

Projet de MiniT’Chat - II Le Retour

Le projet est une **application Web de T’Chat**, les groupes seront définis *aléatoirement*. Ce dont vous avez besoin, c’est ce que vous avez déjà appris : HTML, CSS, PHP, MySQL et JS !

Vous utiliserez le framework Material Design Light. **Vous devez utiliser le fichier structure index-skeleton.php** qui est disponible à la racine de ce dépôt, vous devez suivre le cahier des charges fonctionnel.

Ce sujet est librement inspiré de OpenClassrooms - TP : un mini-chat !

Cahier des charges fonctionnel (déjà rédigé)

- Version française
- Version anglaise

Structure de la table MySQL

- **ID** (type **INT**) : il nous permettra de savoir dans quel ordre ont été postés les messages. Il faudra le mettre en **auto_increment** pour que les numéros s’écrivent tout seuls, et ne pas oublier de sélectionner “Primaire” (cela dit à MySQL que c’est le champ qui numérote les entrées);
- **pseudo** (type **VARCHAR**) : pensez à indiquer la taille maximale du champ (je vous conseille de mettre le maximum, “255”);
- **message** (type **VARCHAR**) : de même, on indiquera une taille maximale de 255 caractères. Si vous pensez que vos messages seront plus longs, utilisez plutôt le type **TEXT**, beaucoup moins limité.

Caractéristiques

C’est un t’chat, vous devez bien sûr laisser les gens discuter ensemble sur un canal global, **sans connexion avec mot de passe**, permettre aux utilisateurs de choisir un pseudo, les messages sont du texte brut.

Bonus

Niveau 1

- Retenir le pseudo. On doit actuellement saisir à nouveau son pseudo à chaque nouveau message. Comme vous le savez probablement, il est possible en HTML de pré-remplir un champ avec l'attribut `value`. Par exemple :

□

- Remplacez `Alice` par le pseudo du visiteur. Ce pseudo peut être issu d'un cookie par exemple : lorsqu'il poste un message, vous inscrivez son pseudo dans un cookie, ce qui vous permet ensuite de pré-remplir le champ.

Niveau 2

- Le mini-chat ne s'actualise pas automatiquement s'il y a de nouveaux messages. En revanche, ce que vous pouvez facilement faire, c'est proposer un lien « Rafraîchir » qui charge à nouveau la page. Ainsi, s'il y a de nouveaux messages, ils apparaîtront après un clic sur le lien.
- Rafraichir automatiquement la page toute les 30 secondes à l'aide de **JavaScript**.

Niveau 3

- N'afficher que les 10 derniers messages.
- Sauriez-vous trouver un moyen d'afficher les anciens messages ? Bien sûr, les afficher tous d'un coup sur la même page n'est pas une bonne idée. Vous pourriez imaginer un paramètre `$_GET[page]` qui permet de choisir le numéro de page des messages à afficher.

Niveau 4

- Permettre aux utilisateurs d'utiliser des Emojis ! Par exemple : `:smile_cat:` va être remplacé par ``.
- Permettre aux utilisateurs d'écrire en Markdown ! Par exemple :

□

va être remplacé par

□

Niveau 5

- Autoriser d'autres canaux comme des discussions privées.