

# Rapport - Application de Suivi de Grossesse

---

## 1. Contexte et Objectifs

### a. Contexte

Au Sénégal, de nombreuses femmes enceintes, en particulier dans les **zones rurales** (comme le Fouta, le Sine-Saloum ou la Casamance) ou dans les **quartiers urbains défavorisés** (banlieues de Dakar, Pikine, Guédiawaye, etc.), font face à des **difficultés d'accès au suivi médical régulier** durant leur grossesse. Plusieurs facteurs expliquent cette situation : l'éloignement des structures de santé, le coût du transport, le manque d'information ou encore la barrière de la langue.

La majorité des applications mobiles de santé existantes sont proposées en **français**, souvent dans un langage technique peu adapté au quotidien des femmes peu ou pas scolarisées. Or, au Sénégal, une large partie de la population féminine communique principalement en **wolof**, en **pulaar**, voire dans d'autres langues nationales comme le sérère ou le diola, et n'est pas toujours à l'aise à l'écrit.

C'est dans ce contexte que s'inscrit notre projet : développer une **application web et mobile adaptée aux réalités linguistiques, sociales et culturelles sénégalaises**, intégrant des **messages vocaux en wolof et en pulaar**, pour permettre aux femmes de **suivre leur grossesse de manière autonome, sécurisée et contextualisée**.

Il s'agit d'un outil de **santé communautaire numérique** qui ambitionne de **renforcer l'accès à l'information maternelle** dans un langage simple, audio et familier, tout en tenant compte du **niveau d'alphabétisation** et de la **connectivité parfois faible** dans certaines zones.

## b. Objectifs

### ❖ Objectif principal :

Créer une application web et mobile simple et multilingue pour aider les femmes enceintes à suivre leur grossesse, avec des rappels, des conseils santé et des messages vocaux.

### ❖ Objectifs spécifiques :

- Suivi personnalisé de grossesse (semaine par semaine)
- Rappels de rendez-vous médicaux
- Conseils de santé simples et contextualisés
- Messages vocaux en wolof ou pulaar
- Interface accessible et adaptée à des utilisatrices peu lettrées

## 2. État des lieux

Avant de concevoir l'application, il est essentiel d'évaluer les solutions existantes sur le marché. Cette comparaison permet de mettre en évidence les lacunes des applications actuelles et de justifier l'intérêt de notre projet. L'accent est mis ici sur trois critères essentiels : la langue utilisée, l'adaptation au contexte africain et la présence de fonctionnalités vocales.

Le tableau suivant illustre les principales caractéristiques de quelques applications de suivi de grossesse bien connues et permet de situer notre application dans ce paysage

État des lieux			
Application	Langue	Afrique	Vocal
Grossesse+	Français/Anglais	✗	✗
Flo	Français/Anglais	✗	✗
Notre App	Wolof/Français/Pulaar	✓	✓

### 3. Description du besoin et faisabilité

#### ❖ Cible

Notre application s'adresse en priorité à :

- - Femmes enceintes
- - Peu ou pas alphabétisées
- - Wolof ou pulaar comme langue principale
- - Afrique de l'Ouest francophone

#### ❖ Fonctionnalités clés

Pour répondre à ces besoins spécifiques, l'application proposera :

- - Calcul automatique de la semaine de grossesse via DPA
- - Conseils santé hebdomadaires (texte + audio)
- - Notifications pour les rendez-vous et suivis
- - Lecture vocale des messages dans la langue choisie
- - Fonctionnement hors ligne partiel (cache local)

### 4. Architecture fonctionnelle

L'**architecture fonctionnelle** de l'application décrit les principales fonctionnalités accessibles à l'utilisatrice. Ces fonctions sont organisées autour de la simplicité d'usage, l'accessibilité linguistique, et le respect des besoins spécifiques des femmes enceintes dans un contexte africain. L'interface est volontairement épurée, intuitive et centrée sur les usages concrets.

- ❖ Tableau de bord de suivi
- ❖ Mon Profil (langue, prénom, DPA)
- ❖ Conseils hebdomadaires
- ❖ Mes rendez-vous (ajout/rappel)
- ❖ Lecteur audio multilingue
- ❖ Paramètres (langue, alertes)

## 5. Architecture applicative

L'architecture applicative repose sur une organisation **modulaire** et **moderne**, pensée pour garantir la simplicité d'usage, la maintenabilité du code et la facilité de déploiement dans des contextes à faible connectivité.

Elle est composée de plusieurs couches clairement définies :

Planning prévisionnel		
Couche	Outils/techno	Détail
Frontend	React.js (web)	L'interface utilisateur est développée en <b>React.js</b> , un framework léger, rapide et bien adapté pour des applications web responsives accessibles depuis mobile ou ordinateur.
Backend	NestJS (Node.js + TypeScript)	L'API est gérée par <b>NestJS</b> , un framework Node.js basé sur TypeScript, offrant une architecture propre et évolutive.
Base de données	PostgreSQL	<b>PostgreSQL</b> , base robuste et open source, stocke toutes les données : utilisateurs, conseils, rendez-vous, notifications.
API vocal	Google Text-to-Speech API	L'intégration de <b>Google Text-to-Speech</b> permet de générer automatiquement des fichiers audio lisibles dans les langues locales (wolof, pulaar).
Notifications	Push via web (Web Push)	Des <b>Web Push Notifications</b> assurent des rappels automatisés pour les rendez-vous ou les conseils hebdomadaires.
Stockage audio	Google Cloud Storage / local	Les fichiers vocaux sont hébergés via <b>Google Cloud Storage</b> ou en local selon les besoins et la connectivité.
Authentification	Auth par téléphone / anonymat	Une authentification simple par numéro de téléphone (ou anonyme) garantit une prise en main facile, sans complexité inutile.

## 6. Modèle de données (PostgreSQL)

### a. Utilisateur

Contient les informations de base de chaque femme enceinte.

Champ	Type	Description
id	UUID	Identifiant unique
prénom	Texte	Prénom de l'utilisatrice
langue	Texte	Langue préférée (wolof, pulaar, français)
DPA	Date	Date Présumée d'Accouchement
dateInscription	DateTime	Date de création du compte

### b. ConseilGrossesse

Stocke les conseils à fournir chaque semaine, dans plusieurs langues et avec support vocal.

Champ	Type	Description
id	UUID	Identifiant unique
semaine	Int	Numéro de la semaine de grossesse
titre	Texte	Titre du conseil
texte	Texte	Contenu du message en texte
audioURL	Texte	Lien vers le fichier audio généré (Google TTS)
langue	Texte	Langue du message

### c. RendezVous

Permet à l'utilisatrice de suivre ses rendez-vous médicaux et recevoir des rappels.

Champ	Type	Description
id	UUID	Identifiant unique
utilisateurId	UUID	Clé étrangère vers l'utilisateur concerné
type	Texte	Type de rendez-vous (consultation, vaccin)
date	Date	Date prévue du rendez-vous
rappelActivé	Booléen	Indique si un rappel est activé

### d. Notification

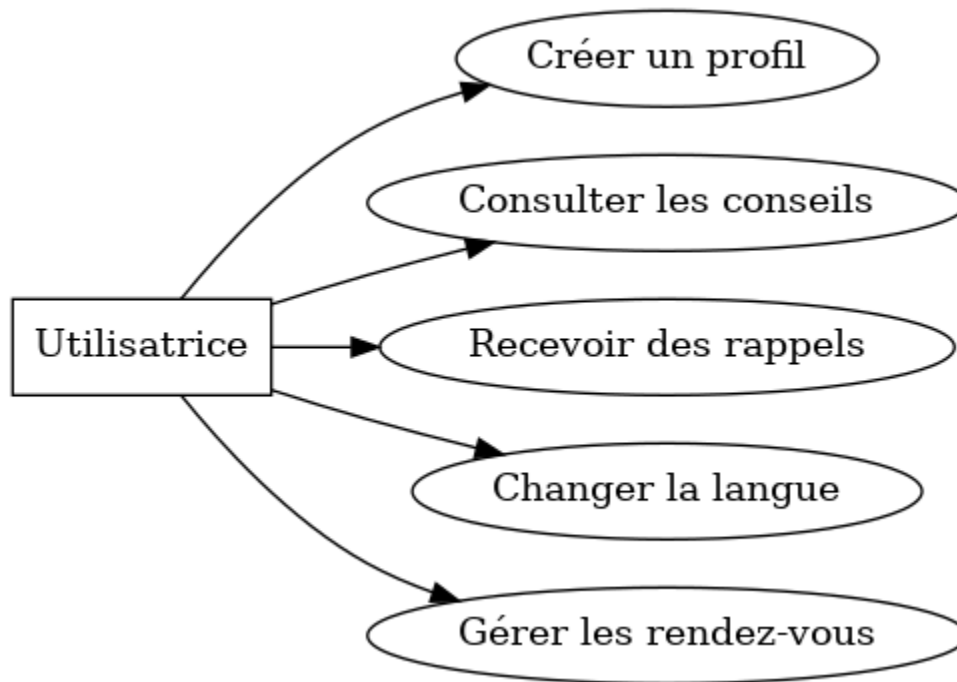
Historique des messages envoyés à chaque utilisatrice.

Champ	Type	Description
id	UUID	Identifiant unique
utilisateurId	UUID	Référence à l'utilisatrice concernée
message	Texte	Contenu du message de notification
type	Texte	Nature (rappel, information, conseil)
dateEnvoi	DateTime	Date et heure de l'envoi

## 7. Modèle de données (PostgreSQL)

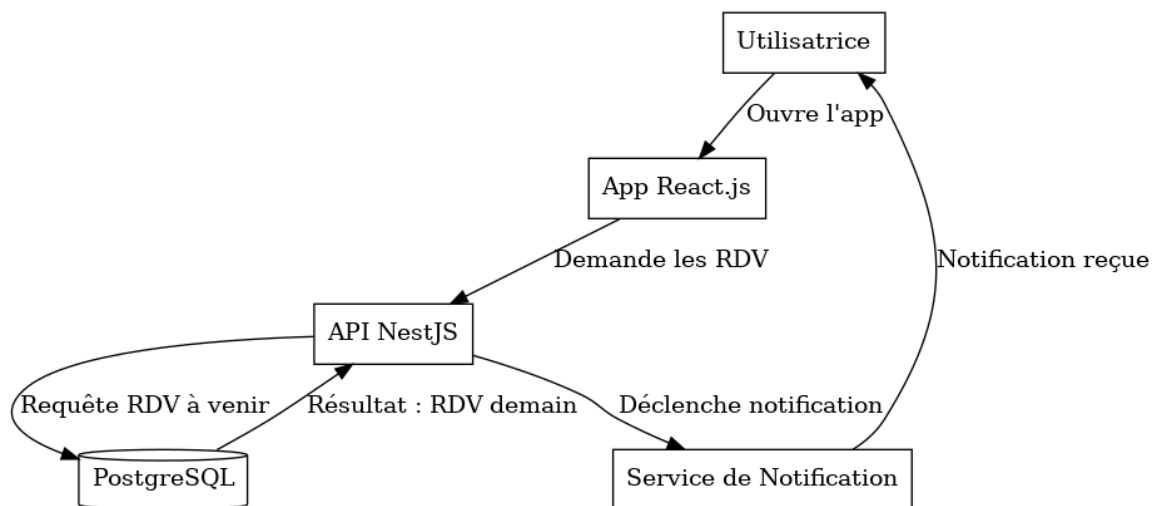
### a. Diagramme de cas d'utilisation

Montre les principales actions accessibles à l'utilisatrice



### b. Diagramme de séquence

Décrit la chaîne d'événements lors de la réception d'une notification



## 8. Technologies utilisées

Le projet repose sur un ensemble de **technologies modernes, performantes et accessibles**, sélectionnées pour répondre à la fois aux **contraintes locales** (connectivité, simplicité d'accès) et aux **exigences techniques** (modularité, maintenabilité, évolutivité).

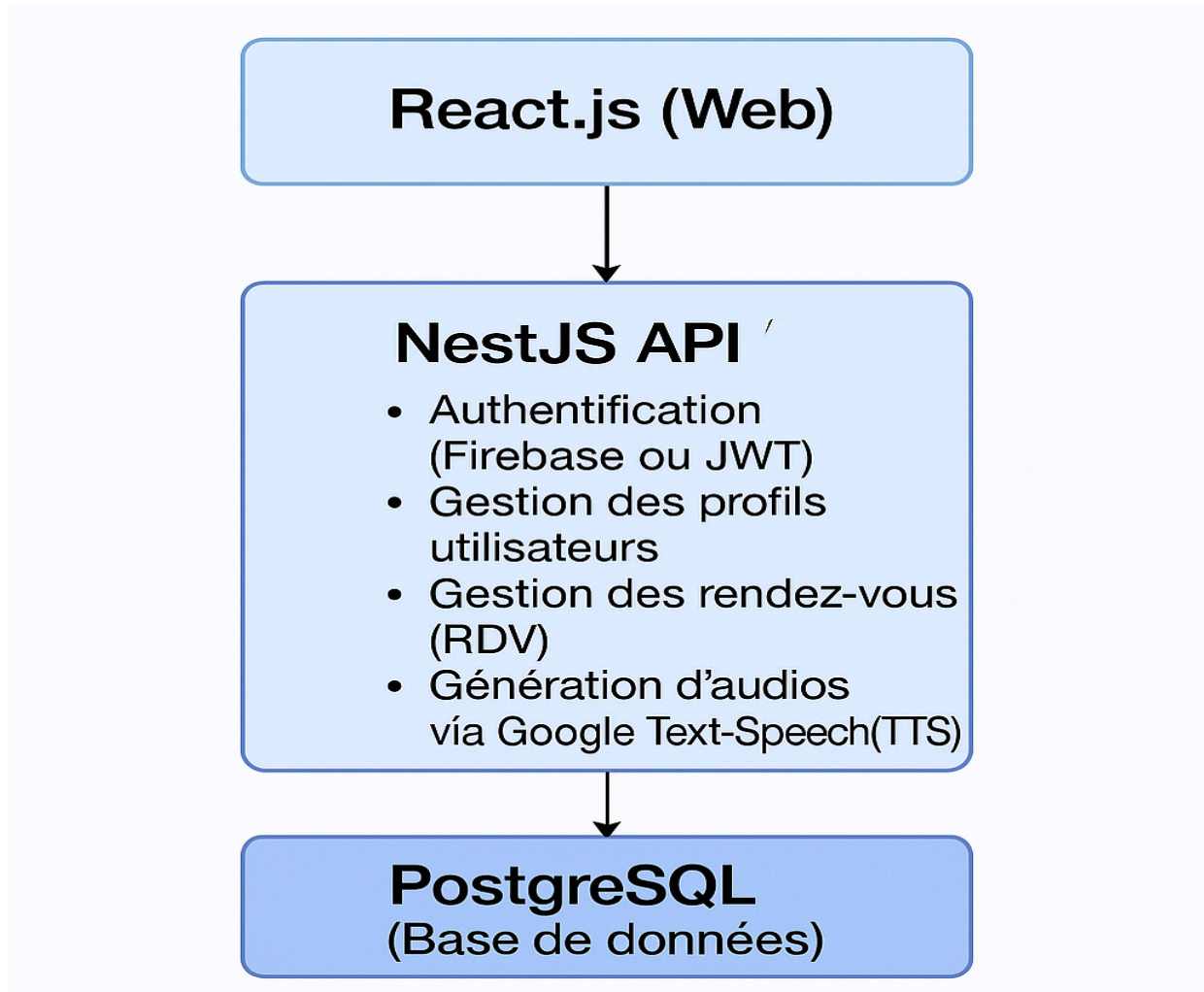
Technologies utilisées	
Domaine	Outils / Langages
Frontend	React.js, Tailwind CSS
Backend API	NestJS, TypeScript
Base de données	PostgreSQL
Authentification	Magic link / OTP via Twilio (option)
Notifications	Web Push API
Audio	Google TTS / enregistrement manuel
Hébergement	Render, Railway, Vercel, Supabase
Stockage fichiers	Google Cloud ou Supabase Storage
Versionnage	Git + GitHub
Design UI	Figma



## 9. Architecture technique

L'application repose sur une architecture **en couches**, avec des responsabilités clairement séparées entre l'interface utilisateur, la logique métier (API), et les services de données. Elle est conçue pour être **scalable**, **sécurisée** et facilement **maintenable**.

Schéma technique simplifié :



## 10. Estimation technique

L'objectif ici est d'évaluer la **faisabilité pratique et économique** du projet, en tenant compte des ressources disponibles, des contraintes d'hébergement, et des coûts liés aux fonctionnalités comme l'audio ou les notifications.

Estimation technique	
Élément	Estimation
Utilisatrices test	100 à 300
Audio (par langue)	10-20 Mo (compressé)
Coût Google TTS	0 FCFA jusqu'à 4M caractères/mois
Notifications	gratuites via Web Push API
Hébergement	Gratuit (Railway, Vercel, Supabase)

## 11. Planning prévisionnel

Le planning suivant répartit les tâches de développement sur une période de **8 semaines**, en tenant compte de la phase de conception, de développement, de test, et de la démonstration..

Planning prévisionnel	
Semaine	Tâches
1	Étude besoins, planification
2	Maquettes UI (Figma) + Modélisation BDD
3	Dév frontend React.js (formulaires, tableau bord)
4	Dév backend NestJS (routes, logique, PostgreSQL)
5	Intégration API audio TTS
6	Notifications + tests
7	Ajustements, validation
8	Démo

## 12. Budget prévisionnel

L'objectif du projet est de **minimiser les coûts** tout en garantissant un service fonctionnel et fiable. Grâce à l'utilisation de **services cloud gratuits** dans leurs versions de base (Vercel, Supabase, Google TTS), le budget reste très réduit.

Budget prévisionnel	
Ressource	Coût
Google TTS	Gratuit
Vercel / Railway / Supabase	Gratuit
Nom de domaine (optionnel)	18 400 CFA
Total	18 400 CFA

## 13. Équipe projet

Equipe de Projet	
Membre	Membre
Elhadji abdou aziz Diagne	Frontend (React), UI/UX
???	Frontend (React), UI/UX

## 14. Analyse des risques

Comme tout projet technologique, cette application présente certains risques, notamment liés à la connectivité, à l'usage de services externes (API vocales), ou à la gestion du planning. Une **anticipation rigoureuse** et des **solutions concrètes** ont été envisagées pour chaque point critique.

Analyse des risques		
Risque	Risque	Solution
Voix locale indisponible	Moyen	Utiliser Google TTS avec bonne prononciation
Connexion lente ou absente	Élevé	Cache audio/text en local
Retard de développement	Élevé	Découpage précis + Trello suivi
Difficulté API audio	Moyen	Génération batch des fichiers audio