Utopic Village

Spécification des web-services

2012

Guillaume

Exod Creation

11/06/2012

# Sommaire

Table des matières

[Sommaire 1](#_Toc327287492)

[Introduction 2](#_Toc327287493)

[WebServices 3](#_Toc327287494)

# Introduction

Ce document contient un descriptif de tous les web-services nécessaires au bon fonctionnement de l’application. Ces web-services servent à la communication entre le serveur web (la base de données) et l’application mobile.

L’intégralité de web-service renverra des flux JSON. J’utiliserai ce format car il est facile à mettre en œuvre (natif en php, il est possible de sérialiser un objet directement en flux JSON) et il est bien plus léger que les autres types de flux.

Les flux renvoyés sont des objets JSON généré automatiquement à partir de l’entité correspondante.

# WebServices

|  |  |  |  |
| --- | --- | --- | --- |
| Données retournées | url du webservice | Description | Paramètre(s) |
| Array JSON d’HELP | json/nearHelps | Retourne les n plus proches demande d’aide | Latitude, longitude (double) |
| JSON HELP | json/help | Retourne l’intégralité des informations d’une demande d’aide | Id (int) |
| JSON status | json/insertHelp | Insert une demande d’aide dans la base de données | Id(int), montant(int), texte(String) |
| Array JSON HELP | json/yourHelps | Retourne toutes vos demandes d’aide | Id (int) |
| Array JSON HELP | json/yourHelpsActives | Retourne toutes vos demandes d’aide encore valides | Id (int) |
| Array JSON HELP | json/yourParticipations | Retourne toutes les aides que vous avez fournit | Id (int) |
| JSON USER | json/profil | Retourne les informations d’un joueur | Id (int), position (boolean) |
| Array JSON MESSAGE | json/youMessages | Retourne les messages qui vous sont destinés | Id (int) |
| Array JSON MESSAGE | json/yourNewMessages | Retour tous les nouveaux messages | Id (int) |
| JSON status | json/sendMessage | Envoie d’un message à un autre joueur | Id (int), texte(String) |
| Array JSON USER | json/volunteers | Renvoie la liste des volontaires pour aider (sur une aide précise) | id (int) |
| Array JSON USER | json/profilsBySearch | Renvoie la liste des joueurs pouvant correspondre à la recherche | %recherche% (String) |
| JSON status | Json/confirmParticipation | Le joueur est volontaire pour cette demande d’aide | idPlayer(int),idHelp(int) |
| Array JSON Feats | json/yourFeats | Renvoie tous les exploits déjà réalisés | Id (int) |
| Array JSON Feats | json/allFeats.json | Renvoie tous les exploits disponibles |  |
| JSON status | json/confirmModified | Envoi les modifications du compte au serveur | Id (int), nom, prenom, email, login, password(String), date de naissance(Date) |
| - | json/connexion | Signaler au serveur que le joueur se connecte (lancement durant le splatch screen) | Id (int) |
| JSON status | json/reportPlayer | Signaler un joueur | Id (int) |
| JSON status | json/reportHelp | Signaler une annonce | Id (int) |
| JSON status | json/testConnect | Tester le login et mot de passe, ainsi que les sanctions en cours et encore le compte active | Login (String), password (String) |
| JSON status | Json/insertJoueur | Insertion d’un nouveau joueur | Login, password, name, firstname, email (String), birthdate (Date) |
| JSON status | Json/savePostition | Sauvegarde de la position pour l’user selectionné | Id(int), lat, long(double) |

JSON Status = {“Status” : ???}

JSON HELP = {toutes les propriétés de l’entity HELP}

JSON MESSAGE = {toutes les propriétés de l’entity MESSAGE}

JSON USER = {toutes les propriétés de l’entity USER}

Les webservices seront par la suite mieux documenté avec les fichiers d’exemples pour chaques requêtes