

Université Claude Bernard Lyon 1

Institut Universitaire de Technologie

BUT Génie Électrique et Informatique Industrielle

Parcours AII



SAE – Bases de Données Avancées

Compte rendu de projet – Mini réseau social interne à l'IUT

Projet : UniLink

Étudiants :

- Bonilla Rafael

Enseignants :

Année universitaire :

Groupe : Gr 521

2025 – 2026

Autoévaluation et répartition des contributions

Étudiants	Contributions %	Raison
	25%	Conception BDD et Compte rendu
Bonilla Rafael	25%	Codage interface et tests
	25%	Conception BDD, correction du code et tests
	25%	Design de l'interface et Compte rendu

Autoévaluation	15/20
Motif	<p>Nous estimons que notre travail mérite la note de 15/20, car le projet répond pleinement aux objectifs pédagogiques de la SAE “Bases de données avancées”, mais aussi prend en compte certaines améliorations.</p> <p>En effet, nous avons conçu et développé une application fonctionnelle. Celle-ci fonctionne à l’aide d’une base de donnée relationnelle structurée, intégrant plusieurs tables interconnectées entre elles (utilisateur, posts, likes, signalements, ...). Les relations entre ces tables ont été précédemment établies grâce au modèle E/A, afin de permettre une gestion optimale.</p> <p>Le site UniLink permet une authentification sur différents facteurs (rôles, mail, nom, prénom, id étudiant). La gestion des différents rôles permet de différencier les droits de chacun sur ce site. Il permet aussi de poster ses questions afin d’aider la communication entre les professeurs, le personnel administratif et les étudiants.</p> <p>Celui-ci possède un fil d’actualité dynamique, de réponses aux posts, de like, hashtags,...</p> <p>De plus, nous avons réussi à mettre en place un système de modérations complet, incluant le signalement de post afin qu’ils puissent être surveillés par les administrateurs et si besoin supprimer ou alors accepter par ceux-ci en cas de signalement abusif.</p> <p>Ce projet, a été réalisé avec une architecture claire séparant les codes HTML, CSS. JavaScript et PHP, mais aussi la base de données. Les codes sont lisibles, commentés afin de pouvoir faire des maintenances sur celui-ci plus tard.</p> <p>In fine, le travail de groupe a été structuré, chaque partie a été séparée. Un binôme a été attribué à la base de données, tandis qu’en parallèle un autre créé le début du site. Pour finir, les binômes se sont rassemblés dans le but de rendre le site plus optimal.</p> <p>La création d’un GitHub au début du projet a aidé de manière considérable le travail. Il nous a permis un accès direct aux codes sources pour</p>

	<p>notre équipe et les professeurs de façon à pouvoir faire des modifications en temps réel.</p> <p>Malgré certaines fonctionnalités comme la cybersécurité avec la vérification des mails, et la suppression d'utilisateurs qui n'ont au final pas été implantés, leurs réflexions durant le projet, montrent une réflexion de travail d'équipe très prometteuse pour l'avenir.</p> <p>Pour toutes ces raisons, nous pensons que ce travail correspond à un travail abouti et réfléchi, justifiant une note de 15/20.</p>
--	--

Sommaire

Université Claude Bernard Lyon 1	1
Institut Universitaire de Technologie	1
BUT Génie Électrique et Informatique Industrielle	1
SAE – Bases de Données Avancées	1
Compte rendu de projet – Mini réseau social interne à l'IUT.....	1
1. Introduction.....	6
2. Présentation générale du projet.....	7
3. Architecture du système.....	7
4. Fonctionnalités principales.....	9
5. Base de données.....	13
6. Organisation du travail.....	18
7. Limites du prototype et améliorations possibles.....	19
8. Conclusion.....	20
9. Procédure d'installation.....	21
Procédure fonctionnement local (ordinateur uniquement).....	21
Procédure fonctionnement en ligne.....	22

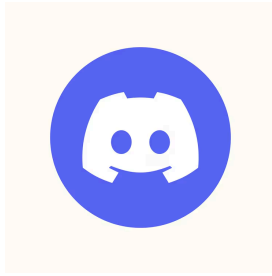
1. Introduction

Ce rapport présente le travail réalisé dans le cadre de la SAE “**Bases de données avancées**”. Le projet consistait à développer une **application web complète** reposant sur une base de données, avec pour objectif de créer un **mini réseau social interne à l’IUT**. Le cahier des charges imposait plusieurs éléments essentiels : permettre aux utilisateurs de **publier des messages, échanger entre eux, consulter un fil d’actualité clair**, gérer **quatre profils d’utilisateur** (Élève, Enseignant, Modérateur, Admin) et assurer la **gestion ainsi que la modération des publications**.

Pour répondre à ces attentes, nous avons conçu une application appelée **UniLink**, pensée comme un **prototype fonctionnant en local et en ligne**. Ce choix s’explique par notre **manque de connaissance pour réaliser un fonctionnement réel avec un serveur internet**. L’objectif n’était pas de créer un réseau social destiné à être réellement utilisé, mais de montrer que nous sommes capables de **concevoir une base de données, de développer une interface web** et de faire **fonctionner l’ensemble de manière cohérente**. Cependant c’est un **bon exemple d’amélioration** pour mener le projet à un niveau supérieur.



Notre groupe était composé de **quatre étudiants**, répartis en **deux binômes** : un binôme chargé de la **base de données et du back-end (SQL et PHP)**, et un autre binôme chargé de **l’interface (HTML, CSS et JavaScript)**. Nous avons travaillé ensemble en utilisant **Discord** pour communiquer et **GitHub** pour partager et suivre le code. Cette organisation nous a permis d’avancer efficacement tout en gardant une **bonne cohérence dans le projet**.



Le prototype UniLink propose les **fonctionnalités principales d'un réseau social** : **création de compte, connexion, publication de messages, gestion des hashtags, tri des posts, système de likes, affichage du fil d'actualité et outils de modération**. Certaines fonctionnalités plus avancées, comme un **upload dans un serveur web** ou une **meilleure protection de cybersécurité** avec une **vérification des comptes mail existants**, n'ont pas encore été mises en place.

Ce rapport présente notre **démarche**, nos **choix techniques**, ce que nous avons **réussi à mettre en place**, ce qui **reste à améliorer**, ainsi que ce que nous avons **appris grâce à ce projet**.

2. Présentation générale du projet

Le projet **UniLink** avait pour objectif de créer un **petit réseau social interne**, pensé pour fonctionner uniquement au sein de l'IUT. L'idée était de proposer un espace simple où les utilisateurs peuvent **publier des messages, répondre aux autres et consulter un fil d'actualité clair et organisé**. Même si l'application reste un prototype, elle reprend les bases d'un vrai réseau social, avec une **structure cohérente** et des **fonctionnalités essentielles**.

Pour rendre l'application utilisable par différents types d'utilisateurs, nous avons mis en place **plusieurs rôles** : **étudiant, enseignant, modérateur et administrateur**. Chaque rôle possède des **droits différents**. Par exemple, les étudiants peuvent **publier et signaler** des posts ; les enseignants peuvent **publier, répondre et signaler** ; tandis que les administrateurs et modérateurs peuvent, en plus de répondre, **supprimer et modérer les publications**. Cette gestion des rôles était importante pour reproduire un fonctionnement proche de celui d'une **plateforme réelle**.

UniLink fonctionne grâce à une **base de données MySQL** qui stocke les utilisateurs, les messages, les conversations, les likes, les hashtags et les signalements. Le site web, développé en **PHP**, interagit avec cette base pour afficher les informations et permettre aux utilisateurs d'agir dessus. L'interface, réalisée en **HTML, CSS et JavaScript**, a été pensée pour être **simple à comprendre et agréable à utiliser**.

Certaines fonctionnalités plus avancées ont été réfléchies mais pas traitées par manque de temps ou de connaissances, comme les notions de **cybersécurité**, la **protection des données**, la **vérification des comptes mails** ou encore la possibilité de **restreindre l'accès aux personnes extérieures** à la formation.

Cette partie du rapport permet donc de comprendre l'**objectif global du projet** et la manière dont nous avons organisé UniLink pour répondre aux besoins de la SAE. Dans la section suivante, nous

détaillerons plus précisément **l'architecture du système** et la façon dont les différents éléments communiquent entre eux.

3. Architecture du système

L'architecture d'UniLink a été pensée pour rester **simple**, tout en couvrant les **besoins essentiels d'un réseau social**. L'application repose sur une structure classique en **trois parties** : une **interface web** visible par l'utilisateur, un ensemble de **pages PHP** qui gèrent les actions, et une **base de données MySQL** qui stocke toutes les informations. Même si le projet reste un prototype, nous avons essayé de garder une **organisation propre et logique** pour faciliter la compréhension et l'évolution du code.

Le fonctionnement général est basé sur un **système de sessions**. Lorsqu'un utilisateur se connecte, ses informations (comme son rôle ou son identifiant) sont enregistrées dans la session. Cela permet au site de savoir **qui est connecté** et de vérifier si l'utilisateur a le droit d'effectuer certaines actions, comme **modifier un message** ou **accéder à la modération**. Cette **gestion des permissions** est un point important du projet, car elle garantit que chaque rôle ne peut faire que ce qui lui est autorisé.

L'interface du site est composée de plusieurs **pages principales** : la **page de connexion**, la **page d'accueil** avec le fil d'actualité, la **page permettant de créer un post**, la **page d'un post** avec les réponses, et la **page de profil**. Chaque page a un rôle précis et communique avec la base de données grâce à des **requêtes SQL**. Par exemple, la page du fil d'actualité récupère les **posts les plus récents**, tandis que la page d'un post récupère les **messages associés** et les affiche dans l'ordre.



Page de connexion à UniLink et extrait du code associé (login.php)

```
/*
SECTION : Vérification existence utilisateur
*/

$sql = "SELECT * FROM utilisateur WHERE email = ?";
$stmt = $conn->prepare($sql);
$stmt->bind_param("s", $email);
$stmt->execute();

$result = $stmt->get_result();
$user = $result->fetch_assoc();

/*
SECTION : Gestion erreurs (email ou mdp)
*/

if (!$user) {
    header("Location: index.html?error=email");
    exit;
}

if (!password_verify($mdp, $user["hash_mdp"])) {
    header("Location: index.html?error=mdp");
    exit;
}
```

L'interface a également été pensée pour fonctionner aussi bien **sur ordinateur que sur téléphone**. La mise en page s'adapte automatiquement à la taille de l'écran, ce qui permet aux utilisateurs de consulter le fil d'actualité, de publier un message ou de gérer leur profil depuis **n'importe quel appareil**. Cette flexibilité rend l'application plus **pratique et accessible**.

Le **back-end**, développé en **PHP**, s'occupe de **traiter les formulaires**, d'**insérer ou modifier les données** dans la base, et de **vérifier les droits** des utilisateurs. Nous avons séparé les fichiers selon

leur fonction : certains **affichent des pages**, d'autres **traitent des actions** comme la création d'un post, la suppression d'un message ou l'ajout d'un like. Cette

```
/* -----  
SECTION : Suppression du média si existant  
----- */  
  
if (!empty($msg["media_path"])) {  
  
    // Le message contient seulement le nom: "xxxx.png"  
    // Le fichier est physiquement dans public/uploads/  
    $fullPath = "../uploads/" . $msg["media_path"];  
  
    if (file_exists($fullPath)) {  
        unlink($fullPath);  
    }  
}  
  
/* -----  
SECTION : Suppression du message en BDD  
----- */  
  
$sqlDel = "DELETE FROM message WHERE id_message = ?";  
$stmtDel = $conn->prepare($sqlDel);  
$stmtDel->bind_param("i", $id_msg);  
$stmtDel->execute();
```

```
/* -----  
SECTION : Toggle du like (AJAX)  
----- */  
  
if ($already) {  
  
    /* ---- Suppression du like ---- */  
    $sqlDel = "  
        DELETE FROM likes  
        WHERE id_utilisateur = ?  
        AND id_conversation = ?  
    ";  
    $stmtDel = $conn->prepare($sqlDel);  
    $stmtDel->bind_param("ii", $id_user, $id_conv);  
    $stmtDel->execute();  
  
    $label = "J'aime";  
}  
else {  
  
    /* ---- Ajout du like ---- */  
    $sqlIns = "  
        INSERT INTO likes (id_utilisateur, id_conversation)  
        VALUES (?, ?)  
    ";  
    $stmtIns = $conn->prepare($sqlIns);  
    $stmtIns->bind_param("ii", $id_user, $id_conv);  
    $stmtIns->execute();  
  
    $label = "Je n'aime plus";  
}
```

séparation permet de garder un **code plus clair et plus facile à maintenir**.

[Exemples de traitements réalisés dans différents script PHP \(delete_message.php et like_post.php\)](#)

Enfin, la **base de données** joue un rôle central dans l'architecture. Elle contient les **utilisateurs**, les **posts**, les **messages**, les **likes**, les **hashtags** et les **signalements**. Chaque table est liée aux autres de manière **logique**, ce qui permet au site de retrouver rapidement les informations nécessaires. Même si certaines fonctionnalités avancées n'ont pas été développées, la structure de la base a été pensée pour pouvoir **les accueillir plus tard**.

Cette architecture **simple mais organisée** nous a permis de développer UniLink de manière progressive, en ajoutant les fonctionnalités une par une tout en gardant une **cohérence globale**. Dans la partie suivante, nous présenterons plus en détail les **fonctionnalités principales** que nous avons mises en place.

4. Fonctionnalités principales

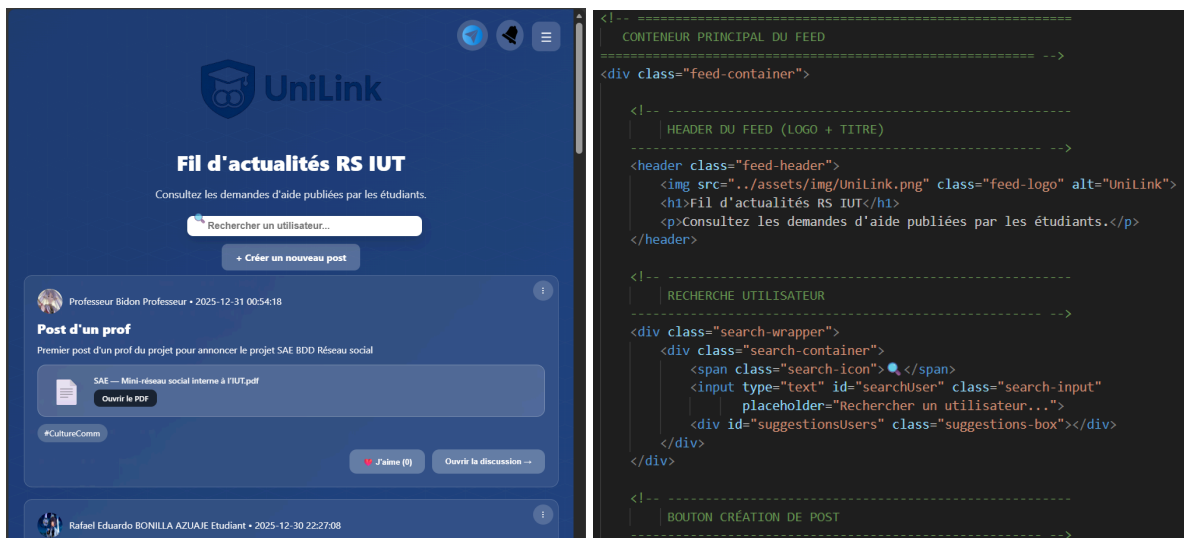
Le prototype **UniLink** intègre plusieurs **fonctionnalités essentielles** qui permettent de reproduire le fonctionnement d'un **réseau social**. Même si l'application reste simple, nous avons essayé de couvrir les **actions les plus importantes** pour que les utilisateurs puissent réellement échanger entre eux.

La première fonctionnalité mise en place est la **création de compte**. Chaque utilisateur peut s'inscrire en fournissant ses informations, puis se connecter grâce à un **système d'authentification basé sur une session**. (Par raison de sécurité, seuls les comptes **étudiants** et **enseignants** peuvent être créés via ce formulaire ; les comptes **administrateurs** et **modérateurs** sont créés directement dans la base de données.) Une fois connecté, il accède au **fil d'actualité**, qui affiche les **posts les plus récents** publiés

par les autres membres de l'IUT. Cette page est le **point central de l'application**, car elle permet de voir rapidement ce qui a été posté et d'ouvrir les discussions qui nous intéressent.



Page d'inscription à UniLink et extrait du code associé (inscription.php)



Fil d'actualités et extrait du code associé (feed.php)

Les utilisateurs peuvent ensuite **créer un post**, en ajoutant un **titre**, un **message** et éventuellement un **média** comme une image ou un PDF. Ils peuvent aussi ajouter des **hashtags**, ce qui permet de **classer les posts** et de **faciliter la recherche**. Une fois le post publié, il apparaît dans le **fil d'actualité** et peut être ouvert pour afficher la **discussion complète**.

```

/* -----
SECTION : Association des hashtags au post
----- */

if ( !empty($hashtags) && is_array($hashtags) ) {

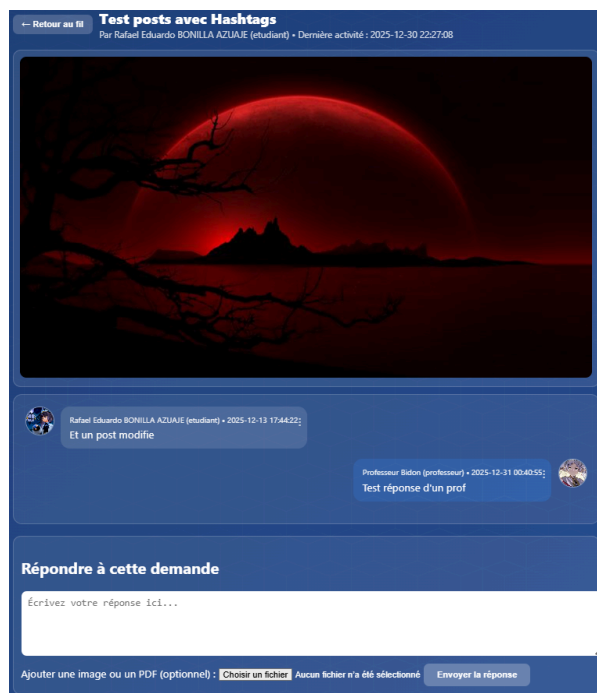
    $sqlTag = "
        INSERT INTO hashtag_conversation (id_hashtag, id_conversation)
        VALUES (?, ?)
    ";
    $stmtTag = $conn->prepare($sqlTag);

    foreach ($hashtags as $id_hashtag) {
        if ( ctype_digit((string)$id_hashtag) ) {
            $id_hashtag = (int)$id_hashtag;
            $stmtTag->bind_param("ii", $id_hashtag, $id_conversation);
            $stmtTag->execute();
        }
    }
}

```

Page de création de post et extrait du code permettant l'association de hashtags au post (post_create.php)

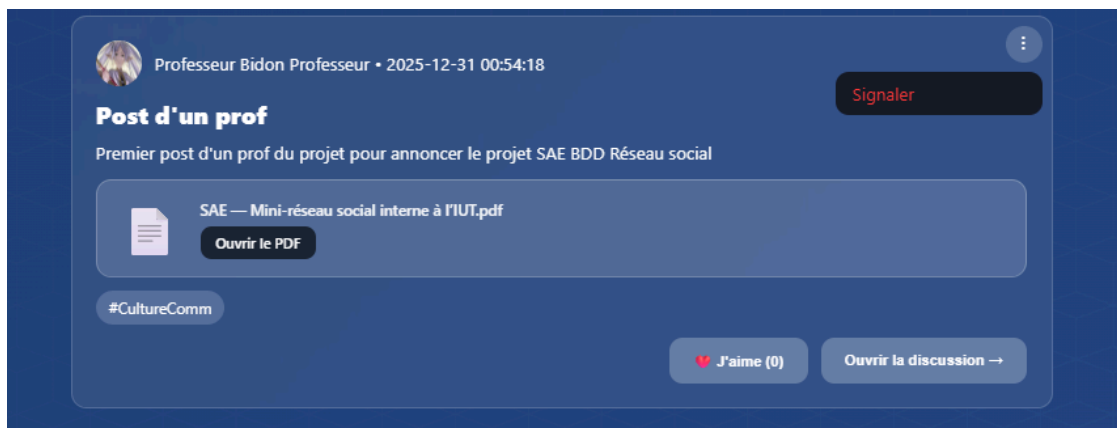
Dans chaque post, les utilisateurs **autres que les étudiants** peuvent **répondre** en ajoutant un message. Les réponses sont affichées dans **l'ordre chronologique**, ce qui permet de **suivre facilement la conversation**. Nous avons aussi ajouté un **système de likes**, qui permet de montrer qu'un message ou un post est apprécié. Le like fonctionne en **"toggle"**, c'est-à-dire qu'on peut **liker ou retirer son like** en cliquant sur le **même bouton**.



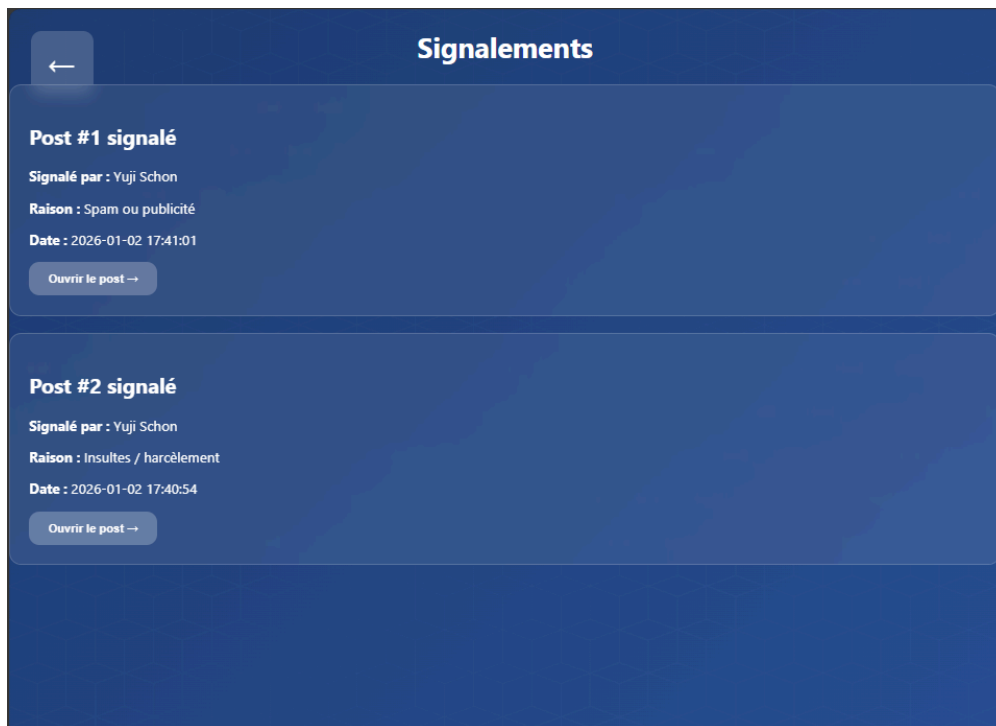


Exemple de visualisation d'un post depuis un compte étudiant (à gauche) et depuis un compte enseignant (à droite)

Pour garantir un **minimum de sécurité** et éviter les abus, nous avons intégré un **système de signalement**. Si un utilisateur estime qu'un contenu est inapproprié, il peut le **signaler**. Les **modérateurs** et **administrateurs** peuvent ensuite consulter ces signalements dans une **page dédiée** qui rassemble les posts signalés avec leur **raison de signalement**, puis ils peuvent décider de **supprimer le contenu** si nécessaire. Cette fonctionnalité est importante pour reproduire le fonctionnement d'un **vrai réseau social**, même si le projet reste un prototype.

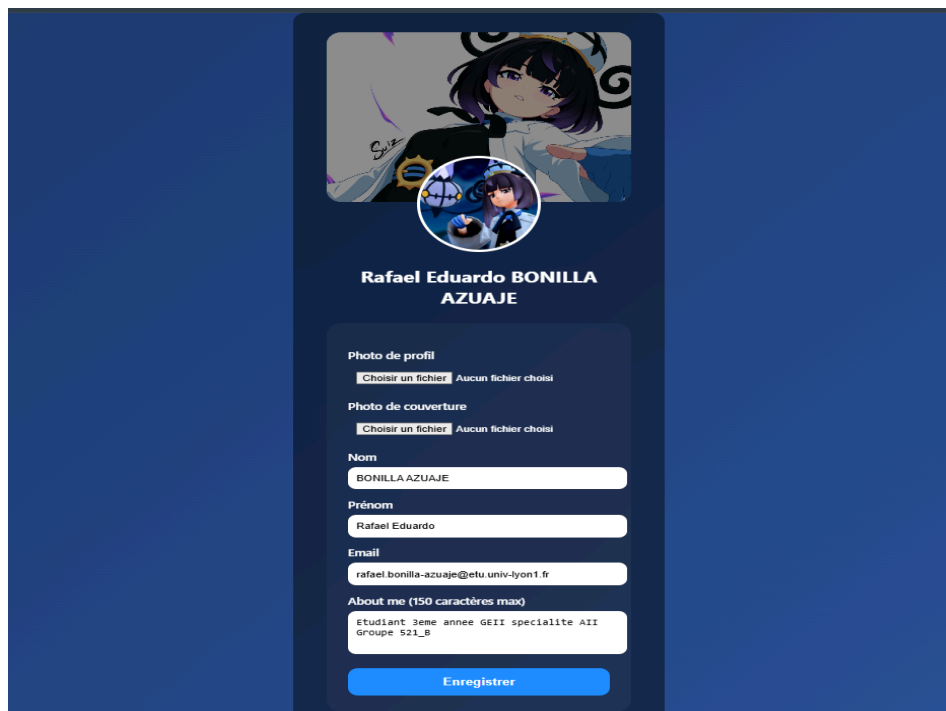


Le bouton "Signaler" apparaît lors d'un appui sur les 3 points d'un post dans le fil d'actualités

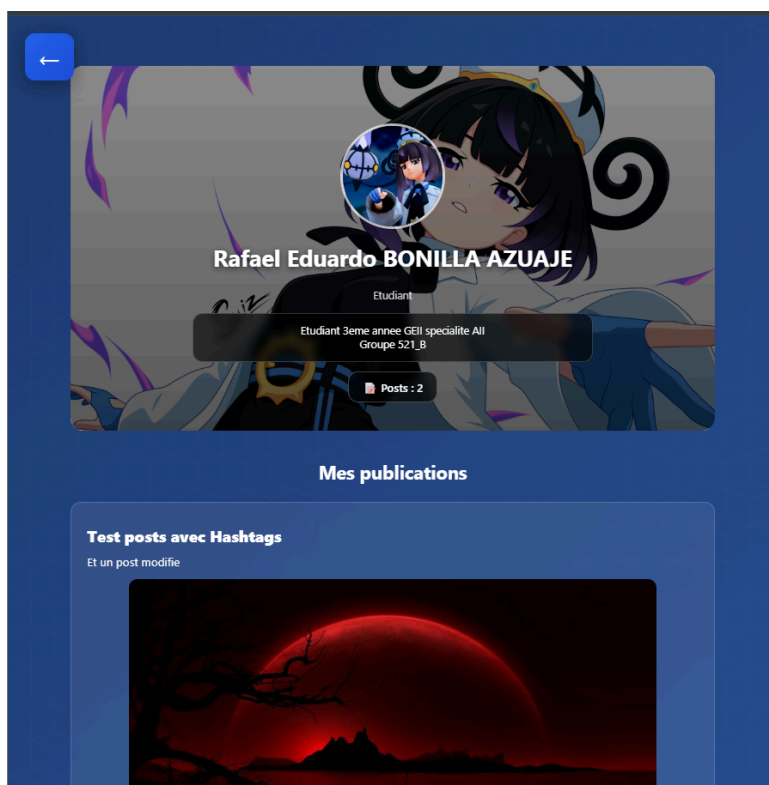


Page des signalements accessible par les modérateurs et administrateurs

Enfin, chaque utilisateur dispose d'une **page de profil**. Il peut y modifier ses **informations personnelles**, son **avatar**, sa **photo de couverture** et sa **biographie**. Une autre page permet aussi de consulter **tous les posts qu'il a publiés**, ce qui facilite la **navigation** et le **suivi de ses propres contributions**.



Exemple de page de profil



Exemple de page mes posts

Ces fonctionnalités constituent la **base du prototype UniLink**. Elles permettent de **publier, échanger, réagir** et **gérer son profil**, ce qui correspond aux **objectifs fixés au début du projet**. Dans la partie suivante, nous présenterons plus en détail la **structure de la base de données** qui permet de faire fonctionner l'ensemble.

5. Base de données

La **base de données** est un élément **central** du projet UniLink, car elle permet de **stocker toutes les informations nécessaires** au fonctionnement du réseau social. Nous avons utilisé **MySQL**, car c'est un système que nous connaissons bien et qui est **adapté à une application web** de ce type. L'objectif était d'avoir une base **claire, organisée** et suffisamment **flexible** pour accueillir les différentes fonctionnalités du prototype.



Pour structurer les données, nous avons créé **plusieurs tables** correspondant aux **éléments principaux de l'application**. La **table des utilisateurs** stocke les informations de chaque membre de l'IUT, comme son **nom**, son **rôle** ou son **avatar**.

Table des utilisateurs :

	id_utilisateur	id_univ	nom	prenom	email	hash_md5	role	avatar	avatar_bg	bio
<input type="checkbox"/>		1	BONILLA AZUJUE	Rafael Eduardo	rafael.bonilla-azuje@etu-univ-lyon1.fr	\$2y\$10\$Z.ZaO2TzaPDWGWf9aTeUy,EncDcqHr0YUHGk0ld...	etudiant	1764689178_3659.png	1767003638_bg_7735.png	Etudiant 3eme annee GEII specialite All Groupe 52...
<input type="checkbox"/>		2	p0000000	Admin	principal.admin@univ-lyon1.fr	\$2y\$10\$Vjwm8t86UEBXP7MxjczuzGmWDL5Wt9SNGvOv3UBMFE...	admin	1764689499_5612.jpg	NULL	NULL
<input type="checkbox"/>		3	p0000001	Un	moderateur.un@univ-lyon1.fr	\$2y\$10\$8hWvW8Mc5ob8vHB/MmpWO5FkccQPcEv.S0Wz3ud...	moderateur	1764689433_7214.png	bg_6954720334cca.webp	NULL
<input type="checkbox"/>		4	p0000002	Deux	moderateur.deux@univ-lyon1.fr	\$2y\$10\$a0nScg33gk50dMLUGacqKewOvmqNVCrgJ1SSCOBAC4...	moderateur	1764689464_7181.png	bg_69547211aa1cc.avif	NULL
<input type="checkbox"/>		5	p0000003	Lyonnais	eleve.lyonnais@etu-univ-lyon1.fr	\$2y\$10\$4qQDWAmdpDVPuBw1YlK2zDDfFm3OC98EPEoMC5eb...	etudiant	1764689385_7072.jpg	NULL	NULL
<input type="checkbox"/>		6	p2300699	LEFORT	amandine.lefort@etu-univ-lyon1.fr	\$2y\$10\$wy1F45LX3DKOzpHDYFKW.upCOkFZVWkbbJK00CbL...	etudiant	default_user.png	NULL	NULL
<input type="checkbox"/>		7	p2307401	Schon	yuji.schon@etu-univ-lyon1.fr	\$2y\$10\$OzZ8D9VXWuRhKBWAs5Oe5wCRob80AaTwCUjpd85...	etudiant	1765185956_8108.jpg	NULL	NULL
<input type="checkbox"/>		8	p0000004	Bidon	professeur.bidon@univ-lyon1.fr	\$2y\$10\$DXtqKzvc0Cf0cMESYL7e00q25UFEZDRSaQz8ZU...	professeur	avatar_695461a502503.jpg	bg_695461a562860.jpg	Profil d'un prof bidon pour tester un quatrième t...

La **table des conversations** représente les **posts publiés** sur le fil d'actualité, tandis que la **table des messages** contient les **réponses associées** à chaque post.

Table des posts publiés (nom de table conversation) :

	id_conversation	Titre	media_path	Statutmoderation	Datedernieremodif	id_utilisateur
<input type="checkbox"/>		1	Premier post apres bdd corrompu	valide	2025-12-02 16:31:44	1
<input type="checkbox"/>		2	Deuxieme post apres bdd corrompu	valide	2025-12-02 16:35:30	5
<input type="checkbox"/>		3	Premier post dun admin	valide	2025-12-02 16:38:46	2
<input type="checkbox"/>		7	Joyeux Noël !	valide	2025-12-08 10:28:59	7
<input type="checkbox"/>		8	J'aime pas Noël ! C'est Surcôlé !! Halloween c'est...	valide	2025-12-08 11:10:47	6
<input type="checkbox"/>		12	Test posts avec Hashtags	valide	2025-12-30 22:27:08	1
<input type="checkbox"/>		13	Post d'un prof	valide	2025-12-31 00:54:18	8

Table des messages :

	id_message	Contenu	media_path	Statutmessage	Date&heure	id_utilisateur	id_conversation
<input type="checkbox"/>		1	Ceci est un premier test post eleve	valide	2025-12-02 16:31:44	1	1
<input type="checkbox"/>		2	Ceci est un premier test de reponse dun admin apre...	valide	2025-12-02 16:32:28	2	1
<input type="checkbox"/>		3	Ceci est la reponse du modeun apres bdd corrompu L...	valide	2025-12-02 16:33:13	3	1
<input type="checkbox"/>		4	Ceci est la reponse du moddeux apres bdd corrompu u...	valide	2025-12-02 16:34:31	4	1
<input type="checkbox"/>		5	Ceci est un test de demande dun eleve avec une pho...	valide	2025-12-02 16:35:30	5	2
<input type="checkbox"/>		6	Apres une bdd corrompu vaut mieux faire autant des...	valide	2025-12-02 16:38:46	2	3
<input type="checkbox"/>		10	Voici un Haro de Gundam avec un bonnet de Noël	valide	2025-12-08 10:28:59	7	7
<input type="checkbox"/>		11	Faite Halloweennnnn les GENS !!!	valide	2025-12-08 11:10:47	6	8
<input type="checkbox"/>		15	Et un post modifie	valide	2025-12-13 17:44:22	1	12
<input type="checkbox"/>		16	Test réponse d'un prof	valide	2025-12-31 00:40:55	8	12
<input type="checkbox"/>		17	Premier post d'un prof du projet pour annoncer le ...	valide	2025-12-31 00:54:18	8	13

La **table de DM conversation** représente le **stockage des conversations**, tandis que la **table des DM messages** contient la **date d'envoi** et le **contenu des messages**. La **table de DM participant** représente les **participants** dans cette conversation.












Table DM conversation :

	id_dm_conversation	date_creation
<input type="checkbox"/>	1	2025-12-30 23:05:58
<input type="checkbox"/>	2	2025-12-31 01:08:35

Table DM message :




	id_dm_message	contenu	media_path	date_envoi	id_utilisateur	id_dm_conversation	vu
<input type="checkbox"/>	1	Salut ceci est un premier dm	NULL	2025-12-31 01:08:46	8	2	1
<input type="checkbox"/>	2	Salut ceci est une premiere reponse	NULL	2025-12-31 01:09:27	1	2	1

Table DM participant :

		id_dm_conversation	id_utilisateur	derniere_lecture	supprime
<input type="checkbox"/>	 Éditer  Copier  Supprimer	1	1	NULL	0
<input type="checkbox"/>	 Éditer  Copier  Supprimer	1	6	NULL	0
<input type="checkbox"/>	 Éditer  Copier  Supprimer	2	1	NULL	0
<input type="checkbox"/>	 Éditer  Copier  Supprimer	2	8	NULL	0

La **table des Hashtag conversation** fait le **lien entre les hashtags** et les **conversations**.

Table des Hashtag conversation :

		id_hashtag	id_conversation
<input type="checkbox"/>	 Éditer  Copier  Supprimer	1	1
<input type="checkbox"/>	 Éditer  Copier  Supprimer	2	12
<input type="checkbox"/>	 Éditer  Copier  Supprimer	3	12
<input type="checkbox"/>	 Éditer  Copier  Supprimer	6	12
<input type="checkbox"/>	 Éditer  Copier  Supprimer	8	12
<input type="checkbox"/>	 Éditer  Copier  Supprimer	9	12
<input type="checkbox"/>	 Éditer  Copier  Supprimer	11	13

Nous avons aussi ajouté des **tables** pour gérer les **likes**, les **hashtags**, les **signalements** et les **notifications**, même si certaines de ces fonctionnalités ne sont pas encore totalement **exploitées** dans le prototype.

Table des Hashtag :











		id_hashtag	Descriptif
<input type="checkbox"/>	 Éditer  Copier  Supprimer	1	Mathématiques
<input type="checkbox"/>	 Éditer  Copier  Supprimer	2	Informatique
<input type="checkbox"/>	 Éditer  Copier  Supprimer	3	Électronique
<input type="checkbox"/>	 Éditer  Copier  Supprimer	4	Automatisme
<input type="checkbox"/>	 Éditer  Copier  Supprimer	5	Réseaux
<input type="checkbox"/>	 Éditer  Copier  Supprimer	6	BDD
<input type="checkbox"/>	 Éditer  Copier  Supprimer	7	Énergie
<input type="checkbox"/>	 Éditer  Copier  Supprimer	8	Automatique
<input type="checkbox"/>	 Éditer  Copier  Supprimer	9	Anglais
<input type="checkbox"/>	 Éditer  Copier  Supprimer	10	PPP
<input type="checkbox"/>	 Éditer  Copier  Supprimer	11	CultureComm

Table des signalements :










		id_signalement	id_post	id_user_reporteur	raison	date_signalement	etat
<input type="checkbox"/>	 Éditer  Copier  Supprimer	1	1		5 Autre	2025-12-03 15:54:45	traite
<input type="checkbox"/>	 Éditer  Copier  Supprimer	7	2		1 Hors sujet	2025-12-29 16:04:45	traite
<input type="checkbox"/>	 Éditer  Copier  Supprimer	8	3		8 Autre	2025-12-31 01:06:26	traite

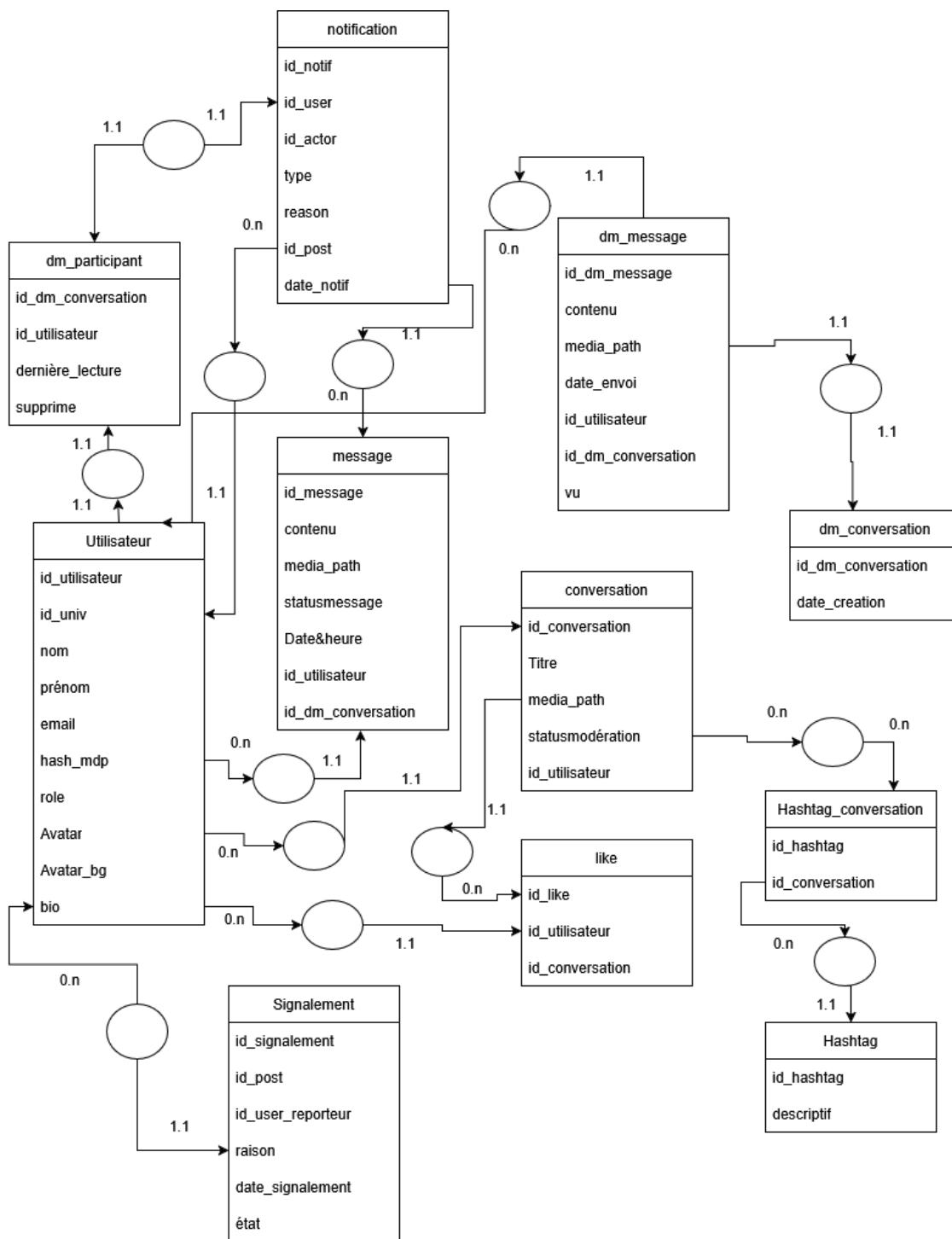
Table des likes :

			id_like	id_utilisateur	id_conversation	
<input type="checkbox"/>	Éditer	Copier	Supprimer	5	1	1
<input type="checkbox"/>	Éditer	Copier	Supprimer	6	1	2
<input type="checkbox"/>	Éditer	Copier	Supprimer	7	1	3
<input type="checkbox"/>	Éditer	Copier	Supprimer	10	2	1
<input type="checkbox"/>	Éditer	Copier	Supprimer	14	2	2
<input type="checkbox"/>	Éditer	Copier	Supprimer	15	2	3
<input type="checkbox"/>	Éditer	Copier	Supprimer	25	2	8
<input type="checkbox"/>	Éditer	Copier	Supprimer	3	3	1
<input type="checkbox"/>	Éditer	Copier	Supprimer	17	3	2
<input type="checkbox"/>	Éditer	Copier	Supprimer	16	3	3
<input type="checkbox"/>	Éditer	Copier	Supprimer	26	3	8
<input type="checkbox"/>	Éditer	Copier	Supprimer	33	3	12
<input type="checkbox"/>	Éditer	Copier	Supprimer	4	4	1
<input type="checkbox"/>	Éditer	Copier	Supprimer	19	4	2
<input type="checkbox"/>	Éditer	Copier	Supprimer	18	4	3
<input type="checkbox"/>	Éditer	Copier	Supprimer	27	4	8

Table des notifications :

				id_notif	id_user	id_actor	type	reason	id_post	date_notif
<input type="checkbox"/>	Éditer	Copier	Supprimer	1	5	1	like	NULL	2	2025-12-02 16:50:12
<input type="checkbox"/>	Éditer	Copier	Supprimer	2	2	2	like	NULL	3	2025-12-02 16:50:15
<input type="checkbox"/>	Éditer	Copier	Supprimer	3	5	2	like	NULL	2	2025-12-02 16:51:31
<input type="checkbox"/>	Éditer	Copier	Supprimer	4	1	2	like	NULL	1	2025-12-02 16:51:35
<input type="checkbox"/>	Éditer	Copier	Supprimer	5	5	2	like	NULL	2	2025-12-03 09:38:15
<input type="checkbox"/>	Éditer	Copier	Supprimer	6	5	2	like	NULL	2	2025-12-03 09:50:13
<input type="checkbox"/>	Éditer	Copier	Supprimer	7	2	3	like	NULL	3	2025-12-03 09:51:36
<input type="checkbox"/>	Éditer	Copier	Supprimer	8	5	3	like	NULL	2	2025-12-03 09:51:43
<input type="checkbox"/>	Éditer	Copier	Supprimer	9	2	4	like	NULL	3	2025-12-03 09:51:55
<input type="checkbox"/>	Éditer	Copier	Supprimer	10	5	4	like	NULL	2	2025-12-03 09:51:56
<input type="checkbox"/>	Éditer	Copier	Supprimer	11	5	1	report	Hors sujet	2	2025-12-03 15:08:52
<input type="checkbox"/>	Éditer	Copier	Supprimer	12	3	5	report	Autre	3	2025-12-03 15:13:25
<input type="checkbox"/>	Éditer	Copier	Supprimer	13	4	5	report	Autre	3	2025-12-03 15:13:25
<input type="checkbox"/>	Éditer	Copier	Supprimer	14	2	5	report	Autre	1	2025-12-03 15:54:45
<input type="checkbox"/>	Éditer	Copier	Supprimer	15	3	5	report	Autre	1	2025-12-03 15:54:45
<input type="checkbox"/>	Éditer	Copier	Supprimer	16	4	5	report	Autre	1	2025-12-03 15:54:45
<input type="checkbox"/>	Éditer	Copier	Supprimer	32	7	1	like	NULL	7	2025-12-24 14:56:25
<input type="checkbox"/>	Éditer	Copier	Supprimer	33	1	3	like	NULL	12	2025-12-29 15:23:53
<input type="checkbox"/>	Éditer	Copier	Supprimer	34	2	1	report	Hors sujet	2	2025-12-29 16:04:45
<input type="checkbox"/>	Éditer	Copier	Supprimer	35	3	1	report	Hors sujet	2	2025-12-29 16:04:45
<input type="checkbox"/>	Éditer	Copier	Supprimer	36	4	1	report	Hors sujet	2	2025-12-29 16:04:45
<input type="checkbox"/>	Éditer	Copier	Supprimer	37	1	8	reply	NULL	12	2025-12-31 00:40:55
<input type="checkbox"/>	Éditer	Copier	Supprimer	38	1	3	like	NULL	12	2025-12-31 00:51:25
<input type="checkbox"/>	Éditer	Copier	Supprimer	39	3	8	report	Autre	3	2025-12-31 01:06:26
<input type="checkbox"/>	Éditer	Copier	Supprimer	40	4	8	report	Autre	3	2025-12-31 01:06:26

La base de données s'organise selon le **modèle E/A** suivant:



Chaque table est liée aux autres de manière **logique**. Par exemple, un **message** est toujours associé à un **utilisateur** et à une **conversation**. De la même façon, un **like** est lié à un **utilisateur** et à un **post**. Ces relations permettent au site de **retrouver rapidement les informations nécessaires** pour afficher le fil d'actualité, les discussions ou les profils. Nous avons également fait attention à ce que les **suppressions soient cohérentes** : lorsqu'un **post est supprimé**, les **messages**, les **likes** et les **médias associés** doivent disparaître pour éviter les **données inutiles**.

Même si la base de données reste relativement **simple**, elle a été pensée pour pouvoir **évoluer**. Certaines tables, comme celles des **notifications** ou des **signalements**, pourraient être utilisées plus largement si le projet était poursuivi. Cela montre que, même dans un **cadre pédagogique**, nous avons essayé de **prévoir la suite** et de rendre l'application plus **facile à améliorer**.

Cette base de données constitue donc la **fondation technique d'UniLink**. Elle permet de gérer les **utilisateurs**, les **posts**, les **interactions** et la **modération**, tout en restant suffisamment **claire** pour être comprise et modifiée facilement. Dans la partie suivante, nous expliquerons comment nous avons **organisé notre travail en groupe** pour mener ce projet à bien.

6. Organisation du travail

Pour mener à bien ce projet, nous avons travaillé en **groupe de quatre étudiants**. Dès le début, nous avons choisi de nous organiser en **deux binômes** afin de **répartir les tâches de manière équilibrée**. Le premier binôme s'est occupé de la **base de données** et du **back-end**, c'est-à-dire toute la partie liée au **PHP**, aux **requêtes SQL** et au **fonctionnement interne du site**. Le second binôme (**Rafael**) s'est concentré sur **l'interface**, avec le développement des **pages HTML**, la **mise en forme en CSS** et l'ajout de **JavaScript** pour rendre l'application plus agréable à utiliser.

Cette séparation nous a permis d'avancer **plus efficacement**, car chacun pouvait se concentrer sur un **domaine précis** tout en restant en contact avec les autres. Pour communiquer, nous avons utilisé **Discord**, ce qui nous a permis d'**échanger rapidement**, de **partager nos idées** et de **résoudre les problèmes** au fur et à mesure. Nous avons également utilisé **GitHub** pour **stocker le code**, **suivre les modifications** et **éviter les conflits** entre les différentes versions du projet. Cela nous a aidés à garder une **bonne organisation** et à travailler de manière plus **professionnelle**.

Même si nous avons des rôles différents, nous avons régulièrement fait des **points ensemble** pour vérifier que tout restait **cohérent**. Par exemple, le binôme chargé du back-end devait s'assurer que la **base de données** correspondait bien aux **besoins de l'interface**, tandis que le binôme chargé du front-end devait vérifier que les **pages affichaient correctement les données** envoyées par le serveur. Ce **travail en équipe** nous a permis de mieux comprendre l'importance de la **communication** dans un projet informatique.

Comme dans tout projet, nous avons rencontré quelques **difficultés**, notamment pour gérer certaines fonctionnalités comme les **médias**, les **permissions** ou les **signalements**. Mais en **discutant ensemble** et en **testant régulièrement**, nous avons réussi à trouver des solutions et à avancer. Cette organisation nous a donc permis de mener le projet **à terme** tout en apprenant à travailler de manière **structurée** et **collaborative**.

Dans la partie suivante, nous présenterons les **limites du prototype** et les **améliorations possibles** si le projet devait être poursuivi.

7. Limites du prototype et améliorations possibles

Même si **UniLink** fonctionne correctement et remplit les **objectifs principaux** de la SAE, il reste un **prototype** avec certaines **limites**. Certaines viennent du **manque de temps**, d'autres du fait que nous avons dû faire des **choix** pour garder le projet réalisable dans les délais. L'application propose les **fonctions essentielles** d'un réseau social, mais plusieurs éléments pourraient être **améliorés** ou **complétés** si le projet devait être poursuivi.

Les **modérateurs** et **administrateurs** peuvent consulter les **signalements**, vérifier les contenus et supprimer les posts ou les messages si nécessaire. Le système fonctionne comme prévu : lorsqu'un utilisateur signale un post, celui-ci apparaît dans la page du thread avec les **options de traitement** réservées au personnel. Même si la modération est **opérationnelle**, elle pourrait être améliorée avec une **meilleure interface visuelle**, un **compteur du nombre de signalements**, une **séparation par raison** si elles diffèrent, et même des **messages préventifs** envoyés aux utilisateurs après la modération de leur post.

L'**interface** pourrait également être améliorée. Nous avons fait en sorte qu'elle soit **simple** et **fonctionnelle**, mais certains éléments pourraient être mieux organisés ou plus modernes. Par exemple, le **fil d'actualité** pourrait proposer un **système de tri plus avancé**, ou l'affichage des posts pourrait être **plus visuel**. De même, la **page de profil** pourrait offrir davantage d'options de **personnalisation**.

Malgré ces limites, le prototype **remplit bien son rôle** dans le cadre de la SAE. Il montre que nous sommes capables de **concevoir une base de données**, de **développer une interface web** et de **faire fonctionner un ensemble cohérent**. Les **améliorations possibles** montrent aussi que le projet pourrait **évoluer facilement** si nous avons plus de temps.

8. Conclusion

Le projet **UniLink** nous a permis de mettre en pratique l'ensemble des compétences travaillées dans la SAE "**Bases de données avancées**". En développant un **mini réseau social interne**, nous avons pu concevoir une **base de données complète**, créer une **interface web fonctionnelle** et communiquer les deux de manière **cohérente**. Même si l'application reste un **prototype**, elle remplit les **objectifs fixés au départ**, respecte le **cahier des charges** et montre que nous sommes capables de réaliser une **application web structurée**.

Ce projet nous a également appris à **travailler en groupe**, à nous **organiser**, afin que l'on puisse tous ensemble suivre l'ensemble du projet et **se répartir les tâches** de manière efficace. L'utilisation de **Discord** et de **GitHub** nous a aidé à collaborer plus facilement et à garder une bonne **cohérence dans le code**. Nous avons aussi dû faire face à plusieurs **difficultés techniques**, ce qui nous a permis d'améliorer notre capacité à **chercher des solutions** et à comprendre le fonctionnement d'un **projet complet**.

Même si certaines fonctionnalités avancées n'ont pas été développées entièrement, comme la **cybersécurité**, nous avons conscience de leur pertinence dans la création d'un site web qui peut traiter des **données sensibles** des utilisateurs. Le prototype propose déjà les **fonctions essentielles** d'un réseau social : **création de compte**, **publications**, **réponses aux publications**, **likes**, **hashtags**, **notifications**, **messages privés**, **modération** et **gestion du profil**.

Finalement, **UniLink** est un projet qui nous a beaucoup apporté, autant sur le **plan technique** que sur le **plan organisationnel**. Il nous a permis de mieux comprendre comment fonctionne une **application web complète** et de nous préparer à des **projets plus ambitieux**. Ce rapport présente notre **démarche**, nos **choix** et les **résultats obtenus**, et montre le **travail réalisé** tout au long de cette SAE.

9. Procédure d'installation

Procédure fonctionnement local (ordinateur uniquement)

- Prérequis techniques : Environnement serveur
 - Localhost XAMPP
 - PHP version 8.0 ou supérieure
 - MySQL ou MariaDB version 10.4
 - Serveur Apache
- Logiciels nécessaires
 - Navigateur web (Chrome/Firefox/Edge)
 - Réceptacle du code entier (Visual Studio code recommandé)
 - phpMyAdmin (Fourni avec XAMPP)
 - GitHub, accès a repertoire projet https://github.com/SAE_reseau_social
- Récupération de l'ensemble du projet
 - Télécharger l'archive du projet ou cloner le dépôt Git
 - Extraire le dossier du projet
 - Placer le dossier a la racine du serveur web :
 - htdocs / pour XAMPP
 - www/ pour WAMP
 - Exemple : c:\xampp\htdocs\projet_reseaux_social
- Installation de la base de données
 - Création de la base de données
 - Ouvrir phpMyAdmin
 - Créer une bdd nommée "projet_reseaux_social"
 - Choisir l'encodage utf8mb4_general_ci
 - Importer le fichier SQL disponible dans le répertoire Git
 - Sélectionner la bdd créé
 - Aller dans l'onglet "Importer"
 - Importer le fichier SQL fournis avec Git
 - Verifier la presence des tables : utilisateur ; conversation ; message ; commentaire ; likes ; signalement ; notification ; dm ; etc
- Configuration de la connexion à la base de données
 - Ouvrir /config/db.php
 - Vérifier ou adapter les paramètres
 - \$host = "localhost";
 - \$user = "root";
 - \$pass = "";
 - \$dbname = "projet_reseaux_social";
- Lancement de l'application
 - Démarrer Apache et MySQL depuis XAMPP/WAMP
 - Ouvrir navigateur
 - Acceder via URL : http://localhost/projet_reseaux_social
 - Vérifier l'affichage de la page de connexion

Procedure fonctionnement en ligne

- Logiciels nécessaires
 - Navigateur web (Chrome/Firefox/Edge)
- Lancement de l'application
 - Ouvrir navigateur
 - Acceder vía URL : <https://geii-book.great-site.net/>
 - Vérifier l'affichage de la page de connexion