

# Cahier des Charges Fonctionnel

**Simulation de Stabilité des Structures Civiles  
par Apprentissage Profond**

Projet académique – École Supérieure d'Ingénierie

## Informations du document

**Version :** 1.0

**Date :** 2 novembre 2025

**Statut :** Validé

**Confidentialité :** Confidentiel

# **Notice de confidentialité**

## **DOCUMENT CONFIDENTIEL**

Ce document contient des informations confidentielles  
et propriétaires.

Il est strictement interdit de le diffuser sans  
autorisation écrite.

Maître d'ouvrage : École Supérieure d'Ingénierie

© 2025 – Tous droits réservés

## **Historique des versions**

<b>Version</b>	<b>Date</b>	<b>Auteur</b>	<b>Commentaires</b>
1.0	02/11/2025	Équipe Projet	Version initiale validée

# Table des matières

<b>1 Présentation générale du projet</b>	<b>5</b>
1.1 Contexte . . . . .	5
1.2 Objectifs du projet . . . . .	5
1.3 Pérимètre fonctionnel . . . . .	5
<b>2 Architecture et interfaces</b>	<b>6</b>
2.1 Architecture générale . . . . .	6
2.2 Interfaces utilisateur (Flutter) . . . . .	6
2.3 Interfaces backend (Spring Boot) . . . . .	6
<b>3 Modèle de données</b>	<b>7</b>
3.1 Schéma relationnel . . . . .	7
3.2 Technologies . . . . .	7
<b>4 Plan de développement</b>	<b>8</b>
4.1 Méthodologie . . . . .	8
4.2 Phases et planning . . . . .	8
<b>5 Gestion des risques</b>	<b>10</b>
<b>6 Critères d'acceptation</b>	<b>11</b>
6.1 Fonctionnels . . . . .	11
<b>7 Maintenance et évolution</b>	<b>12</b>
7.1 Types de maintenance . . . . .	12
7.2 Évolutions prévues . . . . .	12
<b>8 Aspects juridiques et RGPD</b>	<b>13</b>
<b>9 Budget prévisionnel</b>	<b>14</b>
<b>10 Livrables</b>	<b>15</b>
<b>11 Validation finale</b>	<b>16</b>

# 1 Présentation générale du projet

## 1.1 Contexte

Les structures civiles (bâtiments, ponts, poutres, etc.) nécessitent une évaluation précise de leur stabilité pour garantir la sécurité des occupants et la durabilité des ouvrages. Les méthodes classiques, souvent basées sur la méthode des éléments finis (FEM), sont coûteuses en temps et en ressources.

Ce projet vise à développer une application mobile intelligente permettant de simuler la stabilité d'une structure civile via un modèle d'apprentissage profond, accessible et performant.

## 1.2 Objectifs du projet

- Fournir un outil mobile intuitif pour ingénieurs et étudiants en génie civil.
- Réduire le temps de simulation par rapport aux méthodes FEM traditionnelles.
- Intégrer un modèle IA capable de prédire la stabilité et les contraintes principales.
- Offrir une visualisation 3D interactive des résultats.

## 1.3 Périmètre fonctionnel

Le système permettra :

- L'authentification sécurisée des utilisateurs.
- La saisie guidée des paramètres structurels.
- La simulation IA côté serveur via API REST.
- L'affichage des résultats : stabilité, contraintes, déformations.
- La génération d'un rapport PDF exportable.

## 2 Architecture et interfaces

### 2.1 Architecture générale

Architecture 3-Tiers :

- **Frontend mobile** : Flutter (UI/UX multiplateforme Android/iOS)
- **Backend** : Spring Boot (API REST sécurisée)
- **IA Engine** : Modèle Deep Learning (PyTorch/TensorFlow, intégré via service Python)
- **Base de données** : PostgreSQL (hébergement cloud)

### 2.2 Interfaces utilisateur (Flutter)

- Écran de connexion / inscription / biométrie
- Tableau de bord (Dashboard) avec statistiques et accès rapide
- Formulaire de paramètres de simulation (multi-step wizard)
- Résultats et visualisation 3D interactive
- Historique et gestion de simulations enregistrées
- Profil utilisateur et paramètres d'application

### 2.3 Interfaces backend (Spring Boot)

- Authentification JWT
- Gestion utilisateurs
- Simulation IA (création, exécution, suivi, export PDF)
- API d'administration et statistiques système

## **3 Modèle de données**

### **3.1 Schéma relationnel**

- users (gestion des comptes)
- simulations (métadonnées des simulations)
- structure\_parameters (paramètres géométriques, matériaux, charges)
- prediction\_results (résultats IA : contraintes, déformations, verdicts)
- audit\_logs (journalisation des actions et sécurité)

### **3.2 Technologies**

- PostgreSQL
- ORM : Hibernate / JPA
- Sauvegardes quotidiennes automatisées

## 4 Plan de développement

### 4.1 Méthodologie

Approche Agile SCRUM :

- Sprints de 2 semaines
- Daily meetings
- Backlog produit priorisé
- Livraison incrémentielle

### 4.2 Phases et planning

Phase	Durée
Description	
Initialisation	2 sem.
Setup, maquettes, CI/CD	
Backend Core	4 sem.
API Auth, CRUD	
Simulation	
Modèle IA	6 sem.
Dataset, entraînement, validation	
Intégration IA	2 sem.
Service d'inférence	
Mobile Flutter	6 sem.
UI/UX, intégration API	
Visualisation 3D	2 sem.
Rendu interactif	
Tests et Qualité	3 sem.
Tests unitaires, intégration, UAT	
Documentation &	2 sem.
Déploiement	

Livraison finale

---

**Durée totale estimée :** 27 semaines ( $\approx$  6,5 mois)

## 5 Gestion des risques

ID	Risque	Proba.	Impact
<b>Mesures de mitigation</b>			
R01	Modèle IA imprécis Enrichir dataset, tuning hyper-paramètres	Moyen	Élevé
R02	Temps de réponse lent Optimisation IA + GPU inference	Moyen	Élevé
R03	Dépassement planning Révision planning, buffer 20%	Élevé	Moyen
R04	Failles de sécurité Audit sécurité, tests d'intrusion	Moyen	Critique
R05	Bugs critiques Tests exhaustifs + rollback plan	Faible	Élevé

# 6 Critères d'acceptation

## 6.1 Fonctionnels

<b>Code Seuil</b>	<b>Critère</b>	<b>Description</b>
CA-01 OK	Authentification	Connexion et biométrie fonctionnelles
CA-02 OK	Simulation	Résultats complets < 3s
CA-03 OK	IA Accuracy	$\geq 95\%$ sur test set
CA-04 OK	Performance	100 users simultanés sans latence
CA-05 OK	Sécurité	0 vulnérabilité critique
CA-06 OK	Code qualité	80% couverture tests

## 7 Maintenance et évolution

### 7.1 Types de maintenance

- **Maintenance corrective** : correction rapide des anomalies
- **Maintenance préventive** : mises à jour dépendances, audit sécurité
- **Maintenance évolutive** : ajout nouvelles structures et optimisation IA

### 7.2 Évolutions prévues

- **V2.0** : Structures complexes, collaboration, CAO
- **V3.0** : Analyse dynamique et IA générative

## **8 Aspects juridiques et RGPD**

- Consentement explicite et droit à l'oubli (RGPD)
- Données chiffrées (SSL, AES-256)
- Clause de non-responsabilité : outil d'aide à la décision
- Code et modèle IA : propriété du maître d'ouvrage

## 9 Budget prévisionnel

Catégorie	Montant (€)
Ressources Humaines	256 500
Infrastructure & Outils	20 824
Validation & Formation	33 000
<b>Total</b>	<b>310 324</b>
Imprévus (15%)	46 549
<b>rowgrayBudget global</b>	<b>≈ 357 000</b>

## **10 Livrables**

- Code source complet : Flutter, Spring Boot, IA (Python)
- Artefacts déployables : Docker, APK, modèle IA exporté
- Documentation :
  - Technique (DAT, Swagger, UML)
  - Fonctionnelle (guides utilisateurs, maquettes)
  - IA (architecture, dataset, métriques)
  - Qualité (plan de test, rapports, audits)

## **11 Validation finale**

Le présent Cahier des Charges Fonctionnel est approuvé par :

---

Chef de projet

---

Responsable technique

---

Responsable qualité

---

Maître d'ouvrage