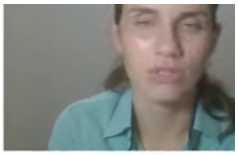
Introdução ao HTTP/HTTPS

▼ Protocolo HTTP

- Formação
  - Mestre em Modelagem Computacional – LNCC
  - Engenheira da Computação - UCP
  - Técnica de Telecomunicações - CEFET-RJ
- Contatos:
  - Linkedin: Juliana Mascarenhas,
  - https://github.com/julianazanelatto/

DIGITAL INNOVATION ONE

Objetivo do curso

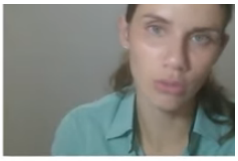


O Dev será capaz de aprender a estrutura e funcionamento do protocolo HTTP, assim como o HTTPS. Assuntos correlatos que formam a base para completa compreenção do DEV serão abordados.

Além disso, colocará em prática o acionamento do protocolo através de APIs JAVA.

DIGITAL INNOVATION ONE

Percurso

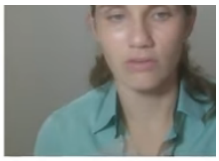


Aula 1 Protocolo HTTP

Aula 2 Interface de Programação de Aplicações -API

DIGITAL INNOVATION ONE

Requisitos



Conceitos básicos:

- ✓ Criptografia por chave
- ✓ Certificação digital

Conceitos básicos necessários apresentados neste curso

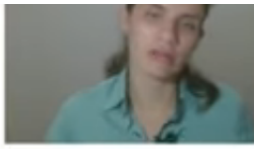
DIGITAL INNOVATION ONE

Objetivos

1. Protocolo HTTP
2. Conceitos básicos de segurança
3. Protocolo SSL

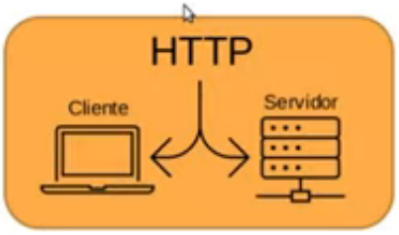
DIGITAL INNOVATION ONE

Protocolo HTTP



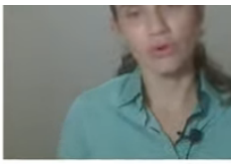
HyperText Transfer Protocol (HTTP)

- Definido pelos RFCs 1945 e 2116
- Protocolo de comunicação
  Rege a estrutura as mensagens
- Browser -> implementa o cliente HTTP
- Servidor -> host objetos web



DIGITAL INNOVATION ONE

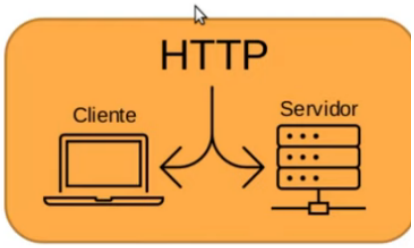
Protocolo HTTP



HyperText Transfer Protocol (HTTP)

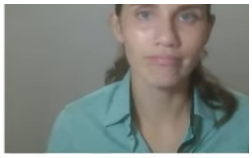
Objetos web

- Cliente
  Mensagens - Request HTTP
- Servidor
  Mensagens – Response HTTP



DIGITAL INNOVATION ONE

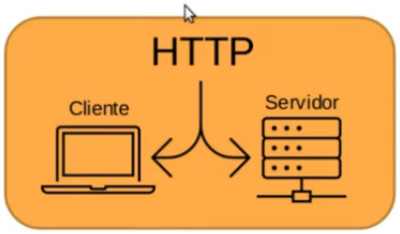
Protocolo HTTP

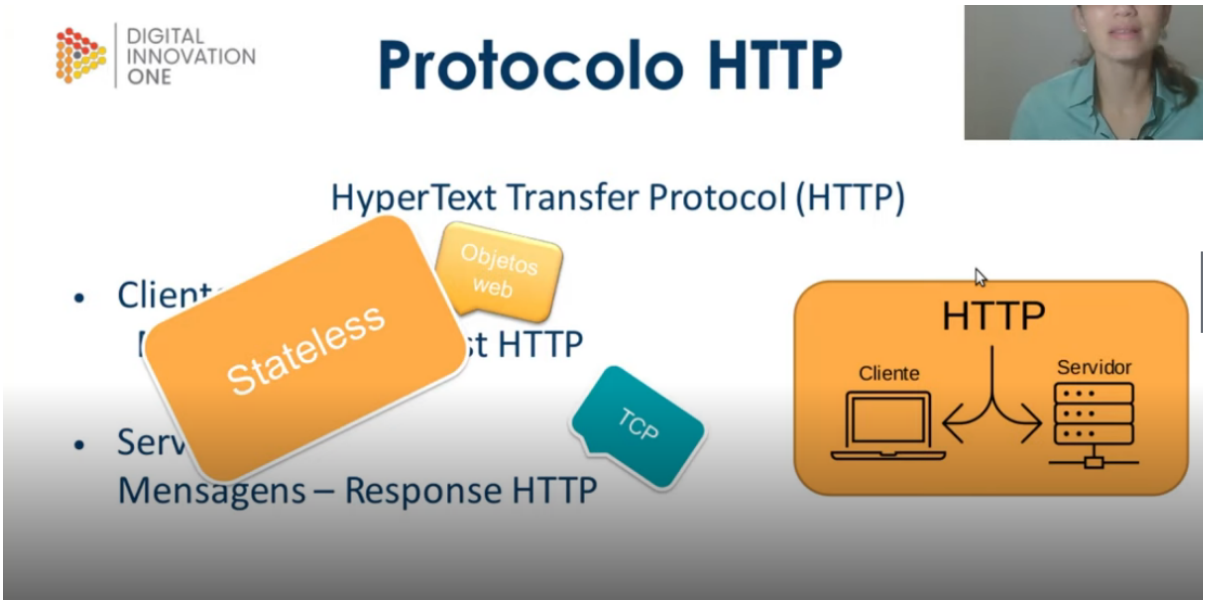


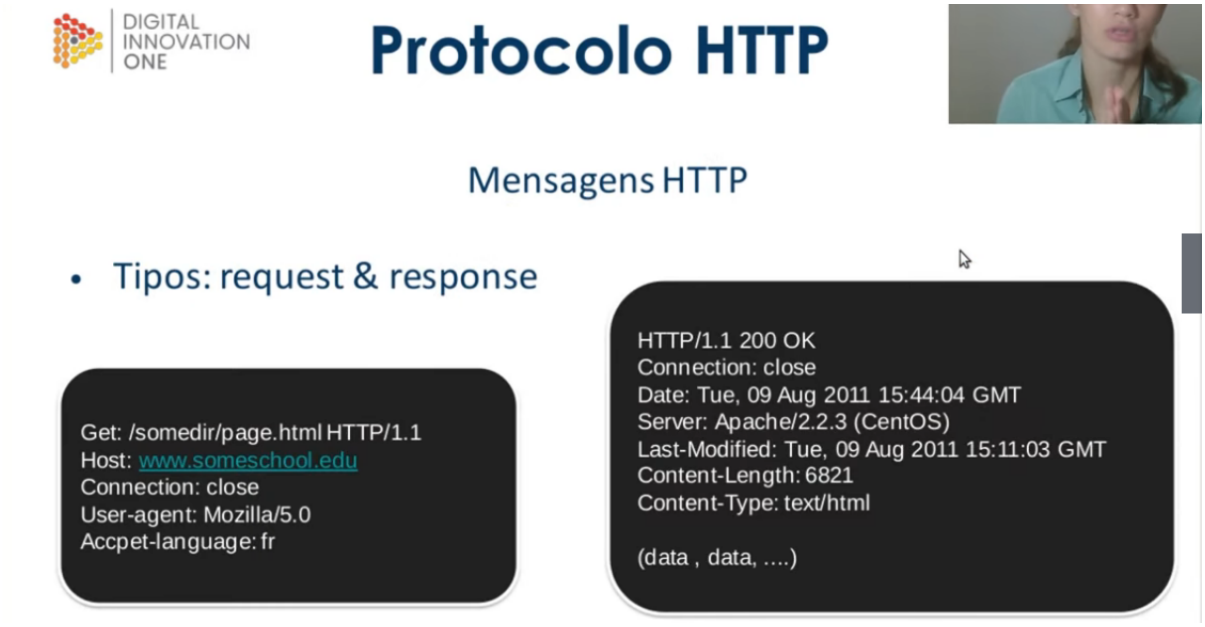
HyperText Transfer Protocol (HTTP)

Arquitetura Client-server

- Cliente
  Mens
- Serv
  Mens




DIGITAL INNOVATION ONE
Protocolo HTTP
HyperText Transfer Protocol (HTTP)
Objetos web
Stateless
TCP
HTTP
Cliente
Servidor
Mensagens – Response HTTP


DIGITAL INNOVATION ONE
Protocolo HTTP
Mensagens HTTP
Tipos: request & response
Get: /somedir/page.html HTTP/1.1
Host: www.someschool.edu
Connection: close
User-agent: Mozilla/5.0
Accpet-language: fr
HTTP/1.1 200 OK
Connection: close
Date: Tue, 09 Aug 2011 15:44:04 GMT
Server: Apache/2.2.3 (CentOS)
Last-Modified: Tue, 09 Aug 2011 15:11:03 GMT
Content-Length: 6821
Content-Type: text/html
(data , data, ...)

DIGITAL INNOVATION ONE

Protocolo HTTP

Mensagens HTTP - Request

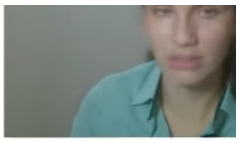
Texto em ASCII

- Estrutura:

```
Get: /somedir/page.html HTTP/1.1
Host: www.someschool.edu
Connection: close
User-agent: Mozilla/5.0
Accept-language: fr
```

DIGITAL INNOVATION ONE

Protocolo HTTP



Search fields

Mensagens HTTP - Request

- Entity body

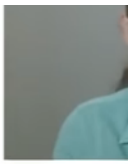
  Campo da mensagem request HTTP. Não utilizada pelo GET, mas pelo método POST

```
Get: /somedir/page.html HTTP/1.1
Host: www.someschool.edu
Connection: close
User-agent: Mozilla/5.0
Accpet-language: fr
```

- Metódo GET – 90%

DIGITAL INNOVATION ONE

Protocolo HTTP



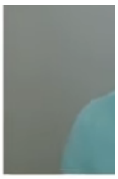
Mensagens HTTP

- Métodos

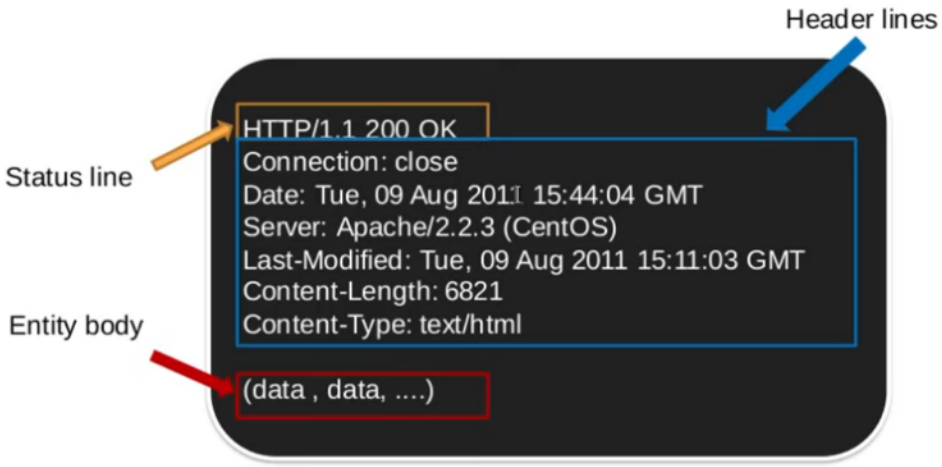
GET	solicita um recurso do servidor
HEAD	GET sem corpo de resposta
POST	submete uma entidade a um recurso
PUT	substituição de recursos pelos dados da requisição
DELETE	remoção de um recurso
TRACE	chamada de loop-back a um determinado recurso
OPTION	opções de comunição com recurso
CONNECT	tunelamento identificado pelo recurso
PATCH	modificação parcial

DIGITAL INNOVATION ONE

Protocolo HTTP

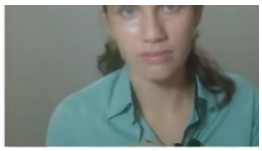


Mensagens HTTP - Response



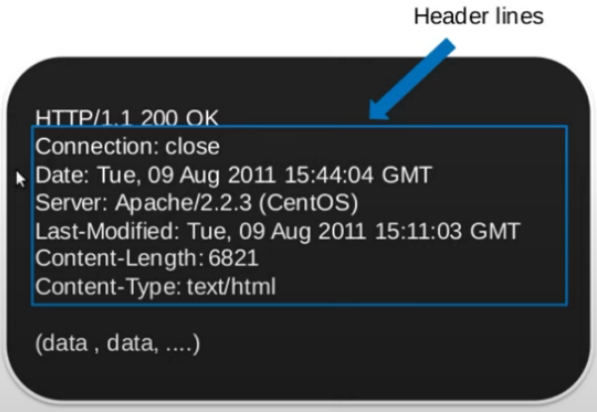
DIGITAL INNOVATION ONE

Protocolo HTTP



Mensagens HTTP - Response

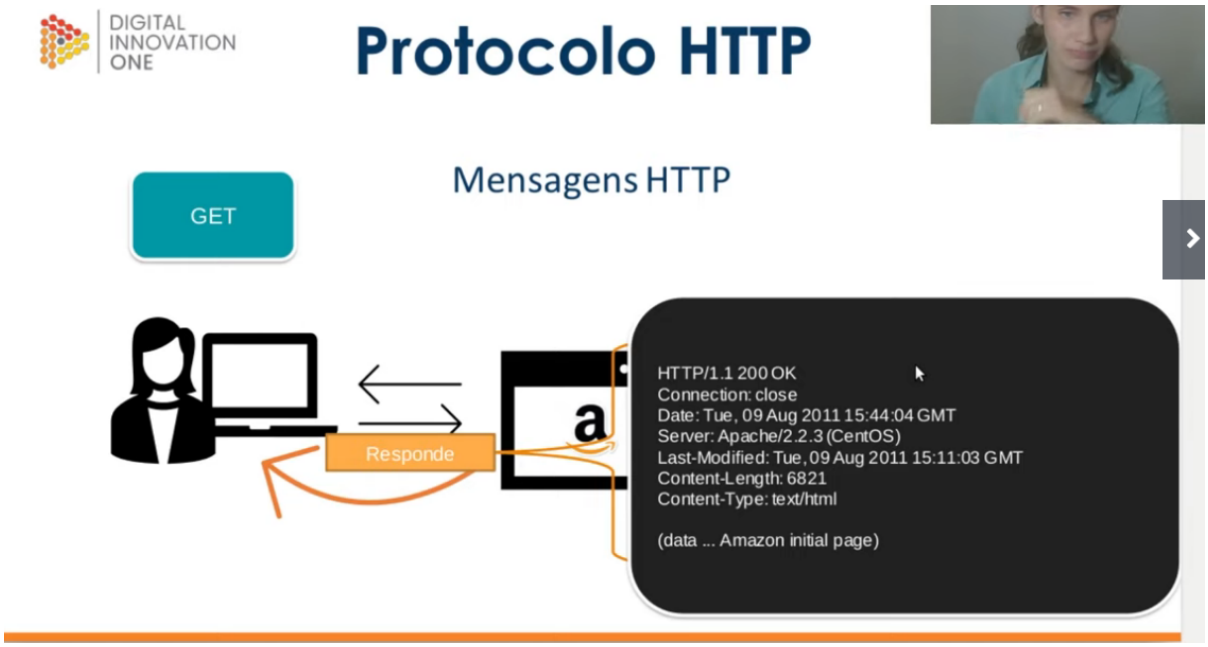
- Conexão encerrada
- Dados da mensagem:
  Data, servidor, …
- Content-type:
  Tipo de dado

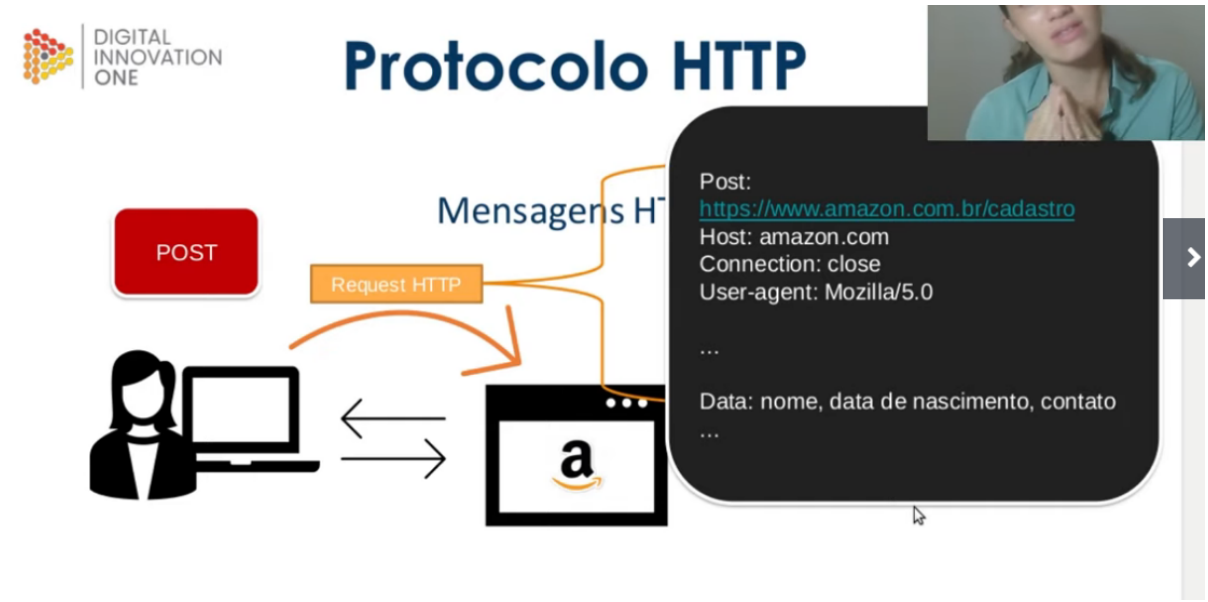


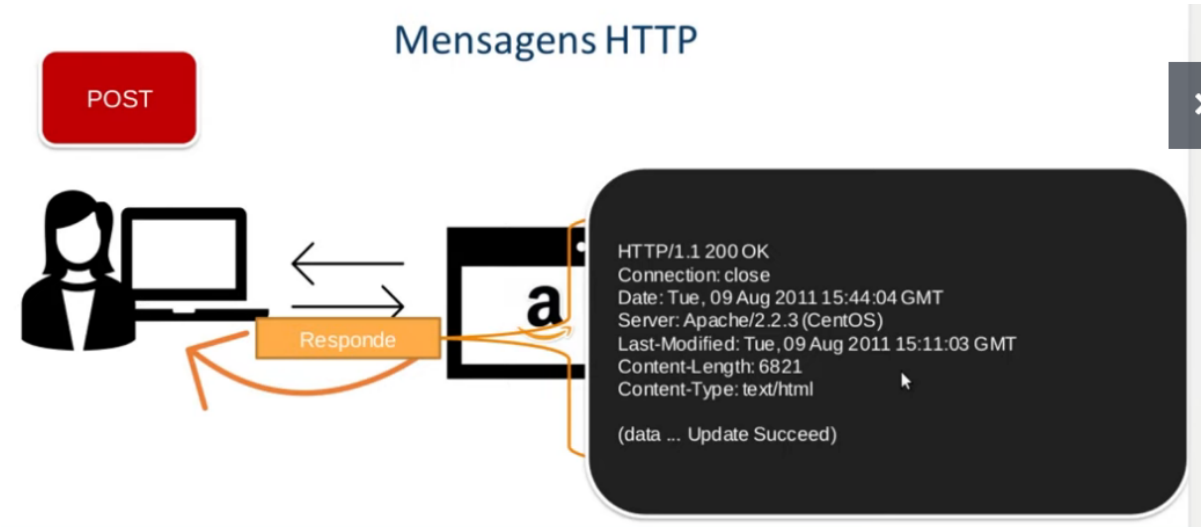
Mensagens HTTP - Response

- Status code

200 OK:	request bem sucedida e objeto enviado
301 Moved Permanently:	objeto realocado nova URL no campo Location
400 Bad Request:	resposta genérica - servidor não entendeu a mensagem
404 Not Found:	o documento solicitado inexistente
505 HTTP Version Not Supported:	versão do protocolo não suportada pelo servidor


DIGITAL INNOVATION ONE
Protocolo HTTP
Mensagens HTTP
GET
Responde
HTTP/1.1 200 OK
Connection: close
Date: Tue, 09 Aug 2011 15:44:04 GMT
Server: Apache/2.2.3 (CentOS)
Last-Modified: Tue, 09 Aug 2011 15:11:03 GMT
Content-Length: 6821
Content-Type: text/html
(data ... Amazon initial page)


DIGITAL INNOVATION ONE
Protocolo HTTP
POST
Request HTTP
Post:
https://www.amazon.com.br/cadastro
Host: amazon.com
Connection: close
User-agent: Mozilla/5.0
...
Data: nome, data de nascimento, contato
...



Cookies

- RFC 6265

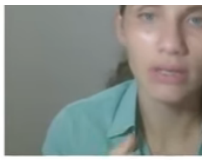
- Rastreamento
- Identificação

Restrição ou fornecimento de funções

Usuários

DIGITAL INNOVATION ONE

Protocolo HTTP



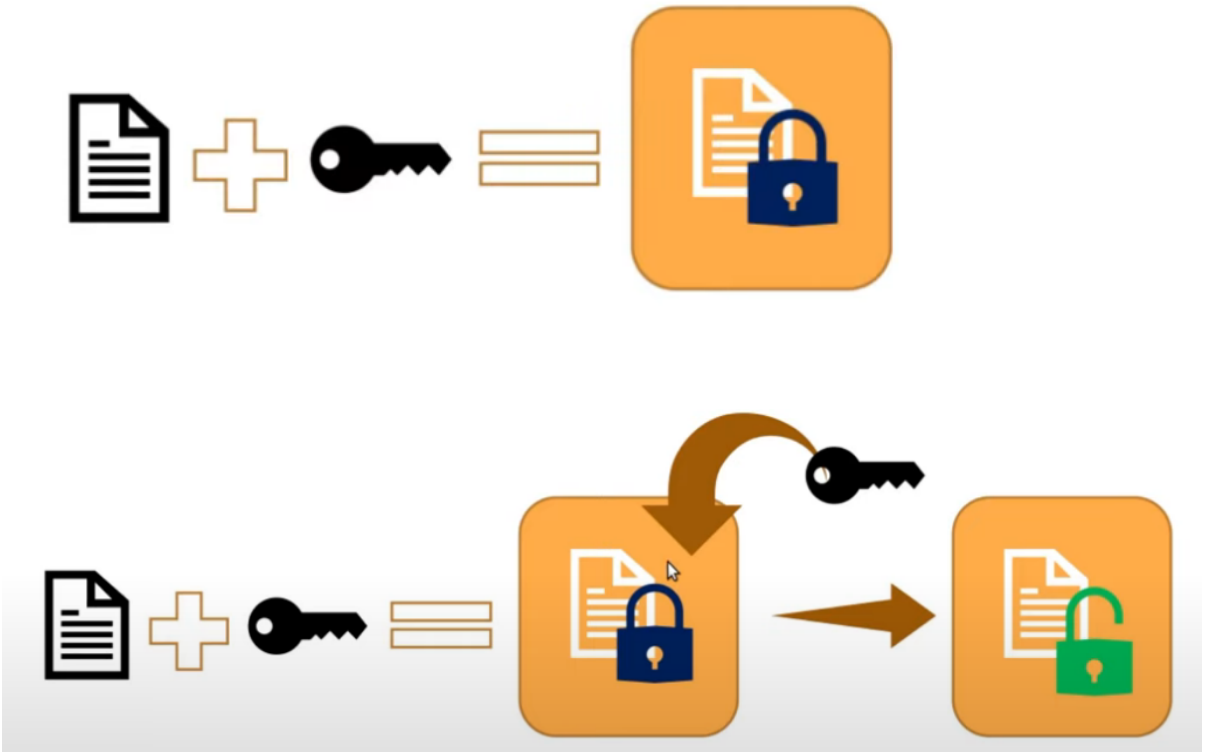
Cookies - Componentes

- Cookie header line
  - Response message
  - Request message
- Cookie file
  - Mantido no cliente e servidor
  - Browser: gerencia
- Banco de Dados Back-end
  - Web site

DIGITAL INNOVATION ONE

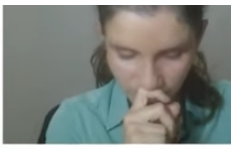
Conceitos básicos

Criptografia por Chave



DIGITAL INNOVATION ONE

Conceitos básicos



Criptografia por Chave

- Assimétrica
- Simétrica

- Chave privada
  Assinatura - criptografia
- Chave pública
  verificação de autenticidade

Criptografia por Chave

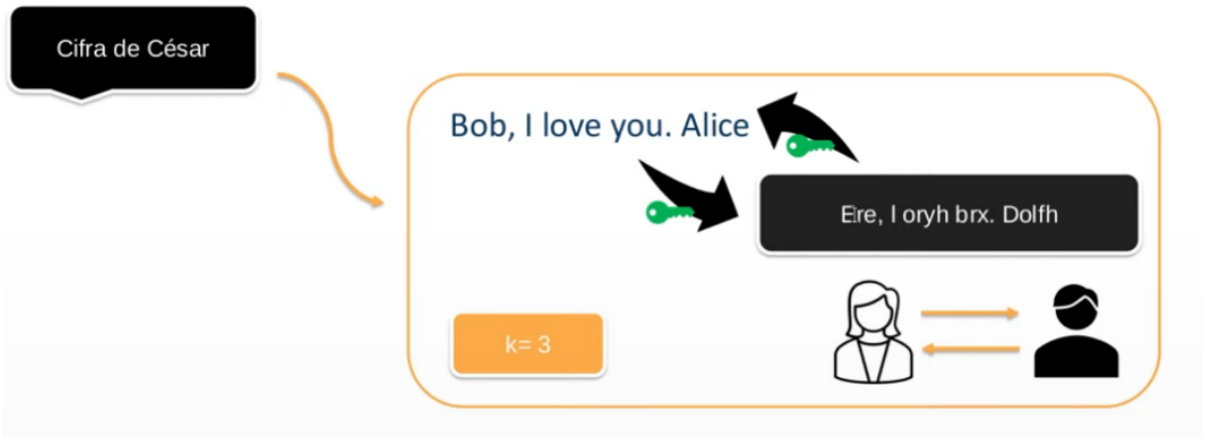
Cifra de César

Funcionamento

Substituição da letra pela k-ésima do alfabeto
Rotatividade do alfabeto

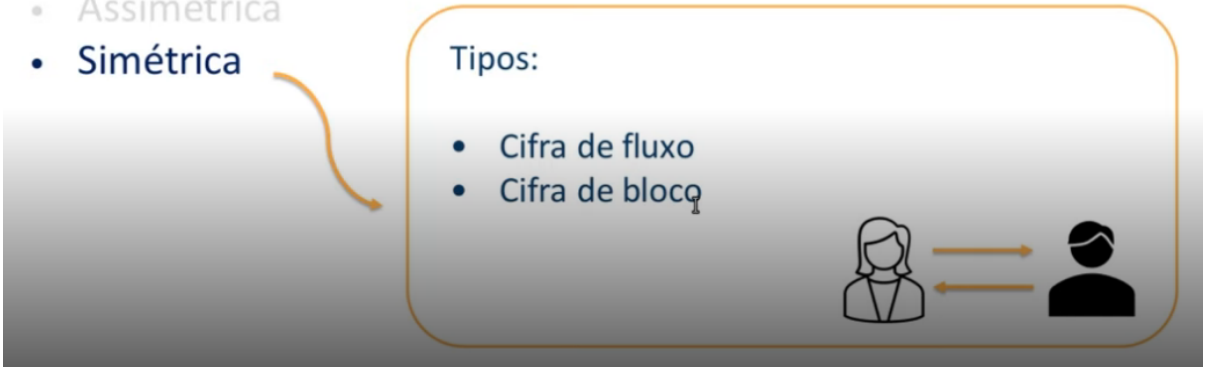
k= [1,26]

Criptografia por Chave



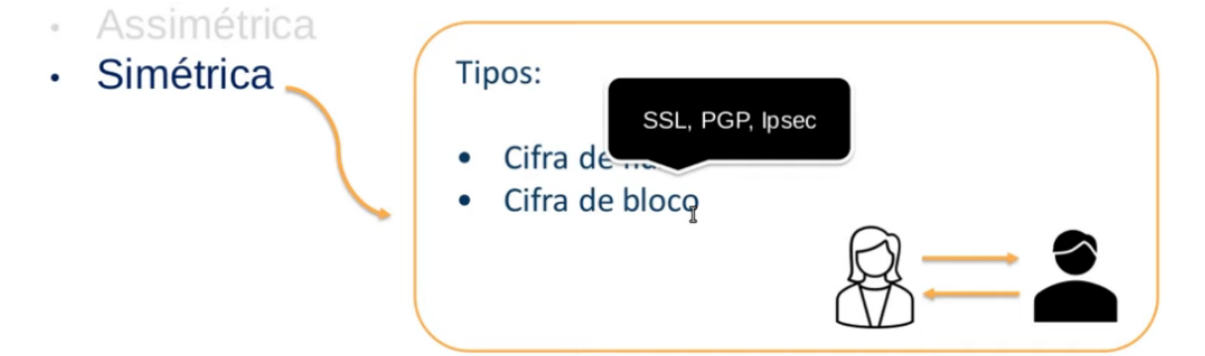
Criptografia por Chave

- Assimétrica
- Simétrica

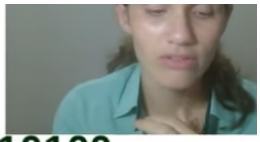


Criptografia por Chave

- Assimétrica
- Simétrica



Conceitos básicos

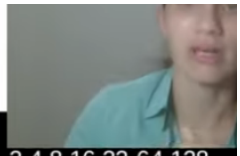


101010100
010100101

Criptografia por Chave

- Cifra de fluxo
  - Sequência de bits pseudo-aleatório
  - Mapeamento 1 para 1

Conceitos básicos



1 2 4 8 16 32 64 128

Criptografia por Chave

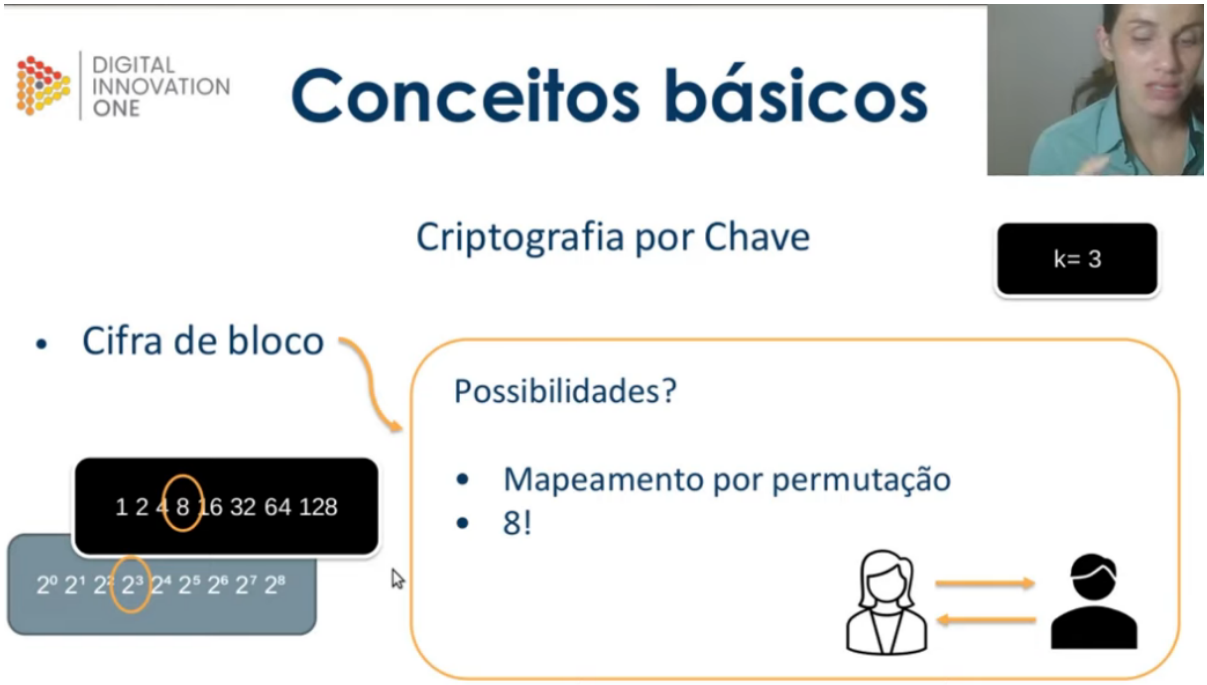
2^0 2^1 2^2 2^3 2^4 2^5 2^6 2^7 2^8

- Cifra de bloco
  - Blocos de bits
    - k= número de bits
    - Ex:

k = 64

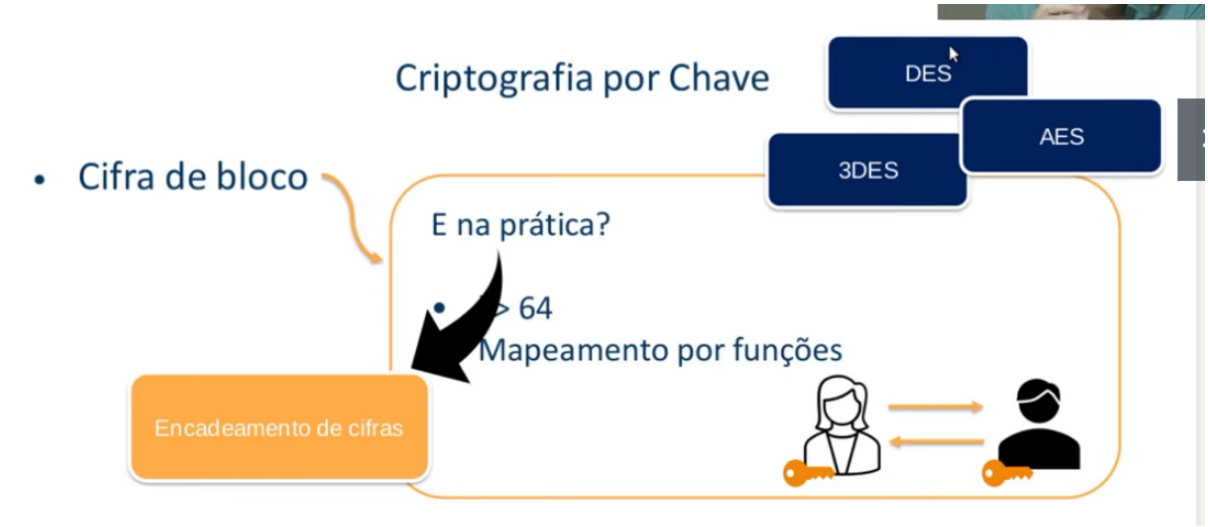
Blocos
64 bits

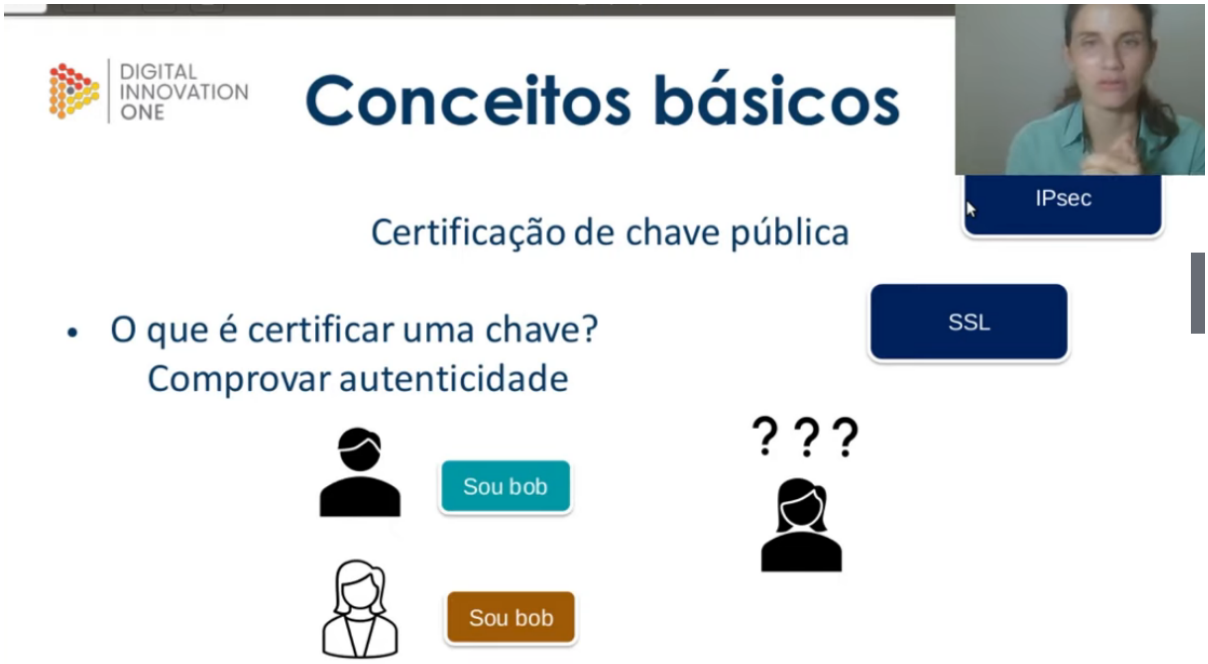

DIGITAL INNOVATION ONE
Conceitos básicos
Criptografia por Chave
k= 3
Cifra de bloco
010 110 001 111
101 000 111 001
Entrada Saída Entrada Saída
000 110 100 011
001 111 101 010
010 101 110 000
011 100 111 001
Fonte: kurose - 5° ed


DIGITAL INNOVATION ONE
Conceitos básicos
Criptografia por Chave
k= 3
Cifra de bloco
Possibilidades?
Mapeamento por permutação
8!
1 2 4 8 16 32 64 128


Criptografia por Chave
k= 3
Cifra de bloco
Possibilidades?
Mapeamento por permutação
8!
1 2 4 8 16 32 64 128
40.320


Criptografia por Chave
Cifra de bloco
E na prática?
k> 64
Mapeamento por funções


Criptografia por Chave
DES
AES
3DES
Cifra de bloco
E na prática?
Mapeamento por funções
Encadeamento de cifras

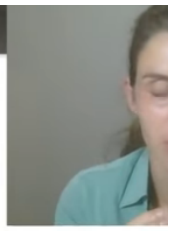

DIGITAL INNOVATION ONE
Conceitos básicos
IPsec
Certificação de chave pública
SSL
O que é certificar uma chave?
Comprovar autenticidade
Sou bob
???
Sou bob


Aula 1 | Etapa 3:
Protocolo SSL
Introdução ao HTTP

of 36 aula1_etapa3.pdf

DIGITAL INNOVATION ONE

Protocolo SSL



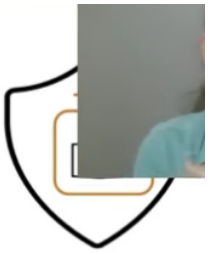
Secure Socket Layer – SSL

- Segurança para conexões TCP

- Confidencialidade
- Integridade
- Autenticidade end-point

DIGITAL INNOVATION ONE

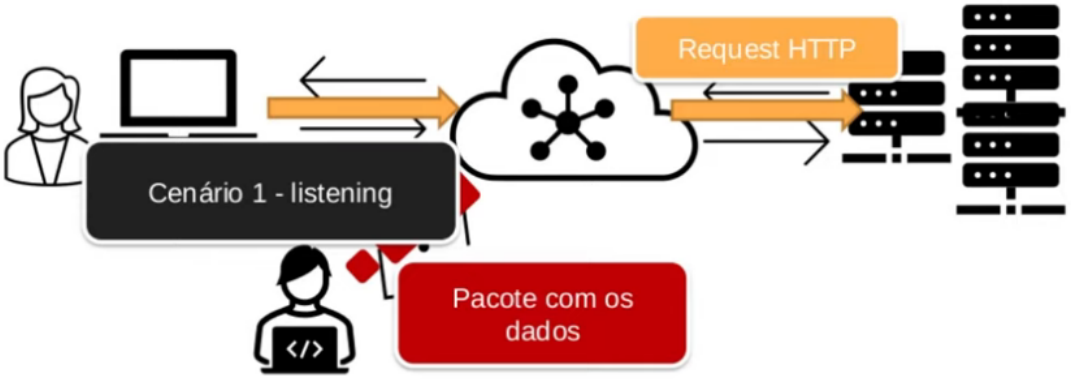
Protocolo SSL



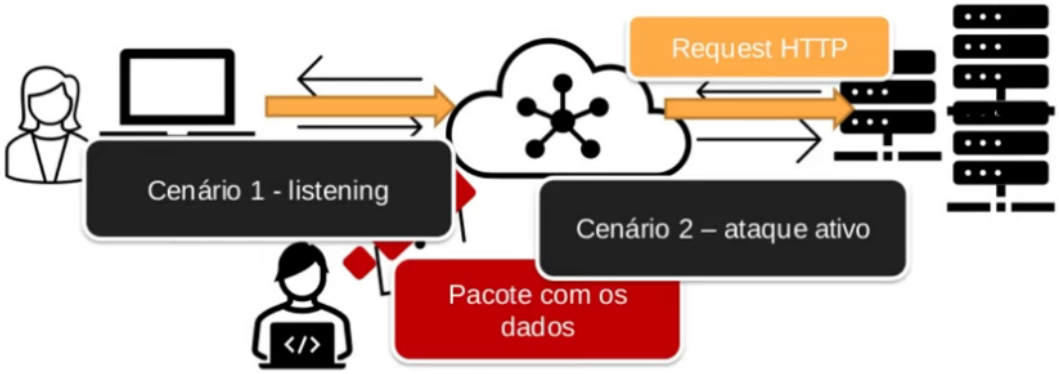
Secure Socket Layer – SSL

- Qual a importancia do SSL?

- Qual a importancia do SSL?

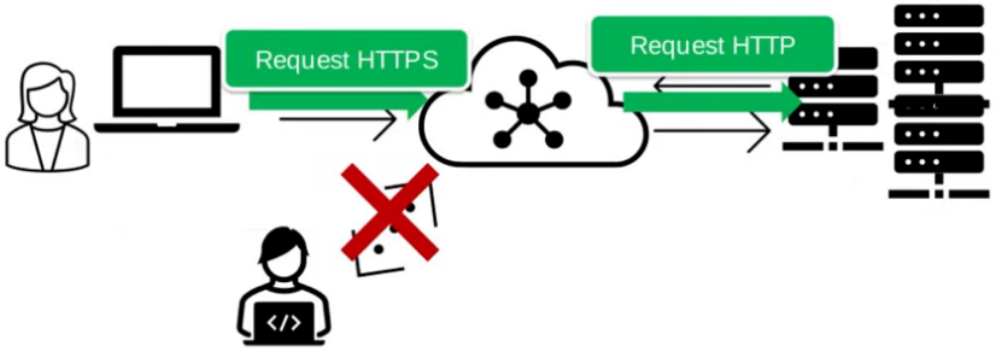


- Qual a importancia do SSL?



Secure Socket Layer – SSL

- Qual a importancia do SSL?



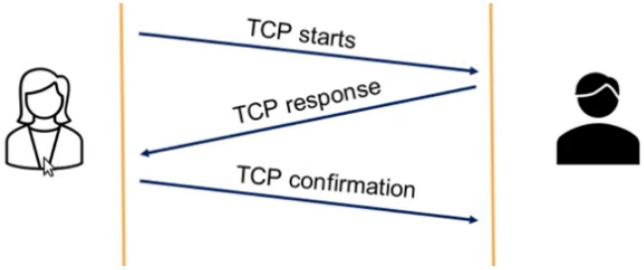
Secure Socket Layer – SSL

- Qual a importancia do SSL?

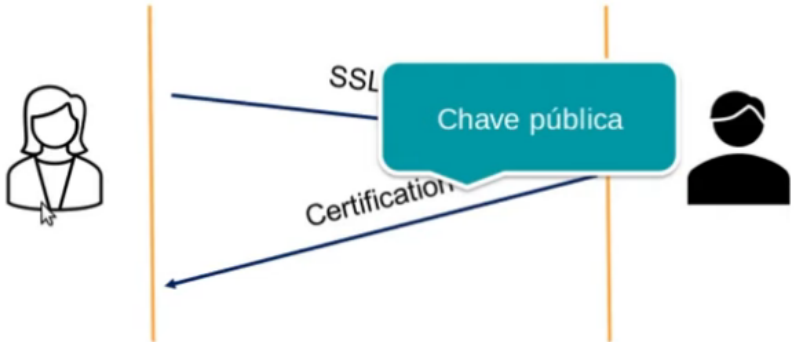
Operação - fases

Operação - fases

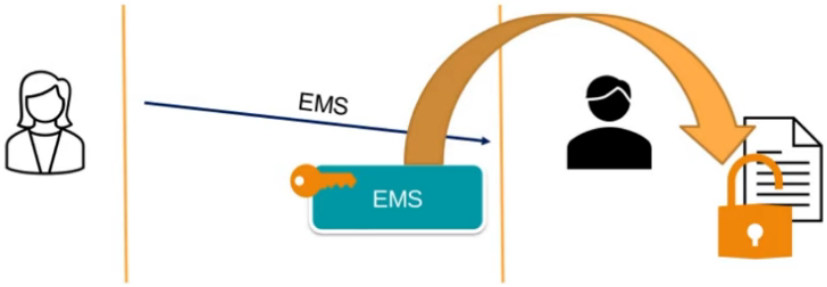
- Estabelecer conexão TCP



- Verificação de autenticidade

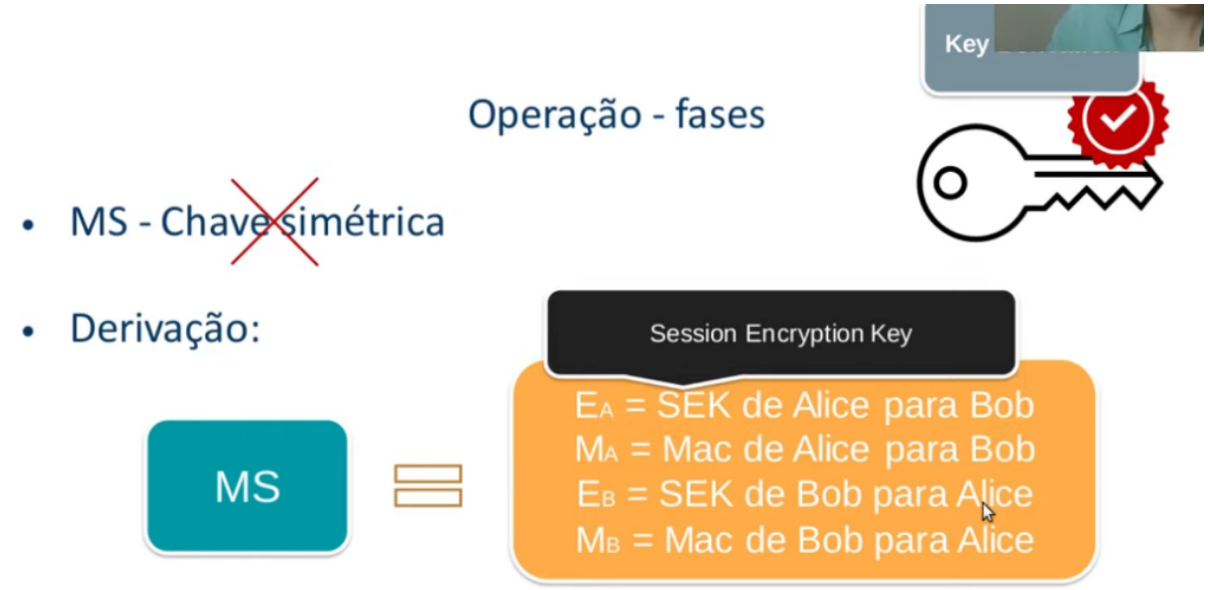


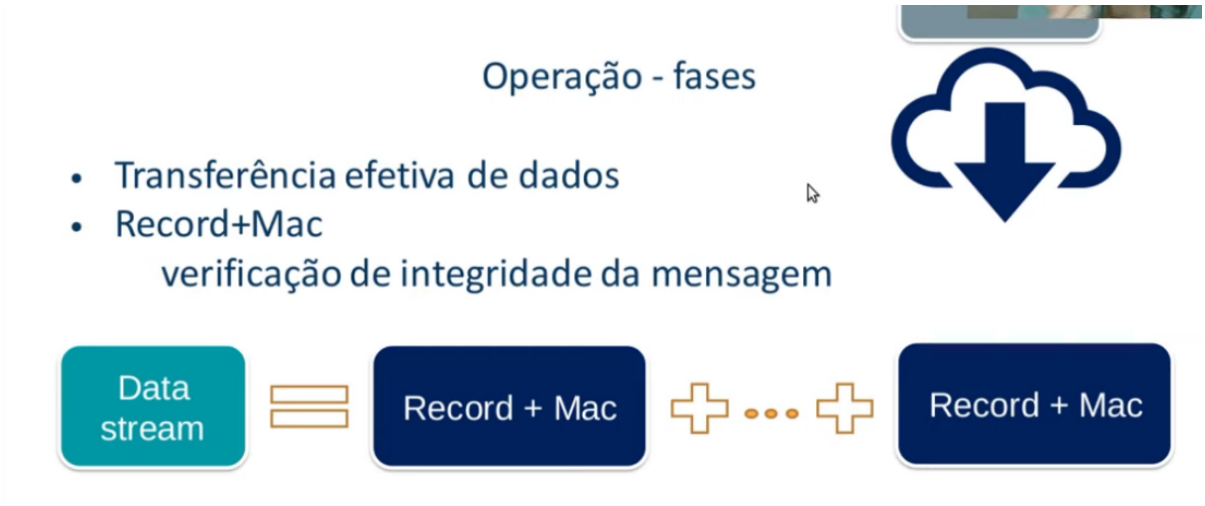
- Envio da Master Secret Key



- Envio da Master Secret Key


Document Viewer
qua 01:12
of 36
aula1_etapa3.pdf
DIGITAL INNOVATION ONE
SSL
Key
Operação - fases
MS - Chave simétrica
Derivação:
MS
Session Encryption Key
EA = SEK de Alice para Bob
MA = Mac de Alice para Bob
EB = SEK de Bob para Alice
MB = Mac de Bob para Alice


Key
Operação - fases
MS - Chave simétrica
Derivação:
MS
Session Encryption Key
EA = SEK de Alice para Bob
MA = Mac de Alice para Bob
EB = SEK de Bob para Alice
MB = Mac de Bob para Alice


Operação - fases
• Transferência efetiva de dados
• Record+Mac
verificação de integridade da mensagem
Data stream
Record + Mac
Record + Mac


Operação - fases
EA
Record + Mac
Record + Mac

Segurança na comunicação - HTTP
Over TCP

Verificação da autenticidade por certificados digitais
Porta 443

DIGITAL INNOVATION ONE

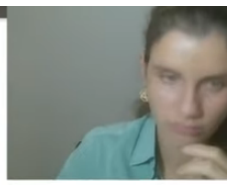
HTTPS

- Há sites que não utilizam? https://whynohttps.com/
- Maioria dos site utilizam a versão segura

Proposta de valor
Proteção fishing e Privacidade

DIGITAL INNOVATION ONE

HTTPS



LGPD – Lei Geral de Proteção de Dados

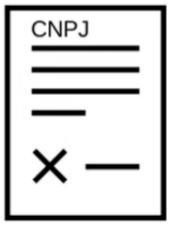
- Promulgada em 2018
- Entrou em vigor em Agosto de 2020
  Lei de segurança da informação

- N° 13.709/2018

LGPD – Lei Geral de Proteção de Dados

O que a lei protege?

- Dados de identificação dos usuários
- Dados sensíveis
  Religião, etnia ...



LGPD – Lei Geral de Proteção de Dados

Criação da ANPD (Agencia Nacional de proteção de dados)

- Zelar pela proteção dos dados;
- Elaboração de diretrizes para politica nacional de proteção;
- Promover conhecimento das normas
- Editar regulamentos
- Realizar auditorias

LGPD – Lei Geral de Proteção de Dados

Quem deve seguir a lei?

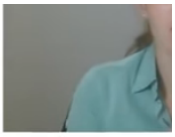
- Empresas que precisam manter um BD
  - Funcionários e Usuários

Devem garantir ao titular sigilo das informações

▼ API e Padrão Rest

DIGITAL INNOVATION ONE

API JAVA

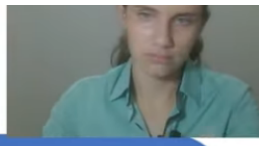


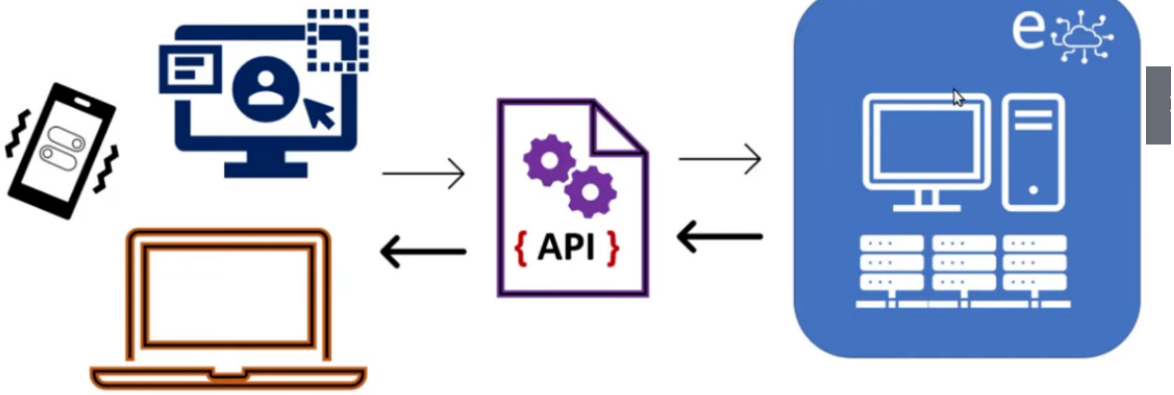
O que é
Application Programming Interface?

- Coleção de métodos disponibilizados por um serviço para interação indireta

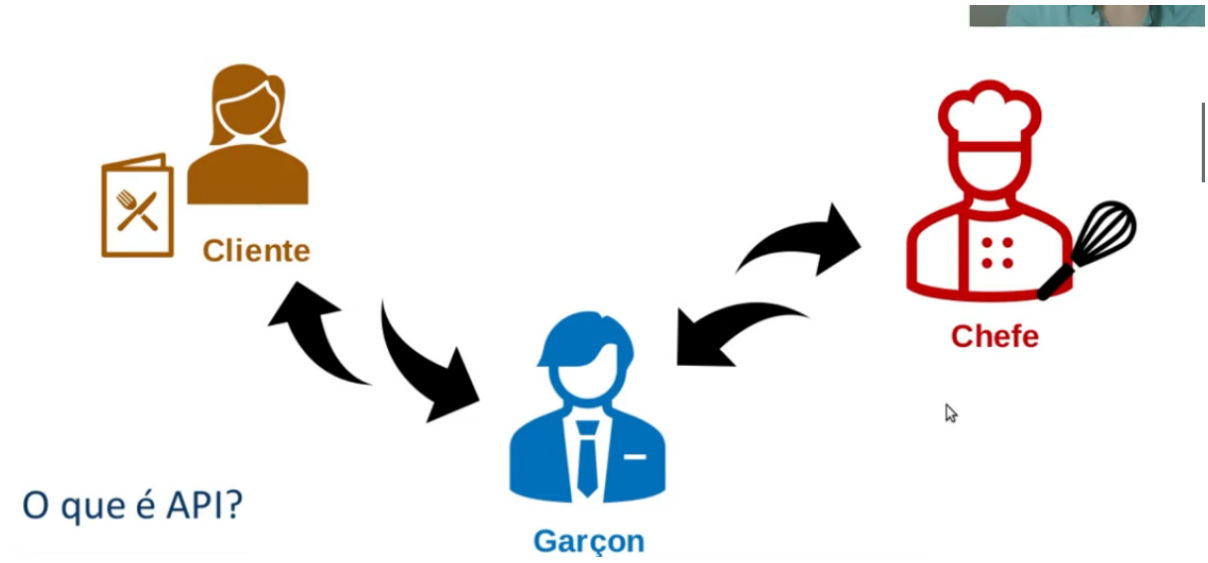
DIGITAL INNOVATION ONE

API JAVA





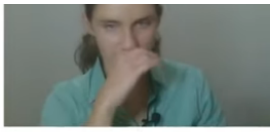
O que é API?

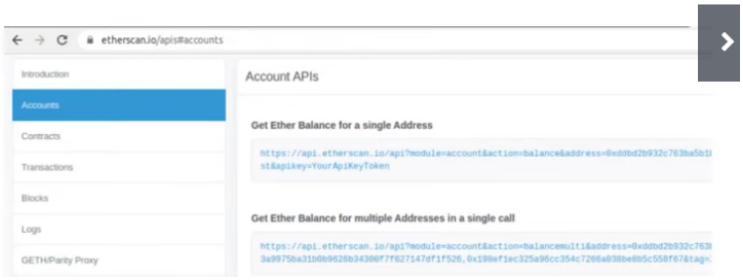

Cliente
Chefe
Garçon
O que é API?


DIGITAL INNOVATION ONE
API JAVA
Google Play
{ API }
"Google Maps" Would Like to Use Your Current Location
Don't Allow
OK
Allow Android Permissions Example to access your location?
Never ask again
DENY
ALLOW
iOS
O que é API?

DIGITAL INNOVATION ONE

API JAVA





Programação

- Rotinas e padrões
- Acesso:
  sistema ou plataforma web

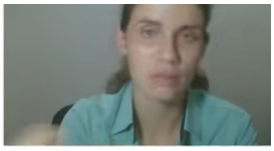
Motivação

- Utilização dos serviços da empresa

Ex: maps

DIGITAL INNOVATION ONE

API JAVA



O que é API em Java?

java.io

- Recursos disponíveis em uma API padrão Java
- Programas de suporte
- Possui partes funcionais chamadas pacotes (o que nós denominamos bibliotecas)

java.applet

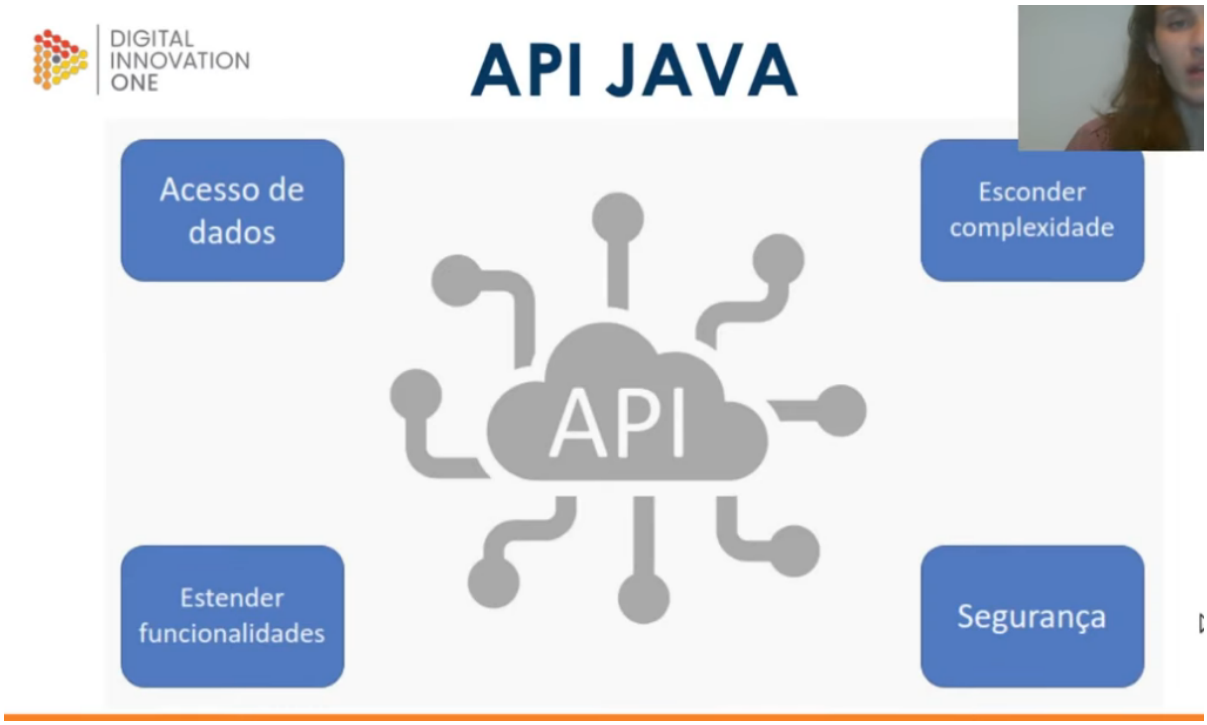
java.security

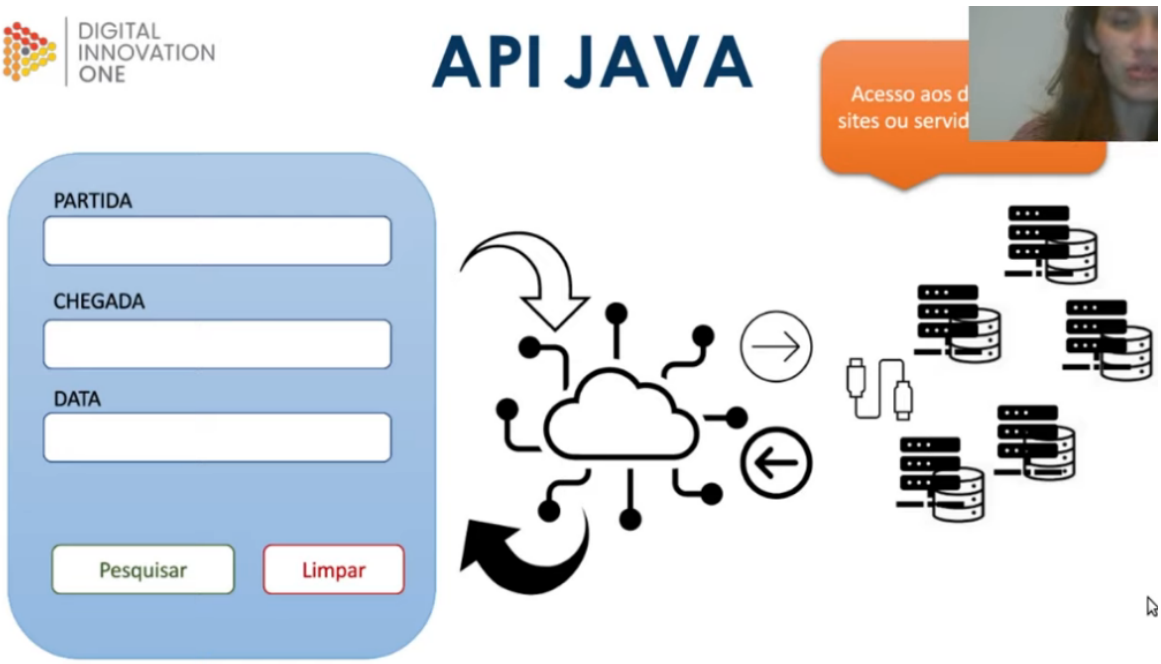
java.math

Aula 2| Etapa 2:

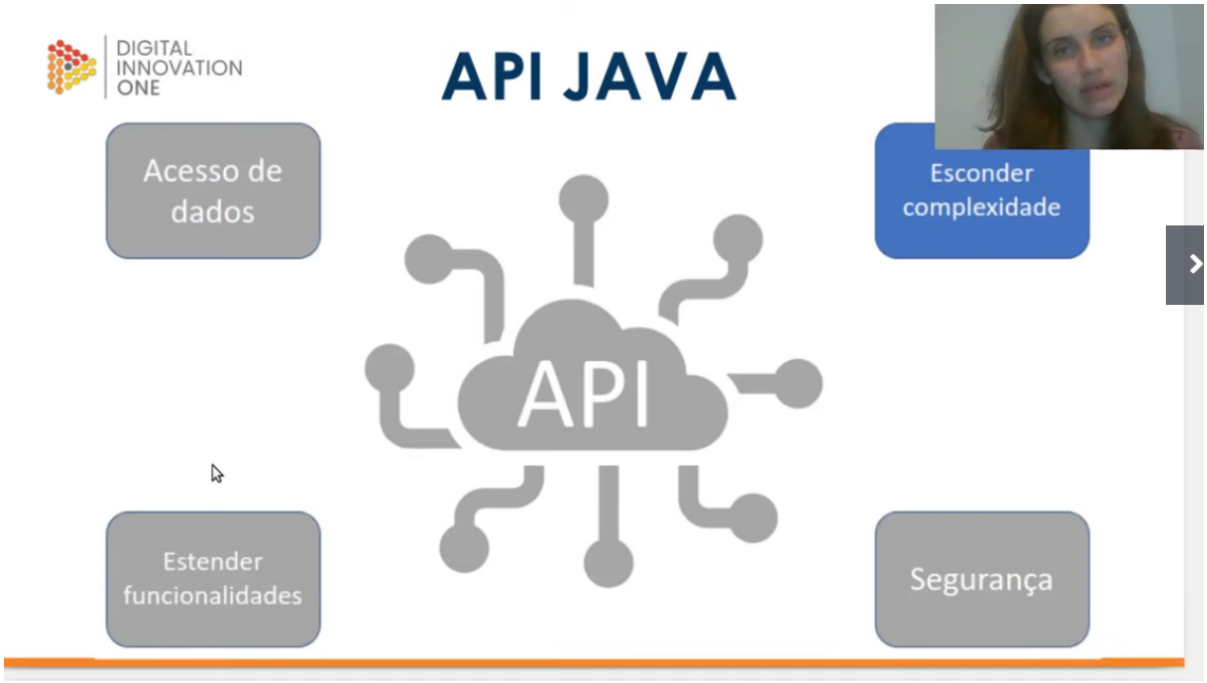
Propriedades da API

Introdução ao HTTP




DIGITAL INNOVATION ONE
API JAVA
Acesso aos d
sites ou servid
PARTIDA
CHEGADA
DATA
Pesquisar
Limpar


DIGITAL INNOVATION ONE
API JAVA
Programação
• Acesso à dados:
Arquivos,
Banco de dados, ...
SQL
C
Programming Language
• Executar operações complexas
Rotinas e funções


DIGITAL INNOVATION ONE
API JAVA
Acesso de dados
Esconder complexidade
API
Estender funcionalidades
Segurança


Programação
Acesso à dados:
Arquivos,
Banco de dados, ...
SQL
C
Programming Language
Executar operações complexas
Rotinas e funções

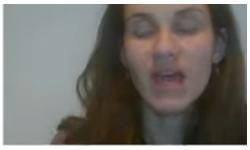
Transparente:

- Comandos hardware
- Funções específicas do sistema

- Intermédio de execução
- Dev:
  Foca na funcinalidade do app

DIGITAL INNOVATION ONE

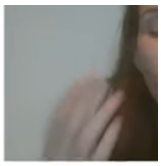
API JAVA



Comunicação entre software e hardware

DIGITAL INNOVATION ONE

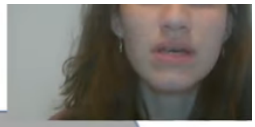
API JAVA



Comunicação entre aplicativos

DIGITAL INNOVATION ONE

API JAVA

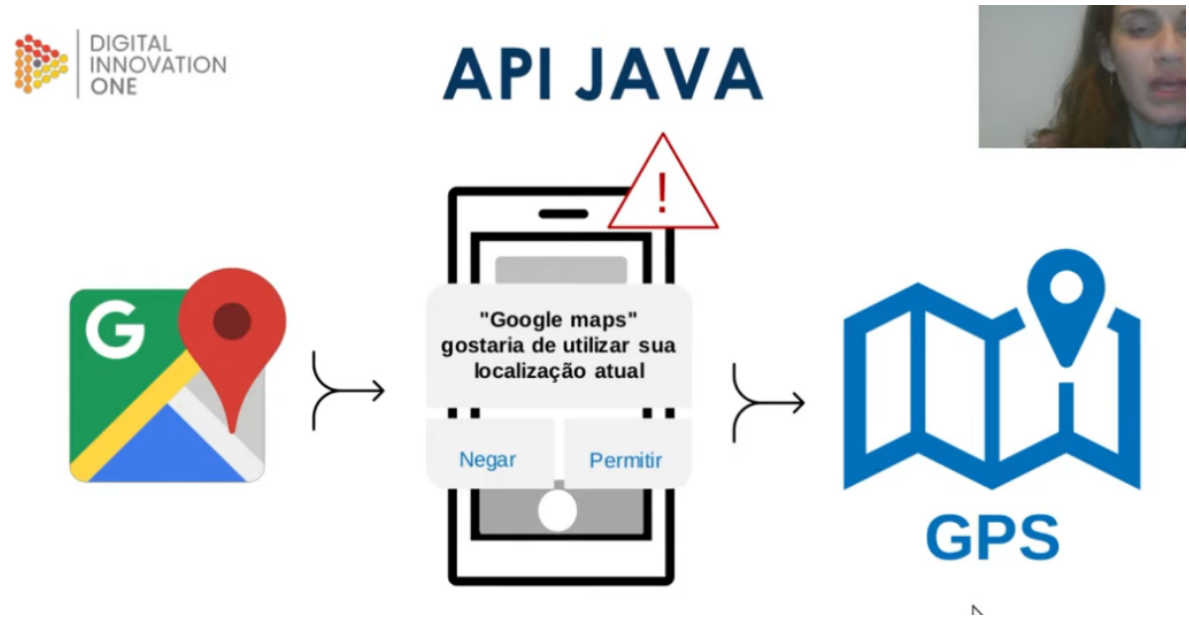


Acesso de dados

Esconder complexidade

Estender funcionalidades

Segurança

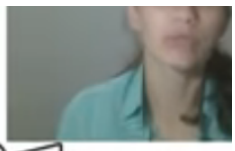

DIGITAL
INNOVATION
ONE
API JAVA
"Google maps"
gostaria de utilizar sua
localização atual
Negar
Permitir
GPS

▼ Padrão REST


Aula 2 | Etapa 3:
Padrão REST
Introdução ao HTTP

DIGITAL INNOVATION ONE

Padrão REST

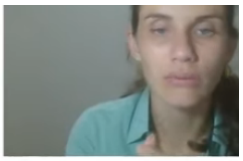


- HTTP:
  Comunicação entre API e Sistema

- Regras:
  Arquitetura REST

DIGITAL INNOVATION ONE

Padrão REST



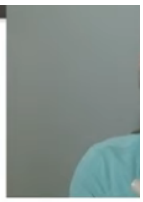
REpresentational State Transfer – REST

- Apresentada por Roy Fielding - 2000
  https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm

- Boas práticas - regras bem definidas
- Comunição entre sistemas
- Padrão de linguagem

DIGITAL INNOVATION ONE

Padrão REST

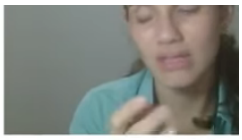


REpresentational State Transfer – REST

- Vantagens da padronização:
  - Maior compreenção
  - Redução do esforço
  - Ganho em agilidade e
  - Eficiência
  - Migração de sistemas

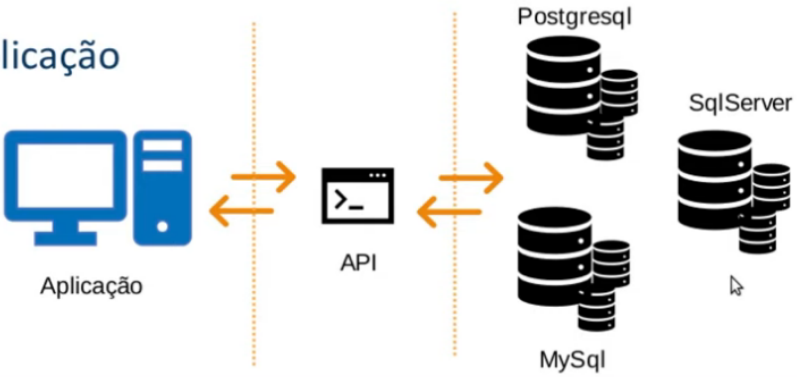
DIGITAL INNOVATION ONE

Padrão REST



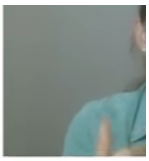
REpresentational State Transfer – REST

- Vantagem - Independência da aplicação



DIGITAL INNOVATION ONE

Padrão REST



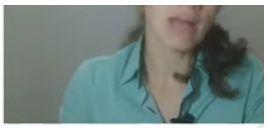
Modelo

- Client-server
- Stateless
- Cache
- Uniform Interface
- Layered System
- Code on Demand (Opcional)

Constrains

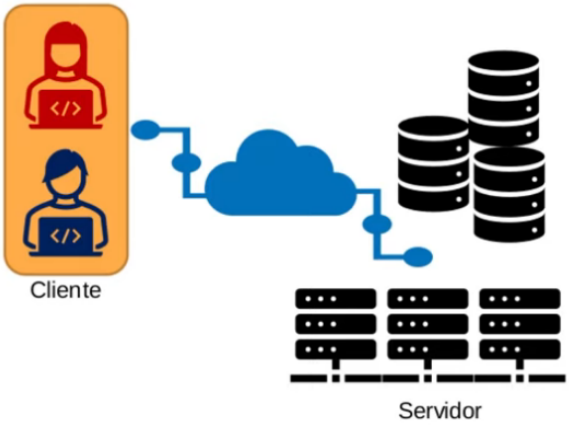
DIGITAL INNOVATION ONE

Padrão REST



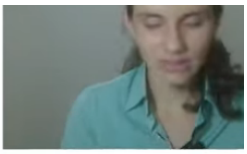
Client-server

- Front x Back
  - Princípio da separação
  - Menos complexidade
  - Organização dos Devs



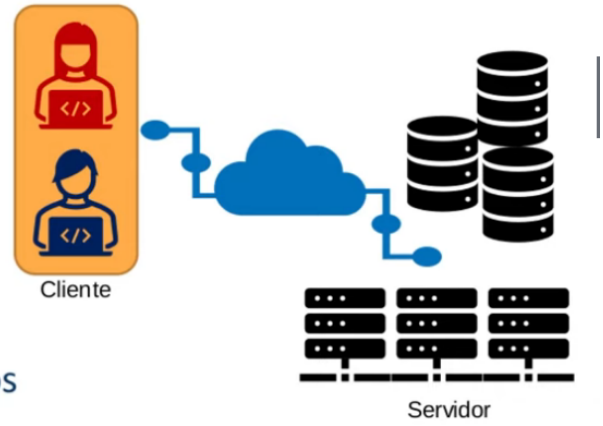
DIGITAL INNOVATION ONE

Padrão REST



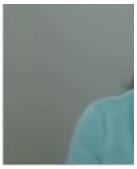
Client-server

- Portabilidade Interface do usuário
- Aumento da escalabilidade
- Simplifica os componentes dos servidores



DIGITAL INNOVATION ONE

Padrão REST



Stateless

- Request
  Fornecer completo entendimento para o servidor

- Visibilidade
- Confiabilidade
- Escalabilidade

Padrão REST

Stateless

- Escolha – Tradeoff
  - Repetição de dados
  - Per-interaction overhead
  - Performance da rede x propriedades do REST

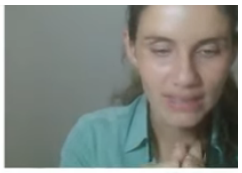
Padrão REST

Cache

- Objetivo
  aumentar eficiência da rede
- Label Requests
  Cacheable or non-cacheable
- Reduz latência e interação

DIGITAL INNOVATION ONE

Padrão REST



Uniform Interface

- Diferencial
  Ênfase de uma interface uniforme entre componentes
- Generalidade - princípio de eng. de software

Uniform Interface

- Arquitetura de multiplas restrições
  - Identificação de recursos
  - Manipulação de recursos
    representações - ex: verbos HTTP
  - Auto-descrição
    Processamento da informação

DIGITAL INNOVATION ONE

Padrão REST

Uniform Interface

- HATEOAS

  Hypermedia As The Engine Of Application State.

Ex: métodos HTTPS

GET http://api.project.net/users/1

DIGITAL INNOVATION ONE

Padrão REST

Elementos de dados

- Aspecto chave

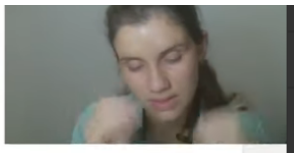
  - Estado e elementos de dados

Table 5-1: REST Data Elements

Data Element	Modern Web Examples
resource	the intended conceptual target of a hypertext reference
resource identifier	URL, URN
representation	HTML document, JPEG image
representation metadata	media type, last-modified time
resource metadata	source link, alternates, vary
control data	if-modified-since, cache-control

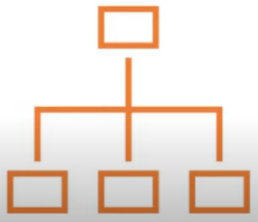
DIGITAL INNOVATION ONE

Padrão REST



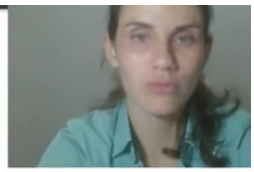
Layered System

- Composição hierárquiva de camadas
  - Encapsulamento
  - Proteção de dados
- Latência e overhead de dados
  - Compensado pelo uso de cache



DIGITAL INNOVATION ONE

Padrão REST



Code on Demand

Optional constraint

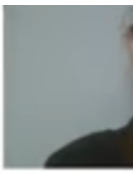
- Funcionalidade de clientes -> applets ou scripts
- Simplificação
  - redução de features à serem pré-implementadas

Sistema extensível x diminuição de visibilidade

▼ API HTTP JAVA

DIGITAL INNOVATION ONE

API JAVA



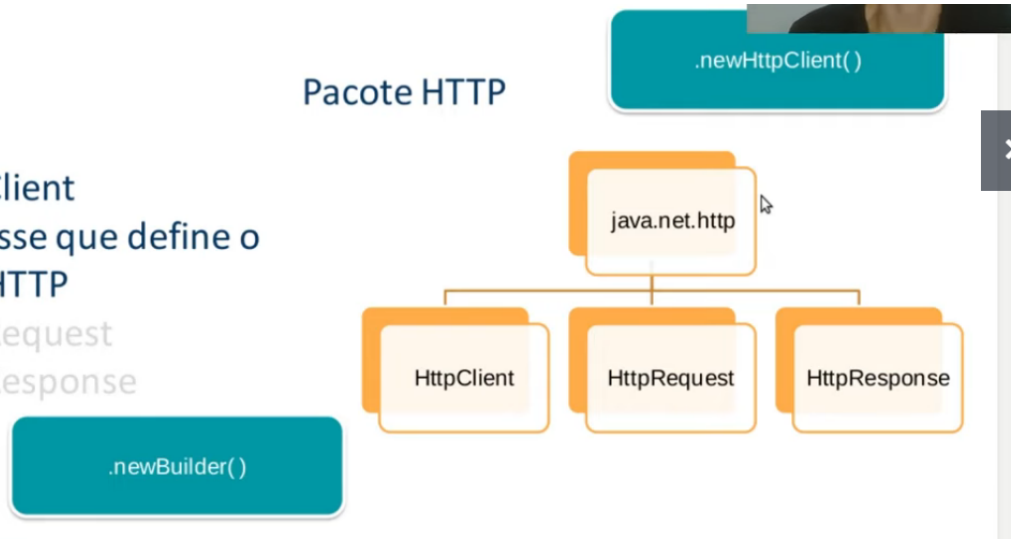
API – HTTP

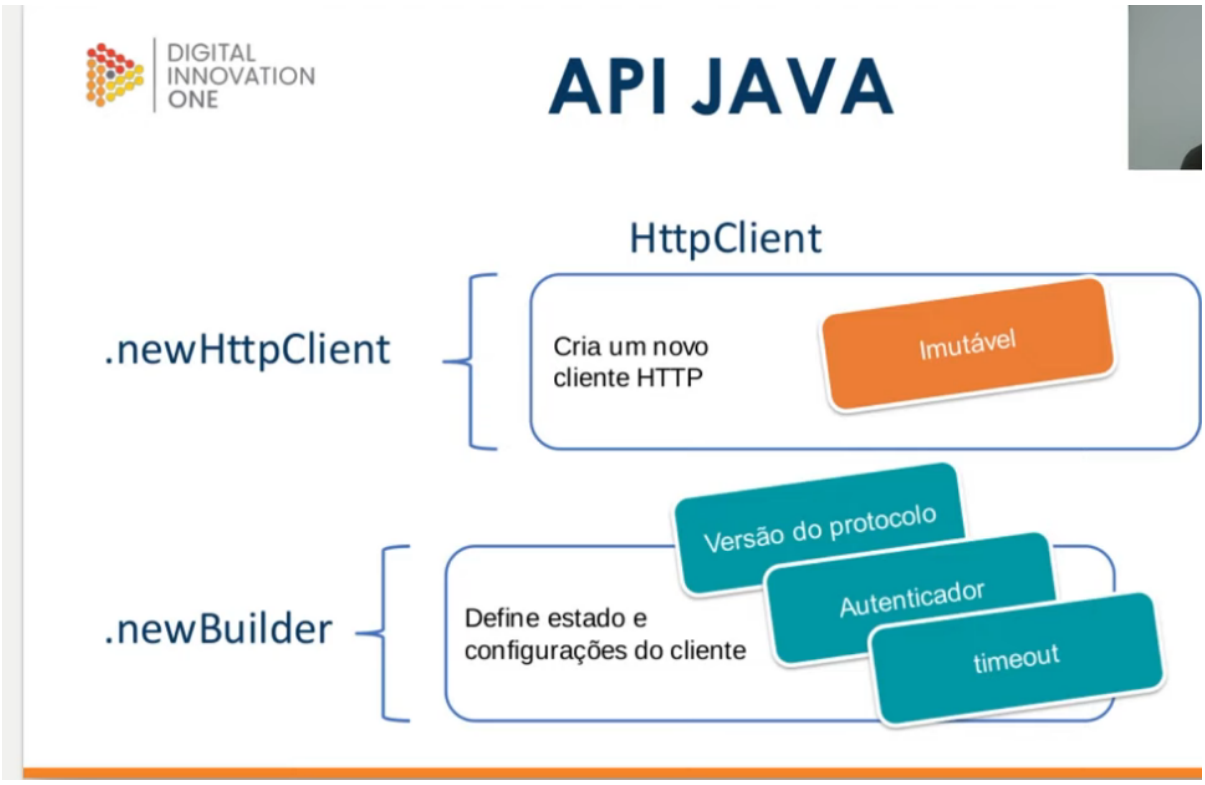
- Adicionado à versão 11 do Java
- Suporte à HTTP/1.1 e HTTP/2.0
- Requisições
  Síncrona e assíncrona

OpenJDK

Pacote HTTP

- HttpClient
  Classe que define o cliente HTTP
- HttpRequest
- HttpResponse

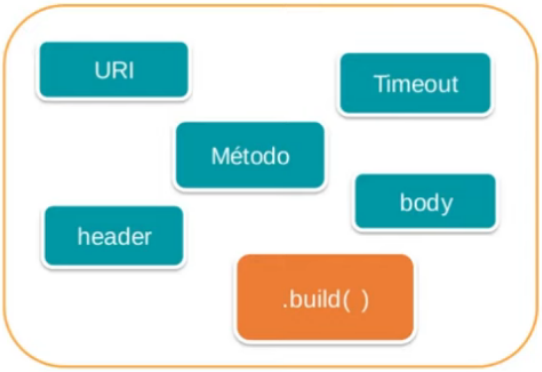


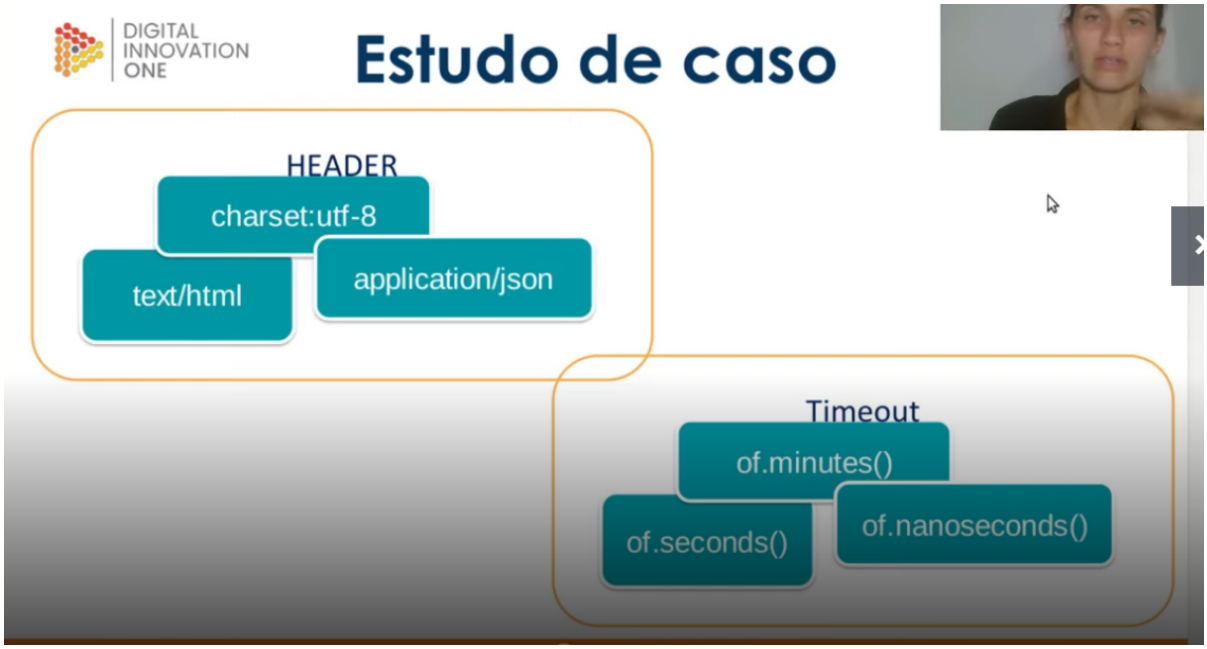

DIGITAL INNOVATION ONE
API JAVA
HttpClient
.newHttpClient
Cria um novo cliente HTTP
Imutável
.newBuilder
Define estado e configurações do cliente
Versão do protocolo
Autenticador
timeout


HttpClient
.newHttpClient
HttpClient client = HttpClient.newHttpClient();
.newBuilder
HttpClient builderClient= HttpClient.newBuilder()
.version(HttpClient.Version.HTTP_1_1)
.connectTimeout(Duration.ofMinutes(2))
.build();

HttpRequest

- Criado a partir do builder
- Métodos que definem os parâmetros de configuração





HttpRequest

- Criado a partir do builder
- Métodos que definem os parâmetros de configuração

```
HttpRequest request = HttpRequest.newBuilder()
        .GET()
        .header( name: "accept", value: "application/json")
        .timeout(Duration.ofMinutes(2))
        .uri(URI.create(URI_JSON_POSTS_URL))
        .build();
```

```
public static final String URI_JSON_POSTS_URL = "https://jsonplaceholder.typicode.com/posts";
```

- Criado a partir do builder
- Métodos que definem os parâmetros de configuração

```
.GET()
.header( name: "accept", value: "application/json")
.timeout(Duration.ofMinutes(2))
```

```
import java.nio.file.Paths;
import java.time.Duration;
```

```
public static final String URI_JSON_POSTS_URL = "https://jsonplaceholder.typicode.com/posts";
```

HttpResponse

- Classe criada indiretamente

HttpRequest

- Retornado como resultado do envio de uma requisição do cliente

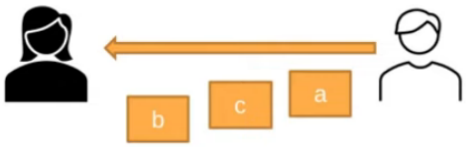
```
interface HttpResponse<T>{
    Int statuscode;
    HttpHeaders heads( );
    T boby ( );
    Version version( );
    HttpRequest request();
    . . .
}
```

HttpResponse

- Síncrona e Assíncrona

BodyHandelers.ofString( )

Trata os bytes da resposta HTTP

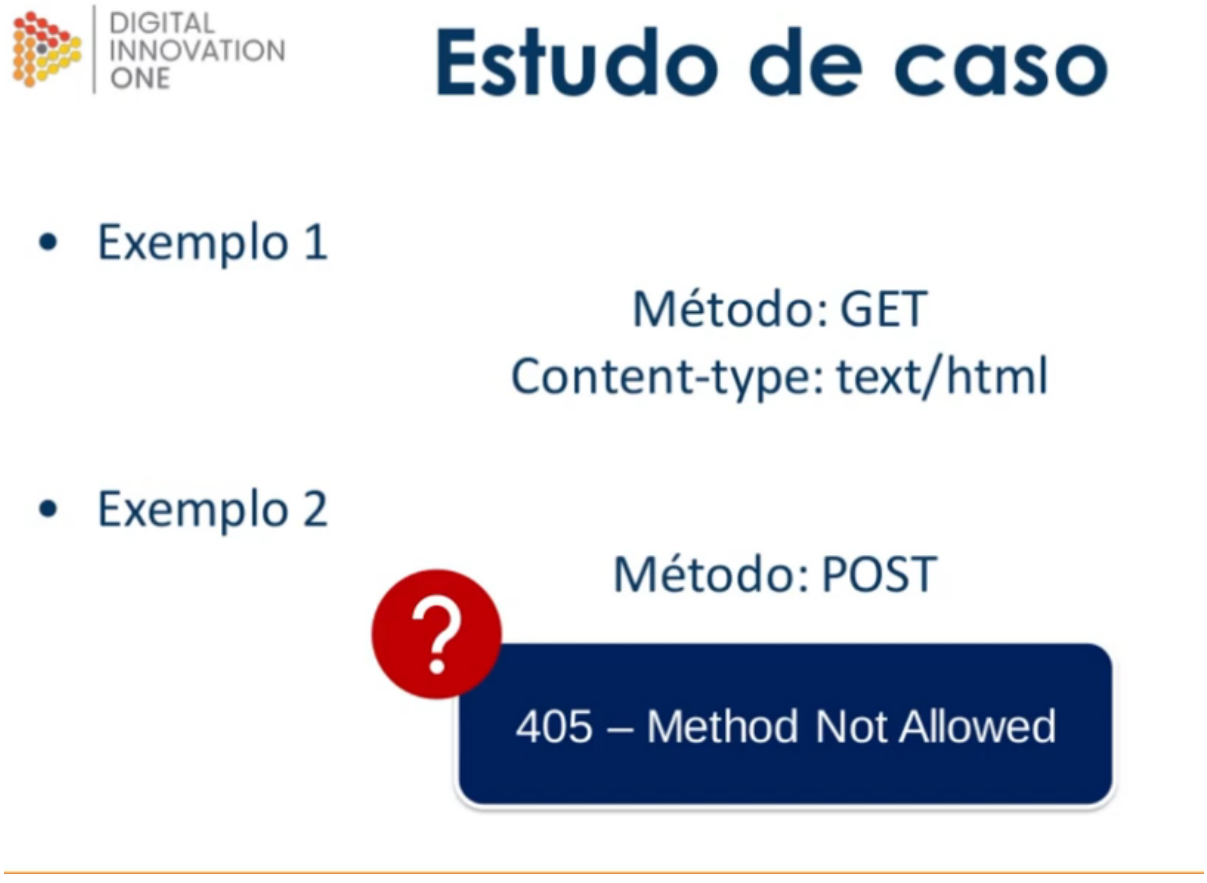



Aula 2| Etapa 5:
Estudo de caso
Introdução ao HTTP

▼ Exercícios


Aula 2| Etapa 4:
API HTTP - JAVA
Introdução ao HTTP


DIGITAL INNOVATION ONE
Estudo de caso
IJ
IDE
Intellig
POST ( )
GET ( )
Requisições HTTP
URIs
https://jsonplaceholder.typicode.com/posts
http://httpbin.org/forms/post
http://httpbin.org/get


DIGITAL INNOVATION ONE
Estudo de caso
Exemplo 1
Método: GET
Content-type: text/html
Exemplo 2
Método: POST
?
405 – Method Not Allowed