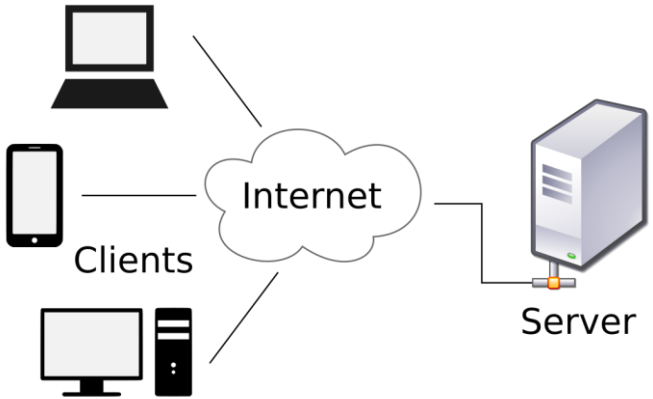
Auth

Aplicações Web

Instituto Politécnico de Viana do Castelo

Programming / **web** development is divided into **two** major **domains**:

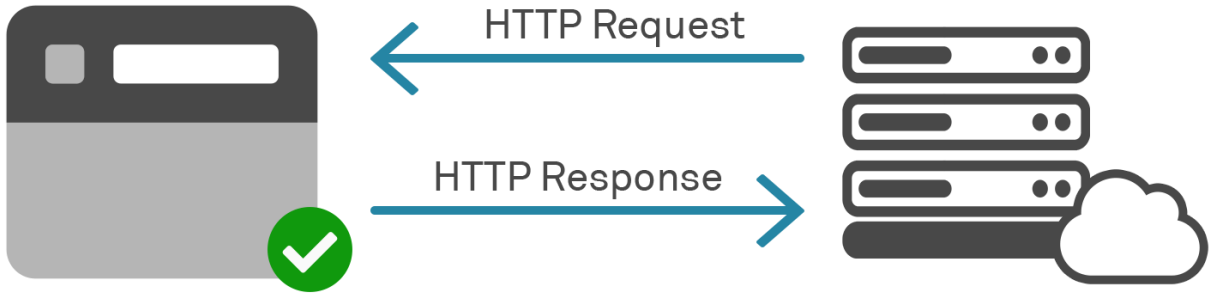
1. Development on the **client** side (front end);
2. **Server-side** development (back end).



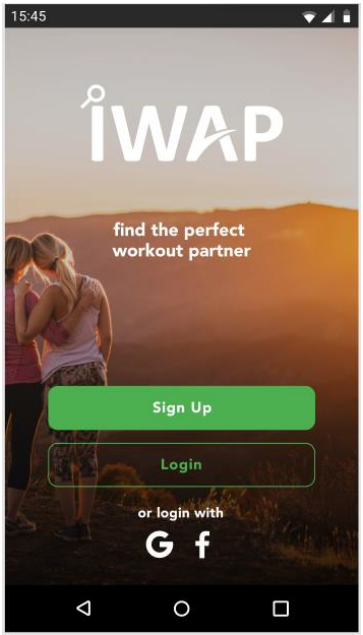
Rest API Model

An API, or **application programming interface**, is a set of rules that define how applications or devices can connect to and communicate with each other.

https://www.ibm.com/topics/rest-apis



https://appcheck-ng.com/http-verbs-security-risks


15:45
iWAP
find the perfect workout partner
Sign Up
Login
or login with


Sign in with Facebook
Login with Twitter
Sign in with Google
Sign in with Microsoft
Log in with Linkedin

Authentication is the process of verifying a user's identification through the acquisition of credentials and using those credentials to confirm the user's identity. The authorization process begins if the credentials are legitimate. The authorization process always follows the authentication procedure.

You were already aware of the authentication process because we all do it daily, whether at work (logging into your computer) or at home (logging into a website). Yet, the truth is that most "things" connected to the Internet require you to prove your identity by providing credentials

https://www.section.io/engineering-education/how-to-build-authentication-api-with-jwt-token-in-nodejs/

Authorization is the process of allowing authenticated users access to resources by determining whether they have system access permissions. By giving or denying specific licenses to an authenticated user, authorization enables you to control access privileges.

So, authorization occurs after the system authenticates your identity, granting you complete access to resources such as information, files, databases, funds, places, and anything else. That said, authorization affects your capacity to access the system and the extent to which you can do so.

https://www.section.io/engineering-education/how-to-build-authentication-api-with-jwt-token-in-nodejs/

JSON Web Tokens (JWT) are an RFC 7519 open industry standard for representing claims between two parties. For example, you can use jwt.io to decode, verify, and produce JWT.

JWT specifies a compact and self-contained method for communicating information as a JSON object between two parties. Because it is signed, this information can be checked and trusted. JWTs can be signed using a secret (using the HMAC algorithm) or an RSA or ECDSA public/private key combination. In a moment, we'll see some examples of how to use them.

https://www.section.io/engineering-education/how-to-build-authentication-api-with-jwt-token-in-nodejs/

JWTs are a singular object of three concatenated strings separated by a . A sample JWT would look something like this: aaaaaaaaaaaaaaa.bbbbbbbbbbbbbbbbbbbbb.ccccccccccccccccccc

The first string (aaa...a) is the header which contains the algorithm used to encrypt it and the type of token which of course is JWT.

The second string (bbb...b) is the payload which contains all the meat. It can contain what you want, but generally you'll include the user id and username etc.

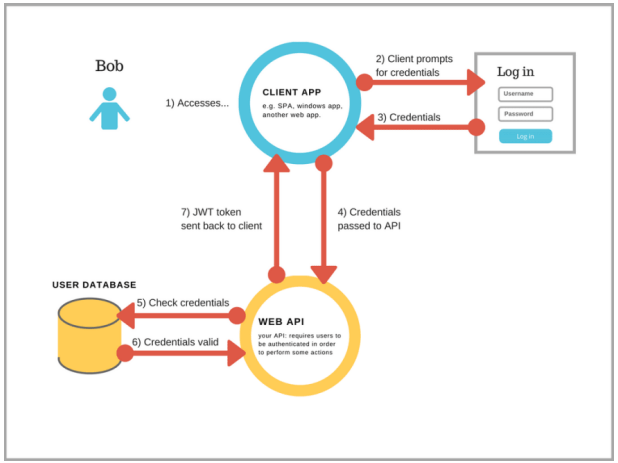
The third string (ccc...c) is the signature which contains a hash (encryption) of the header and the payload. It is hashed with a secret key that is provided by the developer.

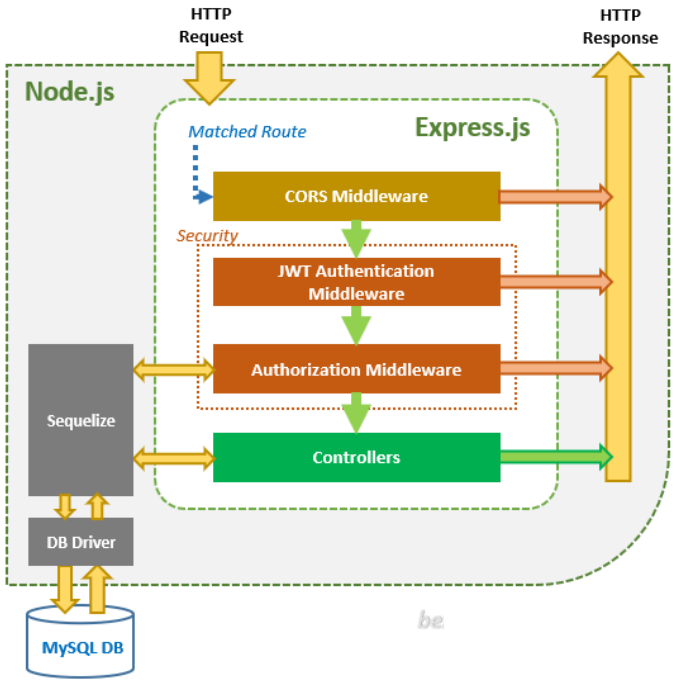
Instituto Politécnico de Viana do Castelo

When a user signs in with a valid username and password (validated by the back end) the backend gives a signed token to the client.

The client will store the token and it will be supplied in the headers to every request that needs authorization.

Effectively, it is an access pass for the user that stores unique encoded identifiers and the unique signature from the backend. It proves the user is who they say they are.




HTTP Request
HTTP Response
Node.js
Express.js
Matched Route
CORS Middleware
Security
JWT Authentication Middleware
Authorization Middleware
Controllers
Sequelize
DB Driver
MySQL DB

o **teu** ● de partida

www.ipvc.pt