

# Documentation



## Introduction :

LIVEQUESTION est un mini réseau social qui permet de poser des questions ouvertes en ligne avec d'autres utilisateurs.

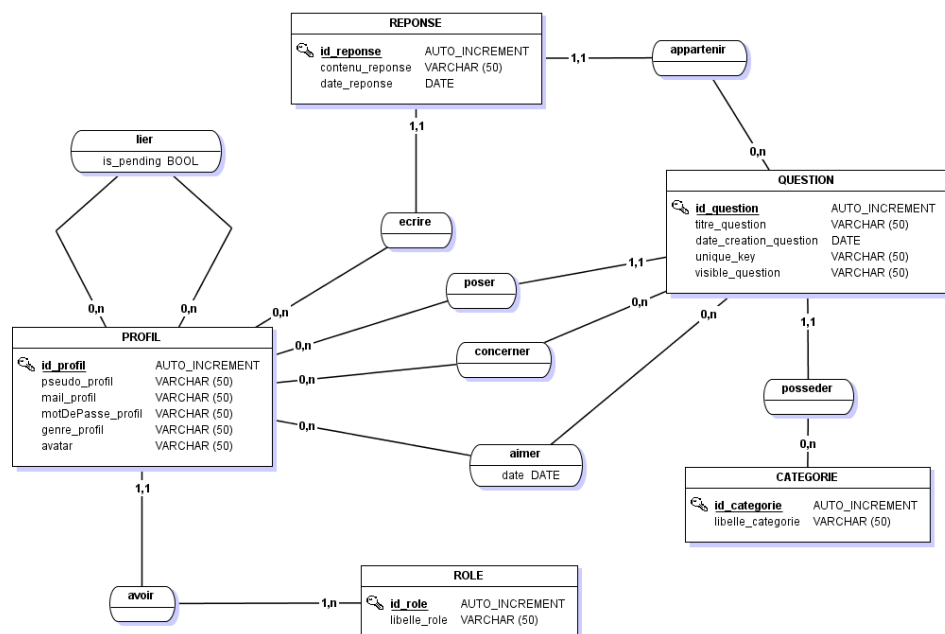
Le cahier des charges du projet LIVEQUESTION comprend :

- Une page d'accueil : le menu principal contient : une barre de navigation, une vue sur les questions posées, une FAQ.
- Une page d'administration : Une fois connecté, l'administrateur a accès à la page qui lui permet soit de modifier, soit de supprimer des questions.
- Une page d'inscription : L'utilisateur devra renseigner son pseudo, son adresse mail, son genre et son mot de passe.
- Un formulaire de connexion : Le bouton en haut à droite de la barre de navigation, permettra à l'utilisateur de se connecter en renseignant le nom d'utilisateur et le mot de passe.
- Une page d'ajout de question : L'utilisateur pourra poser une question (255 caractères maximum)

## Partie 1 : Conception

I)

→ MEA



## → Recueil du besoin

Le client « Mozart du Web » est par C. Espargellière.

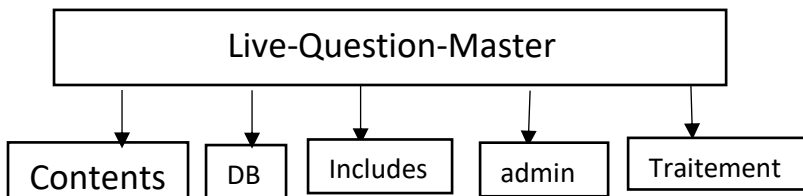
Son besoin est un site interactif de questions-réponses dans lequel il pourra poser des question-réponses en ligne.

II)

## → Les fonctionnalités :

- Base de données
- Gestion des profils
- Connexion
- Gestion des questions

## → Arborescence de l'application



# Partie 2 : Technique

I)

## → BDR

- Script implémentation (requêtes SQL sans les infos)

Question : CREATE \* TABLE 'Question'

Réponse : CREATE \* TABLE 'Reponse'

## → Extrait de code (commenté) = Page d'inscription

```
<?php
// Connexion et requête dans la base de données
require_once('../DB/request.php');

// fonction qui permet le traitement de l'inscription
function PreTraitement(array $informations){
    $erreurs = [];
    if (empty($informations['user'])) {
        $erreurs['user'] = 'Veuillez saisir un nom d\'utilisateur';
    }
    if (empty($informations['email'])) {
        $erreurs['email'] = 'Veuillez saisir votre email';
    }
    if ($informations['genre'] != 'M' && $informations['genre'] != 'F') {
        $erreurs['genre'] = 'Page corrompue veuillez recharger la page';
    }
    if (empty($informations['mdp'])) {
        $erreurs['mdp'] = 'Veuillez saisir votre mot de passe';
    }
    if (!empty($erreurs)) {
        return [
            'success' => false,
            'erreurs' => $erreurs,
        ];
    }
    else{
        return [
            'success' => true,
        ];
    }
}

// Insertion dans la BDD
function Traitement(PDO $db, array $informations){
    $req = Request($db, "INSERT INTO 'utilisateur' ('pseudo_utilisateur', 'mdp_utilisateur', 'mail_utilisateur',
    ['genre_utilisateur', 'id_date', 'role_utilisateur'] VALUES (:user, :mdp, :email, :genre, 1, 0)", $informations);
    return $req;
}

// vérifier si la BDD existe
function PostTraitement($db, array $informations){
    if($db == null){
        $erreurs['database'] = 'Database non initialisée contactez un administrateur';
    }
    else{
        // Dans le cas contraire
        $req = Request($db, "SELECT * FROM 'utilisateur' WHERE 'pseudo_utilisateur' = :user", ['user'=>$informations['user']])->fetch();
        if(empty($req)){
            $erreurs['user'] = 'nom d\'utilisateur déjà existant';
        }
    }
    if (!empty($erreurs)) {
        return [
            'success' => false,
            'erreurs' => $erreurs,
        ];
    }
    else{
        return [
            'success' => true,
        ];
    }
}
?>
```