

CAMILLE LELOUTRE
SILVIA TARGA

JEU DU PENDU

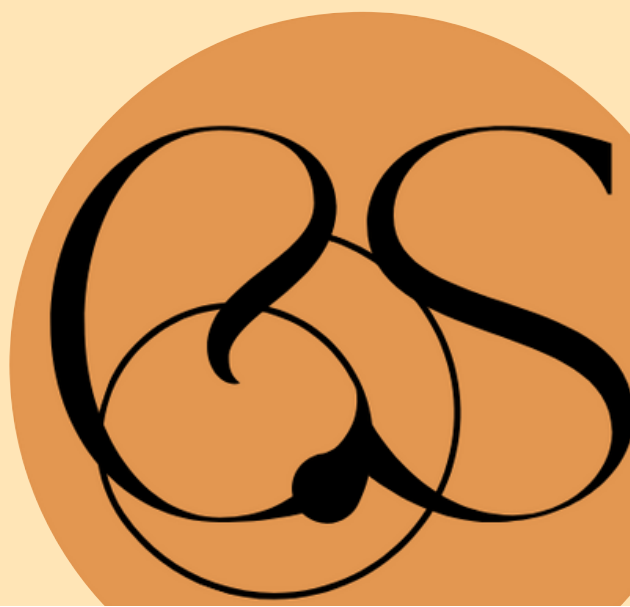


TABLE DU CONTENU

1. Enoncé
 - a. Notre objectif
 - b. Le fonctionnement de l'application
 - c. Le schéma de l'application
2. L'analyse et le développement
 - a. L'organisation du projet
 - b. Construction du projet
 - i. HTML
 - ii. Javascript
 - iii. CSS
3. Rendu visuel de l'application
4. Test de l'application
 - a. Test de l'application sur différents navigateurs
 - b. Nos difficultés
5. Conclusion

1.ENONCE

a. NOTRE OBJECTIF

Après de multiples recherche nous avons décidé de faire un PENDU. C'est un jeu très rependu depuis de nombreuses années qui pour le type de jeu nous a mis d'accord. Notre objectif final est de créer un PENDU différent de ceux repandu, avec des fonctionnalité qui rend le jeu plus emotionnant.

b. FONCTIONNEMENT DE L'APPLICATION

Le jeu consiste à deviner des lettres (A-Z) pour former un mot. Il se joue traditionnellement à deux, avec un papier et un crayon, mais dans notre version on peut y jouer sur ordinateur ou sur mobile.

Notre PENDU consiste à faire deviner un mot en anglais choisi aléatoirement en sélectionnant une lettre dans le temps imparti.

Si le joueur devine la bonne lettre qui se trouve dans le mot, la lettre s'affiche dans l'écran.

Si le joueur se trompe de lettre, le PENDU commence à se dessiner en commençant par la tête.

A mesure des erreurs, le PENDU continu de se dessiner des chaque membres du bonhomme.

Le joueur doit trouver le mot dans le temps imparti et dispose de 6 chances avant de perdre.

Le jeu est divisé en 3 niveaux de difficulté avec trois temps différents et avec des mots plus difficiles à deviner.

On a décidé de nous concentrer sur le thème de l'informatique et des langages de programmation.

On a choisi de le faire en anglais pour faciliter les mots en éliminant les caractères spéciaux et pour rendre le jeu internationale.

c. SCHEMA DE L'APPLICATION

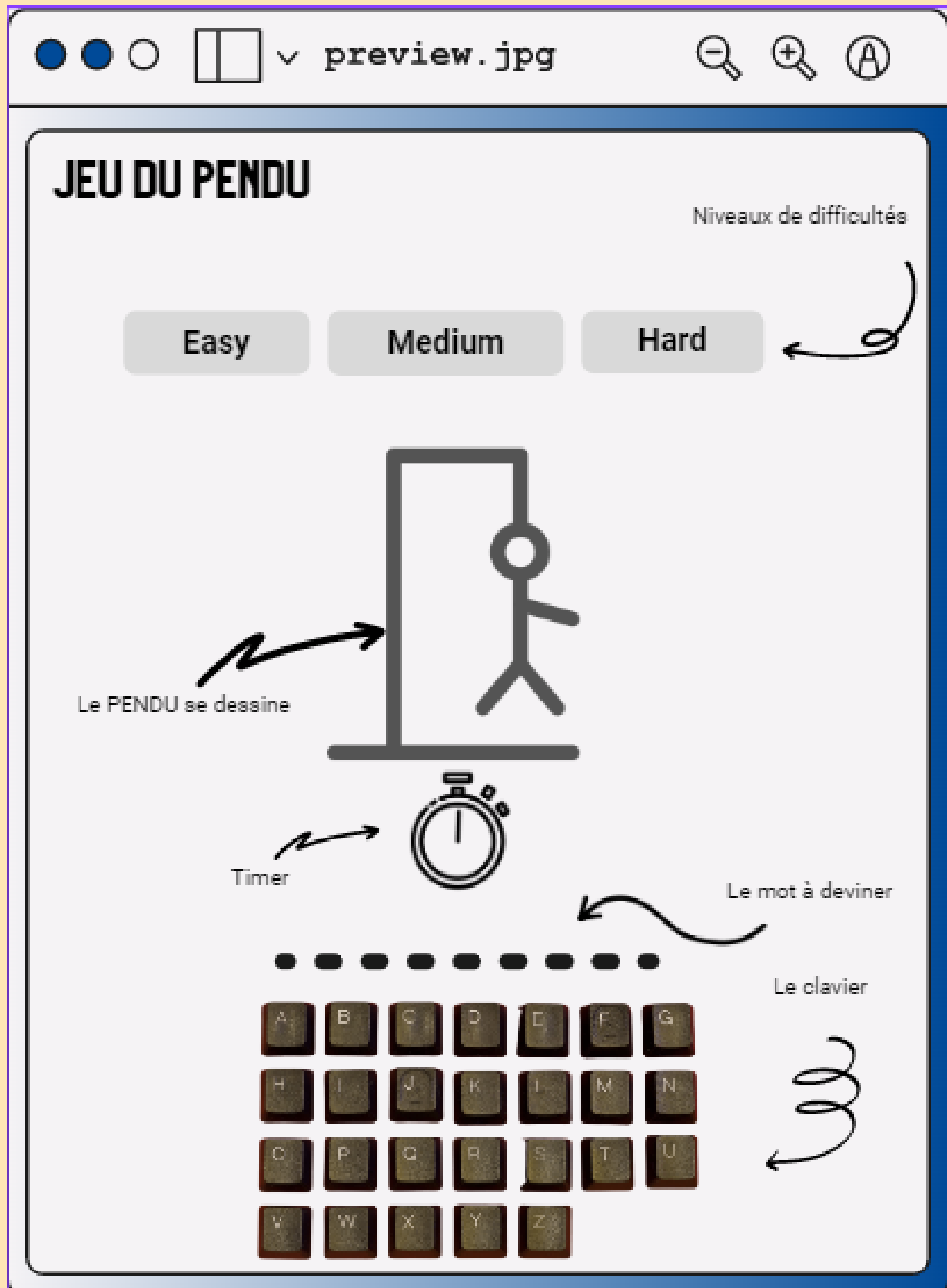
L'application comporte une page qui change en fonction de choix de l'utent.

La première page qui s'affiche est la page d'accueil qui a pour fonction de choisir entre les niveaux et de lire les règles.

En se basant sur le choix de l'utent on a en ordre:

- Le niveau plus facile avec des mots plus courts avec un timer plus long,
- Le niveau intermédiaire avec des mots de niveau moyen et un temps moyen,

- Le niveau le plus difficile avec des mots plus long et un temps plus court.



2. ANALYSE ET DEVELOPEMENT

a. L'ORGANISATION DU PROJET

Nous avons décidé d'utiliser l'interface VisualStudio Code pour écrire notre chaîne de caractères. On a décidé pour une meilleure compréhension de diviser en trois documents différents le code en se basant sur le langage utilisé. On a un document HTML, un document Javascript et un document CSS.

b. CONSTRUCTION DU PROJET

i. HTML

On a inséré une section unique avec "article", trois tables et un canvas.

Les règles, pour la compréhension du jeu et des difficultés, sont insérées dans une première partie "article".

On a ajouté une table avec trois boutons pour le choix des différents niveaux de difficultés.

On a inséré le canvas pour y dessiner notre pendu.

En succession il y a une table avec le code javascript pour la sélection du mot par rapport au niveau de difficulté et un autre avec différents boutons chacun pour les lettres de l'alphabet anglais.

ii. JAVASCRIPT

Dans la première partie on peut trouver deux fonctions démarre le jeu et réagir quand le joueur interagit avec la plateforme.

Le jeu démarre avec une des trois fonctions "startEasy", "startMedium", "startHard" qui s'occupent du choix de mot à deviner, démarrer le timer et dessiner le pendu.

La fonction principale est "guessletter" qui vient utilisé quand le joueur interagit avec le button du clavier virtuel. Si la lettre est deviné il fait un update des lettre deviné dans le mot, si non le joueur perd une vie et le pendu se dessine. Cette fonction contrôle aussi si le joueur à gagné ou perdu et met une alerte.

Une deuxième partie est composée par le timer.

On a définit la date, le minutes et seconds et avec des calculs mathématiques on découvre la différence entre le début du jeu et le moment actuel. Chaque 0.50 seconds la fonction updatetimer vient appelé.

Une troisième partie est composée par le dessine du pendu dans un canvas avec toutes les spécificité de chaque partie pour mettre à jour le pendu même.

iii. CSS

Pour définir la mise en page on a utilisé un document CSS.

RENDU VISUEL DE L'APPLICATION

Dans notre **page** on a deux parties.
Les règles

The Hangman Game

Rules

Hello we are Comille and Silvia and we present you our hangman game.
There are three difficulty levels, in each you have six lives and a different timer.
A player or a team try to guess what it is one letter at a time. The server draws a number of dashes equivalent to the number of letters in the word.
If a guessing player suggests a letter that occurs in the word, the server fills in the blanks with that letter in the right places. If the word does not contain the suggested letter, the server draws one element of a hangman's gallows.
As the game progresses, a segment of the gallows and of a victim is added for every suggested letter not in the word. The number of incorrect guesses before the game ends is six. If you complete the word before the timeout and before losing the six lives you win.

To start you have to decide the level, press the bottom and guess the letters. When you win or you lose you can restart pressing the bottom again.
If you find the same word you can repress the bottom to change it.

Objective: **Guess the word/words before your man gets hung!**

Theme: **Computer Science/Programming languages**

Levels:

- **Easy:** in this level you have to find **short word in 4 minutes**.
- **Medium:** here you have to find **medium word in 3 minutes**.
- **Hard:** the most difficult level, you have to find **long word in only 2 minutes**.

L ' interface

Start new game

Easy

Medium

Hard

Lives left: 6

Time left:

a

b

c

d

e

f

g

h

i

j

k

l

m

n

o

p

q

r

s

t

u

v

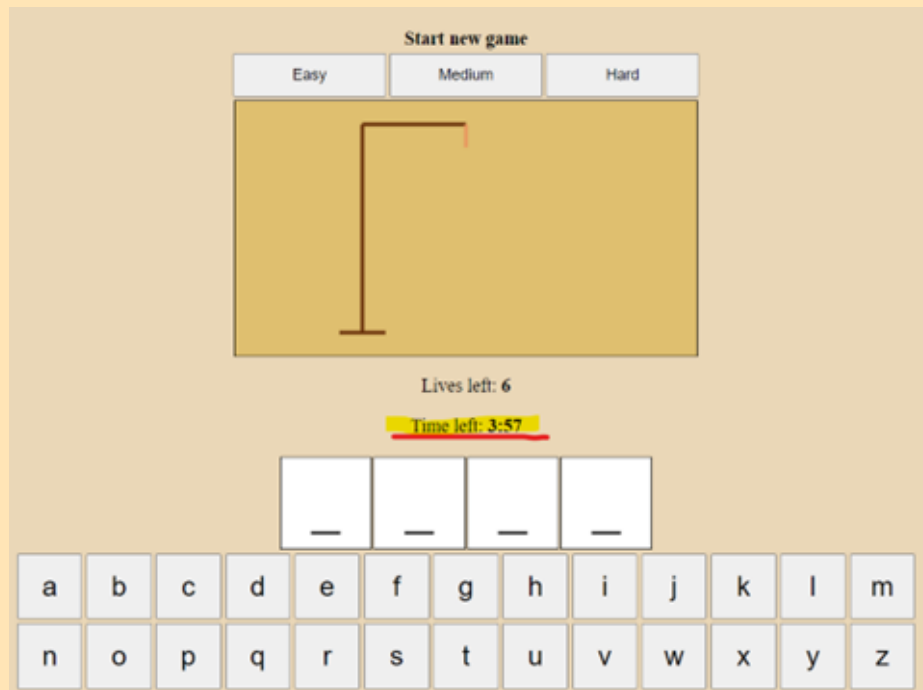
w

x

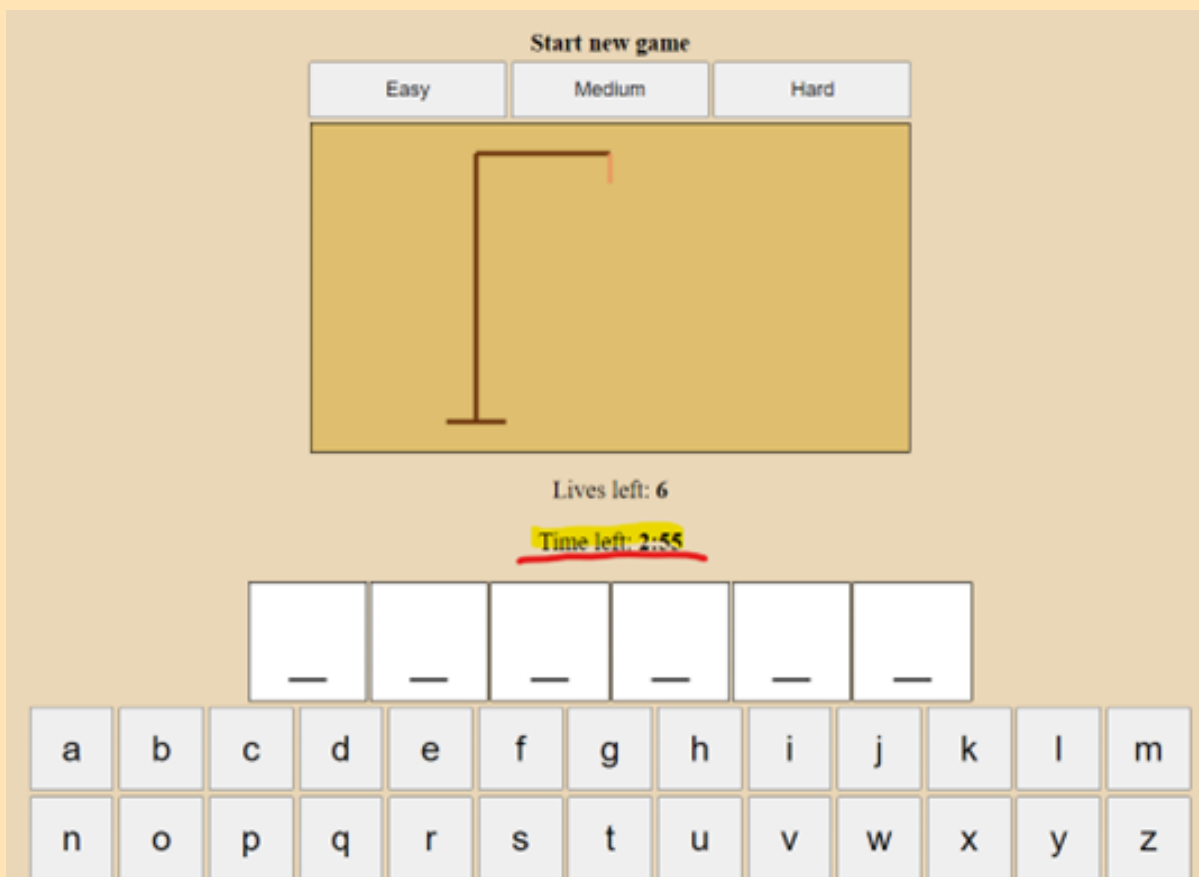
y

z

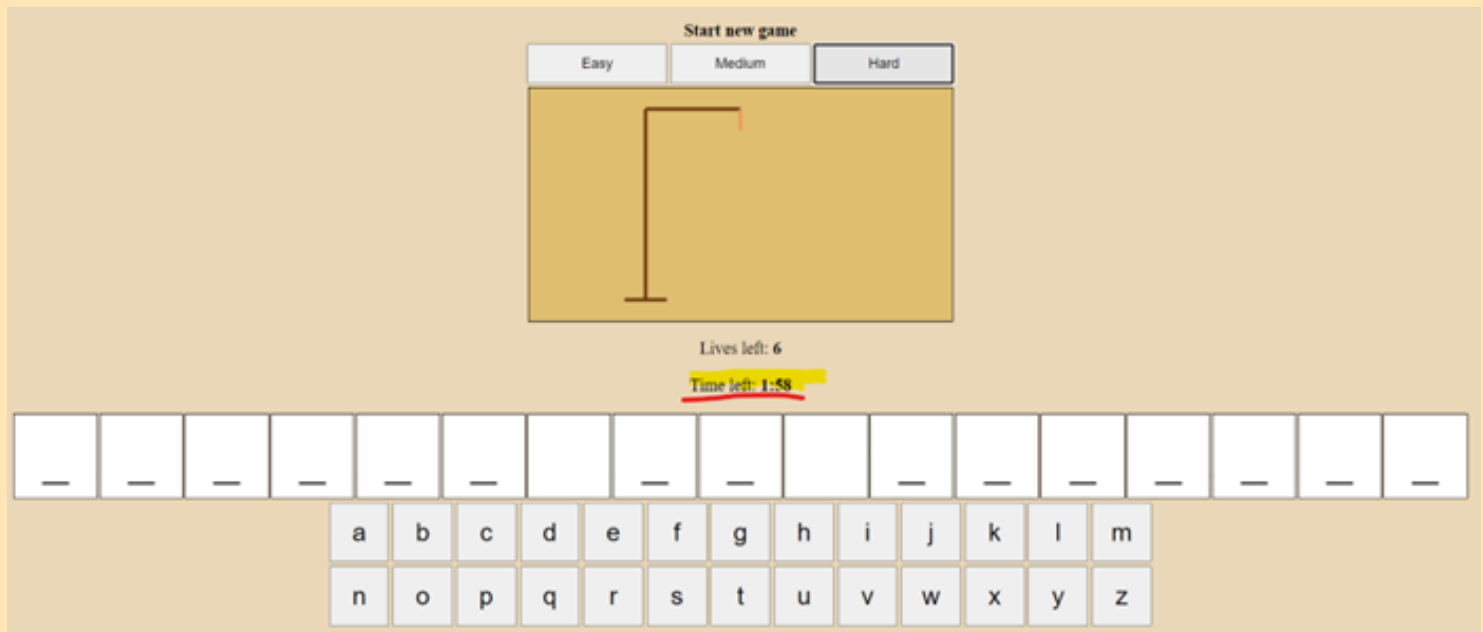
Dans le niveau **easy**, on a le timer qui commence à partir de 4:00 minutes et les mots qui sont plus courts.



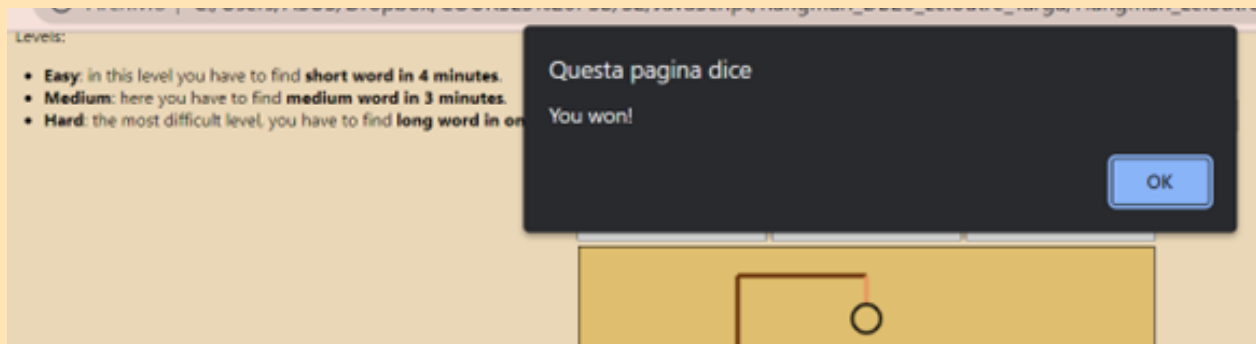
Dans le niveau **medium** on a les timer qui est sur 3:00 minutes et les mots qui sont moyens.



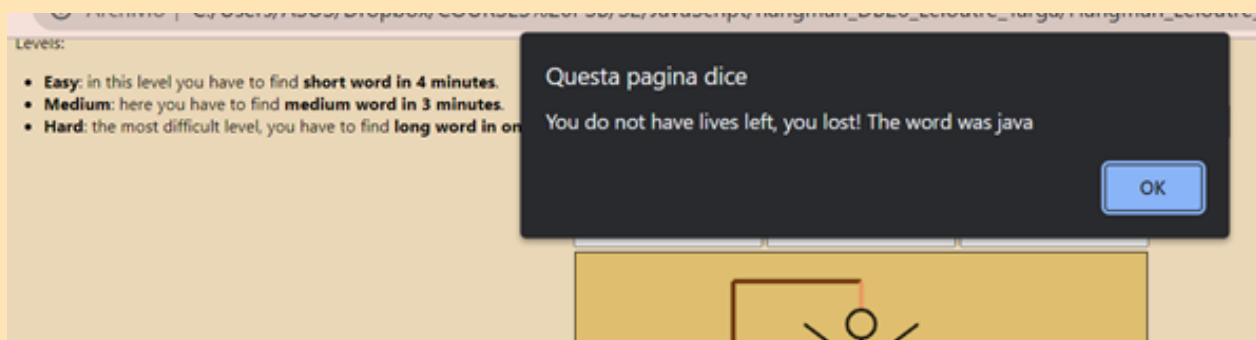
Dans le niveau **hard** qui est le niveau le plus difficile on a un timer avec 2:00 minutes et des mots qui sont plus longs.



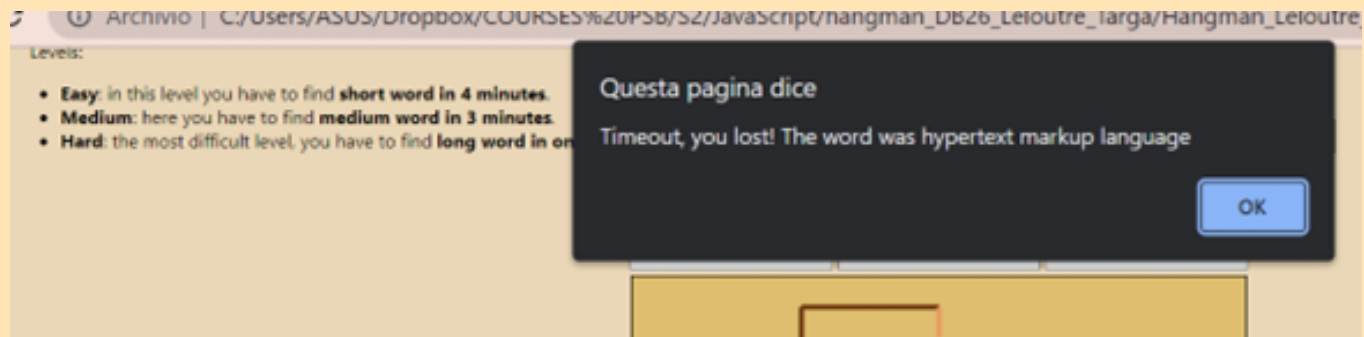
Lorsqu'on gagne, il y a un bouton alerte qui s'affiche avec le message "You Won!"



Lorsqu'on perd toutes les six vies, il y a un bouton alerte qui s'affiche avec l'écriture "You do not have lives left, you lost!" et le mot que le joueur aurait dû deviner.



Lorsqu'on perd à cause du timeout il y a un bouton alerte qui s'affiche avec l'écriture "Timeout, you lost!" et le mot que le joueur aurait dû deviner.



4. TEST DE L'APPLICATION

a. TEST DE L'APPLICATION SUR DIFFERENTS NAVIGATEURS

Nous avons testé notre PENDU sur différents navigateurs comme on peut voir dans la capture d'écran: Microsoft Edge, Firefox, Chrome, et nous avons bien constaté ont un bon fonctionnement. Nous n'avons pas rencontré de différences lors de nos tests sur différents navigateurs.

MICROSOFT EDGE



FIREFOX



CHROME



b. DIFFICULTES

On a eu des difficulté à implémenter le timer. En détail on a du penser à toutes les possibilités et les mettre en place et utiliser des formules mathématique pour faire l'update. L'update devait rappeler soi meme chaque mi-second et on a du implémener un timer avec simplement de formules mathématique de différence sans utiliser une bibliothèque externe.

5. CONCLUSION

En conclusion, nous avons mis en place le jeu du PENDU, avec trois niveaux de difficultés liés à la longueur des mots et au timer. Nous avons rendu un jeu sur une plateforme digitale. Le principe est le même, le joueur a droit à six tentatives pour deviner le mots en ajoutant des lettres.

En ce qui concerne le fonctionnement de l'application, elle comporte une page avec les règles et qui présente notre interface. Celle-ci change selon le niveaux choisi par le joueur. Quand un joueur fait son choix un mot aléatoire apparait. Concernant la mise en page on l'a fait avec CSS et la structure avec HTML.



CAMILLE LELOUTRE



SILVIA TARGA