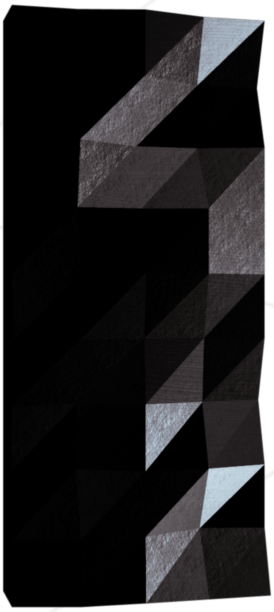

ft_transcendence

Soon you will know that you've already known things that you thought you didn't know

Summary: This is not C or C++ basic functions! To do something that you've never done before without a piscine. Remind yourself at the beginning of your journey in computer science. Now, look where you are standing. Time to shine

Version: 9

Contents

Chapter I

Mandatory part

I.1 Overview

In this subject, you will need to build a website for the mighty pong contest.

Your website will help users play pong against each other.

There will be an admin view, chat with moderators, real-time multiplayer online games.

You will need to follow those rules:

- you must use the last stable version of every framework or library.
- Your website backend should be written in NestJS.
- You must use a PostgreSQL database and no other databases.
- The front end must be written with any typescript framework.
- Your website should be a single page app, but the user should be able to use the back button on the browser https://en.wikipedia.org/wiki/Singlepage_application
- Your website must be usable on the latest version to date on Google Chrome, Firefox, Safari.
- There must be no unhandled errors or warnings when browsing through the website.
- You can use any library.
- Everything should run with a single call to `docker-compose up -build`

You should read the difference between a library and a framework

You should look at route and router

I.2 Security concerns

Because you are creating a fully working website, there are a few security concerns that you will have to tackle

- Any password stored in your database must be encrypted
- Your website must be protected against SQL injections
- You must implement some kind of server-side validation for forms and any user input

I.3 User Account

- A user must log in using the OAuth system of 42 intranet
- A user must be able to choose a unique name that will be displayed on the website
- A user has several victories and losses and other stats (ladder level, achievements etc...)
- A user must have an avatar generated or uploaded by the user
- A user must be able to activate a 2-factor authentication (like google authenticator or an SMS etc...)
- A user can add other users as friends, and see their current status (online, offline, in a game...)
- Each user has a match history (including duel, ladder) that can be consulted by anyone logged-in

I.4 Chat

- Users must be able to create channels public/private or protected by a password
- Users must be able to send direct messages to other users
- Users must be able to block other user and therefore they will not see their messages anymore
- A user that creates a new channel is automatically its owner until he leaves the channel
    - owner of a channel can add/change/remove a password to access the channel
    - owner can select a user to be an administrator and is also an administrator of the channel
        - administrator can ban or mute users for a certain amount of time
- Through the chat interface users should be able to ask other players to do a Pong match
- Through the chat interface users must be able to see other players profiles

I.5 Game

The main purpose of this website is to play pong against other players and show everyone how good you are!

Therefore we should be able to play pong directly on the website and live against another player.

There must be a match-making system, user can join a game queue and are automatically matched with another player.

It can be on a canvas or it can be with 3d effects, it can be ugly but it must be a pong like the one from 1972.

You need to have some game customization options (power-ups, different maps etc) but the user must be able to play a default pong game without any added stuff.

The game must be responsive!

Other users can watch the game live without interfering in it.

`Think about network issues, like disconnects or lag. The user experience should be the best possible.`