

# Projet-Live Question

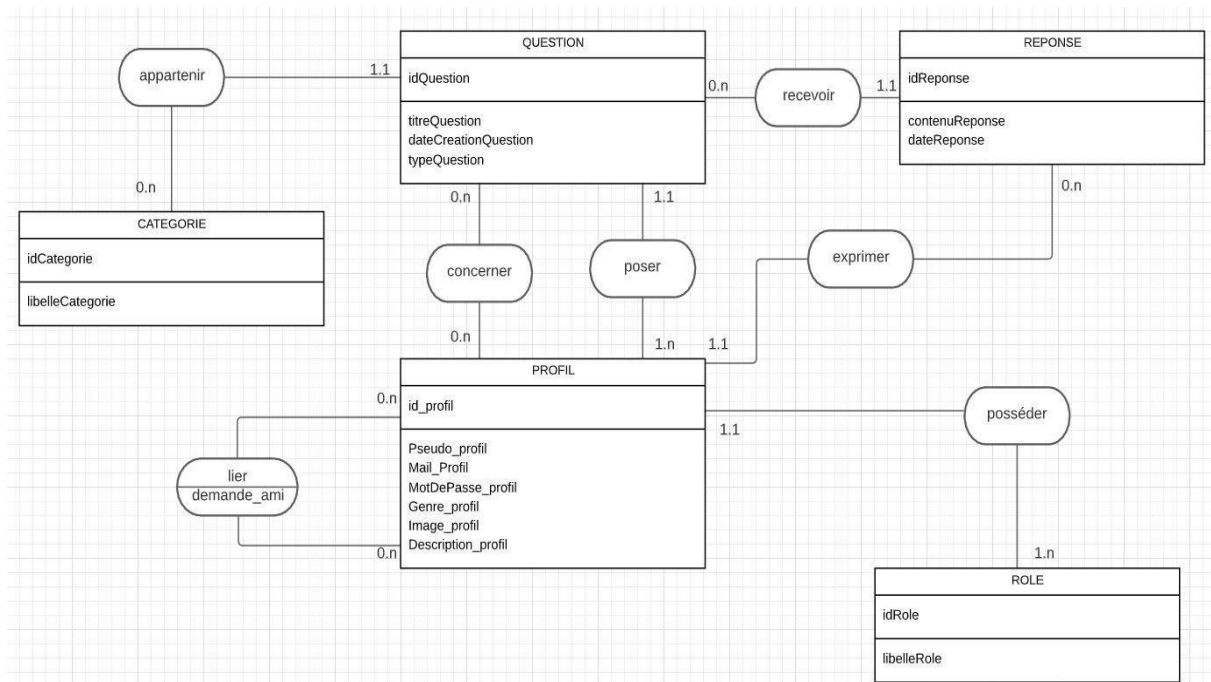
Ce projet consiste à réaliser un site interactif en fonction d'un cahier des charges et de maquettes données. Le site comporte 2 grandes parties : La partie *non-connectée*, avec une page design précise à réaliser, et une partie connectée, où l'utilisateur change d'interface et peut envoyer et/ou répondre à des questions.

## Dictionnaire de données :

Nom	Type	Commentaire
Id_categorie	INT	Identifiant de la catégorie
Libelle_categorie	VARCHAR	Libelle de la catégorie
Id_question	INT	Identifiant de la question
Titre_question	VARCHAR	La question posée
Date_creation_question	DATE	Date d'envoi de la question
Type_question	VARCHAR	Si la question est privée ou non
Id_profil	INT	Identifiant du profil
Pseudo_profil	VARCHAR	Nom d'utilisateur
Mail_profil	VARCHAR	Email de l'utilisateur
Motdepasse_profil	VARCHAR	Mot de passe de l'utilisateur
Genre_profil	VARCHAR	Genre de l'utilisateur
Image_profil	VARCHAR	Lien à l'image de l'utilisateur
Description_profil	VARCHAR	Description du profil
Id_reponse	INT	Identifiant de la réponse
Contenu_reponse	VARCHAR	Contenu de la réponse

Date_reponse	DATE	Date d'envoi de la réponse
Id_role	INT	Identifiant d'un rôle
Libelle_role	VARCHAR	Description du rôle

## Schéma relationnel :



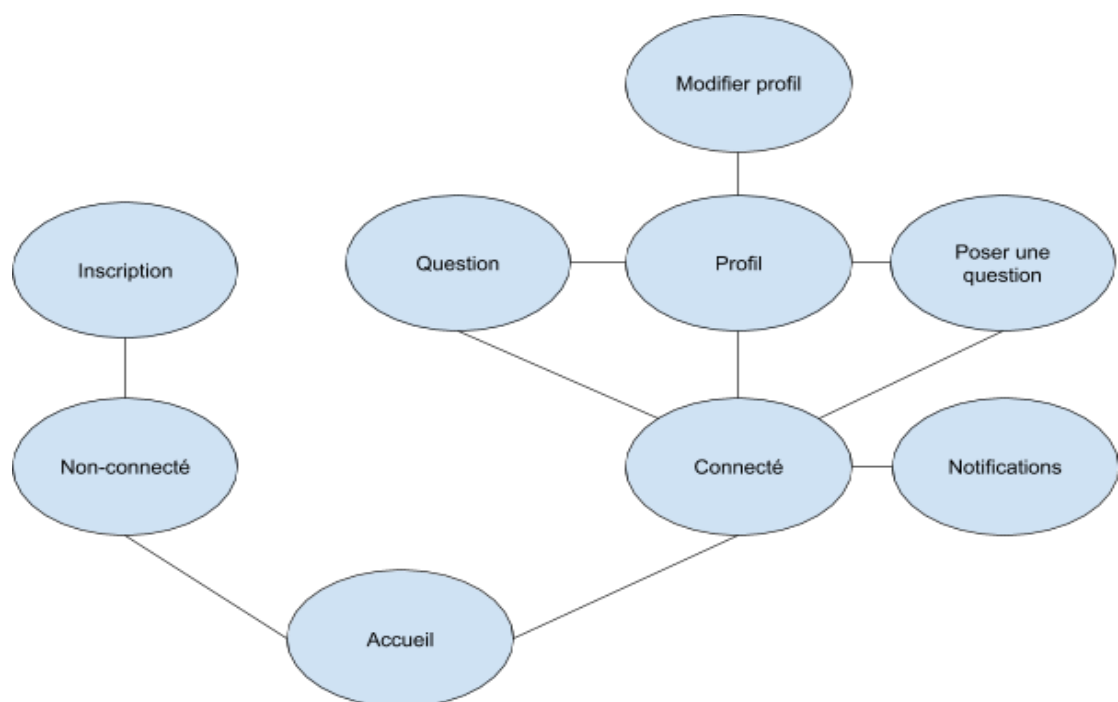
## Les fonctionnalités du site :

### Toutes les fonctionnalités :

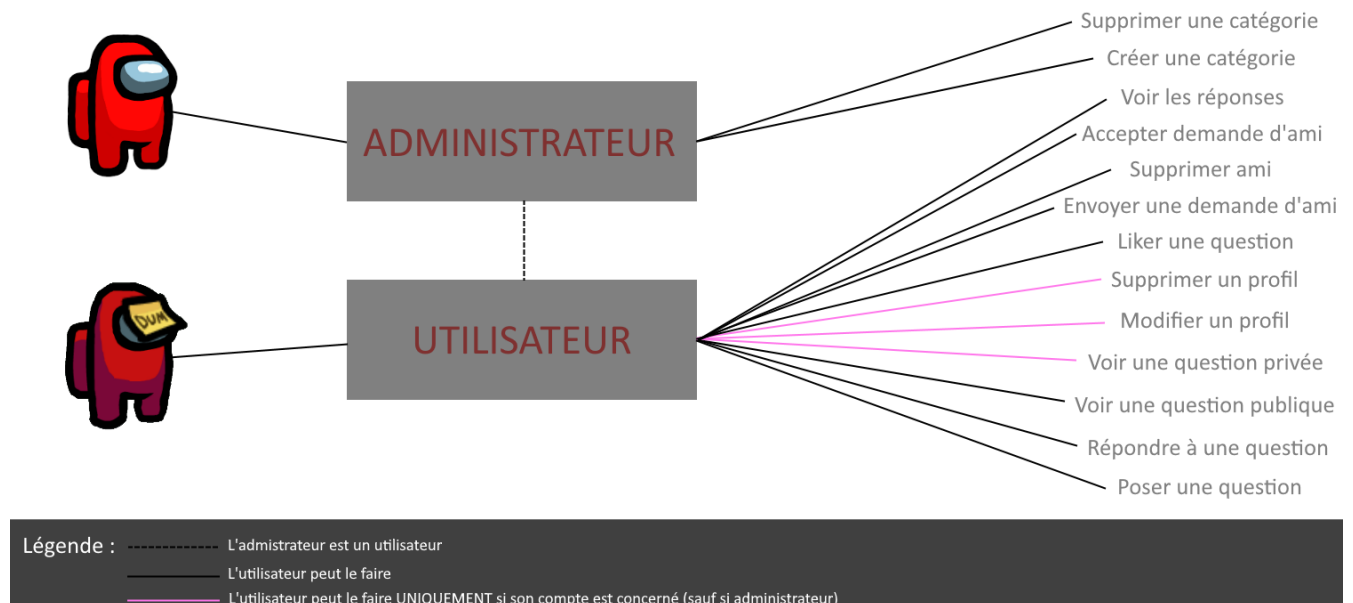
- L'utilisateur peut *créer* un compte ou *se connecter* s'il en a déjà un.
- L'*e-mail* est utilisé comme moyen d'identification unique. Un pseudo déjà existant peut être réutilisé.
- Le *mot de passe* est *crypté* une fois dans la base de données, aucun moyen d'obtenir le mot de passe d'un compte dedans donc. Le mot de passe est à confirmer 2 fois lors de l'inscription ou de la modification de ce dernier.
- Une fois *connecté*, l'utilisateur peut voir une nouvelle page d'accueil qui regroupe toutes les questions posées de la plus récente à la plus ancienne.

- L'utilisateur peut alors *répondre* à une question ou en *poser* une lui-même.
- L'utilisateur peut aussi *modifier* son profil : son pseudo, son e-mail (s'il n'existe pas déjà), son mot de passe (besoin du mot de passe actuel + double vérification), sa description, son genre et son image de profil.
- Une fonction *supprimer* est également présente : l'utilisateur peut supprimer une question qu'il a posée, ou bien son profil. Un administrateur peut supprimer tout ce qui lui semble bon.
- Les comptes sont mis administrateurs directement depuis la base de données (#Id\_role = 1 étant égal au statut administrateur). • D'autres fonctionnalités secondaires sont présentes.
- Essayer d'accéder à une page du site partie *connecté* sans être connecté résulte d'un accès refusé.
- **Update 30/06/20** : L'utilisateur possède maintenant une liste d'amis, des questions uniquement pour les amis, un système de triage de questions, un système de like et une meilleure gestion de l'image de profil. Les administrateurs peuvent supprimer/ajouter une catégorie

### Arborescence du projet :



## Cas d'utilisation :



## Comment installer le site :

- Télécharger le projet en .zip,
- Démarrer WAMP, dans le dossier www : exporter tout le .zip (à l'exception de fixtures.sql et README.md) dans un dossier du nom de votre choix,
- Créer un nouveau Virtual Host avec un nom au choix et le répertoire www du dossier contenant le projet téléchargé,
- Redémarrer les services WAMP,
- Sur phpMyAdmin, créer une base de données en latin1\_general\_ci avec le nom **live\_question** et importer la base de données fixtures.sql en utf-8,
- Le site est maintenant fonctionnel.
- **Si un problème avec la base de données survient : modifiez vos identifiants de connexion ou le nom de la base de données dans db/base\_PDO.php.**

## Script SQL :

```
--  
-- Structure de la table `profil`  
--  
  
DROP TABLE IF EXISTS `profil`;  
CREATE TABLE IF NOT EXISTS `profil` (  
  `Id_profil` int(11) NOT NULL AUTO_INCREMENT,  
  `Pseudo_profil` varchar(255) COLLATE latin1_general_ci NOT NULL,  
  `Mail_profil` varchar(255) COLLATE latin1_general_ci NOT NULL,  
  `MotDePasse_profil` varchar(255) COLLATE latin1_general_ci NOT NULL,  
  `Genre_profil` varchar(50) COLLATE latin1_general_ci NOT NULL,  
  `Image_profil` varchar(255) CHARACTER SET utf8 NOT NULL,  
  `Description_profil` varchar(255) COLLATE latin1_general_ci NOT NULL,  
  `#Id_role` int(11) NOT NULL,  
  PRIMARY KEY (`Id_profil`),  
  KEY `#Id_role` (`#Id_role`)  
) ENGINE=InnoDB AUTO_INCREMENT=71 DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci;
```

Exemple de création de table (profil)

[Lien du script complet](#)

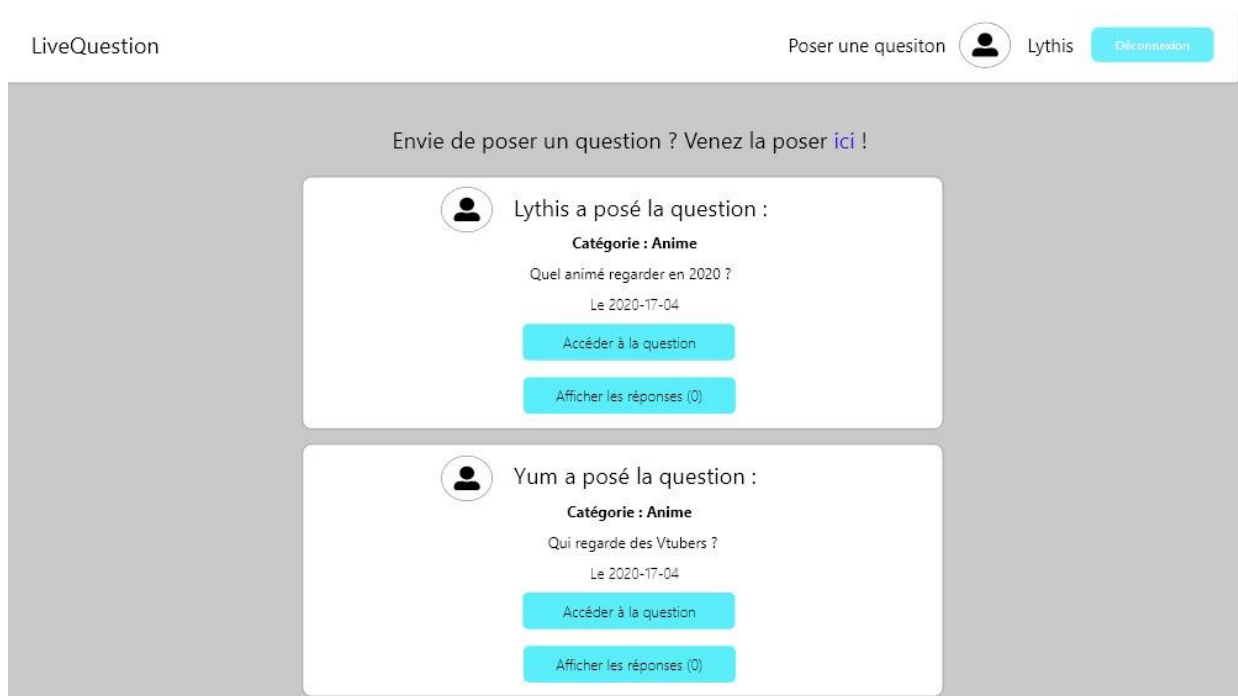
## Jeu d'essai :

```
INSERT INTO `profil` (`Id_profil`, `Pseudo_profil`, `Mail_profil`, `MotDePasse_profil`, `Genre_profil`, `Image_profil`, `Description_profil`, `#Id_role`) VALUES  
(1, 'Yam', 'root@livequestion.com', '$2y$10$wFNN./VGapt0JTYVbcWQb0NnEXq1nVFLN4VoALfY6agFs0sFkQTWk', 'Non-binaire', './image_profil/1.jpg', 'uwu', 1),
```

Exemple d'insertion d'un profil

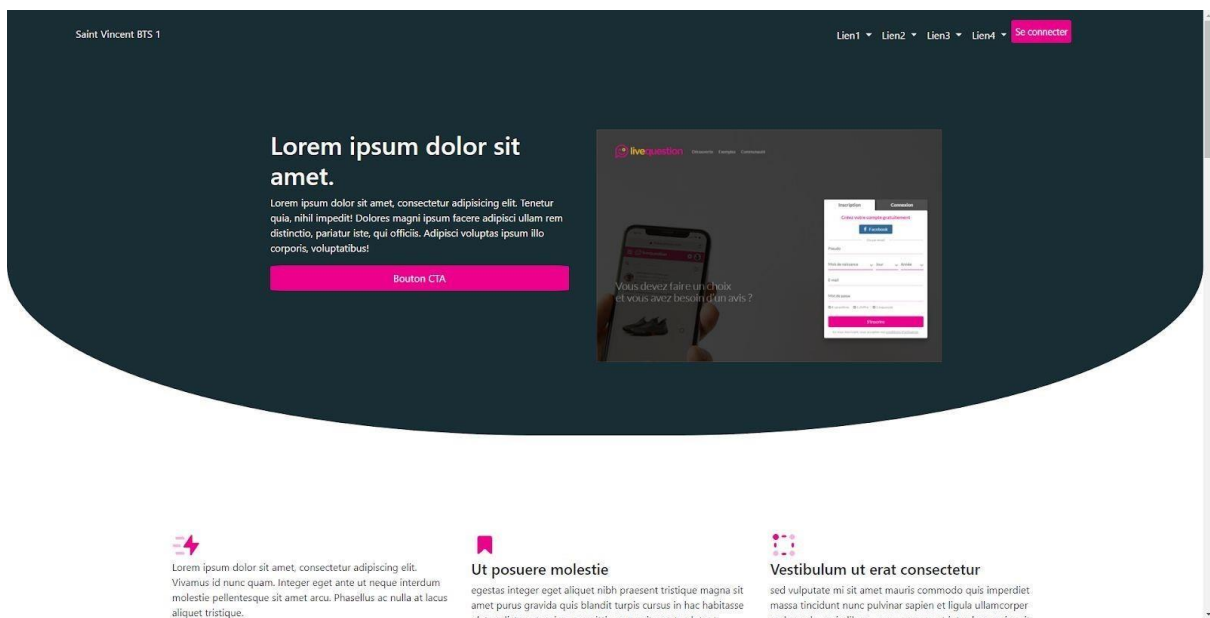
[Lien de la fixture](#)

## Maquettes :

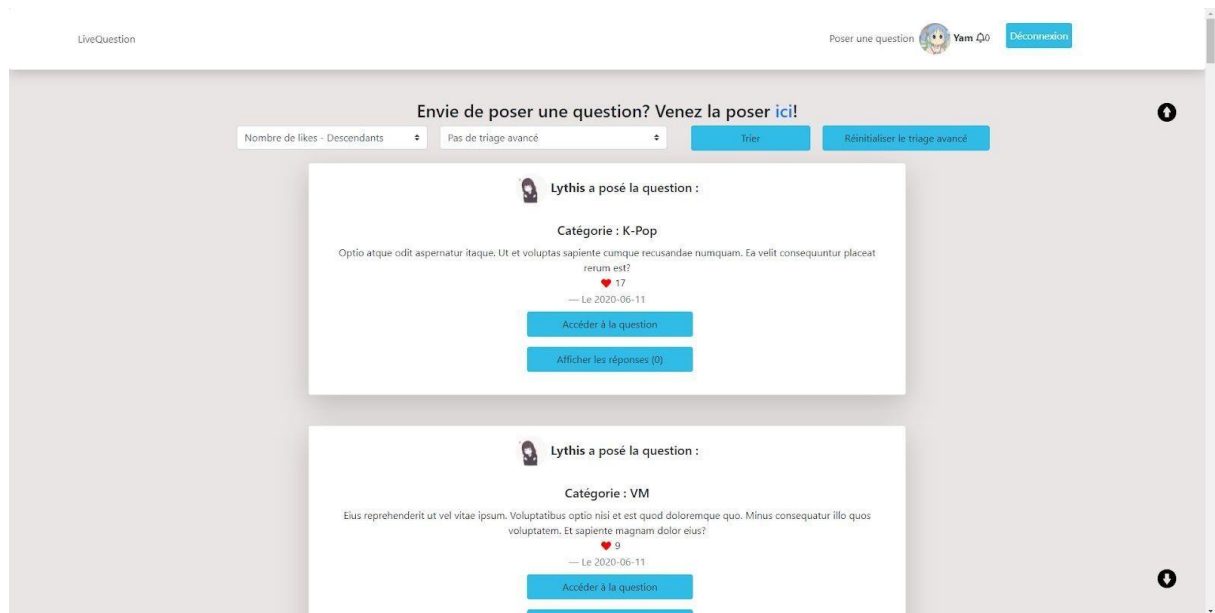




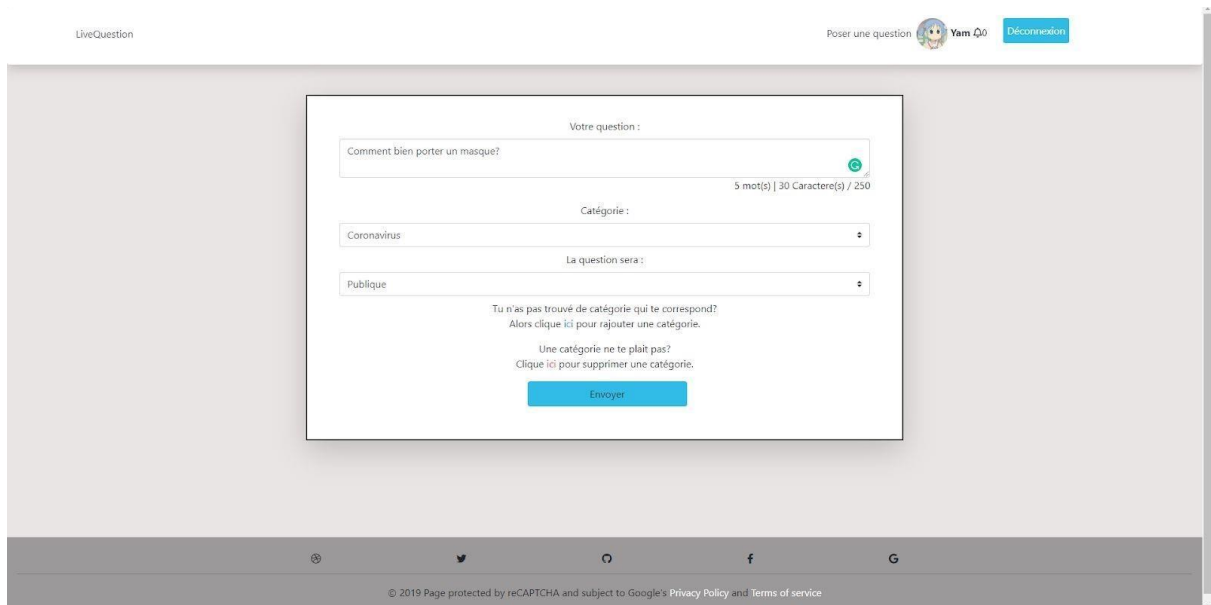
## Screenshots du site :



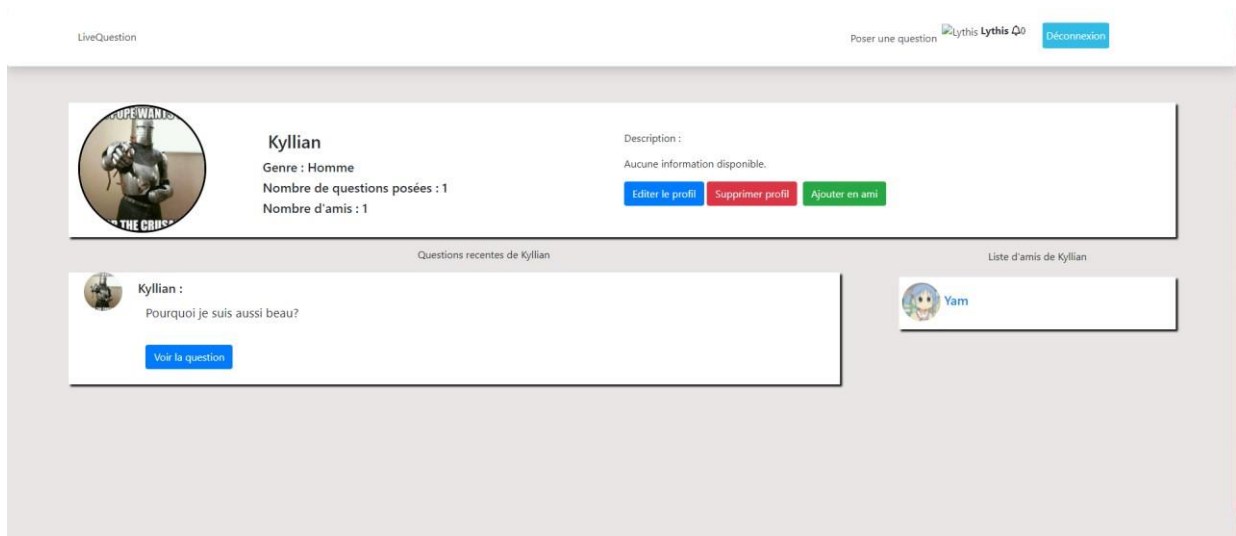
## Accueil non-connecté



## Accueil connecté



## Poser une question



Page profil

## Les fonctionnalités principales :

- Poser une question aux autres utilisateurs

Cette fonctionnalité est la plus importante du site car c'est la fonctionnalité qui est à l'origine du site.

```
#Fonction pour insérer une question dans la base de données, ne retourne rien
function insertIntoQuestion($question, $date, $utilisateur, $categorie, $visible) {
    $con = connexionBdd();

    $query = $con->prepare('INSERT INTO `question` (`Titre_question`, `Date_creation_question`, `Type`, `#Id_profil`, `#Id_categorie`) VALUES (:question, :dateajd, :typeQ, :id_user, :id_categorie)');
    $query->bindParam(':question', $question);
    $query->bindParam(':dateajd', $date);
    $query->bindParam(':typeQ', $visible);
    $query->bindParam(':id_user', $utilisateur);
    $query->bindParam(':id_categorie', $categorie);
    $query->execute();
}
```

- Répondre à une question

La seconde fonctionnalité du site la plus importante est le système de réponse. Celle-ci accompagne la fonctionnalité permettant de poser des questions et permet au site d'avoir un aspect social beaucoup plus important.

```
#Fonction pour insérer une réponse dans la base de données, ne retourne rien
function insertIntoReponse($reponse, $date, $utilisateur, $question) {
    $con = connexionBdd();

    $query = $con->prepare('INSERT INTO `reponse` (`Contenu_reponse`, `Date_reponse`, `#Id_profil`, `#Id_question`) VALUES (:reponse, :dateajd, :id_user, :id_question)');
    $query->bindParam(':reponse', $reponse);
    $query->bindParam(':dateajd', $date);
    $query->bindParam(':id_user', $utilisateur);
    $query->bindParam(':id_question', $question);
    $query->execute();
}
```



## Veille réalisée :

Pour la réalisation de notre projet, nous nous sommes formés principalement sur différents langages comme PHP et SQL ainsi que sur Ajax qui nous a permis de faire le système de like. Pour la réalisation de ces veilles, nous avons eu l'aide de professionnel ainsi que des recherches internet, en utilisant très souvent la documentation officielle suivie de tests.

## Conduite de projet :

Pour la conduite de projet, nous nous retrouvions de 15h à 17h, 5 jours sur 7. Chacun était assigné à une tâche qui était directement notée sur le GitHub. Lorsqu'une tâche devenait trop compliquée ou demandait une autre tâche en complément, on prenait le temps de travailler ensemble pour réaliser au mieux cette tâche. Le projet s'est déroulé à distance et nous n'avons pas rencontré de difficultés majeures, en allant même jusqu'à ajouter des fonctionnalités supplémentaires pour le client comme un système de triage, un meilleur profile.

Pour l'organisation du groupe :

Nicolas : Backend

Morgan : Frontend/Backend

Kylian : Backend

## Compte rendu de réunion :

Appels sur Discord chaque jour de la semaine hors week-end (sauf si nécessaire). Respect de deadlines en fonction du Gantt créé sur GitHub. La réunion commence en demandant ce que chacun compte faire cette journée et si de l'aide est nécessaire. On demande aussi si quelqu'un peut finir quelque chose en priorité car nécessaire pour tester des fonctionnalités (Exemple : peux-tu terminer le design pour la suppression de catégorie ? J'aimerais tester la fonction en cas réel.). Le temps moyen de réunion était de 2h par jour en même temps que de travailler.

## Membres du projet :

- Nicolas Valenti ([@YummYume](#))
- Morgan Watbled ([@Lythis](#))
- Kyllian Louis Joseph ([@prxph3te](#))