

TP3 - Préparation du côté serveur avec ExpressJs

9 octobre 2015

1 Objectifs

Le but de ce travail pratique est de vous familiariser avec CSS3. Au cours du TP1, vous avez eu l'occasion d'élaborer la structure des pages à l'aide de HTML5. Au cours du TP2, vous allez élaborer la mise en forme de votre site web avec CSS3.

2 Introduction

Le livre-jeu est un genre né de la saga Donjons et Dragons des années 70. Ce sont des romans interactifs, où le déroulement de l'histoire dépend des choix du lecteur et de la chance. L'auteur Joe Dever a gracieusement accepté de publier un bon nombre de ses livres jeu sur internet. Pour le laboratoire, nous nous inspirerons du troisième livre de la série Lone Wolf : Les Grottes de Kalte ou The Caverns of Kalte. Vous trouverez une version web de ce jeu à l'endroit suivant : <http://www.projectaon.org/en/xhtml1/lw/03tcok/title.htm>. Une version française du livre est également disponible à l'endroit suivant : <http://www.scribd.com/doc/127964396/Loup-Solitaire-03-Les-Grottes-de-Kalte>.

3 Principes de fonctionnement

Nous présentons dans cette section les principes de fonctionnement du jeu, afin de vous fournir une vision globale de ce que vous devrez implémenter tout au long du trimestre. Mais toute la logique et les calculs derrière ces principes ne seront pas implémentés avant le TP3.

Lorsqu'on débute le jeu, trois valeurs sont générées aléatoirement. La première, que nous appellerons points d'HABILETÉ, comprise entre 10 et 19, détermine l'HABILETÉ au combat. La deuxième représente les points d'ENDURANCE et est comprise entre 20 et 29. Lorsqu'un joueur est blessé dans un combat, il perd des points d'ENDURANCE.

Lorsque la valeur sera à zéro, il sera considéré mort et son aventure sera donc terminée. Dans certaines situations, il sera possible de récupérer des points d'ENDURANCE, mais on ne pourra jamais dépasser la valeur initiale. La troisième valeur, comprise entre 10 et 19, indique le nombre de Pièces d'Or que possède le joueur au départ. À noter qu'à aucun moment le joueur ne pourra posséder sur lui plus de 50 Pièces d'Or.

3.1 Disciplines Kai

Le joueur débute avec le rang d'Initié, ce qui signifie qu'il peut maîtriser 5 disciplines parmi les 10 disponibles :

Le Camouflage : Cette technique permet au Seigneur Kaï de se fondre dans le paysage.

À la campagne, il peut se cacher parmi les arbres et les rochers et se rendre de cette façon invisible à l'ennemi même s'il passe tout près de lui. Dans une ville, cette Discipline donnera à celui qui l'utilise la faculté d'avoir l'air d'un habitant du cru, tant par l'apparence que par l'accent ou la langue employée. On peut ainsi trouver un abri où se cacher en toute sécurité.

La Chasse : Cette Discipline donne au Seigneur Kaï l'assurance qu'il ne mourra jamais de faim, même s'il se trouve dans un environnement hostile. Il aura toujours la possibilité de chasser pour se procurer de la nourriture, sauf dans les déserts et autres régions arides. Vous devez savoir que la Vassagonie est un empire situé au cœur du désert. Dans ce pays aride, les occasions de faire bonne chasse seront peut-être rares, pour ne pas dire inexistantes. Cette technique, cependant, se révèlera très utile, car elle permet à un Seigneur Kaï de se déplacer très vite et avec une grande agilité.

Sixième sens : Grâce à cette technique, le Seigneur Kaï devine les dangers imminents qui le menacent. Ce Sixième Sens peut également lui révéler les intentions véritables d'un inconnu ou la nature d'un objet étrange rencontré au cours d'une aventure.

L'Orientaion : Chaque fois qu'il se trouvera dans l'obligation de décider quelle direction il doit prendre, le Seigneur Kaï fera toujours le bon choix grâce à cette technique. Il saura ainsi quel chemin il convient d'emprunter dans une forêt et il pourra également, dans une ville, découvrir l'endroit où est caché une personne ou un objet. Par ailleurs, il saura interpréter chaque trace de pas, chaque empreinte qui pourrait lui permettre de remonter une piste.

La Guérison : Cette Discipline donne la faculté de récupérer des points d'ENDURANCE perdus lors d'un combat. Si vous maîtrisez cette technique, vous pourrez ajouter 1 point d'ENDURANCE à votre total chaque fois qu'il vous sera possible d'aller d'un bout à l'autre d'un paragraphe sans avoir à combattre un ennemi. (Vous n'aurez le droit d'utiliser cette technique de la Guérison que lorsque vos points d'ENDURANCE seront tombés en dessous de votre total initial. Rappelez-vous que vos points d'ENDURANCE ne peuvent en aucun cas excéder votre total de départ.)

La Maîtrise des Armes : En entrant au monastère Kaï, chaque élève a la possibilité d'être initié au maniement d'une arme. Si vous choisissez d'avoir la Maîtrise d'une arme, choisissez un chiffre aléatoire entre 0 et 9 pour obtenir un chiffre qui correspondra, dans la liste ci-dessous, à l'arme dont on vous aura enseigné le maniement. Vous aurez dès lors la parfaite Maîtrise de cette arme et, chaque fois que vous combattrez avec elle, vous aurez droit à 2 points d'HABILETÉ supplémentaires. Le fait que vous ayez la Maîtrise d'une arme ne signifie nullement que vous disposerez de cette arme dès le début de votre aventure ; au cours de celle-ci, cependant, l'occasion vous sera donnée de vous la procurer. Vous pourrez d'ailleurs acquérir diverses armes au cours de votre quête mais vous n'aurez pas le droit d'en posséder plus de 2 à la fois.

0 : Le poignard

1 : La lance

2 : La masse d'armes

3 : Le sabre

4 : Le marteau de guerre

5 : L'épée

6 : La hâche

7 : L'épée

8 : Le bâton

9 : Le glaive

Bouclier Psychique : Certaines créatures malfaisantes de Magnamund ont la faculté de vous porter atteinte en faisant usage de leur force psychique. La technique du Bouclier Psychique vous permet cependant de ne pas perdre de points d'ENDURANCE lorsque vous vous trouvez soumis à une telle agression.

Puissance Psychique : Cette technique permet au Seigneur Kaï d'attaquer un ennemi en se servant de la force de son esprit : on peut l'utiliser en même temps qu'une arme de combat habituelle et l'on dispose alors de 2 points supplémentaires d'HABILETÉ. Cette Puissance Psychique, cependant, n'est pas forcément efficace avec toutes les créatures : il se peut que certaines d'entre elles y soient insensibles. Si le cas se présente, vous en serez averti au cours de votre mission.

Communication Animale : Grâce à cette technique, un Seigneur Kaï peut communiquer avec certains animaux et deviner les intentions de certains autres.

Maîtrise Psychique de la Matière : C'est une technique qui donne à un Seigneur Kaï la faculté de déplacer de petits objets par le simple pouvoir de sa concentration mentale.

3.2 Équipement

Finalement, avant de commencer l'aventure, il doit monter son équipement, jusqu'à un maximum de deux items. Le joueur ne peut jamais porter plus de deux armes, et son sac à dos ne peut contenir plus de huit objets incluant les repas (à noter que les Pièces d'Or ne sont pas dans le sac à dos, mais dans une bourse portée à la ceinture). Vous avez le droit à tout moment d'échanger l'un de ces objets ou tout simplement de vous en débarrasser, mais il vous est interdit de le faire lorsque vous êtes engagé dans un combat. Les objets spéciaux ne sont pas rangés dans le sac à dos, donc ils ne prennent pas d'espaces. Il choisit deux items parmi les items suivants :

Épée : (Arme).

Sabre : (Arme).

Lance : (Arme).

Masse d'armes : (Arme).

Marteau de guerre : (Arme).

Hâche : (Arme).

Bâton : (Arme).

Glaive : (Arme).

Gilet de cuir matelassé : (Objet spécial). Ce gilet donne deux points d'ENDURANCE supplémentaires.

Potion de Lampsur : (Objet contenu dans le sac à dos). Cette potion vous rendra 4 points d'ENDURANCE lorsque vous la boirez à la fin d'un combat. Le flacon que l'on vous donne n'en contient qu'une seule dose.

Rations spéciales : (Objet contenu dans le sac à dos). Elles comptent pour un repas.

3.3 Combats

Durant l'aventure, le joueur peut être impliqué dans plusieurs combats. À chaque fois, les points d'HABILETÉ et d'ENDURANCE de l'ennemi seront spécifiés. Le joueur devra alors s'efforcer de tuer son adversaire en réduisant à zéro ses points d'ENDURANCE, tout en essayant lui-même de perdre le moins possible de points d'ENDURANCE. Chaque combat se déroule de la manière suivante :

1. Ajouter au total d'HABILETÉ du joueurs les points supplémentaires associés à certaines disciplines.
2. Un nombre aléatoire entre 0 et 9 est choisi. En utilisant ce nombre et le quotient d'attaque, on consulte la table de correspondance illustrée aux figures 1a et 1b, afin de déterminer le nombre de points d'ENDURANCE qui seront perdus par le joueur et l'ennemi.

3. Soustraire de ce total les points d'HABILETÉ de l'adversaire. Ce résultat représente le quotient d'attaque.
4. Un nombre aléatoire entre 0 et 9 est choisi. En utilisant ce nombre et le quotient d'attaque, on consulte la table de correspondance illustrée aux figures 1a et 1b, afin de déterminer le nombre de points d'ENDURANCE qui seront perdus par le joueur et l'ennemi.
5. On diminue les points d'ENDURANCE du joueur et de l'ennemi.
6. À moins qu'une possibilité de fuite soit offerte ou que des indications différentes soient fournies, on recommence un nouveau round de combat, en retournant à l'étape 3. On continue ainsi jusqu'à ce qu'un des deux combattant soit mort.

Si le joueur a la possibilité de fuir, il peut le faire à la fin d'un round, après avoir diminué ses points d'ENDURANCE.

Combat Ratio

	-11 or less	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0
Random Number	1 0 K	0 K	0 8	0 8	0 6	1 6	2 5	3 5	4 4	5 4	6 3	7 2
2	0 K	0 8	0 7	1 6	2 5	3 5	4 4	5 4	6 3	7 2	8 1	9 0
3	0 8	0 7	1 6	2 5	3 5	4 4	5 4	6 3	7 2	8 1	9 0	10 0
4	0 8	1 7	2 6	3 5	4 4	5 4	6 3	7 2	8 1	9 0	10 0	11 0
5	1 7	2 6	3 5	4 4	5 4	6 3	7 2	8 1	9 0	10 0	11 0	12 0
6	2 6	3 5	4 4	5 4	6 3	7 2	8 1	9 0	10 0	11 0	12 0	13 0
7	3 5	4 4	5 4	6 3	7 2	8 1	9 0	10 0	11 0	12 0	13 0	14 0
8	4 4	5 4	6 3	7 2	8 1	9 0	10 0	11 0	12 0	13 0	14 0	15 0
9	5 3	6 3	7 2	8 1	9 0	10 0	11 0	12 0	13 0	14 0	15 0	16 0
0	6 0	7 0	8 0	9 0	10 0	11 0	12 0	13 0	14 0	15 0	16 0	17 0

ENDURANCE LOSS ENEMY LONE WOLF

K = AUTOMATICALLY KILLED

(a)

Combat Ratio

	0	1	2	3	4	5	6	7	8	9	10	11 or greater
Random Number	1 3 5	4 5	5 4	6 4	7 4	8 3	9 3	10 2	11 2	12 2	13 1	14 1
2	4 4	5 4	6 3	7 3	8 3	9 2	10 2	11 2	12 1	13 1	14 0	15 0
3	5 4	6 3	7 3	8 2	9 2	10 2	11 1	12 1	13 0	14 0	15 0	16 0
4	6 3	7 3	8 2	9 2	10 2	11 1	12 1	13 0	14 0	15 0	16 0	17 0
5	7 2	8 2	9 2	10 1	11 1	12 0	13 0	14 0	15 0	16 0	17 0	18 0
6	8 2	9 2	10 1	11 1	12 0	13 0	14 0	15 0	16 0	17 0	18 0	19 0
7	9 1	10 1	11 1	12 0	13 0	14 0	15 0	16 0	17 0	18 0	19 0	20 0
8	10 0	11 0	12 0	13 0	14 0	15 0	16 0	17 0	18 0	19 0	20 0	21 0
9	11 0	12 0	13 0	14 0	15 0	16 0	17 0	18 0	19 0	20 0	21 0	22 0
0	12 0	13 0	14 0	15 0	16 0	17 0	18 0	19 0	20 0	21 0	22 0	23 0

ENDURANCE LOSS ENEMY LONE WOLF

K = AUTOMATICALLY KILLED

(b)

FIGURE 1 – Tables pour calculer pertes d'ENDURANCE

D'autres règles du livre-jeu original, comme les niveaux d'entraînement et les rangs \titre accordés aux seigneurs Kaï, ne seront pas utilisées dans ce travail.

L'aventure se déroule comme suit. Le joueur commence au paragraphe 1 du livre. À certains moments, il peut avoir des choix à faire, qui l'emmèneront à un autre paragraphe du livre. À d'autres moments, il sera engagé dans des combats, qu'il devra mener en suivant les règles citées précédemment.

4 Travail à réaliser

Dans ce TP3, nous allons préparer le côté serveur fournir des fonctionnalités au client. Toutes les fonctions nécessaires qui seront utilisées par le client seront implémentées au TP3 et TP4. Cependant, le TP4 va créer les fonctions qui vont interagir avec la base de données.

Quelques fonctionnalités à implémenter nécessitent l'utilisation d'une base de données. Cependant, nous allons utiliser les sessions sur ExpressJs pour sauvegarder les données. Nous allons utiliser le module *express-session* de ExpressJs (<https://www.npmjs.com/package/express-session>). Pour l'installer, NE PAS faire *npm install express-session*. Je vous demande de l'insérer dans package.json avec la version 1.11.3. Ensuite, faire *npm install*. Pour l'utiliser, un exemple se trouve dans le bas de la page du lien.

Dans ce travail, nous allons parler de Service Web. Dans les travaux précédents, votre site web envoyait du HTML au client avec l'instruction *res.render('page')*. Dans un service web, seul du JSON est envoyé au client (pour ceux qui connaissent xml, c'est un autre format de données, mais beaucoup plus verbose que JSON). L'envoi de JSON se fait par la commande *res.json('votre json dans une chaîne de caractère')*. Voici un exemple d'un JSON <http://www.w3schools.com/json/>. Vous êtes libre de définir votre propre structure.

Dans ce travail, les Services Web suivent la philosophie REST (Michel vous en parlera davantage dans le cours). Il faut donc faire attention aux différences entre GET, POST, PUT et DELETE. Cette distinction sera plus évidente dans le TP4.

Au cours du TP3 et TP4, le côté client de votre site web n'offrira pas beaucoup d'interactions à l'utilisateur. Les interactions seront implémentées au TP5.

Voici les directives de ce travail pratique :

- Dans votre page de création de joueur, le formulaire enverra une requête POST vers une route. Cette route s'occupera de sauvegarder un objet Javascript Joueur (que vous créerez) et qui sera sauvegarder dans la Session de Express. (Au TP4, le joueur sera sauvegarder dans une base de données, mais en attendant...). Vous devrez générer des chiffres aléatoires pour ses statistiques.
- Dans la route du point précédent, vous devrez ajouter un nouveau champ du joueur qui correspond à ses points d'habiletés calculés en fonction de bonus de ses disciplines (maîtrise d'arme + arme). Il faut faire de même pour les points d'endurance (gilet de cuir donne 2 vie de plus).
- Dans la route POST de la directive précédente, vous devez effectuer la validation des données reçues. Exactement 5 disciplines et 2 objets pourront être choisis. Si les conditions ne sont pas remplies, alors on réaffiche la page de création avec un message d'erreur inséré dans le Jade. Si les conditions sont remplies, on passe à la page 1 de l'histoire.
- Dans un fichier du serveur, vous devez créer des objets Javascript qui contiennent les noms des disciplines et des objets, et de les utiliser dans chaque fichier jade.

Cette démarche nous permettra de ne pas manipuler des chaînes de caractères afin d'uniformer les données et d'éviter les erreurs de frappe. Par exemple : `var disciplines = { CAMOUFLAGE : "camouflage", CHASSE : "chasse", ... }` (avec saut de ligne). Ensuite, on pourra l'appeler avec `disciplines.CAMOUFLAGE`.

- Vous devez un service web qui envoie le JSON du joueur.
- Dans certaines pages, vous remarquerez qu'on retrouve certaines mise en situation où on demande une entrée de l'utilisateur (confirmation de perte de points, ajouter/jeter des objets, combat, etc.) Afin de pouvoir les implémenter pour le TP6, vous devrez séparer chaque page de l'histoire en plusieurs pages jade. Par exemple, si j'insère l'URL `/page/78/1`, je devrais voir UNIQUEMENT le HTML de l'histoire de la page 78 jusqu'au combat. L'URL `/page/78/2` devrait me ramener le reste de la page, e.g. la prise de décision. Vous devez le faire pour chaque page implémentée au cours des derniers travaux pratiques.
- Une fois que ces changements ont été effectués, nous allons procéder à la création d'un service web. L'URL `/page/:id` ne retournera plus le HTML de la page, mais bien le JSON qui représente la page. Le JSON peut contenir l'id de la page (id : 78), le HTML de chaque section d'une page (HTML de `/page/78/1`), les informations sur le combat s'il y a lieu, et les numéros de page qu'on peut accéder. Pour chaque numéro de page potentielle, vous pouvez ajouter davantage d'informations telles que la discipline/objet requis pour accéder à la page.
- Nous savons que plusieurs pages demandent à l'utilisateur de faire un calcul selon un chiffre aléatoire. Selon le chiffre obtenu, le joueur sera obligé d'aller vers un chemin particulier dans le jeu. Dans le serveur, vous allez créer un tableau d'objets Javascript qui contient l'ensemble des pages qui contiennent de telles décisions. Chaque objet contient l'id de la page, la fonction à appliquer, et l'intervalle de chaque page qu'on peut accéder. Afin d'éviter de dupliquer les données, je vous conseille de faire appel à l'objet de pages créer précédemment.
- Selon le point précédent, vous devez créer un Service Web qui applique la fonction et qui retourne en JSON le choix selon l'intervalle. Je devrais pouvoir accéder à travers une route telle que `/choixAleatoire/167`. Les pages qui nécessitent les points d'endurance du joueur pour trouver le chiffre ne seront pas implémentées dans le TP3.
- Pour devez implémenter la logique des combats sur le serveur. Pour se faire, vous devez créer un service web. Ce service web ne maintiendra pas d'état sur le déroulement du combat. Il sera de forme `/combat/20/19` où le chiffre 20 représente les points d'habileté du joueur et le chiffre 19 représente les points d'habileté de l'ennemi. Si j'accède à cette URL, le service web me retournera un JSON qui contient le quotient d'attaque, le chiffre aléatoire, endurance perdu du joueur et l'endurance perdu de l'ennemi.
- À partir de maintenant, j'évalue la qualité de votre code. Votre code doit être clair et lisible. Vous devez vous entendre sur une norme et la respecter. Je ne veux pas du code mal indenté, rempli d'espaces inutiles avec des blocs de codes en commentaires. Je m'attends également que vous ajoutiez des commentaires sur vos

fonctions, objets, etc. La sémantique de vos champs JSON devront être décrits à moins qu'ils soient trop évidents.

Les Services Web pour modifier l'état du joueur seront implémentés au TP5 avec la base de données.

5 Évaluation

Globalement, vous serez évalué la récupération/validation du formulaire et sur les services web créés. Plus précisément, le barème d'évaluation est le suivant :

Exigences	Points
Récupération et validation du formulaire	10
Service web des pages du jeu	30
Calcul des combats	20
Choix aléatoire	20
Qualité du code et aspect globale	20

Le fureteur **Firefox** sera utilisé pour tester votre site web.

6 Directions de la remise

1. Le dossier *node_modules* doit être supprimer pour la remise. De plus l'ensemble de votre projet devrait contenir seulement les fichiers nécessaires. Par exemple, je ne veux pas voir des fichiers avec un tilt.
2. Placer votre projet Express dans un dossier compressé nommé : TP3_matricule1_matricule2.
3. Le travail pratique doit être remis avant 23h55 le dimanche 25 octobre 2014 sur Moodle.

7 Pénalités

1. La politique pour les travaux pratiques remis en retard : -20% pour chaque jour de retard.
2. Le travail doit être remis en équipe de 2. Une remise individuelle aura la note 0.

Bonne chance !