

e-Cotisation – Fiche Projet

1. Nom du projet

e-Cotisation – Plateforme de gestion des dons pour enfants par niveau scolaire

2. Objectif du projet

Développer une application web de collecte de dons destinée à soutenir des enfants inscrits dans différents niveaux scolaires. Les dons sont faits manuellement par les donateurs, et leur enregistrement est confirmé par un administrateur via une interface d'administration sécurisée.

3. Acteurs du système

- Donateur : Personne qui souhaite faire un don pour soutenir un niveau scolaire.
- Administrateur : Personne chargée de gérer les enfants, niveaux, donateurs et dons depuis l'interface d'administration.

4. Fonctionnalités principales

- Front-End (public) :
 - - Affichage des niveaux scolaires et de leur objectif de collecte.
 - - Formulaire de don : Nom du donateur, Montant du don, Niveau scolaire ciblé.
- Back-End (admin Django) :
 - - Gestion des niveaux, enfants, donateurs et dons.
 - - Confirmation manuelle des dons (après réception physique).
 - - Suivi des dons par niveau.
 - - Visualisation des totaux collectés et comparaison avec les objectifs.

5. Modèles de données

```
class Niveau(models.Model):
```

```
    nom = models.CharField(max_length=100, unique=True)
```

```
    objectif_collecte = models.DecimalField(max_digits=10, decimal_places=2)
```

```
class Enfant(models.Model):
```

```
    nom = models.CharField(max_length=100)
```

```
    prenom = models.CharField(max_length=100)
```

```
    date_naissance = models.DateField()
```

```
    niveau = models.ForeignKey(Niveau, on_delete=models.CASCADE, related_name="enfants")
```

```
class Donateur(models.Model):
```

```
    nom = models.CharField(max_length=100)
```

```
    email = models.EmailField()
```

```
    telephone = models.CharField(max_length=20)
```

```
class Don(models.Model):
```

```
    donateur = models.ForeignKey(Donateur, on_delete=models.SET_NULL, null=True,
```

```
blank=True)
niveau = models.ForeignKey(Niveau, on_delete=models.CASCADE, related_name="dons")
montant = models.DecimalField(max_digits=10, decimal_places=2)
date_don = models.DateTimeField(auto_now_add=True)
confirme = models.BooleanField(default=False)
```

6. Authentification et sécurité

- L'inscription et la connexion des donateurs ne sont pas obligatoires.
- L'accès à la gestion (CRUD) se fait uniquement via l'espace admin Django.

7. Technologies utilisées

- Back-End : Django (Python)
- Front-End : HTML/CSS (et potentiellement Bootstrap)
- Base de données : SQLite (ou autre selon déploiement)
- Interface d'administration : Django Admin + Jazzmin (thème amélioré)