

# Cahier de charge du projet du module ERP

## Table des matières

I.	Introduction :	3
II.	Description du projet :	3
1.	Objectifs du projet :	3
2.	Présentation du projet :	3
3.	Expression des besoins :	3
a.	Besoins fonctionnels :	3
b.	Besoins non fonctionnels :	4
III.	Contraintes du projet :	4
1.	Contraintes en termes de délais :	4
2.	Contraintes de sécurité :	4
3.	Contraintes technique :	4
IV.	Déroulement du projet :	5
1.	Planification :	5
a.	Spécifications :	5
b.	Conception :	5
c.	Programmation :	5
d.	Tests unitaires :	5
e.	Intégration :	5
f.	Test global :	5
2.	Ressources :	5

# I. Introduction :

Dans le cadre du module ERP de la troisième année de cycle ingénieur de l'école nationale de sciences appliquées, il a été demandé de travailler sur un projet de développement d'une solution E-BANKING.

Ce document décrit le contexte, les besoins fonctionnels et les objectifs du projet. Un premier découpage des étapes nécessaires à la réalisation d'un tel projet donne lieu dans ce document à un planning prévisionnel. Ce document a pour finalités de définir le projet de manière simple et détaillée et de définir les objectifs auxquels devra répondre une future spécification technique.

## II. Description du projet :

### 1. Objectifs du projet :

L'objectif principal du projet est la création d'une application E-Banking qui permet d'effectuer les transactions bancaires en toute sécurité et rapidité.

Le projet a pour but de créer un lien solide entre l'agent de la banque et les clients. Grâce à cette solution, on facilitera les transactions bancaires en suivant un processus sécurisé et fiable. Et En garantissant à chaque type d'utilisateur un espace afin de bénéficier des fonctionnalités qui aura les privilèges à les exécuter.

### 2. Présentation du projet :

Le projet est né afin de simplifier la vie des clients de la banque et aider les banques à renforcer leur stratégie en respectant leurs spécificités et contraintes et exigences.

Ainsi, les objectifs principaux du projet sont :

- Création d'un espace d'administration afin de paramétrer les devises et les variables globales de la solution.
- Création d'un espace réservé à la banque afin de créer les enrôlements des clients et la gestion des abonnés.
- Création d'un espace réservé au client afin de :
  - Visualiser les comptes associés à son enrôlement
  - Réserver un espace pour effectuer leurs virements.
  - Réserver un espace pour effectuer les recharges téléphoniques.
  - Réserver un espace pour gestion de son compte.
  - Assurer une authentification sécurisée.
- Création d'un espace afin d'effectuer une vérification des transactions bancaires.

### 3. Expression des besoins

#### a. Besoins fonctionnels

Cette phase représente un point de vue « fonctionnel » de l'architecture système. Par le biais des cas d'utilisation, nous serons en contact permanent avec l'acteur du système en vue de définir les limites de celui-ci, et ainsi éviter de trop s'éloigner des besoins réels de l'utilisateur final.

Les principaux profils qui auront à utiliser sont les suivants :

- ❖ Client : Un utilisateur qui pourra y accéder à son compte où il aura la possibilité de visualiser ces Compte de facturation et son solde. Ainsi que d'effectuer des virements en saisissant le numéro de compte de bénéficiaires.
- ❖ Agent de la banque : Un utilisateur qui pourra y accéder à son propre espace afin d'enrôler des nouveaux clients ou bien de résilier un client déjà enrôlés, ainsi que de gérer les abonnées, il pourra soit activer ou suspendre ou lever la suspendu d'un client.
- ❖ Admin : Un utilisateur qui aura la possibilité de créer une nouvelle agence ainsi de gérer les agents. Ainsi que de contrôler les paramètres globales de l'application compris (Les précisions de solde, ainsi de contrôler les libellés des statuts des clients et leurs contrats ....).

#### *b. Besoins non fonctionnels*

Ce sont des exigences qui ne concernent pas spécifiquement le comportement du système mais plutôt identifient des contraintes internes et externes du système. Les principaux besoins non fonctionnels de notre application se résument dans les points suivants :

- ❖ Le code doit être clair pour permettre des futures évolutions ou améliorations. Et d'être paramétrable le plus possible afin d'assurer une simplicité lors changements sans être obliger de modifier le code source.
- ❖ L'ergonomie : l'application offre une interface conviviale et facile à utiliser.
- ❖ La sécurité : l'application doit respecter la confidentialité des données. Ainsi d'assurer une fiabilité lors de l'exécution des transactions bancaires...

### **III. Contraintes du projet :**

#### **1. Contraintes en termes de délais :**

A partir de la livraison du cahier des charges, il disposera d'environ deux mois pour la réalisation du projet.

#### **2. Contraintes de sécurité :**

La gestion de la sécurité est la principale contrainte de ce système. L'application doit posséder une gestion de privilèges et de niveaux d'accès pour les différents types d'utilisateurs (client, agent, ..) selon leur profil.

#### **3. Contraintes technique :**

Pour le développement de ce système, il disposera d'une architecture sur laquelle il devra être basé cette application. De plus, le développement devra suivre toutes les normes techniques pour une meilleure performance, maintenance et facilité de mise à jour

## IV. Déroulement du projet

### 1. Planification :

Une première analyse du projet nous a permis de définir une suite d'étapes à suivre pour sa réalisation.

#### a. Spécifications

Durant cette étape, il faudrait définir clairement ce qui doit être réalisé pour atteindre les objectifs du projet.

#### b. Conception

Cette phase permettra de faire le choix entre les différentes solutions techniques en respectant les contraintes du projet.

#### c. Programmation

Écriture du code des différents modules et leurs fonctionnalités.

#### d. Tests unitaires

Cette étape se chevauche avec l'étape de programmation, les tests unitaires devront être effectués pendant le développement pour s'assurer du bon fonctionnement de chaque module de l'application.

#### e. Intégration

Regroupement de l'ensemble des modules après leur validation par chaque développeur des composants qu'il a réalisés.

#### f. Test global

Cette partie permet de s'assurer que toutes les fonctionnalités de l'application sont valides, après l'intégration.

### 2. Ressources

Maintenant il faudrait décrire l'ensemble des ressources mises à disposition pour la réalisation du projet.

Ressources humaines : Le projet sera réalisé par un groupe de quatre étudiants. Il est nécessaire d'évaluer la charge de travail que chaque membre sera en mesure de fournir, notamment pour les tâches qui nécessitent une répartition : programmation et tests unitaires.

Ressources matérielles : L'application sera basée sur l'architecture backend et frontend. Cette architecture a une structure arborescente, ce qui nous permettra de séparer la partie métier (traitement de données)

- SpringMVC
- Spring-Data
- Spring-Security

Et de l'interface de notre système.

- Application client : une application distante en Angular.
- Application Banque : une application locale en Angular.
- Application Admin : une application locale en JSF ou Thymleaf.