TP NoSQL

## Contexte

La « gamification », ou « ludification » est l’application de mécaniques issues du monde ludique dans un autre contexte, pour susciter l’acceptation, l’adhésion et l’engagement des utilisateurs.

Par exemple, l’introduction d’un système de micro-challenges dans une formation professionnelle, l’utilisation d’un classement par contribution dans un forum…

Par groupes de 4 ou 5, vous devrez créer un service de gamification en mode SaaS (***S****oftware* ***A****s* ***A******S****ervice*), c’est-à-dire utilisable au travers d’API (***A****pplication* ***P****rogramming* ***I****nterface*) REST (***RE****presentational* ***S****tate* ***T****ransfert*).

Ce service proposera devra également proposer une interface Web de consultation.

## Description du service

Il doit permettre à différents jeux de soumettre des requêtes indiquant l’évolution de leurs joueurs (identifiés par des pseudonymes).

Ces requêtes permettrons, par exemple, d’attribuer un badge à un joueur, de comptabiliser un score, un nombre d’heure de jeu, un nombre de parties…

Les joueurs peuvent ensuite venir consulter leurs résultats individuels sur l’interface Web du service, et avoir par jeu leurs badges, les badges qu’il leur reste à découvrir, leur score…  
Ainsi qu’un certain nombre de classements, par jeu, et globaux.

## Déroulement du TP

Vous serez libre de choisir le langage de programmation, les librairies, les framework, le système NoSQL les plus appropriés à la réalisation de ce service.

Les seules contraintes sont :

1. L’utilisation d’un système NoSQL pour le stockage des informations
2. Le respect de l’API fournie dans ce document
3. La mise à disposition d’une interface Web de consultation

Pour vous aider, un programme vous sera fourni pour générer aléatoirement des requêtes REST sur votre service, et fournir ainsi des données.

Vous aurez 1h à 1h30 par séance consacrées à ce TP. Lors du dernier cours, vous ferez une rapide présentation (fonctionnelle et technique) de votre service, à l’ensemble du groupe.

La notation prendra en compte la présentation, l’étude du code du service, et de son comportement.

## API

### Attribution d’un badge

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Verbe | URI | Paramètre | Corps | Retour |
| PUT | http://host:port/stats/(:game) | game: nom du jeu concerné | Objet JSON contenant :  - player : login du joueur  - badge : id du badge ajouté | 204 |

Exemple:

PUT <http://localhost:80/stats/race> {player: ‘john’, badge: ‘10K\_miles’}