

Cahier des charges pour l'application TECH SERVICES



1. Introduction

Le présent Cahier des Charges Technique (CCT) énonce les exigences techniques, fonctionnelles et opérationnelles pour le développement de l'application mobile et web Tech Services. Cette application a pour objectif de faciliter la mise en relation entre les clients et les prestataires du secteur du BTP, en intégrant un système de paiement pour les clients et un modèle d'abonnement pour les prestataires.

1.1 Objectifs du CCT

L'objectif principal de ce CCT est de définir clairement les spécifications techniques de l'application Tech Services. Il vise à établir une base solide pour le développement du système, garantissant ainsi une mise en œuvre réussie des fonctionnalités requises.

1.2 Contexte du Projet

Le secteur du BTP nécessite une solution efficace pour la gestion des demandes de services et des interventions. Tech Services se positionne comme une application mobile web innovante, visant à simplifier le processus de recherche, de réservation et d'exécution de services techniques, tout en introduisant des fonctionnalités de paiement sécurisé pour les clients.

1.3 Portée du Projet

L'application Tech Services permettra aux clients de consulter la liste des prestataires, de sélectionner des services adaptés à leurs besoins, de soumettre des demandes, et de payer les prestataires via la plateforme. Les prestataires auront la possibilité de souscrire à des abonnements pour accéder à des fonctionnalités avancées. La portée du projet englobe le développement, le déploiement et la maintenance de l'application.

1.4 Structure du Document

Le CCT est organisé de manière à couvrir toutes les facettes du projet. Chaque section du document aborde des aspects spécifiques du développement, allant des exigences fonctionnelles aux détails de mise en œuvre du système de paiement et du modèle d'abonnement.

1.5 Parties Prenantes

Les principales parties prenantes du projet incluent les clients, les prestataires, les développeurs, les testeurs, les concepteurs, ainsi que les responsables financiers. La collaboration efficace entre ces parties est cruciale pour garantir la réussite du projet.

1.6 Terminologie et Abréviations

Un glossaire détaillant la terminologie spécifique au projet et les abréviations utilisées dans le CCT est fourni en annexe. Ceci vise à assurer une compréhension claire et unifiée entre toutes les parties prenantes.

1.7 Révisions du Document

Le document peut être révisé tout au long du développement pour refléter les changements et les ajustements nécessaires. Chaque révision sera documentée avec une justification claire.

1.8 Approbation du Document

Avant le début effectif du développement, ce document doit être approuvé par toutes les parties prenantes. L'approbation confirme l'engagement et la compréhension mutuelle des objectifs et des spécifications du projet.

Ce CCT établit un cadre pour le développement de l'application Tech Services, intégrant un système de paiement pour les clients et un modèle d'abonnement pour les prestataires. Son approbation marque le démarrage du processus de développement.

2. Description Générale du Projet

L'application Tech Services a pour objectif de simplifier et d'optimiser la mise en relation entre les clients et les prestataires du secteur BTP (Bâtiment et Travaux Publics). Cette section offre une vue d'ensemble des objectifs, des fonctionnalités clés et des avantages du projet.

2.1 Objectifs du Projet

Les objectifs principaux de l'application "Tech Services" sont les suivants :

Faciliter la recherche et la sélection de prestataires qualifiés par les clients.

Proposer une variété de services du secteur du BTP pour répondre aux besoins divers des clients.

Intégrer un système de paiement sécurisé pour les transactions entre clients et prestataires.

Mettre en place un modèle d'abonnement pour les prestataires afin d'accéder à des fonctionnalités avancées.

Optimiser le processus de gestion des interventions pour les prestataires.

2.2 Fonctionnalités Clés

2.2.1 Consultation des Prestataires

Les clients peuvent parcourir une liste de prestataires comprenant des techniciens et des entreprises du BTP.

Affichage des profils des prestataires avec des informations détaillées.

2.2.2 Sélection de Services

Les clients peuvent choisir parmi une liste de services proposés en fonction de leurs besoins spécifiques.

2.2.3 Localisation des Prestataires

Affichage des emplacements des prestataires sur une carte interactive pour une sélection facilitée.

2.2.4 Soumission de Demandes de Services

Les clients peuvent soumettre des demandes de services avec une description détaillée de leurs besoins.

2.2.5 Système de Paiement

Intégration d'un système de paiement sécurisé pour les transactions entre clients et prestataires.

2.2.6 Modèle d'Abonnement pour les Prestataires

Mise en place d'un modèle d'abonnement pour les prestataires, offrant des avantages spécifiques en fonction du niveau d'abonnement.

2.2.7 Validation des Devis

Les clients peuvent valider les devis proposés par les prestataires pour démarrer les travaux.

2.2.8 Notation des Services

Les clients peuvent évaluer et noter les services rendus par les prestataires.

2.3 Avantages du Projet

Les avantages clés de l'application "Tech Services" incluent :

Simplification du processus de recherche et de réservation de services du BTP.

Gain de temps pour les clients dans la sélection de prestataires qualifiés.

Optimisation du workflow pour les prestataires grâce à une gestion centralisée des demandes et des interventions.

Sécurisation des transactions financières grâce à un système de paiement intégré.

Amélioration de la confiance entre clients et prestataires grâce à un système de notation transparent.

L'application Tech Services vise à révolutionner la manière dont les services du BTP sont recherchés, réservés et évalués, offrant ainsi une solution complète et efficace pour les clients et les prestataires du secteur.

3. Exigences Fonctionnelles

Les exigences fonctionnelles détaillent les fonctionnalités spécifiques que l'application Tech Services doit offrir pour atteindre ses objectifs. Ces exigences couvrent divers aspects, de la consultation des prestataires à la validation des devis et à la notation des services.

3.1 Consultation des Prestataires

3.1.1 Liste des Prestataires

Afficher une liste de prestataires, comprenant des techniciens et des entreprises du secteur du BTP.

Fournir des filtres de recherche permettant aux clients de trouver des prestataires en fonction de critères spécifiques.

3.1.2 Profils des Prestataires

Afficher des profils détaillés pour chaque prestataire, incluant les compétences, les certifications, les évaluations, et les informations de contact.

Intégrer une option de messagerie pour faciliter la communication initiale entre le client et le prestataire.

3.2 Sélection de Services

3.2.1 Liste de Services

Proposer une liste de services du secteur BTP couvrant une variété de besoins client.

Permettre aux clients de sélectionner les services pertinents pour leur situation.

3.3 Localisation des Prestataires

3.3.1 Carte Interactive

Intégrer une carte interactive affichant les emplacements des prestataires.

Permettre aux clients de choisir des prestataires en fonction de leur proximité géographique.

3.4 Soumission de Demandes de Services

3.4.1 Formulaire de Demande

Mettre à disposition un formulaire de demande de services comprenant des champs pour une description détaillée du problème du client.

Permettre aux clients d'ajouter des images ou des fichiers pertinents à leur demande.

3.4.2 Notifications aux Prestataires

Envoyer des notifications instantanées aux prestataires concernés lorsqu'une nouvelle demande de service est soumise.

3.5 Système de Paiement

3.5.1 Enregistrement des Informations de Paiement

Permettre aux clients d'enregistrer en toute sécurité leurs informations de paiement.

Intégrer plusieurs options de paiement, y compris cartes de crédit , OM, MoMo.

3.5.2 Paiement des Prestataires

Faciliter le paiement sécurisé des prestataires via la plateforme après la validation d'un service.

3.6 Modèle d'Abonnement pour les Prestataires

3.6.1 Types d'Abonnements

Définir différents niveaux d'abonnements pour les prestataires (par exemple, basique, premium).

Spécifier les fonctionnalités accessibles à chaque niveau d'abonnement.

3.6.2 Processus d'Abonnement

Mettre en place une interface intuitive pour que les prestataires souscrivent et gèrent leurs abonnements.

Intégrer une passerelle de paiement pour les transactions d'abonnement.

3.7 Validation des Devis

3.7.1 Affichage des Devis

Afficher les devis proposés par les prestataires dans une section dédiée de l'application.

Inclure les détails spécifiques des travaux, le coût estimatif, et le statut actuel du devis.

3.7.2 Validation des Devis

Permettre aux clients de valider ou de refuser un devis.

En cas de validation, déclencher le début des travaux.

3.8 Notation des Services

3.8.1 Interface de Notation

Permettre aux clients de noter et de donner un retour sur le service rendu par le prestataire après la conclusion des travaux.

Inclure une échelle de notation et la possibilité d'ajouter des commentaires.

3.8.2 Effets de la Notation

Utiliser les notations pour influencer la visibilité des prestataires sur la plateforme.

Assurer la confidentialité des notations entre le client et le prestataire.

Ces exigences fonctionnelles forment la base du développement de l'application Tech Services, garantissant la mise en œuvre efficace des fonctionnalités clés pour une expérience utilisateur optimale.

4. Exigences Non-Fonctionnelles

Les exigences non-fonctionnelles définissent les caractéristiques de l'application Tech Services qui ne concernent pas directement ses fonctionnalités, mais qui sont essentielles pour garantir ses performances, sa sécurité, et son adoption par les utilisateurs.

4.1 Performance

4.1.1 Temps de Réponse

L'application doit garantir un temps de réponse optimal pour les interactions utilisateur, ne dépassant pas 4 secondes dans des conditions normales d'utilisation.

4.1.2 Capacité de Charge

L'application doit être capable de gérer simultanément un nombre important d'utilisateurs, avec une capacité minimale de 10 000 utilisateurs actifs simultanément.

4.1.3 Stabilité

La stabilité de l'application doit être assurée même en période de charge élevée, minimisant les risques de crash ou d'indisponibilité.

4.2 Sécurité

4.2.1 Sécurité des Données Utilisateur

Les données utilisateur, y compris les informations de paiement, doivent être stockées de manière sécurisée conformément aux normes de sécurité PCI-DSS.

4.2.2 Authentification et Autorisation

Mise en place d'un système d'authentification robuste pour assurer que seuls les utilisateurs autorisés ont accès à leurs comptes.

Gestion stricte des autorisations pour les actions sensibles telles que la validation des devis et le paiement.

4.2.3 Protection contre les Attaques

Mise en œuvre de mécanismes de protection contre les attaques telles que l'injection SQL, la falsification de requêtes, et les attaques par force brute.

4.3 Interface Utilisateur

4.3.1 Convivialité

L'interface utilisateur doit être conviviale et intuitive, facilitant la navigation pour les utilisateurs de tous niveaux de compétence.

4.3.2 Accessibilité

L'application doit être accessible aux utilisateurs ayant des besoins spécifiques, conformément aux normes d'accessibilité WCAG.

4.4 Compatibilité

4.4.1 Compatibilité des Navigateurs

L'application doit être compatible avec les principaux navigateurs web mobiles tels que Chrome, Safari, Firefox, et Edge.

4.4.2 Compatibilité des Dispositifs

L'application doit être conçue pour s'adapter aux différentes tailles d'écran des dispositifs mobiles, assurant une expérience utilisateur cohérente.

4.5 Disponibilité

4.5.1 Temps de Disponibilité

Garantir un temps de disponibilité de l'application de 99,9%, avec des fenêtres de maintenance minimales.

4.5.2 Sauvegarde et Récupération

Mettre en place des mécanismes de sauvegarde réguliers pour assurer la récupération rapide des données en cas de défaillance.

4.6 Réglementation et Conformité

4.6.1 Conformité Légale

L'application doit respecter toutes les réglementations locales et internationales relatives à la protection des données et aux transactions en ligne.

4.6.2 Confidentialité des Données

Assurer la confidentialité des données des utilisateurs et la transparence quant à la manière dont les données sont utilisées.

4.7 Évolutivité

4.7.1 Extensibilité

L'architecture de l'application doit permettre une extensibilité facile pour l'ajout de nouvelles fonctionnalités à l'avenir.

4.7.2 Gestion des Mises à Jour

Mettre en place un processus efficace pour la gestion des mises à jour logicielles afin de garantir que l'application reste à jour en termes de sécurité et de fonctionnalités.

4.8 Documentation

4.8.1 Documentation Technique

Fournir une documentation technique exhaustive pour les développeurs, couvrant l'architecture, les API, et les détails d'implémentation.

4.8.2 Documentation Utilisateur

Élaborer une documentation utilisateur complète pour guider les clients et les prestataires dans l'utilisation de l'application.

Ces exigences non-fonctionnelles sont essentielles pour garantir le succès de l'application Tech Services en termes de performances, de sécurité, d'expérience utilisateur, et de conformité aux normes réglementaires.

5. Exigences Techniques - Tech Services

Les exigences techniques détaillent les spécifications techniques nécessaires à la conception, au développement, et au déploiement de l'application Tech Services. Ces exigences couvrent des aspects tels que l'infrastructure, les langages de programmation, les bases de données, et les normes de développement.

5.1 Langages de Programmation et Cadres de Développement

5.1.1 Frontend

Utilisation d'un framework JavaScript moderne pour le développement du frontend, tel que React Native.

Assurer une conception réactive pour garantir une expérience utilisateur optimale sur une variété de dispositifs.

5.1.2 Backend

Choix d'un langage de programmation côté serveur robuste, comme Node.js, et PHP(Laravel)

Utilisation d'un framework backend pour accélérer le développement et garantir une architecture solide.

5.2 Base de Données

5.2.1 Type de Base de Données

Utilisation d'une base de données relationnelle telle que MySQL pour stocker les données critiques telles que les profils des prestataires et les transactions.

5.2.2 Sécurité des Données

Mise en place de mécanismes de cryptage des données sensibles.

Utilisation de techniques de hachage pour les informations confidentielles comme les mots de passe.

5.3 Infrastructure Cloud

5.3.1 Hébergement

Utilisation d'un service d'hébergement cloud fiable et évolutif, tel que AWS, Azure ou Google Cloud Platform.

5.3.2 Évolutivité et Tolérance aux Pannes

Mise en place d'une architecture cloud permettant l'évolutivité horizontale pour gérer la montée en charge.

Configuration de mécanismes de tolérance aux pannes pour assurer une disponibilité continue.

5.4 API et Intégrations

5.4.1 API Backend

Mise en place d'une API robuste pour permettre l'intégration avec d'autres services et applications.

5.4.2 Intégration de Services de Paiement

Intégration d'un service de paiement sécurisé, en respectant les normes PCI-DSS.

5.5 Sécurité de l'Application

5.5.1 Protocole HTTPS

Activation de HTTPS pour toutes les communications entre le client et le serveur, garantissant la confidentialité et l'intégrité des données.

5.5.2 Pare-feu et Filtres de Sécurité

Configuration de pare-feu pour protéger l'application contre les attaques externes.

Utilisation de filtres de sécurité pour détecter et prévenir les injections de code.

5.6 Développement Agile

5.6.1 Méthodologie Agile

Adoption d'une approche de développement agile, avec des cycles de développement itératifs et des livraisons continues.

5.6.2 Gestion de Projet

Utilisation d'outils de gestion de projet (**Jira**) pour suivre les progrès, gérer les tâches et assurer une communication transparente.

5.7 Normes et Conventions de Codage

5.7.1 Conventions de Codage

Adoption de conventions de codage claires pour assurer une cohérence au sein de l'équipe de développement.

5.7.2 Documentation du Code

Documentation complète du code source, y compris des commentaires explicatifs et des guides pour faciliter la maintenance et les mises à jour.

Ces exigences techniques servent de fondement pour le développement de l'application "Tech Services". Elles garantissent une architecture solide, une sécurité robuste, et une évolutivité nécessaire pour répondre aux besoins des utilisateurs.

6. Architecture système

- Les différents diagrammes

Diagramme de cas d'utilisation

Diagramme de class

MLD

7. Interfaces

a. Frontend Web - Système d'Administration (Super Admin) :

a.1 Interface du Super Admin :

Le super admin accède à un tableau de bord administratif complet.

Fonctionnalités de gestion des utilisateurs, des prestataires, des services, et des transactions.

Suivi des activités globales et génération de rapports.

a.2 Panneau d'Administration Entreprise (Admin-Entreprise) :

Les administrateurs d'entreprises ont accès à un tableau de bord dédié.

Gestion des profils d'entreprise, des prestataires affiliés, et des demandes de service.

Suivi des performances de l'entreprise et des statistiques clés.

b. Interface Mobile - Clients :

b.1 Application Mobile pour les Clients :

Les clients accèdent à une interface conviviale sur leurs appareils mobiles.

Recherche de prestataires, soumission de demandes de service, et suivi des interventions en cours.

Possibilité de noter et de laisser des commentaires sur les services rendus.

c. Interface Mobile - Prestataires Techniciens :

c.1 Application Mobile pour les Techniciens :

Les techniciens prestataires accèdent à une interface mobile dédiée.

Consultation des demandes de service, gestion des devis, et suivi des interventions.

Accès aux outils de diagnostic et de gestion des interventions sur le terrain.

Cette répartition des interfaces garantit une expérience utilisateur optimale en fonction des besoins spécifiques de chaque type d'utilisateur dans l'application "Tech Services". Les interfaces sont adaptées pour fournir les fonctionnalités et les outils nécessaires à chaque rôle, qu'il s'agisse d'administrateurs, d'entreprises, de clients, ou de techniciens prestataires.

8. Tests et Validation - Tech Services

Les tests et la validation sont des étapes cruciales pour garantir la qualité et la fiabilité de l'application "Tech Services". Voici une stratégie de test, des scénarios de test, et des critères de validation pour assurer le bon fonctionnement de l'application.

8.1 Stratégie de Test

La stratégie de test vise à couvrir toutes les fonctionnalités critiques de l'application, en mettant l'accent sur la performance, la sécurité, et la convivialité. Elle comprend les types de tests suivants :

8.1.1 Tests Unitaires :

Vérifier la fonctionnalité de chaque composant individuel, y compris le frontend, le backend, et les API.

Utilisation de frameworks de test appropriés (par exemple, Jest pour le frontend, Mocha/Chai pour le backend).

8.1.2 Tests d'Intégration :

Valider les interactions entre les composants pour garantir la cohérence du système dans son ensemble.

S'assurer que les différentes parties du système communiquent correctement.

8.1.3 Tests de Système :

Évaluer le système dans son ensemble pour vérifier son comportement dans des conditions réelles.

Tester l'application dans un environnement similaire à celui de la production.

8.1.4 Tests de Performance :

Évaluer la réactivité et la stabilité de l'application sous des charges de travail variables.

Identifier les goulots d'étranglement et optimiser les performances.

8.1.5 Tests de Sécurité :

Effectuer des tests de pénétration pour identifier les vulnérabilités potentielles.

Assurer la conformité aux normes de sécurité, notamment PCI-DSS pour les transactions financières.

8.1.6 Tests de Convivialité (UI/UX) :

Évaluer l'interface utilisateur pour garantir une expérience utilisateur conviviale.

Tester la navigation, la clarté des informations, et l'ergonomie.

8.2 Scénarios de Test

8.2.1 Scénarios de Test pour les Clients :

Inscription et Connexion

Recherche de Prestataires

Soumission d'une Demande de Service

Suivi des Interventions en Cours

Paiement et Facturation

Notation et Commentaires

8.2.2 Scénarios de Test pour les Prestataires :

Inscription et Connexion

Acceptation/Refus des Demandes de Service

Diagnostic et Rédaction de Devis

Validation et Début des Travaux

Gestion des Interventions sur le Terrain

Paiement des Prestataires

8.2.3 Scénarios de Test pour les Administrateurs :

Gestion des Utilisateurs (Clients, Prestataires, Admin-Entreprises)

Suivi des Activités et Rapports

Gestion des Services et des Profils d'Entreprises

Gestion des Transactions et des Paiements

8.3 Critères de Validation

8.3.1 Critères de Validation Généraux :

Aucun bug critique dans les fonctionnalités principales.

Temps de réponse inférieur à 2 secondes dans des conditions normales d'utilisation.

Temps de disponibilité de l'application de 99,9% ou plus.

8.3.2 Critères de Validation Spécifiques :

Les clients peuvent soumettre des demandes de service avec succès.

Les prestataires peuvent accepter, refuser, et compléter les travaux avec succès.

Les paiements sont traités de manière sécurisée et sans erreurs.

Les notations et commentaires des clients sont enregistrés correctement.

Cette stratégie de test, combinée avec des scénarios de test spécifiques et des critères de validation, garantit que l'application "Tech Services" est robuste, performante, sécurisée, et offre une expérience utilisateur de haute qualité.

9. Gestion des Erreurs - Tech Services

La gestion des erreurs est cruciale pour assurer la stabilité et la fiabilité de l'application "Tech Services". Elle comprend la gestion appropriée des erreurs côté client et côté serveur, ainsi que la mise en place de mécanismes de surveillance pour détecter et résoudre rapidement les problèmes.

9.1 Gestion des Erreurs et des Exceptions

9.1.1 Côté Client :

Affichage d'un message d'erreur clair et convivial lorsqu'une action échoue côté client.

Journalisation des erreurs côté client pour une analyse ultérieure.

Mise en place d'une page d'erreur générale pour les erreurs inattendues.

9.1.2 Côté Serveur (Backend) :

Retour de codes d'état HTTP appropriés pour indiquer le résultat de l'action (200 OK, 404 Not Found, 500 Internal Server Error, etc.).

Journalisation détaillée des erreurs côté serveur, y compris les informations sur la requête et les données pertinentes.

Mise en œuvre de mécanismes de reprise après échec pour les opérations critiques.

9.2 Mécanismes de Surveillance

9.2.1 Journalisation Centralisée :

Centralisation des journaux d'erreur dans un système de journalisation centralisé (par exemple, ELK Stack) pour une analyse approfondie.

Analyse des journaux pour détecter les tendances d'erreur et les problèmes récurrents.

9.2.2 Alertes en Temps Réel :

Mise en place d'alertes en temps réel pour les erreurs critiques via des outils de surveillance comme Prometheus ou Nagios.

Notification immédiate des équipes de développement en cas d'incident majeur.

9.2.3 Surveillance de la Performance :

Utilisation d'outils de surveillance de la performance pour détecter les goulots d'étranglement et optimiser les performances du système.

Surveiller les temps de réponse, la charge du serveur, et d'autres métriques clés.

9.2.4 Suivi des Transactions :

Mise en place d'un suivi des transactions pour suivre le parcours d'une requête à travers le système.

Identifie les points de défaillance et facilite le débogage.

9.2.5 Rapports d'Erreur Automatiques :

Envoi automatique de rapports d'erreur détaillés aux équipes de développement pour une analyse proactive.

Utilisation de services comme Sentry ou Rollbar pour la gestion des erreurs.

9.3 Plans de Reprise Après Échec (PRAE)

9.3.1 Sauvegarde et Restauration :

Mise en place de procédures de sauvegarde régulières pour les bases de données et les données critiques.

Capacité à restaurer rapidement le système en cas de défaillance.

9.3.2 Redondance des Serveurs :

Utilisation de redondance des serveurs pour minimiser les temps d'arrêt en cas de défaillance matérielle.

Mise en œuvre de mécanismes de basculement automatique.

9.3.3 Tests Réguliers de PRAE :

Exécution régulière de tests pour garantir l'efficacité des mécanismes de récupération.

La gestion des erreurs et des exceptions, combinée à des mécanismes de surveillance robustes, permet d'assurer une réactivité rapide aux problèmes potentiels et d'améliorer la stabilité globale de l'application "Tech Services".