

# Énoncé du mini-projet 2 : Memory

Grégory Baltus

2025

## 1 Introduction

L'objectif de ce mini-projet est de créer un site web sur lequel vous pourrez jouer à un Memory. Pour rappel, un Memory est un jeu qui consiste à retourner des cartes disposées face cachée afin de retrouver des paires identiques. Le joueur retourne deux cartes à chaque tour : si elles forment une paire, il les garde et rejoue ; sinon, il les replace face cachée et peut ensuite retourner deux autres cartes. Le jeu se termine lorsque toutes les paires ont été trouvées ou lorsque le temps imparti est écoulé.

## 2 Rendu visuel du projet

Le projet doit ressembler à ce squelette :



Figure 1: Squelette du Memory

Vous avez une liberté sur plusieurs points :

- les couleurs de fond et du texte ;
- la police ;
- les images (les images contenues sur les cartes) ;
- les différentes proportions.

Malgré ces libertés, veuillez à ce que vos choix respectent certains critères :

- le rendu visuel doit être harmonieux (les éléments sont espacés correctement, les proportions sont logiques et agréables à l'œil, le choix des couleurs est cohérent) ;
- le rendu doit être lisible et compréhensible par tous (pensez aux daltoniens et aux personnes souffrant de dyslexie).

## 3 Contraintes

### 3.1 Contraintes sur le visuel

Bien que vous ayez certaines libertés, plusieurs contraintes doivent être respectées. Commençons par celles liées aux éléments à inclure sur votre site :



Figure 2: Éléments indispensables

1. votre site doit contenir un *header* comportant le titre de votre site et une petite phrase d'explication ;
2. une première *section* contenant les trois éléments suivants :
3. une zone pour le chronomètre comprenant le temps restant et le temps déjà écoulé ;
4. une zone pour diverses informations, comprenant au minimum le nombre de paires trouvées, le nombre de paires restantes et le nombre total de fois que le joueur a retourné des cartes ;
5. un bouton permettant de démarrer la partie. C'est ce bouton qui lancera le chronomètre et rendra les cartes cliquables ;
6. une *section* comprenant les paires trouvées. Une fois qu'une paire a été trouvée, elle apparaît dans cette zone (voir 4) ;
7. une *section* correspondant à la zone de jeu. Celle-ci devra contenir les cartes (voir le point suivant). Il y aura par défaut 4x4 cartes, c'est-à-dire 16 cartes au total ;
8. les cartes : celles-ci peuvent être face cachée ou face visible (voir 3) ;
9. un *footer* avec le copyright.

### 3.1.1 Contraintes de responsivité

Votre site doit être un minimum responsive, c'est-à-dire que s'il est redimensionné, il doit rester lisible et utilisable :

- pas de débordement de texte ;
- pas de superposition inadaptée des éléments ;
- pas de compression exagérée des images.

### 3.2 Contraintes JS

Les cartes qui sont dans la *section* zone de jeu doivent être générées par JavaScript. C'est-à-dire qu'il ne peut pas y avoir de *div* (ou autre) représentant les cartes dans votre code HTML. Il faut les générer dans le code JavaScript à l'aide d'une boucle.

## 4 Fonctionnement du jeu

### 4.1 Avant le début de la partie

Votre site devra initialement ressembler à 1, seul le bouton "Commencer la partie" est cliquable. Si l'utilisateur essaie de cliquer sur des cartes avant d'appuyer sur le bouton "Commencer la partie", rien ne se passe, les cartes ne se retournent pas.

### 4.2 Début de la partie

Quand l'utilisateur clique sur le bouton "Commencer la partie", celui-ci se grise et devient non cliquable (voir image 3). Le chronomètre est également lancé, il est visible au point 3 de l'image 2. Les cartes deviennent cliquables.

### 4.3 Cliquer sur une carte

Quand l'utilisateur clique sur une carte, si aucune autre carte n'était retournée, la carte sur laquelle il a cliqué se retourne, c'est-à-dire que l'image sur la carte devient visible. Cette carte reste retournée tant que l'utilisateur ne clique pas sur une autre carte.

Si une autre carte était déjà retournée, la carte sur laquelle l'utilisateur a cliqué se retourne également (l'image sur la carte devient visible). Si les deux cartes ont une image différente, elles restent alors visibles pendant 1 seconde avant de se retourner et donc de ne plus avoir leur image visible. Durant cette seconde où les deux cartes sont révélées, cliquer sur une autre carte n'a pas d'effet.

L'image 3 représente une séquence d'actions lorsqu'on clique sur une première carte, puis sur une seconde, et finalement lorsque les cartes redeviennent faces cachées.

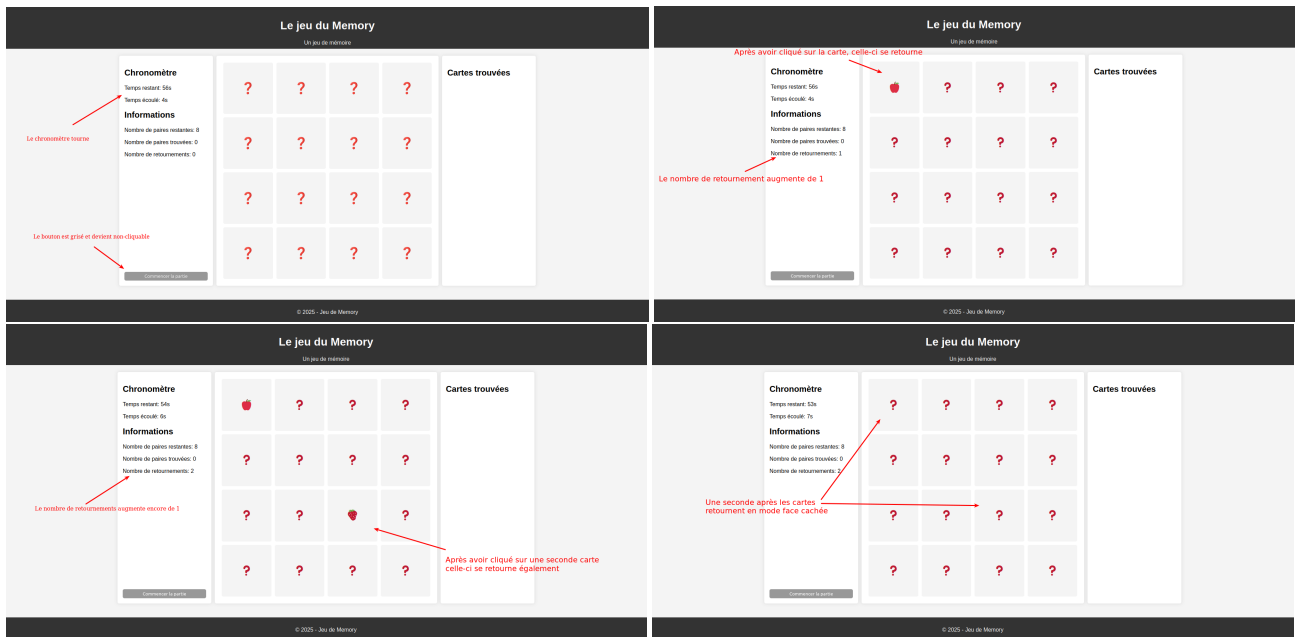


Figure 3: Haut à gauche : le joueur a appuyé sur "Commencer la partie". Haut à droite : le joueur a appuyé sur une carte. Bas à gauche : le joueur a appuyé sur une deuxième carte. Bas à droite : une seconde plus tard, les cartes se retournent.

#### 4.4 Trouver deux cartes identiques

Si la deuxième carte retournée a la même image que la première, ces deux cartes restent affichées 0.5 seconde avant de disparaître. De plus, dans la *section* "cartes trouvées", une carte apparaît avec l'image de la paire trouvée. Le nombre de paires restantes décrémente de 1 tandis que le nombre de paires trouvées augmente de 1. L'image 4 représente ce comportement.

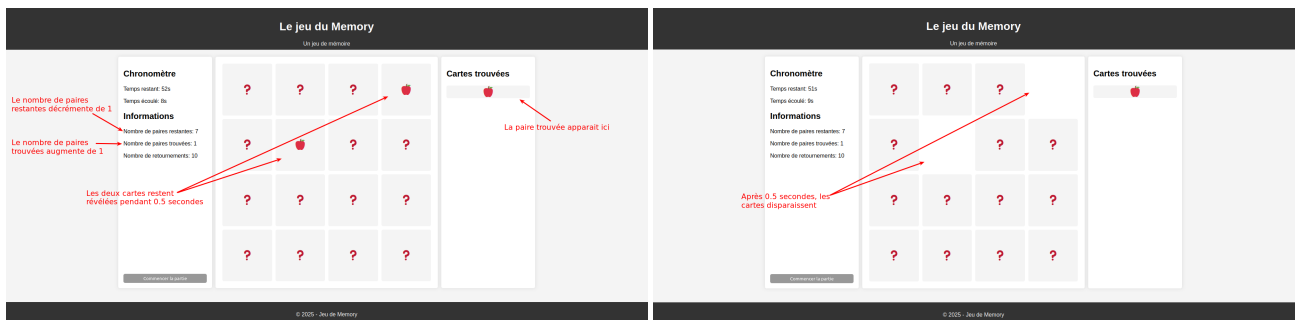


Figure 4: Gauche : le joueur a appuyé sur une seconde carte identique à la première. Droite : après 0.5 seconde, ces deux cartes disparaissent.

#### 4.5 Milieu de partie

Après quelques cartes trouvées, le jeu ressemble à ceci :

#### 4.6 Fin de partie : Victoire

Lors d'une victoire, c'est-à-dire quand toutes les paires ont été trouvées, une pop-up s'affiche pour signaler la victoire :  
Le chronomètre s'arrête.

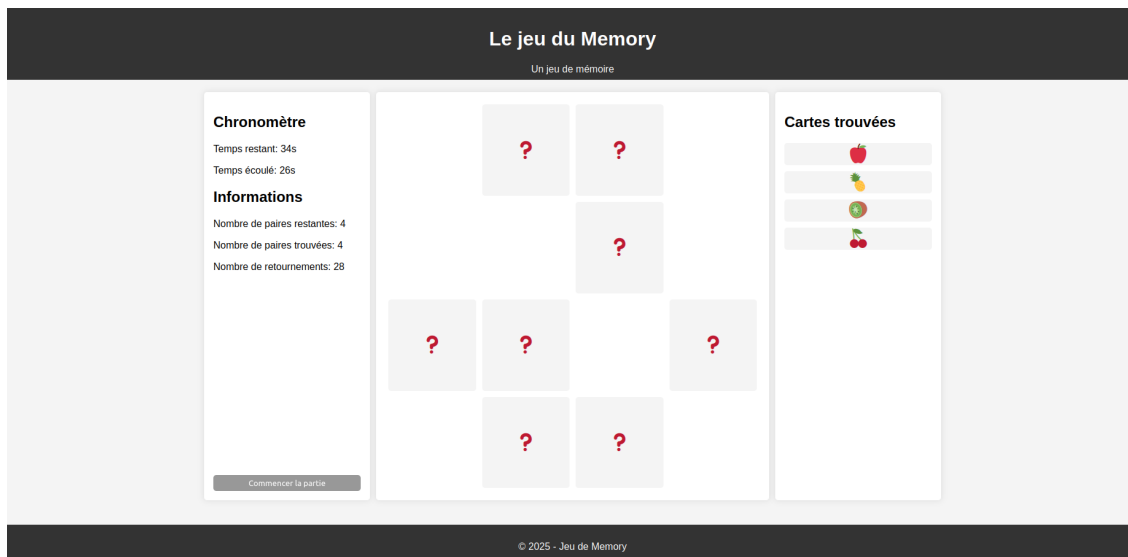


Figure 5: Milieu de partie

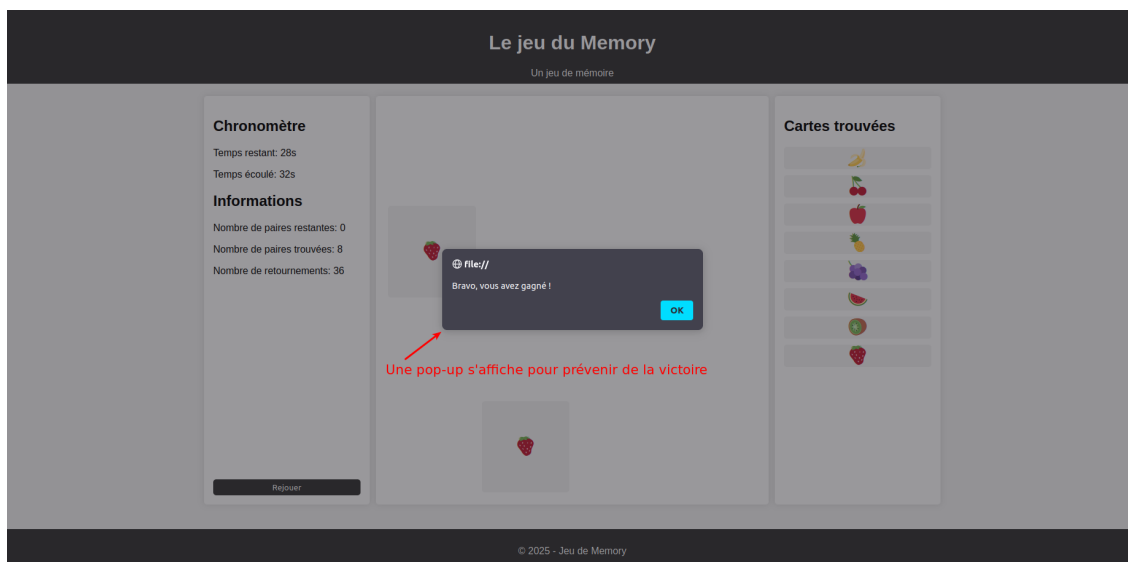


Figure 6: Victoire

#### 4.7 Fin de partie : Défaite

Lors d'une défaite, c'est-à-dire quand le chronomètre est arrivé à 0 et que toutes les paires n'ont pas été trouvées, une pop-up s'affiche pour signaler la défaite :

Le chronomètre s'arrête et les cartes restantes deviennent non cliquables.

#### 4.8 Fin de partie : Rejouer

Quand le joueur a appuyé sur le bouton "OK" de la pop-up signalant soit une victoire soit une défaite, la pop-up disparaît et le bouton "Commencer la partie" est remplacé par un bouton "Rejouer", qui n'est pas grisé et est cliquable.

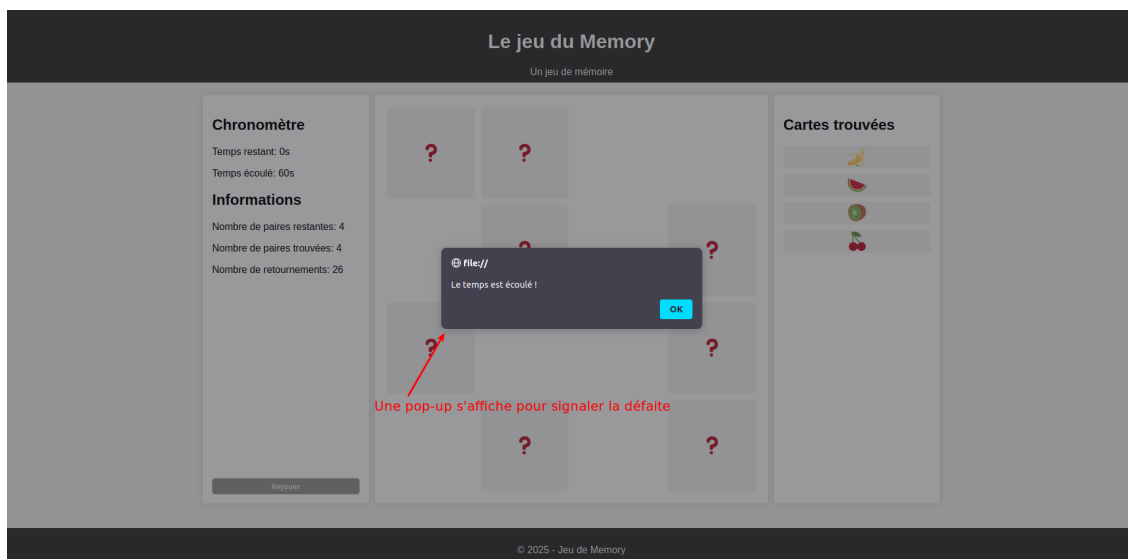


Figure 7: Défaite

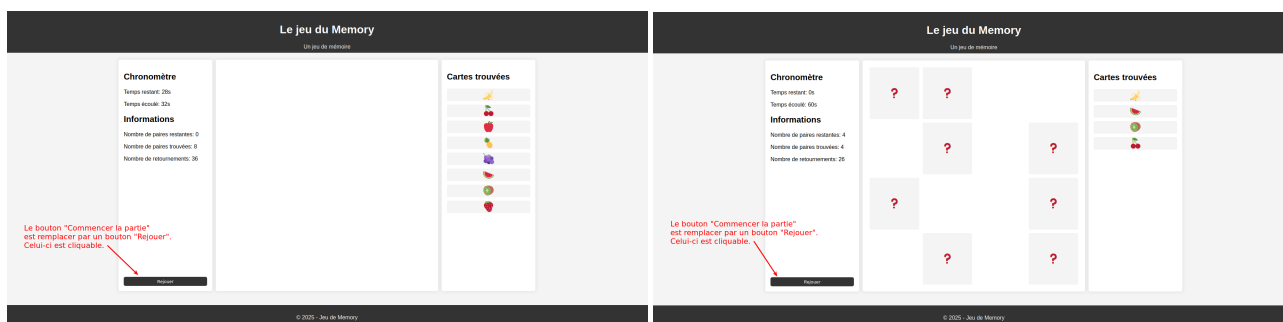


Figure 8: Gauche : Rejouer dans le cas d'une victoire. Droite : Rejouer dans le cas d'une défaite.

## 4.9 Une nouvelle partie

Quand le joueur appuie sur le bouton "Rejouer", celui-ci se grise et devient non cliquable, une nouvelle partie démarre.



Figure 9: Une nouvelle partie démarre

## 5 Bien démarrer

### 5.1 HTML, CSS et JS

Pour bien démarrer, commencez par créer un fichier `index.html`, un fichier `style.css` (dans un dossier nommé `css`) et un fichier `script.js` (dans un dossier `js`). Vous pouvez ajouter d'autres fichiers `.js` si vous en trouvez la nécessité.

Commencez par compléter le fichier `index.html`. Une fois cela fait, remplissez le fichier `style.css`. Pour réaliser la zone de jeu, il est intéressant d'utiliser un nouveau *display*, il s'agit du *display : grid*. Pour plus d'informations sur ce *display* consulter la documentation [https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_grid\\_layout/Basic\\_concepts\\_of\\_grid\\_layout](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_grid_layout/Basic_concepts_of_grid_layout).

Quand cela est terminé, vous pouvez enfin passer au JavaScript.

### 5.2 Conseils pour le JavaScript

Voici quelques conseils pour la partie JavaScript :

- Commencez par définir les différentes variables dont vous aurez besoin ;
- Générez les cartes sur la zone de jeu grâce à une boucle ;
- Pour chaque carte, ajoutez un *addEventListener* qui appellera une fonction lors d'un clic, permettant de révéler cette carte ;
- Dans ce même *addEventListener*, ajoutez une fonction qui vérifie s'il s'agit de la deuxième carte révélée. Si c'est le cas, les cartes doivent repasser en face cachée après 1 seconde ;
- Ajoutez une méthode qui vérifie si les deux cartes retournées sont identiques. Si c'est le cas, faites en sorte qu'elles disparaissent après 0,5 seconde ;
- Créez une méthode asynchrone pour faire écouler le chronomètre ;
- Ajoutez un *addEventListener* sur le bouton "Commencer la partie" permettant de déclencher le chronomètre et de rendre les cartes cliquables (utilisez une variable `gameOn`), puis modifiez le reste de votre code au besoin ;

- Modifiez vos fonctions pour qu'elles mettent à jour les différentes statistiques (mise à jour du chronomètre, des cartes trouvées, du nombre de cartes retournées, etc.) ;
- Ajoutez la condition de défaite ainsi qu'une pop-up ;
- Ajoutez la condition de victoire ainsi qu'une pop-up ;
- Créez une méthode qui permet d'afficher les cartes trouvées dans la zone "cartes trouvées" ;
- Modifiez votre code pour y ajouter une méthode qui va préparer la partie, c'est-à-dire initialiser les différentes variables et créer les cartes ;
- Ajoutez un bouton "Rejouer" dans le HTML et faites en sorte que celui-ci ne soit pas visible au départ ;
- Ajoutez un *addEventListener* pour le bouton "Rejouer" ;
- Ajoutez une méthode qui, en fin de partie, va cacher le bouton "Commencer la partie" et afficher le bouton "Rejouer".

## 6 Dépôt

Veillez à déposer votre projet compressé sur la plateforme Learn du cours avant le **\*\*25/04/2025\*\*** à 23h59.

Bon travail !