

# Projet personnel (10%)

---

Design Web – Automne 2025

## Consignes

- Le projet compte pour 10% de la note finale.
- La correction se fera **en présence de votre enseignant** durant les cours du **5 et 8 décembre**.
- Vous pouvez prendre rendez-vous avant cette date si vous êtes prêt à être évalué.
- Vous devez aussi **remettre une version de votre code** sur Github.
- Retourner le devoir Teams avec l'adresse url de votre projet et assurez-vous que le dépôt soit **public**.

## Description du projet

Vous devez une version simplifiée du jeu « Cookie Clicker ». Si vous ne connaissez pas ce type de jeux, vous en avez un exemple ici : <https://orteil.dashnet.org/cookieclicker/>. Le but du jeu est d'amasser des biscuits. Quand on clique sur le biscuit, on ajoute un nombre x de biscuits à notre total. On peut acheter des curseurs qui vont cliquer à notre place et aussi améliorer le nombre de biscuits qu'on récolte à chaque clique que l'on fait.

La mise en page et les fonctionnalités à implémenter sont décrites plus bas.

## Mise en page

Vous retrouverez en annexe A un aperçu de la mise en page à recréer. Les couleurs et les images à utiliser sont laissées à votre discrétion mais vous devez respecter les points suivants :

- La zone de jeu doit être centré horizontalement dans et à une distance de 5rem du haut de la page.
- Une image doit couvrir l'entièreté du fond de la page.
- Reproduisez l'effet de transparence de la zone de jeu (*glassmorphism* en anglais).
- Ajoutez un favicon de biscuit à votre page

# Fonctionnalités JavaScript

## Produire des biscuits

Quand on **clique** sur le biscuit, on ajoute un nombre de biscuits à notre total.

- Ce nombre est de 1 au chargement du jeu mais on peut l'améliorer (voir plus bas)
- Ajouter une animation sur le biscuit lors du clic de la souris.
- Lors du clic, incrémenter en conséquence le total de biscuits du nombre produit.

À chaque seconde, les **curseurs** que l'on possède produisent aussi des biscuits.

- Chaque curseur produit un seul biscuit. Donc si on possède 5 curseurs, c'est 5 biscuits qui sont produits à chaque seconde.

## Acheter des curseurs

- On peut cliquer sur le bouton pour acheter un curseur uniquement si le total de biscuits est supérieur ou égal au prix d'un curseur.
- Modifier le style du bouton selon que l'achat est possible ou non.
- Le prix du curseur s'incrémente selon le nombre de curseurs qu'on possède. Utilisez la fonction que je vous fournis à l'annexe B pour calculer le prix courant d'un curseur.
- Dans le bouton d'achat vous avez à droite le nombre de curseurs possédés.
- Quand on clique sur le bouton, diminuez le total de biscuits du prix du curseur.
- Sous le total de biscuits vous avez aussi le nombre de biscuits produits par seconde qui correspond au nombre de curseurs possédés.

## Amélioration du clic de souris

Il est possible d'améliorer le clic de la souris pour produire plus de biscuits à chaque clic.

- On peut cliquer sur le bouton uniquement si le total de biscuits est supérieur ou égal au prix de l'amélioration.
- Modifier le style du bouton selon que l'achat est possible ou non.
- Le prix de l'amélioration s'incrémente selon une fonction. Utilisez la fonction que je vous fournis à l'annexe B pour calculer le prix courant d'un curseur.
- Dans le bouton d'achat vous avez à droite le nombre de biscuits produits à chaque clic.
- Quand on clique sur le bouton, diminuez le total de biscuits du prix de l'amélioration.
- Ajustez aussi la valeur du nombre de biscuits par clic affiché sous le total de biscuits.

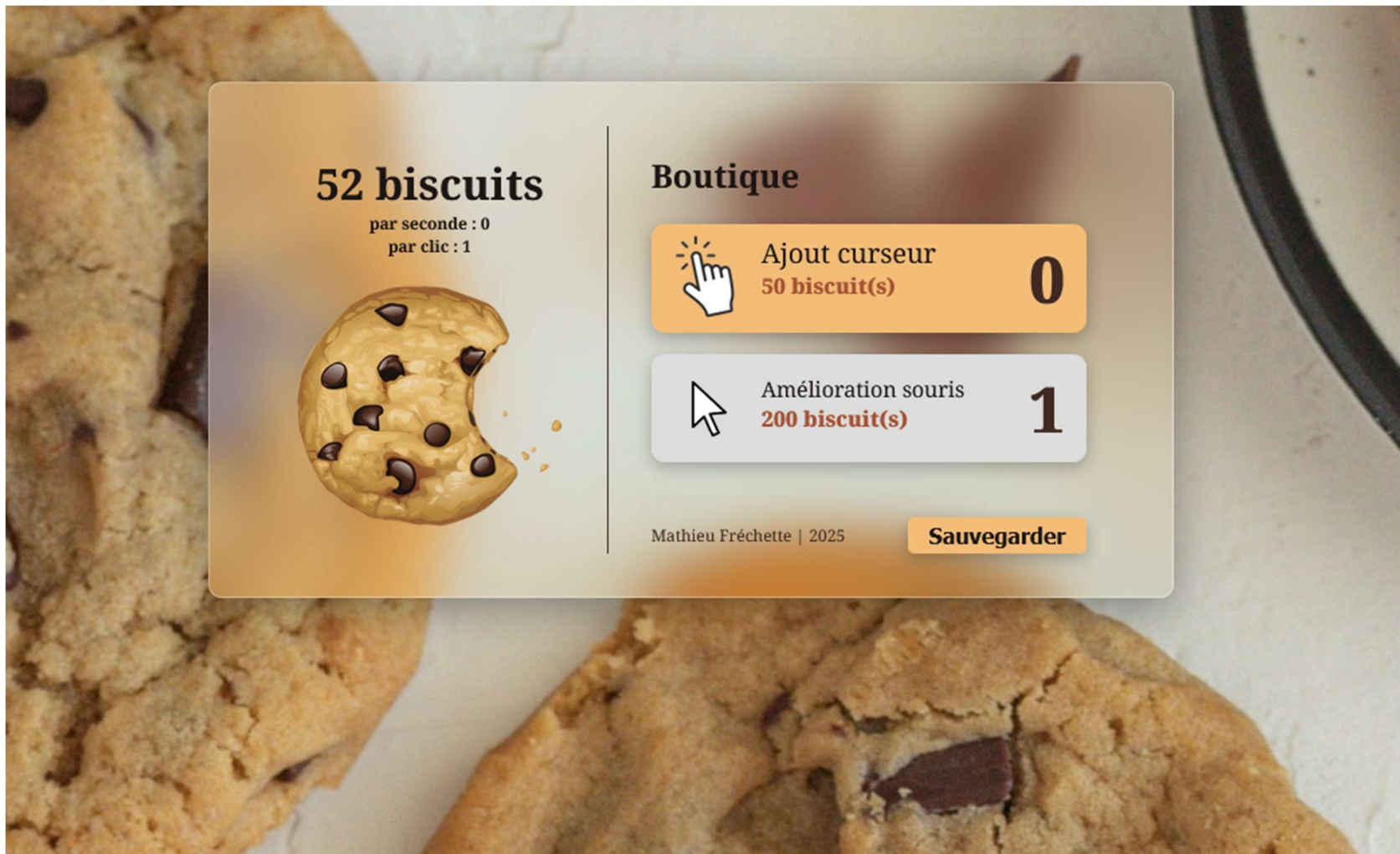
## Sauvegarder la progression

- En cliquant sur le bouton Sauvegarder enregistrer les valeurs suivantes dans le stockage local :
  - Le total courant de biscuits
  - Le nombre de curseurs possédés
  - Le nombre de biscuits produit à chaque clic
- Au chargement de la page, récupérez ces valeurs et ajuster le jeu en conséquence.
- N'oubliez pas de recalculer le prix du curseur et de l'amélioration après avoir chargé les valeurs du nombre de curseurs et de clic.

## Grille d'évaluation

CRITÈRES	INDICATEURS
<b>Respect de la mise en page</b>	<ul style="list-style-type: none"> <li>• La zone de jeu est centrée horizontalement et distant du haut de page.</li> <li>• Une image couvre tout le fond de la page.</li> <li>• La mise en page de la zone de jeu respecte le gabarit.</li> <li>• Il y a un favicon associé à la page</li> </ul>
	/ 2
<b>Production de biscuits</b>	<ul style="list-style-type: none"> <li>• Quand on clique sur l'image de biscuit le nombre adéquat de biscuits est produit.</li> <li>• Il y a une animation sur le biscuit lors du clic.</li> </ul>
	/ 2
<b>Fonctionnement des curseurs</b>	<ul style="list-style-type: none"> <li>• L'achat de curseurs est fonctionnel.</li> <li>• L'affichage est rafraîchi adéquatement.</li> </ul>
	/ 2
<b>Amélioration du clic</b>	<ul style="list-style-type: none"> <li>• L'amélioration du clic est fonctionnelle.</li> <li>• L'affichage est rafraîchi adéquatement.</li> </ul>
	/ 2
<b>Sauvegarde de la partie</b>	<ul style="list-style-type: none"> <li>• Les informations sont bien sauvegardées.</li> <li>• Au chargement de la page les informations sont récupérées et le jeu est ajusté.</li> </ul>
	/ 2
<b>Total</b>	<b>/ 10</b>

## Annexe A



## Annexe B

```
function CalculPrixCurseur() {  
    // base = Le prix de base  
    // alpha = taux de croissance de base (ex: 0{,}12 – 0{,}20).  
    // beta = facteur de courbure  
    // (ex: 1 pour quasi-exponentiel standard,  
    //      <1 pour ralentir la croissance,  
    //      >1 pour l'accélérer légèrement).  
  
    const base = 50;  
    const alpha = 0.25;  
    const beta = 1.05;  
    const prix = base * Math.pow(1 + alpha, Math.pow(totalCurseur, beta));  
    return Math.round(prix);  
}
```

Fonction à utiliser pour calculer le prix d'un curseur selon le total des curseurs que le joueur possède.

La variable *totalCurseur* contient le total des curseurs possédés. Elle est déclarée globalement dans le code et initialisé à 0.

Vous pouvez faire une autre fonction similaire pour calculer le coût d'amélioration d'un clic. Vous devrez créer une variable globale pour remplacer *totalCurseur* et ajuster le prix de base. J'ai utilisé 200 mais c'est à vous de voir.