

Projet V5.4.0



Cette œuvre est mise à disposition selon les termes de la [licence Creative Commons Attribution – Pas d'Utilisation Commerciale – Partage à l'Identique 3.0 non transposé](#).

Document en ligne : www.mickael-martin-nevot.com

Date de rendu : **03/11/2019**

Travail : **groupe de quatre**

1 Assistance

Vous pouvez contacter l'enseignant en cas de besoin en formalisant et en ciblant précisément votre demande. Pour ce faire, vous devez respecter les règles de communication et d'envoi (ci-dessous).

2 Communication et envoi

2.1 Généralités

En joignant vos coordonnées (*e-mail* et téléphone portable notamment) à un message ou à votre livraison, vous pourrez être joint en cas de problème.

2.2 Communication

Chaque communication devra être faite :

- à destination de votre enseignant responsable :
 - Amélie Duvernet : amelieonline@gmail.com ;
 - Olivier Gérard : oliv.gerard@gmail.com ;
 - Philippe Mondou : philippe.mondou@gmail.com ;
 - Mickaël Martin-Nevot : mmartin.nevot@gmail.com ;
- en faisant figurer [AMU][IUT][S3] en début de sujet.

2.3 AMeTICE

2.3.1 Présentation

Pour votre projet, vous devez utiliser la plateforme pédagogique de l'Université d'Aix-Marseille : **AMeTICE** (<http://ametice.univ-amu.fr>), à l'adresse suivante : Accueil / Cours / IUT / DUT – secteur Électronique – Informatique Mécanique / DUT – Informatique (INFO) – Aix-en-Provence / 2^e année / [19]–S3–WIF200–201–M3104– Programmation Web côté serveur (M. Martin-Nevot).

2.3.2 Sélection du groupe

Sélectionnez, sur AMeTICE, le **groupe** d'étudiant auquel vous avez été affecté.

2.3.3 Livraison

Votre livrable devra être :

- nommé de la manière suivante (Nom1, Nom2, Nom3, Nom4 étant vos noms et Prénom1, Prénom2, Prénom3, Prénom4 vos prénoms) : Nom1 Prénom1 – Nom2 Prénom2 – Nom3 Prénom3 – Nom4 Prénom4 ;
- compressé dans une seule archive au format ZIP **n'excédant pas 10 Mo** ;
- remis, avant la date de rendu, sur AMeTICE.

3 Disponibilité

Votre site Web doit rester disponible jusqu'à la fin de votre année universitaire de formation. En outre, l'ensemble des identifiants de connexion du site Web et de la base de données doivent également rester valides durant toute cette période.

4 Sujet

4.1 Présentation

Réseau social d'un nouveau genre, **FreeNote** consiste à créer des fils de discussions comprenant des messages participatifs au sein desquels chaque utilisateur ne peut ajouter qu'un ou deux mots.

Exemple

L'utilisateur 1 envoie « Je ». Si aucune discussion n'est ouverte elle est créée et l'entrée de l'utilisateur est le premier mot du premier message de cette discussion.

L'utilisateur 2 répond « n'ai ». Le message se complète et est ainsi devenu : « Je n'ai ».

L'utilisateur 3 ajoute : « rien compris ». Le message devient « Je n'ai rien compris ».

Etc.

4.2 Organisation générale de l'application

4.2.1 Page d'accueil

La page d'accueil doit présenter les caractéristiques suivantes :

- une description du service ;
- la liste des discussions accessibles, avec marquage du statut de celles-ci, ouvert ou fermé (avec une pagination allant de deux en deux par défaut mais paramétrable par le super administrateur) ;
- chaque discussion pointe vers une page dédiée ;
- un système de connexion/déconnexion avec lien vers le compte utilisateur, lequel affiche les informations de celui-ci et la possibilité de les modifier ;
- un système de **mot de passe oublié ou perdu** (au choix : possibilité de réinitialiser un

nouveau mot de passe et/ou par *e-mail*, génération automatique d'un mot de passe temporaire ou non, etc.).

4.3 Détail des fonctionnalités

4.3.1 Fonctionnement général

- un utilisateur ne peut intervenir qu'une seule fois dans un message ;
- n'importe quel utilisateur peut clore un message à tout moment après y avoir inséré un mot ;
- lorsqu'un message est clos, un suivant est créé dans la discussion en cours ;
- une discussion est close selon un critère à définir et paramétrable (nombre de messages, nombre de messages aléatoire, etc. : à votre convenance) ;
- vous devez déterminer combien de discussions peuvent être ouvertes simultanément (avec une limite maximum paramétrable) ;
- optionnellement, un système de vote permet de mettre en avant les meilleures discussions sur la page d'accueil.

4.3.1.1 Questions à vous poser

- Comment gérer de manière efficace le suivi utilisateur dans un message au fil du temps.
- Détection des abus et tentatives de contournement.

4.3.2 Comptes utilisateurs

La page compte utilisateur doit présenter les caractéristiques suivantes :

- nom (ou pseudo) ;
- *e-mail* ;
- mot de passe (justifiez les mesures de sécurité mises en œuvre dans votre fichier `lisez-moi.txt`).

4.3.3 Rôles

Voici les différents rôles :

- (super) administrateur (tous les droits) ;
- membre **FreeNote** (disposant d'un compte avec des privilèges limités) ;
- visiteur (tous publics).

4.3.4 Privilèges/rôles

Voici la répartition des rôles par rapport aux privilèges :

| | |
|------------------------|---|
| (Super) administrateur | Tous les privilèges : <ul style="list-style-type: none"> – CRUD¹ utilisateurs, discussions, messages etc. ; – ne peut pas modifier le mot de passe d'un membre. |
| Membres | <ul style="list-style-type: none"> – CR (C avec limites du paramétrage du nombre maximum de discussions possibles) discussions comprenant donc les messages de celles-ci ; – CRU message avec une limite de une fois pour l'<i>update</i> (ajout d'un mot). |
| Visiteurs | <ul style="list-style-type: none"> – R discussions. |

¹ CRUD : *create, read, update, delete*.

4.3.5 Données

Jeu de test des discussions

Un jeu de test de dix discussions sera mis en place.

4.4 Evènement perturbateur (optionnel)

Un élément peut venir perturber le bon déroulement de l'avancement de votre projet à tout moment : malheureusement il est imprévisible, aussi bien sur sa date d'apparition que sur son objet. Vous savez seulement que vous devrez alors adapter l'application.

5 Réalisation

Vous devez réaliser, **sans** système de gestion de contenu préexistant (CMS open source) ou *framework* PHP, votre projet ; et livrer à votre enseignant responsable, avant la date de rendu :

- le **code source** (sans ressources binaires) correctement **commenté** ;
- un fichier nommé `lisez-moi.txt` succinct comportant **au minimum** :
 - le **nom** de l'équipe et de tous ces membres ;
 - l'URL de l'**index** du site Web ;
 - une **présentation** concise du projet réalisé ;
 - les **choix techniques** (technologies, algorithmes, etc.) et leurs raisons ;
 - la **configuration logicielle** minimale et conseillée ;
 - l'ensemble des **identifiants de connexion** :
 - du site Web (au minimum un compte de test et un d'administration) ;
 - de la base de données ;
 - de l'hébergement ;
 - du compte du système de version (**obligatoire**).

Vous utiliserez pour cela, au **minimum**, PHP, MySQL, HTML et CSS.

Vous devez apporter un soin tout particulier à la **programmation orientée objet** et à l'utilisation du modèle **MVC** et éventuellement de divers **modèles de conception**.

Votre application doit être **évolutive**, **modulaire** et professionnelle (**robuste**, **fiable** et intégralement **fonctionnelle**).

En outre, vous devrez impérativement utiliser des comptes reliés à un **système de version** (Git). Chaque membre du groupe disposera d'un compte permettant de tracer son activité. La gestion du temps et le partage des tâches donneront lieu à une **évaluation**.

Enfin, vous devez respecter les impératifs suivants :

- bonne indentation et commentarisation de l'ensemble des codes sources ;
- bonne architecture des répertoires sources ;
- validation W3C de toutes vos pages HTML (documents au minimum de type HTML5) ;
- validation W3C de toutes vos pages CSS (documents au minimum de profil CSS niveau 3 avec aucun avertissement et en tenant compte des extensions propriétaires comme avertissement).

6 Conseils

Voici quelques conseils :

- faites une recherche des solutions concurrentes ;
- n'oubliez pas de tester le site Web sur différents systèmes d'exploitation et différents navigateurs Web (en utilisant des versions portables ou des services Web comme <http://www.browsershots.org/>) ;
- pour les redoublants et ceux qui sont à l'aise, vous pouvez mettre en place quelques fonctionnalités intéressantes supplémentaires. À faire **uniquement** après avoir terminé tous les points essentiels notés ci-dessus et après avoir optimisé et approfondi les fonctionnalités demandées ;
- pensez à utiliser des services gratuits de stockage de ressources en ligne (Youtube, ImageShack, Scribd, etc.) afin d'éviter d'encombrer inutilement la taille restreinte du projet (et, bien souvent, de gagner en performance) ;
- faites un site Web adaptatif (*responsive*) pouvant être notamment utilisable sur PC, *smartphone* et tablettes ;
- réalisez quelques éléments d'une bibliothèque (logicielle) ou même un *framework* minimaliste ;
- utilisez des systèmes de gestion de version (Git : Bitbucket et <https://try.github.com> ; etc.), des *frameworks* de test unitaire (PHPUnit, etc.) et d'intégration continue (Jenkins, TravisCI, etc.) ;
- activez le SSH par défaut pour votre solution d'hébergement ;
- utilisez les outils de développement de vos navigateurs.