Commençons par l'intégration des champs nécessaires dans le modèle Employe.

Étape 1 : Intégration des champs liés à la retraite dans le modèle Employe

1. Modifiez le modèle Employe (dans models.py) :

from django.db import models

from django.utils import timezone

class Employe(models.Model):

# ... autres champs existants ...

date\_naissance = models.DateField()

date\_embauche = models.DateField()

# Nouveaux champs pour la gestion de la retraite

age\_legal\_retraite = models.IntegerField(default=62)

date\_retraite\_prevue = models.DateField(null=True, blank=True)

def calculer\_date\_retraite(self):

if not self.date\_retraite\_prevue:

self.date\_retraite\_prevue = self.date\_naissance.replace(year=self.date\_naissance.year + self.age\_legal\_retraite)

self.save()

return self.date\_retraite\_prevue

def annees\_avant\_retraite(self):

if self.date\_retraite\_prevue:

aujourd\_hui = timezone.now().date()

return (self.date\_retraite\_prevue - aujourd\_hui).days // 365

return None

def \_\_str\_\_(self):

return f"{self.nom} {self.prenom}"

1. Créez une migration pour appliquer ces changements :

python manage.py makemigrations

python manage.py migrate

1. Mettez à jour le formulaire d'employé (dans forms.py) :

from django import forms

from .models import Employe

class EmployeForm(forms.ModelForm):

class Meta:

model = Employe

fields = [

# ... autres champs ...

'date\_naissance',

'date\_embauche',

'age\_legal\_retraite',

'date\_retraite\_prevue',

]

widgets = {

'date\_naissance': forms.DateInput(attrs={'type': 'date'}),

'date\_embauche': forms.DateInput(attrs={'type': 'date'}),

'date\_retraite\_prevue': forms.DateInput(attrs={'type': 'date'}),

}

1. Modifiez la vue pour inclure le calcul automatique de la date de retraite (dans views.py) :

from django.shortcuts import render, redirect

from .models import Employe

from .forms import EmployeForm

def ajouter\_employe(request):

if request.method == 'POST':

form = EmployeForm(request.POST)

if form.is\_valid():

employe = form.save(commit=False)

employe.calculer\_date\_retraite()

employe.save()

return redirect('liste\_employes')

else:

form = EmployeForm()

return render(request, 'gpec/ajouter\_employe.html', {'form': form})

def modifier\_employe(request, employe\_id):

employe = get\_object\_or\_404(Employe, id=employe\_id)

if request.method == 'POST':

form = EmployeForm(request.POST, instance=employe)

if form.is\_valid():

employe = form.save(commit=False)

employe.calculer\_date\_retraite()

employe.save()

return redirect('detail\_employe', employe\_id=employe.id)

else:

form = EmployeForm(instance=employe)

return render(request, 'gpec/modifier\_employe.html', {'form': form, 'employe': employe})

1. Mettez à jour le template de détail de l'employé pour afficher ces nouvelles informations (dans detail\_employe.html) :

{% extends 'base.html' %}

{% block content %}

<h1>Détails de l'employé</h1>

<!-- ... autres informations ... -->

<p><strong>Date de naissance:</strong> {{ employe.date\_naissance }}</p>

<p><strong>Date d'embauche:</strong> {{ employe.date\_embauche }}</p>

<p><strong>Âge légal de retraite:</strong> {{ employe.age\_legal\_retraite }} ans</p>

<p><strong>Date de retraite prévue:</strong> {{ employe.date\_retraite\_prevue }}</p>

<p><strong>Années avant la retraite:</strong> {{ employe.annees\_avant\_retraite }}</p>

<!-- ... autres informations ... -->

{% endblock %}