

Cahier des Charges pour l'Application Web de Gestion de Bureau d'Ordre

1. Description du Projet

Titre du projet : Application web de gestion de bureau d'ordre

Contexte : Les entreprises reçoivent souvent des demandes sous forme de documents physiques ou numériques qui doivent être traitées par différents services. Une application web centralisée permettra d'automatiser et de simplifier ce processus, améliorant ainsi l'efficacité et la traçabilité des demandes.

Objectif : Concevoir et développer une application web permettant de gérer les demandes de clients via un bureau d'ordre d'une entreprise, centralisant et automatisant le traitement des documents physiques ou numériques, améliorant ainsi l'efficacité et la traçabilité des demandes.

2. Fonctionnalités Principales de l'Application

Gestion des Utilisateurs

- Inscription et Connexion : Création de comptes et accès sécurisé.
- Gestion des Rôles et Permissions : Attribution de rôles spécifiques (client, agent, administrateur).

Gestion des Demandes

- Création de Demandes : Téléchargement et numérisation de documents par les clients.
- Affectation des Demandes : Assignment automatique des demandes aux services appropriés.
- Suivi des Demandes : Suivi de l'état des demandes par les clients et les agents.

Gestion des Services

- Création et Modification des Services : Gestion des services internes.
- Définition des Critères d'Affectation : Configuration des règles d'affectation des demandes.

Consultation et Recherche

- Recherche de Demandes : Recherche par critères (date, type de document, service).
- Consultation des Détails des Demandes : Vue détaillée des demandes.

Sécurité et Confidentialité

- Chiffrement des Données : Protection des données sensibles.

3. Technologies et Environnements Utilisés

- Backend : C#, .NET 8, API REST
- Frontend: Vue.JS 3, TypeScript, HTML, CSS, JavaScript
- Base de données : SQL Server
- Outils de développement : Visual Studio, Visual Studio Code
- Outils de collaboration : Azure DevOps, GitHub, Teams
- Design : Figma

4. Description des Entités

1. Utilisateur

- ID : Identifiant unique
- Nom : Nom de l'utilisateur
- Email : Adresse email
- Mot de passe : Mot de passe chiffré
- Rôle : Rôle de l'utilisateur (client, agent, administrateur)

2. Demande

- ID : Identifiant unique
- UtilisateurID : Identifiant de l'utilisateur ayant soumis la demande
- DateSoumission : Date de soumission de la demande
- Statut : Statut actuel de la demande (en attente, en cours, traitée)
- Documents : Liste des documents associés

3. Document

- ID : Identifiant unique
- DemandelD : Identifiant de la demande associée
- NomFichier : Nom du fichier
- Type : Type de document
- URL : Lien de téléchargement du document

4. Service

- ID : Identifiant unique
- Nom : Nom du service
- Description : Description du service

5. Analyse des Besoins Fonctionnels et Non Fonctionnels

Besoins Fonctionnels :

1. Gestion des Utilisateurs

- **Inscription** : Les utilisateurs peuvent créer un compte avec des informations de base (nom, email, mot de passe).
- **Connexion** : Les utilisateurs peuvent se connecter en utilisant leur adresse email et leur mot de passe.
- **Gestion des rôles** : Les administrateurs peuvent attribuer des rôles (client, agent, administrateur) et définir des permissions spécifiques pour chaque rôle.

Scénario : Inscription et Connexion

- Un nouveau client visite l'application et souhaite soumettre une demande. Il clique sur "S'inscrire", remplit le formulaire avec son nom, email et mot de passe, puis **soumet le formulaire. Le système crée un nouveau compte utilisateur et l'utilisateur reçoit un email de confirmation. Ensuite, il se connecte avec ses nouvelles informations d'identification.**

2. Gestion des Demandes

- **Création de Demandes** : Les clients peuvent soumettre des demandes en téléchargeant des documents. Les agents peuvent créer des demandes en scannant et téléchargeant des documents physiques.

- **Affectation des Demandes** : Les demandes sont automatiquement assignées à des services en fonction de la nature des documents. Les agents peuvent aussi manuellement affecter ou réaffecter des demandes.

- **Suivi des Demandes** : Les clients et les agents peuvent consulter l'état des demandes (en attente, en cours, traitée). Les agents peuvent mettre à jour le statut des demandes et ajouter des commentaires.

Scénario : Création et Suivi d'une Demande

- Un client se connecte, accède à la page de création de demande, télécharge plusieurs documents et soumet la demande. L'application crée une demande, notifie un agent et affiche la demande dans le tableau de bord du client. L'agent attribue la demande au service approprié. Le client reçoit des notifications par email à chaque mise à jour de statut et peut suivre l'état de sa demande dans son tableau de bord.

3. Gestion des Services

- **Création et Modification des Services** : Les administrateurs peuvent ajouter, modifier et supprimer des services. Chaque service a un nom et une description.

- **Définition des Critères d'Affectation** : Les administrateurs définissent les critères d'affectation pour automatiser la distribution des demandes aux services appropriés.

Scénario : Gestion des Services

- Un administrateur se connecte, accède à la page de gestion des services, ajoute un nouveau service "Support IT" avec une description détaillée. Il définit ensuite des critères d'affectation basés sur des mots-clés présents dans les documents soumis.

4. Consultation et Recherche

- **Recherche de Demandes** : Les utilisateurs peuvent rechercher des demandes en utilisant des critères comme la date de soumission, le type de document, et le service assigné.

- **Consultation des Détails des Demandes** : Les utilisateurs peuvent consulter les détails de chaque demande, y compris les documents soumis, l'état actuel, et l'historique des mises à jour.

Scénario : Recherche et Consultation des Demandes

- Un agent se connecte, accède à la page de recherche des demandes, entre des critères de recherche pour trouver toutes les demandes en cours pour le service juridique. Il clique sur une demande spécifique pour voir tous les détails, y compris les documents soumis et les commentaires des autres agents.

5. Sécurité et Confidentialité

- **Chiffrement des Données** : Toutes les données sensibles, y compris les mots de passe et les documents scannés, doivent être chiffrées pour garantir la confidentialité.
- **Gestion des Sessions** : Les sessions utilisateur doivent être sécurisées et expirer après une période d'inactivité.

Scénario : Sécurité des Données

- Un utilisateur se connecte, et toutes les communications entre son navigateur et le serveur sont sécurisées par SSL/TLS. Lorsqu'il télécharge un document, ce document est chiffré avant d'être stocké dans la base de données. Après 15 minutes d'inactivité, la session de l'utilisateur expire automatiquement, nécessitant une nouvelle connexion.

Besoins Non Fonctionnels :

- **Performance** : Le temps de réponse doit être inférieur à 2 secondes pour toutes les opérations courantes.
- **Sécurité** : Les données sensibles doivent être protégées par des mécanismes de chiffrement et les sessions utilisateur doivent être sécurisées.
- **Scalabilité** : L'application doit pouvoir gérer une augmentation du nombre d'utilisateurs et de demandes sans dégradation significative des performances.
- **Disponibilité** : L'application doit être disponible 99,9% du temps, avec des mécanismes de tolérance aux pannes en place.

6. Méthodologie

- Gestion de projet : Méthodologie agile avec des sprints de 2 semaines.

- Suivi des tâches : Utilisation d'Azure DevOps pour la gestion des tâches et le suivi des progrès.
- Collaboration : Utilisation de GitHub pour le contrôle de version et la collaboration entre les développeurs.

7. Planification

1. Analyse des besoins : 1 semaine
2. Conception de la maquette : 1 semaine
3. Développement de l'API : 2 semaines
4. Développement du frontend : 1 semaine
5. Intégration et tests : 2 semaines
6. Mise en production et documentation : 1 semaine

Durée totale estimée : 8 semaines

8. Contraintes et Risques

Contraintes :

- Disponibilité et compétence des membres de l'équipe.
- Respect des délais.
- Compatibilité

Risques :

- Problèmes d'intégration des différents modules.
- Difficultés techniques imprévues.