

TP5 - Services Webs

18 novembre 2013

1 Objectif

Le but de ce travail pratique est de vous familiariser avec les Services Web, et de l'utilisation du framework AngularJS.

Vous implémenterez votre propre Service Web qui communiquera avec le client à l'aide de JSON. AngularJS s'occupera de séparer l'application avec le patron MVC (Modèle-Vue-Contrôleur) et il mettra à jour la vue selon le modèle; le modèle sera le service web et le localStorage.

2 Introduction

Au cours des TP's précédents, vous avez eu l'occasion de créer un site web complet pour votre application livre-jeu *Lone Wolf : Castle Death*.

Votre application est complète, et elle a été implémentée côté serveur au cours du TP4. Dans ce TP5, nous allons proposer une amélioration au TP4 en ajoutant un Service Web et le framework AngularJS.

Nous allons communiquer avec un Service Web pour recevoir les données des pages de jeu. La communication entre le service web et le client se fera à l'aide du format JSON.

La librairie AngularJS est un framework qui implémente le patron MVC. Il permet d'efficacement séparer la vue (le HTML), le contrôleur (le Javascript), et le modèle (BD, Service Web, localStorage, etc.) Il a été conçu par Google pour des SPA (Single Page Application).

3 Principes de fonctionnement

Cette section est la même que le premier travail pratique. Elle fournit la vision globale de ce que vous devrez implémenter tout au long du trimestre.

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 (le surplus pourra être déposé dans un coffre-fort). Une fois que ces trois valeurs ont été générées, le joueur doit choisir trois disciplines parmi les 10 disponibles :

La science des armes : Permet de manier toutes sortes d'armes. Il faudra en choisir trois parmi celles-ci : le poignard, la lance, la masse d'armes, le marteau de guerre, la hache, l'épée, le bâton et le glaive. Par contre, au cours de l'aventure, il ne sera pas possible de posséder plus de deux armes à la fois. Chaque fois que le joueur combat avec une de ces armes, il gagne trois points d'endurance supplémentaires.

Contrôle animal : Permet de communiquer avec certains animaux ou deviner leurs intentions.

Science médicale : Permet d'ajouter 1 point d'endurance à chaque fois que le joueur a parcouru un paragraphe complet sans avoir à combattre un ennemi. Cette discipline permet aussi de soigner certaines maladies, de guérir la cécité et toutes sortes de blessures, aussi bien celles reçues que celles infligées. Elle fait également connaître les propriétés des plantes, des herbes, des racines et des potions.

Invisibilité : Permet de devenir invisible. Elle donne la possibilité de masquer la chaleur et l'odeur du corps, d'adopter les moeurs et de parler le dialecte de n'importe quelle ville visitée.

Art de la chasse : Avec cette discipline, le joueur pourra chasser pour se procurer de la nourriture (même dans le désert ou une région aride). Elle permet aussi de se déplacer très vite et avec agilité, évitant ainsi de perdre des points d'habileté lors d'une attaque surprise ou d'une embuscade.

Exploration : Permet de toujours choisir le bon chemin, de lire les langues étrangères, décrypter les symboles, reconnaître les empreintes et autres traces (même si elles ont été brouillées) et déceler la présence de la plupart des pièges. Elle permet aussi de repérer intuitivement le nord.

Foudroiement psychique : Permet d'attaquer un ennemi par la seule force de l'esprit. On peut l'utiliser en même temps qu'une arme de combat ordinaire et l'on dispose alors de 4 points supplémentaires d'habileté. À chaque utilisation du foudroiement psychique, le joueur perd 2 points d'endurance. Une forme atténuée, appelée *puissance psychique*, peut être utilisée contre un ennemi sans perdre des points d'endurance, mais elle ne procure que 2 points d'habileté supplémentaires. Le foudroiement psychique ne peut pas être utilisé si le total d'endurance est égal ou inférieur à 6.

Écran psychique : Permet de ne pas perdre de points d'endurance lorsqu'on est attaqué par un créature faisant usage de force psychique. Elle augmente les moyens de défense contre

l'hypnose et les illusions surnaturelles.

Nexus : Permet de supporter la chaleur et le froid sans perdre de points d'endurance. Permet aussi de déplacer des objets par le biais de la pensée.

Intuition : Permet de deviner les dangers imminents, les intentions d'un inconnu et la nature des objets étranges. Permet aussi de communiquer par télépathie.

Finalement, avant de commencer l'aventure, il doit monter son équipement, jusqu'à un maximum de cinq items. Le joueur ne peut jamais porter plus de deux armes, et son sac à dos ne peut contenir plus de huit objets (à noter que les pièces d'or ne sont pas dans le sac à dos, mais dans une bourse portée à la ceinture). Il choisit parmi les items suivants :

Épée : (Arme). On la porte à la main.

Arc : (Arme). On la porte à la main.

Carquois : (Objet spécial) Contient six flèches. On le porte en bandoulière.

Corde : (Objet contenu dans le sac à dos).

Potion de Laumspur : (Objet contenu dans le sac à dos) Permet de reprendre 4 points d'endurance si on la boit à la fin d'un combat.

Poignard : (Arme). On le porte à la main.

Lanterne : (Objet contenu dans le sac à dos).

Masse d'armes : (Arme). On la porte à la main.

Trois rations spéciales : (Repas) Chaque ration représente un repas et occupe la place d'un objet dans le sac à dos.

Trois graines de feu : (Objet spécial) Lorsqu'on jette une de ces graines sur le feu, elle explose et brûle d'un feu vif. On les met dans la poche.

À tout moment dans le jeu, sauf dans un combat, le joueur peut échanger des objets qu'il possède ou s'en débarrasser. Les règles d'utilisation des objets sont les suivantes :

Armes Pour pouvoir les utiliser, il faut avoir choisi la discipline de la science des armes. Dans ce cas, si le joueur dispose d'une arme en bon état et qu'il est impliqué dans un combat avec un ennemi, il pourra ajouter 3 points d'habileté à son total. Un combat sans arme fait perdre 4 points d'habileté.

Arcs et flèches La flèche permet de frapper les ennemis à distance. Il faut un carquois pour contenir les flèches. Chaque fois qu'une flèche est utilisée, elle n'est plus récupérable. On ne peut pas utiliser les flèches pour les combats au corps à corps. Nous verrons plus loin que le lancer d'une flèche implique un nombre choisi aléatoirement. Si le joueur possède la science des armes dans ses disciplines, il ajoutera 3 points à cette valeur.

Pièces d'or Chaque fois que le joueur tue un ennemi et qu'il décide de fouiller son cadavre, il peut conserver les pièces d'or qu'il trouve.

Nourriture Si le joueur se trouve dans une situation où il doit prendre un repas et qu'il n'a aucune ration, il perdra 3 points d'endurance, sauf s'il maîtrise la discipline de la chasse.

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 joueur les points supplémentaires associés à certaines disciplines.
2. Soustraire de ce total les points d'habileté de l'adversaire. Ce résultat représente le quotient d'attaque.
3. 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.
4. On diminue les points d'endurance du joueur et de l'ennemi.
5. 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 combattants 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.

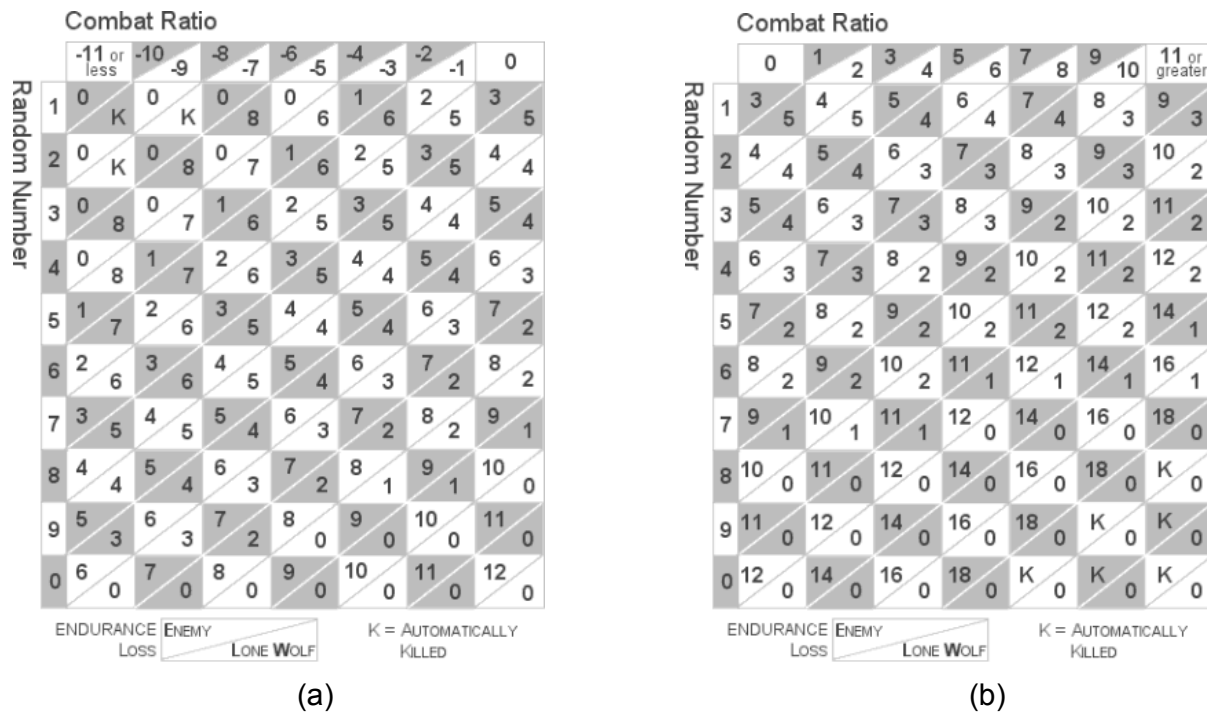


FIGURE 1 - Tables pour calculer pertes d'endurance

D'autres règles du livre-jeu original, comme les niveaux d'entraînement et les cercles de savoir Magnakaï, 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

Vous partirez selon l'implémentation que vous avez fait au cours du TP4. Vous aurez également la possibilité d'utiliser la correction fournie pour effectuer le TP5.

Le travail à réaliser consiste à utiliser AngularJS pour implémenter le MVC dans les pages d'histoire du livre.

Cette fois-ci, lorsqu'on se trouve dans une page de l'histoire, aucune requête ne sera fait pour recharger une nouvelle page de Jeu. La mise à jour du contenu de la page va se faire à l'aide de AJAX d'AngularJS. Si nous sommes à la première page de Jeu et que l'utilisateur demande la prochaine page, une requête AJAX sera fait au service web qui s'occupera d'envoyer le bon contenu de la page à l'utilisateur. Le contenu correspondra à la structure de donnée que vous avez utilisé au TP4. Le service web représentera le modèle du MVC d'AngularJS.

Pour la sauvegarde des données du joueur, nous utiliserons le `localStorage` tel que le TP3. Le `localStorage` représentera le deuxième modèle du MVC d'AngularJS. C'est ainsi que les données du joueur seront à jour.

Voici une séquence possible de l'application:

- 1- L'utilisateur arrive à la page d'Accueil.
- 2- Il décide de créer son personnage à travers le formulaire de la page d'initialisation et il clique sur «Commencer».
- 3- Yesod s'occupera d'envoyer les données du formulaire à la page numéro 1 du jeu à l'aide d'une requête POST.
- 4- Le handler POST de la page de jeu recevra la requête contenu les données du joueur et la page numéro 1 sera créer avec les bonnes données.
- 5- L'utilisateur décidera d'accéder à la prochaine page (par exemple, la page 135). Une fois qu'il clique sur cette page, une requête sera fait au service web qui s'occupera de lui envoyer les données de la prochaine page en format JSON; le JSON contiendra des informations tel que le numéro de la page voulu, le contenu de la page voulu et les autres pages accessibles à partir de cette page. Cette requête sera une requête AJAX; il n'y aura pas de recharges complètes de la page, seul le contenu de la page sera modifié.
- 6- Ainsi de suite jusqu'à la fin...

Voici les directives de ce travail pratique :

- Création d'un service web envoyant du JSON à l'utilisateur. Le service web résidera dans l'URL de votre choix (par exemple, `/service...`).
- Intégration de AngularJS respectant le modèle MVC.
- Mise à jour du contenu de la page de jeu sans recharger la page.
- Mise à jour des statistiques du joueur en utilisant le `localStorage`.
- Respect **général** des directives des travaux pratiques précédents.

5 Évaluation

Globalement, vous serez évalués sur la qualité de votre code/structure du service web et de AngularJS dans le gabarit Yesod / Haskell. Plus précisément, le barème d'évaluation est le suivant :

Exigences	Points
Création du service web envoyant du JSON	20
Intégration de AngularJS respectant le modèle MVC.	20
Mise à jour du contenu de la page de jeu sans recharger la page.	20
Mise à jour des statistiques du joueur en utilisant le localStorage avec AngularJS.	10
Respect des directives globales du livre-jeu.	10
Qualité et appréciation globale	10

6 Direction de la remise

1. Pour la remise, on demande de compresser votre dossier Yesod et le nommer : TP5_*matricule1_matricule2*. Vous avez le choix de compression.
2. Vous devrez également créer un fichier nommé *temps.txt* à l'intérieur du dossier de votre projet. Vous indiquerez le temps passé **au total** pour ce TP.
3. Vous verrez un **champ texte** lors de la remise sur Moodle, où vous spécifierez les **noms** et **matricules** des membres de l'équipe.
4. Le travail pratique doit être remis **avant 23h55 le dimanche 1 décembre 2013**.

7 Pénalités

1. La politique pour les travaux pratiques remis en retard: -20% pour chaque jour de retard. Bonne chance!