

INFO7 – Projet : duel de heros

- Site angular : <https://angular.io>
- Tutoriel : <https://angular.io/docs/ts/latest/tutorial/>

Présentation

L'éditeur de jeux « Bizzard » aimerait que vous réalisiez un éditeur de troll pour son futur jeu en ligne « duel de trolls ». Sur votre conseil, cet éditeur sera une application client riche réalisée à l'aide d'Angular.

La version minimale de cette application sera composée de cinq fenêtres de travail :

- Un dashboard pour voir les héros les plus « marquants » :
 - afficher les détails d'un héros si l'utilisateur le souhaite
 - rediriger vers la fenêtre d'édition d'un héros ;
- Une fenêtre listant tous les héros présents, elle permet de :
 - afficher les détails d'un héros si l'utilisateur le souhaite
 - rediriger vers la fenêtre d'édition d'un héros ;
 - créer un nouvel héros (redirection version la fenêtre d'édition) ;
- Une fenêtre listant toutes les armes disponibles, elle permet de :
 - afficher les détails d'une arme si l'utilisateur le souhaite ;
 - rediriger vers la fenêtre d'édition d'une arme ;
 - créer une nouvelle arme (redirection version la fenêtre d'édition) ;
- Une fenêtre d'édition d'un héros avec répartition de ses points de compétence et association avec une arme ;
- Une fenêtre d'édition d'une arme avec répartition de ses points de compétence.

Etape 1 : Tour of heroes

Objectif : manipuler les principales fonctionnalités d'angular : component, service, route, etc.

Finir le tutoriel Hero Editor proposé par le site angular.io. Ce tutoriel sera la base à votre application.

Etape 2 : Améliorer les héros

Objectif : comprendre le lien automatique entre le model et la vue.

Améliorer les héros pour prendre en compte les demandes de la société :

Un héros a 4 caractéristiques : *attaque*, *esquive*, *dégâts*, *point de vie (pv)*. Une base de 40 points est à répartir entre ces caractéristiques. Exemple de répartition pour un héros nommé *Bayard* : 15pt en attaque, 5pt en esquive, 15pt en dégâts, 5pt en points de vie.

ATTENTION CONTRAINTES concernant l'édition d'un héros :

- La somme de l'attaque, esquive, dégât et pv doit être inférieur ou égale à 40 ;
- Aucune caractéristique en dessous de 1pt ;
- Afficher les points restant à répartir pour aider l'utilisateur.

Etape 3 : Ajouter les armes

Objectif : manipuler à nouveau les notions de service, route et classe.

Ajouter la possibilité de créer des armes. Le concept est identique aux héros : une classe, un service associé, etc. Prenez exemple sur les héros pour réaliser l'édition des armes.

Une arme a les mêmes caractéristiques qu'un héros : *attaque*, *esquive*, *dégâts*, *point de vie (pv)*. Mais la répartition est différente !!! Exemple de répartition pour une arme nommée *épée de la finesse* : 0pt en attaque, -5pt en esquive, 5pt en dégâts, 0pt en points de vie.

ATTENTION CONTRAINTES concernant l'édition d'une arme:

- La somme de l'attaque, esquive, dégât et pv doit être égale à 0 ;
- Une caractéristique est limitée entre -5pt et 5pt (inclus) ;
- Afficher les points restant à répartir pour aider l'utilisateur.

Etape 4 : Lier les armes aux héros

Objectif : manipulation des classes et api-memory.

Un héros peut être équipé d'une arme (ou non). Prendre en compte ce lien et modifier l'éditeur en conséquence.

Etape 5 : Filtrer/Trier les héros/armes

Objectif : comprendre le lien automatique entre le modèle et la vue.

Les utilisateurs souhaitent pouvoir filtrer/trier les héros/armes en fonction de leurs caractéristiques. Améliorer les fenêtres listant tous les héros et toutes les armes pour prendre en compte ce besoin.

Etape 6 : Développer plus pour gagner plus

A vous de laisser libre cours à votre imagination ! Associer une image à héros, Intégration du jeu ... soyez inventif !