La gestion du temps de travail (outil de pointage, constat des présences...)". Cette fonctionnalité permettra de suivre les heures de travail des employés, les présences et les absences.

Voici le code détaillé :

1. Modèles (ajoutez ceci à models.py) :

from django.db import models

from django.utils import timezone

from employe.models import Employe

class Pointage(models.Model):

employe = models.ForeignKey(Employe, on\_delete=models.CASCADE, related\_name='pointages')

date = models.DateField(default=timezone.now)

heure\_arrivee = models.TimeField(null=True, blank=True)

heure\_depart = models.TimeField(null=True, blank=True)

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

return f"{self.employe.nom} - {self.date}"

class Absence(models.Model):

TYPE\_ABSENCE = [

('CONGE', 'Congé'),

('MALADIE', 'Maladie'),

('AUTRE', 'Autre'),

]

employe = models.ForeignKey(Employe, on\_delete=models.CASCADE, related\_name='absences')

date\_debut = models.DateField()

date\_fin = models.DateField()

type\_absence = models.CharField(max\_length=20, choices=TYPE\_ABSENCE)

justification = models.TextField(blank=True)

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

return f"{self.employe.nom} - {self.get\_type\_absence\_display()} - {self.date\_debut}"

1. Formulaires (ajoutez ceci à forms.py) :

from django import forms

from .models import Pointage, Absence

class PointageForm(forms.ModelForm):

class Meta:

model = Pointage

fields = ['employe', 'date', 'heure\_arrivee', 'heure\_depart']

widgets = {

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

'heure\_arrivee': forms.TimeInput(attrs={'type': 'time'}),

'heure\_depart': forms.TimeInput(attrs={'type': 'time'}),

}

class AbsenceForm(forms.ModelForm):

class Meta:

model = Absence

fields = ['employe', 'date\_debut', 'date\_fin', 'type\_absence', 'justification']

widgets = {

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

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

}

1. Vues (ajoutez ceci à views.py) :

from django.shortcuts import render, redirect, get\_object\_or\_404

from django.contrib import messages

from django.utils import timezone

from .models import Pointage, Absence, Employe

from .forms import PointageForm, AbsenceForm

def gestion\_temps\_travail(request):

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

pointages = Pointage.objects.filter(date=aujourd\_hui).order\_by('employe\_\_nom')

absences = Absence.objects.filter(date\_debut\_\_lte=aujourd\_hui, date\_fin\_\_gte=aujourd\_hui).order\_by('employe\_\_nom')

return render(request, 'gpec/gestion\_temps\_travail.html', {'pointages': pointages, 'absences': absences})

def ajouter\_pointage(request):

if request.method == 'POST':

form = PointageForm(request.POST)

if form.is\_valid():

form.save()

messages.success(request, "Pointage ajouté avec succès.")

return redirect('gestion\_temps\_travail')

else:

form = PointageForm(initial={'date': timezone.now().date()})

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

def modifier\_pointage(request, pointage\_id):

pointage = get\_object\_or\_404(Pointage, id=pointage\_id)

if request.method == 'POST':

form = PointageForm(request.POST, instance=pointage)

if form.is\_valid():

form.save()

messages.success(request, "Pointage modifié avec succès.")

return redirect('gestion\_temps\_travail')

else:

form = PointageForm(instance=pointage)

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

def ajouter\_absence(request):

if request.method == 'POST':

form = AbsenceForm(request.POST)

if form.is\_valid():

form.save()

messages.success(request, "Absence ajoutée avec succès.")

return redirect('gestion\_temps\_travail')

else:

form = AbsenceForm()

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

def rapport\_temps\_travail(request):

employes = Employe.objects.all()

debut = request.GET.get('debut')

fin = request.GET.get('fin')

if debut and fin:

pointages = Pointage.objects.filter(date\_\_range=[debut, fin]).order\_by('employe', 'date')

absences = Absence.objects.filter(date\_debut\_\_lte=fin, date\_fin\_\_gte=debut).order\_by('employe', 'date\_debut')

else:

pointages = Absence.objects.none()

absences = Absence.objects.none()

return render(request, 'gpec/rapport\_temps\_travail.html', {

'employes': employes,

'pointages': pointages,

'absences': absences,

'debut': debut,

'fin': fin,

})

1. URLs (ajoutez ceci à urls.py) :

from django.urls import path

from . import views

urlpatterns = [

# ... autres URLs ...

path('gpec/gestion-temps-travail/', views.gestion\_temps\_travail, name='gestion\_temps\_travail'),

path('gpec/ajouter-pointage/', views.ajouter\_pointage, name='ajouter\_pointage'),

path('gpec/modifier-pointage/<int:pointage\_id>/', views.modifier\_pointage, name='modifier\_pointage'),

path('gpec/ajouter-absence/', views.ajouter\_absence, name='ajouter\_absence'),

path('gpec/rapport-temps-travail/', views.rapport\_temps\_travail, name='rapport\_temps\_travail'),

]

1. Templates :
   1. gestion\_temps\_travail.html :

{% extends 'base.html' %}

{% block content %}

<h1>Gestion du Temps de Travail</h1>

<h2>Pointages du jour</h2>

<a href="{% url 'ajouter\_pointage' %}" class="btn btn-primary mb-3">Ajouter un pointage</a>

<table class="table">

<thead>

<tr>

<th>Employé</th>

<th>Heure d'arrivée</th>

<th>Heure de départ</th>

<th>Actions</th>

</tr>

</thead>

<tbody>

{% for pointage in pointages %}

<tr>

<td>{{ pointage.employe.nom }} {{ pointage.employe.prenom }}</td>

<td>{{ pointage.heure\_arrivee|default\_if\_none:"-" }}</td>

<td>{{ pointage.heure\_depart|default\_if\_none:"-" }}</td>

<td>

<a href="{% url 'modifier\_pointage' pointage.id %}" class="btn btn-warning btn-sm">Modifier</a>

</td>

</tr>

{% empty %}

<tr>

<td colspan="4">Aucun pointage pour aujourd'hui.</td>

</tr>

{% endfor %}

</tbody>

</table>

<h2>Absences en cours</h2>

<a href="{% url 'ajouter\_absence' %}" class="btn btn-primary mb-3">Ajouter une absence</a>

<table class="table">

<thead>

<tr>

<th>Employé</th>

<th>Type d'absence</th>

<th>Date de début</th>

<th>Date de fin</th>

</tr>

</thead>

<tbody>

{% for absence in absences %}

<tr>

<td>{{ absence.employe.nom }} {{ absence.employe.prenom }}</td>

<td>{{ absence.get\_type\_absence\_display }}</td>

<td>{{ absence.date\_debut }}</td>

<td>{{ absence.date\_fin }}</td>

</tr>

{% empty %}

<tr>

<td colspan="4">Aucune absence en cours.</td>

</tr>

{% endfor %}

</tbody>

</table>

<a href="{% url 'rapport\_temps\_travail' %}" class="btn btn-info">Voir le rapport de temps de travail</a>

{% endblock %}

* 1. ajouter\_pointage.html :

{% extends 'base.html' %}

{% block content %}

<h1>Ajouter un Pointage</h1>

<form method="post">

{% csrf\_token %}

{{ form.as\_p }}

<button type="submit" class="btn btn-primary">Enregistrer</button>

</form>

{% endblock %}

* 1. modifier\_pointage.html :

{% extends 'base.html' %}

{% block content %}

<h1>Modifier un Pointage</h1>

<form method="post">

{% csrf\_token %}

{{ form.as\_p }}

<button type="submit" class="btn btn-primary">Enregistrer</button>

</form>

{% endblock %}

* 1. ajouter\_absence.html :

{% extends 'base.html' %}

{% block content %}

<h1>Ajouter une Absence</h1>

<form method="post">

{% csrf\_token %}

{{ form.as\_p }}

<button type="submit" class="btn btn-primary">Enregistrer</button>

</form>

{% endblock %}

* 1. rapport\_temps\_travail.html :

{% extends 'base.html' %}

{% block content %}

<h1>Rapport de Temps de Travail</h1>

<form method="get" class="mb-4">

<div class="row">

<div class="col-md-4">

<label for="debut">Date de début :</label>

<input type="date" id="debut" name="debut" value="{{ debut|default:'' }}" class="form-control">

</div>

<div class="col-md-4">

<label for="fin">Date de fin :</label>

<input type="date" id="fin" name="fin" value="{{ fin|default:'' }}" class="form-control">

</div>

<div class="col-md-4 d-flex align-items-end">

<button type="submit" class="btn btn-primary">Générer le rapport</button>

</div>

</div>

</form>

{% if debut and fin %}

<h2>Pointages</h2>

<table class="table">

<thead>

<tr>

<th>Employé</th>

<th>Date</th>

<th>Heure d'arrivée</th>

<th>Heure de départ</th>

</tr>

</thead>

<tbody>

{% for pointage in pointages %}

<tr>

<td>{{ pointage.employe.nom }} {{ pointage.employe.prenom }}</td>

<td>{{ pointage.date }}</td>

<td>{{ pointage.heure\_arrivee|default\_if\_none:"-" }}</td>

<td>{{ pointage.heure\_depart|default\_if\_none:"-" }}</td>

</tr>

{% empty %}

<tr>

<td colspan="4">Aucun pointage trouvé pour cette période.</td>

</tr>

{% endfor %}

</tbody>

</table>

<h2>Absences</h2>

<table class="table">

<thead>

<tr>

<th>Employé</th>

<th>Type d'absence</th>

<th>Date de début</th>

<th>Date de fin</th>

</tr>

</thead>

<tbody>

{% for absence in absences %}

<tr>

<td>{{ absence.employe.nom }} {{ absence.employe.prenom }}</td>

<td>{{ absence.get\_type\_absence\_display }}</td>

<td>{{ absence.date\_debut }}</td>

<td>{{ absence.date\_fin }}</td>

</tr>

{% empty %}

<tr>

<td colspan="4">Aucune absence trouvée pour cette période.</td>

</tr>

{% endfor %}

</tbody>

</table>

{% endif %}

{% endblock %}

N'oubliez pas d'effectuer les migrations après avoir ajouté ces nouveaux modèles :

python manage.py makemigrations

python manage.py migrate

Cette implémentation vous permet de :

1. Enregistrer les pointages quotidiens des employés
2. Gérer les absences
3. Visualiser les pointages et absences du jour
4. Générer des rapports de temps de travail sur une période donnée

Assurez-vous d'ajouter un lien vers cette nouvelle fonctionnalité dans votre dashboard GPEC ou dans le menu de navigation principal.

Cette fonctionnalité vous donnera un outil complet pour gérer le temps de travail de vos employés, ce qui est essentiel pour le suivi des présences, la gestion des heures supplémentaires et l'analyse de la productivité.