

# Cahier des charges — Evalya Smart

Plateforme Web (React) + Mobile (React Native) de gestion intelligente des devoirs, examens et suivi pédagogique

## 1. Objectif

Concevoir en **6 semaines** une solution unifiée et intelligente permettant :

1. **Planifier, distribuer et collecter** numériquement les devoirs ;
2. **Organiser** tout le cycle des examens : calendrier, convocations, dépôt/scan, notation ;
3. **Automatiser** correction & analyse via IA (LLM + OCR) ;
4. Fournir des **tableaux de bord temps-réel** et des alertes précoces à l'ensemble des acteurs ;
5. Offrir une expérience **Web (direction/enseignants)** et **mobile iOS + Android (élèves, parents, enseignants)** réalisée en **React Native**.

## 2. Périmètre fonctionnel

### 2.1 Devoirs

Fonction	Exigences
Planification	Création rapide (classe, matière, date), duplication, brouillons.
Notifications	Web-push, e-mail, push mobile.
Soumission élève	PDF, DOCX, images, audio, éditeur riche ; horodatage.
Rappels automatisés	Relance J-1 et H-1 paramétrables.
Correction	Annotations manuelles + correction IA (QCM, réponses courtes, rédaction).
Feedback	Note, commentaire global & in-line, suggestions IA.
Historique	Versioning, export PDF, recherche full-text.

### 2.2 Examens

Fonction	Exigences
Calendrier central	Détection de conflits ; multi-matières ; vue rôle-dépendante.
Convocations	PDF + QR-Code sécurisé.
Téléversement copies	Scan multi-pages, association via QR, OCR auto.
Correction IA	QCM auto-notés, propositions IA questions ouvertes.
PV & archive	Procès-verbal numérique, signature électronique, stockage WORM.

### 2.3 Intelligence Artificielle

Module	Description
Correction IA	LLM (GPT-compat.) + règles barème.
OCR manuscrit	PaddleOCR + post-traitement NLP.
Générateur exercices	Prompt → QCM / texte à trous / problème guidé.
Analyse prédictive	LightGBM : risque d'échec, alertes.
Rapport hebdo parents	Résumé naturel + graphes tendances.

## 2.4 Tableaux de bord

- **Enseignant** : retards, moyenne, progression élève.
- **Direction** : comparaison classes/matières, heat-map.
- **Parent** : progression simplifiée, alertes IA.

## 3. Architecture technique

Couche	Technologie
Frontend Web	<b>React 18 + TypeScript + Vite + Tailwind CSS</b>
Mobile	<b>React Native 0.74</b> (Expo SDK)
Backend API	<b>NestJS (Node 20)</b> – REST/GraphQL
Base de données	<b>PostgreSQL 15</b>
Fichiers	<b>S3-compatible (MinIO / AWS S3)</b>
IA services	Python FastAPI (Transformers, spaCy, LightGBM)
OCR	PaddleOCR (Docker)
Auth & sécurité	OAuth 2.1 + JWT + MFA (personnel pédago)
CI/CD	GitHub Actions → Docker → K3s
Observabilité	Prometheus, Grafana, Sentry

## 4. Sécurité & conformité

- TLS 1.3 end-to-end ; chiffrement fichiers AES-256 ; TDE sur DB.
- Rôles stricts, logs immuables ; conformité CNDP & RGPD.
- PRA / PCA : backup chiffré quotidien + réplication inter-région.

## 5. Accessibilité & UX

- **WCAG 2.1 AA** (contraste, navigation clavier, screen-reader).
- Mode sombre/clair ; temps de réponse  $\leq 300$  ms pour 95 % des requêtes.
- I18n : FR, AR (v1) prêt pour EN, AMZ.

## 6. Livrables

1. Code mono-repo (React, React Native, Nest) + Docker-files.
2. Helm charts K3s & pipeline CI/CD.
3. OpenAPI 3.1, schémas DB, diagrammes C4.
4. Guides PDF + micro- vidéos.
5. Jeux de données de démo.
6. Rapport complet de tests (unit, intégration, charge).

## 7. Planning opérationnel — 6 semaines

Semaine	Abderrahim (Back-end & Infra)	Mohamed (Front Web React)	Ibtissam (React Native)	Jalons clés
1	Repo Git, CI/CD Docker, K3s.	Valider maquettes Figma, init React + Tailwind.	Init Expo, structure modulaire.	Spécifications détaillées signées.
2	Modèle PostgreSQL, Auth JWT/OAuth, OpenAPI.	Pages Login & Dashboard, design system.	Écrans Login & Liste devoirs, connexion API.	Auth fonctionnelle (web + mobile).
3	End-points Devoirs CRUD + upload S3.	Calendrier devoirs & création.	Soumission devoir (fichier & texte) + preview.	Premier devoir créé, publié, rendu.
4	End-points Examens + convocation PDF/QR, micro-service OCR.	Tableau correction PDF + annotation.	Push FCM, écran devoir corrigé.	Chaîne complète Devoir → Correction.
5	Micro-service IA : correction QCM + résumé parent.	Intégration résultats IA, graphes Chart.js.	Graphes progrès & alertes retard.	Correction IA opérationnelle.
6	Optimisation requêtes, monitoring, dataset démo.	Tests e2e Playwright + polish UI/UX.	Tests iOS/Android réels, optimisation perf.	Démo école pilote, dispo SaaS ≥ 99 %.

## 8. Critères de réussite

- 90 % des enseignants pilotes créent & corrigent devoir + exam sans assistance.
- Temps de correction QCM ↓ ≥ 50 %.
- Taux de remise des devoirs ↑ ≥ 40 % (rappels IA).
- Disponibilité mesurée > 99,5 % sur 4 semaines pilote.

### Slogan interne

**Evalya Smart** – *Devoirs, Examens, Excellence.*