Module Odoo Feedback & Application Django

Projet de Développement Web IV - Architecture intégrée pour la gestion des retours d'employés

OBJECTIFS

Les objectifs du projet

Développer un module Odoo

Créer un système complet de gestion des feedbacks employés intégré au framework Odoo

Intégration modulaire

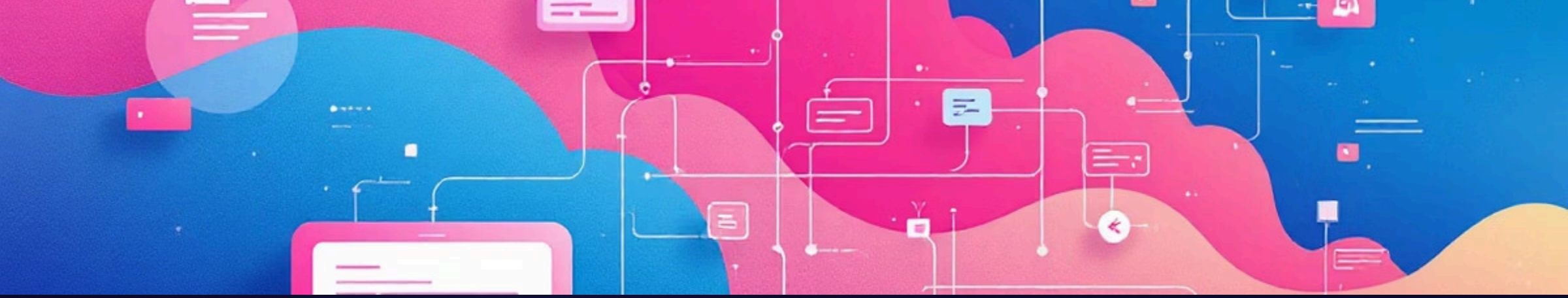
Connecter le module aux fonctionnalités existantes comme le recrutement et les ressources humaines

Communication XML-RPC

Établir une communication robuste entre Odoo et une application Django externe

Interface Django

Développer une application web pour consulter et analyser les feedbacks collectés



Architecture technique du projet

Backend Odoo

- Framework Python pour la logique métier
- Modèles ORM pour la persistance
- API XML-RPC pour l'exposition des données
- Vues et contrôleurs intégrés

Frontend Django

- Application web indépendante
- Connexion via XML-RPC client
- Interface de consultation moderne
- Traitement et affichage des feedbacks

L'architecture repose sur une séparation claire entre la gestion des données (Odoo) et leur présentation (Django), permettant flexibilité et maintenabilité.

MODÈLES

Structure des données : modèles Odoo

1

hr.feedback

Modèle principal stockant les retours d'expérience

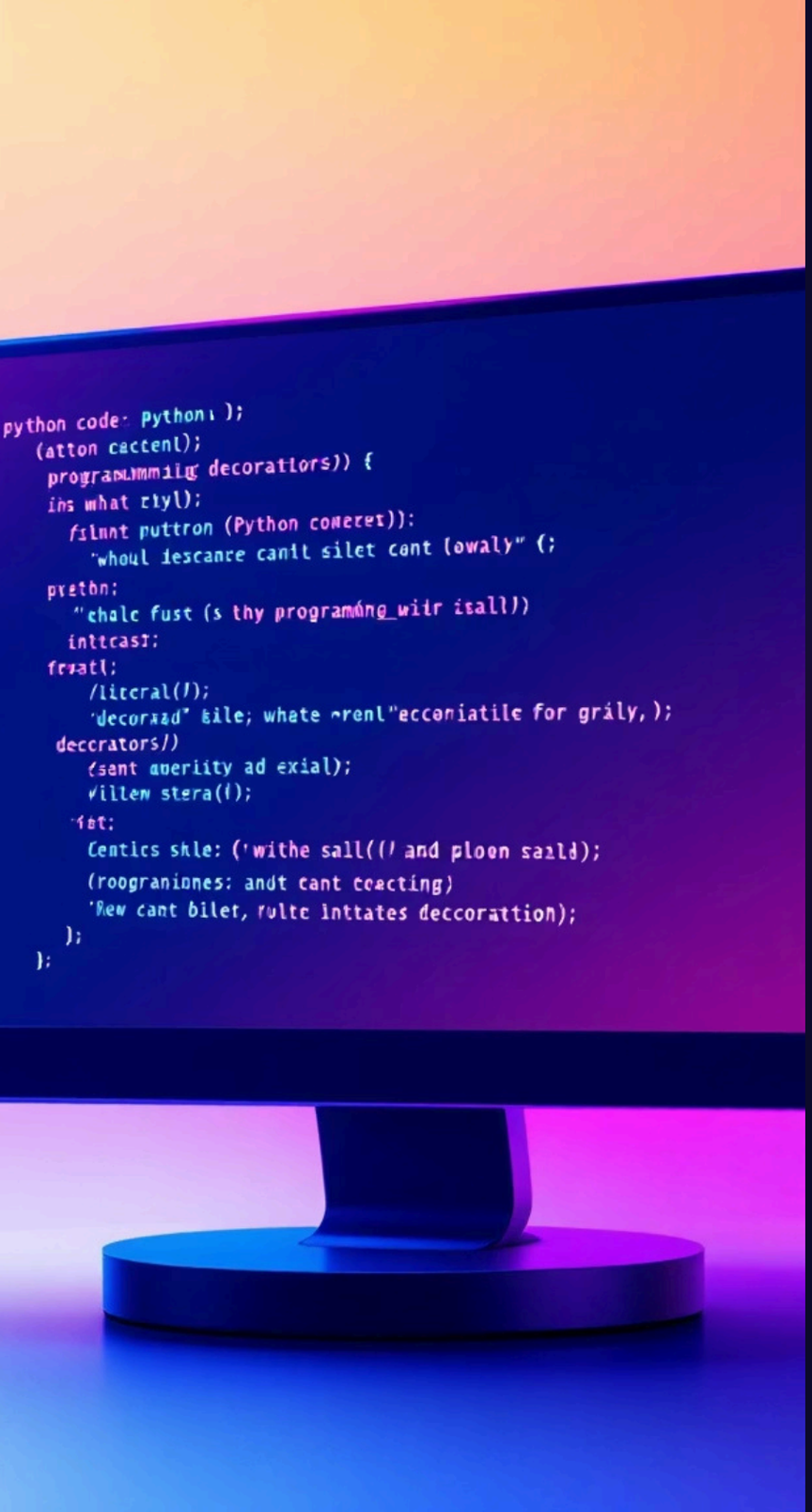
- employee_id : référence à l'employé
- applicant_id : lien avec le recrutement
- feedback_text : contenu du retour
- rating : note d'évaluation
- question_ids : questions associées

2

hr.feedback.question

Questions personnalisables pour structurer les feedbacks

- name : intitulé de la question
- question_type : type (texte, choix multiple)
- feedback_id : relation Many2one
- answer : réponse fournie



Décorateurs Odoo utilisés

01

@api.model

Utilisé pour les méthodes qui ne dépendent pas d'un enregistrement spécifique. Parfait pour les opérations de création et recherche globales.

02

@api.depends

Définit les dépendances des champs calculés. Assure la mise à jour automatique quand les champs sources changent.

03

@api.constrains

Implémente les validations métier. Garantit l'intégrité des données avant la sauvegarde en base.

04

@api.onchange

Réagit aux modifications dans l'interface utilisateur. Permet des mises à jour dynamiques côté client.

Vues Odoo : interfaces utilisateur

Vue Liste (Tree)

Affichage tabulaire des feedbacks avec colonnes clés :

- Nom de l'employé
- Date de création
- Note d'évaluation
- Statut du feedback

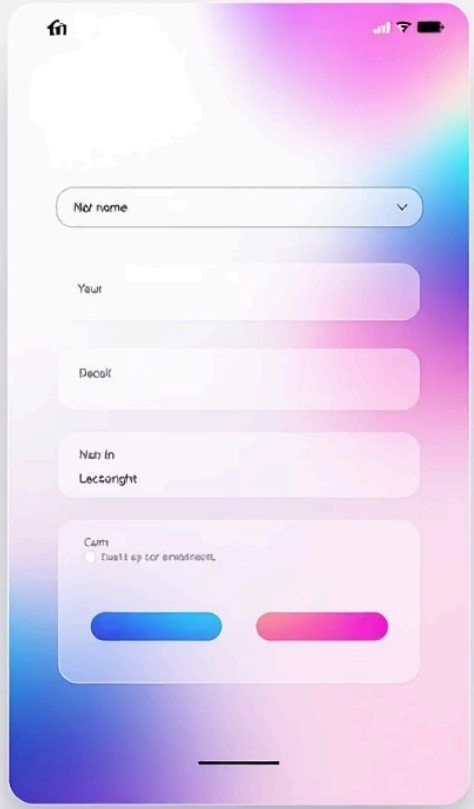
Permet le tri, filtrage et recherche rapide dans l'ensemble des enregistrements.

Vue Formulaire (Form)

Détails complets d'un feedback avec :

- Champs de saisie structurés
- Relations vers employés/candidats
- Liste des questions-réponses
- Actions contextuelles disponibles

Interface intuitive pour la création et modification des retours.



DONNÉES

Données de démonstration

Employés de test

Profils fictifs avec différents rôles et départements pour simuler un environnement réel.

Feedbacks types

Exemples variés de retours : onboarding, performance, satisfaction, avec notes diverses.

Questions prédéfinies

Bibliothèque de questions standard couvrant différents aspects de l'expérience employé.

Les données de démonstration facilitent les tests et permettent aux utilisateurs de découvrir rapidement les fonctionnalités du module.

Intégration avec le module Recruitment

Candidat postule

Création d'un enregistrement applicant dans hr.recruitment

Processus d'entretien

Suivi des étapes de recrutement et évaluations

Feedback lié

Création automatique d'un feedback associé au candidat

Décision éclairée

Utilisation des retours pour le processus de décision

Le champ applicant_id établit une relation directe entre les feedbacks et les candidatures, enrichissant le processus de recrutement.

Communication XML-RPC et application Django

Configuration XML-RPC

- URL du serveur Odoo
- Base de données cible
- Authentification utilisateur
- Endpoint /xmlrpc/2/object

```
import xmlrpc.client

url = 'http://localhost:8069'
db = 'odoo_db'
username = 'admin'
password = 'admin'

common = xmlrpc.client.ServerProxy(
    f'{url}/xmlrpc/2/common'
)
uid = common.authenticate(
    db, username, password, {}
)
```

Application Django

Interface web pour consulter les feedbacks :

- Connexion au serveur Odoo via XML-RPC
- Récupération des enregistrements hr.feedback
- Affichage dans des templates Django
- Filtrage et recherche côté client

L'application offre une vue alternative et personnalisable des données Odoo, adaptée aux besoins spécifiques de consultation.

AMÉLIORATION

Proposition d'amélioration

Filtrage avancé des feedbacks

Ajouter des critères de recherche sophistiqués :

- Filtrage par période temporelle
- Recherche par plage de notes
- Tri par département ou type d'employé
- Filtres combinés et sauvegardés

Analytique et visualisation

Enrichir l'application Django avec :

- Tableaux de bord avec graphiques
- Statistiques d'évolution temporelle
- Export des données au format CSV/Excel
- Alertes sur feedbacks négatifs

Conclusion

Ce projet démontre l'intégration réussie entre Odoo et Django, créant un système complet de gestion des feedbacks. L'architecture modulaire permet des évolutions futures et l'ajout de fonctionnalités avancées selon les besoins métier.