

## **SERVICE CREATE USER (CREATION D'UN NOUVEL UTILISATEUR)**

**Nom du web servlet :** servlets.users.CreateUser\_

**Nom du web service :** services.users.CreateUserService

**URL du web service :** TWISTER/user/CreateUser.

**Descriptions du web service :** Sert à créer un nouvel utilisateur dans la base de données.

**Paramètres en entrée :** login (le login de l'utilisateur), password (le mot de passe de l'utilisateur), nom (le nom de l'utilisateur), prénom (le prénom de l'utilisateur).

**Format de sortie :** JSON

**Exemple de sortie :** { « Création Utilisateur» : « OK »}.

**Erreur Possible :**

- Paramètres vides.
- Login déjà existant.
- SQL exception.
- JSON exception.

**Avancement du service :** TERMINE

**Classes JAVA en rapport avec le web service :**

- Servlets.users.CreateUser.java
- Services.friends.CreateUserService.java
- Tools.ToolUser.java
- Services.ErrorJSON.java
- Toutes les classes du package bd

**Informations additionnelles :** nan.

## **SERVICE DE LOGIN (CONNEXION D'UN UTILISATEUR)**

**Nom du web servlet :** servlets.users.Login\_

**Nom du web service :** services.users.LoginService

**URL du web service :** TWISTER/user/Login.

**Descriptions du web service :** Sert à ce que l'utilisateur puisse se connecter à son compte.

**Paramètres en entrée :** login (le login de l'utilisateur), password (le mot de passe de l'utilisateur).

**Format de sortie :** JSON

**Exemple de sortie :** { « key » : « 353JGR63090292Y4579J »  
« id » : « root »}.

**Erreur Possible :**

- Paramètres vides.
- Login inexistant.
- Mot de passe incorrect.
- SQL exception.
- JSON exception.

**Avancement du service :** TERMINE

**Classes JAVA en rapport avec le web service :**

- Servlets.users.Login.java
- Services.friends.LoginService.java
- Tools.ToolUser.java
- Services.ErrorJSON.java
- Toutes les classes du package bd

**Informations additionnelles :** La clé est générée aléatoirement et l'utilisateur sera automatiquement déconnecté après un temps définie de connexion.

## **SERVICE DE LOGOUT (DECONNEXION D'UN UTILISATEUR)**

**Nom du web servlet :** servlets.users.Logout\_

**Nom du web service :** services.users.LogoutService

**URL du web service :** TWISTER/user/Logout.

**Descriptions du web service :** Sert à ce que l'utilisateur puisse se déconnecter.

**Paramètres en entrée :** key (la clé de l'utilisateur).

**Format de sortie :** JSON

**Exemple de sortie :** { « Logout » : « OK»}.

**Erreur Possible :**

- Paramètres vides.
- Utilisateur non connecté.
- SQL exception.
- JSON exception.

**Avancement du service :** TERMINE

**Classes JAVA en rapport avec le web service :**

- Servlets.users.Logout.java
- Services.friends.LogoutService.java
- Tools.ToolUser.java
- Services.ErrorJSON.java
- Toutes les classes du package bd

**Informations additionnelles :** nan.

## **SERVICE DELETE USER (SUPPRESSION D'UN UTILISATEUR)**

**Nom du web servlet :** servlets.users.DeleteUser\_

**Nom du web service :** services.users.DeleteUserService

**URL du web service :** TWISTER/user/DeleteUser

**Descriptions du web service :** Sert à supprimer l'utilisateur de la base de données (de la table messages de mongoDB, friends et users de MySQL).

**Paramètres en entrée :** key (la clé de l'utilisateur).

**Format de sortie :** JSON

**Exemple de sortie :** {« Suppression Utilisateur » : « OK »}.

**Erreur Possible :**

- Paramètres vides.
- Utilisateur non connecté.
- SQL exception.
- JSON exception.

**Avancement du service :** TERMINE.

**Classes JAVA en rapport avec le web service :**

- Servlets.users.DeleteUser.java
- Services.users.DeleteUserService.java
- Tools.ToolUser.java
- Tools.ToolFriend.java
- Tools.ToolMessage.java
- Services.ErrorJSON.java
- Toutes les classes du package bd

**Informations additionnelles :** nan.

## SERVICE D'AJOUT D'UN AMI

**Nom du web servlet :** servlets.friends.NewFriend\_

**Nom du web service :** services.friends.NewFriendService

**URL du web service:** TWISTER/friend/NewFriend

**Descriptions du web service :** Sert à ajouter un ami dans la base (relation unidirectionnelle).

**Paramètres en entrée :** key (la clé de la connexion de l'utilisateur), idFriend (l'identifiant de l'ami).

**Format de sortie :** JSON

**Exemple de sortie :** { « Service Ajout Friend » : « OK »}.

**Erreur Possible :** 6 sont erreurs possibles dans ce service :

- Paramètres vides.
- Utilisateur non connecté.
- idFriend non-existant.
- Déjà amis dans la base.
- SQL exception.
- JSON exception.

**Avancement du service :** TERMINE

**Classes JAVA en rapport avec le web service :**

- Servlets.friends.AddFriend.java
- Services.friends.AddFriendService.java
- Tools.ToolFriend.java
- Tools.ToolUser.java
- Services.ErrorJSON.java
- Toutes les classes du package bd

**Informations additionnelles :** nan.

## **SERVICE DE SUPPRESSION D'UN AMI**

**Nom du web servlet :** servlets.friends.DeleteFriend\_

**Nom du web service :** services.friends.DeleteFriendService

**URL du web service :** TWISTER/friend/DeleteFriend

**Descriptions du web service :** Sert à supprimer un ami de la base (suppression unidirectionnelle).

**Paramètres en entrée :** key (la clé de la connexion de l'utilisateur), idFriend (l'identifiant de l'ami).

**Format de sortie :** JSON.

**Exemple de sortie :** { « Service delete friend » : « OK »}.

**Erreur Possible :**

- Paramètres vides.
- Utilisateur non connecté.
- idFriend non-existant.
- Les deux utilisateurs ne sont pas amis dans la base.
- SQL exception.
- JSON exception.

**Avancement du service :** TERMINE

**Classes JAVA en rapport avec le web service :**

- Servlets.friends.DeleteFriend.java
- Services.friends.DeleteFriendService.java
- Tools.ToolFriend.java
- Tools.ToolUser.java
- Services.ErrorJSON.java
- Toutes les classes du package bd

**Information additionnelles :** nan.

## SERVICE DE RECHERCHE D'UN AMI

**Nom du web servlet :** servlets.friends.GetFriend\_

**Nom du web service :** services.friends.GetFriendService

**URL du web service :** TWISTER/friend/GetFriend

**Descriptions du web service :** Sert à rechercher un ami ou des amis dans la base à partir de critères en fonction des paramètres fournis.

**Paramètres en entrée :** key (la clé de la connexion de l'utilisateur), idUtilisateur (l'identifiant de l'utilisateur dont on cherche les amis), login (le login qu'on cherche si on cherche un login), onLine (non null quand on cherche les on line friends), Followers (non null lorsque on cherche les followers d'un user).

**Format de sortie :** JSON

**Exemple de sortie :** {« id\_user » : 9, « login » : « babou », « nom » : « toto », « prenom » : « titi »}

**Erreur Possible :**

- Paramètres vides.
- Utilisateur non connecté.
- SQL exception.
- JSON exception.

**Avancement du service :** TERMINE

**Classes JAVA en rapport avec le web service :**

- Servlets.friends.GetFriend.java
- Services.friends.GetFriendService.java
- Tools.ToolFriend.java
- Tools.ToolUser.java
- Services.ErrorJSON.java
- Toutes les classes du package bd

**Information additionnelles :** nan.

## **SERVICE New Message (ajout d'un message)**

**Nom du web servlet :** servlets.messages.NewMessage

**Nom du webservice:**services.messages.NewMessageService

**URL du web service :** /TWISTER/message/NewMessage

**Descriptions du web service :** Sert à rajouter un message d'un utilisateur dans la base.

**Paramètres en entrée :** key (la clé de connexion de l'utilisateur), message (Le contenu du message à rajouter),login

**Format de sortie :** JSON

**Exemple de sortie :** {"idMessage":"5a9539498c8a520307895aa0"}.

**Erreur Possible :**

- Paramètres vides.
- Utilisateur non connecté.
- Message vide
- SQL exception.
- JSON exception.
- MongoDB Exception.

**Avancement du service :** TERMINE

**Classes JAVA en rapport avec le web service :**

- Servlets.messages.NewMessage.java
- Services.messages.NewMessageService.java
- Tools.ToolMessage.java
- Tools.ToolUser.java
- Services.ErrorJSON.java
- Toutes les classes du package bd

**Information additionnelles :** à chaque ajout d'un message on met à jour l'index pour la recherches avec les mots clés après.



## **SERVICE Delete Message (suppression d'un message)**

**Nom du web servlet :** servlets.messages.DeleteMessage

**Nom du webservice:**services.messages.DeleteMessageService

**URL du web service :** /TWISTER/message/DeleteMessage

**Descriptions du web service :** Sert à supprimer le message d'un utilisateur de la base.

**Paramètres en entrée :** key (la clé de connexion de l'utilisateur), , idMessage (L'id du message dans la base)

**Format de sortie :** JSON

**Exemple de sortie :** {"message supprime": "OK"}.

**Erreur Possible :**

- Paramètres vides.
- Utilisateur non connecté.
- Message non existant dans la base.
- Utilisateur non propriétaire du message.
- SQL exception.
- JSON exception.
- MongoDB Exception.

**Avancement du service :** TERMINE

**Classes JAVA en rapport avec le web service :**

- Servlets.messages.DeleteMessage.java
- Services.messages.DeleteMessageService.java
- Tools.ToolMessage.java
- Tools.ToolUser.java
- Services.ErrorJSON.java
- Toutes les classes du package bd

**Information additionnelles :** nan.

## **SERVICE Get Message (recherche d'un/des message(s))**

**Nom du web servlet :** servlets.messages.GetMessage

**Nom du webservice:**services.messages.GetMessageService

**URL du web service :** /TWISTER/message/GetMessage

**Descriptions du web service :** Sert à chercher des messages dans la base en fonction des critères basés sur les paramètres.

**Paramètres en entrée :** key (la clé de connexion de l'utilisateur), motsCles (les mots clés pour la recherche), idUtilisateurs (une liste d'ids des utilisateurs pour lesquels on cherche les messages),idMin( l'indice dans la liste des messages à partir duquel on va récupérer),nb(nombre des messages à charger).

**Format de sortie :** JSON

**Exemple de sortie :** {"list":[{"message":{"date":"Mon Feb 26 13:05:19 CET 2018","\_id":"5a93f7ff8c8a18768c5cc951","id\_user":15,"comments":[],"contenu":"mess2"}}],"recherche":"OK"}.

**Erreur Possible :**

- Paramètres vides.
- Utilisateur non connecté.
- SQL exception.
- JSON exception.
- MongoDB Exception.

**Avancement du service :** IL RESTE LES FONCTIONS DE RECHERCHE PAR MOTS CLES DANS TOOLMESSAGE.

**Classes JAVA en rapport avec le web service :**

- Servlets.messages.GetMessage.java
- Services.messages.GetMessageService.java
- Tools.ToolMessage.java
- Tools.ToolUser.java
- Services.ErrorJSON.java
- Toutes les classes du package bd

**Information additionnelles :** nan.

## **SERVICE Add Comment (Rajout d'un commentaire)**

**Nom du web servlet :** servlets.comments.AddComment\_

**Nom du web service :** services.comments.AddCommentService

**URL du web service :** /TWISTER/comment/AddComment

**Descriptions du web service :** Sert à rajouter des commentaires sur un message dans la base.

**Paramètres en entrée :** key (la clé de connexion de l'utilisateur), idMessage(l'identifiant du message), comment(le commentaire à rajouter),login,

**Format de sortie :** JSON

**Exemple de sortie :** {« Comment », « c'est un commentaire »}.

**Erreur Possible :**

- Paramètres vides.
- Utilisateur non connecté.
- Message non existant.
- SQL exception.
- JSON exception.
- MongoDB Exception.

**Avancement du service :** TERMINE.

**Classes JAVA en rapport avec le web service :**

- Servlets.comments.AddComment.java
- Services.comments.AddCommentService.java
- Tools.ToolComment.java
- Tools.ToolMessage.java
- Tools.ToolUser.java
- Services.ErrorJSON.java
- Toutes les classes du package bd

**Information additionnelles :** nan.

## **SERVICE Delete Comment (Suppression d'un commentaire)**

**Nom du web servlet :** servlets.comments.DeleteComment\_

**Nom du web service :** services.comments.DeleteCommentService

**URL du web service :** /TWISTER/comment/DeleteComment

**Descriptions du web service :** Sert à supprimer un commentaire sur un message de la base.

**Paramètres en entrée :** key (la clé de connexion de l'utilisateur), idMessage(l'identifiant du message), idCommentaire (l'id du commentaire à supprimer).

**Format de sortie :** JSON

**Exemple de sortie :** {« delete commentaire », « OK »}.

**Erreur Possible :**

- Paramètres vides.
- Utilisateur non connecté.
- Message non existant.
- Commentaire non existant.
- Utilisateur non propriétaire du commentaire.
- SQL exception.
- JSON exception.
- MongoDB Exception.

**Avancement du service :** TERMINE.

**Classes JAVA en rapport avec le web service :**

- Servlets.comments.DeleteComment.java
- Services.comments.DeleteCommentService.java
- Tools.ToolComment.java
- Tools.ToolMessage.java
- Tools.ToolUser.java
- Services.ErrorJSON.java
- Toutes les classes du package bd

**Information additionnelles :** nan.

