420-5W5

Projet

Super Cartes Infinies (aka Super HeartStone Infini)

Date de remise : Semaine 6

# L’application

## **Objectifs** :

Créer un application web à l’aide d’Angular, Web API et MVC pour faire un jeu de carte.

Le site Angular sera utilisé par des membres inscrits.

L’administrateur pourra configurer le contenu de l’application à l’aide de MVC.

## Les règles :

* Deux joueurs s’affrontent tour à tour et joue une carte chacun par tour.
* Chaque carte a une certaine quantité de points d’attaque et une certaine quantité de points de défense.
* Après avoir joué une carte, toutes les cartes présente du jouer attaque une carte de l’adversaire.
* Chaque attaque diminue les points défense de la carte ciblée jusqu’à ce que la carte n’ait plus de points de défense.
* Chaque joueur à un certain nombre de points de vies, lorsque qu’un joueur n’a pas de carte en jeu pour se défendre, il reçoit des dégâts qui font descendre ses points de vie.
* Le premier joueur à ne plus avoir ne points de vie perds la partie.

# Le détail des sections

## Section d’administration

* L’administrateur peut créer, modifier, voir et supprimer une carte.
* L’administrateur peut modifier les cartes de départ des nouveaux joueurs.

## Carte

* Une carte doit avoir une valeur d’attaque et une valeur en défense.
* Les cartes peuvent être triés pour être plus faciles à repérer.

## Utilisateur

* Un nouvel utilisateur doit obtenir un paquet de carte de départ.

## Partie

* Un utilisateur peut jouer contre un autre utilisateur présent.
* Chaque utilisateur doit jouer une carte par tour.
* Chaque carte en jeu doit attaquer une autre carte à chaque tour et ainsi enlever des points de points de défense à l’autre carte.
* Une carte n’ayant plus de points de défense doit être retirée du jeu.
* Si l’adversaire n’a plus de carte en jeu, il se fera attaquer par la carte et il perdra des points de vie.
* Chaque action des utilisateurs doit être envoyée au serveur pour que l’autre joueur puisse récupérer l’action effectuée.

# Contraintes

* Le travail doit être effectué en équipes de 4.
* Vous devrez utiliser Git.
* Vous devrez utiliser DevOps pour la gestion des tâches.
  + Vous devrez lier DevOps et Git pour qu’on puisse voir les commits dans le suivie des tâches.
* Vous devrez déployer votre application sur Azure.
* Vous devrez faire une application cliente en Angular.
* Vous devrez faire une application serveur en ASP.NET Web API.
* Vous devrez faire une section d’administrateur sur votre serveur en MVC.
* Vous devrez faire des tests unitaires pour l’utilisation des services.
* Vous devrez ajuster les tests unitaires du « gameplay » pour qu’il fonctionne.