

JEU DE PLATEAU JS



JEU DE PLATEAU TOUR PAR TOUR EN JAVASCRIPT

- Présentation des éléments visibles du jeu,
- Le Game Manager,
- L'Event Manager,
- Le Display Manager,
- Index de définitions

GÉNÉRER LA CARTE DU JEU

.....

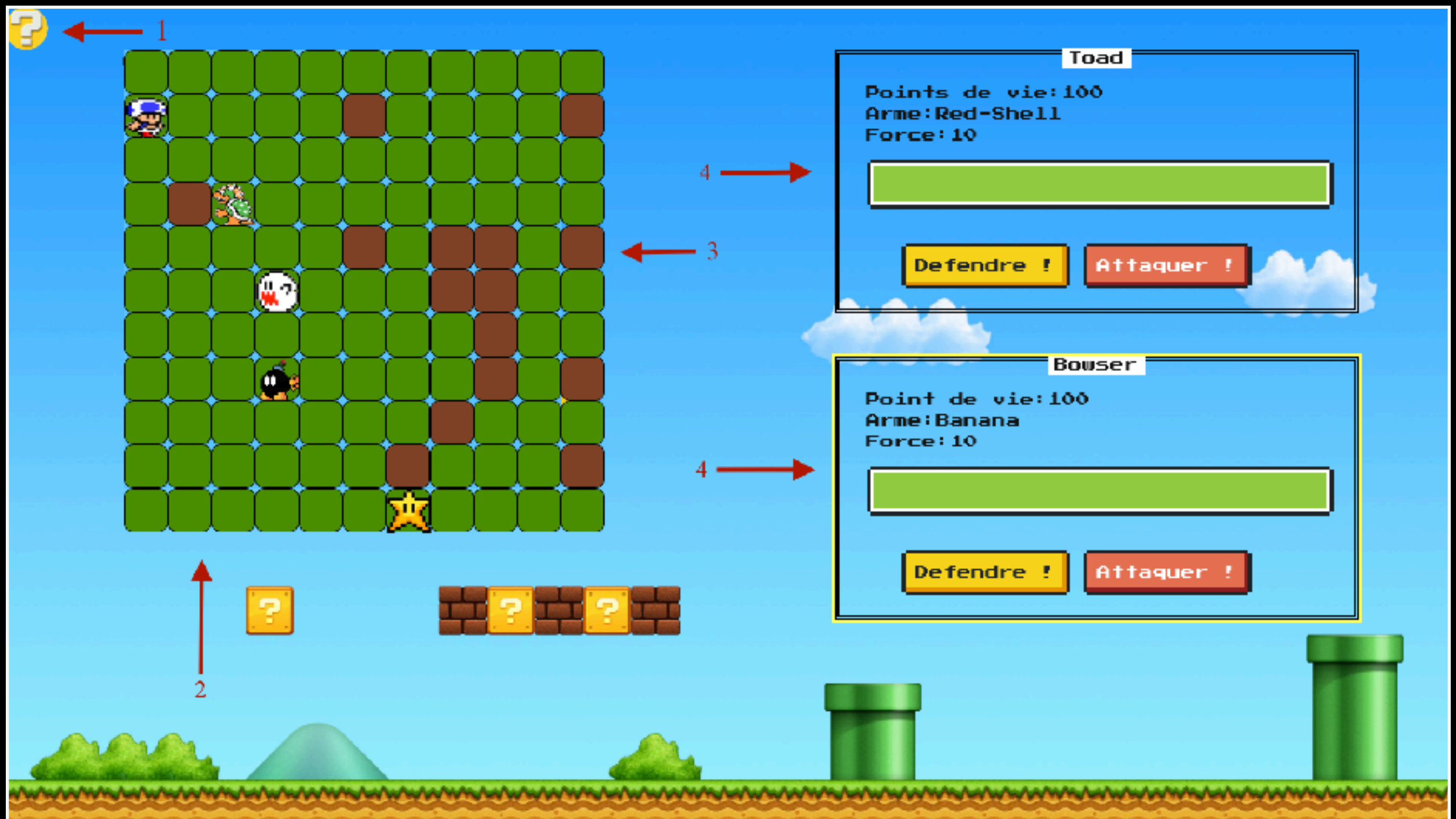
Générer aléatoirement la carte du jeu dont chaque case peut être soit :

- Vide
- Inaccessible (grisée)

Sur la carte, un nombre limité d'armes (4 maximum) sera placé aléatoirement et pourra être récolté par les joueurs qui passeraient dessus.

A chaque tour, un joueur peut se déplacer d'une à trois cases (horizontalement ou verticalement) avant de terminer son tour. Il ne peut évidemment pas passer à travers un obstacle.





Présentation générale du jeu:

1. Info-bulle contenant les règles du jeu,
2. Plateau de jeu, contenant les éléments joueurs et armes,
3. Les cases marrons sont les cases inaccessibles (grisées), les joueurs ne peuvent pas passer dessus ou à travers,
4. Encadrements autobiographiques des joueurs.

LES JOUEURS

.....

Le placement des deux joueurs est aléatoire sur la carte au chargement de la partie. Ils ne doivent pas se toucher (ils ne peuvent pas être côte à côte).

A chaque tour, un joueur peut se déplacer d'une à trois cases (horizontalement ou verticalement) avant de terminer son tour.

Si les joueurs se croisent sur des cases adjacentes (horizontalement ou verticalement), un combat à mort s'engage.

LES PERSONNAGES DISPONIBLES

.....

Le visuel des personnages est déterminé au hasard, sept sont disponibles. Le récapitulatif de leur caractéristique se fait dans les encadrements biographiques qui se situent à droite du plateau.

Chaque encadrement contient le nom du personnage, ses points de vie (100 au départ), son arme (par défaut pour commencer, puis l'arme qu'il récupère si cela est le cas) et la force de celle-ci ainsi que sa jauge de vie et les boutons de combat.

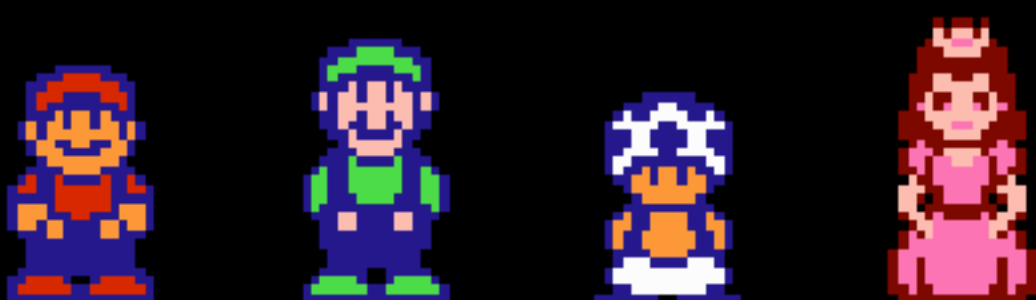
Chaque encadrement est illuminé à tour de rôle pour désigner le joueur dont c'est le tour.

Les personnages disponibles:

- Yoshi, Kirby, Luigi, Mario, Toad, Mario-coon et Bowser.



PLEASE SELECT
PLAYER



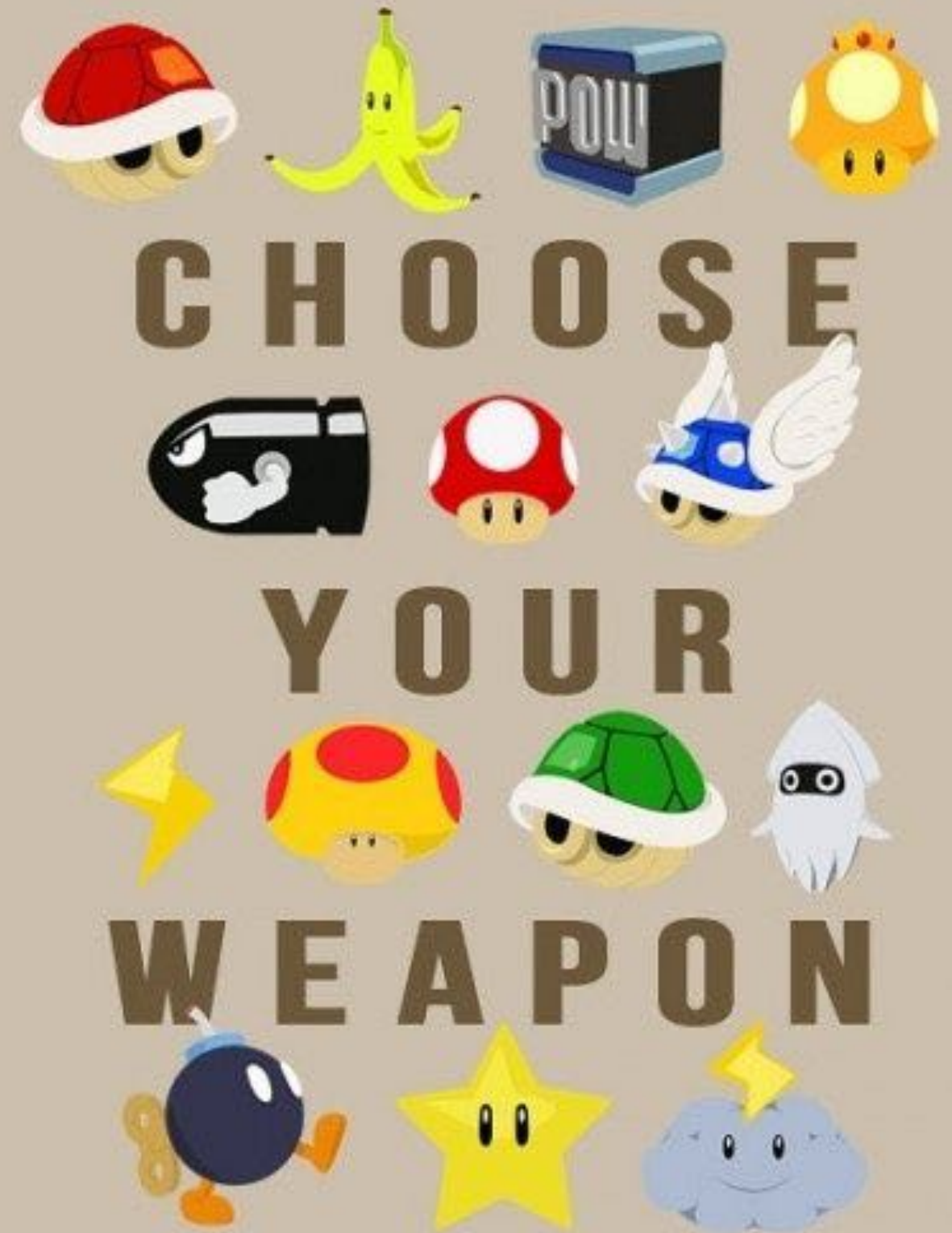
ARMES

.....

Sur la carte, un nombre limité d'armes (4 maximum) sera placé aléatoirement et pourra être récolté par les joueurs qui passeraient dessus.

Au moins 4 types d'arme doivent être disponibles dans le jeu, avec des dégâts différents. L'arme par défaut qui équipe les joueurs doit infliger 10 points de dégâts. Chaque arme a un nom et un visuel associé.

Si un joueur passe sur une case contenant une arme, il laisse son arme actuelle sur place et la remplace par la nouvelle.



LES ARMES DISPONIBLES

.....

Le visuel des armes est déterminé de manière aléatoire ainsi que leurs points de force compris entre 20 et 50.

Les armes par défauts ont une force de 10 points comme spécifié dans les instructions.

Sept Armes différentes sont disponibles, mais le nombre d'armes affichées sur le plateau (max 4) dépend de la taille de celui-ci.

Plus le plateau est grand, moins le nombre d'armes sera élevé. Pour un plateau de 10 colonnes et lignes, celui-ci contiendra les 4 armes. Pour 11 colonnes et ligne, seulement 3 armes. Et enfin 12 colonnes et ligne, 2 armes disponibles.

Armes du jeu:

- Red-shell, Fire-flower, Banana, Star, Boo, Bob-omb et Yoshi-egg.



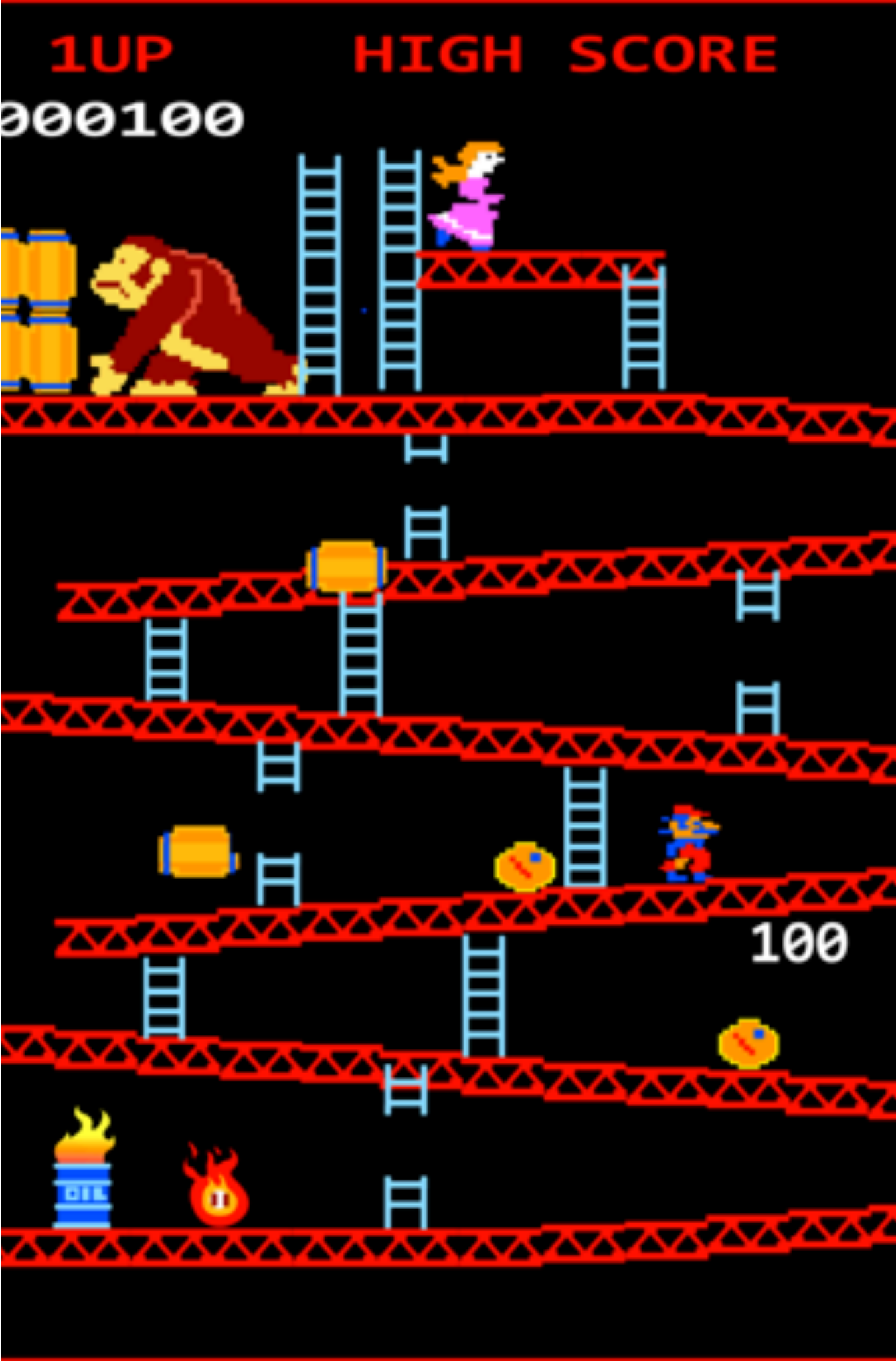


COMBAT

.....

Lors d'un combat, le fonctionnement du jeu est le suivant :

- Chacun attaque à son tour
- Les dégâts infligés dépendent de l'arme possédée par le joueur
- Le joueur peut choisir d'attaquer ou de se défendre contre le prochain coup
- Lorsque le joueur se défend, il encaisse 50% de dégâts en moins qu'en temps normal
- Dès que les points de vie d'un joueur (initialement à 100) tombent à 0 , celui-ci a perdu. Un message s'affiche et la partie est terminée.



GAME MANAGER

.....

Le Game Manager contient tout ce qui va constituer, de manière général, les règles du jeu.

Il contient les fonctions s'occupant:

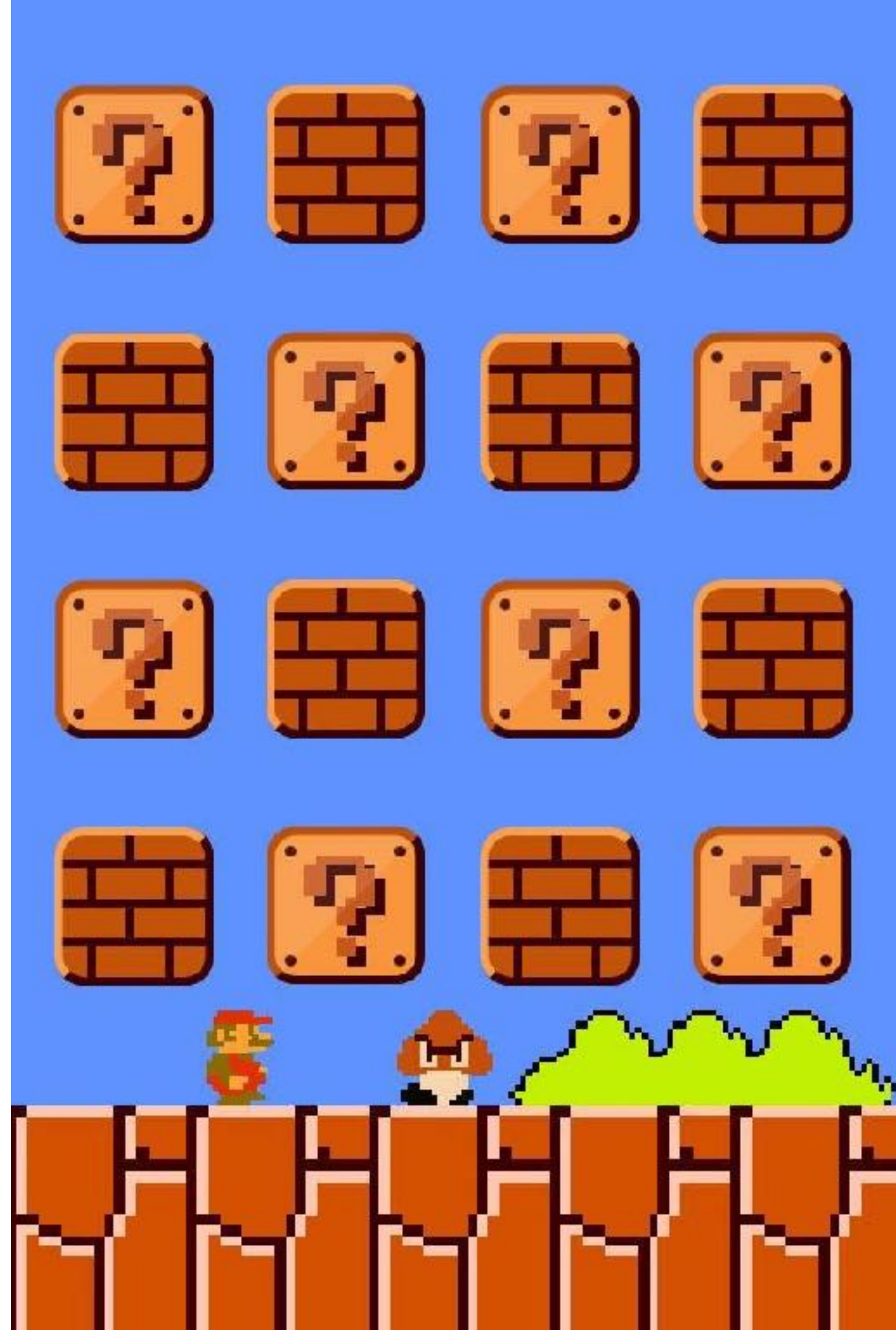
- de la création du plateau (cases vides et cases inaccessibles),
- de la creation des joueurs et des armes.
- du positionnement aléatoire des éléments,
- le déclenchement du combat,
- le déroulement du combat et les règles qui s'y appliquent,
- la gestion du tour par tour.

EVENT MANAGER

L'Event Manager regroupe l'ensemble des événements du programme ainsi que ceux qu'ils vont engendrer.

Cela inclus:

- La possibilité de mouvement,
- le déplacement effectif des joueurs (en vérifiant l'accessibilité de la case suivante)
- passer son tour lorsque que le joueur a terminé,
- si la case du joueur contient une arme et, si cela est le cas, échanger les armes,
- enclencher le compteur à chaque mouvement.





DISPLAY MANAGER

Le Display Manager va se charger de la partie visuel du programme.

Cela inclus:

- le plateau physique,
- les images joueurs et armes,
- les encadrement biographique,
- les overlays,
- L'outline currentPlayer,
- L'accessibilité des boutons d'actions,
- le rafraichissement de la page après chaque action sur la page.

INDEX DES DEFINITIONS

.....

- Une expression de classe est un moyen de définir une classe avec ECMAScript 2015 (ES6). Semblable aux expressions de fonctions, les expressions de classes peuvent être nommées ou anonymes.
- La méthode constructor: est une méthode qui est utilisée pour créer et initialiser un objet lorsqu'on utilise le mot clé class.
- Le mot-clé extends: est utilisé dans les déclarations et expressions de classes afin de signifier qu'un type représenté par une classe hérite d'un autre type.
- La méthode get: permet de lier une propriété d'un objet à une fonction qui sera appelée lorsqu'on accédera à la propriété.
- L'affectation par décomposition est une expression JavaScript qui permet d'extraire des données d'un tableau ou d'un objet grâce à une syntaxe dont la forme ressemble à la structure du tableau ou de l'objet.
- La proposition de décomposition des propriétés et de la syntaxe du reste dans ECMAScript ajoute la syntaxe du reste pour la
- décomposition. La propriété du reste permet de collecter les propriétés énumérables restantes qui n'auraient pas été extraites par la décomposition ({...}) .
- Les fonctions fléchées permettent de raccourcir la syntaxe des fonction (de const = fonction () { } à const = () => { })
- Les modules JS: un module contient une classe ou des librairies de fonctions.
- La méthode querySelector() retourne le premier élément qui correspond à un selector CSS spécifique (ou plusieurs) dans le document.
- Les méthodes setAttribute() / setProperty() ajoutent un attribut spécifique à un élément, et lui donne une valeur spécifique. Si l'attribut/propriété (du même nom) spécifié existe déjà, seule la valeur sera ajoutée/changée.

- La méthode append() insert un contenu spécifique à la fin de l'élément sélectionné (ou des éléments sélectionnés).
- La méthode forEach() appelle une fonction une fois pour chaque élément d'un Array, dans l'ordre.
- La méthode getElementById() retourne l'élément qui possède l'ID demandé avec sa valeur.
- La méthode push() ajoute un nouvel élément à la fin d'un array (tableau).
- La méthode jQuery Event: déclenche ou attache une fonction à un gestionnaire d'événement pour les éléments sélectionnés.
- La méthode addEventListener() attache un gestionnaire d'événement à l'élément spécifié.
- Le mot-clé NEW permet de créer un nouvel objet.
- La propriété keyCode retourne le caractère Unicode correspondant à la touche pressée lors d'un événement onkeypress.
- Find: la méthode find() retourne les descendants de l'élément sélectionné.
- Filter: la méthode filter() cherche des éléments spécifiques. Il effectue une recherche non sensible à la casse.

LES CONSIGNES

.....

Etape 1 : génération de la carte

- ☑ Commencez par générer aléatoirement la carte du jeu. Chaque case peut être soit : Vide ou Inaccessible
- ☑ Sur la carte, un nombre limité d'armes (4 maximum) sera placé aléatoirement et pourra être récolté par les joueurs qui passeraient dessus.
- ☑ Vous inventerez au moins 4 types d'arme dans le jeu, avec des dégâts différents. L'arme par défaut qui équipe les joueurs doit infliger 10 points de dégâts. Chaque arme a un nom et un visuel associé.
- ☑ Le placement des deux joueurs est lui aussi aléatoire sur la carte au chargement de la partie. Ils ne doivent pas se toucher (ils ne peuvent pas être côte à côte).

Etape 2 : les mouvements

- ☑ A chaque tour, un joueur peut se déplacer d'une à trois cases (horizontalement ou verticalement) avant de terminer son tour. Il ne peut évidemment pas passer à travers un obstacle.
- ☑ Si un joueur passe sur une case contenant une arme, il laisse son arme actuelle sur place et la remplace par la nouvelle.

Etape 3 : le combat !

Si les joueurs se croisent sur des cases adjacentes (horizontalement ou verticalement), un combat à mort s'engage. Lors d'un combat, le fonctionnement du jeu est le suivant :

- ☑ Chacun attaque à son tour
- ☑ Les dégâts infligés dépendent de l'arme possédée par le joueur
- ☑ Le joueur peut choisir d'attaquer ou de se défendre contre le prochain coup
- ☑ Lorsque le joueur se défend, il encaisse 50% de dégâts en moins qu'en temps normal
- ☑ Dès que les points de vie d'un joueur (initialement à 100) tombent à 0 , celui-ci a perdu. Un message s'affiche et la partie est terminée.