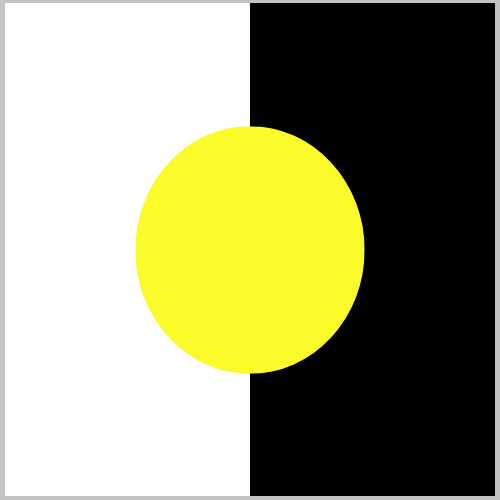
|  |
| --- |
|  |
| Cahier des charges |
| Gestion et localisations des flottes : FLOWATCH |
| **François Nsengimana** |
| **12/11/2024** |
|  |



**IMANA-47**

|  |
| --- |
|  |

# Historique

**Auteur Remarques Date n° version**

François Nsengimana Création du squelette de documents 04/10/2024 0.1

# Auteurs

**Auteur Fonction Contact**

François Nsengimana Concepteur de l’applications [francknsengimana@gmail.com](mailto:francknsengimana@gmail.com)

SOMMAIRE

Sommaire...................................................................................................................................2.

Lexique technique............................................................................................................................................3 . Présenter le contexte

2. Présenter le projet

4 Les objectifs du projet.....................................................................................................4

Les cibles du projets..................................................................................................................................

4 Les interfaces existantes ...........................................................................................................................4

La concurrence..................................................................................................................................................4 La présentation du projet........................................................................................................................... 6

La ou les plateforme(s)..................................................................................................................................6 L’équipement........................................................................................................................................... 6 Modèle économique.................................................................................................................................... 6 Résultat attendu ....................................................................................................................................... 6 Description fonctionnelle...........................................................

7 Parcours utilisateur.................................................................................................................................. 7 Les fonctionnalités de l’application.................................................................................................

7 Back office..............................................................................................................................................................................................8 Contraintes techniques ou non- fonctionnelles ...................................................................................................... 9 éléments graphiques ....................................................................................................................................................................9 Pilotage..........................................................................................................................................................................................................11 Contacts.......................................................................................................................................................................................................12

# Introduction

Ce document est un cahier des charges pour la conception et le développement d’une application web puis mobile dédiée à la gestion, la localisation des véhicules et des objets personnels (comme des téléphones, ordinateurs, etc.). Cette application s’adresse principalement aux entreprises de transport, aux conducteurs de véhicules commerciaux, et aux particuliers souhaitant sécuriser leurs biens qui se nomme « **FLOWATCH** », en particulier en Afrique, tout en ayant une portée internationale.

## Notes de l’auteur

Le responsable du projet **FLOWATCH**  se nomme **François Nsengimana**. Il sera votre principal interlocuteur en cas de questions (ses coordonnées sont définies plus haut).

2. Présenter le projet

### À propos de l’application

L’objectif est de permettre :

* **Gestion des flottes et conducteurs** : Permettre à l’administrateur de gérer les conducteur des flottes.
* **Envoyer et recevoir de l’argent**: Permettre au conducteur d’envoyer et recevoir leurs argent(salaire).
* **Communication via un interface chat**: Permettre à l’administrateur et le conducteur de communiquer par sms, appel.
* **Suivi en temps réel des véhicules** : Permettre aux utilisateurs de localiser et de suivre en temps réel les flottes.
* **Sécurité des flottes** : Intégration de dispositifs de sécurité pour prévenir le vol et faciliter la récupération des flottes volés.
* **Localisation des flottes et objets personnels** : Suivi des objets volés (téléphones, ordinateurs, appareils électroniques) via des technologies de géolocalisation.
* **Optimisation de la gestion de la flotte** : Permettre une gestion efficace des flottes en suivant leur état, leur consommation, et en optimisant les itinéraires.
* **Accessibilité et compatibilité** : Assurer une compatibilité multi-plateformes (Responsive) et multilingue (français, anglais) mais pour la premier version l’application sera en français uniquement.

### Destinataires

* **Entreprises de transport** (public et privé) = Administrateur.
* **Particuliers**
* **Conducteurs de véhicules**
* **Autorités locales**

Note : Pour la première version les principaux acteurs :

**- Administrateur**

Nom :

Prénom :

Téléphone :

Fonction :

Email :

Adresse :

**- Conducteur**

Matricule :

Numéro permis :

Code postale :

Nom :

Prénom :

Date de naissance :

Date d’embauche :

Nationalité :

Téléphone :

Flotte :

Email :

Adresse :

## ****Fonctionnalités** **principales****

## ****Pour la Première versions :****

## ****Localisation:****

* **Suivi GPS en temps réel** :

Localisation précise des flottes à tout moment via un affichage cartographique interactif. Permettre aux utilisateurs de suivre des objets volés comme des téléphones, des ordinateurs portables ou des appareils électroniques. Pour cela, l'application pourrait utiliser des technologies comme la localisation GPS, Traceur, le Wi-Fi, ou encore des balises Bluetooth pour retrouver ces objets ou google maps.

* **Historique des trajets** : Consultation des trajets passés et statistiques associées (distance parcourue, temps, vitesse moyenne).
* **Alertes de géo-clôture** :

Notifications lorsque le véhicule entre ou sort d’une zone géographique

Prédéfinie.

Créer des zones géographiques et recevoir des notifications lorsqu’une flotte

entre ou sort d'une zone spécifique, ce qui peut être très utile pour prévenir

les vols ou suivre les itinéraires des flottes en temps réel.

## ****Gestion des comptes :****

* **Gestion compte conducteurs** et **administrateur** : Modifier, supprimer ajouter le compte de chauffeurs ou de l’administrateur.
* **Suivi du comportement des conducteurs** : Analyse du style de conduite (excès de vitesse, freinages brusques, etc.).
* **Évaluations des conducteurs** : Permettre aux passagers ou aux responsables d’évaluer la conduite du conducteur.

## ****Gestion des**** ****flottes :****

* **Conducteur** : Gérer modifier ajouter supprimer un compte conducteur.
* **Flottes** : Gestion et contrôle, modification ajoute, suppression d’une flotte.
* **Achat et location** : Vérification et modification des information et suppression.
* **Assurance taxe visite** : Vérification des assurance, taxe et visite.

## ****Messagerie :****

* Intégrer un système de communication de chat et appelle entre les conducteur et l’administrateur.

# ****Gestion des finances :****

* **Analyse des coûts** : Suivi des dépenses liées aux véhicules (carburant, entretien, réparations, etc.).
* **Facturation et paiements** : Gestion des paiements pour le salaire des conducteurs.
* **Gestion envoi et réception argent** : Intégrer un système de paiement en ligne pour faciliter les transactions entre le conducteur et l’administrateur.

1. **Paramètre :**

* Contrôler et styliser l’application.

## ****Pour version future :****

## ****Suivi des objets personnels :****

* **Localisation des objets volés** : Suivi des objets électroniques via GPS ou balises Bluetooth ou google maps.
* **Historique des déplacements des objets** : Permet de consulter les trajets effectués par un objet et détecter d’éventuels mouvements suspects.

**Sécurité des données :**

* **Cryptage des données personnelles** : Assurer la sécurité des informations des utilisateurs, en particulier pour les données sensibles liées à la localisation et aux objets suivis.
* **Conformité aux réglementations locales** : Veiller à ce que l'application respecte les lois et règlements locaux concernant la vie privée et la protection des données.

# Description générale

## Environnement

### Les interfaces avec le système

Nous recommandons l’usage d’un système d’exploitation de type Windows, Mac Linux.

Il faut dans leurs dernières versions stables :

* Un outil pour le déploiement : Git, Github,
* Docker .
* Un serveur web : Apache, nginx, ..
* Une base de données : MySQL
* Un moyen d’envoyer des e-mails : système, API, …

### Les interfaces avec les utilisateurs

#### Caractéristiques

Il faut que :

* Le site
  + S’affiche sur une résolution de 1024x768 sur PC.
  + Soit responsive : l’utilisation de Bootstrap.
  + Soit compatible avec les dernières versions des navigateurs : Chrome, Firefox et IE.
* L’ergonomie globale du site respecte les points suivants :
  + Chaque demande de suppression d’une entité fait l’objet d’une confirmation.
  + Le menu du site Internet se situera à droite.

### Les interfaces avec le matériel et contrainte de mémoire

La machine doit être dédiée à l’application.

Le nombre de processeurs et leur fréquence restent à la discrétion du prestataire, ainsi que de la mémoire vive.

Il a cependant été noté que le système de fichiers devait disposer d’un espace d’au moins un Téra ou plus, afin de stocker les données.

### Les interfaces avec les logiciels

Version des différents outils en interaction : nom, numéro de version, source.

Nous préférons l’usage des outils open source suivants :

* Partie serveur :
  + PHP ou Nodejs , Sprint boot ……,
  + Doctrine dans sa dernière version stable,
* Partie client
  + HTML/CSS
  + JavaScript/Type Script
  + Bootstrap 5
  + Angular 16

Nous sommes ouverts à toute proposition.

### Les interfaces de communication

Le protocole http doit être utilisé pour les communications client/serveur. Il est demandé d’ajouter une couche de chiffrement avec le protocole TLS dans sa dernière version, sur l’ensemble du site.

Un serveur de cache http comme Varnish doit être utilisé pour optimiser au maximum les temps de réponse. L’usage des ESI pourra être envisagé.

### Les activités

#### Activité initiée par un utilisateur

Nous considérons tout type d’utilisateurs dans ce paragraphe.

Il est possible pour les utilisateurs :

* D’ajouter, modifier, lire et supprimer des utilisateurs.
* De s’inscrire et se désinscrire de l’ application.
* De payer et recevoir de l’argent
* De contacter les administrateurs.

#### Activité automatisée

Les activités automatisées sont l’ensemble des actions effectuées par la machine de manière périodique.

La liste de ces activités automatisées est :

* La génération d’un e-mail avec l’ensemble des statistiques sur la journée, la semaine et le mois.
* L’archivage des activités de plus de 30 jours.
* Calcule des recettes.
* Notification en de nos dépôts de recette des chauffeurs

#### Activité asynchrone

Les activités automatisées sont l’ensemble des actions effectuées par la machine de manière périodique.

La liste de ces activités automatisées est :

* La validation d’une recette.
* Notification en cas de non versement de salaire.

#### Activité de sauvegarde et de restauration

Les éléments suivants doivent être sauvegardés selon une fréquence définie.

**Éléments Fréquence Type**

Base de données toutes les 4 heures incrémentale  
Système de fichiers toutes les 4 heures incrémentale  
Code source à chaque modification incrémentale  
Système tous les jours totale

La sauvegarde de la base de données doit être cohérente avec la sauvegarde du système de fichier.

### Les exigences d’adaptation aux sites

Le site doit être déployable sur trois environnements :

* Test : où des données pourront être chargées et enlevées autant de fois que nécessaire.
* Préproduction : cet environnement est identique à l’environnement de production, mais il est coupé du public.
* Production : cet environnement est public. C’est l’application en mode « live ».

## Fonctions

Cette partie résume les fonctions principales que l’application doit exécuter.

Notre application peut utiliser une application tierce pour l’envoi d’e-mails.

Il est envisageable de récupérer les données via une API.

## Caractéristiques des utilisateurs

Une étude de marché a été transférée par e-mail. Cette étude définit que le cœur de cible de l’application est un homme ou Femme entre 16 et 70 ans.

Les administrateurs seront eux des personnes ayant des qualifications techniques nécessaires pour la maintenance de l’application.

## Contraintes

Les contraintes suivantes sont à prendre en compte dans la réponse au cahier des charges :

* Le temps de restitution des sauvegardes doit être inférieur à 30 minutes.
  + Une solution de machine dupliquée peut être envisagée.
* La charte de sécurité doit être respectée : faille xss, injection sql, faille RFD, etc.
* Limites imposées par le matériel.
  + Les traitements automatisés ne doivent pas empêcher la bonne marche du site.
* Interfaces avec les autres applications.
  + Si des communications vers l’extérieur sont nécessaires. La mise en place d’une API REST est fortement recommandée.
* Exploitation en parallèle.
  + L’application n’aura qu’une seule instance sur un seul serveur dans un premier temps.
* Exigences relatives aux langages évolués.
  + Le développement en programmation orientée objet est demandé.
* Exigences de fiabilité.
  + L’application doit avoir un taux de disponibilité de 99.999%.
  + Les réponses doivent être fiables à 99.999%.
* Niveau d’importance de l’application.
  + L’application est grand public et risque d’être à fort trafic.

## Répartition des exigences

Il est envisageable que l’application doive être « scalable » et que l’application ne soit plus que sur un serveur dédié.

# Exigences spécifiques

## Exigences des interfaces externes

### Interfaces avec le matériel

L’écriture et la lecture dans le filesystem seront en utilisant les appels classiques. Il est envisageable que les différents outils soient hébergés sur le même système.

### Interfaces avec les logiciels

La communication avec l’outil de mail est envisagée avec des appels système « curl ».

### Interfaces de communication

Toutes les communications entrantes et sortantes doivent être :

* Chiffrées
* Identifiées
* Restreintes par IP

## Exigences fonctionnelles par utilisateur

### Visiteur

#### Suivre les objet personnelle (FO)

* **Localisation des objets volés**
* **Notifications de mouvement**
* **Historique des déplacements des objets**
* **Envoyer et recevoir l’argent**

#### Messagerie

Un visiteur doit pouvoir envoyer et recevoir des sms.

Un Visiteur doit pouvoir envoyer et recevoir de l’argent .

#### Voir le menu

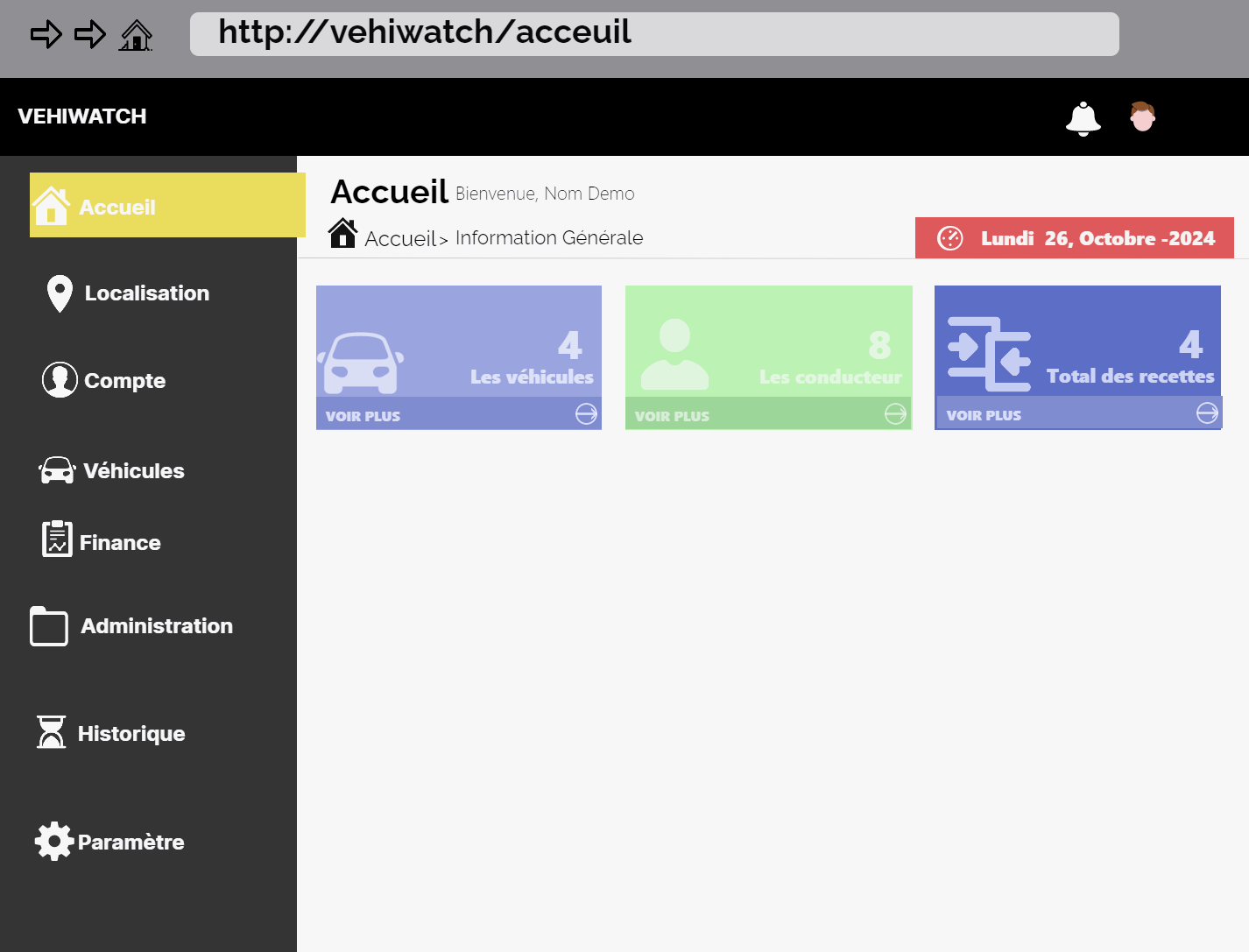
Le visiteur a accès depuis le menu aux pages suivantes :

* Accueil (voir les annonces)
* Mon compte
  + S’inscrire

### Administrateur

#### Accueil(FE)

Permetre a l’adminitrateur de voir l’ensemble de l’application.



#### Localisation

Il doit permettre a l’administrateur de suivre les déplacement.

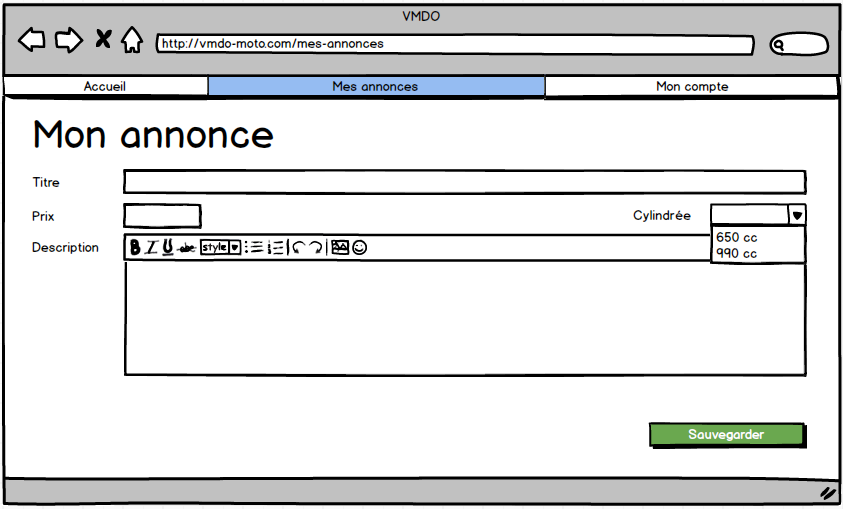
Recevoir une notification en cas dépassement de surface configurer

#### Compte

L’administrateur doit pouvoir modifier supprimer, ajouter des compte.

#### Flottes

Un utilisateur doit pouvoir saisir son annonce.



* Point d’accès : /mes-annonces/modification/{identifiant}
* Points d’entrée :
  + Titre : texte, limité à 255 caractères
  + Prix : numérique, entre 0 et 100'000, unité : euro
  + Description : texte riche, limité à 4095 caractères
  + Cylindrée : liste, à définir
* Actions possibles :
  + Sauvegarder l’annonce
* Notes :
  + Si l’utilisateur sort de la page, le message suivant s’affiche « Attention votre travail va être perdu, êtes-vous sûr de vouloir changer de page ?»
    - Si oui, alors la page demandée est chargée.
    - Si non, alors on reste sur la page de saisie avec les informations souhaitées.

#### Se désinscrire (MO)

À compléter

#### Voir le menu

Le visiteur a accès depuis le menu aux pages suivantes :

* Accueil (voir les annonces)
* Mes annonces
  + Créer une annonce
  + Gérer les annonces
* Mon compte
  + Se désinscrire

### Administrateur

#### Lire les statistiques (Admin)

À compléter

#### Générer des rapports (Admin)

* Point d’accès : /admin/rapport
* Points d’entrée :
  + Date de début – date de fin
  + Action à surveiller : liste (exemple : nombre d’annonces vues)
* Sortie :
  + Fichier CSV
    - Numéro d’annonce
    - Nombre de vue : entier
    - Créé par : nom
    - Créé le : date

#### Gestion des utilisateurs (BO)

À compléter

#### Gestion des annonces (BO)

À compléter

#### Voir le menu

Les utilisateur ont ’accès depuis le menu aux pages suivantes :

* Accueil (voir les résume de l’applications)
* Localisation
* Géolocaliser Véhicule
* Géolocaliser objet
* Compte
  + Créer un compte
  + Gérer les comptes
  + Supprimer compte
* Véhicule
* Finance
* Créer un pointage
* Confirmer un pointage
* Envoyer salaire
* Recevoir salaire
* Imprimer recette
* Administration
  + Rapports
  + Statistiques
* Historique
  + Consulter les trajet

## Exigences de performance

La liste des exigences numériques qui doit être satisfaite :

* Il faut supporter 500 utilisateurs qui s’inscrivent sur une durée de 60 minutes réparties de manière linéaire.
* L’application doit pouvoir gérer :
  + 100'000 utilisateurs,
  + 10'000 visiteurs par jour.
* Avec les conditions citées précédemment, une page doit s’afficher en 200 ms.

## Exigences logiques relatives aux bases de données

L’usage d’une base de données relationnelle est demandé. L’application doit avoir ses propres accès à la base de données. Des identifiants particuliers devront être fournis aux mainteneurs.

### Définition des entités

#### Utilisateur

Un utilisateur a :

* Un rôle liaison vers un rôle par l’identifiant
* Un identifiant unique
* Un e-mail texte
* Un nom et prénom texte
* Photo binaire
* Mot de passe texte
* Téléphone nombre

#### Rôle

La liste des rôles est :

* Utilisateur
* Conducteur
* Administrateur

Un visiteur n’est pas enregistré en base de données.

#### Annonce

Une annonce a :

* Un créateur liaison vers un utilisateur
* Un modérateur liaison vers un utilisateur
* Un titre texte
* Une cylindrée numérique (unité : cc)
* Une description texte riche

## Contraintes de conception

Nous allons utiliser les points suivants pour évaluer la qualité des développements :

* Utilisation de la norme PSR-2,
* PHP Mess Detector,
* Quelle forme normale a été utilisée ?
* Couverture de tests >70 % sur les controllers et entités.

Le prestataire doit fournir un moyen automatisé pour vérifier la conformité avec les points suivants.

### Gestion des droits

## Attributs

### Disponibilité

Le système doit être disponible à 99.999 %, 7 jours sur 7, de 04:00 à 00:00.

Il est toléré que de 00 :00 à 04 :00, le système ne soit disponible qu’à 99.9 % du temps, pour les opérations de maintenance.

### Sécurité

L’accès à l’application doit être ouvert à tous mais uniquement avec l’identifiant et le mot de passe.

L’accès à l’application de préproduction doit être réservé aux IP autorisées et identifiées.

L’accès à la machine se fera via un protocole chiffré par échange de clés.

### Maintenabilité

Le projet doit être maintenable par **Nsengimana François CEO IMANA-47** en attente de création d’une équipe. Il faut que le projet soit suffisamment documenté, aucune technologie « faite maison » ou « exotique » doit être choisi sans l’accord d’**IMANA-47**.

## Autres exigences

### Budget

Pour le projet cout du projet il est à définir.

### Dates

L’objectif est de lancer le site au 10 Janvier 2025.

Voici les dates clés :

* Réponse au cahier des charges : 15/05/2025.
* Conception et spécification : 12/11/2024 - 04/12/2024
* Développement : 20/12/2024 - 30/12/2024
* Lancement de la campagne marketing : Encours

### Les livrables attendus

* Validation du cahier des charges
* Contrats
* Spécifications techniques et fonctionnelles
* Code source
* Documentations
  + Utilisateurs, techniques
  + Statistiques
    - Connexion, utilisation, conversion, etc.
    - Rapport d'analyse (automatisé)
* Tests, qualité

### Propriétés intellectuelles

Le code source du site, son contenu (médias et texte), la base de données appartiendront à IMANA-47.

# Annexes

## Disposition du site

### Une annonce détaillée

À compléter

### Une liste d’informations

À compléter

### Un formulaire

À compléter

## Charte éditoriale

À compléter

## Charte graphique

À compléter

AAID acr. Désigne le numéro d’identification crypté unique de l’appareil, attribué par le système d’exploitation Android. Il permet aux développeurs/responsables marketing d’effectuer le suivi de certaines activités à des fins publicitaires et aux annonceurs de garder une trace des conversions d’achat et de téléchargement

A/B Testing n.m. Le test A/B est une démarche qui consiste à soumettre aux utilisateurs des versions différentes de l’application ou du site web à étudier (variation d’une couleur, disposition d’un bandeau…). AAID acr. Désigne le numéro d’identification crypté unique de l’appareil, attribué par le système d’exploitation Android. Il permet aux développeurs/responsables marketing d’effectuer le suivi de certaines activités à des fins publicitaires et aux annonceurs de garder une trace des conversions d’achat et de téléchargement. Recherche : Adresse IP n.f. Numéro qui identifie chaque appareil connecté à un réseau utilisant le protocole Internet (Internet Protocol). Android n.m. Système d’exploitation mobile fondé sur le noyau Linux et développé actuellement par Google. API abr. Application Programming Interface. Une API est une interface de programmation d’application permettant à des applications de communiquer entre elles et de s’échanger mutuellement des données.

[francknsengimana@gmail.com](mailto:francknsengimana@gmail.com)



[françois -nsengimana](https://www.linkedin.com/in/fran%C3%A7ois-nsengimana)



[IMANA47](https://github.com/IMANA47/)



Brazzaville, République du Congo



*+242 06 948 51 54*



***CONTACT***



***François***

***nsengimana***

**Concepteur de l’application**