Base de Segurança API e WebSite

Client

auth.middleware

All requests

JWT validate

is valid?

yes

log audit data for resource access

no

TokenExpiredError

quota.middleware

rate-limit and max-rate-limit are environment configuration

CACHE

all requests

retrieve how many access this resource had from this IP

checks quota exceeded

increases the response header with:

x-rate-limit
x-rate-limit-remaining
x-rate-limit-time

exceeded

yes

HTTP Status: 429
HTTP Message: `Quota Exceeded`

no

CACHE

Increase access counter for resource

allow the navigation flow

API

login.service

User Authentication

try count and max attempts as environment config

Validate credentials

success

yes

Retrieve additional user profile data and return

DATA

no

max attempts reached?

yes

TooManyLoginAttemptsError

increases invalid attempts counter

no

NotAuthorizedError

CACHE

verify if there are previous invalid attempts for this credential

email and pass

User {
userId: string,
userName: string,
userPhone: string,
userEmail: string,
companyId: string,
companyName: string,
verificado: boolean,
securityPhrase:string,
}

send SMS Token (OTP)

auth user

`{ smsToken, userPhone }`

auth user

Session expires as environment configuration

CACHE

Create and store user session

yes

is token valid?

no

NotAuthorizedError

validate sms token (OTP)

Generate JWT token

auth.controller

Refresh Token

check JWT signature

Extract session ID from JWT body

Try find session

CACHE

Session expires as environment config

session found?

yes

Check if the received token is the last valid session token

no

Token ok?

no

yes

generate new token

HTTP Status: 401
HTTP Message: Impossible to Renew

HTTP Status: 200
Response Body: JWT

Headers:
x-access-token: JWT
x-device-id: DeviceFingerprint

Body:
empty

POST /api/v1/auth/refreshtoken

Send Token

send sms token (OTP)

error?

no

yes

show sms sent successfully

Headers:
x-device-id: DeviceFingerprint

Body:
email: login
password: senha

POST /api/v1/auth/sendtoken

TooManyLoginAttemptsError

NotAuthorizedError

HTTP Status: 429
HTTP Message: Too Many Login Attempts

HTTP Status: 401
HTTP Message: Not Autorized

HTTP Status: 200
Response Body: { userPhone }

Login

login

error?

no

yes

Headers:
x-device-id: DeviceFingerprint
x-device-location: LatLong

Body:
email: login
password: senha
smsToken: token

POST /api/v1/auth/login

TooManyLoginAttemptsError

NotAuthorizedError

HTTP Status: 429
HTTP Message: Too Many Login Attempts

HTTP Status: 401
HTTP Message: Not Autorized

HTTP Status: 200
Response Body: { JWT }

app.component

open site

check if exist token on LocalStore

token expired or not found?

yes

no

redirect to login

enable controls and redirect to target url

Encrypt the information and store in local storage

login.component

login "Email First" + 2fa

check get location permission on browser

Show "How to allow locations" page

no

could get location?

yes

asks for email

asks for password

show error message

Asks for the SMS token (OTP)

show error message

Enable controls and redirect to target URL

Encrypt the data and update local store whit new token

login.service

retrieves workstation fingerprint

call token renew API

return fail

return success

Generate (or retrieve from cache) the workstation fingerprint

call SMS token (OTP) API

return fail

return success

Generate (or retrieve from cache) the workstation fingerprint

get workstation location

call login api

return fail

return success

auth-guard

token expired

token expired or not found?

token ok

allow navigation

verify if there is a token on local storage

app-routing

navigate

do security verification

redirect to target url

http.interceptor

redirect to login

Other

Show generic error message

429

show quota exceeded exception

intercept all external requests

401

HTTP STAUS CODES

200

allow to navigate

General security information

- On browser, the user session data are encrypted.
- The key used on local storage is a SHA-256 hash and the value an AES crypto that uses a real time computed workspace fingerprint.
- All requests send the workstation fingerprint, this fingerprint is calculated based on workstation's hardware and software.
- All access request sends the workstation location. It's not possible to navigate without permission to collect the location.
- The location is sent as latitude/longitude and precision
- The frontend code is minified e ugllified.
- Client server communication must be under HTTPS
- All access to resources generates a new audit log entry
- All authentication and authorisation operation generates a new security log entry
- valid users must have valid email under the same organisation domain.
- There are white and black IPs list
- User must to register a security phrase that will be send in all text communication (email).
- Every-time an access is done from a new device, an email will be sent to the user address.
- New users must activate his user before use. A link will be send to his email
- All private info logged data (including console.log output) is automatically obfuscated