

Odoo

Odoo est un logiciel ERP open-source développé par l'entreprise belge portant le même nom (Odoo SA) fondée par Fabien Pinckaers. Cet ERP est composé de plusieurs modules ayant pour but de simplifier la gestion des entreprises en leur permettant d'utiliser un seul environnement centralisé ce qui élimine les communications croisées ainsi que les problèmes de réplique de données. L'utilisation de l'ERP permet également de faciliter la circulation de l'information entre toutes les fonctions de l'entreprise et de gérer les contacts avec les parties prenantes externes.

Le logiciel Odoo existe en deux versions : Odoo Community qui est un logiciel gratuit contenant les fonctionnalités de base d'Odoo et Odoo Enterprise qui est payant et apporte des fonctionnalités supplémentaires à Odoo Community. Afin de s'adapter aux différents processus de productions des entreprises, Odoo SA fournit également un service de personnalisation pour Odoo Enterprise.

Noviat

Noviat SA est une entreprise située à Bruxelles, fondée en 2009 par Mark Pirenne et Luc de Meyer. C'est une entreprise partenaire de Odoo SA, une autre entreprise belge qui développe un ERP (Enterprise Resource Planning) open-source comprenant de très nombreux modules permettant de répondre à de nombreux besoins de gestion des entreprises.

La société est spécialisée dans l'intégration et le développement sur mesure de solutions basées sur l'ERP Odoo dans les entreprises. Elle est également membre de l'OCA (Odoo Community Association), une organisation qui promeut l'utilisation généralisée d'Odoo et soutient le développement collaboratif des fonctionnalités d'Odoo. Le fait d'être membre permet également à Noviat d'acquérir une certaine renommée en tant qu'intégrateur Odoo.

De manière générale, Noviat aide des entreprises du secteur privé en intégrant des solutions pour différents modules d'Odoo, ou en adaptant certaines fonctionnalités au besoin des entreprises. Cependant, Noviat travaille également avec des administrations du secteur public comme la Ville de Liège, la Commune d'Ixelles et la Commune de Saint-Gilles qui souhaitent intégrer l'ERP dans leur administration. Pour ce dernier type de client, Noviat a développé l'ARP (Administration Resource Planning), une déclinaison d'Odoo conçue spécialement pour le secteur public.

L'ARP permet donc aux administrations publiques de structurer les différents domaines fonctionnels, de faciliter l'accès à une information centralisée et pertinente pour l'ensemble de leurs agents, et aussi d'améliorer leurs services rendus aux citoyens.

SAC

Le système des sanctions administratives communales (SAC), est un système existant depuis 1999 permettant aux communes d'intervenir plus rapidement et plus efficacement contre les nuisances publiques. Ce système leur permet également de collaborer avec les zones de police dans la lutte contre les incivilités ayant un impact négatif sur la qualité de vie des citoyens, en reprenant les infractions punissables de SAC dans leur règlement de police.

Les incivilités punissables de SAC peuvent être constatées par la police locale ou par d'autres personnes, comme les agents communaux et régionaux, et les gardiens de la paix. L'amende administrative est infligée par le fonctionnaire sanctionneur, qui est désigné par le conseil communal.

Les comportements pouvant sanctionnés par une amende administrative sont distingués en trois catégories :

- Les nuisances commises par des personnes
- Les nuisances commises par des établissements
- Les infractions en matière d'arrêt et stationnement

Le constat d'infraction doit être transmis au fonctionnaire sanctionnateur dans les deux mois suivant la constatation de l'infraction. Le fonctionnaire sanctionnateur envoie une copie du constat au contrevenant en lui communiquant les faits de l'infraction.

Le contrevenant a ensuite 15 jours pour se défendre et demander à être entendu si le montant de l'amende prévue est supérieur à 70 euros. Après expiration de ce délai, le fonctionnaire sanctionnateur peut infliger l'amende. Le contrevenant doit alors payer l'amende ou introduire une plainte auprès du tribunal de police dans un délai d'un mois.

Inforius

Actuellement, la plupart des communes de la région de Bruxelles capitale utilisent une application développée par la société Inforius qui leur permet de faciliter grandement leur gestion des SAC. Cette application permet :

- Aux agents constatateurs, d'encoder les données relatives au contrevenant et à l'infraction commise sur un appareil mobile et donc de générer rapidement un constat qui sera ensuite envoyé vers le fonctionnaire sanctionnateur,
- Au fonctionnaire sanctionnateur, d'enregistrer et de suivre des dossiers tout en ayant un aperçu des délais pour chaque dossier et en ayant la possibilité de prendre facilement contact avec les différents acteurs (contrevenant, constatateur, médiateur, ...). Le fonctionnaire peut également intégrer des modèles de document et générer automatiquement des rapports et des statistiques.

Inforius possède un partenariat avec la société Ipex, une entreprise spécialisée dans les solutions d'impression, ce qui permet à l'application de s'occuper automatiquement de l'impression et de l'envoi du courrier du fonctionnaire sanctionnateur.

Use case

Le but de ce travail est donc d'offrir une solution qui permettra à différents acteurs de réaliser des actions similaires à celles qu'ils peuvent effectuer avec l'application d'Inforius tout en étant intégré dans l'ARP. Le travail sera constitué en deux parties :

- Une application mobile qui permettra aux constatateurs de relever des infractions tout en leur permettant d'y associer des photos prises sur le terrain. L'application devra également pouvoir utiliser la géolocalisation et fonctionner hors ligne. La base de données de l'application doit également pouvoir se synchroniser avec le serveur Odoo.
- Un module Odoo qui permettra au fonctionnaire sanctionnateur de valider ou non les sanctions constatées et de gérer et suivre les dossiers. Il devra également pouvoir prendre contact avec les différents acteurs même avec l'huissier si nécessaire.

Flutter

Le développement de l'application se fera avec Flutter, un framework open-source qui existe depuis 2018 et développé par Google. Les applications Flutter sont écrites en Dart, un langage de programmation orienté objet utilisant le moteur graphique Skia C++, également conçu par Google.

Le principe de Flutter repose sur l'utilisation des widgets et des composants. Les widgets permettent d'utiliser la programmation orienté objet en ayant directement un rendu sur l'interface utilisateur. Concrètement les widgets représentent chaque bouton et texte affiché dans l'interface.

Selon une enquête réalisée en 2020 par Statista, Flutter est le deuxième framework de développement d'applications multiplateforme le plus utilisé par les développeurs de logiciels dans le monde avec une forte augmentation (9%) d'utilisateurs en 2020.

Flutter est un framework multiplateforme ce qui permet de développer des applications pour Android, iOS, Linux, Mac, Windows, Google Fuchsia et le web à partir d'une seule base de code. On évite ainsi les problèmes de cohérence de l'interface utilisateur sur différentes plates-formes. De plus grâce au moteur Skia, les applications Flutter ne sont pas affectées par une mise à jour ou une personnalisation du système d'exploitation et garderont donc la même apparence.

Le langage Dart propose deux modes de compilation : JIT (Just In time) et AOT (Ahead Of Time). Avec AOT le code sera optimisé directement pour l'architecture sur laquelle il fonctionnera ce qui permet ainsi d'avoir de meilleures performances et de réduire la taille de l'application. JIT offre la fonctionnalité de Hot Reload lors des compilations, ce qui permet de réduire le temps entre chaque build à quelques millisecondes et facilite ainsi le développement des applications. A titre de comparaison, la compilation d'une application Android de taille moyenne habituelle prend au moins 40 secondes pour être expédiée sur l'appareil de test.

Fonctionnement général

1. L'utilisateur doit tout d'abord s'authentifier à l'aide de son identifiant et son mot de passe avant de pouvoir utiliser l'application.
2. L'utilisateur est ensuite renvoyé vers la page d'accueil où il a la possibilité de visualiser la liste des SAC et voir si elles ont été envoyées vers le serveur Odoo ou non, de créer de nouvelles SAC et de synchroniser les données de l'application avec celles du serveur Odoo.
3. L'utilisateur peut également accéder aux informations et à la liste des sanctions d'un contrevenant en particulier.

Réalisation

Lorsqu'il ouvre l'application, l'utilisateur doit tout d'abord s'authentifier à l'aide son identifiant et de son mot de passe. La première fois qu'il se connecte, l'application doit être connectée au serveur afin de copier les données du serveur Odoo de tous les constatateurs dans la base de données Flutter.

Après l'authentification, l'utilisateur est dirigé vers la page d'accueil qui affiche la liste de toutes les SAC qui se trouvent dans la base de données de l'application.

Lorsque l'utilisateur constate une nouvelle infraction, il peut créer une nouvelle sanction en indiquant dans le formulaire si le contrevenant est une entreprise ou une personne ainsi que son numéro d'identification, le type de sanction, ainsi qu'une description de l'infraction. Il peut également y joindre une ou plusieurs photos. Au moment où la sanction est enregistrée, la date et l'heure de création sont également enregistré et le lieu où l'infraction a été constaté est détecté par

géolocalisation. On vérifie également si le contrevenant n'est pas déjà répertorié dans la base de données, et si ce n'est pas le cas, un nouvel élément est ajouté dans la table Offender.

Une fois la SAC créée, il est toujours possible de la modifier ou de la supprimer tant qu'elle n'a pas encore été envoyée vers le serveur Odoo. On ne peut cependant pas modifier le lieu et la date du constat de l'infraction.

Aspects techniques d'Odoo

Odoo est un logiciel basé sur une architecture client-serveur dans laquelle les clients sont des navigateurs web accédant au serveur Odoo à l'aide du protocole JSON-RPC (Remote Procedure Call).

L'architecture d'Odoo se compose de deux tiers :

- Le serveur de base de données PostgreSQL qui gère l'ensemble bases de données Odoo qui contiennent toutes les données des applications, et aussi la plupart des éléments de configuration du système Odoo.
- Le serveur Odoo qui contient toute la logique d'entreprise et assure le fonctionnement d'Odoo. Ce serveur fournit également un ORM (Object-Relational Mapping) qui s'occupe de faire la liaison entre les objets Python et la base de données PostgreSQL tout en gérant les relations entre objet. Le serveur Odoo utilise Werkzeug comme serveur WSGI (Web Server Gateway Interface) qui permet la communication entre un serveur web et une application python.

Le fonctionnement d'Odoo se base sur le patron de conception MVC qui permet de séparer les données et l'interface utilisateur en introduisant un composant intermédiaire : le contrôleur.

Odoo permet d'étendre un module existant de manière modulaire grâce à trois mécanismes d'héritage différents :

- L'héritage classique : on crée un nouveau modèle à partir d'un modèle déjà existant, les changements apportés aux nouveaux modèles ne modifient pas le modèle de référence.
- L'héritage par extension : on rajoute des nouveaux champs à un modèle déjà existant.
- L'héritage par délégation : ce type d'héritage permet de créer un nouveau modèle héritant de plusieurs modèles déjà existant à la fois.

Fonctionnement général

Après s'être connecter au serveur et au moment où il accède au module, l'utilisateur (l'agent traitant) accède à la liste de toutes les nouvelles SAC qui viennent d'être envoyées depuis l'application Flutter. Il peut ensuite accéder aux détails de chaque sanctions et décider de valider ou non la SAC. L'utilisateur a également accès à la liste des SAC validées ou il peut imprimer une SAC selon un modèle d'impression qu'il aura choisi afin de pouvoir l'envoyer au contrevenant.

L'utilisation peut également prendre contact avec le constatateur d'une SAC et assigner le poste d'agent constatateur à d'autres employés de l'organisation.

Conception

Trois modèles ont été définis au sein du module, chacun lié à une table de la base de données grâce à l'ORM :

- `sac` : modèle qui représente une SAC
- `type_sac` : représente un type de SAC

- offender : héritage par extension du modèle res.partner et représente un contrevenant.

Dans Odoo, l’affichage est basé sur une arborescence de menus, chaque menu possédant des sous menus menant à des actions qui mènent vers des vues. Une vue définit la manière dont les objets d’un modèle sont affichés.

- Liste des SAC : on utilise une ‘tree view’ qui permet d’afficher les éléments d’un modèle sous forme de table ainsi qu’une ‘search view’ qui permet de filtrer les objets du modèle
- Détails d’une SAC : on utilise une ‘form view’ qui permet d’afficher les données d’un seul objet du modèle

Plusieurs routes ont été définies dans le contrôleur afin de permettre la synchronisation des données avec l’application Flutter (plus de détails au chapitre suivant)

- /sac_module/new_offender : permet d’ajouter un nouveau contrevenant
- /sac_module/new_sac : permet d’ajouter une nouvelle SAC
- /sac_module/new_picture : permet d’ajouter de nouvelles images associées à une SAC
- /sac_module/get_sac : renvoie toutes les SAC d’un constatateur en particulier
- /sac_module/get_users : renvoie les utilisateurs des employés assignés aux postes d’agent constatateur

Synchronisation connexion

La première fois que l’on s’authentifie dans l’application Flutter, la base de données Flutter copie les données de la table res.users depuis le serveur Odoo dans sa table User. L’application recopie uniquement les données des utilisateurs assignés au poste d’agent constatateur. Pour cela, on commence par chercher l’ensemble des objets contenu dans la table hr.employee.job dont le champ name = ‘Agent constatateur’, et l’on récupère la valeur du champ employee_id. Enfin, on récupère l’objet de la table res.user dont la valeur du champ employee_id correspond à la valeur récupérée précédemment.

Synchronisation des SAC

Après authentification, l’ensemble des données des SAC du constatateur lié à l’utilisateur, sont copiées dans la base de données Flutter depuis le serveur Odoo. Pour cela, on cherche l’ensemble des objets contenus dans la table ‘sac’ dont le champ ‘employee_id’ correspond à l’id de l’utilisateur.

Synchronisation des données des contrevenants et des types de SAC

Au moment où la base de données Flutter copie les données des SAC du serveur, les données des types de SAC et des contrevenants liés aux SAC, contenus dans les tables sac_type et res_partner, sont également copiées dans les tables SAC_type et Offender de Flutter.

Au moment où l’on envoie de nouvelles SAC de l’application Flutter vers le serveur, on vérifie tout d’abord si les informations liées au contrevenant n’existent pas déjà dans la base de données Odoo en comparant le champ ‘id_register’.