Licence 3

MIAGE Apprentissage

Rapport de projet

**BELLAHSENE Isame**

**EL JAYI Bilal**

**ZAÏRE Jérémy**

# Sommaire

[Sommaire 2](#_Toc420493404)

[1. Introduction 3](#_Toc420493405)

[2. Etude du projet 3](#_Toc420493406)

[2.1. Situations initiales 3](#_Toc420493407)

[2.2. Modèle physique de données 4](#_Toc420493408)

[2.3. Objectifs à réaliser 5](#_Toc420493409)

[3. Elaboration du projet 6](#_Toc420493410)

[3.1. Les différents phases 6](#_Toc420493411)

[3.2. Répartition du travail 7](#_Toc420493412)

[3.3 Problèmes et solutions envisagées 7](#_Toc420493413)

[4. Résultats 7](#_Toc420493414)

[4.1. Etat final du projet 7](#_Toc420493415)

[4.2. Améliorations possible 8](#_Toc420493416)

[5. Conclusion 8](#_Toc420493417)

# 1. Introduction

Dans le cadre de notre licence 3 MIAGE en apprentissage, il nous ai demandé de réaliser un projet en PHP en rapport avec ce que nous avons appris durant le semestre et mettre en application les différents procédés de manipulation d'une base de données.

L'objectif de ce projet est de réaliser la conception et le développement d'un blog communautaire ergonomique, paramétrables et évolutif.

Ainsi un utilisateur quelconque doit pouvoir s'inscrire et s'y connecter comme il le souhaite pour y ajouter du contenu et commenter le contenu d'un autre utilisateur/inscrit.

Pour réaliser ce projet nous composons, une équipe de 3 étudiants issue d'une formation MIAGE. Les connaissances acquises au cours de l'année en base de données, PHP et Bootstrap nous permettrons de répondre à cette attente.

# 2. Etude du projet

## 2.1. Situations initiales

Pour réussir le développement de notre site nous avons utilisé Verbatelo pour illustrer notre base :

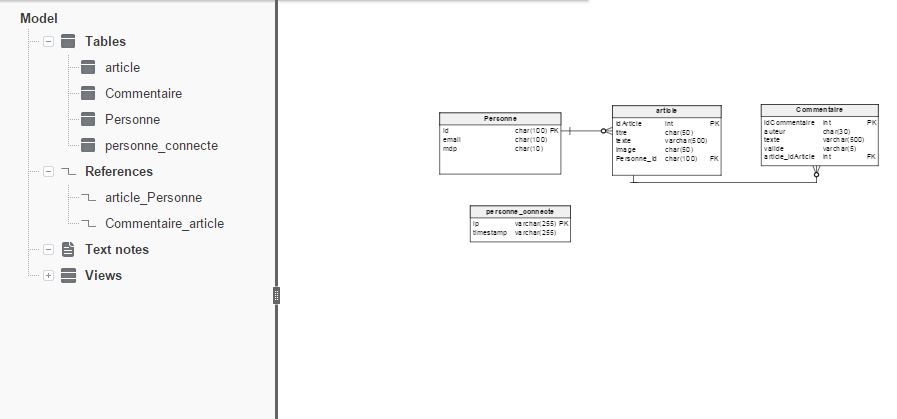


Figure 1 – Schéma de la BDD

Nous avons voulu laisser la possibilité à un utilisateur de laisser un commentaire sans s’identifier. Car nous avons pensé que laisser la possibilité à un utilisateur de s’inscrire ou non était important.

Ce projet, pour être le plus cohérent possible doit être réalisé en association avec une base de données, ainsi avant de débuter directement dans le développement il aura fallu réfléchir à la conception de notre site. Le blog doit contenir plusieurs "parties" obligatoires pour fonctionner correctement, nous appellerons ces "parties", des entités. Elles sont au nombre de quatre :

* Une entité Article
* Une entité Personne
* Une entité Commentaire
* Une entité Personne\_Connecté correspondant compteur de personnes qui visitent le site

Pour garder une cohérence complète dans la conception de notre site et les relations entre les tables, il est nécessaire d'avoir une clé primaire et des clés étrangères en fonction des différents types de relation.

Pour réaliser notre site web plusieurs points vus en cours sont à respecter, notre blog doit obligatoirement contenir :

* Un serveur Apache
* Des scripts PHP
* Des pages HTML
* Des feuilles de style CSS
* Une Base de données relationnelle
* Une authentification par session des utilisateurs
* Des concepts web avancées

## 2.2. Modèle physique de données

CREATE TABLE IF NOT EXISTS `article` (

`idArticle` int(100) NOT NULL,

`titre` char(50) NOT NULL,

`texte` varchar(500) NOT NULL,

`image` char(50) NOT NULL,

`idPersonne` char(100) NOT NULL

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO\_INCREMENT=25 ;

CREATE TABLE IF NOT EXISTS `commentaire` (

`idCommentaire` int(100) NOT NULL,

`auteur` char(30) NOT NULL,

`texte` varchar(500) NOT NULL,

`valide` varchar(5) NOT NULL,

`idArticle` char(100) NOT NULL

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO\_INCREMENT=74 ;

CREATE TABLE IF NOT EXISTS `personne` (

`id` char(100) NOT NULL,

`email` char(100) NOT NULL,

`mdp` char(10) NOT NULL

) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE IF NOT EXISTS `personne\_connecte` (

`ip` varchar(255) COLLATE latin1\_german2\_ci NOT NULL,

`timestamp` varchar(255) COLLATE latin1\_german2\_ci NOT NULL

) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1\_german2\_ci;

ALTER TABLE `article`

ADD PRIMARY KEY (`idArticle`), ADD KEY `idPersonne` (`idPersonne`);

ALTER TABLE `commentaire`

ADD PRIMARY KEY (`idCommentaire`), ADD KEY `idArticle` (`idArticle`);

ALTER TABLE `personne`

ADD PRIMARY KEY (`id`);

ALTER TABLE `article`

MODIFY `idArticle` int(100) NOT NULL AUTO\_INCREMENT,AUTO\_INCREMENT=25;

ALTER TABLE `commentaire`

MODIFY `idCommentaire` int(100) NOT NULL AUTO\_INCREMENT,AUTO\_INCREMENT=74;

## 2.3. Objectifs à réaliser

Plusieurs objectifs sont à réaliser afin que notre blog soit considéré comme valide. Dans un premier temps, donner la possibilité à un visiteur de s'inscrire pour rendre le blog évolutif.

Il pourra alors s'identifier par un login et un mot de passe pour pouvoir y ajouter un article.

Dans un deuxième temps, l'ordre d'apparition des articles doit se faire du plus récent au plus vieux afin qu'un utilisateur lambda, voit directement les dernières modifications. Il pourra y laisser des commentaires qui seront validés par la personne concerné (c'est-à-dire l'auteur de l'article)

Lorsque le blog devient de plus en plus complet, il devient difficile de lire tout les articles sur une page, ainsi le blog doit pouvoir être séparé en plusieurs pages facilitant la navigation.

Pour finir le blog doit contenir un compteur de visite journalier, global et un compteur de personnes connectées.

Pour éviter les bots, il est aussi obligatoire de mettre un système de captcha pour la validation des articles.

# 3. Elaboration du projet

## 3.1. Les différents phases

La première phase aura été la phase de conception, en effet malgré que le projet demandé ne soit pas complexe en terme de données, il fallait réfléchir pour savoir comment organiser notre base et comment l'utiliser pour ne pas avoir à tout reprendre. Ainsi ne pas démarrer rapidement mais en prenant le temps poser les bases, de réfléchir à la conception nous permettra de ne pas perdre le fil.

Pour la conception de la base de données, nous avons utilisé l'outil Vertabelo.

Puis viens la phase de développement, pour cette étape nous avons choisi de mettre en place le projet en intégration continue en utilisant des outils tel que TortoiseSVN et le site Github pour héberger nos sources.

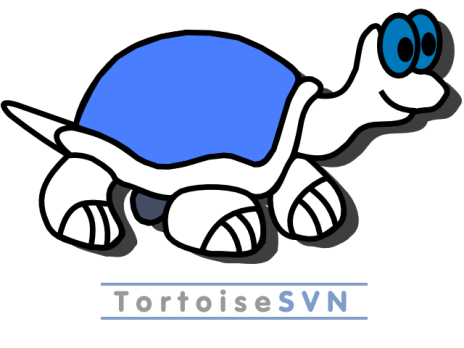


Figure 2 - TortoiseSVN

Cette solution nous a permis de travailler en même temps et séparément sur les mêmes fichiers sans perdre notre travail, sans conflit de sources et sans perte de temps pour assembler les fichiers.

Figure 3 - GitHub

Cette méthode de travail a été choisie à l'unanimité vu l'efficacité que cela permet d'avoir dans un projet en groupe.

## 3.2. Répartition du travail

Le rythme de l'alternance ne permet pas de se synchroniser facilement, ainsi le cours de Projet SI chaque semaine permettait de faire un point facilement sur notre avancement.

Malgré la possibilité de se voir chaque semaine, la mise en place d'une intégration continue était nécessaire mais pas obligatoire.

Concernant la répartition des tâches, nous nous sommes partagés chaque objectif.

L'intégration continue nous as permis de suivre l'évolution du projet, sans se faire avoir par le temps.

## 3.3 Problèmes et solutions envisagées

Le premier problème que nous avons rencontré est la synchronisation de nos sources au début du développement. Ainsi nous avons opté pour l'intégration continue qui à réglé ce problème.

En début de projet, lors de la création de la base de données, nous avons eu des problèmes pour le partage de la structure. Si l'un d'entre nous modifiait la structure sans commit sur GitHub, le site pouvait devenir inaccessible lors de la récupération des sources.

En fin de projet, l'un de nous n'avait plus de connexion internet ce qui a rendu les commit du projet difficile. Le travail se faisait alors en local de son côté avant de commit ses fichiers en entreprise ou à l'université.

# 4. Résultats

## 4.1. Etat final du projet

Le projet à été réalisé avec succès, les objectifs ont été remplies, nous avons même ouvert un site gratuit en ligne pour héberger notre projet visible sur : <http://webzaire.uphero.com>

Le site web est aussi ergonomique puisqu'il facilite la navigation, par exemple la fermeture automatique de la fenêtre lors de l'ajout d'un commentaire.

Notre projet a été développé en utilisant Bootstrap ce qui permet de le rendre portable et accessible sur un grand nombre de plateforme.

## 4.2. Améliorations possible

Une amélioration envisageable proposé dans le sujet qui n'a pu être exploré par manque de temps, est l'ajout d'un espace administrateur. Permettant à une personne de surveiller le blog. Il pourrait alors vérifier les articles avant la publication, supprimer des articles trop vieux ou malveillant, supprimer ou bannir des utilisateurs via leur adresse IP ou encore donner des droits supplémentaires à des utilisateurs réguliers.

Une amélioration supplémentaire serait de pouvoir ajouter un autre type de contenu dans les articles, comme par exemple un lecteur vidéo intégré (lecteur youtube) ou un lecteur de musique MP3 (type soundcloud) pour partager sa musique.

Un éditeur de texte pour la rédaction des articles permettrait aux utilisateurs de diversifié les articles et de rendre le blog encore un peu plus accrocheur et la possibilité de redimensionner une image que l'utilisateur importe.

Enfin une amélioration visuelle peut être effectué pour rendre le site web plus attractif, pour donner envie à l'utilisateur de partager des articles qui l'intéresse.

# 5. Conclusion

Ce projet fut l'occasion d'avoir une approche un peu plus professionnelle des bases de données. C'est l'occasion de faire à petite échelle, ce qui pourrait nous être demandé au sein d'une entreprise à l'avenir. Le fait de commencer sur un sujet comportant peu d'entité (quatre sur ce projet) permet d'apprendre correctement les mécanismes et les relations qui peuvent exister dans la gestion d'une base de données. Et ainsi mettre en application ce que nous avons appris, pour des sites web beaucoup plus complexe et plus important tout en utilisant des langages de programmation objet.