

# Projet Scripts-library

En tant qu'administrateur système, vous utilisez régulièrement des scripts permettant d'automatiser des tâches, qu'ils soient créés par vos soins, ou trouvés sur Internet. Vous êtes également amené à créer/modifier des fichiers de configuration.

Après quelques années d'expérience, vous en avez créé quelques-uns... que vous ne voudriez pas perdre, compte tenu du travail effectué.

Vos collègues du service sont dans le même cas que vous.

Vous décidez à plusieurs de créer une application web qui permettra de recenser et de sauvegarder les scripts créés par les différents intervenants.

## Résumé

|                                    |                                                                                                                                                                        |
|------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Outils à utiliser</b>           | <ul style="list-style-type: none"><li>• <a href="#">Ubiquity</a></li><li>• <a href="#">Semantic-UI</a></li><li>• <a href="#">phpMv-UI</a></li></ul>                    |
| <b>Principales fonctionnalités</b> | <ul style="list-style-type: none"><li>• Gestion des scripts (Ajout, modification, suppression),</li><li>• recherche,</li><li>• gestion des droits</li></ul>            |
| <b>Livraison</b>                   | <ul style="list-style-type: none"><li>• Jusqu'au jour de la soutenance (Semaine à déterminer), par gitHub, ou par Moodle (en fonction de votre chargé de TP)</li></ul> |

## Ressources

- [Ubiquity guide](#)
- [Ubiquity api](#)
- [Twig documentation](#)

- **Lire impérativement les [Modalités de remise de votre travail](#)**
- Pour vous préparer à l'oral, consulter la [grille d'évaluation](#)

## Règles de gestion

Les utilisateurs peuvent créer un compte, et se connecter à l'application.

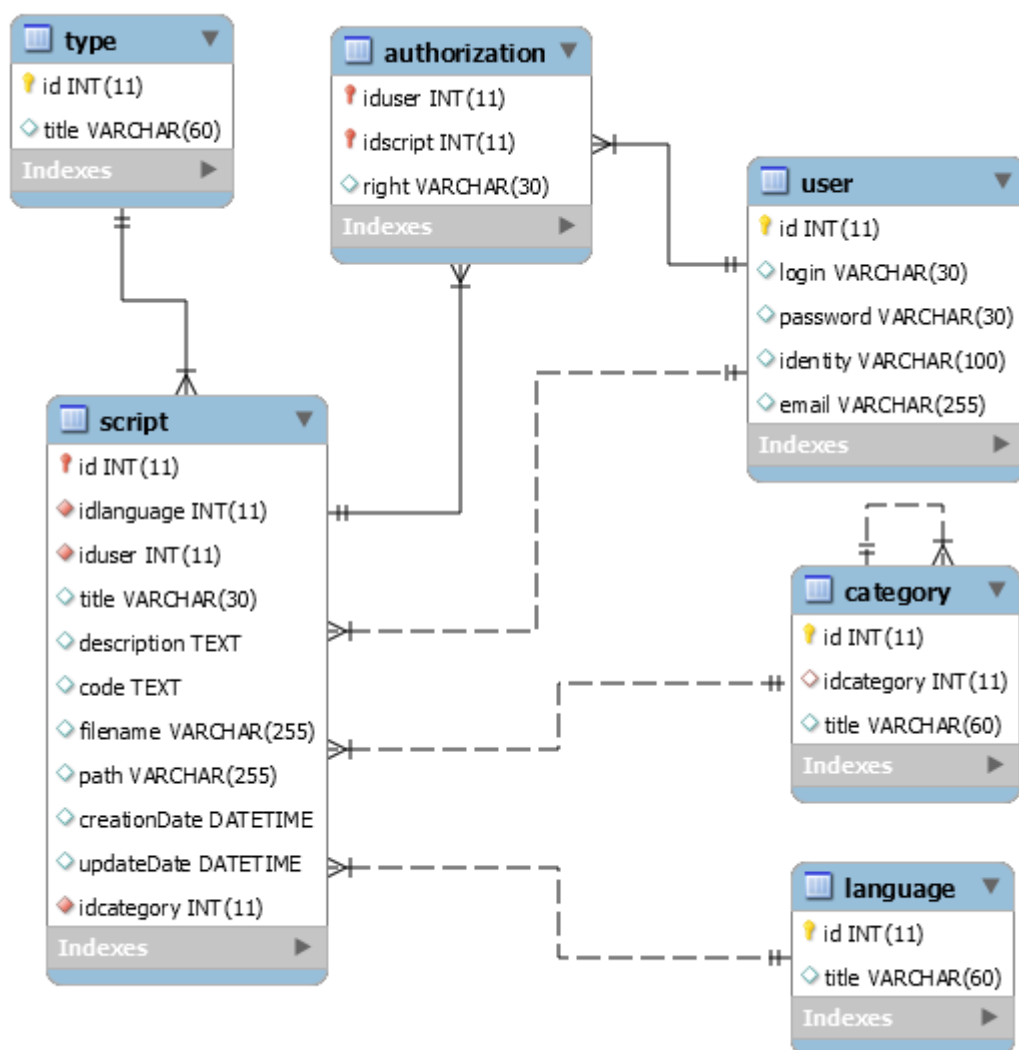
Un script :

- est créé par un utilisateur
- Appartient à une catégorie (config files, bash scripts...)
- est écrit dans un langage (php, bash, python...)
- est d'un certain type (Apache config file, DNS file, CRON...)

Une catégorie peut avoir une catégorie parent : les catégories sont donc structurées de manière arborescente. Les scripts sont classés par catégorie.

Comme avec un système d'exploitation, le propriétaire d'un script peut en modifier les droits :

## Schéma de la base



## Détail des fonctionnalités à mettre en place

- Connexion à l'application
- Affichage des scripts de l'utilisateur, éventuellement classés par catégorie
- Affichage des scripts partagés avec l'utilisateur (sur lesquels il a des droits), éventuellement classés par catégorie
- Parcours par catégories
- Recherche de script
- Affichage des détails d'un script
- Ajout/modification d'un script
- Attribution de droits sur un script
- Gestion des accès (prise en compte des droits)

From:  
<http://slamwiki2.kobject.net/> - SlamWiki 2.1

Permanent link:  
<http://slamwiki2.kobject.net/php-rt/projets/projet-2018?rev=1524465050>

Last update: 2019/08/31 14:26



