Web Programming

HTTP

Eva Oliveira
Licenciatura em Engenharia de Sistemas Informáticos, 2024

Who invented the Internet?

Any guess?

1958	Americans founded the Defence Advanced Research Project Agency (DARPA) and created the first network ARPANET **PAUL BARAN (1926–2011), LAWRENCE ROBERTS (1937–2018)**
1960	The package switching concept emerged in NPL (National Physical Laboratory) in England
1962	Distributed network concept emerged in CYCLADES in France
1982	TCP/IP- a standard that guarantees compatibility between networks **PAUL MOCKAPETRIS (1948–) AND JON POSTEL (1943–98)**
1989	Tim Berners-Lee proposes the hypertex at CERN (European Particle Physics Laboratory) **MARC ANDREESSEN (1971–) the inventor of the moisaic**
1990	The internet born as Tim wrote the first web client and server in 1990, the first browser and fundamental protocols and algorithms allowing web scale
1992	The World Wide Web (WWW) (HTTP/HTML)

More history, follow the link

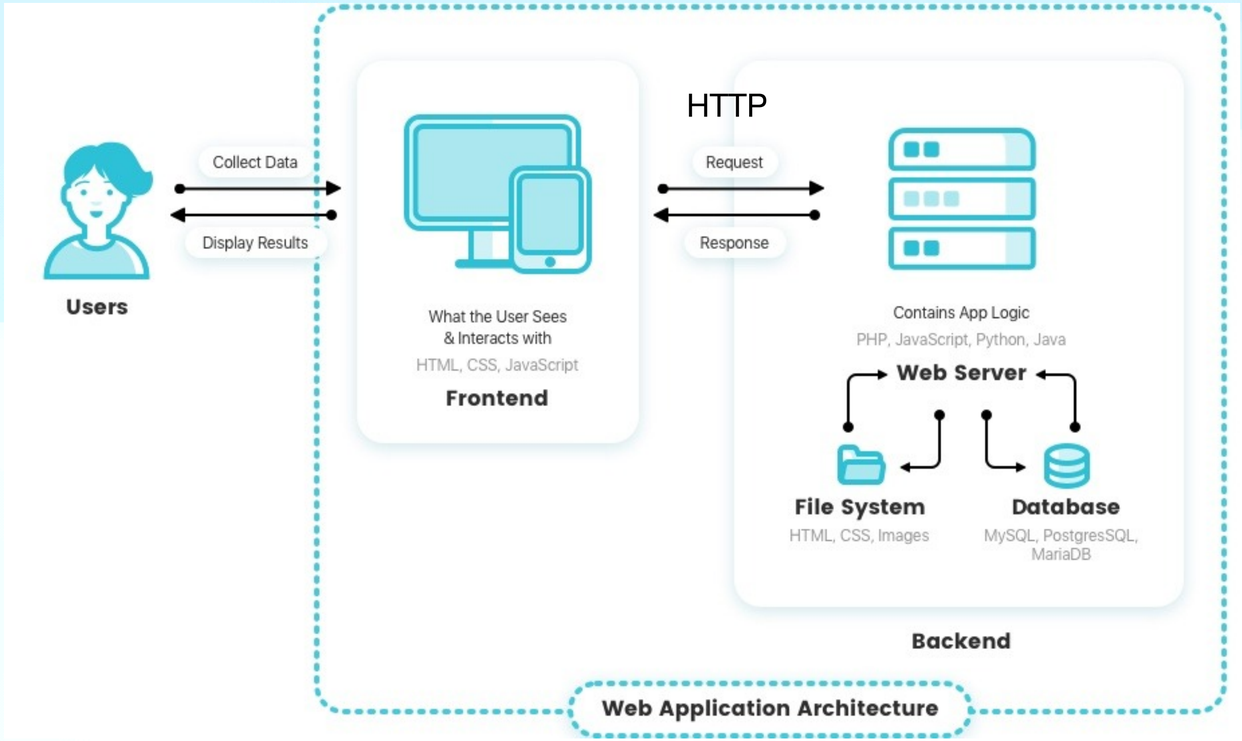
https://www.scienceandmediamuseum.org.uk/objects-and-stories/short-history-internet

Tim Berners-Lee

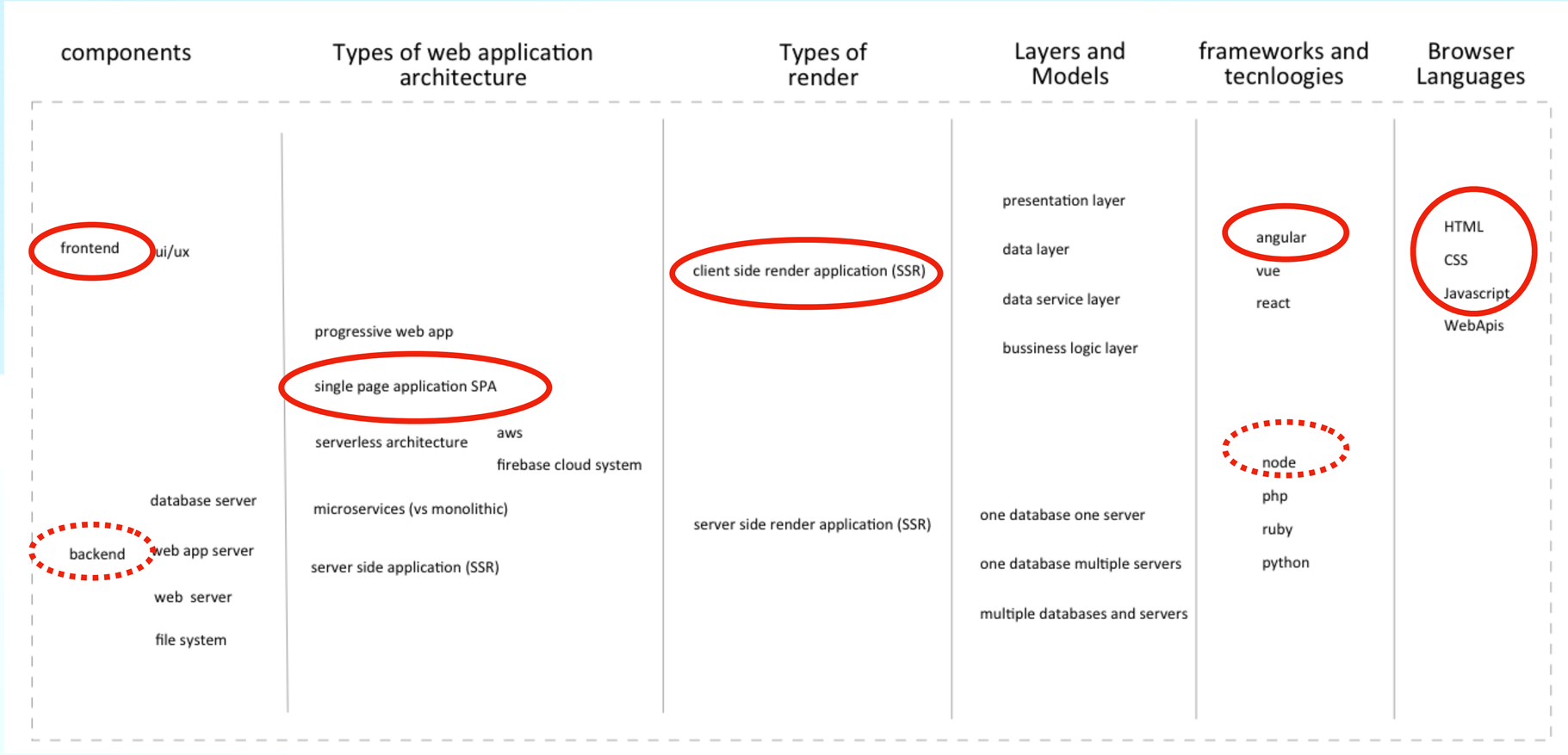
Hypertext inventor

- Oxford Professor
- Invented the World Wide Web in 1989
- He is the Director of the World Wide Web Consortium (W3C), a Web standards organisation founded in 1994 which develops interoperable technologies (specifications, guidelines, software, and tools)
- W3C standards define an Open Web Platform for application development.
- References
  - https://www.w3.org/People/Berners-Lee/
  - https://www.w3.org

Web Application Architecture



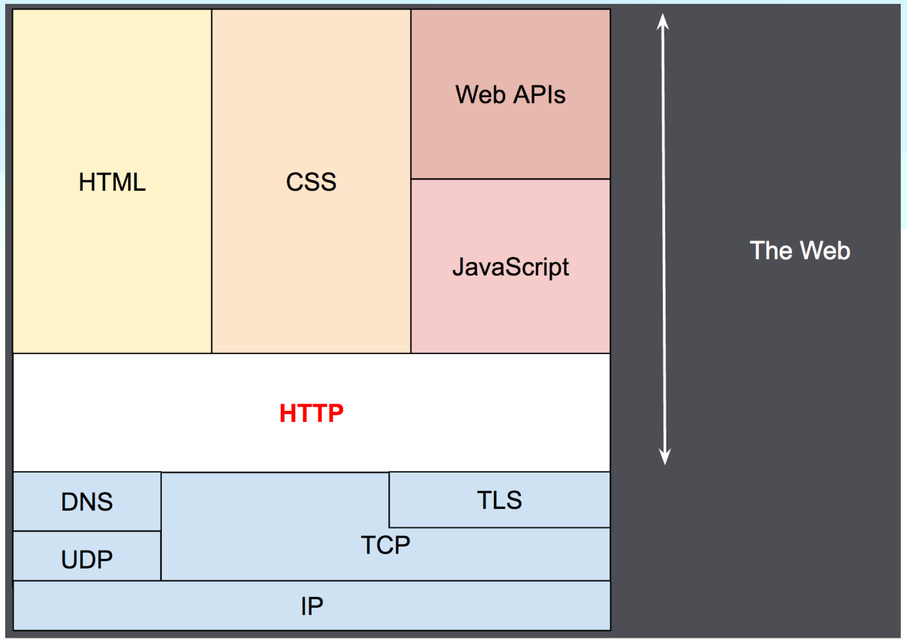
Web Concepts and Technologies



Web development

Layers

- The web is one application running on top of the internet, allowing us to get web pages.
- **HTTP**, or Hypertext Transfer Protocol, governs how web browsers and web servers communicate within TCP/IP packets.
- HTTPS is HTTP with an encrypted layer



HTTP

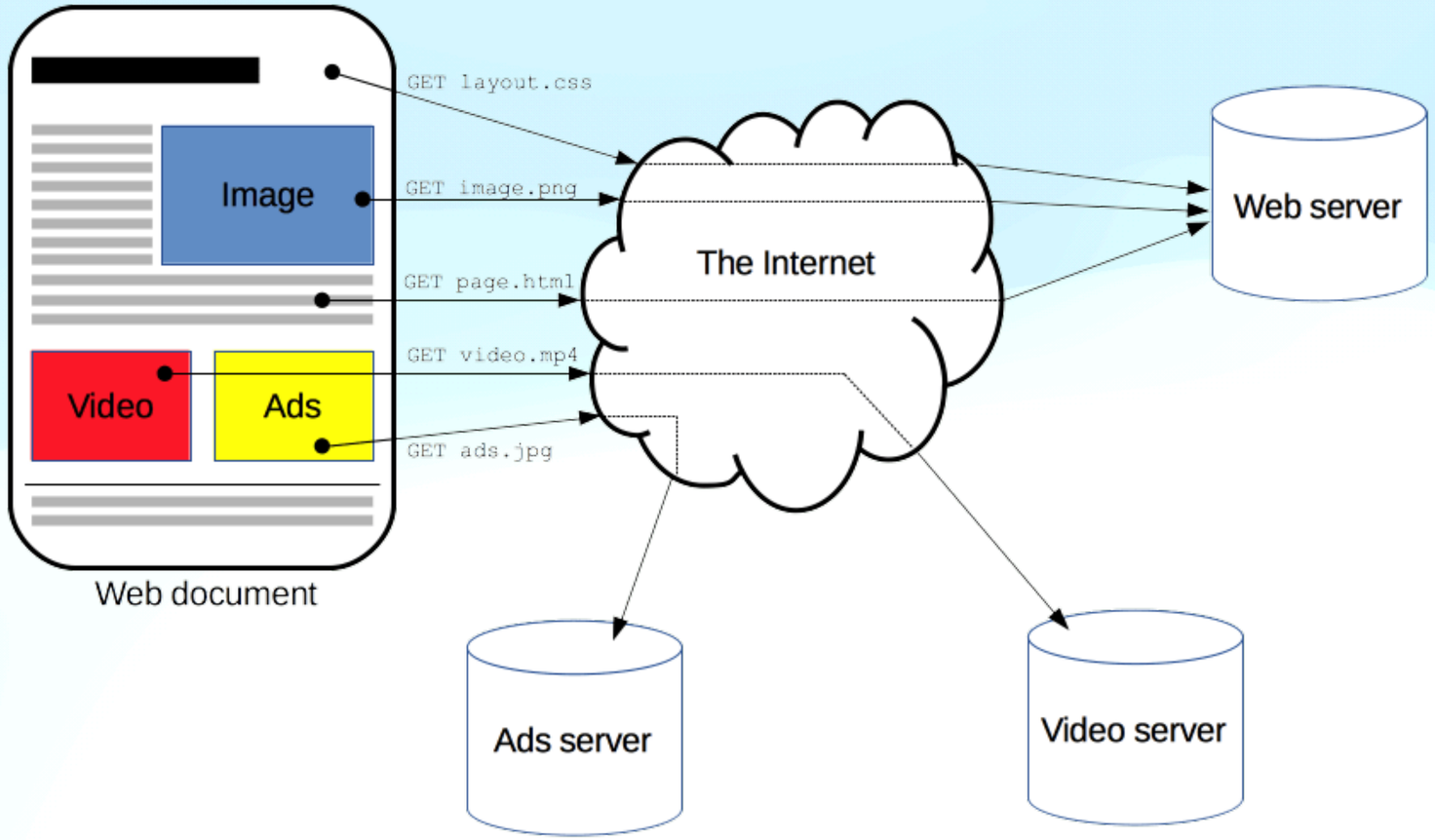
The backbone of Web Communication

- Stands for Hypertext Transfer Protocol
- Governs the communication between web browsers and servers
- Uses request and response messages

https://developer.mozilla.org/en-US/docs/Web/HTTP

HTTP

Overview



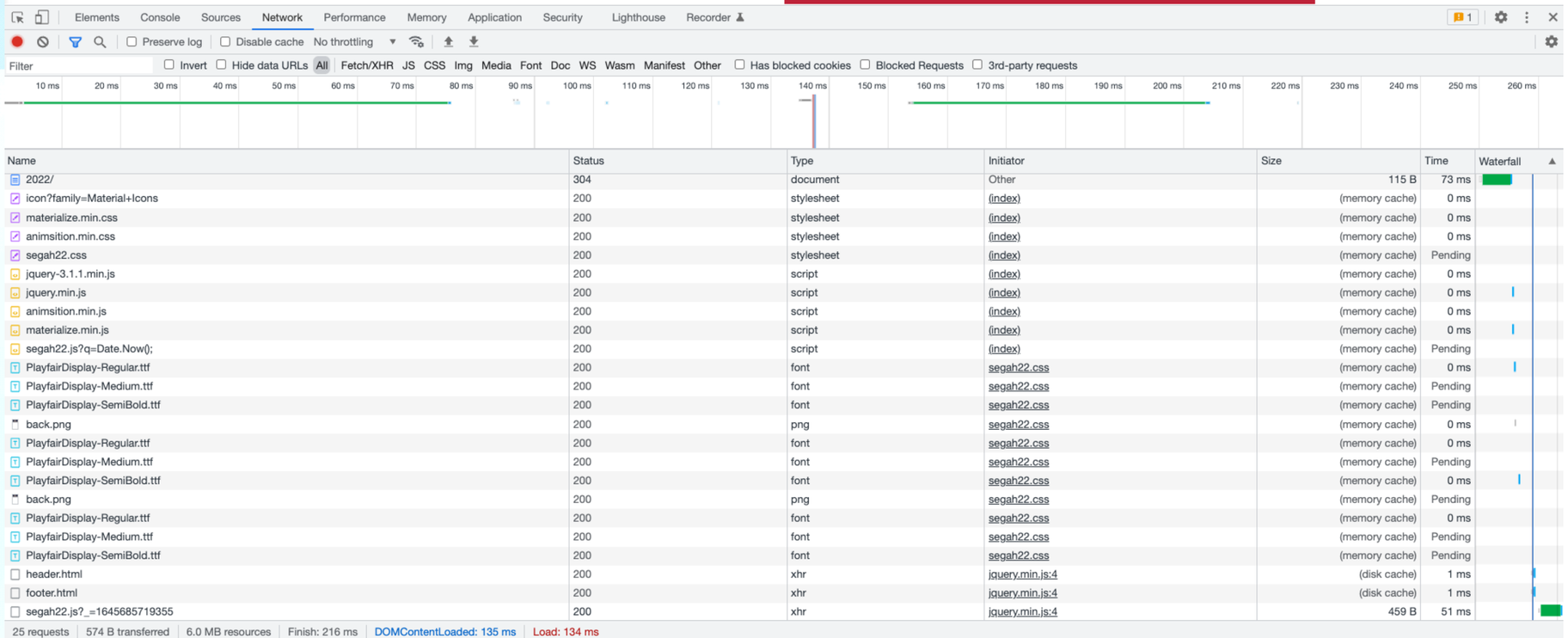
It is with great pleasure that we invite you to participate in the 10th International Conference on Serious Games and Applications for Health 2022, to be held in Sydney, Australia on 10th - 12 of August at University of Technology Sydney (UTS).

News

August 2022 - Website launched

Important Dates

Conference:	**August 10-12, 2022**
Full Paper Submission:	**March 25, 2022**



HTTP

Evolution

HTTP 1.1

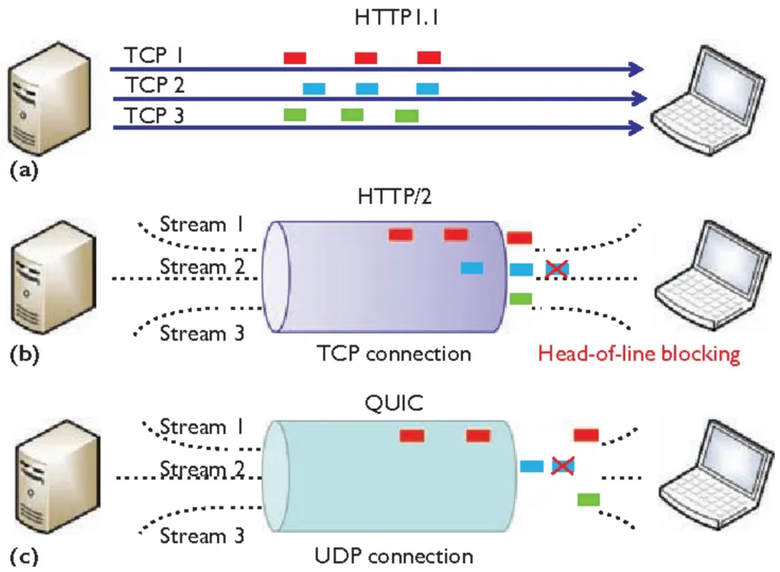
- Uncompressed headers
- One file at a time
- No default encryption
- **But allow https**

HTTP 2

- Developed for performance
- Uses encryption algorithms
- Multiplex (multifiles in the same request)

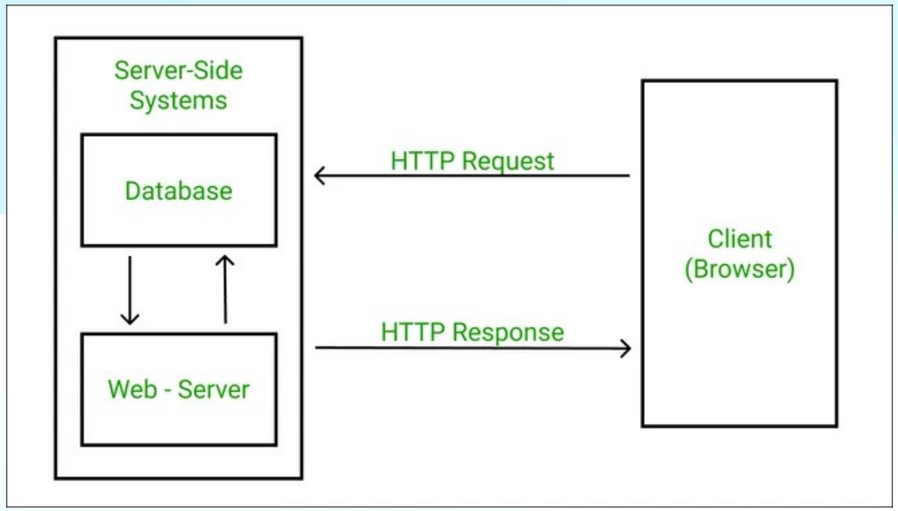
HTTP 3.0

- The latest version of the Hypertext Transfer Protocol
- Built on the QUIC transport protocol
- Aims to overcome the limitations of TCP



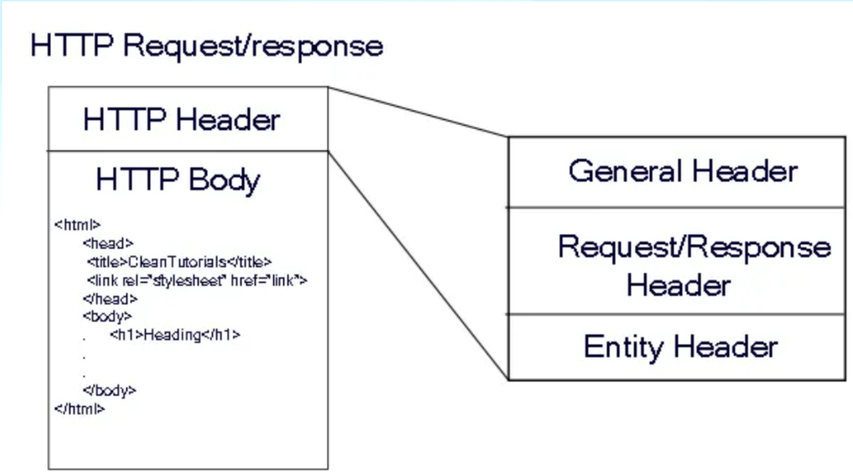
HTTP

Flow



HTTP

Architecture



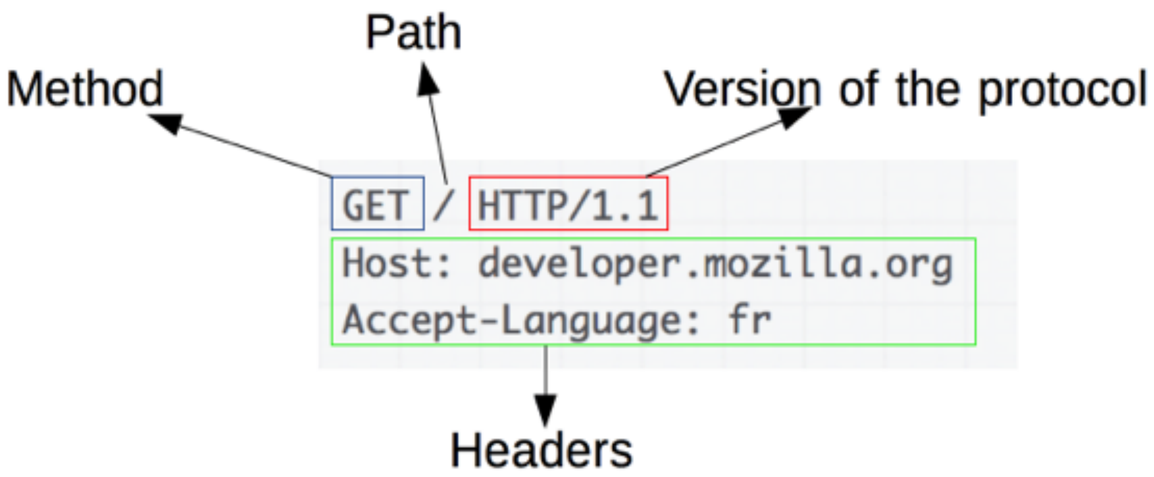
https://developer.mozilla.org

HTTP

Messages

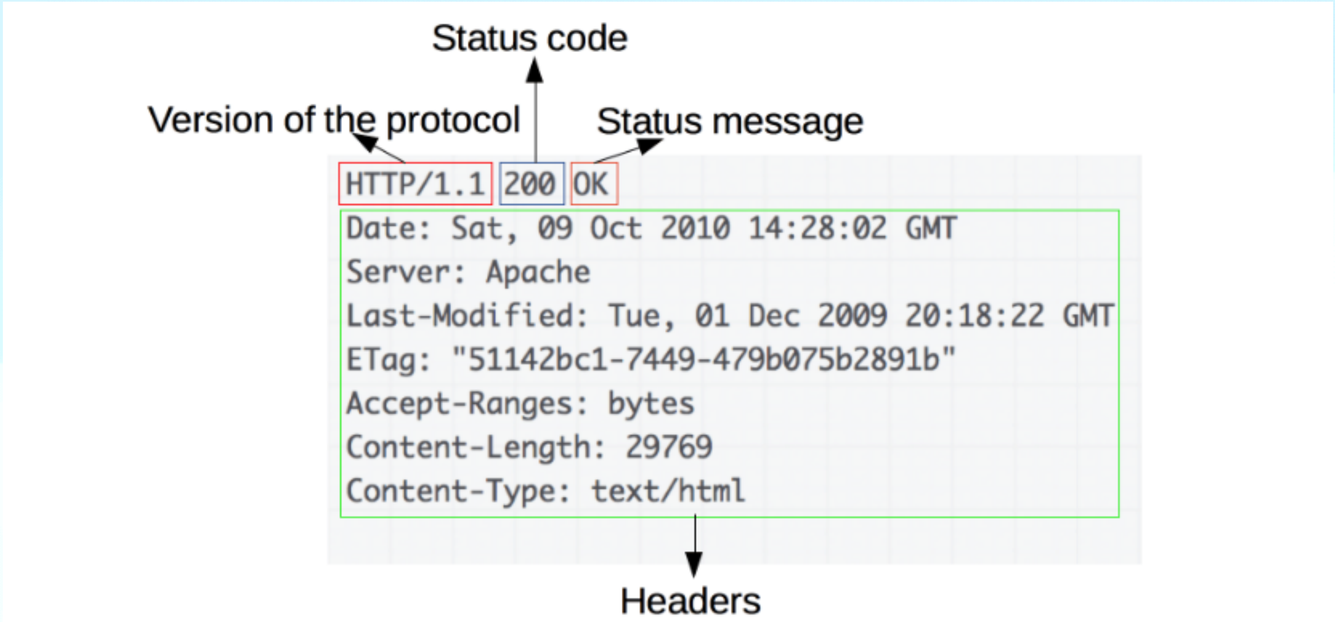
Requests

An example HTTP request:



HTTP

Messages



HTTP

Status

HTTP response status codes indicate whether a specific HTTP request has been successfully completed. Responses are grouped in five classes:

1. Informational responses (`100–199`)
2. Successful responses (`200–299`)
3. Redirection messages (`300–399`)
4. Client error responses (`400–499`)
5. Server error responses (`500–599`)

The below status codes are defined by section 10 of RFC 2616. You can find an updated specification in RFC 7231.

- `200 OK`
- `301 Moved Permanently`
- `304 Not Modified`
  - This allows the browser to use its cache, or local copy, of some resource like an image, instead of having the server send it back again.
- `307 Temporary Redirect`
- `401 Unauthorized`
- `403 Forbidden`
- `404 Not Found`
- `418 I'm a Teapot`
- `500 Internal Server Error`
  `Buggy code on a server might result in this status code.`
  `503 Service Unavailable`
- ...

HTTP

Main characteristics

- How we communicate the resources we need to access from the Internet
  - It uses plain human language (GET, POST, PUT, DELETE, CONNECT, HEAD)
  - Statless : every request is unique and independent
  - Allow session creation with the use of cookies
  - Headers : meta information about requests and responses
  - Request / Responses pair

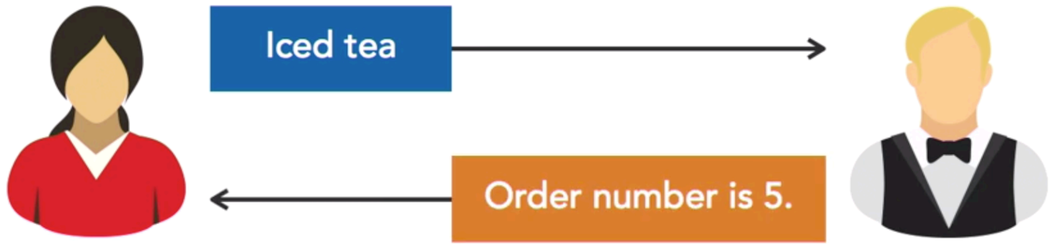
REST and HTTP

- REST is a style of operating on resources over a network
- Everything is a resource(text, video, audio, json, etc)
- HTTP is protocol that have a lot in common with the REST style
  - Both use the same methods for GET, PUT , POST, DELETE
  - When using REST style over HTTP, we are dealing with a RESTFull API

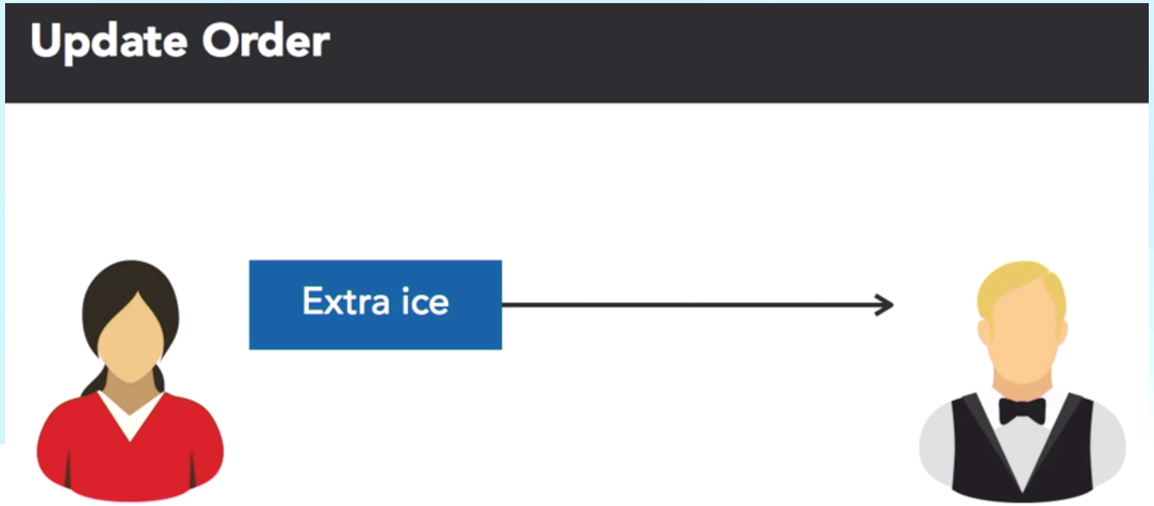
REST API session

POST to create a new item

Request Response



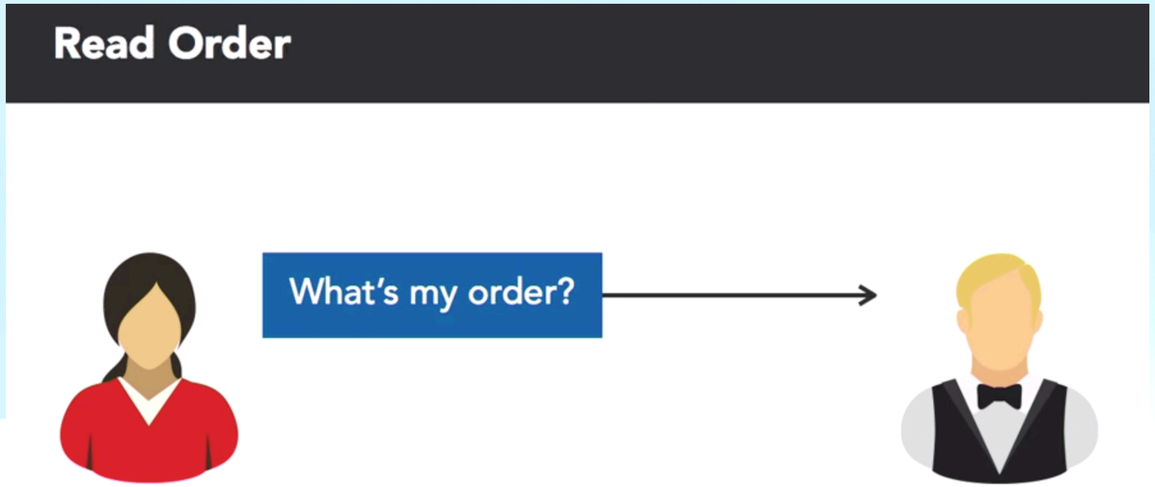
REST API session



PUT to update the existing resource

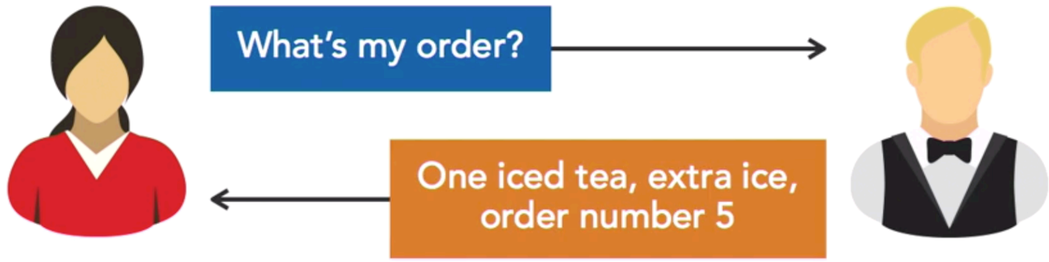
Request Response

REST API session

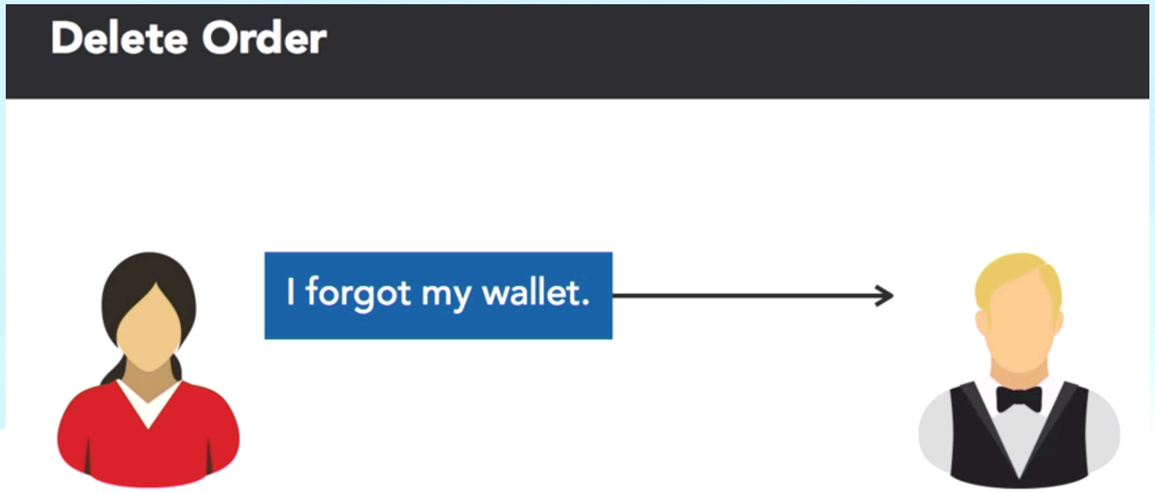


GET to read a resource

Request Response

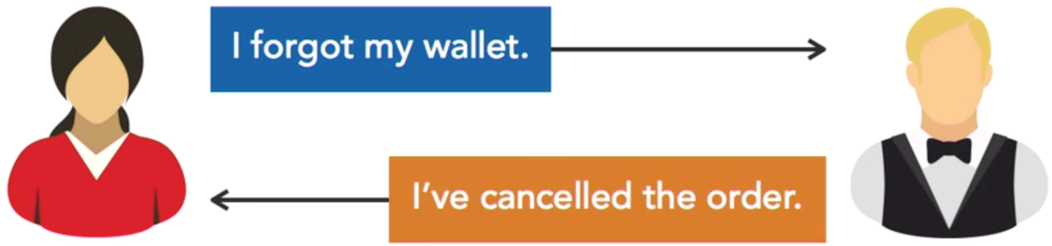


REST API session



DELETE request

Request response



In more detail

Create Order

Request	Response
Method: POST	Status code: 201 Created
Address: /api/v1/orders	Body: { "id":5, "item":"iced tea" }
Body: {"item":"iced tea"}	

In more detail

Update Order

Request	Response
Method: PUT	Status code: 200 OK
Address: /api/v1/orders/5	Body: { “id”:5, “item”:”iced tea”, “option”:”extra ice” }
Body: { “id”:5, “item”:”iced tea”, “option”:”extra ice” }	

In more detail

Get Order

Request	Response
Method: GET	Status code: 200 OK
Address: /api/v1/orders/5	Body: { “id”:5, “item”:”iced tea”, “option”:”extra ice” }

In more detail

Delete Order

Request	Response
Method: DELETE	Status code: 204 No content
Address: /api/v1/orders/5	