

REPUBLIQUE DU CAMEROUN

Paix – Travail – Patrie

\*\*\*\*\*

MINISTERE DE  
L'ENSEIGNEMENT SUPERIEUR

\*\*\*\*\*

ECOLE NATIONALE  
SUPERIEURE POLYTECHNIQUE DE  
DOUALA



REPUBLIC OF CAMEROON

Peace – Work – Fatherland

\*\*\*\*\*

MINISTRY OF HIGHER  
EDUCATION

\*\*\*\*\*

NATIONAL ADVANCED  
SCHOOL OF ENGINEERING OF DOUALA



# INGENIEURIE ET CONCEPTION LOGICIELLE

CAHIER DE CHARGE POUR APPEL D'OFFRE A LA  
REALISATION D'UNE APPLICATION POUR LA MODELISATION DES  
RISQUES DE CREDIT : CAS D'UNE BANQUE

**Département :** Génie informatique et  
télécommunications

**Axe :** Génie Logiciel – Groupe 14

Rédigé et présenté par :

- JEUDIEU TSAGUE AUDREY DANIELLE 22G00167
- FEUPA TCHAKOUNTE PIERRE NATHAN 22G00131
- KITIO MATCHOUSSA GERBEAU 22G00196
- MPACKO DARREN NGOLE 24G01115
- AKOUDJOU MBA NOËLLY CHRISTELA 22G00007

Sous la supervision de  
Dr. IHONOCK

Année académique  
2025-2026

## **Table des matières**

Table des matières .....	ii
Liste des tableaux .....	iv
I- BUT OU OBJECTIF DU PROJET.....	1
1- Objectif général .....	1
2- Objectifs spécifiques.....	1
a) Objectifs Stratégiques (Impact sur l'organisation) .....	2
b) Objectifs Opérationnels (Efficacité métier) .....	2
c) Objectifs de Performance (Qualité technique).....	2
II- PRESENTATION DE L'EXISTANT .....	3
1- Caractéristiques du processus actuel .....	3
2- Conséquences du processus actuel .....	4
3- Conclusion de l'analyse et corrélation avec la solution proposée.....	4
III- SPÉCIFICATIONS TECHNIQUES FONCTIONNELLES .....	4
1- Utilisateurs et rôles .....	5
2- Établissement des fonctionnalités par utilisateur.....	5
a) Fonctionnalités de l'Agent de risque (Interface de Saisie).....	5
b) Fonctionnalités du Gestionnaire de risque (Analyse et Décision) .....	6
c) Fonctionnalités du Client (Consultation assistée).....	6
d) Fonctionnalités de l'Administrateur (Gestion Système).....	6
e) Fonctionnalités Communes (Système) .....	6
IV- SPÉCIFICATIONS TECHNIQUES NON FONCTIONNELLES .....	7
3- Fiabilité et Précision (Le cœur du métier) .....	7
4- Performance et Fluidité .....	7
5- Sécurité et Confidentialité des données .....	7
6- Ergonomie, Esthétique et UX (Expérience Utilisateur) .....	8
7- Évolutivité et Maintenance .....	8
V- PARTIE MATÉRIELLE OU LOGICIELLE / SYSTÈMES ET MATÉRIELS .....	8
1- Environnement Logiciel (Stack Technologique).....	9
2- Environnement de développement (Outils) .....	9
3- Ressources Matérielles (Hardware).....	9

VI- TYPES DE MAINTENANCE.....	10
1- Maintenance Corrective (Réparation) .....	10
2- Maintenance Évolutive (Amélioration) .....	10
3- Maintenance Préventive et Adaptative (Anticipation) .....	11
VII- BUDGET .....	11
1- Coûts des Ressources Humaines .....	11
2- Coûts Logiciels et Matériels .....	12
3- Récapitulatif Budgétaire .....	13
VIII- RÉFÉRENT.....	13
1- Référentiels Théoriques et Métier .....	13
2- Référentiels Techniques (Documentation) .....	14
3- Référentiels Normatifs et de Sécurité .....	14
4- Sources d'Inspiration Logicielle .....	14
IX- PARTIE JURIDIQUE.....	15
1- Protection des Données à Caractère Personnel.....	15
2- Responsabilité de la Décision Automatisée.....	15
▪ Propriété Intellectuelle .....	15
▪ Sécurité et Conservation.....	16
▪ Conformité aux Normes Bancaires Internationales .....	16
3- Éthique et Transparence (Algorithmic Accountability) .....	17

## **Liste des tableaux**

<b>Tableau 1: Budget ressources humaines .....</b>	<b>12</b>
<b>Tableau 2: Récapitulatif budgétaire .....</b>	<b>13</b>

# **CAHIER DE CHARGE POUR APPEL D'OFFRE A LA REALISATION D'UNE APPLICATION POUR LA MODELISATION DES RISQUES DE CREDIT : CAS D'UNE BANQUE**

## **I- BUT OU OBJECTIF DU PROJET**

Dans un environnement financier de plus en plus volatil, la gestion du risque de crédit est devenue le pilier central de la stabilité des institutions bancaires. Traditionnellement, l'évaluation des demandes de prêt reposait sur des analyses manuelles et des jugements d'experts qui, bien que précieux, comportent des limites intrinsèques : lenteur de traitement, manque d'uniformité dans les décisions et vulnérabilité face aux erreurs humaines. L'augmentation constante du volume de données disponibles et la nécessité de réduire drastiquement le taux de créances douteuses (prêts non remboursés) imposent aujourd'hui la transition vers des outils numériques intelligents. C'est dans ce contexte que s'inscrit le présent projet, visant à concevoir une application de modélisation du risque capable de transformer des données brutes en indicateurs décisionnels fiables et instantanés.

### **1- Objectif général**

L'objectif principal de ce projet est de concevoir et de déployer une solution logicielle d'aide à la décision basée sur le « Credit Scoring ». Cette application doit permettre d'évaluer avec précision la solvabilité des clients dès la phase d'instruction, afin de sécuriser les actifs de la banque en minimisant les pertes liées aux défauts de paiement.

### **2- Objectifs spécifiques**

L'atteinte des objectifs de ce projet repose sur une approche multidimensionnelle. Pour garantir la pertinence de la solution, les objectifs spécifiques ont été déclinés en trois (03) axes stratégiques : la performance économique de l'institution, l'optimisation des processus opérationnels et l'excellence technique du dispositif logiciel.

a) **Objectifs Stratégiques** (Impact sur l'organisation)

Il s'agit pour nous ici de :

- **Réduire le coût du risque** : Diminuer significativement le volume des créances douteuses en identifiant et en écartant les profils à forte probabilité de défaut (PD) avant l'engagement des fonds.
- **Standardiser la politique de crédit** : Garantir une équité de traitement des dossiers en appliquant des règles de décision uniformes sur l'ensemble du réseau d'agences, éliminant ainsi les biais de jugement personnels.
- **Améliorer la rentabilité** : Optimiser le portefeuille de prêts en favorisant l'octroi de crédits aux clients présentant un profil de risque maîtrisé.

b) **Objectifs Opérationnels** (Efficacité métier)

Il s'agit pour nous ici de :

- **Automatiser le flux de décision** : Digitaliser le parcours de saisie et d'analyse des données pour transformer une procédure souvent administrative en un processus fluide et rapide.
- **Aider à une prise de décision structurée** : Fournir aux agents une recommandation claire (Acceptation, Étude approfondie, Refus) basée sur un moteur de calcul robuste, tout en permettant une escalade vers les superviseurs pour les dossiers en zone grise.
- **Permettre une justification du risque** : Offrir une transparence sur le score obtenu en listant les facteurs de risque dominants (ex: ratio d'endettement, instabilité professionnelle), facilitant ainsi le dialogue entre l'agent et le client.

c) **Objectifs de Performance** (Qualité technique)

Il s'agit pour nous ici de :

- **Faire une précision prédictive** : Développer un algorithme de scoring capable de classer correctement les clients avec un taux de fiabilité élevé, limitant les "faux négatifs" (clients risqués acceptés par erreur).

- **Assurer la réactivité du système** : Assurer un temps de réponse quasi instantané lors du calcul du score, permettant une expérience fluide en agence lors de l'entretien avec le client.
- **Assurer l'intégrité et la sécurité des données** : Garantir que les informations financières et personnelles recueillies sont stockées et traitées selon les normes de confidentialité bancaire en vigueur.

## **II- PRESENTATION DE L'EXISTANT**

Avant de définir les spécifications (fonctionnelles et non fonctionnelles) de notre solution, il est impératif d'analyser les mécanismes de gestion du crédit actuellement en vigueur. Cette analyse de l'existant permet d'identifier les points de friction, les inefficacités et les risques auxquels l'institution est exposée. En comprenant comment les décisions sont prises aujourd'hui, nous pouvons mieux mesurer l'écart technologique à combler pour sécuriser le portefeuille de la banque.

### **1- Caractéristiques du processus actuel**

Le mode de fonctionnement actuel repose sur une **approche conventionnelle et largement manuelle**, dont les piliers sont :

- **L'Analyse Fondée sur l'Expertise Humaine** : La décision d'octroi repose essentiellement sur le jugement de l'agent de crédit. L'évaluation est qualitative, basée sur la lecture des pièces justificatives et l'entretien avec le client.
- **Utilisation d'Outils Bureautiques Statiques** : Le traitement des données financières (calcul du ratio d'endettement, reste à vivre) s'effectue via des feuilles de calcul type Excel ou des outils de saisie déconnectés de tout moteur prédictif.
- **Critères de Sélection Rigides** : La banque applique des filtres éliminatoires simples (ex : "être en CDI", "avoir moins de 33% d'endettement") sans croiser ces variables entre elles pour détecter des profils atypiques mais solvables.
- **Processus Séquentiel et Lent** : Le dossier doit physiquement ou numériquement circuler entre plusieurs services pour validation, ce qui allonge les délais de réponse.

## **2- Conséquences du processus actuel**

Le maintien de ce système "artisanal" face à une demande de crédit croissante entraîne des impacts négatifs majeurs :

- **Augmentation des créances douteuses** : L'absence d'un modèle statistique laisse passer des profils dont la fragilité n'est pas détectable à l'œil nu, augmentant ainsi le taux d'impayés.
- **Hétérogénéité des décisions** : Pour deux dossiers identiques, deux agents différents peuvent prendre des décisions opposées, créant un manque de cohérence institutionnelle.
- **Délais de traitement excessifs** : La lourdeur du processus manuel nuit à la compétitivité de la banque face à des concurrents plus réactifs.
- **Difficulté d'audit** : En l'absence d'une trace numérique claire du "pourquoi" de chaque décision, il est complexe pour les régulateurs de vérifier l'application de la politique de risque.

## **3- Conclusion de l'analyse et corrélation avec la solution proposée**

L'analyse de cet existant met en évidence une rupture entre les besoins de sécurité financière de la banque et les outils archaïques actuellement utilisés. Le constat est sans appel : la subjectivité humaine et la lenteur des outils statiques sont les causes directes de l'augmentation des créances douteuses.

C'est précisément pour corriger ces failles que la mise sur pied de notre solution logicielle devient une nécessité stratégique. Notre application ne se contentera pas de digitaliser la saisie des données ; elle viendra substituer l'analyse empirique par une modélisation mathématique du risque (Scoring). En automatisant le calcul de la probabilité de défaut et en uniformisant les critères de décision, le futur logiciel permettra de verrouiller le processus d'octroi, garantissant ainsi une réduction drastique et mesurable des impayés dès sa mise en exploitation.

## **III- SPÉCIFICATIONS TECHNIQUES FONCTIONNELLES**

L'architecture fonctionnelle de l'application est conçue pour transformer un processus manuel en un flux de travail numérique rigoureux. En répartissant les rôles entre le front-office,



l'analyse des risques et l'administration technique, le système garantit une traçabilité totale et une réduction efficace des créances douteuses.

## **1- Utilisateurs et rôles**

Le système distingue désormais quatre profils d'utilisateurs :

- Le **Client** : Fournisseur des données sources, il interagit avec le système via l'agent pour obtenir sa simulation.
- L'**Agent de risque** : Utilisateur opérationnel en agence, chargé de la collecte, de la saisie et de l'initialisation du scoring.
- Le **Gestionnaire de risque** : Expert décisionnaire qui analyse les scores, traite les dossiers complexes et valide l'octroi final.
- L'**Administrateur** : Responsable technique garant de la sécurité, des accès et de la maintenance du moteur de calcul.

## **2- Établissement des fonctionnalités par utilisateur**

### **a) Fonctionnalités de l'Agent de risque (Interface de Saisie)**

- **Authentification** : Connexion sécurisée avec gestion des sessions.
- **Formulaire de Saisie Dynamique** : Enregistrement des variables clés :
- **Informations personnelles** : Âge, situation matrimoniale, enfants à charge, statut résidentiel.
- **Informations professionnelles** : Type de contrat (CDI/CDD), ancienneté, secteur.
- **Données financières** : Revenus nets, charges fixes, crédits en cours.
- Le **Projet** : Montant demandé, durée, apport personnel.
- **Vérification de Cohérence** : Contrôle en temps réel des données saisies (ex: interdiction de montants négatifs).
- **Lancement du Scoring** : Déclenchement automatique de l'algorithme pour obtenir une tendance immédiate.
- **Édition de Simulation** : Génération d'un récapitulatif PDF pour le client.

**b) Fonctionnalités du Gestionnaire de risque (Analyse et Décision)**

- **Visualisation du Verdict** : Affichage du score (0 à 100) et du code couleur (Vert, Orange, Rouge).
- **Justification du Score (Explicabilité)** : Affichage des principaux facteurs ayant influencé le résultat (ex: "Capacité de remboursement insuffisante").
- **Arbitrage des "Zones Oranges"** : Possibilité d'étudier manuellement les dossiers ambigus et de forcer une décision (Approbation/Refus) avec commentaire obligatoire.
- **Dashboard Risque** : Vue d'ensemble sur le volume de crédits et le taux de risque moyen du portefeuille.
- **Historisation** : Consultation des anciennes demandes pour détecter d'éventuelles répétitions suspectes.

**c) Fonctionnalités du Client (Consultation assistée)**

- **Soumission des données** : Transmission des informations via l'interface gérée par l'agent.
- **Réception de la synthèse** : Consultation des conditions du crédit simulé.
- **Droit à l'information** : Accès aux motifs simplifiés de la recommandation du système.

**d) Fonctionnalités de l'Administrateur (Gestion Système)**

- **Gestion des Habilitations** : Création, modification et révocation des comptes (Agent, Gestionnaire).
- **Paramétrage des Seuils** : Mise à jour des limites de score pour les zones de décision.
- **Journal d'Audit (Logs)** : Traçabilité complète de "qui a fait quoi et quand" pour prévenir la fraude interne.
- **Extraction de données** : Génération de rapports statistiques pour la direction.

**e) Fonctionnalités Communes (Système)**

- **Persistance** : Sauvegarde automatique dans une base de données sécurisée.
- **Gestion des erreurs** : Messages d'alerte en cas de données manquantes ou de panne du moteur de calcul.

Cette organisation fonctionnelle assure une séparation stricte des pouvoirs. L'agent se concentre sur la qualité des données collectées, tandis que le gestionnaire de risque se concentre sur l'analyse de la probabilité de défaut. Ce flux de travail automatisé est la pierre angulaire de notre solution pour éradiquer les décisions d'octroi subjectives.

## **IV- SPÉCIFICATIONS TECHNIQUES NON FONCTIONNELLES**

Les spécifications non fonctionnelles définissent les contraintes de qualité et de comportement du système. Elles ne décrivent pas ce que l'application "fait", mais la manière dont elle doit le faire pour être jugée performante, sûre et adoptée par les utilisateurs finaux.

### **3- Fiabilité et Précision (Le cœur du métier)**

- **Précision du Modèle (Accuracy) :** Le moteur de scoring doit atteindre un taux de précision minimal (ex: 80%) sur les données de test. Le taux de "faux négatifs" (clients insolvable classés comme bons) doit être minimisé de manière critique.
- **Disponibilité :** L'application doit être opérationnelle à 99,9% pendant les heures d'ouverture des agences (9h00 - 18h00), afin de ne jamais bloquer un entretien client.

### **4- Performance et Fluidité**

- **Temps de réponse (Latence) :** Le calcul du score, après validation du formulaire, doit être quasi-instantané (moins de 3 secondes).
- **Fluidité de navigation :** Le passage d'une étape du formulaire à une autre doit se faire sans rechargement lourd de la page, assurant une transition fluide pendant que l'agent discute avec le client.

### **5- Sécurité et Confidentialité des données**

- **Confidentialité :** Les données financières sensibles (revenus, soldes bancaires) doivent être chiffrées dans la base de données. Seuls les utilisateurs dûment authentifiés peuvent y accéder.
- **Intégrité :** Le système doit garantir que les données saisies ne peuvent pas être modifiées par un tiers une fois le score généré (horodatage et gel des données).

- **Auditabilité** : Chaque action (consultation, modification, suppression) doit être enregistrée dans un journal de bord (log) pour permettre une traçabilité totale en cas de contrôle interne.

## **6- Ergonomie, Esthétique et UX (Expérience Utilisateur)**

- **Ergonomie** : L'interface doit être conçue pour limiter le nombre de clics. La saisie doit être intuitive (menus déroulants, auto complétion) pour éviter les erreurs de frappe.
- **Esthétique** : L'interface graphique doit respecter la charte visuelle de la banque (couleurs, logos, polices) pour inspirer confiance au client qui observe l'écran avec l'agent.
- **Accessibilité** : Le design doit être lisible (contraste élevé, taille de police adaptable) pour s'adapter aux différents environnements de travail en agence.

## **7- Évolutivité et Maintenance**

- **Extensibilité** : Le code doit être structuré de manière modulaire (architecture en couches) pour permettre l'ajout futur de nouvelles variables de scoring ou de nouveaux produits financiers sans refondre l'intégralité du système.
- **Interopérabilité** : Bien que l'application soit autonome dans un premier temps, elle doit être conçue pour pouvoir s'interfacer ultérieurement avec le Core Banking System (CBS) de la banque.

Ces spécifications non fonctionnelles garantissent que la solution logicielle est non seulement un outil de calcul puissant, mais aussi une application robuste et professionnelle. En assurant une sécurité maximale et une précision élevée, nous créons un environnement de confiance indispensable à la réduction des créances douteuses.

## **V- PARTIE MATÉRIELLE OU LOGICIELLE / SYSTÈMES ET MATÉRIELS**

Le choix de la pile technologique (Stack) s'est porté sur des technologies web modernes et performantes, permettant de garantir une interface réactive et une gestion des données locale, sans dépendance à un serveur back-end complexe pour cette phase du projet.

## 1- Environnement Logiciel (Stack Technologique)

L'application est développée comme une **Single Page Application (SPA)** robuste, utilisant les outils suivants :

- **Langage de programmation : TypeScript**
  - **Rôle** : Apporte un typage statique au JavaScript, garantissant la sécurité du code et réduisant les erreurs lors de la manipulation des données financières complexes.
- **Framework Frontend : React.js**
  - 1- **Rôle** : Permet de créer une interface utilisateur dynamique et modulaire grâce à l'architecture par composants (idéal pour les formulaires de saisie et les tableaux de bord).
- **Framework CSS : Tailwind CSS**
  - **Rôle** : Assure un design professionnel, moderne et entièrement "responsive" (adaptable) tout en accélérant le développement de l'interface graphique.
- **Bibliothèque de Visualisation : Recharts**
  - **Rôle** : Utilisée pour transformer les scores de risque en graphiques visuels (jauges, diagrammes) pour le Gestionnaire de risque.
- **Gestion des données : Local Storage (API Web)**
  - **Rôle** : Sert de base de données locale. Il permet de persister les dossiers clients et les historiques de scoring directement dans le navigateur de l'utilisateur sans nécessiter de base de données externe.

## 2- Environnement de développement (Outils)

- **IDE** : Visual Studio Code (avec extensions TypeScript et Tailwind).
- **Gestionnaire de paquets** : NPM ou Yarn pour la gestion des dépendances.
- **Navigateur de test** : Google Chrome / Firefox (outils de développement pour inspecter le Local Storage).

## 3- Ressources Matérielles (Hardware)

L'architecture étant entièrement exécutée côté client, les besoins matériels sont optimisés :

- **Postes de travail (Agents et Gestionnaires)** : \* Ordinateurs équipés d'un processeur standard et de 4 Go de RAM minimum.
- **Navigateur web récent** supportant l'API LocalStorage et ES6+.

- **Stockage :** Aucun serveur de base de données n'est requis à ce stade, les données étant stockées sur le disque local de la machine cliente via le navigateur.

L'utilisation de **TypeScript** combiné à **React** offre une base solide pour une application de scoring rapide et fluide. Le choix du **Local Storage** permet une démonstration autonome du concept (Proof of Concept) tout en simplifiant le déploiement technique au sein de l'agence bancaire.

Puisque nous n'utilisons pas de backend, l'algorithme de scoring est codé directement en **TypeScript** (côté client) ; le moteur de calcul est intégré au frontend.

## **VI- TYPES DE MAINTENANCE**

La mise en service de l'application de scoring n'est pas une fin en soi. Pour garantir la fiabilité des scores de crédit et la sécurité des données stockées, un plan de maintenance rigoureux est indispensable. Nous distinguons trois types de maintenance pour ce projet.

### **1- Maintenance Corrective (Réparation)**

La maintenance corrective intervient pour corriger les anomalies ou "bugs" détectés par les utilisateurs (Agents ou Gestionnaires de risque) après la mise en production.

- **Correction des bugs d'affichage :** Résolution des problèmes liés au rendu de l'interface avec Tailwind CSS sur différents navigateurs.
- **Rectification du moteur de calcul :** Correction d'éventuelles erreurs de logique dans l'algorithme de scoring développé en TypeScript.
- **Gestion des erreurs de stockage :** Résolution des problèmes liés à la saturation ou à la corruption des données dans le Local Storage.

### **2- Maintenance Évolutive (Amélioration)**

Compte tenu de l'évolution constante des politiques bancaires, l'application doit pouvoir s'adapter sans être totalement reconstruite.

- **Mise à jour des critères de risque :** Ajout de nouvelles variables de scoring (ex: prise en compte de l'épargne ou des cryptomonnaies).

- **Amélioration de la visualisation** : Ajout de nouveaux graphiques avec Recharts pour une analyse plus fine par le Gestionnaire de risque.
- **Transition vers un Backend** : À terme, prévoir l'évolution de la persistance des données du Local Storage vers une API sécurisée et une base de données centralisée.

### **3- Maintenance Préventive et Adaptative (Anticipation)**

Il s'agit de s'assurer que l'application reste compatible avec son environnement technique.

- **Veille technologique** : Mise à jour des dépendances (React, TypeScript) pour corriger les vulnérabilités de sécurité et améliorer les performances.
- **Optimisation du stockage** : Mise en place de scripts de nettoyage ou de compression des données pour éviter que le Local Storage n'atteigne sa limite technique (généralement 5 Mo).
- **Adaptation aux navigateurs** : S'assurer que les mises à jour des navigateurs (Chrome, Firefox, Edge) ne bloquent pas les fonctionnalités de l'application.

En structurant la maintenance autour de ces trois axes, la banque s'assure de disposer d'un outil de modélisation des risques pérenne. La maintenance évolutive est ici particulièrement stratégique, car elle permettra de transformer cette application « client-side » en un système d'information bancaire complet à l'avenir.

## **VII- BUDGET**

L'estimation budgétaire de ce projet couvre l'ensemble du cycle de vie du développement logiciel (SDLC), de l'analyse des besoins au déploiement final. Étant donné l'architecture "Client-Side" choisie, les coûts d'infrastructure sont minimisés au profit de l'expertise technique en développement et en modélisation de données.

### **1- Coûts des Ressources Humaines**

C'est le poste de dépense le plus important. Il correspond au temps passé par les différents profils sur le projet (estimé sur une durée de 3 mois).

**Tableau 1: Budget ressources humaines**

Profil	Rôle	Estimation (en unités monétaires)
<b>1 Chef de Projet</b>	Planification, Cahier des charges, Suivi	1 500 000
<b>1 Développeur Frontend (React/TS)</b>	Développement de l'interface et du moteur	2 500 000
<b>1 UI/UX Designer</b>	Design de l'interface et ergonomie	800 000
<b>1 Analyste Risque / Data Scientist</b>	Définition de l'algorithme de scoring	1 200 000
<b>TOTAL RH</b>		<b>6 000 000</b>

## **2- Coûts Logiciels et Matériels**

L'utilisation de technologies Open Source (React, Tailwind, TypeScript) permet de réduire les frais de licence.

- **Licences Logiciels : 0** (Logiciels libres / Open Source).
- **Outils de développement & IDE : 0** (VS Code, Git).
- **Postes de travail : Matériel** déjà existant au sein de la banque (coût d'acquisition non inclus).
- **Frais de déploiement initial : Configuration des navigateurs et formation des agents : 500 000.**



### 3- Récapitulatif Budgétaire

*Tableau 2: Récapitulatif budgétaire*

Poste de dépense	Montant
Ressources Humaines	6 000 000
Déploiement et Formation	500 000
Provision pour Maintenance (1 an)	1 000 000
<b>TOTAL GÉNÉRAL DU PROJET</b>	<b>7 500 000</b>

Ce budget reflète une solution **optimisée et économique**. En évitant les coûts de serveurs backend et de licences logicielles propriétaires, la banque bénéficie d'un outil de scoring performant avec un retour sur investissement (ROI) rapide, principalement grâce à la réduction immédiate des pertes liées aux créances douteuses.

## VIII- RÉFÉRENT

La conception de cette application de Credit Scoring s'appuie sur un socle de références théoriques, techniques et réglementaires. Ces référentiels garantissent que l'outil est aligné avec les standards actuels de l'ingénierie logicielle et les exigences du secteur bancaire.

### 1- Référentiels Théoriques et Métier

- **Gestion du Risque de Crédit :** Principes basés sur les accords de **Bâle II / Bâle III**, notamment concernant l'évaluation de la Probabilité de Défaut (PD) et la gestion des fonds propres.
- **Modélisation Statistique :** Méthodologies de scoring financier pour la classification des risques (Score Card).

## 2- Référentiels Techniques (Documentation)

Le développement de la solution repose sur les documentations officielles des technologies utilisées :

- **Documentation React.js** : Pour l'architecture des composants et la gestion du cycle de vie de l'interface.
- **TypeScript Handbook** : Pour la mise en œuvre du typage statique et la sécurisation du code métier.
- **Tailwind CSS Documentation** : Pour les standards de design "Utility-First" et la cohérence de l'interface utilisateur.
- **MDN Web Docs (Local Storage API)** : Pour les spécifications relatives au stockage persistant côté client.

## 3- Référentiels Normatifs et de Sécurité

- **ISO/IEC 25010** : Norme internationale pour l'évaluation de la qualité des produits logiciels (performance, utilisabilité, sécurité).
- **RGPD (ou équivalent local)** : Référentiel pour la protection des données à caractère personnel et la confidentialité des informations financières traitées dans l'application.

## 4- Sources d'Inspiration Logicielle

- **Fintech Dashboards** : Analyse des interfaces modernes de gestion bancaire (Revolut Business, Qonto) pour garantir une expérience utilisateur (UX) fluide et intuitive pour l'Agent de risque.

L'ensemble de ces référents permet de passer d'un simple outil de calcul à une application bancaire conforme aux exigences du marché. L'alignement sur ces standards assure la crédibilité du projet lors de sa présentation et facilite son intégration future dans un écosystème informatique plus vaste.

## **IX- PARTIE JURIDIQUE**

L'implémentation d'un outil de scoring implique la manipulation de données sensibles (revenus, identité, habitudes financières). Le projet doit donc se conformer au cadre légal en vigueur pour garantir la protection des droits des clients et la responsabilité de la banque.

### **1- Protection des Données à Caractère Personnel**

Bien que les données soient stockées via le **Local Storage**, elles restent sous la responsabilité juridique de l'institution.

- **Consentement éclairé** : Avant toute saisie par l'Agent de risque, le Client doit signer une clause autorisant le traitement automatisé de ses données pour l'évaluation de sa solvabilité.
- **Droit d'accès et de rectification** : Conformément à la loi (ex: RGPD ou lois nationales sur la cyber-sécurité), le client peut demander à consulter ou à faire rectifier les informations saisies dans son dossier.
- **Finalité des données** : Les données collectées ne doivent être utilisées que pour l'évaluation du risque de crédit et ne peuvent être cédées à des tiers sans accord explicite.

### **2- Responsabilité de la Décision Automatisée**

Le cadre juridique impose une transparence sur les algorithmes de décision.

- **Non-discrimination** : L'algorithme de scoring (en TypeScript) ne doit inclure aucun critère discriminatoire interdit par la loi (religion, origine, opinions politiques, etc.).
- **Droit à l'intervention humaine** : Le client a le droit de contester un refus automatique. C'est ici que le rôle du **Gestionnaire de risque** est juridiquement crucial : il assure la médiation humaine et peut réviser une décision prise par la machine.

#### **▪ Propriété Intellectuelle**

- **Propriété du Code Source** : Le code développé (React, TS) est la propriété exclusive de la banque (ou de l'étudiant dans le cadre du projet académique).

- **Utilisation de l'Open Source** : Le projet utilise des bibliothèques libres (MIT License pour React, Tailwind). Les termes de ces licences doivent être respectés, notamment le maintien des mentions de copyright des auteurs originaux.

▪ **Sécurité et Conservation**

- **Durée de conservation** : Les données dans le Local Storage doivent être purgées après une période définie (ex: 5 ans après la fin du crédit) conformément aux obligations légales de conservation des documents bancaires.
- **Confidentialité** : L'Agent et le Gestionnaire de risque sont soumis au secret professionnel bancaire concernant les informations auxquelles ils accèdent via l'interface.

Le respect de ce cadre juridique transforme l'application en un outil "éthique" et conforme. Cela protège la banque contre d'éventuels litiges judiciaires et renforce la confiance des clients envers le processus de digitalisation de leur banque.

▪ **Conformité aux Normes Bancaires Internationales**

L'application intègre les principes fondamentaux issus des instances de régulation financière pour garantir une gestion saine des fonds :

- **Accords de Bâle (Bâle II & III)** : Le moteur de scoring respecte la méthode **IRB (Internal Ratings-Based Approach)**. Cette norme encourage les banques à utiliser leurs propres modèles internes pour évaluer le risque de crédit.
  - Le calcul vise à estimer précisément la **PD (Probability of Default)**, permettant ainsi à la banque de calculer ses besoins en fonds propres réglementaires.
- **Normes IFRS 9 (International Financial Reporting Standards)** :
  - L'application participe à la mise en œuvre de la norme **IFRS 9**, qui impose aux banques de comptabiliser les "pertes de crédit attendues" (*Expected Credit Losses*) dès l'octroi du prêt, et non plus seulement au moment du défaut de paiement.
- **Dispositif KYC (Know Your Customer)** :
  - Le formulaire de saisie de l'Agent de risque est conçu pour respecter le protocole **KYC**. Il assure que l'identité, l'origine des revenus et la situation financière du client sont dûment documentées pour prévenir le blanchiment d'argent et le financement du terrorisme.

### 3- Éthique et Transparence (Algorithmic Accountability)

- **Droit à l'explication** : Conformément aux directives bancaires sur les décisions automatisées, l'application fournit des "Reason Codes" (codes de justification). Si un crédit est refusé ou noté "Orange", le système doit être capable d'expliquer quel facteur a pesé le plus lourd (ex: ratio d'endettement trop élevé).
- **Auditabilité du Modèle** : Le code TypeScript et les pondérations des critères doivent rester auditables par les contrôleurs internes de la banque ou les commissaires aux comptes pour vérifier l'absence de biais algorithmiques.

L'alignement sur les normes **Bâle III** et **IFRS 9**, couplé au respect du **RGPD** et du **KYC**, fait de cette application un outil de confiance. Cette double conformité (technique et réglementaire) assure que la solution logicielle contribue non seulement à la performance commerciale, mais aussi à la résilience et à la réputation de l'institution financière.