Universidade de Taubaté
Departamento de Informática

Sistemas Distribuídos

Web service

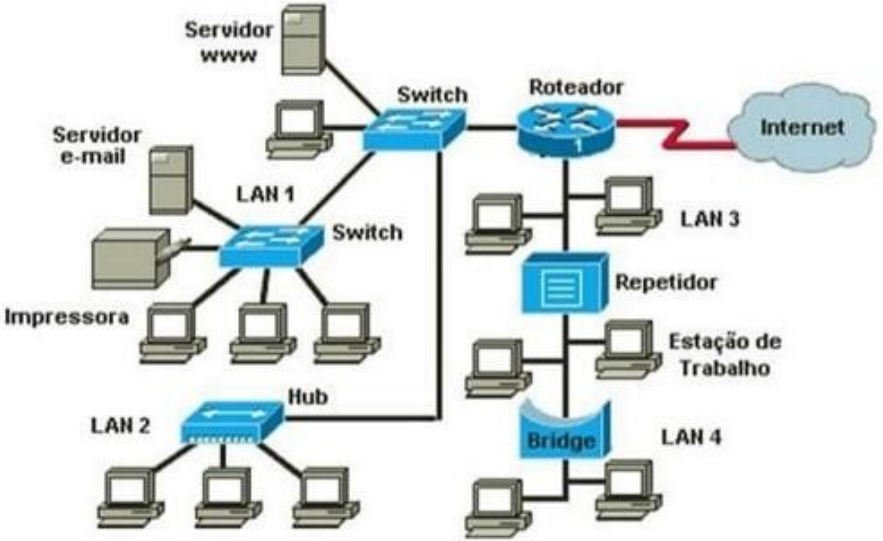
Sistemas de Informação

Engenharia de Computação

Sistemas Distribuídos

REDE

- Uma rede de computadores é formada por um conjunto de máquinas eletrônicas com processadores capaz de trocar informações e compartilhar recursos, interligadas por um sistema de comunicação.



Fonte: https://www.tiflux.com.br/blog/redes-ethernet-um-briefing/

Sistemas Distribuídos

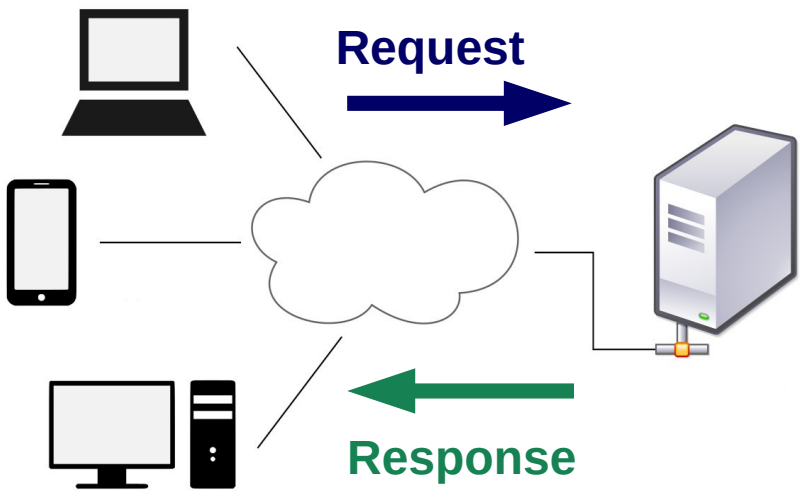
MODELO CLIENTE-SERVIDOR

- É uma estrutura de aplicação que distribui as tarefas e cargas de trabalho entre os fornecedores de um recurso ou de um serviço, designados como **servidores**, e os requerentes dos serviços, designados como **clientes**.
  - Servidor é um host que está executando um ou mais serviços ou programas que compartilham recursos com os clientes.
  - Cliente é quem solicita um conteúdo ou uma função do servidor.
- Geralmente os clientes e os servidores comunicam por meio de uma rede em computadores distintos.

Sistemas Distribuídos

MODELO CLIENTE-SERVIDOR

- Os clientes iniciam sessões de comunicação com os servidores que aguardam requisições de entrada, processam as requisições e enviam o resultado das requisições de volta para os clientes.



Sistemas Distribuídos

PROTOCOLO

- É uma convenção que controla e que possibilita uma conexão, uma comunicação, uma transferência dados entre dois sistemas computacionais.
- São regras que governam a sintaxe, a semântica e a sincronização da comunicação.
- Cada serviço tem um protocolo próprio que define como deve ser a requisição do cliente e como deve ser a resposta do servidor.

Sistemas Distribuídos

PORTA

- Para o sistema operacional do servidor, cada serviço é identificado por uma porta.
- Porta é um número usado para identificar para qual programa as requisições daquele serviço devem ser encaminhados.
- O número da porta de cada serviço é definido na configuração do programa servidor, não sendo possível dois programas distintos ouvirem a mesma porta do sistema operacional.
- Alguns serviços já possuem portas padrões:

- HTTP: Porta 80	- SMTP: Porta 25
- HTTPS: Porta 443	- DNS: Porta 53
- FTP: Portas 20/21	- SSH: Porta 22

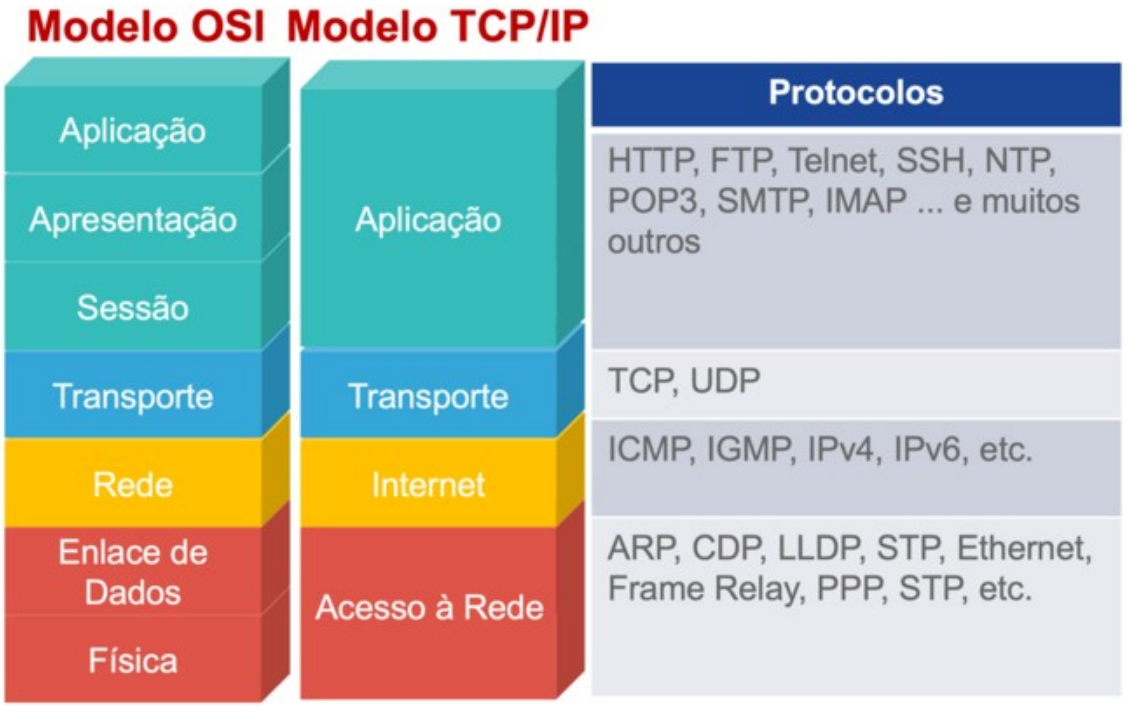
Sistemas Distribuídos

CAMADA DE REDE

- As redes de computadores são divididas em camadas para simplificar a implementação dos softwares e dos hardwares da rede.
- Existem diferentes modelos de rede com diferentes estruturas de camada:
  - O modelo OSI, que divide a rede em 7 camadas, é um modelo de referência da ISO que tinha como objetivo ser um modelo padrão para protocolos de computação entre os mais diversos sistemas.
  - O modelo TCP/IP que divide em 4 camadas.

Sistemas Distribuídos

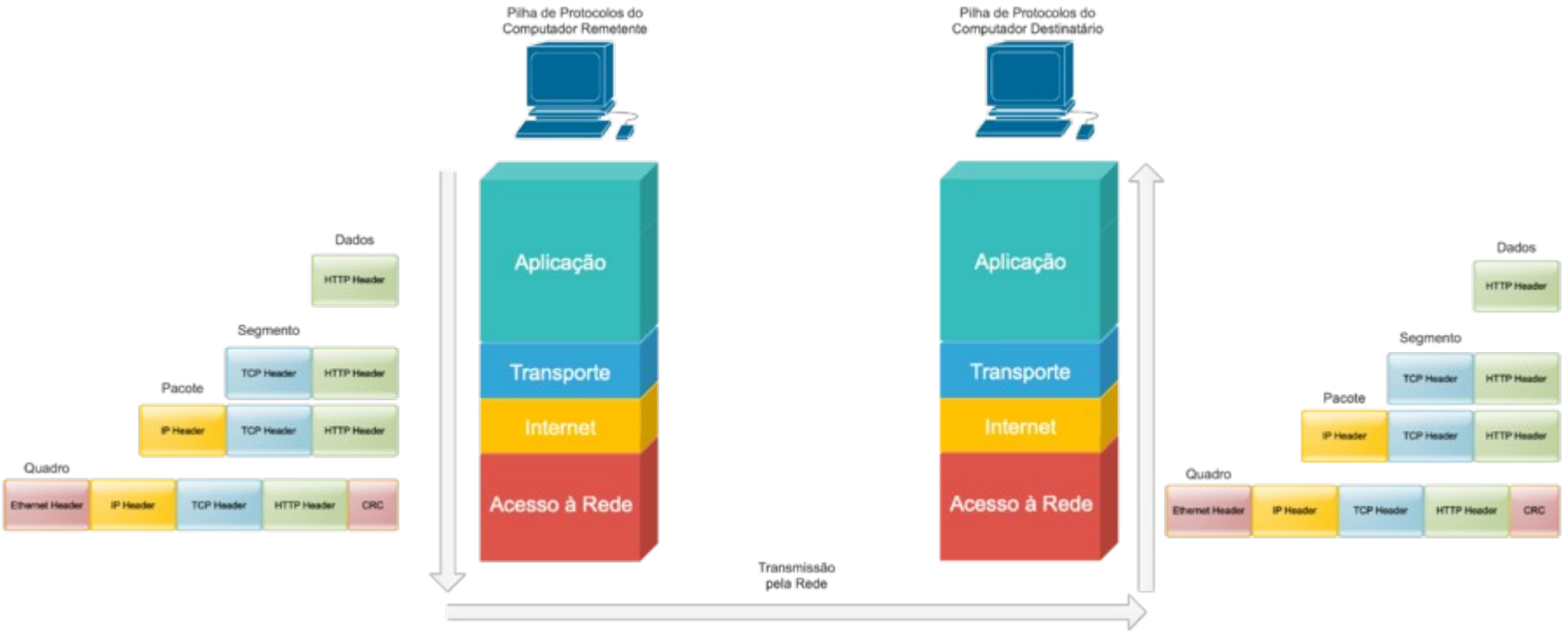
CAMADA DE REDE



Fonte: https://wiki.brasilpeeringforum.org/w/Fundamentos_de_Roteamento_para_Provedores

Sistemas Distribuídos

CAMADA DE REDE



Fonte: https://wiki.brasilpeeringforum.org/w/Fundamentos_de_Roteamento_para_Provedores

Sistemas Distribuídos

HTTP

- O **Hypertext Transfer Protocol (HTTP)** é um protocolo da camada de aplicação do TCP/IP cuja função é de proporcionar a transferência de hipertexto.
- Características do Protocolo HTTP:
  - É um protocolo de camada de aplicação da WEB.
  - É implementado em dois programas: Cliente e Servidor.
  - O HTTP é quem define a estrutura da mensagem que o cliente vai trocar com o servidor e utiliza TCP como seu protocolo de transporte.
  - Protocolo sem estado, isto é, ele não mantém memória / histórico sobre suas ações.

Sistemas Distribuídos

HTTPS

- O **Hyper Text Transfer Protocol Secure (HTTPS)** é uma implementação do protocolo HTTP sobre uma camada adicional de segurança que utiliza o protocolo **SSL/TLS**.
  - Essa camada adicional de segurança permite:
    - Transmitir os dados por meio de uma conexão criptografada
    - Verificar a autenticidade do servidor e do cliente por meio de certificados digitais.
- O protocolo HTTPS é utilizado, em regra, quando se deseja evitar que a informação transmitida entre o cliente e o servidor seja visualizada por terceiros.

Sistemas Distribuídos

URI

- O **Uniform Resource Identifier (URI)** é uma sequência única de caracteres que identifica um recurso lógico ou físico usado por tecnologias da web.
  - Propósito desta identificação é permitir a interação com representações do recurso através de uma rede.
- Um URI pode ser classificado como:
  - **Uniform Resource Locator (URL)**
    - Define um método para localizar e para recuperar um recurso.
  - **Uniform Resource Name (URN)**
    - Define uma identidade única para um recurso.

Sistemas Distribuídos

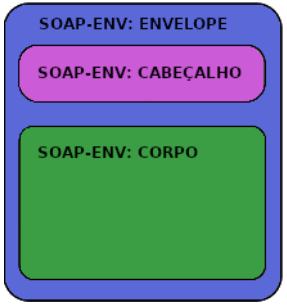
WEB SERVICE

- **Web service** é uma **arquitetura de desenvolvimento** cujo objetivo é facilitar a troca de informações entre diferentes sistemas através da rede utilizando protocolos de comunicação.
  - Surgiu com o objetivo substituir mecanismos e protocolos próprios utilizados anteriormente.
  - Permite às aplicações enviar e receber dados.
    - Cada aplicação pode ter sua própria linguagem que é traduzida para um formato intermediário como XML, JSON, CSV, etc.
  - Utiliza a arquitetura cliente-servidor.
  - Serviço fica acessível através de um endereço, de um endpoint.
- Usualmente, a implementação de Web Services segue o protocolo **SOAP** ou a arquitetura **REST**.

Sistemas Distribuídos

WEB SERVICE - SOAP

- O **Simple Object Access Protocol (SOAP)** é um protocolo para troca de informações estruturadas em uma **plataforma descentralizada e distribuída**.
  - Utiliza o **XML** como formato de mensagem.
  - Normalmente utiliza o **HTTP** para transmissão de mensagens. Também pode ser utilizado SMTP, FTP, etc.
- Consiste de três partes:
  - Um **envelope** que define o que está na mensagem e como processá-la.
  - Um **cabeçalho** com conjunto de informações relacionadas ao aplicativo que são processadas por nós SOAP ao longo do fluxo de mensagens.
    - Por exemplo, contém IP de origem, autenticação.
  - Um **corpo** com convenções para representar chamadas de procedimentos e de respostas.
    - Por exemplo: contém o nome do método que deseja chamar, objeto que será enviado como *payload* da requisição.



Sistemas Distribuídos

WEB SERVICE - SOAP

- Nesse exemplo, a requisição ao método **GetRevista** é enviada para o servidor juntamente com o objeto **RevistaNome**, que servirá de parâmetro de entrada ao método e tem, como conteúdo, a String **“Java Magazine”**.

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header>
  </soap:Header>
  <soap:Body>
    <m:GetRevista xmlns:m="http://www.example.org/revista">
      <m:RevistaNome>Java Magazine</m:ResvistaNome>
    </m:GetRevista>
  </soap:Body>
</soap:Envelope>
```

Fonte: https://www.devmedia.com.br/web-services-rest-versus-soap/32451

Sistemas Distribuídos

WEB SERVICE - SOAP

- O SOAP utiliza a linguagem **WSDL (Web Services Description Language)** para definir os métodos que o Web Service disponibiliza.
  - O WSDL descreve os objetos e os serviços disponibilizados a rede por meio da semântica XML.
  - O WSDL indica a um cliente como compor uma solicitação de um serviço e descreve a interface que é fornecida pelo web service.

```
<wsdl:definitions targetNamespace="http://math.example.com" name="MathFunctionsDef">
<wsdl:message name="addIntResponse">
        <wsdl:part name="addIntReturn" type="xsd:int" />
</wsdl:message>
<wsdl:message name="addIntRequest">
        <wsdl:part name="a" type="xsd:int" />
        <wsdl:part name="b" type="xsd:int" />
</wsdl:message>
<wsdl:portType name="AddFunction">
        <wsdl:operation name="addInt" parameterOrder="a b">
                <wsdl:input message="impl:addIntRequest" name="addIntRequest" />
                <wsdl:output message="impl:addIntResponse" name="addIntResponse" />
        </wsdl:operation>
</wsdl:portType>
<service name="MathFunctions"/>
</wsdl:definitions>
```

Fonte: https://www.researchgate.net/publication/265265754_Approximate_Service_Retrieval

Sistemas Distribuídos

WEB SERVICE - REST

- O **Representational State Transfer (REST)** é uma arquitetura que propõe que o web service disponibilize operações padrões a serem executadas sobre os recursos do sistema.
- Tem como proposta ser uma forma simples de interação entre cliente e servidor.
  - Idealizado para atender as necessidades de aplicações mobile e serviços web simples.
  - Tem por objetivo ser flexível, rápida e de fácil configuração.
- Utiliza o HTTP para transmissão de mensagem.
  - Utiliza métodos do HTTP para definir as operações disponíveis sobre os recursos.
- Retornam mensagens em diversos formatos: HTML, XML, JSON, texto simples.

Sistemas Distribuídos

WEB SERVICE - REST

- Para ser RESTful, a aplicação deve seguir algumas regras:
  - Arquitetura ser formada por recursos, servidores e clientes.
  - Comunicação stateless entre servidor e cliente.
    - Servidor não armazena informações sobre o cliente.
    - Cliente armazena informações o estado da sessão.
  - Dados que podem ser armazenados em cache para eliminar a necessidade algumas interações entre cliente e servidor.
  - Uma interface uniforme de componentes para que as informações sejam transferidas em um formato uniforme.

Sistemas Distribuídos

WEB SERVICE - REST

- Principais métodos do protocolo HTTP e o cenário de utilização

Método	Utilização
GET	Obter os dados de um recurso
POST	Criar um novo recurso
PUT	Substituir os dados de um determinado recurso
DELETE	Excluir um determinado recurso
HEAD	Similar ao GET, mas utilizado apenas para se obter o cabeçalho de resposta, sem os dados em si
OPTIONS	Obter quais manipulações podem ser realizadas em um determinado recurso

Sistemas Distribuídos

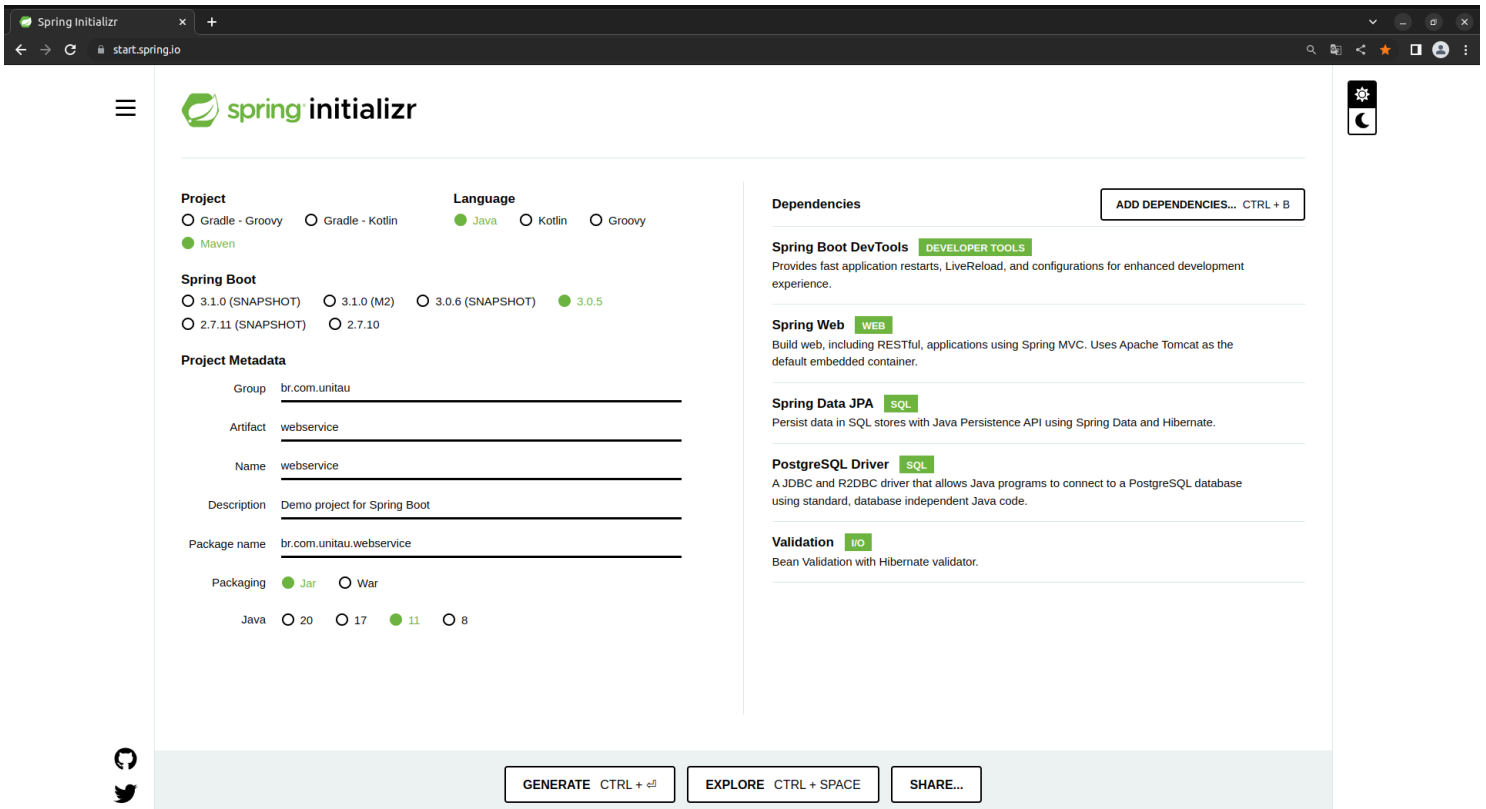
WEB SERVICE - EXEMPLO

- Como exemplo, será apresentado um Web Service criado com **Spring**.

Comando	Endereço	Descrição
GET	`http://127.0.0.1:8080/student`	Obter a lista dos estudantes
GET	`https://127.0.0.1:8080/student/{id}`	Obter os dados de um estudante pelo id
POST	`https://127.0.0.1:8080/student`	Inserir um novo estudante
PUT	`https://127.0.0.1:8080/student/{id}`	Alterar um estudante pelo id
DELETE	`https://127.0.0.1:8080/student/{id}`	Deletar um estudante pelo id

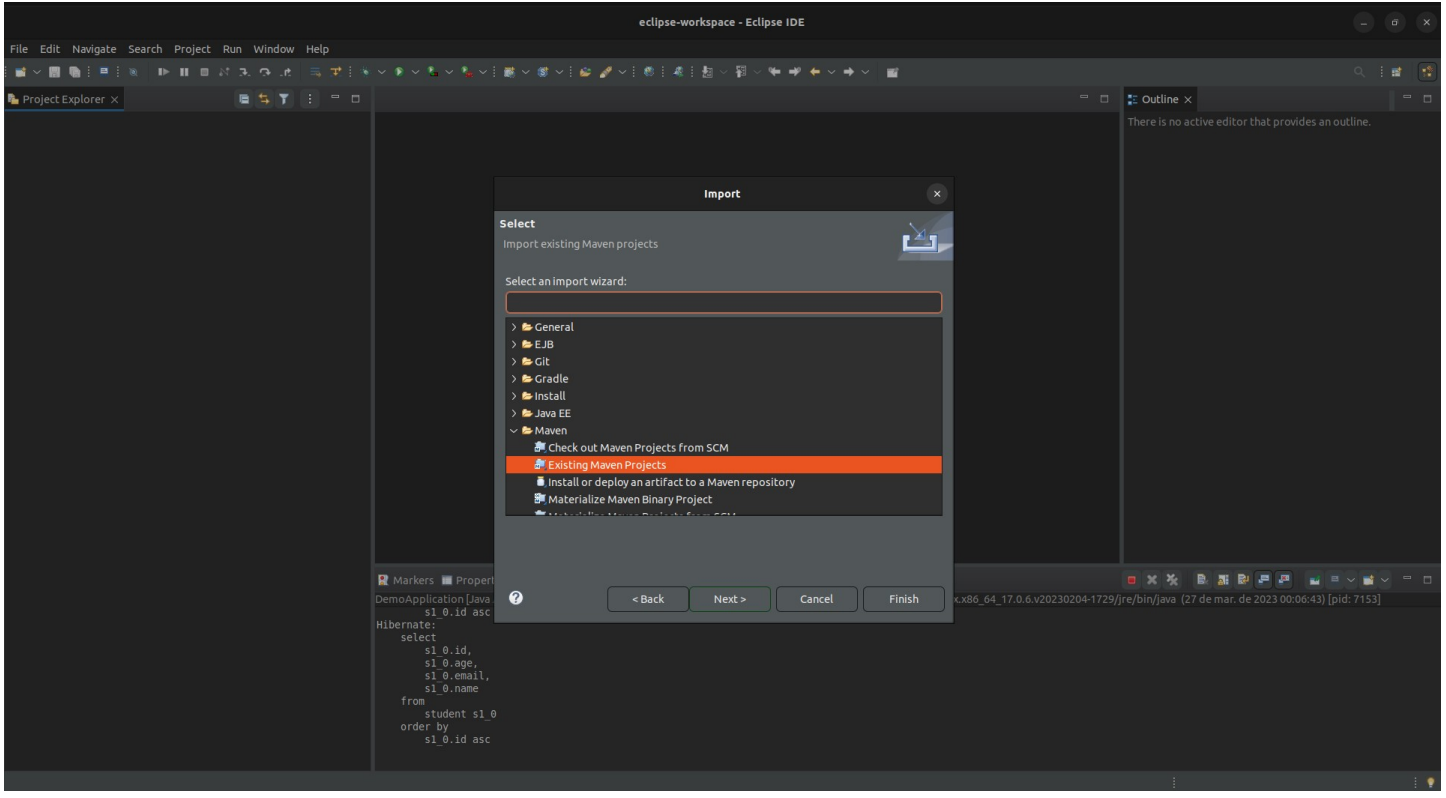
Sistemas Distribuídos

WEB SERVICE – EXEMPLO – SPRING BOOT



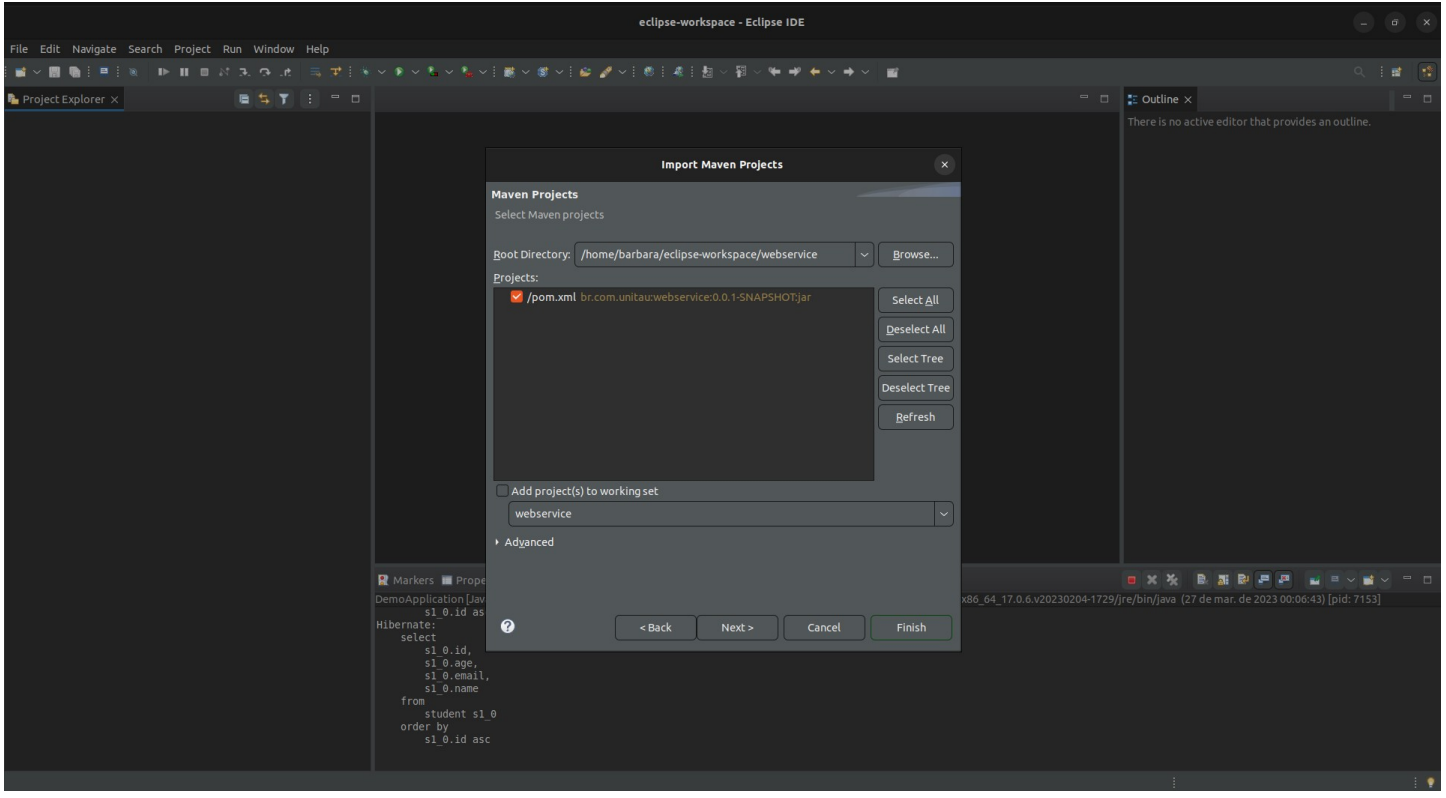
Sistemas Distribuídos

WEB SERVICE – EXEMPLO – ECLIPSE



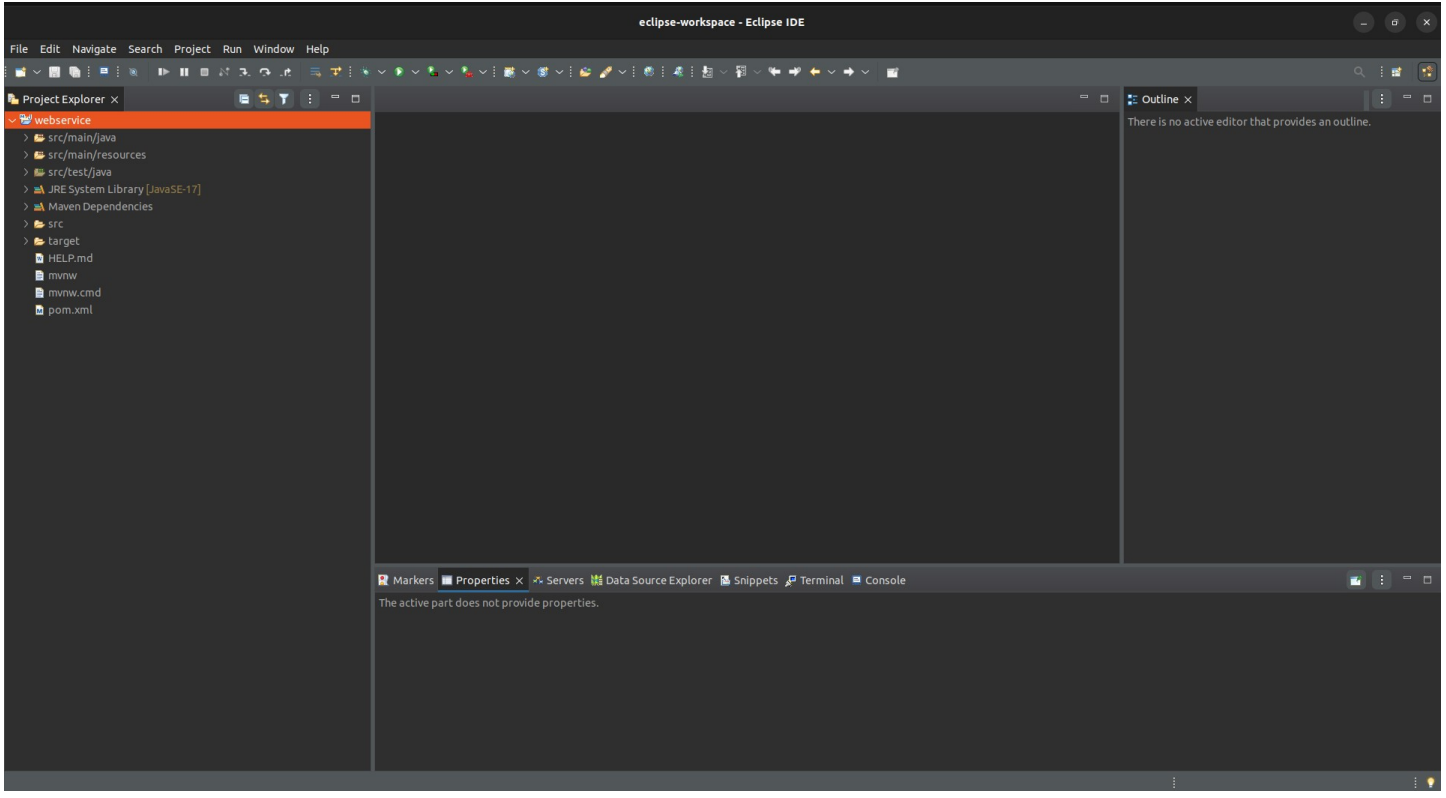
Sistemas Distribuídos

WEB SERVICE – EXEMPLO – ECLIPSE



Sistemas Distribuídos

WEB SERVICE – EXEMPLO – ECLIPSE



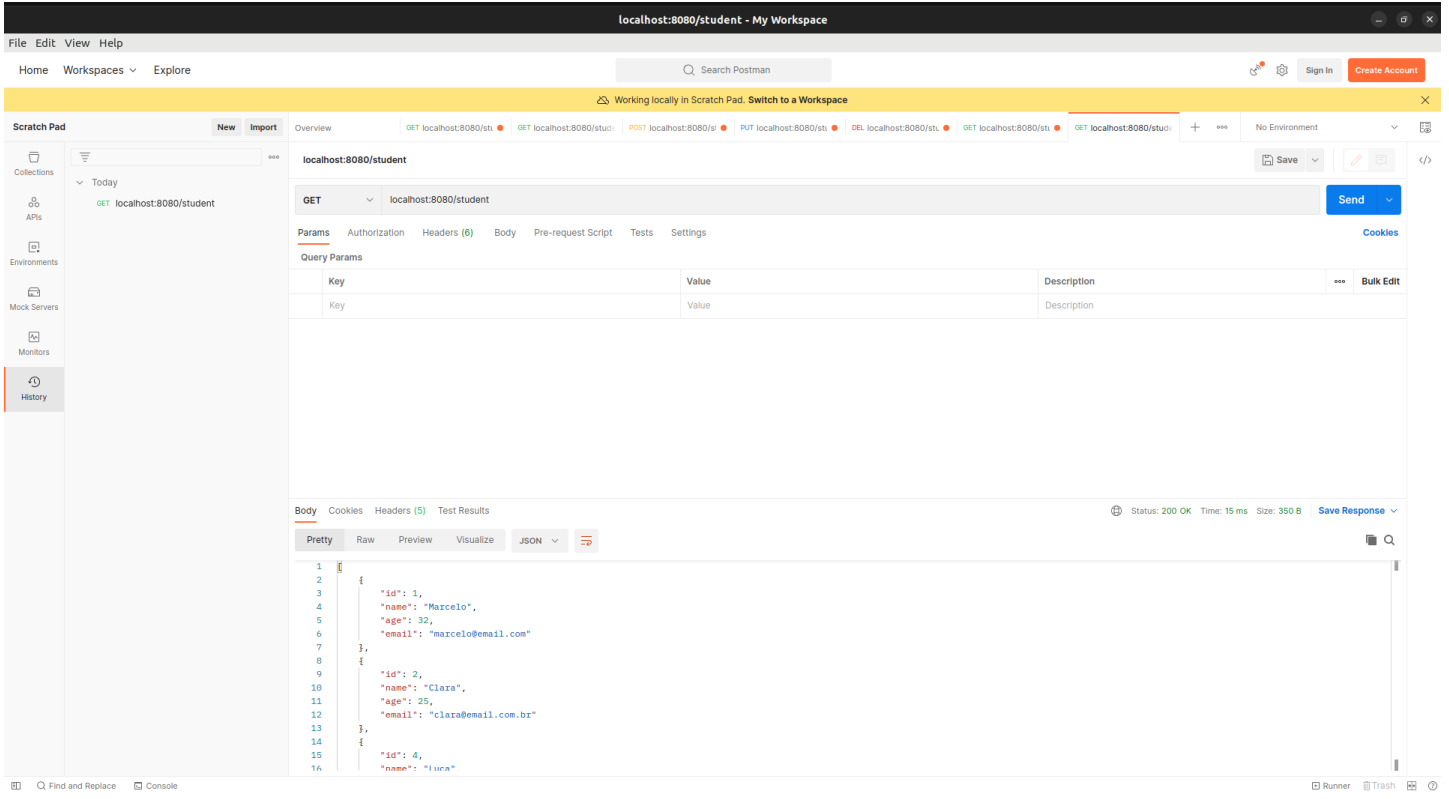
Sistemas Distribuídos

WEB SERVICE – EXEMPLO – TESTE

- O Web Service pode ser testado de várias formas, usando o navegador, o Postman, o Curl, etc.

Sistemas Distribuídos

WEB SERVICE – EXEMPLO – TESTE – POSTMAN



Sistemas Distribuídos

WEB SERVICE – EXEMPLO – TESTE – CURL

Sistemas Distribuídos

WEB SERVICE – EXEMPLO – TESTE – CURL