BTS SERVICES INFORMATIQUES AUX ORGANISATIONS **SESSION 2023**

Épreuve E5 - Conception et développement d'applications (option SLAM)

Fiche descriptive de réalisation professionnelle (recto)

DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE	**N° réalisation : 1**
Nom, prénom : LAU-TAI Dimitry	**N° candidat : 01948596434**
Épreuve ponctuelle ☐ **Contrôle en cours de formation** ☒	**Date :** 04 / 05 / 2023

Organisation support de la réalisation professionnelle
StarPlay

Intitulé de la réalisation professionnelle
Blog pour un streamer

Période de réalisation : SIO2 Semestre 2 **Lieu :** Maison
Modalité : ☐ **Seul(e)** ☒ **En équipe**

Compétences travaillées
- ☒Concevoir et développer une solution applicative
- ☒Assurer la maintenance corrective ou évolutive d'une solution applicative
- ☒Gérer les données

Conditions de réalisation[1] (ressources fournies, résultats attendus)

Ressources fournies :
Développement à partir de zéro

Résultat attendu :
Un blog qui nous permettrait de mettre en ligne des articles sur des thèmes divers à l'image de nautiljon.com ou encore jeuxvideo.com

Description des ressources documentaires, matérielles et logicielles utilisées[2]
Logiciel Utilisé :

IDE : Microsoft Visual Studio Code + Doki Theme (Pour coloration syntaxique)
PhpMyAdmin : 5.2
PHP: 8.2
Apache: 2.4.54.2
MySQL: 8.0.31

Modalités d'accès aux productions[3]et à leur documentation[4]
Lien GitHub du projet :

https://github.com/StephD974/E5Blog

Lien OneDrive :

ProjetsE5

Utilisateurs pour le blog :

ChizuSama || Root1234 (Compte Modérateur)
Destroy || admin (Compte Utilisateur)

[1]En référence aux *conditions de réalisation et ressources nécessaires* du bloc « Conception et développement d'applications » prévues dans le référentiel de certification du BTS SIO.

[2]Les réalisations professionnelles sont élaborées dans un environnement technologique conforme à l'annexe II.E du référentiel du BTS SIO.

[3]Conformément au référentiel du BTS SIO « *Dans tous les cas, les candidats doivent se munir des outils et ressources techniques nécessaires au déroulement de l'épreuve. Ils sont seuls responsables de la disponibilité et de la mise en œuvre de ces outils et ressources. La circulaire nationale d'organisation précise les conditions matérielles de déroulement des interrogations et les pénalités à appliquer aux candidats qui ne se seraient pas munis des éléments nécessaires au déroulement de l'épreuve.* ». Les éléments peuvent être un identifiant, un mot de passe, une adresse réticulaire (URL) d'un espace de stockage et de la présentation de l'organisation du stockage.

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS **SESSION 2023**

Épreuve E5 - Conception et développement d'applications (option SLAM)

Fiche descriptive de réalisation professionnelle (verso, éventuellement pages suivantes)

Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs

Contexte :

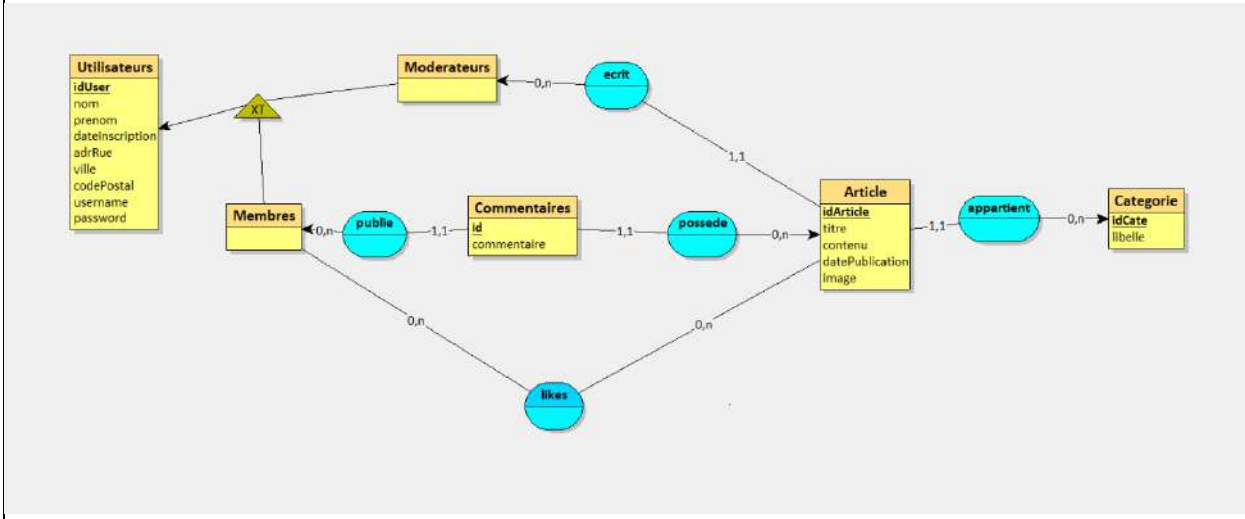
La société StarPlay ayant récemment recruté un nouveau streamer populaire. Le streamer a une grande communauté de fans et souhaite interagir davantage avec eux en dehors des flux de jeu en direct. Pour cela, il a besoin d'un blog personnel où il peut partager ses pensées, ses opinions, ses conseils de jeu et ses dernières nouvelles avec ses fans.

La société contacte donc une entreprise de développement web pour créer un blog personnalisé pour le streamer. Ils veulent que le blog soit facile à utiliser, convivial et reflète l'identité du streamer. Ils ont déjà une ébauche de ce qu'ils veulent que le blog comporte.

La société estime que le développement d'un blog personnalisé permettra au streamer d'interagir plus efficacement avec sa communauté de fans, tout en renforçant la présence de la marque de la société sur le marché des jeux vidéo en ligne.

StarPlay souhaite que le site ne comporte aucun moyen de promouvoir un modérateur, ils souhaitent que l'ajout de modérateur ce fasse manuellement dans la base de données à la suite d'un entretien avec le candidat.

Modèle conceptuel des données :



[4]Lien vers la documentationcomplète, précisant et décrivant, si cela n'a été fait au verso de la fiche, la réalisation professionnelle, par exemples service fourni par la réalisation, interfaces utilisateurs, description des classes ou de la base de données.

Utilité des tables :

- La table **"utilisateurs"** stocke les informations de base sur les utilisateurs, telles que leur nom, prénom, date d'inscription, adresse, ville, code postal, nom d'utilisateur et mot de passe. Chaque utilisateur a un identifiant unique représenté par le champ "id".

- La table **"membres"** est une table de liaison qui relie les utilisateurs qui sont membres à leur identifiant unique "idUser".

- La table **"modérateurs"** est également une table de liaison qui relie les utilisateurs qui sont modérateurs à leur identifiant unique "idUser".

- La table **"catégorie"** stocke les informations sur les catégories d'articles.

- La table **"article"** stocke les informations sur les articles, telles que leur titre, leur contenu, leur date de publication, leur image, l'identifiant unique de la catégorie à laquelle l'article appartient et l'identifiant unique de l'utilisateur qui a créé l'article.

- La table **"commentaires"** stocke les informations sur les commentaires, telles que le commentaire lui-même, l'identifiant unique de l'article auquel le commentaire se réfère et l'identifiant unique de l'utilisateur qui a créé le commentaire.

- La table **"likes"** est une table de liaison qui relie les utilisateurs qui ont aimé un article à l'identifiant unique de cet article.

Trigger et Procédures :

```
DROP PROCEDURE IF EXISTS `addNewUser`;
DELIMITER //
CREATE PROCEDURE `addNewUser`(v_nom VARCHAR(50),v_prenom VARCHAR(50), v_email
VARCHAR(150),
v_adrRue VARCHAR(50),v_codePostal VARCHAR(50),
v_ville VARCHAR(50),v_username VARCHAR(50),v_mdp VARCHAR(255))
BEGIN

INSERT INTO utilisateurs VALUE (NULL,v_nom,v_prenom,v_email,CURDATE(),v_adrRue,
v_ville,v_codePostal,v_username,v_mdp);

INSERT INTO membres VALUE (LAST_INSERT_ID());

END//
DELIMITER ;
```

(Fardeen) - Cette procédure crée une nouvelle procédure stockée dans une base de données MySQL qui insère un nouvel utilisateur et un nouveau membre associé à cet utilisateur en utilisant les paramètres fournis pour nom, prénom, email, adresse, ville, code postal, nom d'utilisateur et mot de passe.

```
DROP TRIGGER IF EXISTS `updateLike`;
DELIMITER $$
CREATE TRIGGER `updateLike` AFTER INSERT ON `likes` FOR EACH ROW begin
DECLARE count INT;
  set count = ( select count(*) from likes where idArticle = new.idArticle );

 update article set likeTotal = count where id = new.idArticle;
end$$
DELIMITER ;
```

(Fardeen) - Ce trigger est déclenché après l'insertion d'une nouvelle ligne dans la table "likes". Il met à jour la colonne "likeTotal" dans la table "article" en comptant le nombre de "likes" pour l'article correspondant et en insérant cette valeur dans la colonne "likeTotal".

```
DROP TRIGGER IF EXISTS `updateLike2`;
DELIMITER $$
CREATE TRIGGER `updateLike2` AFTER DELETE ON `likes` FOR EACH ROW begin
DECLARE count INT;
  set count = ( select count(*) from likes where idArticle = old.idArticle );

 update article set likeTotal = count where id = old.idArticle;
end
$$
DELIMITER ;
```

(Dimitry) - Ce trigger est déclenché après la suppression d'une ligne de la table "likes". Il met à jour la colonne "likeTotal" dans la table "article" en comptant le nombre de "likes" pour l'article correspondant et en insérant cette valeur dans la colonne "likeTotal". Il utilise l'ancienne valeur de "idArticle" (old.idArticle) pour trouver le nombre de "likes" pour l'article qui vient d'être supprimé.

Architecture du site :

- **Index.php** (Dimitry) : La page est une page d'accueil de site web qui présente les dernières actualités et des articles recommandés dans différentes catégories.

  Pour la première section "Dernières actualités" :

```
SELECT * FROM article ORDER BY datePublication DESC LIMIT 1;
```

  Cette requête permet de sélectionner le dernier article de la base.

Pour la deuxième section "À voir aussi" :

```
SELECT * FROM article order by datePublication DESC LIMIT 3 OFFSET 1;
```

Cette requête sélectionne les trois derniers articles en ignorant le premier article (celui qui est déjà affiché dans la première section)

Pour la troisième section "Anime" et "Gaming" :

```
SELECT * FROM `article` WHERE idCate = 2 ORDER BY RAND ( ) LIMIT 1
```

```
SELECT * FROM `article` WHERE idCate = 1 ORDER BY RAND ( ) LIMIT 1
```

Ces requêtes sélectionnent un article de la catégorie "Anime" ou "Gaming" aléatoirement à l'aide de la fonction RAND().

- **news.php** (Fardeen) : Cette page affiche une liste d'articles triés par date de publication décroissante, avec pagination et un lien vers une page détaillée pour chaque article. Elle utilise également un fichier de navigation inclus en haut de la page.

- **viewArticles.php** (Dimitry) : Le code récupère l'identifiant d'un article à partir de l'URL, puis utilise cette information pour récupérer les détails de l'article dans la base de données. Si un utilisateur est connecté, le code affiche également un bouton "like" ou "unlike" pour l'article, en fonction de si l'utilisateur a déjà aimé l'article ou non.
  Si un modérateur est connecté, il aura également la possibilité de modifier ou supprimer l'article.

- **login-form.php** (Fardeen), **register-form.php** (Fardeen), **login.php** (Dimitry) et **register.php** (Fardeen) : Un formulaire d'inscription et un formulaire de connexion. Ainsi que les scripts gérant leur fonctionnement

- **updatePost.php** (Dimitry) et **scriptUpdate.php** (Dimitry) : Page de modification d'article, préremplie avec les informations de l'article sélectionné. Ainsi que sont script mettant à jour l'article.

- **createPost.php** et **insertPost.php** : Page de création d'article. Ainsi que sont script permettant de crée l'article.