

# Technologies du Web - TP 7 : Webservices & AJAX

Dans ce TP, vous allez transformer une application de *chat* pour la moderniser. De statique (Web des années 2000) elle va devenir dynamique (Web des années 2010). N'oubliez pas d'utiliser l'**outil de développement du navigateur** pour connaître les erreurs javascript !

## Préambule - *chat* simple et base de données MySQL

Commencez par **recupérer le code** du “chat simple”, dans le dossier du TP. Téléchargez les fichiers, **ne copiez-collez pas** ! (sinon ça fait foirer des caractères et le code ne marche pas)

Les données (messages du *chat*) manipulées par le programme sont stockées dans une base de données MySQL. Pour faire fonctionner le code, **vous devez d'abord activer votre base de données** sur votre compte SIF :

- rendez-vous sur <http://sif.info-ufr.univ-montp2.fr/>
- connectez-vous à droite; une fois connecté, cliquez sur “mon compte”
- dans votre compte, sous votre nom d'utilisateur, cliquez sur “modifier”
- en bas, dans l'encart “base de données”, cliquez sur “base MySQL”, choisissez “oui” et choisissez un mot de passe
- au bas de la page, cliquez sur “enregistrer”; l'activation peut prendre quelques minutes.

Et voilà, vous avez une base MySQL ! Celle-ci porte votre nom (votre login), et se trouve sur le serveur “venus”. Pour tester la connexion **depuis un terminal en ligne de commandes** (menu des applications, puis “accessoires” et “terminal”) :

```
mysql -h venus -u votrellogin -p
```

Après avoir tapé votre mot de passe puis Entrée, vous devez obtenir l'invite de commandes MySQL :

```
mysql>
```

Vous devez maintenant **initialiser la base de données** pour l'application de *chat*, en créant la table que le programme s'attend à trouver. Pour cela, **utilisez le fichier PHP “init.php”** prévu. Avant de l'exécuter, **vous devrez le modifier** pour remplacer le nom d'utilisateur, le mot de passe et le nom de la base de données par les vôtres (note: pour se connecter à la base de données, le code PHP utilise [PDO](#), une bibliothèque très pratique !).

Vous pouvez vérifier que la table a bien été créée, en utilisant la ligne de commande. Voici un aide mémoire des commandes MySQL :

- `show databases;` // voir les bases de données disponibles (dépend de vos droits)

- `use mabase; // utiliser la base de données "mabase"`
- `show tables; // voir les tables disponibles`
- `show columns from matable; // voir les colonnes de la table "matable"`
- `select * from matable; // voir toutes les données de "matable"`

## Exercice 1 - Webservices

Dans le code du chat simple, deux fichiers PHP sont prévus pour gérer les données : un pour ajouter un message ("add\_msg.php"), et un pour lire les messages existants ("get\_latest\_msg.php"). Vous devez **modifier ces fichiers** pour les transformer en webservices :

- ils doivent retourner des données (si applicable) au format JSON - vous utiliserez [header\(\)](#) pour **spécifier le type de contenu** de la réponse (Content-Type: application/json)
- vous utiliserez [http\\_response\\_code\(\)](#) pour **spécifier le code HTTP** de retour (par exemple 200 en cas de succès, et 500 en cas d'erreur - voir la [liste des codes de retour HTTP](#)).

**Note** : ces deux fichiers accèdent à la base de données, vous devez donc modifier la ligne de connexion de PDO en fonction de vos identifiants, comme dans le fichier "init.php".

## Exercice 2 - Appels AJAX

Maintenant que vos webservices sont écrits, modifiez la page HTML du *chat* pour qu'elle les utilise, et devienne ainsi dynamique. Vous ne devrez **pas utiliser de PHP ailleurs que dans les webservices** ! Commencez donc par renommer "index.php" en "index.html".

Dans "index.html", supprimez le code PHP. Puis **créez un fichier Javascript** (nommé par exemple "chat.js") et liez-le à la page HTML.

C'est maintenant ce fichier Javascript qui devra appeler les webservices.

- Ajoutez un timer qui appelle le service de lecture des messages toutes les 5 secondes, et qui injecte le résultat dans la page afin de rendre les messages visibles à l'écran
- Lorsque l'utilisateur clique sur le bouton "envoyer" pour poster un message, interceptez cet événement et appelez le webservice d'ajout de message

Et voilà ! Votre application est dynamique.