

TD PDO 02 – Application de gestion de budget

PDO, PreparedStatement, SQL, ChartJS

Objectif :

Dans ce TD, vous allez approfondir vos connaissances sur la manipulation de données avec le langage SQL et apprendre à faire interagir vos applications PHP avec les bases de données à l'aide de l'extension PDO.

Introduction :

Pour ce TD, vous travaillerez avec la base de données « **budget** » qu'il faudra installer. Un screen de son schéma vous est fourni dans les sources, pour vous aider à visualiser la base de données lors de l'écriture des requêtes. Vous avez carte blanche sur le design du site que vous devriez soigner avec un Framework CSS de votre choix ou avec du custom CSS.

L'application doit permettre la gestion du budget pour toute une famille de manière simple.

Pages de l'application :

Plan de l'application

L'accueil :

La page d'accueil doit présenter l'application et un lien vers les pages d'accueil du backoffice de l'application.

Accueil du backoffice :

Elle permet d'avoir un récapitulatif des dépenses et revenu pour toute la famille. Elle permet également de naviguer vers les principales pages d'administration de l'application :

- ✦ Ajout d'utilisateurs.
- ✦ Déclaration de revenu.
- ✦ Déclaration de dépense.
- ✦ Liste d'utilisateurs.

Liste d'utilisateurs :

Dans cette page, l'administrateur peut consulter la liste des membres avec les revenus et dépenses de chacun.

Depuis la liste, il peut accéder à la page détails d'un membre.



Détails d'un membre :

La page de détails doit afficher les informations de l'utilisateur sélectionné, ses dépenses et ses revenus.

L'administrateur doit pouvoir modifier ou supprimer les informations suivantes :

- ✦ Les données personnelles de l'utilisateur.
- ✦ Les données sur ses revenus
- ✦ Les données sur ses dépenses.

Il doit également pouvoir déclarer un revenu ou une dépense.

Organisation du développement :

Pour travailler de manière efficace, vous devez regrouper votre code en fonction de sa nature (configuration, fonctions, librairies, assets,).

Testez vos requêtes dans PHPMYAdmin ou dans la console avant de les intégrer dans vos scripts PHP.

Vous devez au maximum éviter de dupliquer du code (DRY).

Attention : toutes les données envoyées par l'utilisateur doivent être nettoyées et validées (**Never Trust User Input**).

Travail à faire :

Installation la base de données.

1. Dézipper le dossier joint.
2. Ouvrir le terminal dans ce dossier, exécuter successivement **«budget.sql»**.

Insertion / lecture de données.

1. A la racine du projet, ajouter une page de création d'un utilisateur. Elle doit permettre l'enregistrement d'un utilisateur dans la table « **users** » de la base de données.
Afficher la liste des utilisateurs à la suite de la création.

2. Ajouter une page de déclaration de revenu. Elle doit permettre de renseigner les revenus d'un utilisateur dans la table « **incomes** » pour un utilisateur donné.
L'administrateur doit pouvoir sélectionner dans une liste l'utilisateur pour lequel il est en train de faire la déclaration et le type du revenu.
3. Ajouter une page de déclaration de dépense. Elle doit permettre de renseigner les dépenses dans la table « **expenses** » pour un utilisateur donné.
L'administrateur doit pouvoir sélectionner dans une liste l'utilisateur pour lequel il est en train de faire la déclaration.

Modification / suppression

Depuis la page de détails, implémenter des fonctionnalités de modification et de suppression pour les revenus et les dépenses.

Bonus

Pagination / Recherche

1. Ajouter une page de gestion de toutes les dépenses. Elle doit afficher la liste de toutes les dépenses.
2. Implémenter une barre de recherche permettant de trouver rapidement une dépense.
3. Implémenter la fonctionnalité de tri sur les dépenses.

ChartJS

Créer et afficher un camembert (dépenses / revenus) dans la page d'accueil du backoffice.