSS com objetivo de estilizar elementos de uma página em HTML. Ele consegue expandir suas opções além de mudar cores e formatos de letras.

Consiste em ser uma ferramenta onde proporciona interagir na página, pois acaba oferecendo series de componentes que acabam facilitando a comunicação com o usuário, como menus de navegação, controle de páginas, barra de progresso e etc.

Além que a funcionalidade que o framework pode acabar oferecendo, onde seu principal foco é permitir construção de site responsivo para dispositivo móvel. Significa que páginas podem desenvolver para funcionar em desktops, tablets e smartphones, de formas mais simples e mais organizadas. (Souza, 2019)

#### **Como surgiu:**

Originalmente criado por designers e desenvolvedores no Twitter, o Bootstrap se tornou um dos frameworks de front-end e projetos de código aberto mais populares

do mundo. Bootstrap foi criado no Twitter em meados de 2010 por @mdo (**Mark Otto**) e @fat (**Jacó Thornton**). Antes de se tornar uma estrutura de código aberto, o Bootstrap era conhecido como Twitter Blueprint. Ele foi desenvolvido como um guia de estilo dentro da ferramenta interna da empresa por mais de um ano antes de seu lançamento público e continua até hoje.

Originalmente lançado na sexta-feira, 19 de agosto de 2011, houve mais de 20 lançamentos desde então, incluindo duas grandes reescritas para v2 e v3. No Bootstrap 2, eles adicionaram a funcionalidade responsiva como uma folha de estilo opcional a todo o framework. Com base no Bootstrap 3, eles reescreveram a biblioteca novamente para torná-la responsiva por padrão com uma abordagem mobile-first. E no Bootstrap 4, reescreveram o projeto novamente para abordar duas importantes mudanças na arquitetura: migrar para Sass e migrar de CSS para flexbox.

### **Como Bootstrap funciona:**

Bootstrap é formado por uma série de arquivos CSS e JS (JavaScript) que serão responsáveis em atribuir caracteres específicos a elementos da sua página. Tendo nele um arquivo principal em seu framework, designado de Bootstrap.css, onde contém definições para todos os estilos utilizáveis. Sendo mais direto, a estrutura framework acaba sendo formada por dois diretórios (uma estrutura para acabar organizando arquivos de computador ou que tenha referências de outros arquivos):

- CSS: que contenha arquivos necessários para a estilização dos elementos e uma alternativa a um tema único;
- JavaScript: contém as versões do registro bootstrap.js (original e minificado), que é causante por dar as aplicações de feitiço que necessitam de manuseamento interativa.

Para pôr uma característica a um constituinte, basta instruir a classe adequada na expressão “class” do constituinte a haver estilizado. Veja o exemplo:

```

```

Fonte: Próprio Autor

Neste exemplo, foi imputado o conteúdo “rounded-sm” para a caráter da imagem. Trata-se de um feitiço quão adiciona cantos arredondados no elemento. Portanto, ao tender a imagem, as características referentes a essa aula reunião aplicadas no elemento. (Souza, 2019)



### **Como baixar o Bootstrap:**

Existem diferentes formas de se baixar o framework. Uma delas é executar o download da versão compilada dos códigos CSS e JavaScript por meio da página do Bootstrap (<https://getbootstrap.com/docs/5.1/getting-started/download/>). Há também a opção de baixar o código-fonte do framework, no instante que ele é uma ferramenta de código aberto.

Quem não quiser baixar os arquivos para poder ter acesso ao framework sem a essencialidade de instalar ele no servidor. Na prática, os arquivos de instalação ficam em outro domínio, ou seja, em outro DNS. Para isso, é apenas usar os links para acessar o BootstrapCDN e, dessa forma, adicionar as referências para os arquivos necessários para a sua utilização.

Segundo o site (<https://rockcontent.com/br/blog/bootstrap/>) – “Outra forma de baixar o framework é por meio de gerenciadores de pacotes. É importante dizer que o Bootstrap pode ser utilizado com diferentes linguagens de programação”

Por este motivo, ele pode acabar sendo baixado com RubyGems, npm, do Node.js, Nuget ou Composer, e sendo utilizado para se criar um site em WordPress, em sites que são desenvolvidos em Ruby on Rails, em Asp.Net entre tais coisas. (Souza, 2019)

### **Como configurar e utilizar o Bootstrap:**

Existem diferentes formas de se configurar o framework para utilizá-lo em uma prática web. Entretanto, para que ele acabe funcionando de forma mais adequada, é necessário que adicione as bibliotecas JQuery e Popper.js, que serão necessárias para a prática de alguns componentes do Bootstrap.

Para se iniciar a empregar o Bootstrap em uma página, é necessário adicionar o registro dos principais arquivos do framework na página matriz da administração. Confira, a seguir, o código de uma página HTML com todas as referências necessárias para que o framework funcione.

```

<!doctype html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
  <!-- Bootstrap CSS -->
  <link rel="stylesheet"
    href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
    integrity="sha384-Vkoo8x4CGs03+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
    crossorigin="anonymous">
  <title>Hello, world!</title>
</head>
<body>
  <h1>Hello, world!</h1>
  <!-- Optional JavaScript -->
  <!-- jQuery first, then Popper.js, then Bootstrap JS -->
  <script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
    integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1yYfoRSJoZ+n"
    crossorigin="anonymous">
  </script>

  <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"
    integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEwdlLvI9I0Yy5n3zV9zzTtmI3UksdQRVvoxMfooAo"
    crossorigin="anonymous">
  </script>

  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js"
    integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7YwaYd1iqfktj0Uod8GCExl3Og8ifwB6"
    crossorigin="anonymous">
  </script>
</body>
</html>

```

Fonte: Próprio Autor

É vital seguir sua ordem de referência conforme o exemplo. Então, sua primeira referência deverá ser o arquivo Bootstrap.css localizado na tag **<head>**. Já suas referências a arquivos (Costa, 2023)Script deverão ser colocadas no final de sua página, antes mesmo de encerrar a tag **</body>**. A ordem de scripts deve ser: JQuery, Popper.js e Bootstrap.js.

É necessário comentar que a forma de configurar o Bootstrap vai variar de acordo com o tipo de ambiente que ele estará presente. Em WordPress, como exemplo, é possível se adicionar o framework por meio de alterações do arquivo **functions.php** do template. (Souza, 2019)

### Funcionalidades do Bootstrap:

O Bootstrap pode oferecer uma vasta série de funcionalidades que poderão implementar seu site. Observe agora o que se pode se fazer com a ferramenta.

### Layout Responsivo:

Uma das funcionalidades principais do Bootstrap é permitir a adaptação do layout da sua página sendo feita de acordo com o dispositivo utilizado. Para se garantir está responsivo, o framework trabalhará a estilização com o elemento **<div>** e com utilização da classe **container**.

Colocando em prática, o elemento **<div>** funcionará criando uma série de grades, se assemelhando com uma tabela, onde será capaz de estruturar uma página com mais aptidão. Tabelas eram utilizadas antigamente para se criar layouts responsivos, porém, possuía limitações onde a função da definição de largura das colunas, o que acabou invalidando sua utilização em dispositivos menores, como celulares.

Enquanto o elemento **<div>** é mais flexível, porque permitirá que a largura será definida e redimensionada com mais facilidade. O Bootstrap atribui o elemento **<div>** a característica da classe **container**, que funcionará para definir dimensões adequadas para elementos que serão inseridos no espaço. Em resumo, o framework trabalhará com três tipos de container:

- Container: em contíguo com a caráter max-width, que determina qual é o comprimento do quadro ideal para que haja a apropriação do layout da página;
- Container-Fluid: considera toda a vastidão do quadro do dispositivo para a firmiação do layout. Para isso, é considerada a caráter width:100% em todos os limites de comprimento de quadro;
- Container (Breakpoint): considera width:100% mesmo que um determinado comprimento seja atingido.

#### **Biblioteca de componentes:**

Outra praticidade curiosa do framework é a porção de componentes que podem ser utilizados para proporcionar maior interatividade e corrigir a ligação com o usuário. Confira, a seguir, alguns dos principais componentes do Bootstrap. (Souza, 2019)

#### **Alerts:**

Com o Bootstrap existe a possibilidade de configurar de forma mais simples e dinâmica tipos de alertas, com específicas cores, conforme a situação. Para demonstrar um alerta ao usuário indicar atenção, como exemplo, basta ser utilizado a

classe. alert-danger e será exposto uma caixa com texto com seu fundo em vermelho. Observe a seguir:

```
<div class="alert alert-danger">  
  <strong>Atenção!</strong> Cuidado com a mensagem de alerta!  
</div>
```

Fonte: Próprio Autor

### **Carousel:**

Um componente bastante empregado no Bootstrap é o Carousel. Trata-se de um slideshow, ou seja, um instrumento que permitirá a amostra de imagens de aparência responsiva. Ele também possibilitará a introdução de efeitos especiais para a trajetória da registro e controles de amostra, como os indicadores de rente e anterior.

### **NavBar:**

Outro forte componente do framework é o NavBar, que permite a edificação de um sistema de navegação responsivo. É provável caracterizar diferentes formas de apresentação do menu, como tomar entre o posicionamento lateral ou superior, e limitar a forma de apresentação, que pode existir estendida ou contraída.

Também é provável estipular a forma de mostra dos links do menu, que pode existir em forma de botão, link, menu suspenso, entre outras configurações, para conceder a implementação da navegação do site. (Souza, 2019)

### **3 Desenvolvimento**

No terceiro capítulo iremos abordar a fase de desenvolvimento do TCC, onde apresentaremos, de forma geral, a explicação e discussão do tema e os seus resultados finais.

#### **3.1 Levantamento de Requisitos**

O levantamento de requisitos será a parte mais importante do processo, onde resultará no desenvolvimento final do TCC. Entender o que o cliente desejará, ou aquilo que o cliente se acredita que necessite e as regras do negócio ou os processos dele.

##### **3.1.1 Entrevista/coleta de dados**

Na nossa coleta de dados preferimos buscar através de pesquisa de campo através de formulários, para facilitar tal pesquisa, procuramos enviar para um determinado grupo de pessoas, onde procurávamos saber se nosso software seria útil, mesmo não procurando fins lucrativos com ele. Tal pesquisa foi feita através do seguinte link:

[https://docs.google.com/forms/d/e/1FAIpQLSczQf8e9\\_z3Clku29bO8WlgCIHEd6bDI9HtCPDIOWG6iiUiA/viewform?embedded=true](https://docs.google.com/forms/d/e/1FAIpQLSczQf8e9_z3Clku29bO8WlgCIHEd6bDI9HtCPDIOWG6iiUiA/viewform?embedded=true)

##### **3.1.2 Protótipos**

Para a criação do protótipo do nosso site e aplicativo, buscamos encontrar sites que pudessem ajudar na parte do funcionamento e na parte visual, mas o mais importante também, que ajudasse a termos uma concepção melhor dos nossos pensamentos e ideias.

Então buscamos utilizar um site chamado (Figma) ele é um editor gráfico de vetores e prototipagens de projetos de design, sendo principalmente baseado em navegadores da web e tudo nele pode ser feito de forma online no navegador.

Com esta ferramenta em mãos conseguimos criar protótipos do nosso aplicativo e site de formas mais produtiva e ágil, pois com ele também conseguimos fazer nossa correção de erros durante o processo de testes do usuário.



Aqui, os visitantes encontrarão informações essenciais sobre os serviços oferecidos pela empresa, incluindo uma barra de navegação destacando os principais serviços e uma breve introdução que descreve o conteúdo da página e o slogan.

Logo abaixo, os visitantes terão acesso a um painel explicativo detalhado sobre o software fornecido pela empresa. Esse painel oferece uma visão completa das funcionalidades e recursos do software, permitindo que os usuários compreendam seu potencial e benefícios. Para facilitar o acesso a mais informações sobre o serviço oferecido, um botão estrategicamente colocado redirecionará os visitantes para uma página dedicada com informações detalhadas.

Nessa página adicional, os visitantes terão a oportunidade de explorar em profundidade as capacidades do software, suas aplicações específicas em diferentes setores e as vantagens competitivas que ele oferece. Além disso, eles poderão solicitar uma demonstração personalizada do software ou até mesmo iniciar um período de teste gratuito, permitindo que experimentem os recursos de primeira mão.

Reconhecemos a importância de fornecer suporte e garantias aos nossos clientes em sua jornada com o software. Portanto, em todas as etapas, nossa equipe estará pronta para oferecer assistência, responder a perguntas e fornecer orientações necessárias. Além disso, nosso compromisso em manter o software atualizado e oferecer melhorias contínuas garantirá que nossos clientes tenham acesso às mais recentes inovações e aprimoramentos.

Agradecemos a visita à nossa landing page e esperamos que ela desperte seu interesse em nosso software feito em Python. Convidamos você a explorar todos os recursos e benefícios que nossa solução pode trazer para o seu negócio. Não hesite em clicar no botão abaixo para obter mais informações e dar o próximo passo em direção ao sucesso. Estamos ansiosos para colaborar com você e ajudá-lo a alcançar seus objetivos.

O Teachable Machine é uma plataforma online desenvolvida pela Google que permite criar modelos de aprendizado de máquina para reconhecimento de objetos, gestos ou sons, tudo isso sem exigir nenhum conhecimento prévio em programação ou aprendizado de máquina. Neste tutorial, vou explicar como usar o Teachable Machine para criar um modelo de reconhecimento de imagens e exportá-lo para um modelo Keras.

### **Passo 1: Acessar o Teachable Machine**

Abra o seu navegador da web e acesse o [site do Teachable Machine] (<https://teachablemachine.withgoogle.com/>).

### **Passo 2: Escolher um Projeto**

Clique no botão "Começar" para iniciar um novo projeto.

### **Passo 3: Capturar ou Carregar Imagens**

Você tem a opção de capturar imagens usando sua câmera ou carregar imagens de seu computador. Escolha a opção mais adequada para o seu projeto.

- **Capturar Imagens:** Selecione "Usar a câmera" para tirar fotos de objetos, gestos ou expressões em tempo real usando sua webcam. Siga as instruções na tela para capturar várias imagens para cada classe (categoria) que você deseja treinar.

- **Carregar Imagens:** Selecione "Fazer upload de imagens" para carregar imagens previamente coletadas em seu computador. Você precisará organizar essas imagens em pastas separadas, cada uma representando uma classe.



#### **Passo 4: Rotular Classes**

Após capturar ou carregar imagens, você precisa rotular cada classe. Clique nas classes padrão (A, B, C...) para nomeá-las de acordo com as categorias do seu projeto.

#### **Passo 5: Treinar o Modelo**

Clique no botão "Treinar modelo" para iniciar o treinamento. O Teachable Machine usará o aprendizado de máquina transferido para criar um modelo de classificação.

#### **Passo 6: Testar o Modelo**

Após o treinamento, você pode testar seu modelo com novas imagens ou gestos para verificar como ele está se saindo. Use a webcam ou faça o upload de imagens para testar.

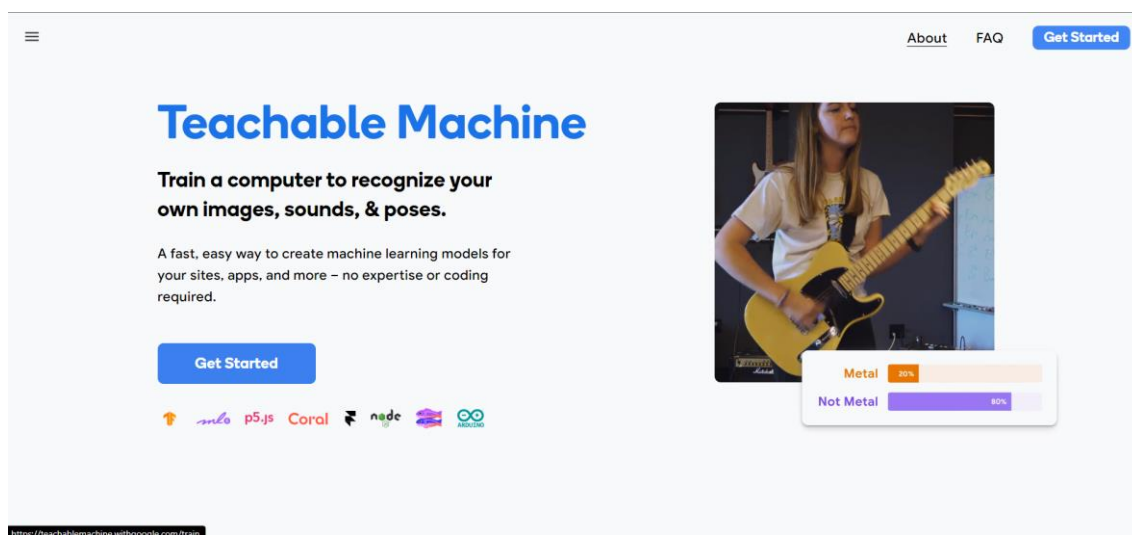
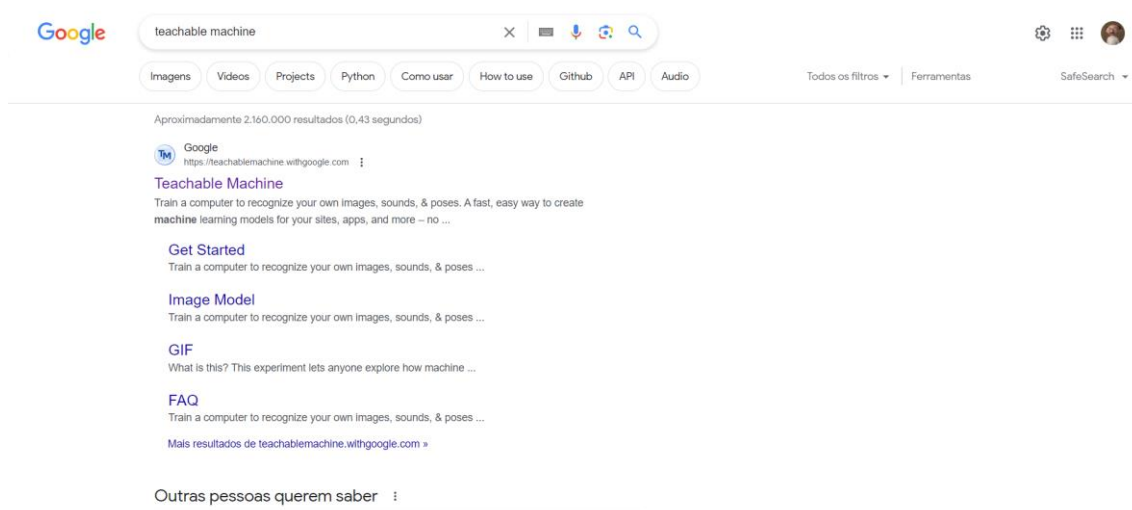
#### **Passo 7: Exportar o Modelo Keras**

Quando estiver satisfeito com o desempenho do seu modelo, você pode exportá-lo para uso em aplicativos ou projetos de aprendizado de máquina.

- Clique no botão "Exportar modelo" no canto inferior direito.
- Escolha a opção "Keras" como formato de exportação.
- O modelo Keras será gerado e você poderá fazer o download do arquivo `.zip` que contém o modelo.

## Passo 8: Usar o Modelo Keras

Agora você pode usar o modelo Keras exportado em seus próprios projetos de aprendizado de máquina ou aplicativos. Basta carregar o modelo em seu código e usá-lo para fazer previsões com base em novas imagens.



**Teachable Machine**

**A** 677 Image Samples

Webcam Upload

**B** 66 Image Samples

**1. Gather Samples**  
Make or add some samples here, or watch this how-to video.

**C** 309 Image Samples

Webcam Upload

**Training**

Train Model

Advanced

**Preview** Export Model

You must train a model on the left before you can preview it here.

**2. Train your Model**  
Now that you have two classes, you can train your model here (or add more classes).

English release-2-4-7 - 2.4.7#5b5b73

**Teachable Machine**

Webcam Upload

**J** 198 Image Samples

Webcam Upload

**K** 150 Image Samples

Webcam Upload

**L** 226 Image Samples

**Training**

Train Model

Advanced

**Preview** Export Model


You must train a model on the left before you can preview it here.

English release-2-4-7 - 2.4.7#5b5b73


## Training

Training...


Preparing training data...

Advanced 

## Preview


Export Model


You must train a model on the left before you can preview it here.

Export your model to use it in projects.


Tensorflow.js ⓘ
Tensorflow ⓘ
Tensorflow Lite ⓘ


Model conversion type:

☒ Keras
☐ Savedmodel

 Download my model

Converts your model to a keras .h5 model. Note the conversion happens in the cloud, but your training data is not being uploaded, only your trained model.

Code snippets to use your model:

Keras
OpenCV Keras
Contribute on Github 

```

from keras.models import load_model # TensorFlow is required for Keras to work
from PIL import Image, ImageOps # Install pillow instead of PIL
import numpy as np


# Disable scientific notation for clarity
np.set_printoptions(suppress=True)

# Load the model
model = load_model("keras_Model.h5", compile=False)

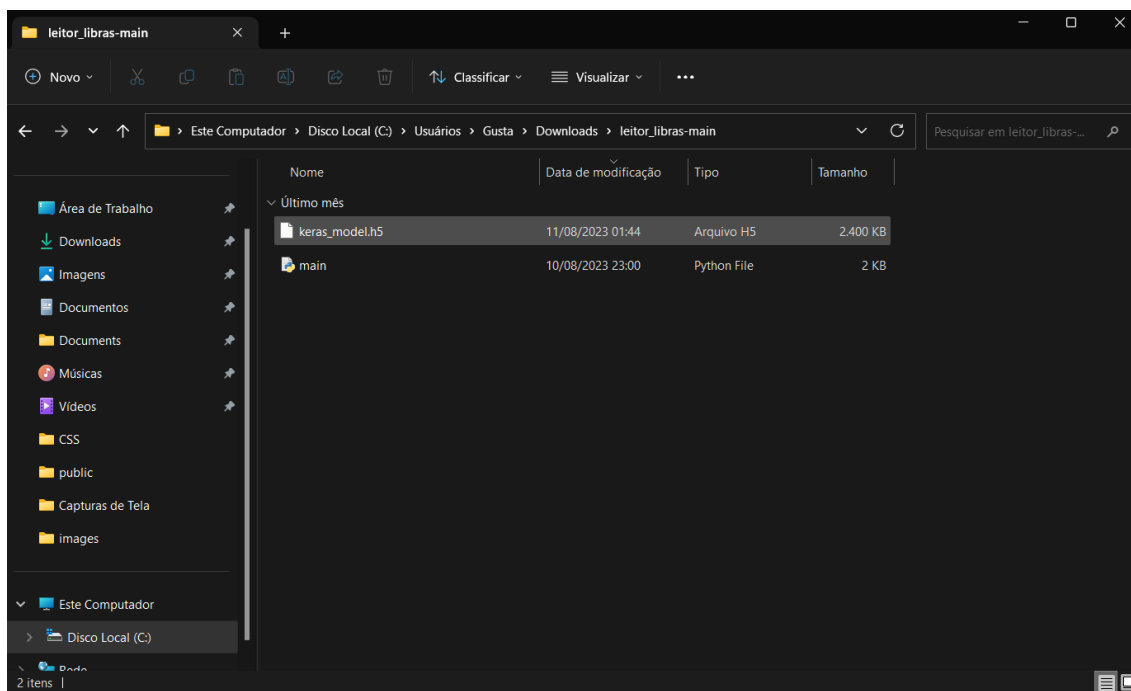
# Load the labels
class_names = open("labels.txt", "r").readlines()

# Create the array of the right shape to feed into the keras model
# The 'length' or number of images you can put into the array is
# determined by the first position in the shape tuple, in this case 1
data = np.ndarray(shape=(1, 224, 224, 3), dtype=np.float32)

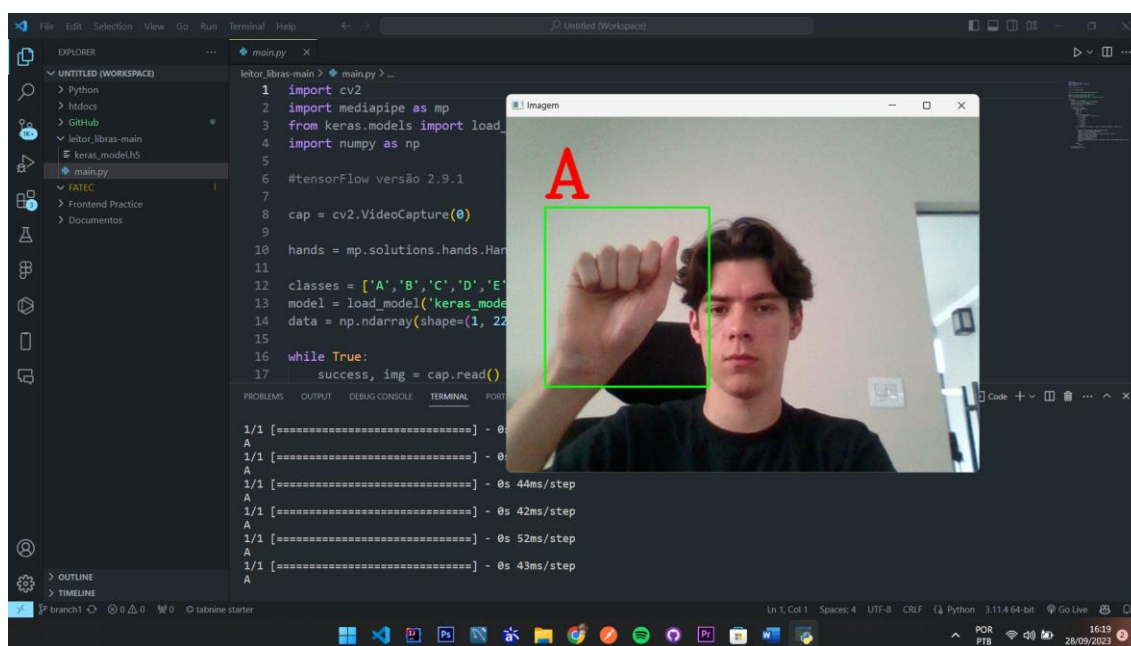
```

Copy 

Coloque o modelo exportado no mesmo diretório que o código



Depois disso é só começar a usar! Lembre-se de que treinar modelos de aprendizado de máquina requer um conjunto de dados representativo e, às vezes, ajustes adicionais para obter o melhor desempenho. Experimente diferentes configurações e colete dados de treinamento de alta qualidade para melhorar seu modelo.



## 4 Referências

**Amazon. 2022.** O que é NoSQL? *Amazon Web Services*. [Online] 05 de 2022. <https://aws.amazon.com/pt/nosql/>.

—, **2022.** O que é o Redis? *Amazon Web Services*. [Online] 05 de 2022. <https://aws.amazon.com/pt/elasticache/what-is-redis/>.

**Andrade, Ana Paula de. 2022.** Os principais SGBDs relacionais. *Treina Web*. [Online] Abril de 2022. <https://www.treinaweb.com.br/blog/os-principais-sgbds-relacionais>.

**Costa, Welligthon. 2023.** História do Python. [Online] 2023. <https://www.dio.me/articles/a-historia-do-python>.

**DevMedia. 2022.** Introdução ao banco de dados NoSQL Cassandra. *DevMedia*. [Online] Maio de 2022. <https://www.devmedia.com.br/introducao-ao-banco-de-dados-nosql-cassandra/30533>.

**Erwin Quest Software. 2022.** Modelagem lógica de dados. *Erwin*. [Online] 04 de 2022. <https://www.erwin.com/br-pt/solutions/data-modeling/logical.aspx>.

**Flexus. 2021.** Microsoft SQL Server é uma boa opção para a minha empresa? *Flexus*. [Online] Maio de 2021. <https://www.flexussolucoes.com.br/microsoft-sql-server/>.

**Horn, Michelle. 2022.** Python: o que é, como usar, guia pra aprender a linguagem. [Online] 2022. <https://blog.betrybe.com/python/>.

**Instituto Federal Santa Catarina. 2015.** PI S8 DSW I DouglasARS. *Enciclopédia Instituto Federal Santa Catarina*. [Online] Novembro de 2015. [https://wiki.ifsc.edu.br/mediawiki/index.php/PI\\_S8\\_DSW\\_I\\_DouglasARS](https://wiki.ifsc.edu.br/mediawiki/index.php/PI_S8_DSW_I_DouglasARS).

**Laundon, Kenneth e Laundon, Jane. 2014.** *Sistemas de Informações Gerenciais*. s.l. : Pearson Universidades, 2014.

**Lopes, Bergson. 2022.** Modelo Conceitual de Dados - Aprenda a utilizar os principais mecanismos de abstração. *BLRDATA*. [Online] 04 de 2022. <https://www.blrdata.com.br/single-post/2016/03/19/modelo-conceitual-de-dados-aprenda-a-utilizar-os-principais-mecanismos-de-abstra%C3%A7%C3%A3o>.

**Lopes, Gabrielle. 2021.** *A importância do mock-up nos projetos de TI.* [Online] 2021. <https://blog.workana.com/pt/sin-categorizar/a-importancia-do-mock-up-nos-projetos-de-ti/>.

**Rob, Peter e Coronel, Carlos. 2010.** *Sistemas de banco de dados: Projeto, implementação e administração.* s.l. : Cengage Learning, 2010.

**Souza, Ivan de. 2019.** *Bootstrap: saiba neste guia para iniciantes o que é, por que e como usá-lo.* [Online] 2019. <https://rockcontent.com/br/blog/bootstrap/>.

**Souza, Ivan. 2020.** *MariaDB ou MySQL: saiba qual tecnologia de banco de dados escolher.* Rockcontent. [Online] Abril de 2020. <https://rockcontent.com/br/blog/mariadb/>.

**X-apps. 2019.** <https://www.x-apps.com.br/node-js>. [Online] 2019.