TERMES DE RÉFÉRENCE (TDR)

Recrutement d'un Développeur Mobile & Front-End pour la digitalisation de la santé

1 Contexte et objectif général

Dans le cadre de la modernisation du système de santé mauritanien, le projet INAYA ELARGI vise à déployer un Système d'Information Hospitalier intégré (HIS) adapté aux environnements aux ressources limitées.

Le/la développeur(se) mobile & front-end sera responsable de la conception, du développement et du déploiement de toutes les interfaces utilisateur (web et mobile) permettant :

- Aux personnels soignants et administratifs d'interagir avec le HIS (même hors ligne)
- Aux enquêteurs de collecter des données terrain (satisfaction, campagnes)
- Aux patients d'accéder à leurs informations et services (rendez-vous, résultats)
- Poste basé à Nouakchott avec déplacements fréquents dans plusieurs régions de Mauritanie.

2 Missions principales

2.1 Développement des interfaces web (front-end)

- **Développer les dashboards web responsives pour l'ensemble des modules HIS:**
  - Modules administratifs et financiers (facturation, tiers payant, FBP)
  - Modules cliniques (visualisation dossier patient, prescriptions, résultats)
  - Statistiques sophistiquées (mortalité, comorbidité, coûts) – graphiques interactifs
  - Gestion de planning/scheduling (visualisation agendas, ressources)
  - Archivage des documents scannés (recherche, visualisation)
  - Gestion RH, parc informatique & automobile
  - Enquêtes de satisfaction (administration des campagnes, visualisation résultats)

2.2 Développement d'applications mobiles

- **Concevoir une application mobile multiplateforme (Android/iOS) pour :**
  - **Personnel médical** : Consultation dossier patient (hors ligne), prescriptions, résultats, messagerie sécurisée, alertes
  - **Enquêteurs terrain :** Collecte hors ligne (formulaires dynamiques), médias (photo/audio), géolocalisation, synchronisation
  - Patients : Prise de rendez-vous, consultation résultats, enquêtes satisfaction, rappels push, téléconsultation
- Implémenter le mode hors ligne avec synchronisation différée (zones rurales)
- Gérer le support multimédia (photos, vidéos, audio) pour le dossier patient et les enquêtes

2.3 Interopérabilité mobile ↔ backend

- Consommer les API REST du backend (HIS, FBP, GRH, etc.)
- Implémenter une couche de cache locale (stockage sécurisé des données sensibles)
- Gérer les notifications push (rappel rendez-vous, alertes biologiques, campagnes)
- Assurer la synchronisation asynchrone en mode hors ligne

2.4 Optimisation pour environnements contraints

- Compression automatique des médias avant envoi (bande passante réduite)
- Réduction de la consommation batterie et données
- Adaptation aux écrans de petite taille, faible luminosité, usage extérieur
- Mode dégradé complet (pas de connexion = fonctionnel)

2.5 Soutien aux applications existantes

- Audit, amélioration ou correction des interfaces existantes (web et mobile) pour toutes les entités du projet

2.6 Sécurité et conformité

- Gestion sécurisée des données sensibles (stockage local chiffré AES-256)
- Authentification biométrique (empreinte, reconnaissance faciale) si disponible
- Conformité RGPD / lois locales (anonymisation, consentement explicite)
- Gestion fine des sessions (timeout, révocation à distance)

3 Fonctionnalités attendues (détaillées par module)

3.1 Interfaces web – Modules HIS

3.1.1 Module administratif & financier

- Gestion des patients (identité, NSS, contact, tiers) – formulaires et recherche avancée
- Facturation et recouvrement (création devis, génération reçus, suivi impayés)
- Interface FBP (visualisation indicateurs, export rapports)
- Tableaux de bord financiers (graphiques recettes/dépenses, évolution temporelle)

3.1.2 Module clinique

- Consultation du dossier patient (chronologie médicale, onglets : hospitalisations, prescriptions, examens)
- Prescriptions (visualisation, impression, historique)
- Résultats biologiques et d'imagerie (visualisation intégrée images DICOM/JPEG)
- Comptes rendus (consultation, export PDF)

3.1.3 Statistiques sophistiquées

- Graphiques interactifs (Chart.js / D3.js)
    - Mortalité par période, service, pathologie
    - Comorbidité (indice de Charlson, regroupements)
    - Coûts par séjour, par pathologie, par patient
- Export multiples (PDF, Excel, JSON, PNG)
- Filtres dynamiques (date, service, tranche âge, sexe)

3.1.4 Planning & scheduling

- Visualisation agendas médicaux (vue jour/semaine/mois)
- Gestion des ressources (lits, blocs, équipements)
- Impression planning (PDF)

3.1.5 Archivage scanné

- Recherche full-text sur documents
- Visualisation HTML5 (PDF, images)
- Métadonnées modifiables (date, type, patient associé)

3.1.6 Enquêtes de satisfaction

- Administration campagnes (création, diffusion, clôture)
- Visualisation résultats (NPS, scores par thème, évolution)
- Export données brutes

3.1.7 GRH & Parc informatique/automobile

- Annuaire personnel (recherche, filtres)
- Inventaire équipements (consultation, mise à jour)
- Alertes visuelles (maintenance due, garantie expirée)

3.2 Application mobile – Personnel médical

Fonctionnalité	***Détail***
Authentification	Biométrie + PIN, session persistante chiffrée
Dashboard médical	Liste des patients du jour, alertes (examens urgents)
Dossier patient hors ligne	Consultation des dernières données (médical, prescriptions récentes) – cache automatique
Prescription	Sélection médicaments/examens, validation, mise en file d'attente (sync ultérieure)
Résultats	Notification push à réception, visualisation PDF/images
Messagerie interne	Échange sécurisé avec autres médecins/administratifs
Appel d'urgence	Signalement patient critique (notification prioritaire)

3.3 Application mobile – Enquêteurs terrain

Fonctionnalité	***Détail***
Campagnes	Téléchargement des questionnaires avant départ (Wi-Fi)
Collecte hors ligne	Formulaires dynamiques (text, choix multiples, échelles, date)
Médias attachés	Photo (compression WebP), audio (OPUS), géolocalisation automatique
Synchronisation	Envoi par lots à la reconnexion, file d'attente persistante
Progression	Visualisation nombre d'enquêtes complétées / restantes
GPS	Point de collecte, itinéraire (optionnel)

3.4 Application mobile – Patients

Fonctionnalité	***Détail***
Inscription / liaison	Création compte, liaison avec dossier hospitalier (NSS/phone)
Rendez-vous	Visualisation créneaux disponibles, prise, modification, annulation
Rappels	Notification push (J-1, H-2), option SMS (secours)
Résultats	Consultation examens (PDF protégé, images)

Enquêtes	Réponse aux campagnes (push invitation)
Téléconsultation	Préparation lien vidéo (WebRTC) – intégration future
Multilingue	Interface en français/arabe + langues nationales

3.5 Multimédia – support complet

Type	*Capture*	*Visualisation*	*Compression*
Images	Appareil photo / galerie	Galerie intégrée	WebP (80% réduction)
Vidéos	Galerie (import)	Lecteur intégré	H.265/HEVC
Audio	Microphone	Lecteur + transcription	OPUS
Documents	Import PDF/JPEG	Visualisation HTML5	PDF réduit

3.6 Mode hors ligne & synchronisation (architecture détaillée)

Composant	*Spécification*
Stratégie cache	LRU (Least Recently Used) – taille max 500 Mo configurable
File d'attente transactions	FIFO persistante (SQLite locale) – retry exponentiel (1min, 5min, 15min, 1h)
Résolution conflits	Timestamp client + serveur – priorité au plus récent + log de conflit
Seuil synchronisation	Wi-Fi uniquement pour médias lourds > 5 Mo
Signalement sync	Indicateur visuel (vert = ok, orange = file non vide, rouge = erreur)

4 Production et documentation

4.1 Livrables logiciels

Livrable	*Description*
Code source	Application mobile (Flutter/RN) + front-end web (React/Vue) – dépôt Git
Application mobile	APK (Android) + IPA (iOS) – version hors ligne fonctionnelle
Interface web	PWA déployable sur serveur central
Assets	Icônes, images, traductions (français/arabe/langues nationales)

4.2 Documentation technique

Document	*Format*
Architecture front-end/mobile	Markdown + diagramme séquence
Guide d'installation (build APK/PWA)	PDF / MkDocs
API consommées	Swagger/OpenAPI (lié au backend)
Gestion hors ligne	Stratégie cache, file d'attente, résolution conflits

4.3 Documentation utilisateur

Profil	*Documents (français + arabe)*
Médecins / infirmiers	Prise en main mobile (consultation, prescription hors ligne)
Enquêteurs	Collecte hors ligne, envoi médias, géolocalisation
Patients	Téléchargement, rendez-vous, résultats, enquêtes
Administratifs	Dashboard web (gestion campagnes, visualisation réponses)

4.4 Rapports périodiques

Périodicité	***Contenu***
Hebdomadaire	Avancement, blocages, versions livrées
Mensuel	Métriques : téléchargements, bugs, satisfaction utilisateurs tests

4.5 Support formation

- Formation utilisateurs (sessions terrain dans les régions)
- Tutoriels vidéo (sous-titrés français/arabe)
- Kit de dépannage (réinitialisation cache, mode hors ligne)

5 Profil recherché

5.1 Formation / Qualifications

- Bac + 3 (Licence) en Informatique, Génie logiciel, Systèmes d'information, ou équivalent reconnu par l'État mauritanien

5.2 Compétences techniques

5.2.1 Développement mobile (un des deux socles obligatoire)

Flutter (socle recommandé) :

- Maîtrise du langage Dart
- Connaissance d'au moins une solution de gestion d'état : BLoC, GetX, ou Provider
- Utilisation courante de plugins essentiels : camera, geolocator, sqflite (ou hive), image_picker, flutter_secure_storage
- Capacité à produire un APK signé et à déployer sur iOS (TestFlight)

React Native (socle alternatif) :

- Maîtrise de JavaScript / TypeScript
- Gestion d'état avec Redux (ou MobX)
- Navigation avec React Navigation
- Utilisation de modules natifs (camera, géolocalisation, stockage sécurisé)

5.2.2 Développement front-end web (un des deux obligatoire)

React.js :

- Hooks (useState, useEffect, useContext)
- React Router pour la navigation
- Context API ou Redux pour l'état global
- Expérience avec Vite ou Next.js (apprécié)

Vue.js :

- Composition API (ou Options API maîtrisée)
- Vue Router
- Pinia ou Vuex pour la gestion d'état

Exigences communes web :

- Développement de PWA (Progressive Web App) avec service worker
- Stratégies de cache hors ligne (Workbox ou équivalent)
- Responsive design (desktop, tablette, mobile)

5.2.3 Gestion de l'état et cache hors ligne

Mobile :

- Stockage local persistant : SQLite (sqflite) ou Hive (Flutter) / AsyncStorage ou Realm (React Native)
- Gestion de file d'attente des transactions non synchronisées (persistante après redémarrage)

Web :

- IndexedDB pour le stockage hors ligne
- Stratégies de cache : cache-first, network-first, stale-while-revalidate

Connaissances transversales :

- Mise en œuvre de la synchronisation différée (replay à la reconnexion)
- Résolution de conflits (timestamp, priorité serveur, journalisation des conflits)
- Gestion de la taille du cache (LRU, purge automatique)

5.2.4 Consommation d'API et communication asynchrone

Clients HTTP :

- Flutter : Dio ou http
- React Native / Web : Axios ou fetch API

Communications temps réel :

- WebSocket pour notifications et messagerie instantanée
- Gestion de la reconnexion automatique

Gestion des files d'attente hors ligne :

- Mise en œuvre d'une file d'attente locale (FIFO)
- Mécanisme de retry exponentiel (1 min, 5 min, 15 min, 1 heure)
- Persistance des actions non envoyées après redémarrage de l'application

5.2.5 Multimédia

Capture et import :

- Images : depuis appareil photo ou galerie
- Vidéos : import depuis galerie
- Audio : enregistrement microphone
- Documents : import PDF, JPEG

Compression :

- Images : conversion automatique vers WebP (réduction d'environ 80%)
- Vidéos : encodage H.265 / HEVC
- Audio : encodage OPUS

Visualisation :

- Galerie d'images intégrée
- Lecteur vidéo embarqué
- Lecteur audio avec visualisation de la piste
- Visualisation PDF (PDF.js ou équivalent)

5.2.6 Géolocalisation et cartographie (GIS)

Mobile :

- geolocator (Flutter) ou react-native-location
- Collecte de points GPS haute précision
- Stockage des traces en GeoJSON

Web :

- Leaflet ou MapLibre GL pour l'intégration cartographique
- Affichage de points d'intérêt et itinéraires

Cas d'usage spécifiques :

- Localisation des enquêtes terrain
- Cartographie des équipements de santé (parc informatique/automobile)
- Itinéraires pour visites à domicile (optionnel)

5.2.7 Notifications push

Outils maîtrisés :

- Firebase Cloud Messaging (FCM) – obligatoire pour Android
- OneSignal (optionnel, pour fallback ou simplification)

Fonctionnalités à implémenter :

- Notifications de rappel (rendez-vous J-1, H-2)
- Alertes biologiques (résultat critique)
- Invitations à répondre aux enquêtes de satisfaction
- Notifications silencieuses (mise à jour cache en arrière-plan)

Web :

- Push API avec service worker (pour PWA)

5.2.8 Sécurité mobile et front-end

Stockage sécurisé local :

- flutter_secure_storage (Flutter) ou react-native-encrypted-storage
- Chiffrement AES-256 des données sensibles (jetons, données patient)

Authentification :

- Intégration de la biométrie (empreinte digitale, reconnaissance faciale) via local_auth (Flutter) ou react-native-biometrics
- Gestion de session avec timeout automatique
- Révocation de session à distance (appel API)

Communications réseau :

- HTTPS obligatoire
- Certificate pinning (recommandé pour environnements sensibles)
- Validation des certificats côté client

Conformité :

- Gestion explicite du consentement utilisateur (traçabilité)
- Anonymisation des données en mode debug / test

5.2.9 Tests (mobile et front-end)

Tests unitaires :

- Flutter : flutter_test
- React Native / Web : Jest
- Couverture minimale exigée : 60% sur le code UI et la logique métier

Tests d'intégration (mobile) :

- Flutter : Patrol (recommandé) ou integration_test
- React Native : Detox

Tests end-to-end (web) :

- Playwright ou Cypress
- Trois parcours critiques à couvrir : patient (prise rendez-vous), médical (consultation dossier), administrateur (campagne enquête)

Tests spécifiques hors ligne :

- Simulation d'absence de réseau (outils : Chrome DevTools, Network Link Conditioner)
- Vérification du fonctionnement complet des formulaires hors ligne
- Validation de la reprise de synchronisation à la reconnexion

Tests de charge (front uniquement si pertinent) :

- Simulation d'affichage de tableaux de bord avec grand volume de données (1000+ lignes)

5.2.10 Outils transversaux

Versionnement :

- Git (obligatoire)
- Stratégie de branches (main, dev, feature/, hotfix/)
- Messages de commit conventionnels (feat:, fix:, docs:, style:)

CI/CD :

- GitLab CI ou GitHub Actions
- Automatisation de la build APK
- Déploiement automatisé de la PWA

Monitoring et reporting d'erreurs :

- Sentry (crash reporting mobile et web)
- Firebase Analytics (statistiques d'usage, anonymisées)

Intégration de maquettes :

- Capacité à intégrer des maquettes Figma ou Adobe XD
- Respect des chartes graphiques et des composants design system

Documentation API :

- Swagger / OpenAPI (lecture et compréhension)

Environnements de développement :

- Android Studio (pour Flutter / RN)
- Xcode (pour build iOS, si disponible)
- VS Code (optionnel)

5.2.11 Méthodologie et pratiques

UML (niveau minimum) :

- Compréhension des diagrammes de séquence (pour comprendre les flux backend)
- Capacité à produire un diagramme de composants front-end / mobile

AGILE :

- Expérience de travail en Scrumban ou Kanban
- Participation aux daily meetings et rétrospectives
- Utilisation d'outils de suivi (Redmine, Taiga, GitLab Issues)

Code assisté par IA :

- Autorisation d'utilisation de GitHub Copilot ou Codeium
- Obligation de relecture humaine et de validation avant intégration

Documentation :

- Rédaction de documentation technique claire en français

- Rédaction de documentation utilisateur multilingue (français / arabe)

5.3 Expériences

5.3.1 Expérience Générale

- 3+ ans (Minimum 3 ans) de développement mobile et front-end
- Justification par CV détaillé, références et portfolio de réalisations

5.3.2 Expérience spécifique – Santé (indispensable)

- Participation à au moins un projet d'application mobile ou web dans le secteur sanitaire (HIS, DPI, e-santé, système hospitalier)
- Connaissance des contraintes liées aux données médicales (confidentialité, traçabilité, consentement)

5.3.3 Expérience FBP (Financement Basé sur la Performance), Atout majeur :

- Attestation
- Expérience dans la visualisation ou la collecte d'indicateurs de performance (tableaux de bord, scores qualité/quantité)

Avoir une expérience avérée dans le domaine du Financement Basé sur la Performance (FBP) appliqué à la santé est un atout majeur.

Les preuves acceptées sont les suivantes :

1) **Attestation professionnelle** signée par un employeur ou un maître d'ouvrage (hôpital, ONG, ministère, projet FBP) décrivant clairement le rôle du candidat et les fonctionnalités FBP développées. Les fonctionnalités attendues concernent la visualisation d'indicateurs (tableaux de bord, scores de qualité et de quantité) ou la collecte de données sur le terrain servant au calcul FBP.
2) **Dépôt de code ou capture d'écran probante** montrant une application mobile ou web affichant des indicateurs de performance (par exemple : scores composites, graphiques de qualité des soins, taux de réalisation, etc.). Un lien vers un dépôt GitHub ou un accès à une démonstration en ligne sera demandé.
3) **Déclaration sur l'honneur détaillée** (2 pages maximum) décrivant le projet FBP auquel le candidat a participé, avec le nom de l'organisation porteuse, la période d'intervention, les fonctionnalités réalisées, et les coordonnées complètes (téléphone, email) d'une personne référente technique vérifiable.

Tout candidat ne fournissant aucun des éléments ci-dessus dans son dossier pour expérience obligatoire sera éliminé d'office à l'étape de présélection.

5.3.4 Expérience terrain

- Travail en environnements à faible connectivité (bande passante réduite, réseau intermittent)
- Mise en œuvre de modes hors ligne et synchronisation différée
- Adaptation d'interfaces pour usage extérieur (forte luminosité, écrans de petite taille)

5.4 Langues

Français :

- Niveau courant à l'oral et à l'écrit
- Capacité à rédiger une documentation technique et utilisateur
- Capacité à animer une formation utilisateur

Arabe :

- Niveau courant à l'oral et à l'écrit
- Capacité à traduire ou superviser la traduction des interfaces et des guides
- Capacité à former des utilisateurs en arabe dans les régions

Anglais :

- Lecture technique indispensable (documentation des packages, API, stack overflow)
- L'expression orale n'est pas exigée mais constitue un atout

Langues nationales (optionnel mais apprécié) :

- Pulaar, Soninké, Wolof – pour les interfaces patients (traduction des libellés simples)

6 Durée et conditions d'exécution

Lieu de résidence :

- Basé à Nouakchott (Mauritanie)

Déplacements :

- Plusieurs régions de Mauritanie à prévoir (tests terrain, formation des utilisateurs, déploiement)
- Prise en charge intégrale par le projet (transport, hébergement, indemnités)

Disponibilité :

- Temps plein (40 heures par semaine)

Durée du contrat :

- 1 an

Matériel fourni :

- PC développement sous Linux (Ubuntu ou Linux Mint) ou Windows 11
- Smartphone Android pour tests (configuration récente, Android 11 minimum)
- Accès VPN sécurisé

7 Environnement technique cible (pour information)

Élément	*Spécification*
Backend	API REST (Python FastAPI/Django / Rust Go Ruby/ Rails) – développée par l'équipe backend
Base de données	PostgreSQL, Redis, MariaDB, SQLite, MongoDB, CouchDB
Hébergement web	Serveur Linux (Debian/Ubuntu) + Nginx/Apache

Environnement mobile	Android 8.0 minimum, iOS 13 minimum
Réseaux ruraux	2G/3G intermittent, bande passante < 512 kbps

8 Critères de sélection

Critères obligatoires (éliminatoires)

- Diplôme : Bac + 3 minimum en Informatique (copie obligatoire)
- Attestation FBP : preuve d'expérience dans un projet de Financement Basé sur la Performance (attestation ou dossier détaillé)
- Langues : français et arabe courants (oral et écrit)
- Disponibilité : temps plein, basé à Nouakchott avec déplacements en régions

Critères de qualification

- Expérience générale en développement (3 ans minimum)
- Expérience spécifique dans le secteur de la santé (HIS, DPI, e-santé)
- Maîtrise des technologies exigées (cf. compétences techniques)
- Portfolio ou réalisations consultables (GitHub, APK, etc.)
- Références techniques vérifiables (dont une dans le secteur santé)

Complément d'appréciation (non bloquant)

- Connaissance de l'anglais
- Expérience en environnements contraints (faible connectivité, zones rurales)
- Connaissance des langues nationales (Pulaar, Soninké, Wolof)

9 Dossier de candidature

Les candidats doivent envoyer :

1. CV détaillé (expériences mobile/front + liens)
2. Lettre de motivation
3. Copie du diplôme Bac + 3 Informatique
4. Attestation FBP (ou déclaration + référence)
5. Liens vers réalisations (Google Play / GitHub / APK testable)
6. Deux références techniques (dont une santé)

Date limite : [À préciser]

Envoi à : [Email du recruteur]

Fait à Nouakchott, le [date]