Projet 5

Blog Professionnel

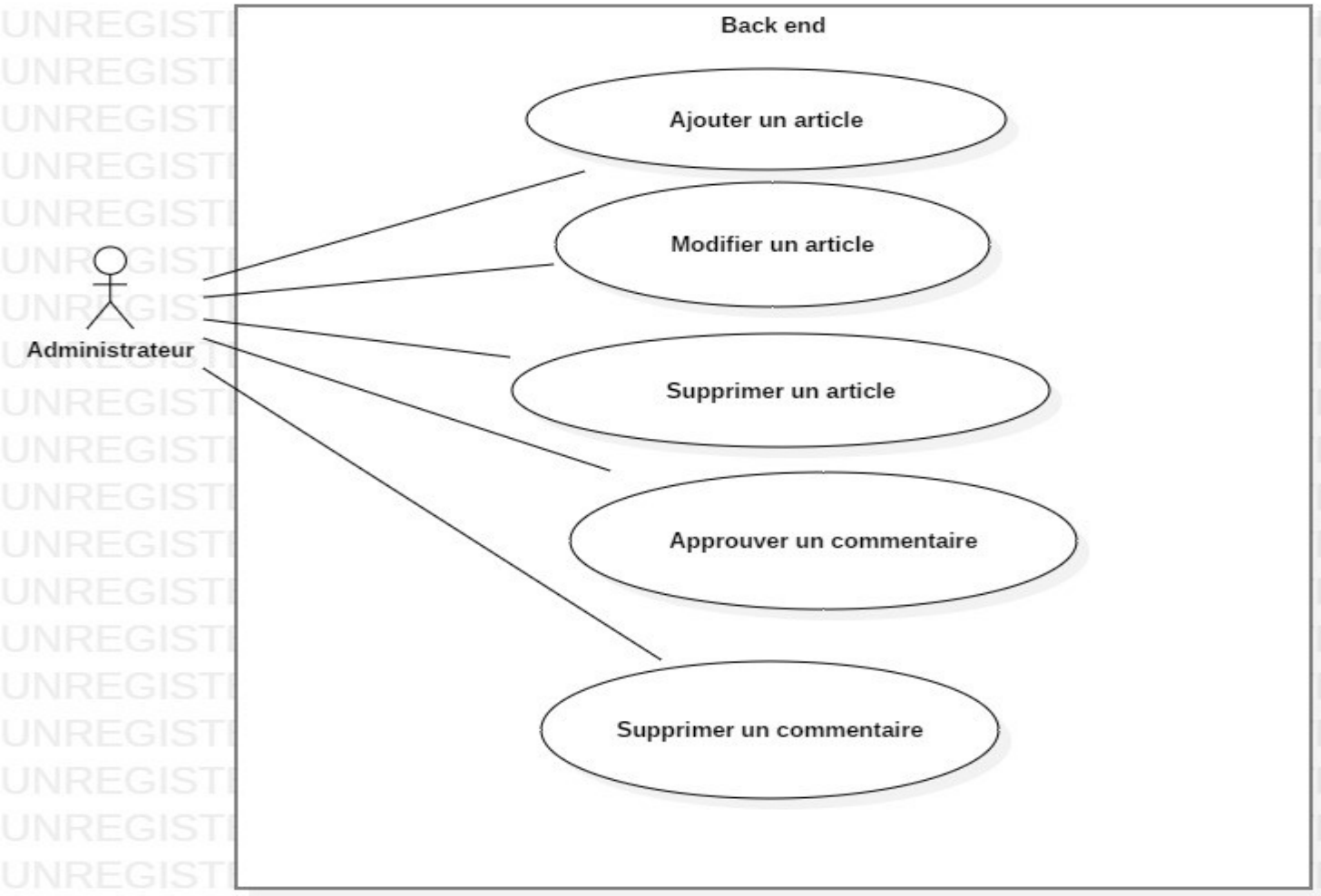
A) Contexte du projet

Création d'un blog professionnel composée de :

- Page d'accueil avec formulaire du contact
- Liste des articles
- Détail de chaque article avec possibilité de soumettre un commentaire soumis à validation
- Espace membres pour pouvoir laisser des commentaires
- Espace administrateur pour la gestion des articles (Crud) et la validation ou suppression des commentaires

B) Analyse du besoin pour l'administration des articles

- Diagramme de cas d'utilisation



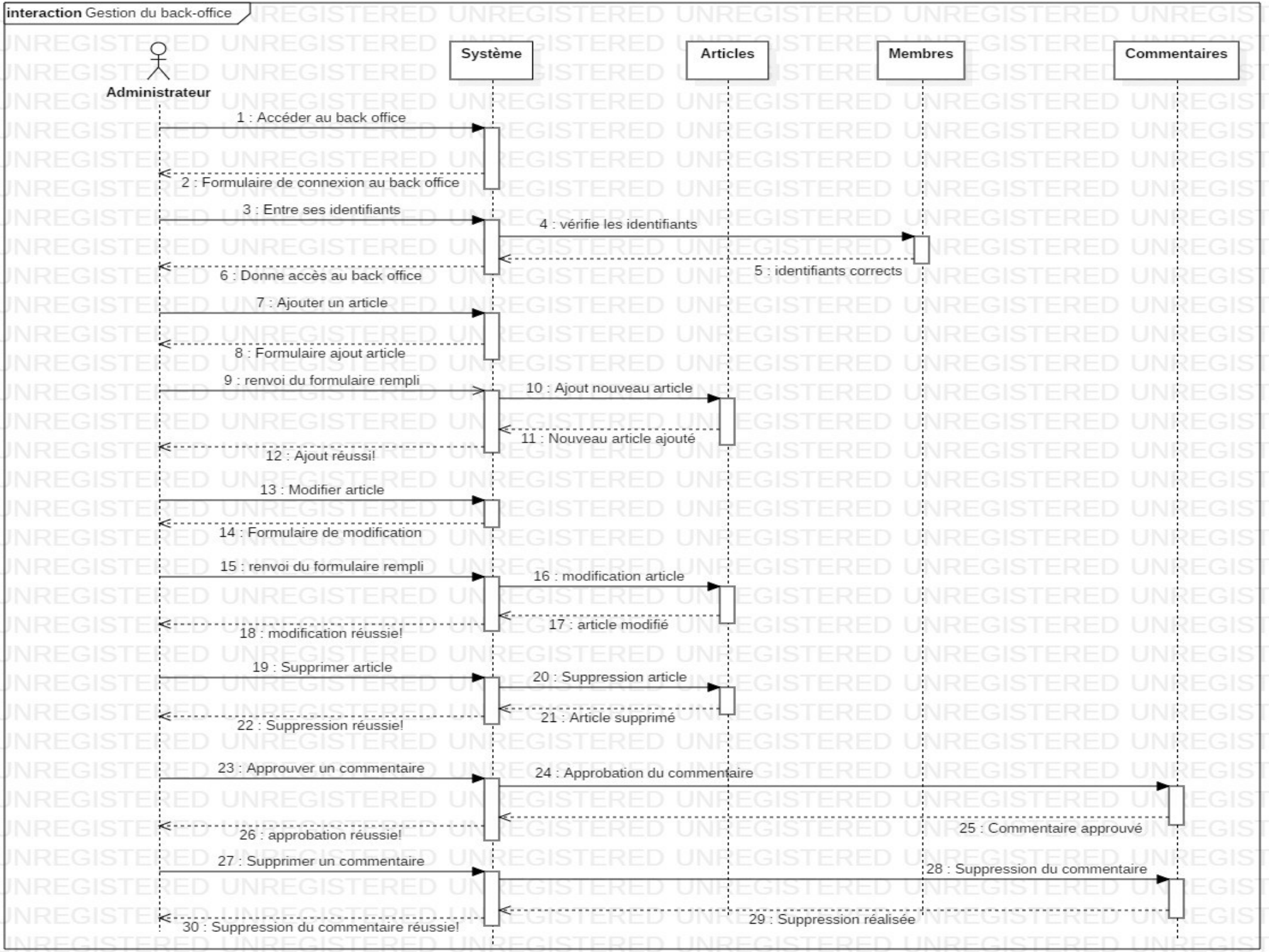
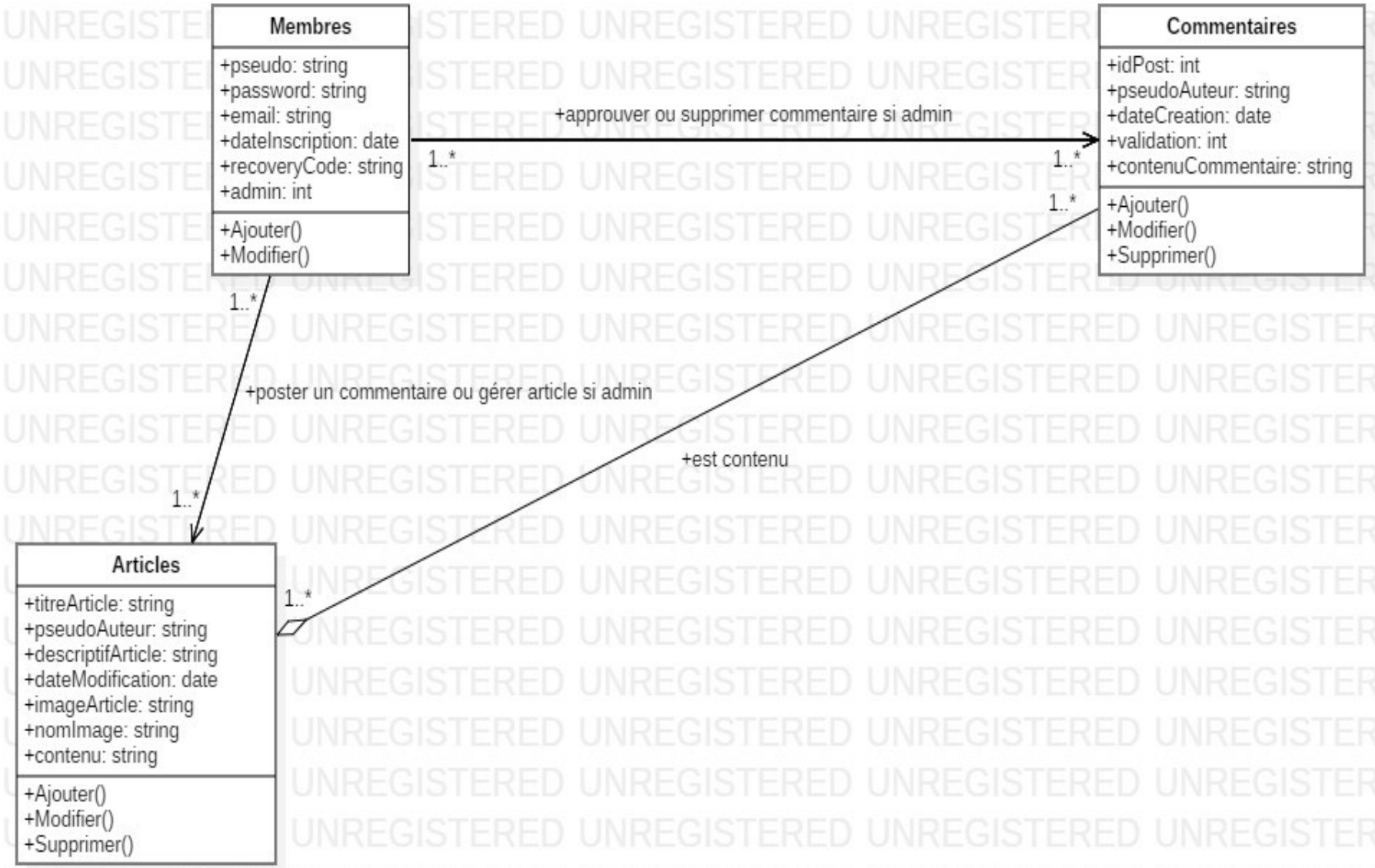

interaction Gestion du back-office
Administrateur
Système
Articles
Membres
Commentaires
1 : Accéder au back office
2 : Formulaire de connexion au back office
3 : Entre ses identifiants
4 : vérifie les identifiants
5 : identifiants corrects
6 : Donne accès au back office
7 : Ajouter un article
8 : Formulaire ajout article
9 : renvoi du formulaire rempli
10 : Ajout nouveau article
11 : Nouveau article ajouté
12 : Ajout réussi!
13 : Modifier article
14 : Formulaire de modification
15 : renvoi du formulaire rempli
16 : modification article
17 : article modifié
18 : modification réussie!
19 : Supprimer article
20 : Suppression article
21 : Article supprimé
22 : Suppression réussie!
23 : Approuver un commentaire
24 : Approbation du commentaire
25 : Commentaire approuvé
26 : approbation réussie!
27 : Supprimer un commentaire
28 : Suppression du commentaire
29 : Suppression réalisée
30 : Suppression du commentaire réussie!

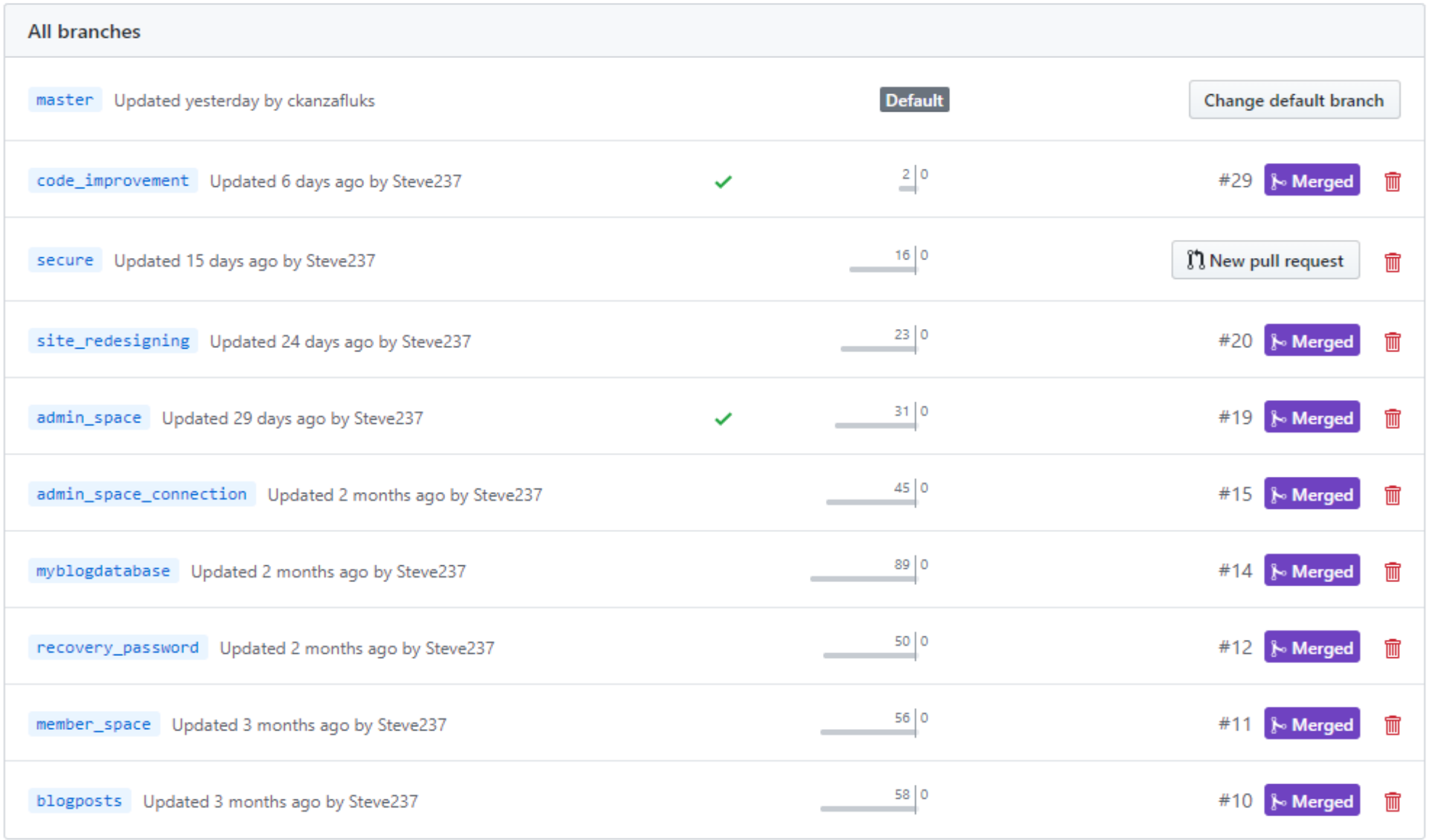
Diagramme de classe.



C) Organisation du projet

- Choix du template
- Création des issues
- Création d'une branche par tâche
- Réalisation de la tâche
- Transfert de la branche sur le dépôt distant
- Validation du code par le mentor
- Clôture des issues
- Analyse du code sur Codacy
- Apport de correctifs au code
- Réalisation des diagrammes et du fichier d'installation
- Soumission du projet

D) Versionning du projet.



E) Architecture technique du projet.

- Utilisation du pattern MVC (Modèle vue contrôleur) :

-Modèle : cette partie a pour fonction d'aller récupérer les informations dans la base de données, de les organiser et de les assembler pour qu'elles puissent ensuite être traitées par le contrôleur. Elle regroupe donc entre autres les requêtes SQL.

-Vue : cette partie se concentre sur l'affichage. Elle ne fait presque aucun calcul et se contente de récupérer des variables pour savoir ce qu'elle doit afficher. On y trouve essentiellement du code HTML mais aussi quelques boucles et conditions PHP très simples, pour afficher par exemple une liste de messages.

-Contrôleur : cette partie gère la logique du code qui prend des décisions. C'est en quelque sorte l'intermédiaire entre le modèle et la vue : le contrôleur va demander au modèle les données, les analyser, prendre des décisions et renvoyer le texte à afficher à la vue. Le contrôleur contient exclusivement du PHP. C'est notamment lui qui détermine si le visiteur a le droit de voir la page ou non (gestion des droits d'accès) .

- Utilisation des mediaqueries et de Bootstrap pour adapter le site en responsive.
- Langages utilisés : Html5, CSS3, PHP en programmation orienté objet, Javascript, Jquery

F) Mesures mis en place pour le respect des pratiques en vigueur

- Les mesures que nous avons utilisés pour le respect des normes de codages sont :

-L'indentation du code.

-L'écriture du nom des variables et fonctions en Camelcase (PSR-1).

-L'écriture du nom des classes en StudlyCaps (PSR-1).

-Utilisation de Autoload pour le chargement des classes (PSR-1).

-Écriture des structures de contrôle (else, elseif, etc...) conformément au PSR-2

-Sécurisation du site contre les failles de sécurité (failles XSS, CRLF, injection sql, la CSRF, session hijacking, hashage des mots de passes en base de données) .

-Redirection vers page 404 en cas d'erreur de requête.

-Commentaires des fonctions des modèles.

-Utilisation du pattern MVC.

-Adaptation du site pour tous les types d'écran (smartphone, tablette, pc) .

-Utilisation d'outil de versionning.

Avoid variables with short names like $db. Configured minimum length is 3.

```
81 $db  = $this->dbconnect();
```

model/UsersManager.php

Avoid variables with short names like $q. Configured minimum length is 3.

```
14 $q = $db->prepare('INSERT INTO membres(pseudo, password, email, dateInscription) VALUES(:pseudo, :password, :email, NOW())');
```

```php
<?php

class CommentManager extends Manager
{

    /**
     * permet d'afficher la liste des commentaires
     */
    public function getListComment()
    {
        $dtb = $this->dbconnect();
        $query = $dtb->query('SELECT idCommentaire, idPost, pseudoAuteur, contenuCommentaire, DATE_FORMAT(dateCreation, "%d/%m/%Y %Hh%imin%ss") AS
        dateCreation, validation FROM commentaires');
        $query->execute();


        $comments = $query->fetchAll(PDO::FETCH_ASSOC);

        $listComments = array();
        foreach ($comments as $donnees)
        {
            // on instancie notre objet
            $comments = new Comments();
            // on hydrate notre objet avec les valeurs récupérées en bdd
            $comments->hydrate($donnees);
            // puis on le met dans notre tableau
            $listComments[] = $comments;
        }
        return $listComments;
    }
```

G) Présentation de la réalisation de la tâche espace administrateur

La tache création du back-office a été réalisé en plusieurs étapes :

-Affichage de la liste des articles à gérer

-Création des boutons pour ajouter, modifier, supprimer un article.

-Création des formulaires d'ajout, et de modification des articles.

-Écriture des fonctions dans le modèle et le contrôleur pour le fonctionnement de l'ajout, modification, ou suppression des articles.

-Création d'un lien pour accéder à l'espace de gestion des commentaires.

-Affichage de la liste des commentaires à gérer dans l'espace gestion des commentaires.

-Création des boutons pour approuver ou supprimer des commentaires.

-Écriture des fonctions dans le modèle et le contrôleur pour le fonctionnement de l'approbation ou la suppression des commentaires.

-Embellissement de l'affichage dans l'espace administrateur en Css3