

Mr. DAGUELOU Rabah

Développeur web

Session 30 mars 2021



Soutenance

Projet 07

Créer un réseau social d'entreprise

Mentor : Mr. Florent Houx

Mentor évaluateur :

Sommaire

Introduction

- 1. Logiciels et technologies utilisées**
- 2. Fonctionnalités de l'application**
- 3. Le code source front-end**
- 4. Le code source back-end**
- 5. La base de données MySql**
- 6. Spécifications de l'application**

Conclusion

Introduction

Groupomania est un réseau social interne de l'entreprise. C'est un outil d'échange et d'interaction entre les collègues. Les employés de l'entreprise peuvent publier, partager et commenter des publications et des articles de sujets divers pour favoriser l'ambiance et la motivation au sein de l'équipe. Plusieurs fonctionnalités ont été développées afin de faciliter la communication entre collègues. Néanmoins, la première version que nous avons développée demeure un MVP susceptible d'être amélioré selon les besoins de l'entreprise.

1) Logiciels et technologies utilisés

Pour coder l'application Groupomania, nous avons utilisé plusieurs technologies de développement dont, essentiellement, le framework Vuejs, un serveur NodeJs, le framework Express et un serveur de base de données relationnel MySQL.

A/ Les technologies de développement :

Front-end :

- **HTML 5** : pour réaliser la structure du site
- **CSS et BOOTSTRAP** : pour la mise en forme et le responsive du site.
- **VUEJS** : J'ai choisi le framework de JavaScript VueJs pour réaliser l'application coté front-end.

Back-end :

- **NodeJs** : Pour créer un serveur
- **Express** : Framework JavaScript pour créer et gérer l'application coté back-end
- **MySQL** : SGBDR (Système de Gestion de Bases de Données Relationnelles) pour créer et gérer ma base de données.

B/ Bibliothèques, packages et modules:

NPM : (Node Package Manager) Gestionnaire de paquets officiel de Node.js. Utilisé pour télécharger et installer les différents paquets (modules) utilisés dans le projet.

VUE-ROUTER : Router officiel pour Vue.js pour faciliter la navigation entre composants et l'imbrication de plusieurs vues de l'application mono-page du front end.

VUEX : (*gestionnaire d'état* (« **state management pattern** »)) C'est une bibliothèque vue.js qui sert à centraliser les données partagées par plusieurs composants de l'application dans une zone de stockage unique appelée source de vérité.

AXIOS : Bibliothèque JavaScript (client http) pour faire communiquer l'application front-end avec l'application back-end via des requêtes http.

BODY-PARSER : Pour analyser les données des requêtes et renvoie un objet Json exploitable.

B-CRYPT : Pour crypter ou hacher les mots de passe utilisateurs avant leur enregistrement dans la base de données afin de sécuriser les données sensibles des utilisateurs.

CORS : Pour accepter le partage de ressources entre différentes origines. Dans notre cas l'application front-end est exécutée sur le port 8080 alors que l'application backend utilise le port 3000.

MULTER : un package de gestion de fichiers entrants avec les requêtes http. Il permet d'accepter des fichiers entrants et de les enregistrer en utilisant un middleware spécifique.

DOTENV : une bibliothèque permettant de stocker et de charger des configurations sensibles relatives à l'environnement dans des variables.

FS : (File System) module NodeJs fournissant plusieurs fonctionnalités relatives au système de fichiers (enregistrement, lecture, écriture, suppression...)

JSONWEBTOKEN : utilisé pour les opérations d'identification créant un access token (jeton d'accès) permettant un échange sécurisé de données entre deux parties (frontend/backend).

PATH : module Node.js qui fournit des fonctionnalités pour interagir avec les chemins de fichiers.

NODEMON : un utilitaire d'interface de ligne de commande (cli) utilisé dans la phase de développement pour surveiller le système de fichiers et redémarrer automatiquement le processus.

C/ Logiciels:

Visuel Studio Code : Pour écrire le code frontend et backend.

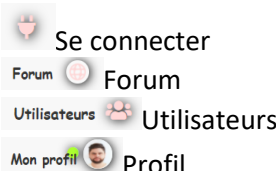
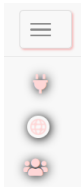



Google Chrome & Firefox : navigateurs pour exécuter l'application

Git Bash 2 : Pour versionner le projet tout au long de la phase de développement.

Git Hub : Bibliothèque open source pour stocker le projet dans un dépôt en ligne.







Lien github : https://github.com/rabah-daguelou/RabahDaguelou_p7_30112021.git







2) Fonctionnalités du site


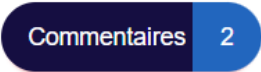



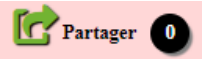
Fonctionnalité	Attendu	Perspectives & suggestions
1/ Page d'accueil (Login)		
Menu de navigation 	Les icones doivent être présentes 	
Erreur après le clic sur les icones	Message d'erreur: <div>Merci de vous connecter d'abord !</div>	
Permuter entre se connecter et s'inscrire	Le titre et le formulaire changent <div>CONNEXION Vous n'avez pas encore de compte? m'inscrire</div> <div>INSCRIPTION Vous avez déjà un compte? me connecter</div>	
Permuter entre l'icône connexion et Déconnexion	Déconnecté : icone de connexion  Connecté : <i>icone de déconnexion</i> 	
ECRAN D'INSCRIPTION D'UN UTILISATEUR 		
Je valide mon compte sans renseigner un ou plusieurs ou tous les champs du formulaire « Champs requis »	Message d'erreur: <div>Le champs Email est requis Le champs Mot de passe est requis Le champs Nom est requis</div> <i>Le formulaire n'est pas soumis</i>	Adresse email réservée (Voir supprimer un compte)
Email Groupomania	Message d'erreur: <div>Cette adresse email est interdite !</div>	
Email invalide	Message d'erreur : <div>Ce mail n'est pas valide</div>	












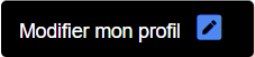
Mot de passe invalide	Message d'erreur : <div>Votre mot de passe est invalide! Au moins 8 caractères, une majuscule, un chiffre !</div>	
Pseudo invalide	Message d'erreur : <div>Vérifiez votre nom: au moins 3 caractères. 20 max !</div>	
Email déjà utilisé <i>« Un seul et unique email dans la base de données »</i>	Message d'erreur : <div>Cette adresse mail est déjà utilisée !</div>	
Email, mot de passe et pseudo valides Bouton « Je valide mon compte » <div>Je valide mon compte</div>	Au clic : <i>Un nouvel utilisateur est créé et enregistré dans la table « users » de la base de données</i> <i>« Un Token est généré et valable 1h. Au-delà, l'utilisateur est déconnecté automatiquement ! »</i> L'utilisateur est dirigé sur la page « Forum »	Valider le formulaire avec la touche entrer du clavier
<div> <div>ECRAN DE CONNEXION D'UN UTILISATEUR</div> <div> <div>CONNEXION</div> <div>Votre E-mail</div> <div>Votre mot de passe</div> <div>Je me connecte</div> <div>Vous n'avez pas encore de compte? m'inscrire</div> </div> </div>		
Je me connecte sans renseigner un ou plusieurs ou tous les champs du formulaire	Message d'erreur: <div> Le champs Email est requis Le champs Mot de passe est requis Le champs Nom est requis </div> Le formulaire n'est pas soumis	
Adresse mail inexistante dans la bdd	Message d'erreur : <div>Utilisateur introuvable!</div>	
Mot de passe incorrect	Message d'erreur : <div> Ce mot de passe est incorrect ! Mot de passe oublié ? </div>	Coder la fonctionnalité de récupération du mot de passe oublié
Email et mot de passe corrects Bouton <div>Je me connecte</div>	Aller sur la page « Forum » <i>« Un Token est généré et valable 1h. Au-delà, l'utilisateur est déconnecté automatiquement ! »</i>	Accéder à la page forum avec la touche entrer

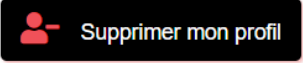
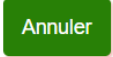

2/ La page du FORUM

<p>Accueil du forum</p> 	<p>Les éléments sur la page : Affichage de la photo, du pseudo de l'utilisateur connecté</p> <p>Le bouton « se déconnecter »  remplace le bouton « Se connecter » </p>	
<p>Le bouton « mon profil »</p> 	<p>Se transforme en photo du profil de l'utilisateur connecté avec un témoin vert</p>	
<p>Si administrateur</p> 	<p>Affichage de l'icône administrateur avec infobulle</p>	
<p>Formulaire de publication</p>		
<p>Publier un post vide</p>	<p>Message d'erreur pendant 3 secondes : « <i>Merci d'ajouter du texte ou une photo</i> »</p>	
<p>Publier un texte uniquement</p>	<p>La publication s'ajoute dans le flux</p>	
<p>Publier une photo volumineuse</p>	<p>Message d'erreur pendant 3 secondes : « Image volumineuse. 500 kb max ! »</p>	
<p>Publier un autre fichier</p>	<p>Message d'erreur pendant 3 secondes : « Uniquement png, jpg et jpeg »</p>	
<p>Publier une photo valide</p>	<p>La publication s'ajoute dans le flux</p>	<p>Glisser la photo dans le formulaire</p>
<p>Publier un texte et une photo</p>	<p>La publication s'ajoute dans le flux</p>	<p>Glisser la photo dans le formulaire</p>
<p>LES PUBLICATIONS</p>		
<p>Photo et nom de l'utilisateur</p> 	<p>Photo et nom de l'utilisateur qui a posté la publication</p>	
<p>Date de publication</p> <p>Le 01/01/2022 à 17:53:24</p>	<p>Publié le (date et heure en français)</p>	
<p>La photo</p> 	<p>Si la publication possède une photo</p>	
<p>Le texte</p> <p>texte</p>	<p>Si la publication a un texte</p>	

Likes et Deslikes		
<p>Le bouton Likes</p> 	<p>Présent sur toutes les publications avec un compteur du nombre de likes</p> <p>Au Clic :</p> <ul style="list-style-type: none"> - Si on a jamais ni liké ni desliké (On incrémente le compteur des likes) - Si on a déjà liké (On décrémente le compteur likes) - Si on a déjà desliké (On décrémente le compteur deslikes et on incrémente le compteur likes) 	
<p>Le bouton deslikes</p> 	<p>Présent sur toutes les publications avec un compteur du nombre de deslikes</p> <p>Au Clic :</p> <ul style="list-style-type: none"> - Si on a jamais ni liké ni desliké (incrémenter le compteur des deslikes) - Si on a déjà liké (Décrémenter le compteur likes et incrémenter le compteur deslikes) - Si on a déjà desliké (Décrémenter le compteur deslikes) 	
Supprimer une publication		
<p>Le bouton supprimer</p> 	<p>S'affiche uniquement sur :</p> <ul style="list-style-type: none"> - les publications Postées par l'utilisateur connecté ou - sur toutes les publications si l'utilisateur connecté est l'administrateur <p>Au clic : <i>La publication est supprimée (elle est retirée instantanément du flux)</i></p>	
Modifier une publication		
<p>Le bouton Modifier</p>  <p>Le bouton envoyer les modifications</p> 	<p>S'affiche uniquement sur :</p> <ul style="list-style-type: none"> - les publications que l'utilisateur connecté a publié <p>Au clic :</p> <ul style="list-style-type: none"> - Le formulaire de modification apparaît - L'utilisateur peut modifier soit le texte soit la photo ou les deux - Gestion des erreurs comme sur le formulaire de publication - La validation se fait via le bouton « envoyer » 	
Modifier le texte	Le texte doit se modifier instantanément	
Modifier la photo	La photo doit se modifier instantanément (L'ancien fichier photo est supprimé du dossier images si la publication n'est pas partagée)	
Modifier la photo et le texte	Toute la publication doit se modifier instantanément (L'ancien fichier photo est supprimé du dossier images si	

	<i>la publication n'est pas partagée)</i>	
Commentaires		
Le bouton « commenter » 	Présent sur la carte de chaque publication Au clic : <ul style="list-style-type: none"> - Un champs pour saisir un commentaire apparaît - Dès que l'utilisateur commence à saisir son commentaire, le bouton envoyer apparaît Au clic sur le bouton envoyer : Le commentaire est envoyé, et le compteur nombre de commentaire s'incrémente. Le commentaire est ajouté à la liste des commentaires.	
Le bouton « Commentaires » 	Est présent uniquement sur une publication déjà commentée. Au clic : Tous les commentaires sont affichés. Le nombre est incrémenté à chaque nouveau commentaire. Il est également décrémenté à chaque commentaire supprimé. Chaque commentaire est précédé de l'utilisateur l'ayant posté et la date de sa publication	
Masquer les commentaires	Fonctionnalité réservée exclusivement à l'administrateur.	
Le bouton « masquer » 	Est présent sur chaque commentaire « uniquement » si l'utilisateur connecté est l'administrateur. Au clic : Le commentaire en question sera masqué pour les utilisateurs	
Le bouton « voir » un commentaire 	Est présent sur chaque commentaire masqué « uniquement » si l'utilisateur connecté est l'administrateur. Au clic : Le commentaire en question sera démasqué pour l'administrateur	
Le bouton supprimer un commentaire 	Présent sur tous les commentaires si l'utilisateur connecté est l'administrateur OU Sur les commentaires postés par l'utilisateur connecté Au clic : Le commentaire est définitivement supprimé !	
Partager une publication		
Le bouton « partager » 	Présent sur chaque publication Sous forme d'une icone avec un compteur	
Partager une publication	Au clic : <ul style="list-style-type: none"> - La publication est partagée et remonte dans le flux des publications - Le nombre de partage est incrémenté 	
Une publication partagée	Contient un en-tête avec la photo du profil , le pseudo	

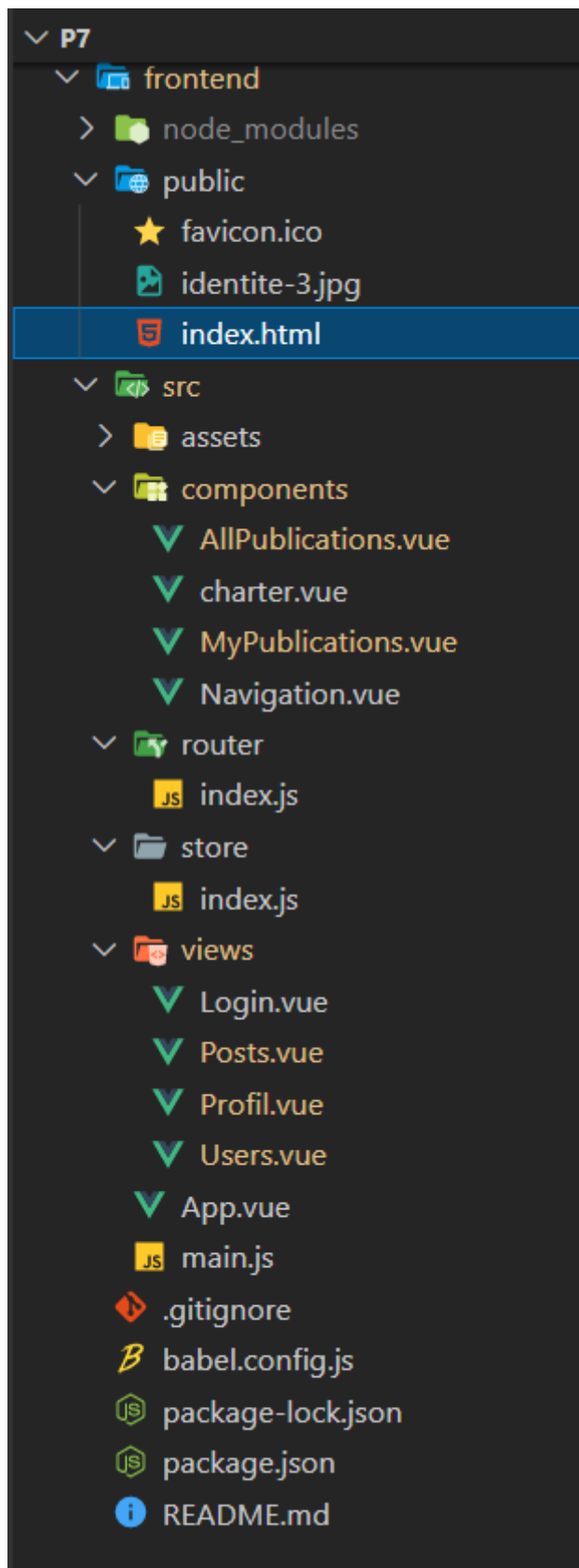
 <small>partagé par Rabah</small> <small>Le 01/01/2022 à 17:03:47</small>	de l'utilisateur ayant partagé le post et la date de partage ainsi que la publication d'origine.	
<div> <div>La page « utilisateurs »</div> <div>Utilisateurs </div> </div>		
Accès à la page 	L'accès à la page « Utilisateurs » se fait via l'icône  à partir du menu de navigation.	
Contenu Liste des salariés	La page affiche le nombre total de salariés inscrits ainsi que la liste sous forme de cartes « cliquables » .	
Le bouton de recherche 	Permet de faire une recherche d'un ou plusieurs salariés. Les cartes dont les pseudos des salariés commencent par les lettres tapées par l'utilisateur sont affichés simultanément. Le nombre de salariés dont le pseudo commence par la recherche se met à jour instantanément.	
La carte d'un salarié 	Contient la photo du profil de l'utilisateur et son pseudo. On distingue la carte de l'administrateur  et celles des comptes supprimés  . Au clic : l'utilisateur est dirigé vers la page du profil du salarié sélectionné !	
<div>La page « Mon Profil »</div>		
L'accès à la page « Mon profil »	1- En cliquant sur la photo du profil de l'utilisateur connecté via la page « forum »  2- En cliquant sur l'icône « Mon profil » du menu de navigation  depuis n'importe quelle page. 3- En cliquant sur une carte de n'importe quel utilisateur dans la liste des salariés via la page « Utilisateurs » 	
Contenu	<ul style="list-style-type: none"> - Photo de profil, pseudo, mail, Matricule, mention administrateur... - Les boutons « Modifier mon profil » et « Supprimer mon profil » - Bouton Afficher toutes les publications 	
<div>Modifier mon profil</div>		
Le bouton « Modifier mon profil » 	<ul style="list-style-type: none"> - Est présent uniquement si le profil correspond à l'utilisateur connecté Au clic : Un formulaire pour modifier l'Email, le mot de passe, le pseudo et la photo du profil	
Modifier L'email	<ul style="list-style-type: none"> - Une fois le champs remplis, le bouton « valider » apparaît. 	

<p>Modifier mon Email</p> <p>Votre nouvel Email</p>	<p>Au clic :</p> <p>Si mail invalide : message d'erreur : Votre nouvel email n'est pas valide!</p> <p>Si mail valide : L'email de l'utilisateur est modifié dans la base de données et instantanément sur la page</p>	
<p>Modifier le mot de passe</p> <p>Modifier mon mot de passe</p> <p>Ancien mot de passe Nouveau mot de passe</p>	<ul style="list-style-type: none"> - Trois champs doivent être remplis et validés : ancien mot de passe, nouveau mot de passe et répéter nouveau mot de passe. <p>Au clic sur valider, Erreurs possibles :</p> <ul style="list-style-type: none"> - Vos nouveaux mots de passe ne correspondent pas ! - Votre mot de passe est invalide! Au moins 8 caractères, une majuscule, un chiffre ! - L'ancien mot de passe est incorrect ! <p>Si pas d'erreurs : Le nouveau mot de passe haché et modifié dans la base de données.</p>	
<p>Modifier mon pseudo</p> <p>Modifier mon pseudo</p> <p>Nouveau pseudo</p>	<p>Au clic sur valider, Erreurs possibles :</p> <p>Vérifiez votre nom: au moins 3 caractères. 20 max !</p> <p>Si pas d'erreurs :</p> <ul style="list-style-type: none"> - Le pseudo est modifié dans toutes les tables de la base de données 	
<p>Modifier ma photo de profil</p> <p>Modifier ma photo de profil</p> <p>Choisir un fichier Aucun fichier choisi</p>	<p>Au clic sur valider :</p> <ul style="list-style-type: none"> - La photo du profil de l'utilisateur est modifiée instantanément sur la page, sur toutes ses publication, ses partages et ses commentaires - L'url de la photo est modifiée dans les tables « users », « posts » et « comments » de la base de données. - L'ancienne photo du profil est supprimée du dossier images 	
<p>Supprimer mon profil</p>		
<p>Le bouton « supprimer mon profil »</p> <p></p>	<p>Est présent uniquement si le profil correspond à l'utilisateur connecté</p> <p>Au clic :</p> <ul style="list-style-type: none"> - Un message d'avertissement : Attention, la suppression de votre profil est irréversible !!! <p> </p> <p>Au clic sur « Annuler », l'opération est annulée</p> <p>Au clic sur « Supprimer » :</p> <ul style="list-style-type: none"> - La photo du profil est remplacée dans toutes les tables de la base de données par l'« icone groupomania »  - L'ancien fichier photo est supprimé du dossier images - L'email de l'utilisateur est remplacé par « Groupomaniaid@groupomania » dans toutes les tables de la base de données - Remplacer le pseudo par « Groupomaniaid » dans toutes les tables de la base de données. 	<p>Pour des raisons de sécurité, de législation et de la réglementation RGPD, j'ai opté pour cette solution afin de garder les publications ainsi que les commentaires des utilisateurs ayant supprimés leurs comptes.</p> <p>J'ai choisi d'anonymiser les comptes supprimés par les utilisateurs et garder leurs publications ainsi que leurs commentaires pour d'éventuels besoins ultérieurs. Néanmoins, personne ne peut se connecter avec les nouvelles données. Seul</p>

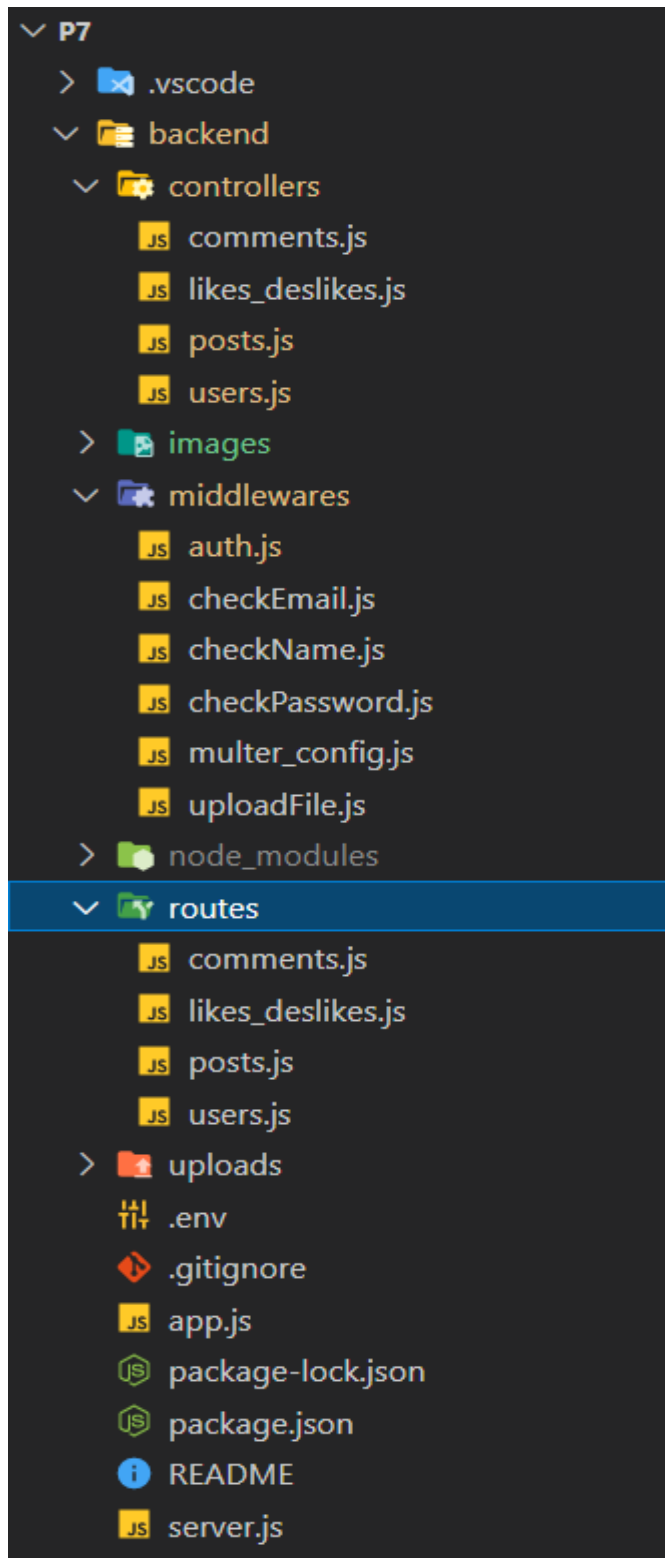
		<p>l'administrateur peut masquer ou supprimer leurs publications et commentaires.</p> <p>Une deuxième solution est préconisée à savoir supprimer le compte et toutes les publications et commentaires de l'utilisateur. Un code « commenté » est prévu à cet effet dans le fichier : <code>backend/controllers/users.js</code></p>
Mes publications		
<p>Le bouton « Afficher toutes les publications »</p> <p>Afficher toutes les publications</p>	<p>Au clic :</p> <ul style="list-style-type: none"> - Toutes les publications du profil sont affichées. - Le bouton se transforme en Masquer les publications - Toutes les fonctionnalités du forum sont disponibles sur la page « Mon profil » (Créer, Modifier, Supprimer, Partager, Commenter, Liker, Desliker ...) <p>Cas :</p> <p>1* L'utilisateur connecté visite son profil :</p> <ul style="list-style-type: none"> - Accès à toutes les fonctionnalités <p>2* L'utilisateur connecté visite un autre profil :</p> <ul style="list-style-type: none"> - Accès à (liker, desliker, commenter et partager) <p>3* L'administrateur sur un autre profil :</p> <ul style="list-style-type: none"> - Accès à toutes les fonctionnalités sauf la création et la modification. 	
SUGGESTIONS		
Inscription	Envoyer un mail avec un identifiant et un mot de passe pour se connecter	<p>L'inscription ne doit pas se faire automatiquement. L'administrateur doit vérifier l'identité de l'utilisateur. Il envoie un mail à l'utilisateur inscrit avec un identifiant et un mot de passe temporaires pour se connecter. L'utilisateur peut les modifier une fois inscrit.</p>

Boite email	Créer une adresse email personnalisée	L'administrateur peut octroyer des adresses email personnalisées au nom de l'entreprise telle que : @Groupomania.fr
Réseaux sociaux	Pouvoir se connecter à l'application via des comptes de réseaux sociaux (facebook, twitter, instagram, linkedin...)	Faciliter la connexion via les réseaux sociaux et améliorer l'expérience utilisateur
Charte d'utilisation	Intégrer une charte d'utilisation de l'application Pour se conformer aux exigences de la RGPD, La charte est à signer et accepter par les utilisateurs au moment de l'inscription (exemple)	<p>1/ Les modalités d'inscription et d'accès au forum</p> <p>2/ Les thèmes de discussion</p> <p>3/ Les données collectées</p> <p>4/ Les droits et devoirs des membres (Informé les utilisateurs sur l'utilisation de leurs données personnelles)</p> <p>5/ La pratique de la modération (Administrateur)</p> <p>6/ La mention légale sur le droit d'accès, de modification, de rectification et de suppression des données qui concernent les participants.</p> <p>7/ Les sanctions ...</p>
Compte modérateur	Création d'un compte « Modérateur »	L'administrateur peut élire un utilisateur au rang de « modérateur » pour s'occuper du masquage et de la suppression de publications ou commentaires ne respectant pas la charte de l'application.

3) La structure front-end



4) La structure back-end



5) La base de données MySql

A/ Les tables :

```
mysql> show tables;
+-----+
| Tables_in_groupomania |
+-----+
| comments               |
| likes_deslikes         |
| posts                  |
| users                   |
+-----+
4 rows in set (0.10 sec)
```

B/ La table « users »

```
mysql> describe users;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default        | Extra          |
+-----+-----+-----+-----+-----+-----+
| userId         | smallint unsigned | NO   | PRI | NULL           | auto_increment |
| email          | varchar(255)    | NO   | UNI | NULL           |                |
| password       | varchar(255)    | NO   |     | NULL           |                |
| isAdmin        | smallint        | NO   |     | 0              |                |
| name           | varchar(50)     | NO   |     | NULL           |                |
| profil_picture | varchar(255)    | YES  |     | _cp850\'icon.png\' | DEFAULT_GENERATED |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.01 sec)
```

C/ La table « posts »

```
mysql> describe posts;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default        | Extra          |
+-----+-----+-----+-----+-----+-----+
| postId         | smallint unsigned | NO   | PRI | NULL           | auto_increment |
| post           | text          | YES  |     | NULL           |                |
| image          | varchar(255)   | YES  |     | NULL           |                |
| user_send      | varchar(50)    | NO   |     | NULL           |                |
| userId         | smallint unsigned | NO   |     | NULL           |                |
| date_send      | datetime      | NO   |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED |
| profil_picture | varchar(255)   | YES  |     | NULL           |                |
| commentNumber  | smallint unsigned | YES  |     | 0              |                |
| likes_number   | smallint unsigned | YES  |     | 0              |                |
| deslikes_number | smallint unsigned | YES  |     | 0              |                |
| shared         | smallint unsigned | NO   |     | 0              |                |
| shared_userId  | smallint unsigned | YES  |     | NULL           |                |
| shared_number  | smallint unsigned | YES  |     | 0              |                |
| shared_date    | varchar(255)   | YES  |     | NULL           |                |
| shared_userName | varchar(50)    | YES  |     | NULL           |                |
| shared_userProfil_picture | varchar(255) | YES  |     | NULL           |                |
+-----+-----+-----+-----+-----+-----+
16 rows in set (0.00 sec)
```

D/ La table « comments »

```
mysql> describe comments;
```

Field	Type	Null	Key	Default	Extra
commentId	smallint unsigned	NO	PRI	NULL	auto_increment
comment	text	NO		NULL	
postId	smallint unsigned	NO		NULL	
userId	smallint unsigned	NO		NULL	
user_send	varchar(50)	NO		NULL	
date_send	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
profil_picture	varchar(255)	YES		icon.png	
masked	smallint	YES		0	

8 rows in set (0.01 sec)

E/ La table « likes_deslikes »

```
mysql> describe likes_deslikes;
```

Field	Type	Null	Key	Default	Extra
id	smallint unsigned	NO	PRI	NULL	auto_increment
postId	smallint unsigned	NO		NULL	
userId	smallint unsigned	YES		NULL	
likes_number	smallint unsigned	YES		0	
deslikes_number	smallint unsigned	YES		0	
user_status	smallint	YES		0	

6 rows in set (0.00 sec)

6) Spécifications de l'Application

L'URL de base : <http://localhost:3000>

Verbe Http	URI d'accès	Body de la requête	Réponse de l'API backend	Fonctionnalité
Les utilisateurs				
POST	/api/signup	{ email : string, Password : string Name : string }	{ message :string}	Hacher le mot de passe Ajouter l'utilisateur à la base de données (table users)
POST	/api/login	{ email : string, Password :string }	{ data :object }	Vérifier les informations utilisateur et renvoyer un objet contenant un token valable 1h avec les informations de l'utilisateur connecté.
GET	Api/userConnected/ :id		{ userConnected : object }	Un objet contenant les informations de l'utilisateur connecté
GET	Api/users		[users : array]	Affiche un tableau de tous les utilisateurs inscrits
GET	Api/users/ :id		{ oneUser : object }	Afficher le profil d'un utilisateur
PATCH	Api/users/photo/ :id		{ Image : string }	Modifier la photo du profil de l'utilisateur connecté
patch	Api/users/email/ :id	{ Email : string }	{ Message : string }	Modifier l'email de l'utilisateur connecté
PATCH	Api/users/name/ :id	{ Name : string }	{ Message : string }	Modifier le pseudo de l'utilisateur connecté
PUT	Api/users/password/ :id	{ Password1 : string, Password2 : string }	{ Message : string }	Modifier le mot de passe de l'utilisateur connecté
DELETE	Api/users/ :id		{ Message : string }	Supprimer le profil d'un utilisateur
Les publications				
POST	Api/posts	{		Publier un post

		Formdata : object { File : string, Post : string, User_send : string, userId : number profile_picture : string } }	[Publications : array]	Et affiche la nouvelle liste des publications
GET	Api/posts		[Publications : array]	Afficher toutes les publications
GET	Api/posts/:id		[Publications : array]	Affiche les publications d'un utilisateur
DELETE	Api/posts/ :id		[Publications : array]	Supprimer un post Et affiche la nouvelle liste des publications
PUT	Api/posts/ :id	{ Formdata : object { File : string, Text_modified : string, } }	[Publications : array]	Modifier un post Et affiche la nouvelle liste des publications
POST	Api/posts/share	{ Publication1 : object { Publication2 : object { Shared : number, Shared_userId : number, Shared_userName : string, Shared_userProfil_picture : string } } }	[Publications : array]	Partager un post Et affiche la nouvelle liste des publications

Les commentaires

POST	Api/comments	{ Comment : string, postId : number, userid : number, user_send : string, profile_picture : string }	[Publications : array]	Publier un commentaire Et affiche la nouvelle liste des publications
GET	Api/comments/ :id		[Publications : array]	Afficher tous les commentaires d'une publication
PATCH	Api/comments/ :id		[Publications : array]	Masquer un commentaire par l'administrateur
DELETE	Api/comments/ :id		[Publications : array]	Supprimer un commentaire

Les likes/Deslikes Comment :

POST	Api/like/ :id	{ postId : number, userId : number }	[Publications : array]	Liker (aimer) une publication Et affiche le nombre de likes et deslikes à nouveau
POST	Api/deslike/ :id	{ postId : number, userId : number }	[Publications : array]	Desliker (ne pas aimer) une publication Et affiche le nombre de likes et deslikes à nouveau

Conclusion

L'application Groupomania reste un MVP avec des fonctionnalités de base. Certaines solutions sont déjà codées.

D'autres (voir suggestions – plus haut) sont proposées pour améliorer l'expérience utilisateur et se conformer aux exigences du RGPD. D'autres encore peuvent être implémentées selon les besoins de l'entreprise et des employés.