

Rapport de projet

Sujet : SmartBudgetAI, application de gestion des finances pour les étudiants.

Lien du git : <https://github.com/Madmire/SmartBudgetAI.git>

Membres du groupe :

BENJELLOUN Nihal

DIPAMA Andrea Mireille

SANOGO Mamadou

SOMA Acharaf Mataman

Nom du prof : Hakim HAFIDI

Table des matières

1. Introduction	3
1.1 Contexte et motivation	3
1.2 Définition du problème	3
1.3 Objectifs du projet.....	3
2. Conception du projet	4
2.1 Idée principale	4
2.2 Architecture de l'application	4
2.3 Fonctionnalités principales	4
2.4 Wireframes	4
3. Développement Backend	5
3.1 Infrastructure Backend	5
3.2 Fonctionnalités Backend	5
3.3 API développées.....	5
3.4 Intégration de l'IA.....	5
3.5 Connexion à la base de données.....	6
4. Développement Frontend	6
4.1 Technologies utilisées	6
4.2 Structure des pages.....	7
4.3 Design et maquettes	7
4.4 Intégration avec le Backend	7
4.5 Captures d'écran.....	8
5. Intégration et Tests.....	10
5.1 Intégration	10
5.2 Tests	11
5.3 Résultats et Analyse	12
6. Défis rencontrés et Solutions apportées	12
6.1 Défis Techniques	12
6.2 Défis de Conception.....	13
6.3 Défis de Sécurité.....	14
7. Présentation et Résultats	14
7.1 Présentation du Produit	14
7.2 Résultats Finaux.....	15
7.3 Captures d'Écran et Démonstration	16
8. Conclusion et Perspectives	16

8.1 Conclusion	16
8.2 Perspectives	16
9. Conclusion Générale	17

1. Introduction

1.1 Contexte et motivation

Les étudiants sont confrontés à de nombreux défis dans la gestion de leurs finances. Avec des revenus limités, un manque d'éducation financière et des dépenses imprévues, beaucoup peinent à établir un équilibre budgétaire. De plus, l'absence d'outils intuitifs pour suivre les dépenses et gérer les économies aggrave le problème, les conduisant à l'endettement ou à des difficultés financières.

1.2 Définition du problème

Le problème principal est que les étudiants manquent de ressources efficaces pour :

- Suivre leurs dépenses et revenus de manière simplifiée.
- Recevoir des conseils personnalisés adaptés à leur situation financière.
- Gérer des budgets spécifiques et établir des objectifs d'épargne.

1.3 Objectifs du projet

Le projet vise à développer une application mobile alimentée par une intelligence artificielle pour :

1. Aider les étudiants à suivre leurs dépenses et revenus.
2. Fournir des conseils financiers personnalisés basés sur leurs habitudes de dépenses.
3. Encourager une gestion budgétaire proactive grâce à des outils d'analyse et de visualisation.
4. Permettre une meilleure planification des objectifs d'épargne pour améliorer leur autonomie financière.

2. Conception du projet

2.1 Idée principale

L'application **Smart Budget AI** vise à fournir une solution pratique et intuitive pour la gestion des finances personnelles des étudiants. Elle intègre une intelligence artificielle capable d'offrir des conseils financiers personnalisés et de simplifier le suivi des budgets et des dépenses.

2.2 Architecture de l'application

L'application suit une architecture en trois couches :

1. **Frontend** : Une interface utilisateur responsive et intuitive construite en HTML, CSS, et JavaScript (supportée par Bootstrap).
2. **Backend** : Une API basée sur FastAPI pour traiter les requêtes utilisateur, interagir avec l'IA, et gérer les transactions financières.
3. **Base de données** : Une base MySQL pour stocker les informations des utilisateurs, leurs transactions et catégories de dépenses.

Schéma d'architecture général

Un diagramme simplifié de l'architecture serait inclus ici pour illustrer :

- Les interactions entre les composants (frontend, backend, IA et base de données).
- Le flux des données (depuis la saisie utilisateur jusqu'aux réponses générées).

2.3 Fonctionnalités principales

- **Suivi des dépenses et revenus** : Les étudiants peuvent enregistrer leurs transactions, catégoriser les dépenses, et consulter leur solde actuel.
- **Analyse des dépenses** : Une visualisation des données sous forme de graphiques et de statistiques pour identifier les habitudes de dépenses.
- **Conseils personnalisés** : Une IA génère des suggestions adaptées, telles que la manière de réduire les coûts ou d'épargner.
- **Gestion de budgets** : Les utilisateurs peuvent définir des limites de dépenses et recevoir des alertes lorsqu'elles sont dépassées.

2.4 Wireframes

Avant le développement, des wireframes ont été créés pour définir les éléments visuels et interactifs des différentes pages :

1. **Tableau de bord** : Vue d'ensemble des finances (solde, revenus, dépenses, et catégories).
2. **Conseils AI** : Interface de chat pour obtenir des recommandations financières.
3. **Table des transactions** : Détails des entrées et sorties financières par catégorie.

3. Développement Backend

3.1 Infrastructure Backend

Le backend de l'application **Smart Budget AI** a été développé avec **FastAPI**, un framework léger et performant pour la création d'API REST. Il sert d'intermédiaire entre l'interface utilisateur (frontend), la base de données, et les modèles d'IA.

3.2 Fonctionnalités Backend

Les principales fonctionnalités gérées par le backend incluent :

- **Authentification des utilisateurs** : Gestion des connexions et des sessions utilisateur à l'aide de sessions PHP côté frontend et vérifications côté backend.
- **Gestion des transactions** : Création, modification et suppression des transactions financières.
- **Analyse des données** : Calcul des statistiques de dépenses et revenus pour le tableau de bord.
- **Intégration de l'IA** : Traitement des requêtes utilisateur pour fournir des réponses adaptées via le modèle d'IA fine-tuné.
- **Validation et sécurité des données** : Contrôle des entrées utilisateur pour éviter les erreurs et les abus.

3.3 API développées

Voici une liste des endpoints principaux :

1. **/register** (*POST*) : Permet l'inscription d'un nouvel utilisateur.
2. **/login** (*POST*) : Gère la connexion utilisateur en validant les identifiants.
3. **/transactions** (*GET, POST, DELETE*) : Récupère, ajoute ou supprime des transactions.
4. **/dashboard** (*GET*) : Fournit les données agrégées pour le tableau de bord.
5. **/chat** (*POST*) : Envoie les requêtes utilisateur au modèle d'IA et retourne les réponses générées.

3.4 Intégration de l'IA

Le backend intègre un modèle **GPT-Neo fine-tuné**, adapté aux questions financières courantes des étudiants. Voici les étapes principales de l'intégration :

1. **Chargement du modèle** : Le modèle est chargé et configuré avec des paramètres spécifiques (e.g., temperature, max_length).
2. **Traitement des requêtes** : Chaque requête utilisateur est envoyée au modèle sous la forme d'un prompt structuré.
3. **Retour des réponses** : Les réponses générées sont nettoyées et formatées avant d'être renvoyées au frontend.

3.5 Connexion à la base de données

Une base de données **MySQL** est utilisée pour stocker les informations utilisateur, les transactions, et les catégories de dépenses.

Schéma des tables principales :

- **users** : Stocke les informations des utilisateurs (id, nom, email, mot de passe).
- **transactions** : Contient les détails des transactions (id, user_id, montant, catégorie, date).

4. Développement Frontend

4.1 Technologies utilisées

Le frontend de l'application **SmartBudgetAI** a été développé en utilisant les technologies suivantes :

- **HTML5 / CSS3** : Pour la structure et le design des pages.
- **Bootstrap 5** : Pour un design moderne, réactif et des composants préconçus.
- **JavaScript (Vanilla)** : Pour les interactions dynamiques et les appels API.
- **PHP** : Intégré pour la gestion des sessions utilisateur et la communication avec le backend.

4.2 Structure des pages

Le frontend comporte plusieurs pages pour offrir une expérience utilisateur fluide et intuitive :

1. Page de connexion / inscription :

- Permet aux utilisateurs de se connecter ou de créer un compte.
- Vérifie les informations avant de transmettre les données au backend.

2. Tableau de bord :

- Affiche un aperçu des soldes, revenus, dépenses, et transactions récentes.
- Inclut des graphiques pour visualiser la répartition des dépenses par catégorie.

3. Conseils AI :

- Permet aux utilisateurs de poser des questions à l'IA et de recevoir des conseils financiers personnalisés.

4. Table des transactions :

- Présente une vue détaillée des transactions passées, avec des options pour ajouter, modifier ou supprimer des entrées.

5. Page de profil :

- Affiche les informations utilisateur avec des options pour les modifier.

4.3 Design et maquettes

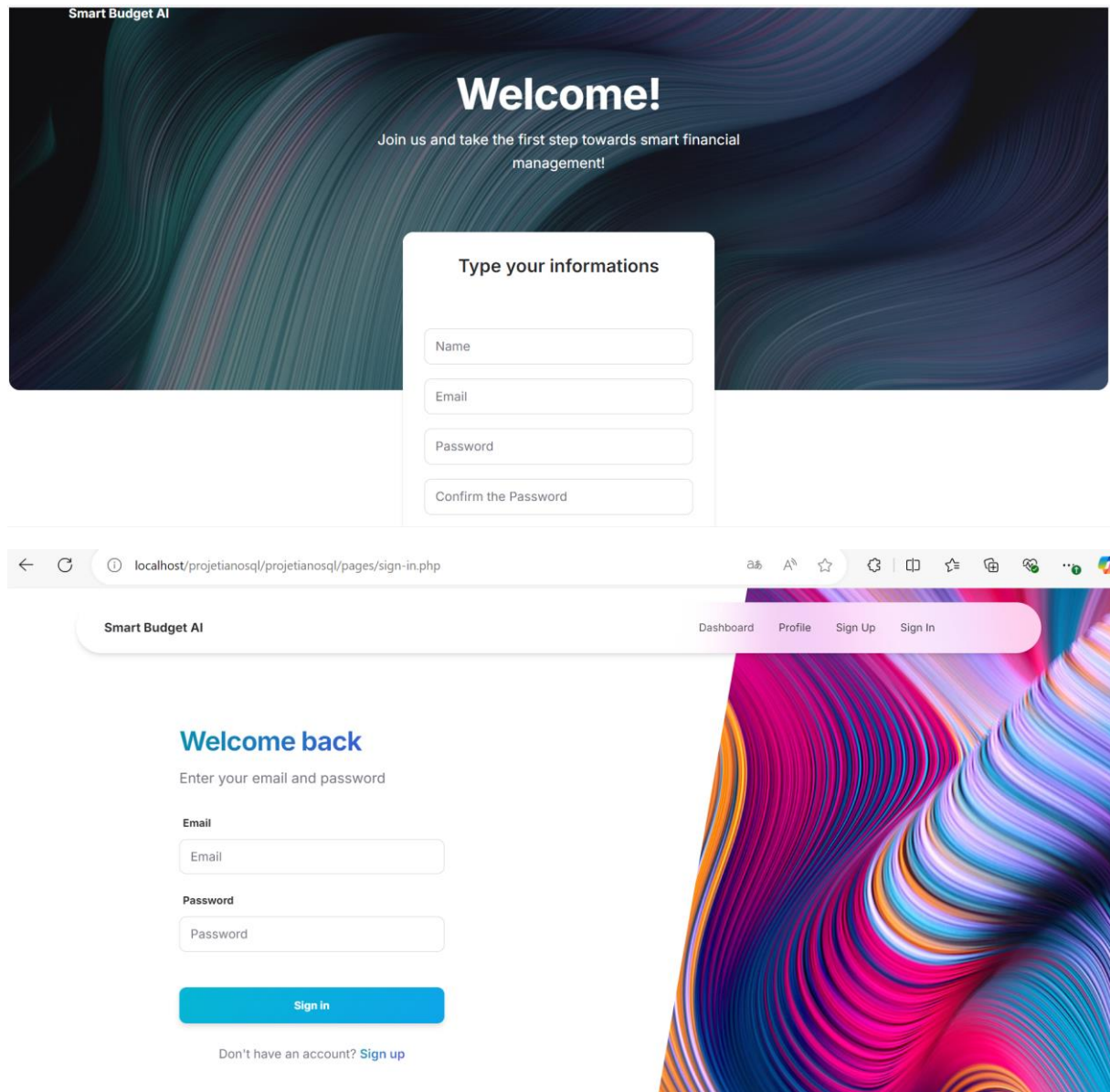
Exemple : Tableau de bord

- Les **cartes d'information** affichent :
 - Solde actuel.
 - Total des revenus et dépenses.
 - Nombre de transactions effectuées.
- Une section graphique illustre la répartition des dépenses par catégorie.
- Une interface utilisateur intuitive permet d'ajouter des transactions facilement.

4.4 Intégration avec le Backend

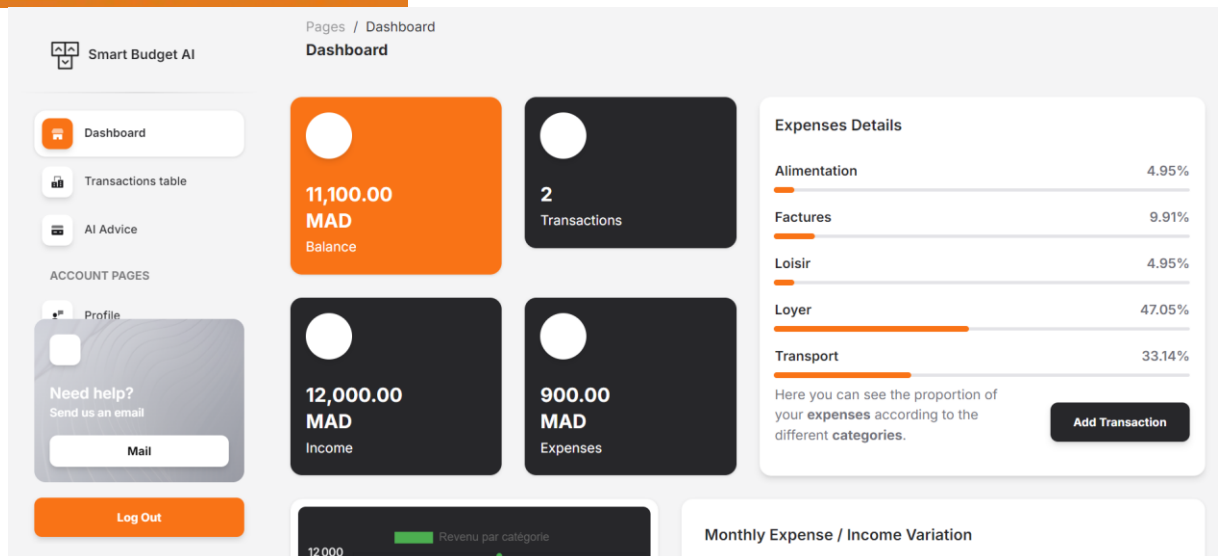
Les pages frontend utilisent **AJAX** et **Fetch API** pour interagir avec le backend en temps réel. Les données sont récupérées ou envoyées sans nécessiter de rechargement complet de la page.

4.5 Captures d'écran



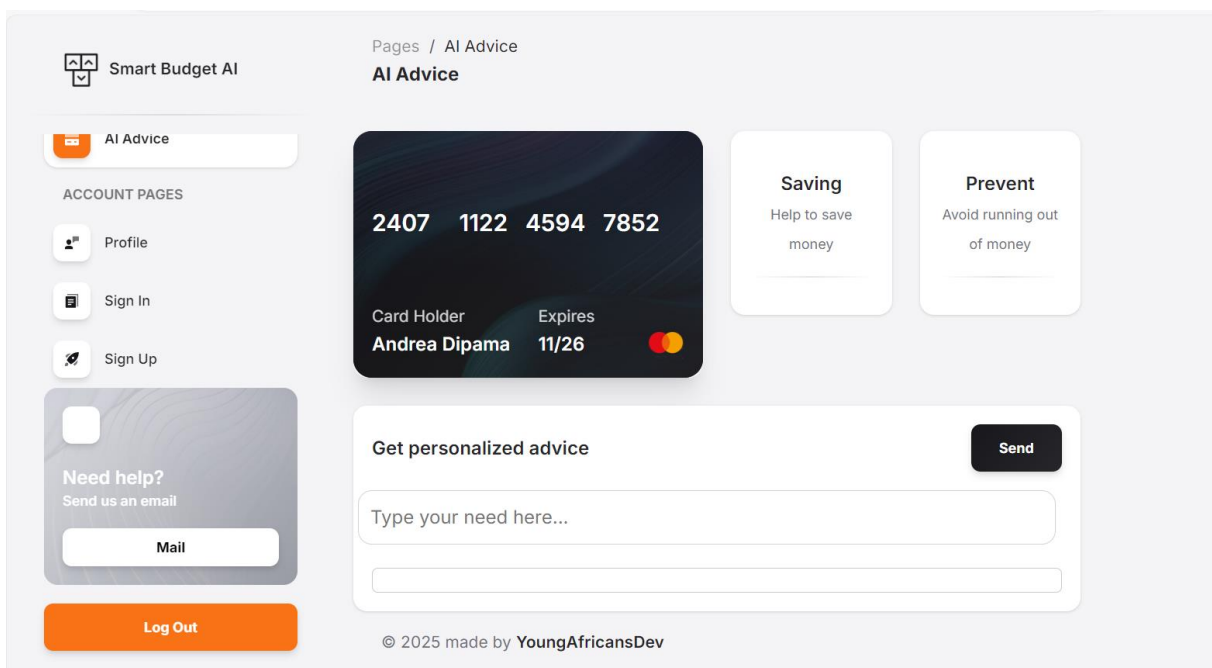
4.5.1 Tableau de bord

- **Description :** Le tableau de bord présente les informations clés sur les finances de l'utilisateur de manière visuelle et condensée.



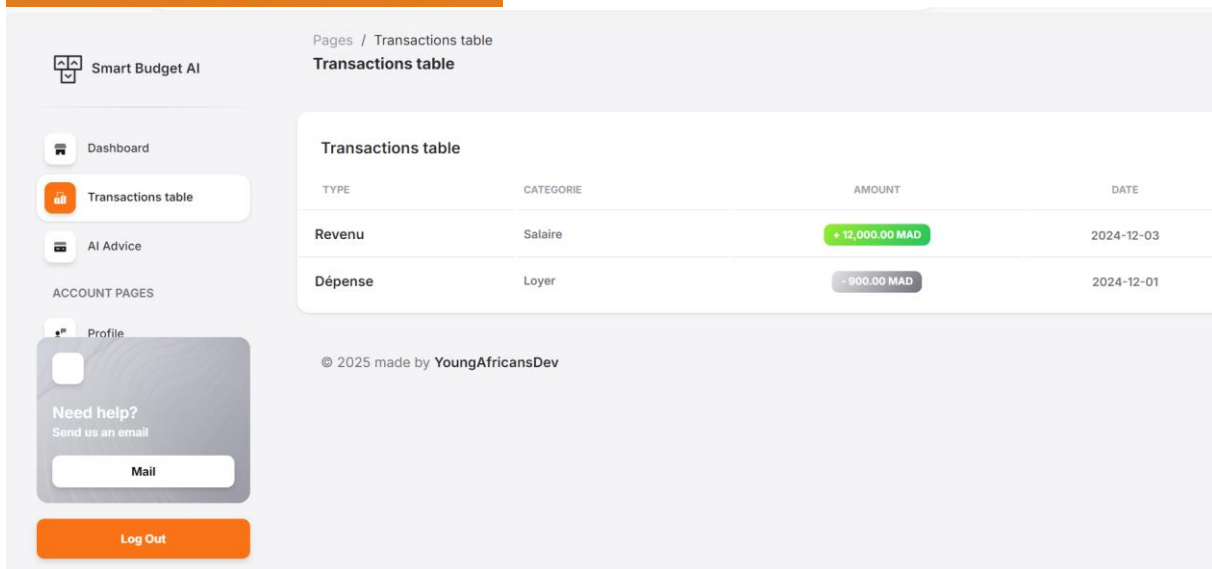
4.5.2 Conseils AI

- **Interface utilisateur :**
 - Une barre de recherche pour poser des questions à l'IA.
 - Une section pour afficher les réponses générées par l'IA en temps réel.



4.5.3 Table des transactions

- **Vue structurée des transactions :**
 - Colonnes pour la catégorie, le montant, la date, et les options d'édition.
 - Bouton **Ajouter une transaction** accessible directement.



4.6 Points d'amélioration

- Améliorer le design mobile pour une meilleure expérience sur les petits écrans.
- Ajouter des animations pour une expérience utilisateur plus fluide.

5. Intégration et Tests

5.1 Intégration

L'intégration de l'application **Smart Budget AI** a consisté à combiner le backend, le frontend, et les API d'IA en un système cohérent et fonctionnel.

5.1.1 Communication Frontend-Backend

La communication entre le frontend et le backend repose sur les technologies suivantes :

- **Fetch API** pour envoyer des requêtes HTTP depuis le frontend.
- Les réponses JSON du backend sont gérées dynamiquement par JavaScript.
- Les endpoints principaux implémentés dans le backend incluent :
 - POST /chat : Pour générer des conseils financiers personnalisés via l'IA.
 - POST /transactions : Pour ajouter une transaction.
 - GET /transactions : Pour récupérer les transactions existantes.
 - POST /auth/login : Pour gérer les connexions utilisateurs.

Exemple : Flux complet d'une requête

1. L'utilisateur ajoute une transaction via l'interface frontend.

2. Les données sont envoyées au backend via POST /transactions.
3. Le backend enregistre les données dans la base de données MySQL et renvoie une confirmation.
4. Le frontend met à jour la table des transactions sans recharger la page.

5.1.2 Intégration du modèle AI

L'API d'IA est utilisée pour fournir des réponses personnalisées aux questions des utilisateurs :

- Les prompts sont formatés pour inclure le contexte utilisateur (par exemple, type de question, historique).
- Le modèle AI retourne une réponse structurée, qui est affichée dans une interface conviviale.

5.2 Tests

Des tests rigoureux ont été réalisés pour s'assurer du bon fonctionnement de l'application.

5.2.1 Tests Fonctionnels

Les tests fonctionnels ont couvert les fonctionnalités principales de l'application :

- **Connexion et inscription** : Vérification des erreurs pour des identifiants incorrects ou inexistants.
- **Ajout et modification de transactions** : Tests pour des données valides et invalides.
- **Tableau de bord** : Vérification de l'affichage correct des soldes et graphiques.
- **Conseils AI** : Validation de la qualité des réponses générées par le modèle.

5.2.2 Tests de Performances

- **Temps de réponse API** :
 - POST /chat : < 1 seconde.
 - POST /transactions : < 500 ms.
- **Frontend** :
 - Chargement initial < 2 secondes sur un réseau standard.
 - Interface responsive sur différentes tailles d'écran.

5.2.3 Tests de Sécurité

- Validation des entrées utilisateurs pour éviter les injections SQL.
- Cryptage des mots de passe avec **bcrypt** dans la base de données.
- Authentification basée sur des sessions sécurisées en PHP.

5.3 Résultats et Analyse

- **Succès :**
 - Toutes les fonctionnalités essentielles fonctionnent comme prévu.
 - Le modèle AI répond avec pertinence dans la majorité des cas.
- **Améliorations suggérées :**
 - Optimiser la gestion des sessions pour des expirations automatiques.
 - Améliorer la vitesse des requêtes d'IA pour les utilisateurs ayant des connexions limitées.

6. Défis rencontrés et Solutions apportées

6.1 Défis Techniques

6.1.1 Intégration du modèle AI

- **Problème :** Les réponses initiales générées par le modèle **GPT-Neo 125M** n'étaient pas toujours cohérentes ou adaptées au contexte des finances étudiantes.

SmartBudgetAI Chatbot API 0.1.0 OAS 3.1
/openapi.json

default ^

POST /chat Get Response ^

Parameters Cancel Reset

No parameters

Request body required application/json v

```
{
  "prompt": "comment puis-je épargner ?"
}
```

```

Curl
curl -X 'POST' \
  'http://localhost:8000/chat' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "prompt": "comment puis-je épargner ?"
  }'

Request URL
http://localhost:8000/chat

Server response

Code    Details
200

Response body
{
  "answer": ": comment puis-je épargner?\n:"
}

Response headers
content-length: 44
content-type: application/json
date: Wed, 15 Jan 2025 13:46:31 GMT
server: uvicorn
  
```

- **Solution :**
 - Fine-tuning du modèle en utilisant un dataset personnalisé avec des instructions financières et des exemples adaptés aux étudiants.
 - Changement du modèle par dbddv01/gpt2-french-small
 - Ajustement des paramètres de génération (par exemple, réduction de la température et utilisation de nucleus sampling) pour améliorer la qualité des réponses.

6.1.2 Gestion de la base de données

- **Problème :** Problèmes d'intégrité des données lorsque plusieurs utilisateurs ajoutaient ou modifiaient des transactions simultanément.
- **Solution :**
 - Ajout de transactions atomiques dans MySQL pour garantir l'intégrité des données.
 - Mise en œuvre d'indices dans les colonnes fréquemment interrogées pour accélérer les requêtes.

6.2 Défis de Conception

6.2.1 Expérience Utilisateur

- **Problème :** La première version de l'interface utilisateur n'était pas intuitive pour les utilisateurs novices.
- **Solution :**
 - Réalisation de tests utilisateur pour recueillir des retours.
 - Simplification de l'interface, ajout d'icônes claires, et amélioration de la disposition des éléments.

- Implémentation de fonctionnalités telles que des barres de progression pour aider à visualiser les objectifs d'épargne.

6.3 Défis de Sécurité

6.3.1 Sécurisation des données sensibles

- **Problème :** Risques liés aux données sensibles (mots de passe, informations personnelles des utilisateurs).
- **Solution :**
 - Hachage des mots de passe.
 - Mise en œuvre de HTTPS pour sécuriser les communications.
 - Ajout de politiques strictes pour éviter les injections SQL dans les requêtes backend.

6.3.2 Prévention des attaques par force brute

- **Problème :** Absence de mécanisme pour limiter les tentatives de connexion.
- **Solution :**
 - Implémentation d'un système de verrouillage temporaire après plusieurs tentatives infructueuses.

7. Présentation et Résultats

7.1 Présentation du Produit

L'application **SmartBudgetAI** offre une solution intuitive et efficace pour aider les étudiants à gérer leurs finances. Elle combine les meilleures pratiques en intelligence artificielle, design utilisateur et gestion de données pour répondre aux besoins spécifiques des étudiants.

Caractéristiques principales :

1. **Suivi des finances personnelles :**
 - Suivi en temps réel du solde, des revenus et des dépenses.
 - Classification automatique des transactions dans différentes catégories (loyer, alimentation, transport, etc.).
2. **Conseils personnalisés :**

- Génération de conseils financiers adaptés à chaque utilisateur grâce à un modèle d'IA fine-tuné.
- Recommandations pour économiser, établir des budgets, et planifier les dépenses.

3. Objectifs financiers :

- Création d'objectifs d'épargne avec suivi de progression.
- Alertes pour encourager les utilisateurs à atteindre leurs objectifs.

4. Interface conviviale :

- Tableau de bord interactif montrant un résumé clair des finances de l'utilisateur.
- Option multilingue et design responsive.

5. Sécurité renforcée :

- Cryptage des données sensibles.
- Accès sécurisé avec authentification et protection contre les attaques par force brute.

7.2 Résultats Finaux

7.2.1 Développement Technique

- **Backend :**
 - API robuste développée avec FastAPI pour connecter l'application au modèle d'IA.
 - Gestion efficace de la base de données MySQL pour stocker les transactions.
- **Frontend :**
 - Interface utilisateur développée avec HTML5, CSS3, et JavaScript.
 - Visualisation des données avec des graphiques dynamiques et des widgets interactifs.
- **Modèle d'IA :**
 - Fine-tuning d'un modèle GPT-Neo sur un dataset personnalisé contenant des scénarios financiers adaptés aux étudiants.
 - Réduction significative des erreurs grâce à une configuration optimisée des hyperparamètres.

7.2.2 Tests Utilisateurs

Pas de tests pour le moment mais dans un futur proche .

7.3 Captures d'Écran et Démonstration

1. Tableau de bord principal :

- Affiche les informations financières essentielles : solde actuel, revenus, dépenses, et répartition des catégories.

8. Conclusion et Perspectives

8.1 Conclusion

Le projet **Smart Budget AI** a permis de répondre à un besoin réel : offrir aux étudiants un outil intelligent et personnalisé pour gérer leurs finances. Grâce à l'intégration d'un modèle d'intelligence artificielle performant, à une interface intuitive, et à une base de données robuste, l'application remplit son objectif de manière efficace.

Les utilisateurs peuvent désormais :

- Suivre leurs dépenses en temps réel.
- Recevoir des conseils financiers adaptés.
- Fixer et atteindre des objectifs d'épargne.
- Améliorer leurs habitudes financières.

SmartBudgetAI se positionne ainsi comme une solution innovante pour aider les étudiants à prendre en main leur santé financière.

8.2 Perspectives

Bien que le projet ait atteint ses objectifs principaux, plusieurs pistes d'amélioration et d'expansion ont été identifiées pour maximiser l'impact de l'application :

1. Améliorations Techniques :

- **Précision accrue de l'IA** : Affiner le modèle avec des données encore plus spécifiques aux besoins des étudiants.
- **Automatisation avancée** : Implémenter un système d'analyse prédictive pour anticiper les dépenses futures et fournir des alertes proactives.

2. Expérience Utilisateur :

- Ajouter des fonctionnalités comme la planification mensuelle automatique des budgets.
- Intégrer un mode sombre pour améliorer le confort visuel.

3. Expansion Fonctionnelle :

- Support multidevise pour les étudiants internationaux.
- Intégration d'une API bancaire pour un suivi encore plus automatisé des transactions.
- Création d'un système de récompenses pour encourager l'épargne régulière : Des flammes seront ajoutées sur l'interface à chaque fois qu'un utilisateur aura atteint son objectif d'épargne.

4. Sécurité et Confidentialité :

- Mise en place de protocoles de sécurité avancés, notamment l'authentification biométrique.
- Conformité renforcée avec les réglementations sur la protection des données (e.g., RGPD).

5. Engagement Communautaire :

- Lancer une communauté en ligne où les utilisateurs peuvent partager des conseils financiers et des astuces.
- Proposer des webinaires éducatifs sur la gestion financière.

9. Conclusion Générale

Le projet **SmartBudgetAI** représente une étape importante dans la démocratisation de la gestion financière assistée par l'IA. Avec des fonctionnalités clés adaptées aux étudiants et un potentiel d'évolution significatif, cette application peut non seulement améliorer les finances personnelles de ses utilisateurs, mais aussi poser les bases d'un avenir financier plus stable et réfléchi.

En combinant innovation, utilité pratique, et accessibilité, **SmartBudgetAI** incarne une réponse technologique au défi universel de la gestion financière.