

Projet Stage

ici je vais décrire le Projet AP1 réalisé en seconde année de BTS SIO

Modèle Conceptuel de Données (MCD)

Le MCD représente de manière abstraite les entités du système et leurs relations, sans se soucier des détails techniques de la base de données. Voici une description textuelle du MCD basé sur l'analyse des tables :

Entités

1. Utilisateur (`users`)

- Attributs :

- `nom`
- `prenom`
- `dateN` (date de naissance)
- `mail` (email)
- `tel` (téléphone)
- `login` , `mdp` (identifiants de connexion)
- `reset_token` et `reset_expires` (pour la réinitialisation de mot de passe)

2. Stage (`stage`)

- Attributs :

- `titre` (titre du stage)
- `dateD` , `dateF` (dates de début et fin du stage)
- `nom_entreprise` , `nom_tuteur` , `tel_tuteur` , `adresse` , `ville`

3. Compte-rendu (`cr`)

- Attributs :

- `sujet` , `contenu` (détail du compte-rendu)

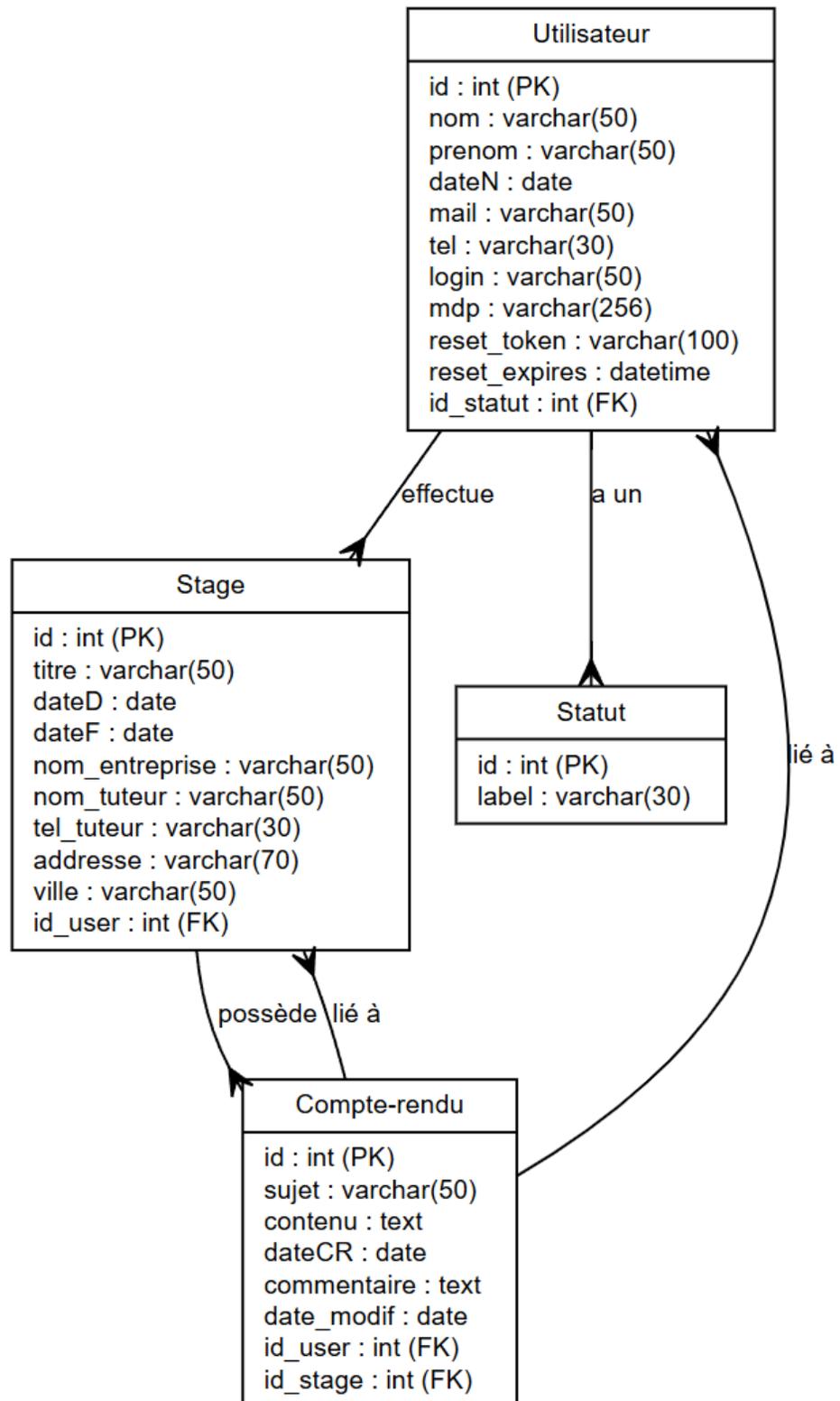
- `dateCR` , `date_modif` (dates associées)
- `commentaire` (retours)

4. Statut (`statut`)

- Attributs :
 - `label` (label du statut, par exemple : Étudiant, Enseignant, etc.)

Relations

1. **Utilisateur a un Statut** : Chaque utilisateur est lié à un statut via l'attribut `id_statut` . C'est une relation de type "Plusieurs utilisateurs peuvent avoir un même statut".
2. **Utilisateur effectue un Stage** : Un utilisateur peut avoir plusieurs stages (relation 1, N). Un stage est lié à un utilisateur via `id_user` .
3. **Stage a des Comptes-rendus** : Un stage peut avoir plusieurs comptes-rendus (également relation 1, N). Un compte-rendu est lié à un stage via `id_stage` .



Modèle Relationnel de Données (MRD)

Le MRD est une version plus technique qui inclut les détails de la base de données, telles que les clés primaires et les clés étrangères.

Tables et Clés

1. `users` (Utilisateur)
 - **Clé primaire** : `id`
 - **Clé étrangère** : `id_statut` (référence à la table `statut`)
2. `stage` (Stage)
 - **Clé primaire** : `id`
 - **Clé étrangère** : `id_user` (référence à la table `users`)
3. `cr` (Compte-rendu)
 - **Clé primaire** : `id`
 - **Clés étrangères** :
 - `id_user` (référence à la table `users`)
 - `id_stage` (référence à la table `stage`)
4. `statut` (Statut)
 - **Clé primaire** : `id`

Les relations sont implémentées à l'aide de clés étrangères. Par exemple, la colonne `id_user` dans la table `stage` est une clé étrangère pointant vers la clé primaire `id` de la table `users`.

Propriétés des Relations

- **Un utilisateur a plusieurs stages** : La relation est réalisée par la colonne `id_user` dans la table `stage`. Chaque utilisateur peut être lié à plusieurs stages.
- **Un stage a plusieurs comptes-rendus** : La relation est assurée par `id_stage` dans la table `cr`, qui pointe vers la table `stage`.

Description et Explication des Différentes Fonctionnalités du Site

L'application web est un système de gestion de stages et de comptes-rendus, destiné principalement aux étudiants et aux enseignants/professeurs. Elle comporte des sections séparées pour les étudiants et les professeurs, ainsi qu'une partie pour la gestion des comptes. Voici une explication des différentes fonctionnalités du site, basée sur les fichiers que vous avez fournis :

Section Élève

1. `ajoutCR.php`
 - **Fonctionnalité** : Cette page permet à un élève d'ajouter un compte-rendu de stage. Elle inclut une zone de texte pour rédiger le contenu du compte-rendu, ainsi qu'une sélection de la date et du stage concerné.
 - **Utilisation** : L'élève peut choisir un stage existant et écrire un compte-rendu associé à ce stage.
2. `createStage.php`
 - **Fonctionnalité** : Cette page offre à un élève la possibilité de créer un nouveau stage. Le formulaire contient des champs tels que le titre du stage, la date de début, la date de fin, l'entreprise, le tuteur, et les coordonnées de contact.
 - **Utilisation** : Les étudiants utilisent cette fonctionnalité pour ajouter des informations sur un nouveau stage avant de commencer à soumettre des comptes-rendus.
3. `editCR.php`
 - **Fonctionnalité** : Permet à un élève de modifier un compte-rendu précédemment rédigé.
 - **Utilisation** : L'utilisateur peut accéder aux comptes-rendus qu'il a soumis et apporter des modifications, par exemple en ajoutant des commentaires ou en mettant à jour le contenu.
4. `modifCR.php`
 - **Fonctionnalité** : Cette page affiche tous les comptes-rendus d'un élève sous forme de tableau avec pagination, permettant à l'utilisateur de les

consulter et de choisir ceux qu'il souhaite modifier.

- **Utilisation** : Permet de sélectionner facilement un compte-rendu précis à éditer.

Section Professeur

1. [CRprof.php](#)

- **Fonctionnalité** : Cette page affiche la liste des comptes-rendus des élèves, permettant aux professeurs de consulter les sujets, les prénoms des étudiants et les dates associées.
- **Utilisation** : Le professeur peut suivre l'avancement des différents élèves sur leurs stages.

2. [listeEleves.php](#)

- **Fonctionnalité** : Affiche une liste dynamique de tous les élèves sous forme de tableau, avec pagination et barre de recherche.
- **Utilisation** : Le professeur peut cliquer sur un élève pour obtenir plus d'informations et suivre son parcours (renvoie à [SpecEleve.php](#)).

3. [SpecEleve.php](#)

- **Fonctionnalité** : Cette page donne des détails sur un élève sélectionné, incluant ses informations personnelles, ses stages, et ses comptes-rendus.
- **Utilisation** : Affiche le parcours de l'élève, les stages effectués, et les comptes-rendus qui y sont liés. Cela permet au professeur d'avoir une vue d'ensemble sur la progression d'un élève précis.

Section Script Général

1. [index.php](#)

- **Fonctionnalité** : Page d'accueil et de connexion. Les utilisateurs peuvent se connecter à leur compte.
- **Utilisation** : Accès initial à l'application pour tous les utilisateurs.

2. [logout.php](#)

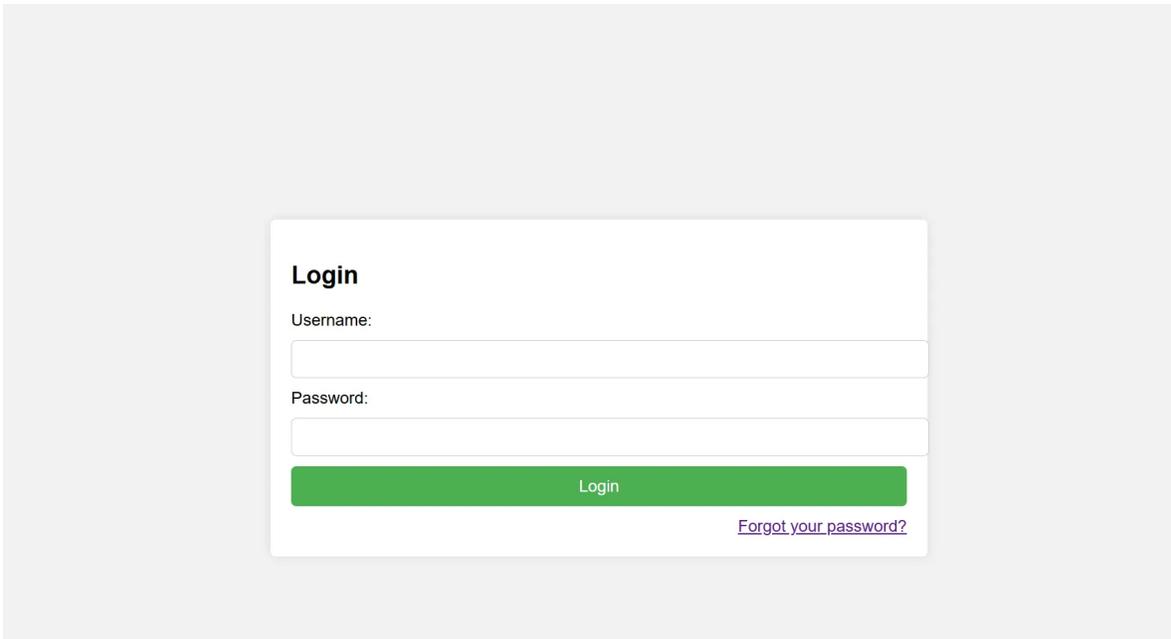
- **Fonctionnalité** : Permet à l'utilisateur de se déconnecter de l'application.
 - **Utilisation** : Déconnecte la session utilisateur et le renvoie à la page de connexion.
3. `request-reset.php` & `reset-password.php`
- **Fonctionnalité** : Ces scripts permettent de gérer la demande de réinitialisation de mot de passe (`request-reset.php`) et le formulaire de réinitialisation (`reset-password.php`).
 - **Utilisation** : Géré par les utilisateurs qui ont oublié leur mot de passe.
4. `submitCR.php`
- **Fonctionnalité** : Enregistre les informations relatives à un compte-rendu ajouté par un élève.
 - **Utilisation** : Ce script est appelé lors de l'ajout ou de la modification d'un compte-rendu.
5. `submitStage.php`
- **Fonctionnalité** : Enregistre les informations d'un nouveau stage créé par un élève.
 - **Utilisation** : Gère la soumission des informations du stage dans la base de données.
6. `updateAccount.php`
- **Fonctionnalité** : Permet la mise à jour des informations de compte utilisateur (par exemple, changer le mot de passe, l'adresse e-mail, etc.).
 - **Utilisation** : Mis en œuvre lorsqu'un utilisateur souhaite modifier ses informations personnelles.
7. `updateCR.php`
- **Fonctionnalité** : Met à jour un compte-rendu déjà existant.
 - **Utilisation** : Utilisé lors de l'édition de comptes-rendus par un élève.
8. `eleve.php` , `prof.php` , `myAccount.php`

- **Fonctionnalité** : Ces pages jouent le rôle de tableau de bord pour chaque type d'utilisateur. `eleve.php` et `prof.php` fournissent des raccourcis vers les principales actions qu'un étudiant ou un professeur peut effectuer. `myAccount.php` permet à l'utilisateur de voir et mettre à jour ses informations personnelles.

Fonctionnalités Générales et Logique de Navigation

- **Pagination et Affichage Dynamique** : La majorité des pages affichant des listes (par exemple, `modifCR.php`, `listeEleves.php`) utilisent une pagination pour faciliter la navigation, évitant de surcharger l'interface utilisateur.
- **Gestion des Comptes-Rendus et Stages** : Chaque élève peut créer des stages et les comptes-rendus associés, tandis que les professeurs ont la possibilité de les consulter pour suivre l'avancement.
- **Sécurité et Connexion** : Les scripts `index.php`, `logout.php`, `request-reset.php`, et `reset-password.php` gèrent les connexions des utilisateurs, la déconnexion, et les problèmes d'accès (par exemple, mot de passe oublié).

En conclusion, l'application offre une interface claire et organisée pour la gestion de stages et de comptes-rendus, avec des différents niveaux d'accès adaptés aux élèves et aux professeurs. Elle permet une navigation fluide, une gestion efficace des données, et une grande facilité d'utilisation pour tous les types d'utilisateurs impliqués.



login



page d'accueil professeur

My Account

[Home](#)[Logout](#)

Name:

Surname:

Birthdate:



Email:

Phone:

[Update Informations](#)

© 2024 Louis

page mon compte (élève et professeur)

Retour

Modifier un Compte-Rendu

Sujet du Compte-Rendu :

Flutter

Stage Lié :

UX Design

Contenu du Compte-Rendu :

apprentissage du dev mobile avec flutter

Enregistrer les Modifications

page de modification de compte rendu

Détails de l'élève

Retour

Informations de l'élève

Nom : petrechum

Email : steslaRedbullo@gmail.com

Prénom : stesla

Téléphone : 0678087896

Date de Naissance : 2014-10-01

Statut : Élève

Stages

Stage : UX Design

Entreprise : Apple

Dates : 2024-12-13 - 2025-03-15

Tuteur : Tim Cook (57541346879)

Adresse : 9 rue des mésanges, Pontoise

[Voir les Comptes-Rendus](#)

Stage : Devops

Entreprise : Thales

Dates : 2024-11-22 - 2024-11-24

Tuteur : Bernard DelaHaute (05985354793)

Comptes-Rendus

Sujet : Bootstrap 2

Date : 2024-11-30

Contenu : j'ai encore fait bootstrap c'est pas cool

Commentaire :

Sujet : Tailwind

Date : 2024-11-24

Contenu : apprentissage de tailwind CSS

Commentaire :

1 2

page détail d'un élève (professeur)