**Project – REUSEIT**

We are creating a project that looks like a « Stackoverflow ». The most basic use case is a connection of a registered user on the website. He will create a list that can contains all the tasks to do. Then he will create his own tasks and he has an issue : he doesn’t know how to perform his tasks. Instead of losting time and procrastinate, he will ask to a forum how to do that. The users will propose him another plan to succeed in the project realization.

The user will be able to save many time !

The entities are the following :

* A **user**, member of a website OR moderator OR admin. Each user is identified by a unique number and a unique username. He has a photo that his public for other users. The access to his member area requires his user email and a password, both crypted by a bcrypt algorithm. The statu is either 0 « banned », 1 « member », 2 « moderator », 3 « admin » or 4 « super administrateur ».
* A **list**. Each list est linked to a single user. The list has a unique identifier, a nam and a visibility status.The user’s lists of tasks is private, except if he makes them public.
* A **task**. Each task refers to a list which is associated with a certain user. Evey task has a unique id, a name, a description, the id of the list whose belongs it, an importance scale (from 1 to 3), a due date, a reminder date, a status.
* (later on) A **category of forum** identified by an id (unique), a name (unique) and has a description.
* A **forum**, which is identified by its name (unique), a unique ID, a description and a status (accessible, archived, deleted). Administrators and super user only can choose to disable or reactivate the access to the forum. Moderators, administrators, and super user are also members, that have some inherited privileges.
* **Un sujet** qui aura un identifiant unique, un nom, un nombre de messages, une date de création, un auteur. Un administrateur et modérateur peuvent supprimer un sujet.

Les sujets auront également un statut (effacé, existant), afin de pouvoir garder les archives. Les sujets auront pour chacune une adresse IP pour pouvoir identifier l’utilisateur. Le topic aura un statut (épinglé, non épinglé). Un sujet est associé à plusieurs messages. Un sujet peut être rattaché à une tâche ou zéro.

* **Un message**, qui est rattaché à un sujet, lui-même rattaché à un forum. Les messages seront dans l’ordre de la date. Un sujet ne comprendra que 25 messages par page. Un utilisateur ne pourra pas envoyer plus d’un message toutes les 10 secondes. Les messages ont comme caractéristique des pseudos, des dates, un contenu. Un message peut être accédé par une ancre. Un message a un corps qui ne peut pas contenir du HTML mais des balises et des smileys. Un message peut avoir des balises pour écrire le code d’un projet PHP. Le premier message d’un sujet peut avoir l’utilité de poser une question au forum afin d’obtenir des réponses. Les autres utilisateurs peuvent réagir et proposer des solutions afin de faire progresser l’utilisateur. Un administrateur et modérateur peuvent supprimer un message.
* **Un ensemble de messages privés**. Les messages privés ont un émetteur et un destinataire. Chaque message a une adresse privée et a un statut supprimé ou non, du côté de la personne qui émet le message et du côté de la personne qui le reçoit. Les messages privés ont une structure identique à un message classique mais ne peuvent être modérés par l’administration. Cependant, un utilisateur peut signaler les messages à l’administrateur.
* **Un CRUD où les administrateurs pourront gérer tout le site**. Un administrateur peut bannir un membre, rechercher les données d’un membre par une adresse IP etc. Un addministrateur peut afficher les statiques secrètes du forum. En plus de cela, l’administrateur a le droit d’effectuer les tâches des modérateurs, à savoir supprimer des sujets/messages etc.