

JEU DE COMBAT

LE PRINCIPE

Il s'agit d'un jeu qui se joue à autant de joueurs que l'on veut (plus on est de fous, plus on rit !)

Le principe est simple :

- on se crée et on paramètre un personnage (on lui donne un pseudo et un mot de passe pour pouvoir se connecter à ce personnage et en prendre le contrôle)
- il doit alors entrer sur le terrain de jeu, se battre ou se défendre, avancer dans toutes les pièces pour atteindre la sortie
- au cours des combats, on perd ou on gagne des points de vie. Si on les perd tous, le personnage meurt (fin de la partie)

Le terrain de jeu est composé de 9 pièces successives, numérotée de 1 à 9 :

Entrée	1	2	3	4	5	6	7	8	9	Sortie
--------	---	---	---	---	---	---	---	---	---	--------

Les zones entrées et sorties sont neutres : on ne peut pas y combattre, ni y voir les autres personnages présents.

Le jeu comporte 3 pages

- La création d'un personnage
- La connexion (saisie du pseudo et du mot de passe d'un personnage) pour contrôler un personnage
- Puis l'écran principal de jeu

Le choix d'un personnage consiste à paramétrer le personnage : son pseudo (unique), son mot de passe, et ses caractéristiques (voir plus loin).

L'écran principal du jeu indique :

- le rappel des caractéristiques du personnage
- le numéro de la pièce où il se trouve
- un bouton pour avancer dans la pièce suivante (si les caractéristiques le permettent)
- un bouton pour revenir à la pièce précédente
- de manière générale, des boutons pour les actions possible
- la liste des autres personnages se trouvant dans la pièce (cette liste est cliquable, il suffit de cliquer sur un personnage pour l'attaquer). On ne voit pas les caractéristiques des autres joueurs, et on ne voit pas les autres pièces.
- l'historique des actions effectuées et subies (attaques effectuées, attaques subies, mouvements effectués), et leur impact sur les 4 paramètres du personnage (voir ci-après : points de vie, force, agilité, résistance). Des couleurs, éventuellement des pictos, permettent de visualiser d'un seul coup d'œil les différents événements, en particulier les attaques subies.

Les informations qui changent sont mises à jour automatiquement toutes les demi-secondes.

CHOIX D'UN PERSONNAGE

Le choix d'un personnage consiste à :

- choisir un pseudo (visible par les joueurs dans la même pièce) et un mot de passe
- acquérir 100 points de vie automatiquement
- répartir 15 points entre les 3 propriétés suivantes :
 - force (force utilisée pour les attaques)
 - agilité (pour les déplacements et les esquives)
 - résistance (en défense)

A tout moment, en fonction de ses propriétés et des règles du jeu, le joueur a accès aux actions suivantes (donc les actions ne sont visibles que si elles sont possibles)

- avancer dans la pièce suivante
- reculer
- attaquer un personnage dans la même pièce (en cliquant sur le personnage)
- transformer un point de force en point de résistance ou réciproquement.

TRANSFORMER UN POINT (FORCE OU RÉSISTANCE)

La transformation d'un point de force en point de résistance, ou réciproquement, consomme 3 points d'agilité.

Cela ne permet pas de dépasser 15 points de force ou 15 points de résistance.

DÉPLACEMENTS

Déplacement vers l'avant :

Un déplacement vers l'avant (dans la pièce suivante ou la zone de sortie) consomme des points d'agilité : il faut autant de points d'agilité que le numéro de la pièce à atteindre, et 10 points pour passer dans la zone de sortie.

Si on n'a pas assez de points d'agilité, on n'a pas accès à la pièce suivante

Déplacement vers l'arrière :

Un déplacement vers l'arrière (dans la pièce précédente ou la zone d'entrée) est toujours possible et ne consomme pas de points d'agilité. On peut le faire même avec zéro point d'agilité.

On gagne alors en points de vie le numéro de la pièce atteinte.

RESTER DANS UNE PIÈCE SANS RIEN FAIRE

Si on reste dans une pièce sans rien faire (sans attaquer) pendant 3 secondes, on récupère 1 point d'agilité.

Cela ne s'applique pas à la zone d'entrée.

Attention : les points d'agilité sont plafonnés à 15 !

ATTAQUER

Lorsqu'on clique sur un personnage (donc présent dans la même pièce), cela signifie qu'on l'attaque.

L'attaque est alors automatique, et se fait avec une force déterminée (cette force d'attaque est utilisée pour déterminer le déroulement du combat, voir chapitre suivant) : la force de l'attaque est la force de l'attaquant.

Si l'adversaire esquivé et que l'on a 10 points de force ou plus, un point de force devient un point de résistance.

Si on gagne le combat, on récupère un point d'agilité (ça motive !), ou un point de vie si on a déjà 15 points d'agilité.

Si on gagne le combat et que en plus l'on tue l'adversaire, on récupère en plus les points de vie qui lui restaient juste avant le combat.

Si on perd le combat : on perd 1 point de vie.

SUBIR UNE ATTAQUE

Lorsque l'on subit une attaque d'une certaine force, on se défend, voire on riposte.

Cette opération est automatique (et apparaît dans l'historique de l'utilisateur).

Si notre agilité dépasse la force d'attaque d'au moins 3 points, on esquivé. Personne n'a alors gagné ou perdu le combat, et on perd 1 point d'agilité.

Si notre force est supérieure strictement à celle de l'attaque, on riposte : voir ci-après la riposte. On gagne le combat et un point de vie si on gagne la riposte, on perd le combat et 2 points de vie si on perd la riposte.

Sinon, on se défend : si notre résistance est supérieure ou égale à la force de l'attaque, on gagne le combat, si elle est inférieure, on le perd et on perd en points de vie la différence entre notre résistance et la force de l'attaque.

RIPOSTE

La riposte fonctionne comme une attaque, mais est déclenchée automatiquement.

On attaque alors notre attaquant avec toute notre force disponible, et notre attaquant applique les règles "subir une attaque".

ORGANISATION DU TRAVAIL

Cette exercice se fait par équipes de 3, avec les séquences de travail suivantes :

- analyse individuelle du schéma d'ergonomie, puis mise en commun des réflexions pour aboutir à un schéma commun
- analyse individuelle du modèle de données, puis mise en commun des réflexions pour aboutir à un modèle commun
- création du modèle physique et de la table
- spécification des contrôleurs, des templates principaux, et des objets du modèle de données (les méthodes de base, les méthodes spécifiques) : pour chaque objet, on a donc un catalogue de ses méthodes disponibles
- lorsque le besoin d'une fonction ou méthode (PHP ou Javascript), d'un template, d'un contrôleur, se fait sentir, définition en commun du nom, du rôle, des paramètres et du retour
- Réflexion individuelle du déroulement général (les grandes étapes), puis mise en commun
- codage individuel puis mise en commun et conservation du meilleur code