

Bem-Vindos à Programação Server-Side

Católica de Santa Catarina – Centro Universitário
Jaraguá do Sul - SC, Brasil

Professor: MSc. Andrei Carniel (Prof. Andrei)

Contato: andrei.carniel@gmail.com

Agenda

1. About me
2. Ementa da disciplina
3. Server-side
4. Aplicações cliente servidor
5. Atividade

1. About me

About



- Graduação:

- Tecnologia em Análise e Desenvolvimento de Sistemas (2010)
- Universidade Tecnológica Federal do Paraná - UTFPR, campus Pato Branco



- Especialização:

- Programação Java (2012)
- Universidade Tecnológica Federal do Paraná - UTFPR, campus Pato Branco



- Mestrado:

- Computação Aplicada, área de Engenharia de Software (2017)
- Universidade do Estado de Santa Catarina - UDESC, campus Joinville



- Doutorado:

- Engenharia Eletrônica e Computação, área de Informática, subarea de Engenharia de Sistemas Críticos (defesa no dia 16/08/23)
- Instituto Tecnológico de Aeronáutica – ITA, São José dos Campos - SP

2. Ementa

Ementa da Disciplina – Programação Server-Side

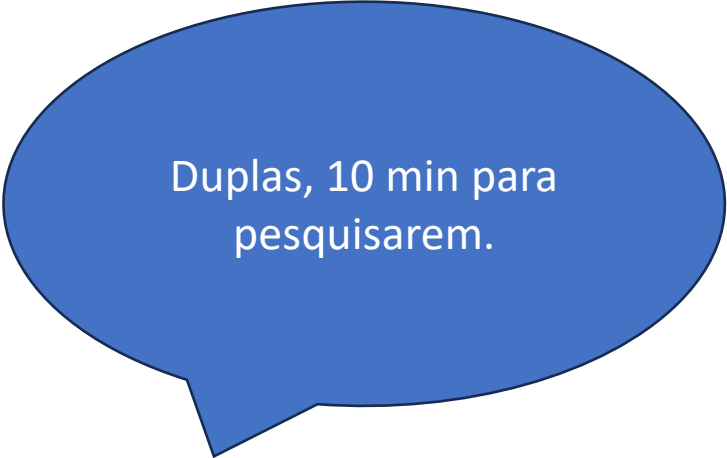
- Conceitos de aplicação Client/Server.
- Mapeamento O/R.
- Aplicações Distribuídas.
- Contextos e containers.
- Unidade de Persistência.
- Segurança em Transações.

- Bibliografia básica
 - VOHRA, D. **Java EE Development with Eclipse: develop**
 - ALUR, Deepak; CRUPI, John; MALKS, Dan. **Core J2EE patterns**
 - BHARGAV, Abhay; KUMAR, B. V. **Secure Java: for web application development**

3. Server-side

Programação Server-Side

- O que é programação Server-Side?
- Quais as vantagens?
- Quais as desvantagens?
- Características da programação Server-Side?
- Em um app de banco, onde está a lógica das transações?
 - Qual o risco de não estar em um servidor?
- Por que precisamos centralizar?



Duplas, 10 min para pesquisarem.

Programação Server-Side

- A programação server-side é a programação do lado do servidor. Refere-se à prática de escrever código e executar a lógica para um aplicativo ou cliente no servidor. Nesse método, existem pelo menos 2 papéis, o de cliente e o servidor.
- Uma alternativa para devolver uma página estática ou uma página específica para um determinado cliente.
- Esse modelo de programação é fundamental em muitos aplicativos web e sistemas online. Onde o cliente solicita um serviço ao servidor e o servidor devolve a resposta para o cliente.

Programação Server-Side

- **Vantagens:**

- É possível ter uma camadas de segurança e privacidade mais robustas.
- Velocidade e escalabilidade.
- Omissão da lógica de funcionamento do sistema.
- Interface do cliente mais leves.
- Interoperabilidade.
- Processamento pesado, a manipulação de dados e a interação com bancos de dados normalmente ocorrem no servidor.

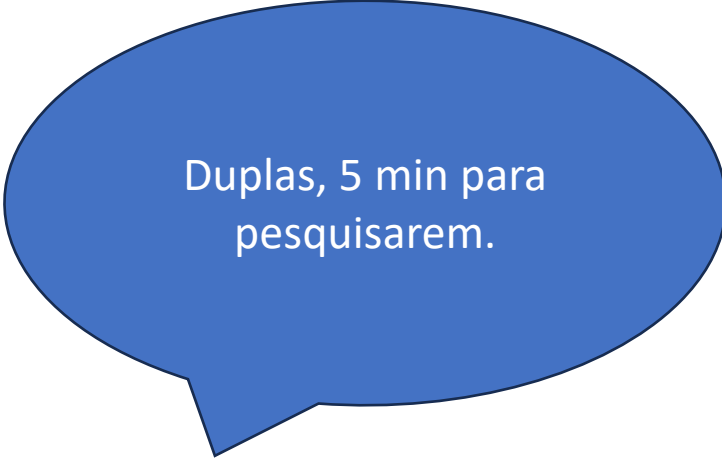
- **Desvantagens:**

- Necessidade de um servidor.
- Necessidade de uma conexão de rede.

- A lógica de negócio está no servidor

Programação Server-Side

Mas qual a diferença entre cliente e servidor? Exemplos...



Duplas, 5 min para
pesquisarem.

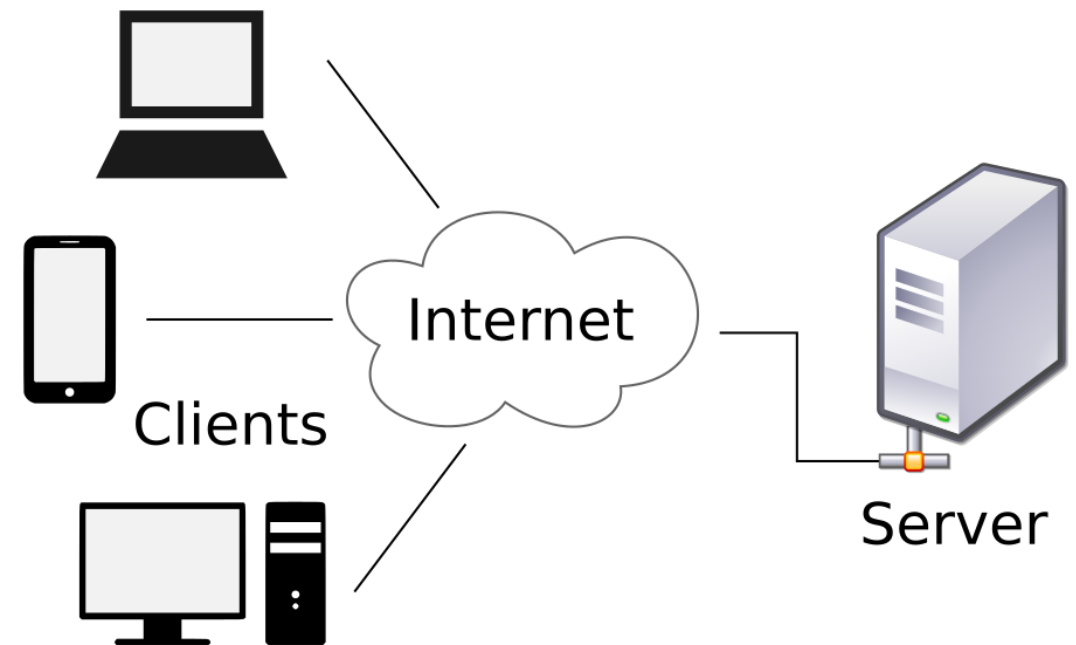
Programação Server-Side

- O que é um Cliente?
 - É um dispositivo de hardware ou software de computador que consome um serviço disponibilizado por um servidor.
- O que é um Servidor?
 - Um servidor é um computador físico ou um software na nuvem dedicado a executar serviços para atender às necessidades de outros computadores. O servidor geralmente (mas nem sempre) está localizado em um computador físico ou uma nuvem separada.
 - Existem diversos serviços, por exemplo: um servidor de arquivos, servidor de banco de dados, servidor de mídia doméstica, servidor de impressão, servidor da web, entre outros.

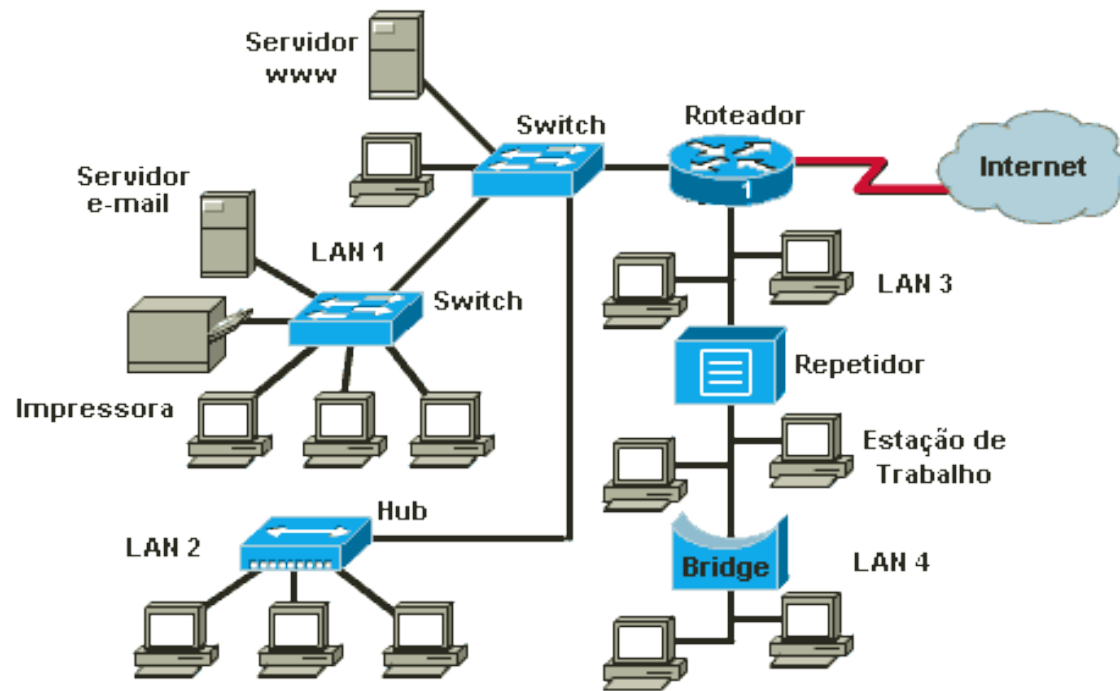
4. Aplicações cliente servidor

Programação Server-Side

- O cliente interage com o aplicativo por meio de um navegador ou outro dispositivo para interagir com o sistema. O servidor processa a solicitação, o servidor computa o resultado de acordo com a lógica empregada, e retorna os resultados ao cliente.
- Aqui entramos no conceito de aplicação distribuída.
- Particionamento de carga de tarefas ou cargas de trabalhos entre os provedores de serviço. Exemplo: *World Wide Web*.



Ponto de partida...



Dúvida... Será que os atuais dispositivos que temos no mercado são suficientes para processar a demanda de informação?



Aplicação Cliente-Servidor

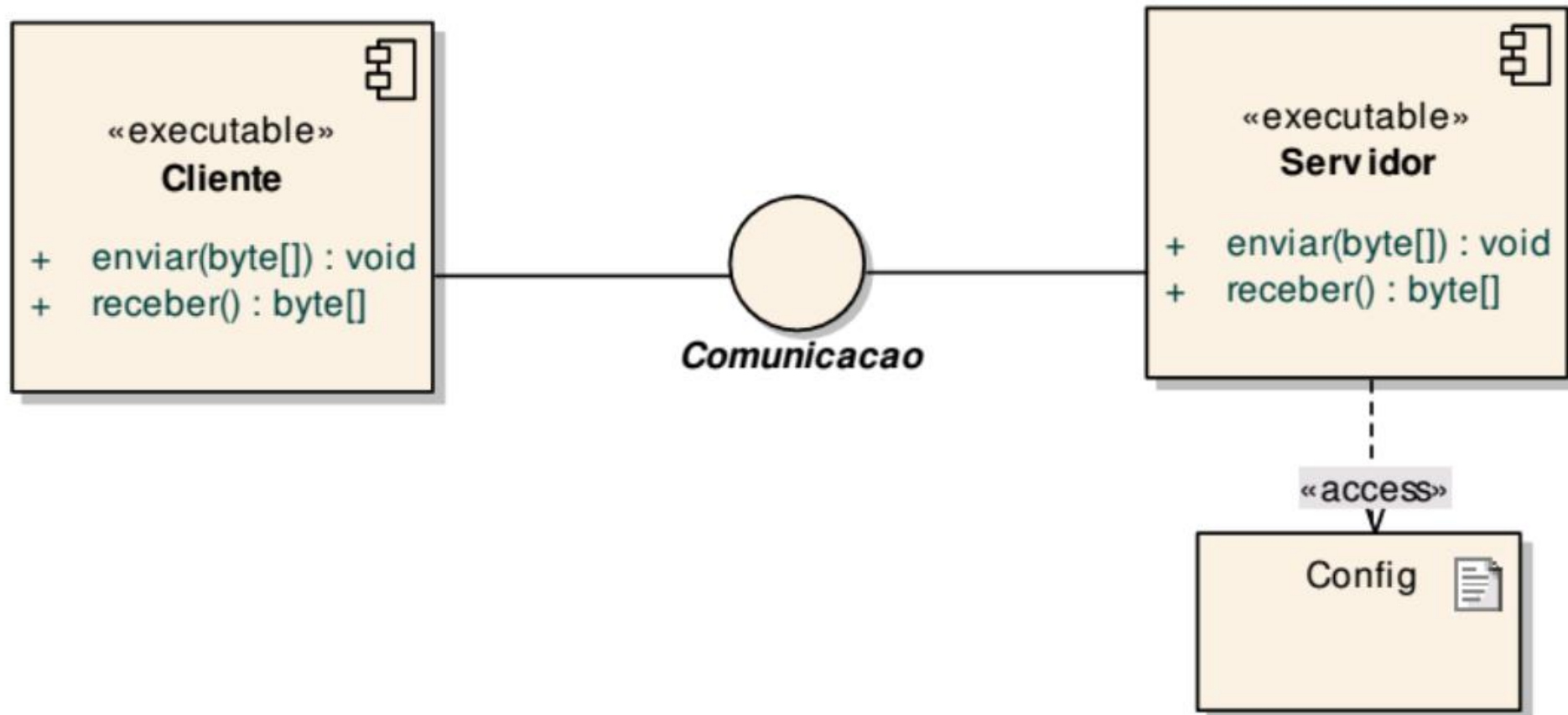
Antes de iniciarmos com a programação...

- Cliente-Servidor

- É um tipo de **sistema distribuído**, em que uma aplicação é modelada como um **conjunto de serviços** fornecidos por um ou mais servidores. Os clientes podem acessar esses serviços e **apresentar os resultados** para os **usuários finais**.
- Os clientes precisam estar cientes dos **servidores** que estão **disponíveis**, mas não devem saber da existência de **outros clientes**.

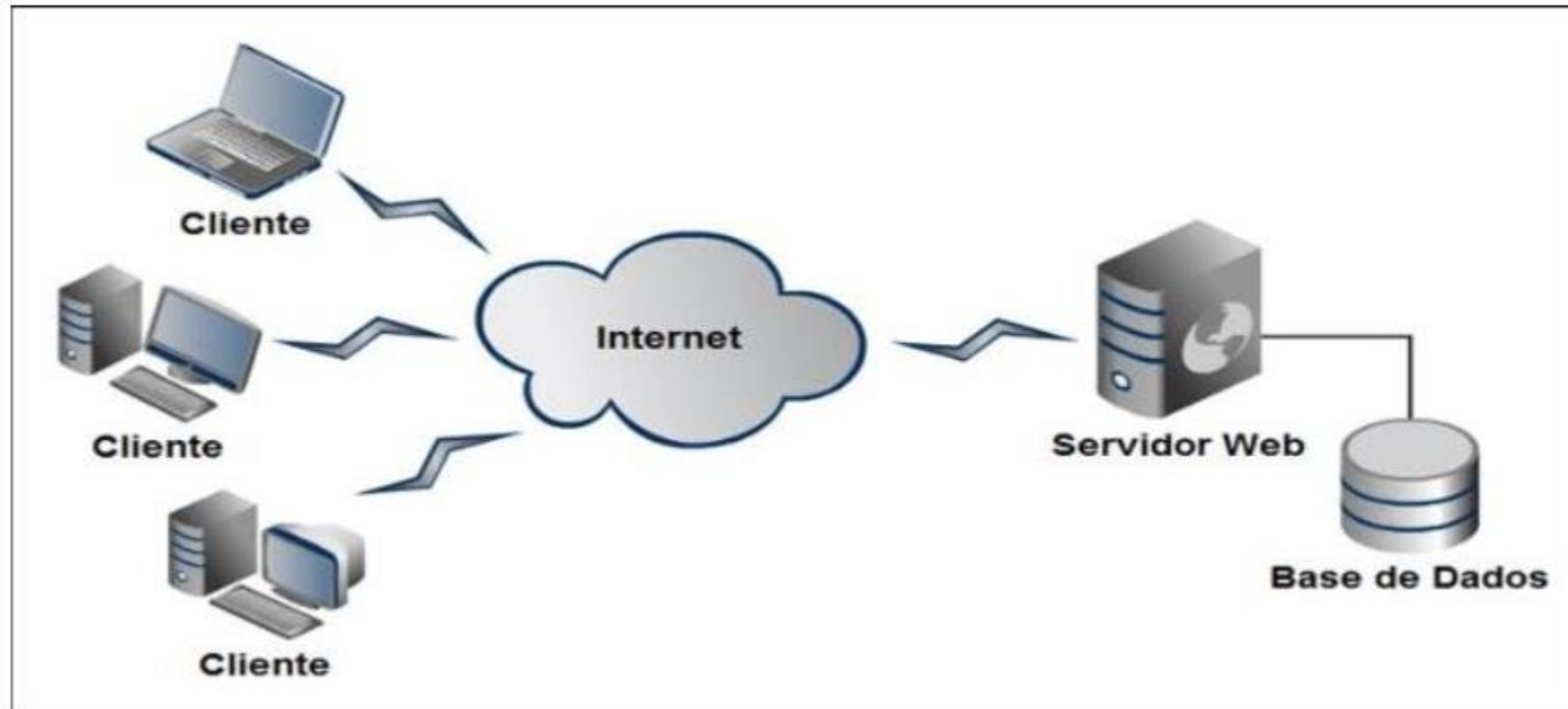
Aplicações Cliente-Servidor

- Modelo de Cliente-Servidor

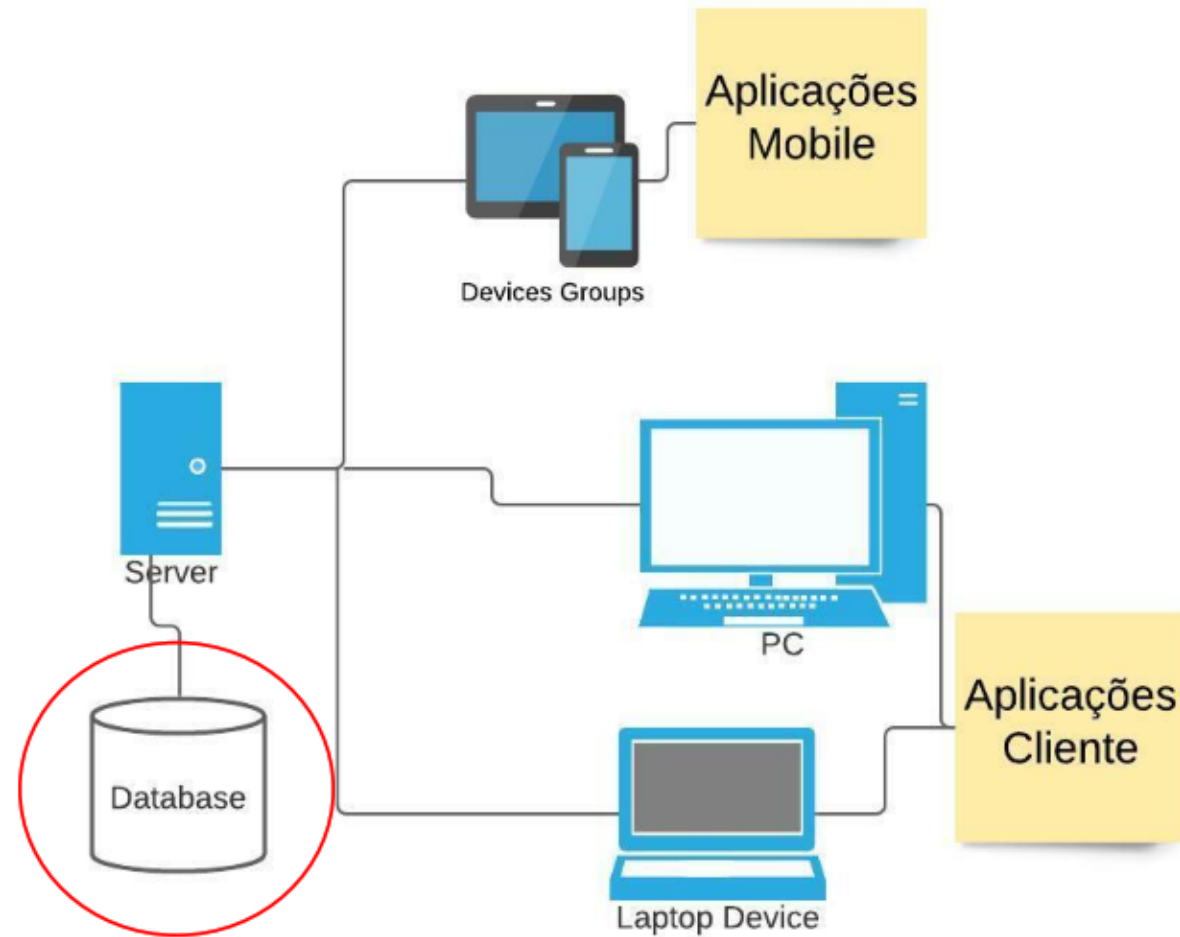


Aplicações Cliente-Servidor

- Arquitetura Cliente-Servidor



Aplicações Cliente-Servidor



Exemplos – Server-Side

- Exemplos de tecnologias de server-side:



Flask

- Framework de aplicativo para Web escrita em Python. Depende do Werkzeug WSGI toolkit and Jinja.
 - WSGI - *Web Server Gateway Interface*: especificação para interface universal entre servidor Web e aplicativo Web.
 - Werkzeug Kit de ferramentas
 - Experiência em HTML e Python.
- Web Application Framework ou simplesmente Web Framework.
 - Coleção de bibliotecas e módulos que permite um desenvolvedor de aplicativos Web crie aplicações sem se preocupar com detalhes de baixo nível.

5. Atividade

Atividade em Laboratório (brainstorming)

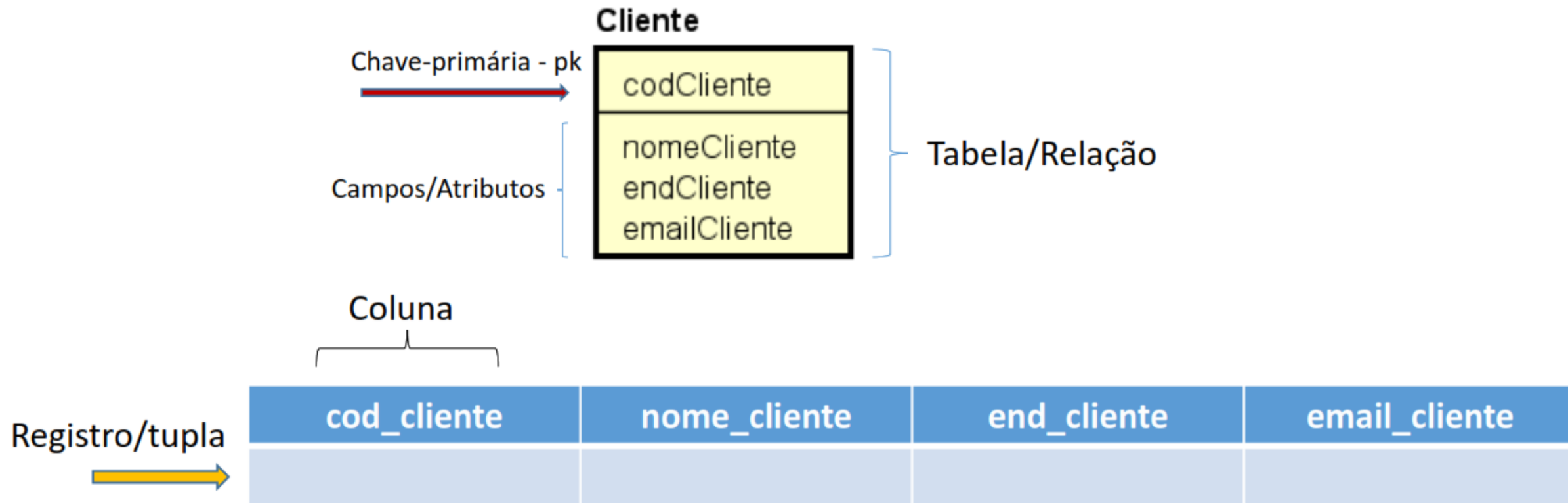
1. O que é uma aplicação Cliente/Servidor?
2. Cite características.
3. O que significa Server-Side Rendering (SSR)?
4. Atualmente, quais as tecnologias (frameworks e linguagens) são comumente utilizadas no desenvolvimento de aplicações SS? Qual você utilizaria e por quê?
 - Liste: 5 pontos positivos, 5 pontos negativos, e pelo menos 5 contextos de usos.

Busque as respostas para essas questões, considere a interação com o(s) colega(s) durante a elaboração das respostas. Para organizar as ideias a respeito das resposta utilize qualquer editor de texto, se preferir combine texto com elementos gráficos.

As dúvidas e respostas serão discutidas após a conclusão desta atividade.

Vamos partir para um exemplo prático

- Exemplo de database



Instância do Banco de Dados – Tabela Cliente

CodCliente	NomeCliente	EndCliente	emailCliente
00123	Asdrúbal	Rua sem nome	asd@server.com
00124	Jeca	Estrada da vida	
00122	Margarida	Rua das flores	mar@server.com