

# **EXPERTS SOLUTIONS**

***Accompagnement à l'Innovation Technologique***

IFU : 3 201 642 040 214 – Courriel : [fanickarthis@experts-solutions.com](mailto:fanickarthis@experts-solutions.com)

Tél. : +1 418 455 7122 +229 96 157 152Qtier AHOGBOHOUE, Ilot 3100 – BP 8094 Cotonou/BÉNIN



## **RAPPORT FINAL**

Présentation

# **APPLICATION MOBILE**

## Table des matières

<b>I. RAPPEL.....</b>	<b>5</b>
<b>1. Contexte et justifications.....</b>	<b>5</b>
1.1. Objectif.....	5
1.2. Résultats attendu.....	5
<b>II. ARCHITECTURE APPLICATIVE.....</b>	<b>6</b>
<b>1. Architecture orientée microservices.....</b>	<b>6</b>
<b>III. ARCHITECTURE INFRASTRUCTURE (Hors cible).....</b>	<b>8</b>
<b>1. Inventaire des technologies.....</b>	<b>9</b>
1.1. Application mobile.....	9
1.1. Portail web et le portail des services en ligne.....	10
1.1. Backend services.....	11
<b>2. Codes sources.....</b>	<b>12</b>
<b>IV. PRESENTATION DU BACKEND SERVICES.....</b>	<b>12</b>
<b>1. Fonctionnalités.....</b>	<b>12</b>
1.1 Intégration avec le modele de données existant.....	13
1.2 Authentification des utilisateurs.....	13
1.3 Rôles et permissions.....	14
1.4 Inventaire des services customs développés.....	16
<b>V. GUIDE APPLICATION MOBILE.....</b>	<b>18</b>
<b>1. Présentation des acteurs qui peuvent interagir avec l'application mobiles.....</b>	<b>18</b>
1.1 Comment installer l'application mobile.....	19
1.2 Pages d'introduction de l'application mobile.....	20
1.3 Page démarrage.....	21
1.4 Page de création d'un compte utilisateur.....	22
1.5 Page de connexion.....	24
1.6 Page d'accueil de l'application mobile.....	25
1.7 Vérifiez si la connexion au serveur est disponible.....	26
1.8 Informations sur l'assuré.....	27

<b>1.9 Consulter les informations de l'employeur.....</b>	<b>.28</b>
<b>1.10 Consulter les informations sur les droits prestation.....</b>	<b>.29</b>
<b>1.11 Consulter les informations du bénéficiaires.....</b>	<b>.32</b>
<b>1.12 Page réservée au employeur de la CNSS.....</b>	<b>.33</b>
<b>1.13 Scan de code Qr.....</b>	<b>.35</b>
<b>1.14 Profile utilisateur.....</b>	<b>.36</b>

## FIGURES

Figure 1: Architecture à haut niveau.....	.6
Figure 2: Architecture cible de l'infrastructure.....	.8
Figure 3: Intégration avec les différentes tables de l'application métier.....	.13
Figure 4: Documentation swagger des méthodes d'authentification.....	.14
Figure 5: Documentation swagger des méthodes d'autorisation.....	.15
Figure 6: Assignment de rôle dans l'outils d'administration.....	.16
Figure 7: Fonctionnement de l'application mobile.....	.18
Figure 8: Page d'introduction.....	.20
Figure 9: Page de démarrage.....	.21
Figure 10: Page de création de compte utilisateur.....	.23
Figure 11: Page de connexion à l'application mobile.....	.24
Figure 12: Page d'accueil de l'application mobile.....	.25
Figure 13: Synchronisation avec le serveur.....	.26
Figure 14: Consultation des informations de l'assuré.....	.27
Figure 15: Consultation des informations de l'employeur.....	.28
Figure 16: Consultation des informations du droit prestation.....	.29
Figure 17: Section réservée aux employers de la CNSS RCA.....	.33
Figure 18: Recherche et scannage de document via Qr code.....	.34
Figure 19: Scan de code QR via la camera du téléphone.....	.35
Figure 20: Consultation du profil utilisateur.....	.36
Figure 21: Modification du profil utilisateur.....	.37

# I. RAPPEL

## 1. Contexte et justifications

La Caisse Nationale de Sécurité Sociale de la République Centrafricaine, dans l'exploitation de son application métier, fait face à un défi majeur, celui de pouvoir vérifier l'authenticité des documents émis par le système aux différents usagers et partenaires de la CNSS RCA. Il a donc été décidé, de mettre en œuvre une application permettant au personnel et aux partenaires autorisés, d'être capables de vérifier les informations sur les documents présentés par les usagers, tels que les bons de caisse et les notifications avec possibilité de consulter les dossiers des assurés avec une application mobile et via une page web.

### 1.1. Objectif

L'objectif de ce projet, est de déployer pour le compte de la CNSS RCA, une application mobile offrant la fonctionnalité de vérification par code QR, et un service REST API accessible par les partenaires de la CNSS RCA. Aussi permettre au usagers de la CNSS RCA de se connecter au dossier assuré et employeur via un site web. L'application mobile et le site web devront partager les mêmes source de données.

### 1.2. Résultats attendu

Au terme du projet, la CNSS RCA doit disposer d'une application fonctionnelle, accessible aux personnels et partenaires autorisés via mobile et un portail web. Elle doit vérifier en temps réel les documents émis par l'application SIPPRESS ou l'application métier hébergeant les données nécessaires à l'authentification.

## II. ARCHITECTURE APPLICATIVE

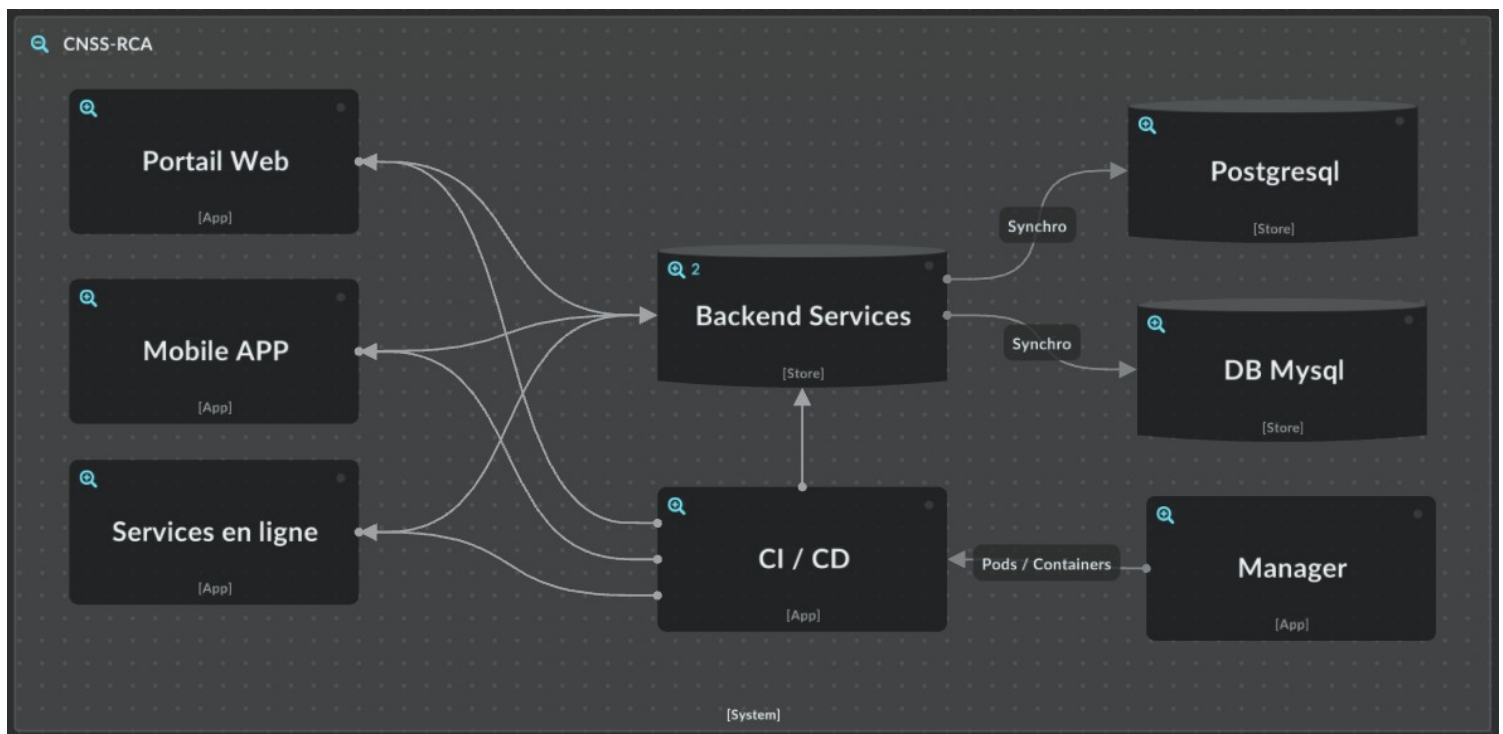


Figure 1: Architecture à haut niveau

### 1. Architecture orientée microservices

L'architecture microservices est une approche de développement de logiciels où une application est construite comme un ensemble de services plus petits et indépendants, qui communiquent entre eux via des interfaces bien définies, souvent des API HTTP. Chaque microservice est conçu pour effectuer une fonction spécifique et peut être développé, déployé, exploité et mis à jour indépendamment des autres services. Voici quelques caractéristiques clés de l'architecture microservices :

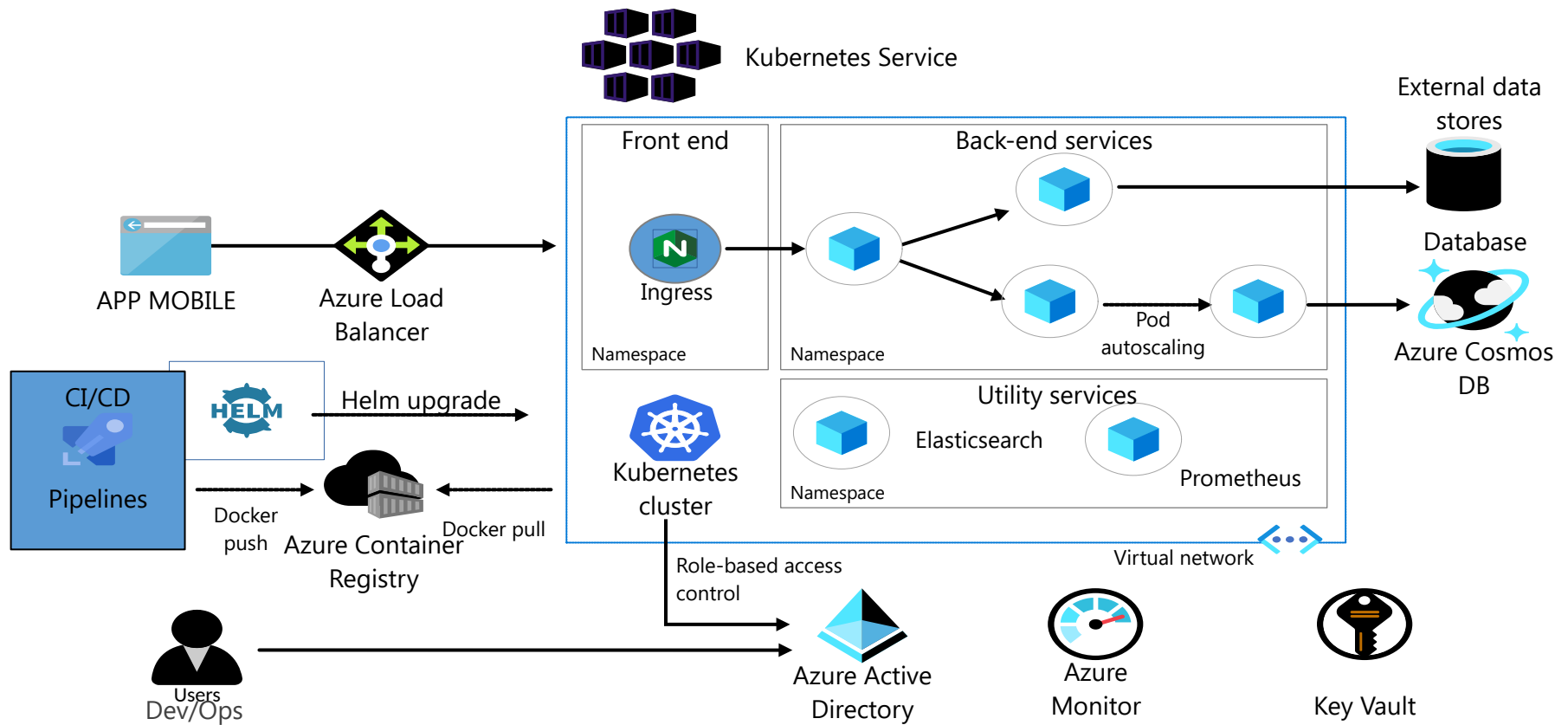
- **Modularité** : Les applications sont divisées en petits services autonomes, ce qui facilite la gestion, le développement et la compréhension de chaque composant séparément.
- **Indépendance** : Chaque microservice peut être développé et déployé indépendamment des autres, permettant des mises à jour plus fréquentes et spécifiques.
- **Diversité technologique** : Les microservices peuvent être écrits dans différents langages de programmation et utiliser différentes technologies de stockage de données, selon les besoins spécifiques de chaque service.

- **Résilience** : Les défaillances dans un microservice spécifique n'affectent généralement pas l'ensemble de l'application, permettant une meilleure gestion des erreurs et une disponibilité accrue.
- **Scalabilité** : Les services peuvent être scalés de manière indépendante, ce qui permet une allocation plus efficace des ressources en fonction de la demande pour des fonctionnalités spécifiques.

L'adoption d'une architecture microservices peut apporter flexibilité et évolutivité aux applications, mais elle introduit également de la complexité en termes de gestion des multiples services et de leur communication, en plus de nécessiter des outils sophistiqués pour la surveillance, la mise en réseau et la sécurité.

### III. ARCHITECTURE INFRASTRUCTURE (Hors cible)

Figure 2: Architecture cible de l'infrastructure





## 1. Inventaire des technologies

### 1.1. Application mobile

L'application mobile est développée avec Flutter est un framework de développement d'applications mobiles, web et de bureau open source créé par Google. Il permet aux développeurs de construire des interfaces utilisateurs (UI) attrayantes et performantes avec un seul codebase en Dart, un langage de programmation également développé par Google. Voici quelques points clés à propos de Flutter :

**Cross-platform** : Flutter permet de développer des applications pour les systèmes d'exploitation iOS et Android, ainsi que pour le web et le bureau à partir d'un seul code source. Cela simplifie le processus de développement et réduit les coûts et les efforts nécessaires pour maintenir plusieurs codebases.

**Widgets** : L'un des concepts clés de Flutter est son utilisation extensive de widgets. L'interface utilisateur dans Flutter est entièrement composée de widgets, depuis les éléments de base comme les boutons et les textes, jusqu'aux éléments complexes comme les animations et les transitions. Les widgets peuvent être imbriqués pour créer des interfaces complexes et personnalisées.

**Performance** : Flutter compile le code source en code ARM natif ou en JavaScript, selon la plateforme cible, ce qui améliore les performances. Cela signifie que les applications Flutter peuvent souvent atteindre des performances de premier plan, rivalisant avec celles des applications natives.

**Hot Reload** : L'une des fonctionnalités les plus appréciées de Flutter est le "Hot Reload", qui permet aux développeurs de voir immédiatement les modifications apportées au code dans l'application en cours d'exécution, sans nécessiter un redémarrage complet de l'application. Cela accélère le cycle de développement.

**Riche en fonctionnalités** : Flutter offre un large éventail de fonctionnalités prêtes à l'emploi, telles que la gestion des gestes, les animations, le rendu graphique 2D, les tests automatisés, etc. De plus, il dispose d'une large bibliothèque de packages tiers qui étend encore plus ses capacités.

**Communauté et support** : Étant un projet open source soutenu par Google, Flutter bénéficie d'une communauté de développement active et en croissance rapide. La documentation, les tutoriels, et le support communautaire sont largement disponibles pour aider les développeurs à résoudre les problèmes et à apprendre à utiliser le framework efficacement.

Flutter est ainsi une option puissante et flexible pour le développement d'applications multiplateformes, offrant à la fois haute performance et efficacité de développement.

Ce choix nous permet de builder à la fois une application IOS et Android.

## 1.1. Portail web et le portail des services en ligne

Ces deux applications sont développées avec Angular qui est un framework de développement d'applications web robuste et open source, maintenu par Google. Il est conçu pour faciliter la création et le maintien d'applications web complexes et évolutives. Voici quelques aspects clés d'Angular :

**Architecture basée sur les composants** : Angular structure les applications en une hiérarchie de composants, chacun représentant une petite partie de l'interface utilisateur. Chaque composant peut contenir sa propre logique et son propre état, rendant le code plus modulaire et facile à réutiliser.

1. **TypeScript** : Angular est construit sur TypeScript, une surcouche de JavaScript qui ajoute des types statiques. Cela aide à détecter des erreurs plus tôt dans le cycle de développement, améliore la lisibilité du code et rend le développement plus prévisible et plus facile à gérer.
2. **Binding bidirectionnel** : Angular facilite la synchronisation des données entre le modèle et la vue avec son système de binding bidirectionnel. Cela signifie que les modifications apportées aux données dans l'interface utilisateur peuvent automatiquement mettre à jour le modèle et vice-versa.
3. **Injection de dépendances** : Angular utilise un puissant système d'injection de dépendances pour fournir les dépendances requises à différents composants et services dans l'application. Cela améliore la modularité, la testabilité et la réutilisation du code.
4. **Directives** : Les directives sont des instructions utilisées pour manipuler le DOM de manière dynamique. Angular inclut plusieurs directives intégrées et permet également aux développeurs de créer leurs propres directives personnalisées pour appliquer un comportement spécifique aux éléments DOM.
5. **Services et modèles** : Angular encourage la séparation des préoccupations en isolant la logique métier loin des composants de l'interface utilisateur dans des services, qui peuvent être injectés là où ils sont nécessaires. Cela facilite le partage de code et la gestion de l'état.
6. **Angular CLI** : Angular inclut une interface en ligne de commande (CLI) qui simplifie les tâches de développement telles que la création d'un nouveau projet, l'ajout de composants, la gestion des dépendances, et l'exécution de tests. La CLI peut également aider à optimiser l'application pour la production.
7. **Routage et navigation** : Angular fournit un service de routage sophistiqué qui permet de gérer la navigation complexe dans l'application, le chargement paresseux des modules, et la protection des routes.
8. **Forms et validation** : Angular offre des outils pour gérer les entrées de formulaire et la validation côté client, ce qui rend la collecte et la vérification des données utilisateur plus facile et plus robuste.

Angular est particulièrement adapté pour les grandes applications d'entreprise ou les applications nécessitant une gestion rigoureuse de l'état et une organisation claire. Il bénéficie également d'un large écosystème, d'une communauté active, et d'un ensemble riche de fonctionnalités qui permettent de créer des applications web interactives et dynamiques.

## 1.1. Backend services

Le service backend est développé avec Strapi est un système de gestion de contenu (CMS) open source basé sur Node.js. Il permet aux développeurs de créer facilement des API personnalisées en quelques clics et de gérer le contenu de manière flexible. Voici quelques points clés à propos de Strapi :

1. **Headless CMS** : Strapi est un CMS "headless", ce qui signifie qu'il fournit uniquement l'arrière-plan (back-end) de la gestion de contenu, sans présenter une façade (front-end) définie. Cela permet aux développeurs de l'utiliser avec n'importe quelle technologie front-end (comme React, Vue.js, Angular, ou même des plateformes mobiles) pour afficher le contenu.
2. **API-Based** : Strapi est conçu pour construire des API REST ou GraphQL. Cela permet une intégration facile avec diverses technologies front-end et facilite l'accès au contenu via des requêtes API standard.
3. **Personnalisable** : Strapi offre une grande flexibilité pour personnaliser les types de contenu. Les développeurs peuvent définir leurs propres types de contenu (modèles), ajouter des champs personnalisés, et configurer des relations complexes entre les données sans avoir besoin d'écrire beaucoup de code.
4. **Auto-génération d'API** : À partir des modèles de contenu définis, Strapi génère automatiquement des API qui permettent de créer, lire, mettre à jour, et supprimer des informations (opérations CRUD). Cela accélère considérablement le développement d'applications nécessitant des opérations de gestion de contenu.
5. **Sécurité** : Strapi intègre des fonctionnalités de sécurité telles que l'authentification, les permissions et les rôles par défaut. Les développeurs peuvent configurer ces paramètres pour contrôler précisément l'accès aux différents types de contenu et aux opérations API.
6. **Plugins et extensions** : Strapi peut être étendu avec une variété de plugins, qui peuvent ajouter des fonctionnalités comme le support pour le SEO, la gestion des médias, le marketing par email, etc. Les développeurs peuvent également créer leurs propres plugins pour répondre à des besoins spécifiques.
7. **Communauté et écosystème** : Étant un projet open source, Strapi bénéficie d'une communauté active de développeurs qui contribuent régulièrement au projet. Il y a aussi un marché croissant de plugins et d'intégrations développés par la communauté.
8. **Facilité d'utilisation** : Strapi offre une interface d'administration intuitive qui permet aux utilisateurs non techniques de gérer efficacement le contenu. Cela inclut la gestion des utilisateurs, la publication de contenu, et la configuration des paramètres du site.

Strapi est particulièrement adapté pour les projets nécessitant une gestion de contenu personnalisée et flexible, tout en utilisant des technologies modernes pour le front-end. Il est apprécié pour sa facilité de développement, sa personnalisation et sa capacité à s'intégrer de manière transparente dans des écosystèmes de développement divers.

## 2. Codes sources

L'ensemble des codes sources des différentes applications sont hébergées sur github. Une autorisation est nécessaire pour accéder aux code source.

Application	Repos Github	Commentaires
Mobile	<a href="https://github.com/cnss-rca/mobile">https://github.com/cnss-rca/mobile</a>	Application mobile déployée sur google play
Admin panel	<a href="https://github.com/cnss-rca/admin-panel">https://github.com/cnss-rca/admin-panel</a>	Portail services en ligne
Portail	<a href="https://github.com/cnss-rca/portail">https://github.com/cnss-rca/portail</a>	Portail web
Backend	<a href="https://github.com/cnss-rca/api-server">https://github.com/cnss-rca/api-server</a>	Application backend
ci	<a href="https://github.com/cnss-rca/ci">https://github.com/cnss-rca/ci</a>	Permet de builder l'ensemble de images docker des différentes applications

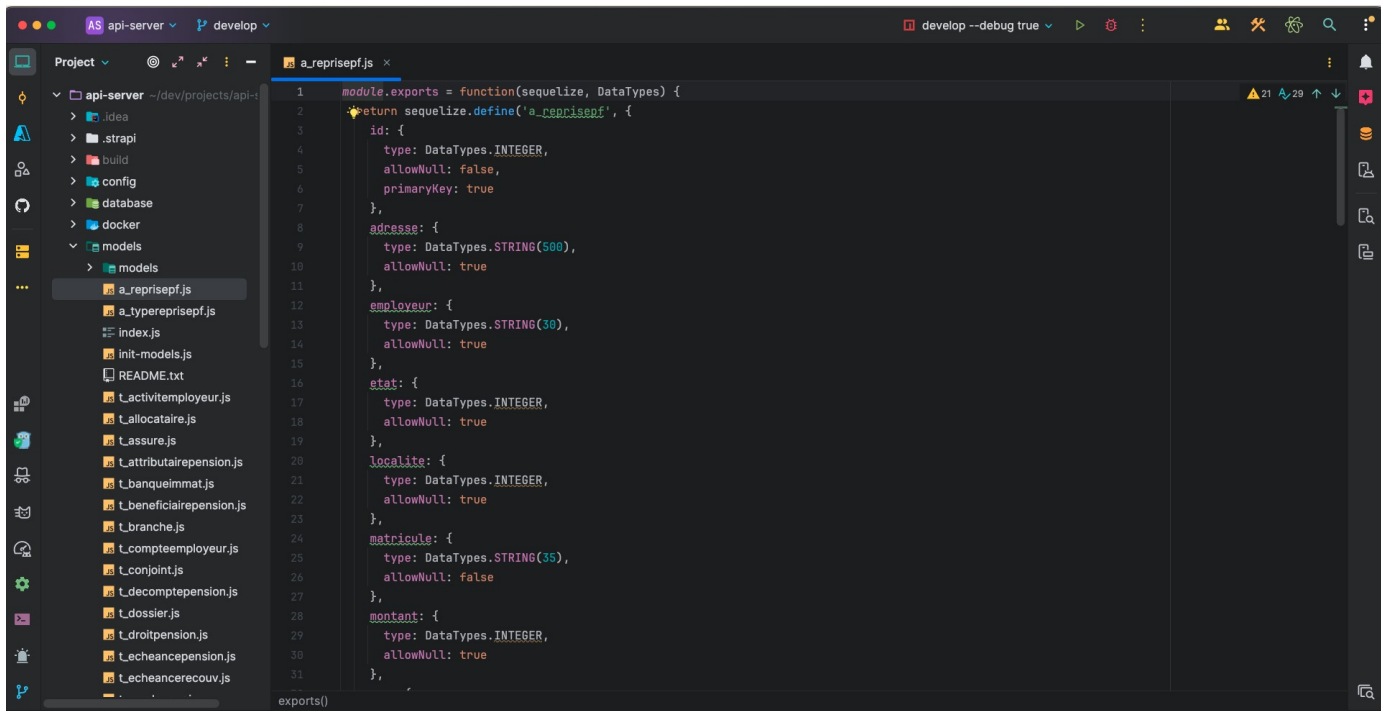
## IV. PRESENTATION DU BACKEND SERVICES

### 1. Fonctionnalités

Le backend service permet d'interfacer un service api avec les différents systèmes de la CNSS, accéder aux données présentes dans une base de données par exemple. Il permet de :

- Faire une abstraction, ce qui évite aux différents clients d'intervenir directement sur les données.
- Permet d'exposer des services à l'ensemble des applications
- Garantie la sécurité des données avec les mécanisme d'authentification d'entreprise tels que le SSO et le OAuth2
- Expose un REST API sécurisé pour les clients
- Centralise l'administration des données

## 1.1 Intégration avec le modele de données existant




*Figure 3: Intégration avec les différentes tables de l'application métier*

Le mapping est effectué du côté backend en utilisant le modèle de données existant de la CNSS. Il est prévu que ce modèle évolue en fonction des besoins. Cela nous a permis de prendre en considération les tables nécessaires pour l'application mobile et les services en ligne.

## 1.2 Authentification des utilisateurs

Toutes les fonctions d'authentification sont implémentées dans l'api. Elle supporte le Oauth2 et le SSO. Une documentation accessible via Swagger UI. Swagger UI est un outil qui génère une page web interactive à partir d'une spécification OpenAPI. Cette interface utilisateur permet aux développeurs et aux autres parties prenantes de visualiser et d'interagir avec l'API sans aucune implémentation logicielle. Elle fournit une méthode pratique pour tester les endpoints de l'API directement dans le navigateur.

 **Swagger**  
Powered by SMARTBEAR

Explore

# DOCUMENTATION

1.0.0 OAS3

[Terms of service](#)  
[TEAM - Website](#)  
[Send email to TEAM](#)  
[Apache 2.0](#)  
[Find out more](#)

Servers

http://0.0.0.0:1337/api - Development server

Authorize

## Users-Permissions - Auth

Authentication endpoints

Find out more

GET	/connect/{provider}	Login with a provider	
POST	/auth/local	Local login	
POST	/auth/local/register	Register a user	
GET	/auth/{provider}/callback	Default Callback from provider auth	
POST	/auth/forgot-password	Send rest password email	
POST	/auth/reset-password	Rest user password	
POST	/auth/change-password	Update user's own password	
GET	/auth/email-confirmation	Confirm user email	
POST	/auth/send-email-confirmation	Send confirmation email	

Figure 4: Documentation swagger des méthodes d'authentification

### 1.3 Rôles et permissions

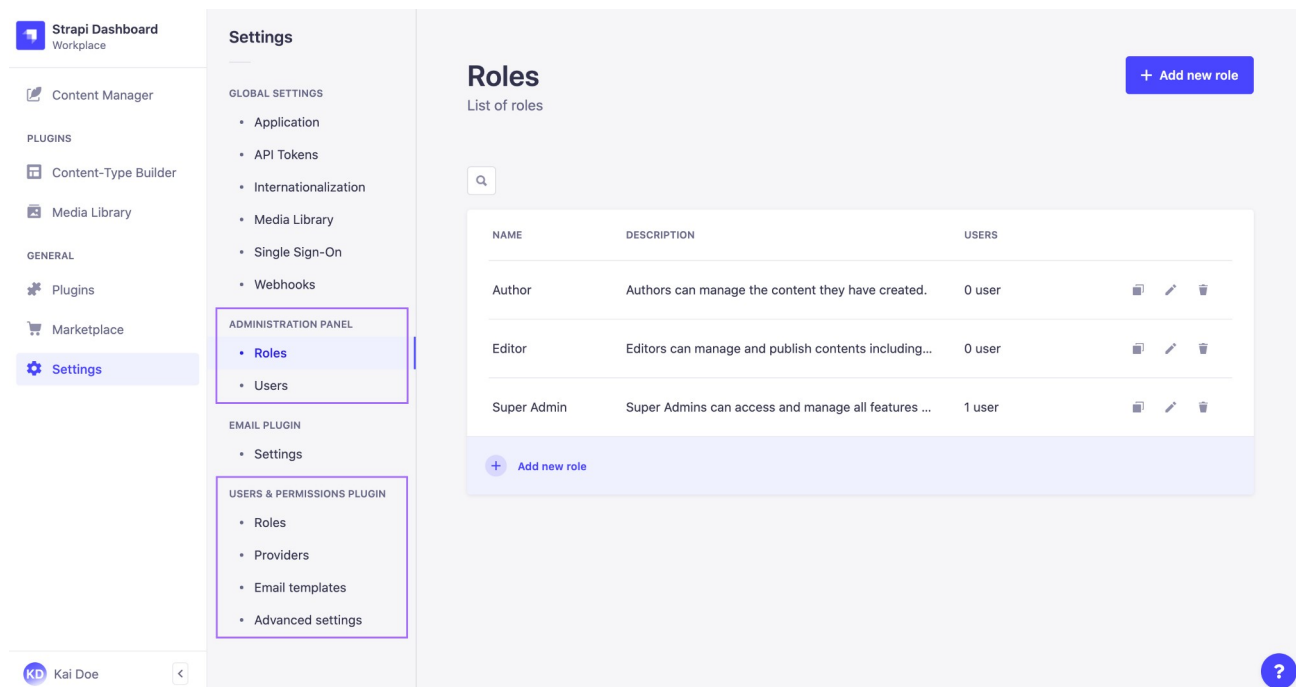
Un modele d'accès doit être défini pour les administrateurs de la plate-forme. Il permettra de cadre les activités. Certaines fonctionnalités du panneau d'administration, ainsi que le contenu géré avec Strapi lui-même, sont régies par un système d'autorisations. Ces autorisations peuvent être attribuées aux rôles, qui sont associés aux utilisateurs qui ont accès au panneau

d'administration, les administrateurs. Mais il est également possible d'accorder des autorisations plus publiquement, de donner accès au contenu aux utilisateurs finaux de votre application Strapi.

En fonction des utilisateurs et de leurs rôles et autorisations que vous souhaitez gérer, vous devez utiliser la fonctionnalité RBAC (Role Based Access Control) ou le plugin Users & Permissions. Les deux sont gérés à partir de l'icône Paramètres, accessible à partir de la navigation principale du panneau d'administration.

Users-Permissions - Users & Roles			Users, roles, and permissions endpoints	<a href="#">Find out more</a>
GET	/users-permissions/permissions	Get default generated permissions	✓	🔒
GET	/users-permissions/roles	List roles	✓	🔒
POST	/users-permissions/roles	Create a role	✓	🔒
GET	/users-permissions/roles/{id}	Get a role	✓	🔒
PUT	/users-permissions/roles/{role}	Update a role	✓	🔒
DELETE	/users-permissions/roles/{role}	Delete a role	✓	🔒
GET	/users	Get list of users	✓	🔒
POST	/users	Create a user	✓	🔒
GET	/users/{id}	Get a user	✓	🔒
PUT	/users/{id}	Update a user	✓	🔒
DELETE	/users/{id}	Delete a user	✓	🔒
GET	/users/me	Get authenticated user info	✓	🔒
GET	/users/count	Get user count	✓	🔒

Figure 5: Documentation swagger des méthodes d'autorisation



**Figure 6: Assignment de rôle dans l'outils d'administration**

Pour plus de détails sur l'outils d'administration voir la documentation officielle de strapi ici <https://docs.strapi.io/dev-docs/intro>

## 1.4 Inventaire des services customs développés

Les services customs ne sont rien d'autre que des fonctionnalités développées qui seront accessibles à d'autres applications via service REST API.

URI	Description
/cnss/assure/matricule	Permet de récupérer un assuré à partir de son matricule
/cnss/attributaire/matricule	Permet de récupérer un attributaire à partir de son matricule
/cnss/beneficiaire	Permet de récupérer un bénéficiaire à partir de son matricule
/cnss/compte/employeur	Permet de récupérer un compte employeur à partir de son id



/cnss/employeur/matricule	Permet de récupérer un employeur à partir de son id
/users/pension	Récupération des informations de pension d'un assuré
/cnss/reprise/matricule	

## V. GUIDE APPLICATION MOBILE

Ce schéma illustre comment fonction l'application mobile les différents acteurs émettent leurs requêtes via l'application mobile. Les appels REST API sont effectués vers le serveur backend par l'application. Le backend assure le traitement de la requête et la réponse à l'application mobile.

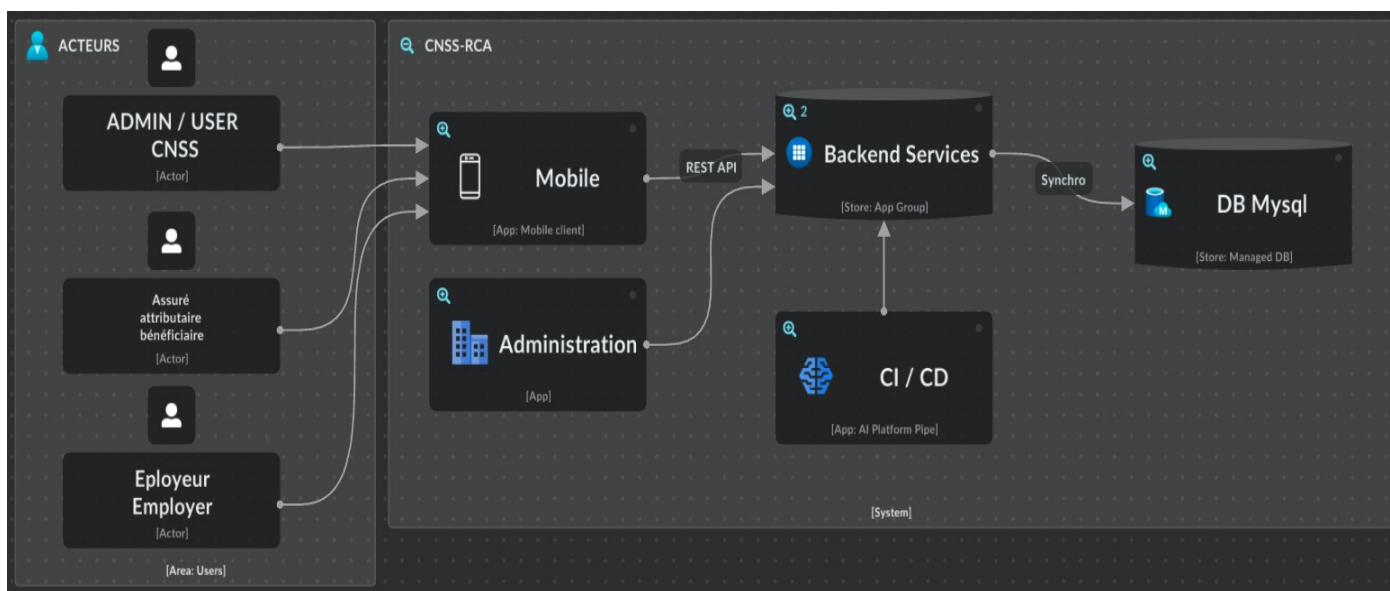


Figure 7: Fonctionnement de l'application mobile

### 1. Présentation des acteurs qui peuvent interagir avec l'application mobiles

Acteurs	Type	Rôles
ADMIN / USER CNSS	internes	Ces acteurs sont obligatoirement des agents de la cnss <ul style="list-style-type: none"> <li>Gestion des comptes des utilisateurs</li> <li>Scan de documents</li> <li>Recherche d'informations</li> </ul>
Assuré Attributaire Bénéficiaire	externes	<ul style="list-style-type: none"> <li>Créer / mettre à jour un profile</li> <li>consulter des informations d'assuré, de bénéficiaire et d'attributaire</li> </ul>
Employeur Employer	externes	<ul style="list-style-type: none"> <li>Créer / mettre à jour un profile</li> <li>consulter des informations d'employeur</li> </ul>

## 1.1 Comment installer l'application mobile

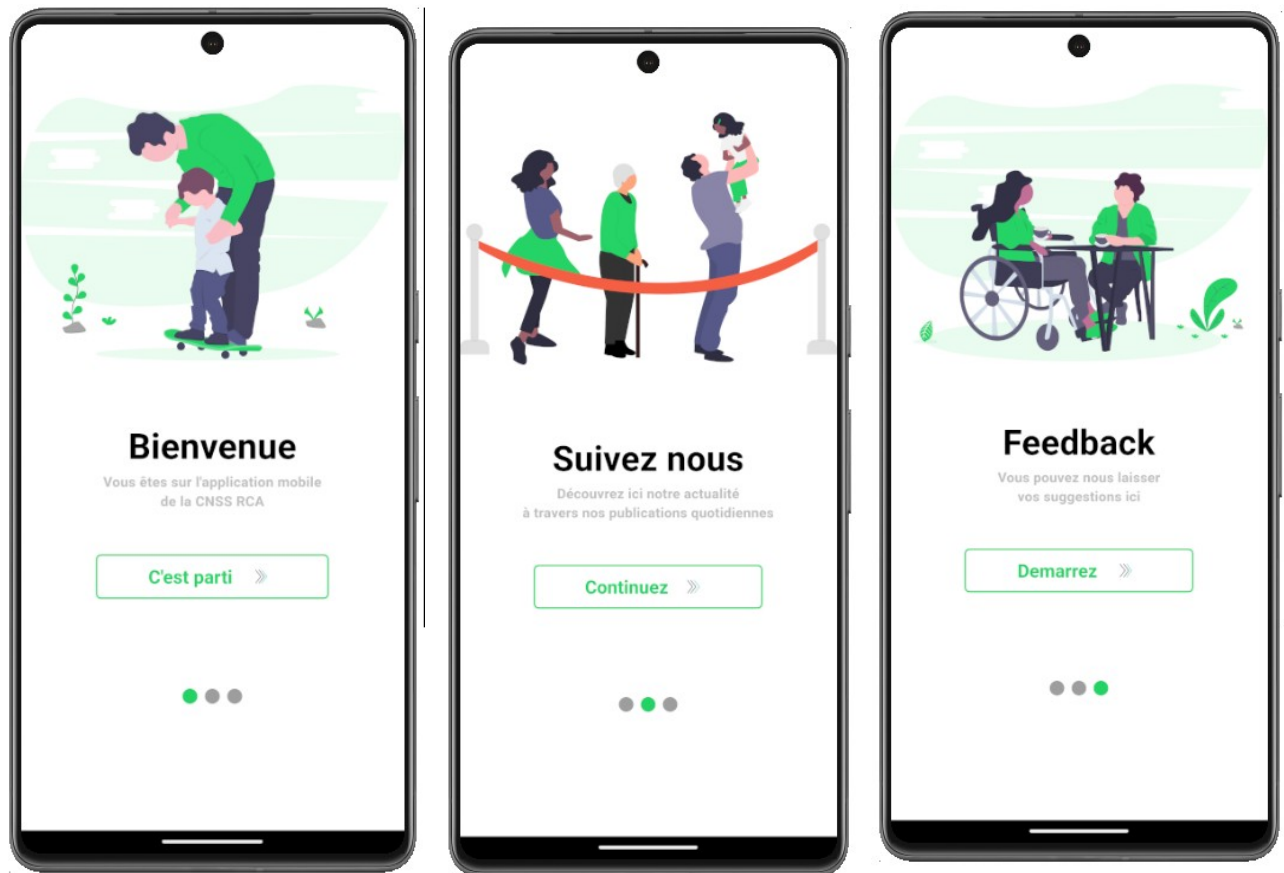
L'application mobile est du type hybride. Elle est conçue pour fonctionner sur plusieurs plateformes mobiles (comme ios et Android) en utilisant un seul code base. Cela signifie que les développeurs peuvent écrire le code une fois et le déployer sur plusieurs plateformes, ce qui réduit le temps et les coûts de développement.

Pour des besoins de démonstration une version bêta sera disponible en téléchargement sur le repositore github de la CNSS. Le déploiement sur Google play et app store nécessitent des autorisations de la CNSS.

Il vous suffit de cliquer sur le lien de téléchargement et de suivre la procédure d'installation.

OS	Lien de téléchargement	Paramètres de connexion
Android	<a href="https://github.com/fanick/cnss-privacy/releases/download/v1.0.0/app-release.apk">https://github.com/fanick/cnss-privacy/releases/download/v1.0.0/app-release.apk</a>	Login : <a href="mailto:test@gmail.com">test@gmail.com</a> Password : Admin@123#

## 1.2 Pages d'introduction de l'application mobile



*Figure 8: Page d'introduction*

La page d'introduction, souvent appelée "splash screen" ou écran d'accueil, sert plusieurs objectifs :

Présenter brièvement l'application et ses fonctionnalités.

- Donner une première impression visuelle attrayante qui reflète la marque.
- Charger les ressources initiales ou effectuer des configurations préliminaires pendant que l'utilisateur regarde l'écran.

La mise à jour de la page d'introduction peut être effectuée en fonction des besoins de communication de la CNSS RCA.

### 1.3 Page démarrage

Cette page offre à l'utilisateur la possibilité de créer un compte utilisateur ou de se connecter s'il a déjà un compte.

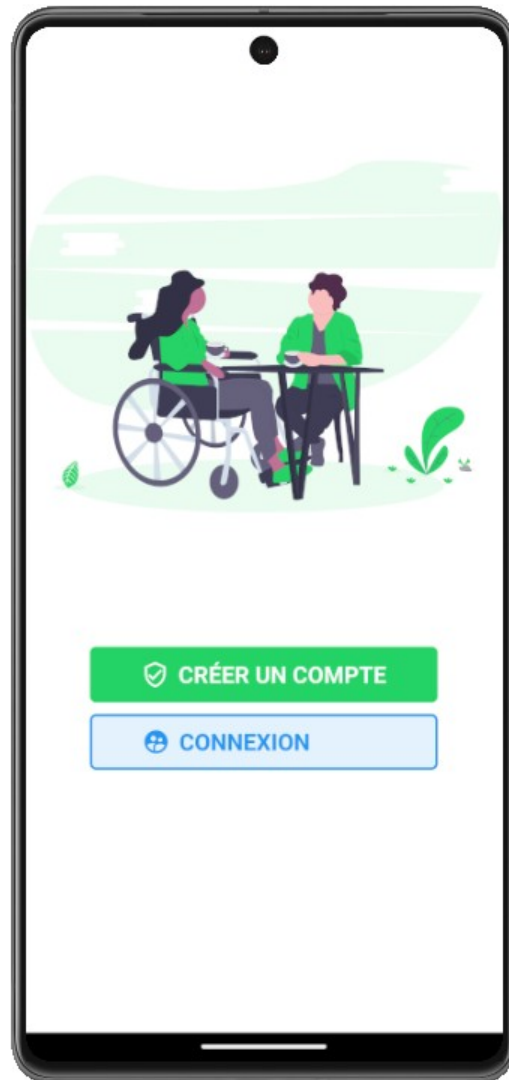


Figure 9: Page de démarrage

## 1.4 Page de création d'un compte utilisateur

Les utilisateurs ont la possibilité de créer leur compte sur cette page. Ce compte n'est pas activé par défaut. Un agent de la CNSS est requis pour valider les informations saisies et activer le compte. Seuls les comptes actifs sont autorisés à se connecter dans l'application.

Champs	Description
Username	Le nom d'utilisateur. Représente un nom utilisateur qu'on associe plus tard au profile
email	Doit être une adresse valide et unique . Elle ne pourra être associé à un seul compte
Password	Un mot de passe complexe constitué de majuscule, de minuscule et de caractères speciaux
Numéro assuré	Numéro de l'assuré dont on veut consulter les données. Doit être valide dans l'application métier de la CNSS
Numéro bénéficiaire	Numéro bénéficiaire dont on veut consulter les données.Doit être valide dans l'application métier de la CNSS
Numéro attributaire	Numéro bénéficiaire dont on veut consulter les données.Doit être valide dans l'application métier de la CNSS
Numero employeur	Numéro de l'employeur dont on veut consulter les données. Doit être valide dans l'application métier de la CNSS

12:21

Créer un compte

**CNSS**  
Caisse Nationale de Sécurité Sociale

**Créer un compte**  
Nom d'utilisateur, Email et Mot de passe

Username  
Entrez votre nom d'utilisateur

Email  
Entrez votre email

Password  
Entrez votre mot de passe

N° assuré  
Entrez le num assuré

N° attributaire  
Entrez num attributaire

N° bénéficiaire  
Entrez num bénéficiaire

N° employeur  
Entrez num employeur

8:15

Créer un compte

**Créer un compte**  
Nom d'utilisateur, Email et Mot de passe

Username  
test

Email  
test@gmail.com

Password  
.....

N° assuré  
20200001309

N° attributaire  
A\_19573\_P0

N° bénéficiaire  
B\_111330\_P0

N° employeur  
57230

Continue

Figure 10: Page de création de compte utilisateur

## 1.5 Page de connexion

Cette page permet à un utilisateur de se connecter à l'application mobile. Il aura besoin d'un compte actif pour se connecter. Lorsque les paramètres de connexion sont invalides, un message lui sera envoyé, lui interdisant l'accès à l'application.

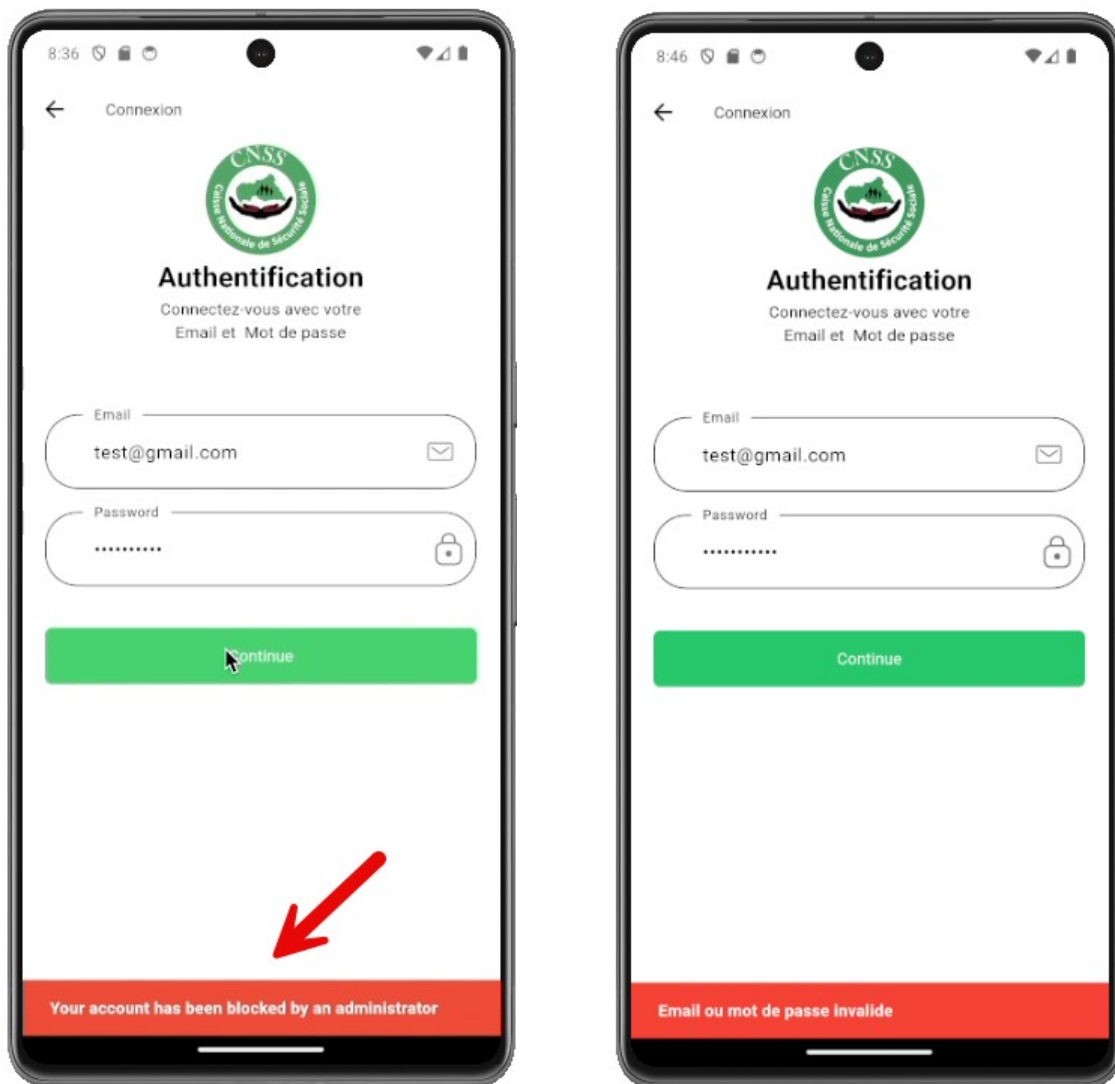


Figure 11: Page de connexion à l'application mobile



## 1.6 Page d'accueil de l'application mobile

Une fois l'utilisateur authentifié, il devrait tomber sur cetAprès l'authentification de l'utilisateur, il devrait être redirigé vers cette page qui récapitule les fonctionnalités de l'application.te page qui fait un récapitulatif des fonctionnalités de l'application.

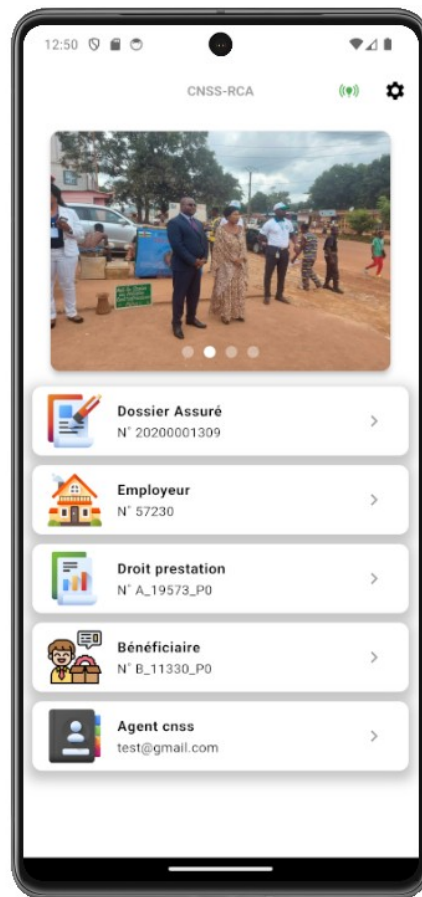


Figure 12: Page d'accueil de l'application mobile

### 1.7 Vérifiez si la connexion au serveur est disponible

Un algorithme se charge de valider, à intervalles réguliers, si le client est capable d'établir une connexion sécurisée avec le serveur. Le statut est automatiquement mis à jour dans l'application. Deux icônes symbolisent l'état de connectivité. Si la première icône est rouge, cela signifie un problème de connexion, tandis que celle en vert indique une connexion établie avec le serveur.

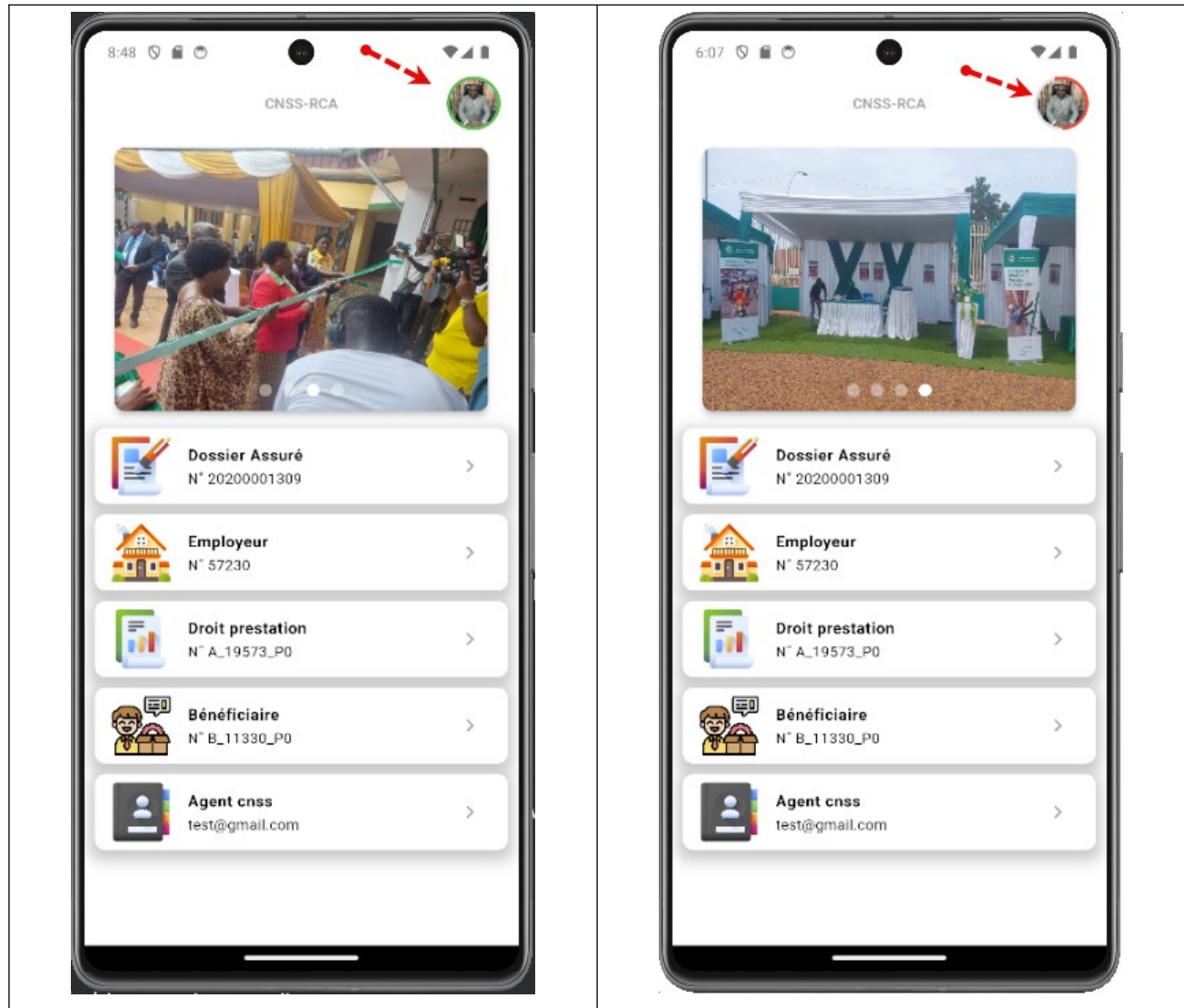


Figure 13: Synchronisation avec le serveur

### 1.8 Informations sur l'assuré

En accédant à ce menu, vous pouvez consulter tous les détails concernant le dossier de l'assuré, y compris les allocations qui lui sont attribuées.

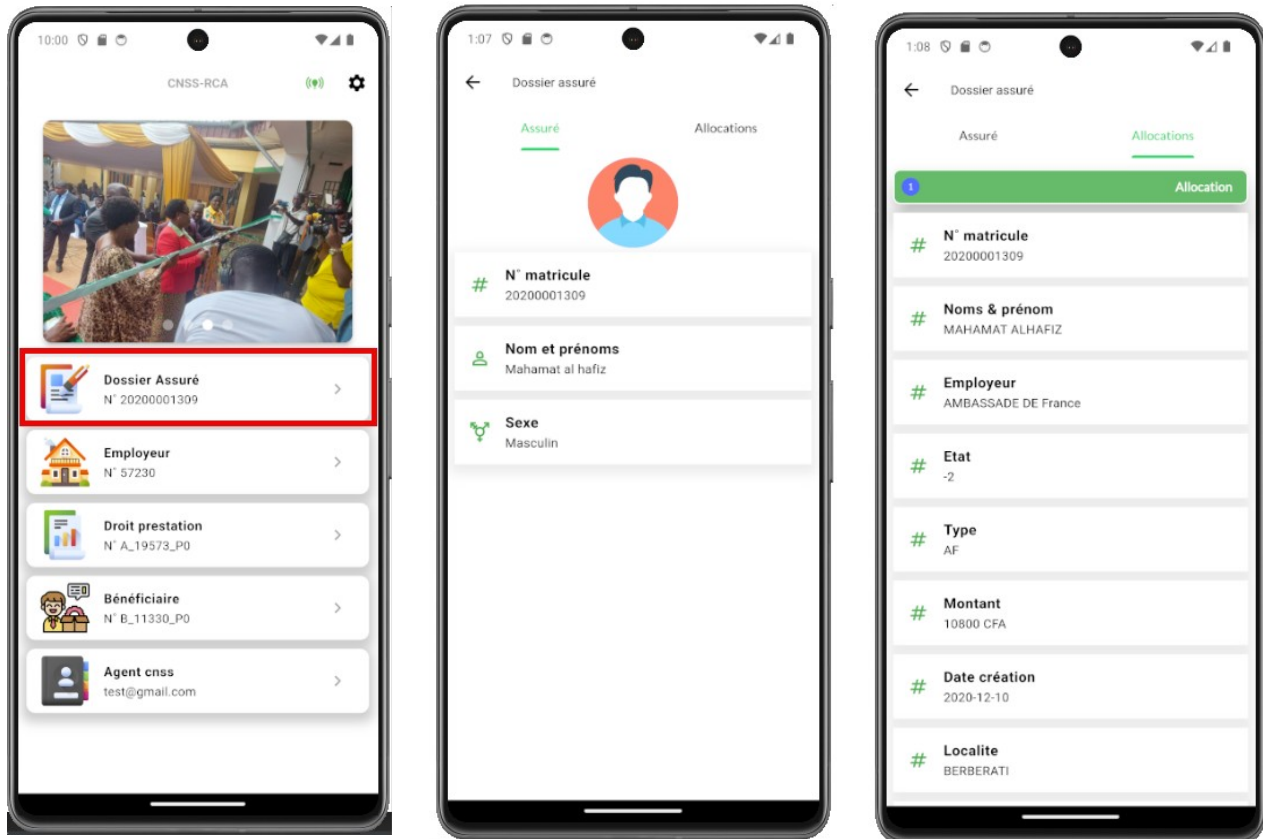


Figure 14: Consultation des informations de l'assuré

## 1.9 Consulter les informations de l'employeur

Cette page offre la possibilité de consulter les renseignements sur l'employeur. En matière d'informations, nous disposons des détails du dirigeant, des informations de l'entreprise et de ces relevés de compte.

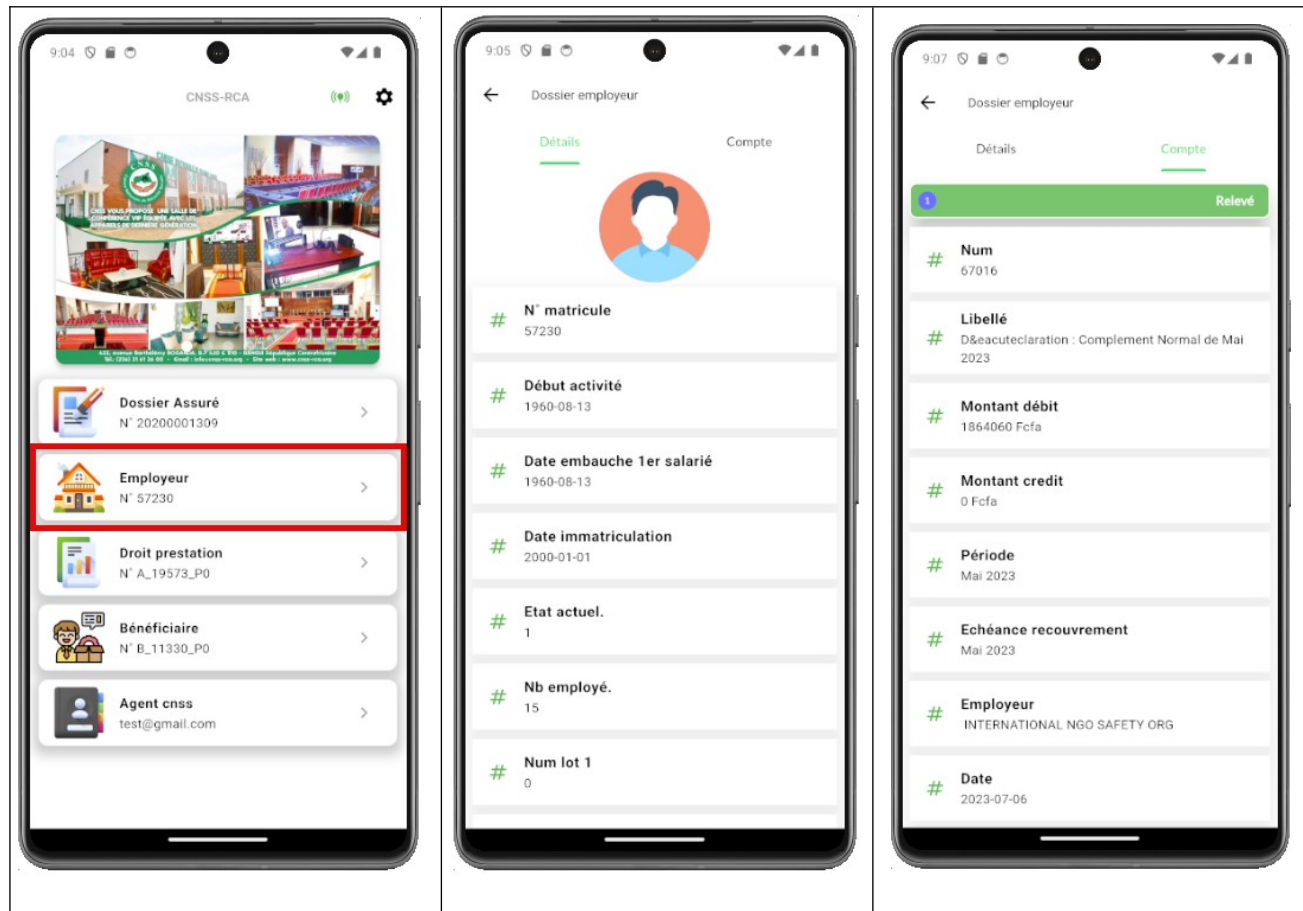


Figure 15: Consultation des informations de l'employeur

## 1.10 Consulter les informations sur les droits prestation

Cette page offre la possibilité de consulter les renseignements un droit les droits de l'assuré. En matière d'informations, nous disposons des détails du dirigeant, des informations de l'entreprise et de ces relevés de compte.

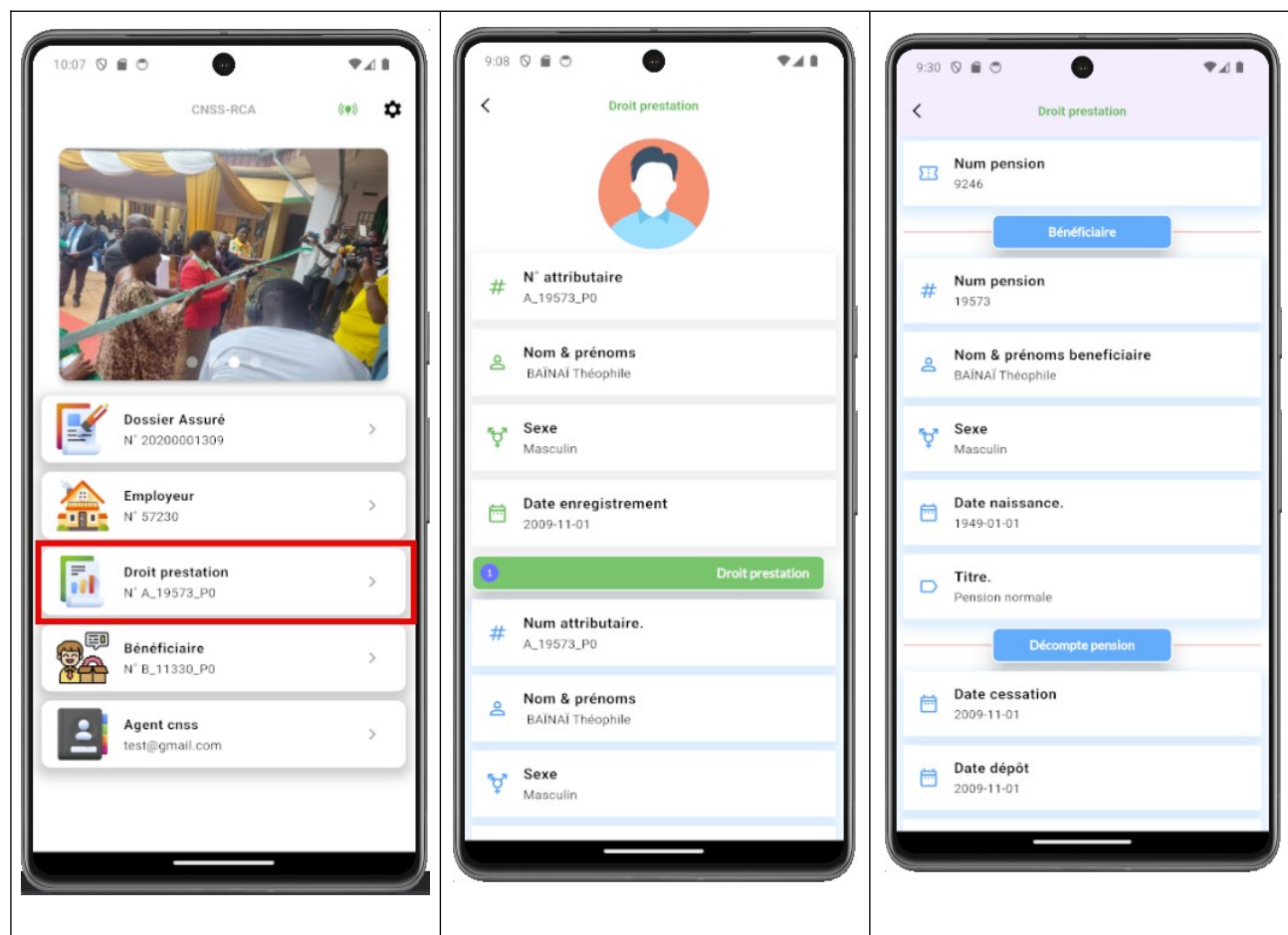


Figure 16: Consultation des informations du droit prestation

9:34

Droit prestation

Décompte pension

**Date cessation**  
2009-11-01

**Date dépôt**  
2009-11-01

**Date effet.**  
2009-11-01

**Date validation**  
2009-11-01

**Décompte pension mode**  
PERIODIQUE

**Décompte pension montant**  
14165.0

**Décompte pension montant trimestre**  
42495.0

Echéance

**Libellé**  
1er Trimestre 2020

9:38

Droit prestation

Echéance

**Libellé**  
1er Trimestre 2020

**Date échéance**  
2020-04-27

Lieu de paiement

**Code.**  
800

**Libelle.**  
ESPECE SOFIA CREDIT

**Type.**  
caisse

Mode de paiement

**Code.**  
ESP

**Libelle.**  
Espèces

Ordonnancement

9:40

**Droit prestation**

---

**Ordonnancement**

---

**Libelle.**  
 Pensions du 1er Trimestre 2020-v-Espèces-ESPECE SOFIA CREDIT-Privé-Autres regime-Autres regime

**Date ordonnancement**  
 2020-04-27

**Ordonnancement période.**  
 1er Trimestre 2020

**Ordonnancement montant.**  
 215661000.0 CFA

**Ordonnancement nb droit.**  
 2896

**Ordonnancement vue**  
 v\_pension\_v\_ESP\_800\_Privé\_autre\_regime\_autre\_regime

---

**Pension**

---

**Date création Pension**  
 Démarrage Utilisateur

**Date Pension**  
 2009-11-01

1:19

**Droit prestation**

---

**Ordonnancement vue**  
 v\_pension\_v\_ESP\_800\_Privé\_autre\_regime\_autre\_regime

---

**Pension**

---

**Date création Pension**  
 Démarrage Utilisateur

**Date Pension**  
 2009-11-01

**Pension date effet**  
 2009-11-01

**Pension num dossier**  
 1

**Nom & prénoms beneficiaire**  
 BAÏNAÏ Théophile

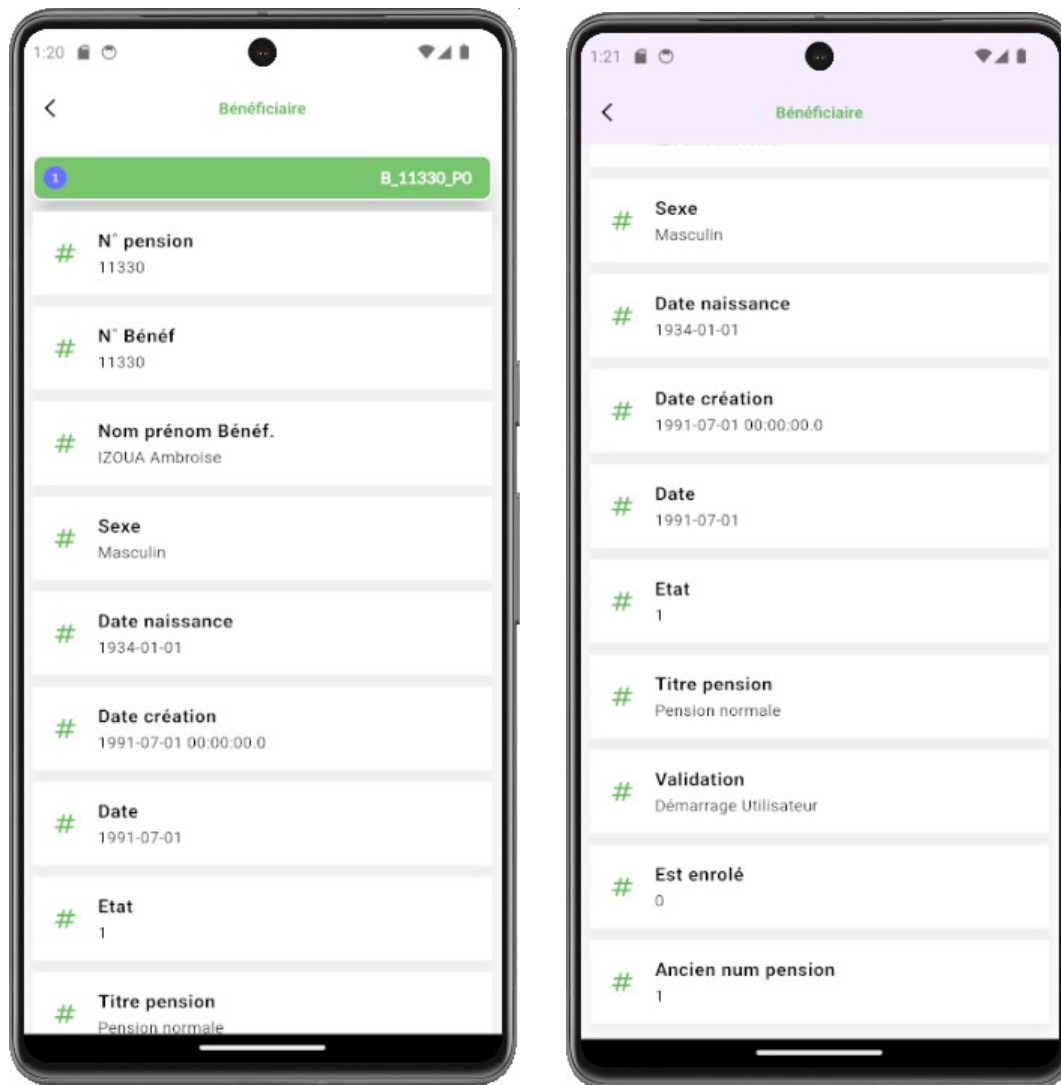
**Pension type**  
 Pension normale

**Pension code**  
 PN

**Supprimer historique**

### 1.11 Consulter les informations du bénéficiaires

Cette page offre la possibilité de consulter les renseignements du bénéficiaire.





### 1.12 Page réservée au employeur de la CNSS

Seuls les employés de la CNSS peuvent accéder à cette section. Ce menu et son contenu ne seront pas visibles par les usagers autres que les agents de la CNSS. Les agents de la CNSS peuvent effectuer des recherches sur les assurés, les bénéficiaires, les attributaires et les employeurs. Elle donne également la possibilité de scanner un document et de vérifier sa validité

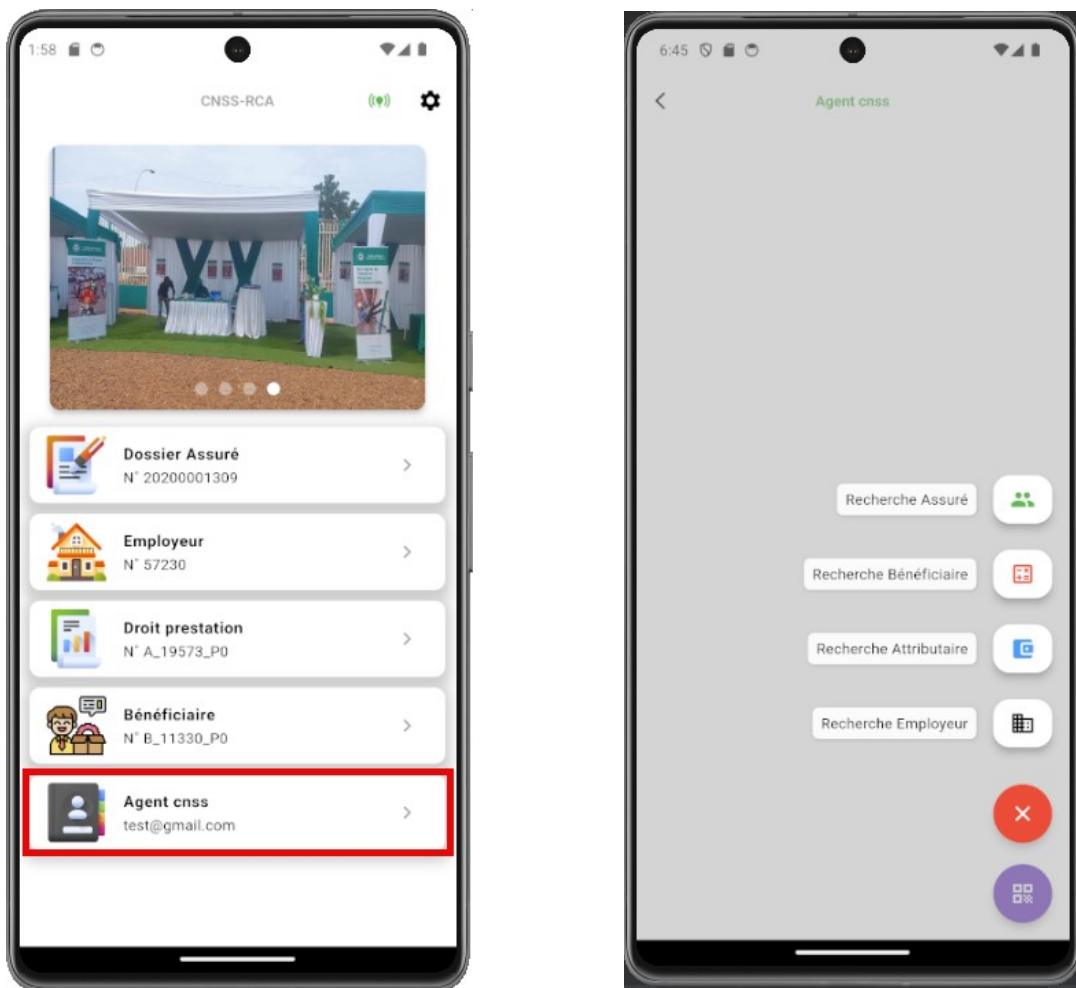


Figure 17: Section réservée aux employeurs de la CNSS RCA

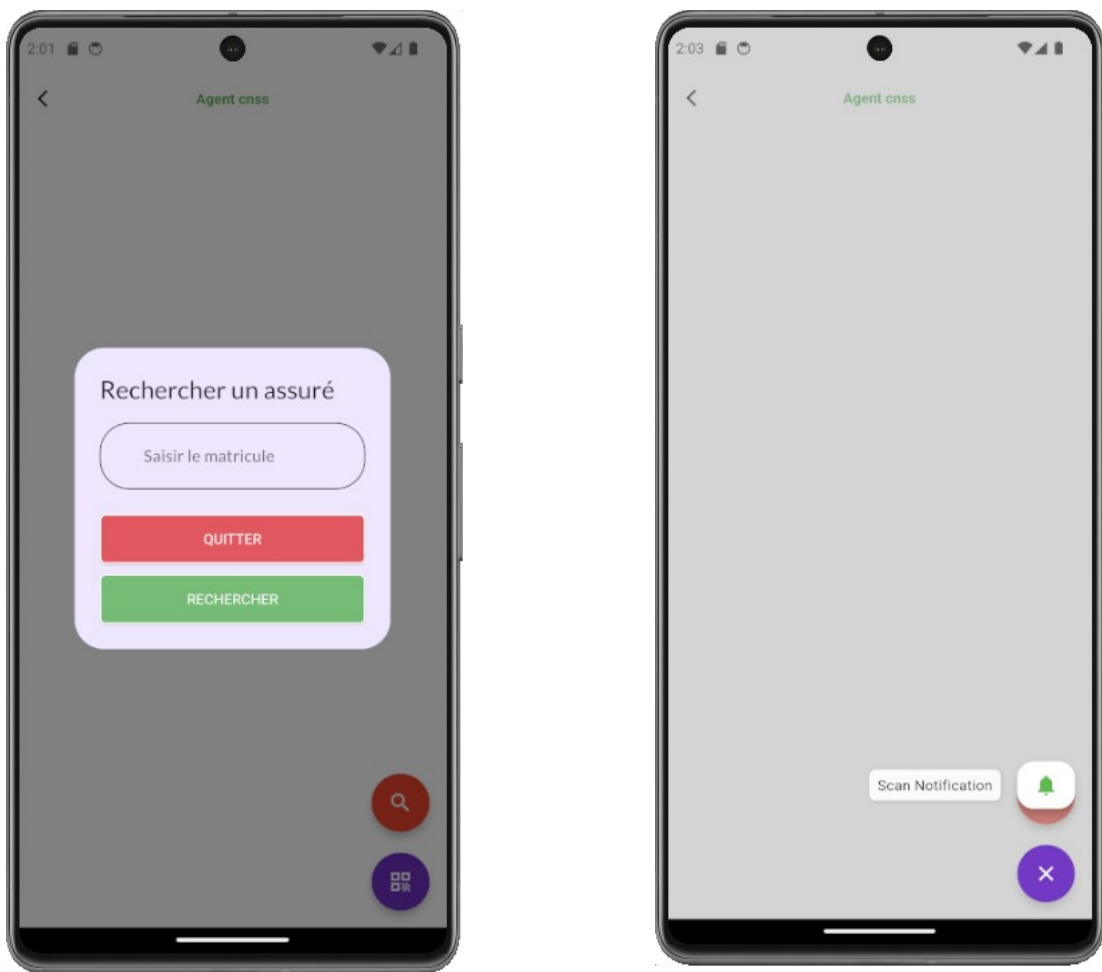


Figure 18: Recherche et scannage de document via Qr code

### 1.13 Scan de code Qr

Le scan permet une identification unique d'un document. Au moment du scan, un hash est extrait du document scan. En utilisant ce hash, une requête est exécutée vers le serveur. Par la suite, le résultat de la recherche est affiché à l'écran pour l'utilisateur.

Pour implémenter le scan de document, il faut prendre en considération les étapes suivantes :

- *Créer le hash dans la base de données.*
- *Emettre sur le document l'image Qr à partir du hash.*
- *Utiliser l'application mobile pour scanner l'image Qr du document*
- *Si le document est valide, les données associées seront renvoyées en résultat*

A titre d'exemple scanner l'un des Qr code à l'ecran.

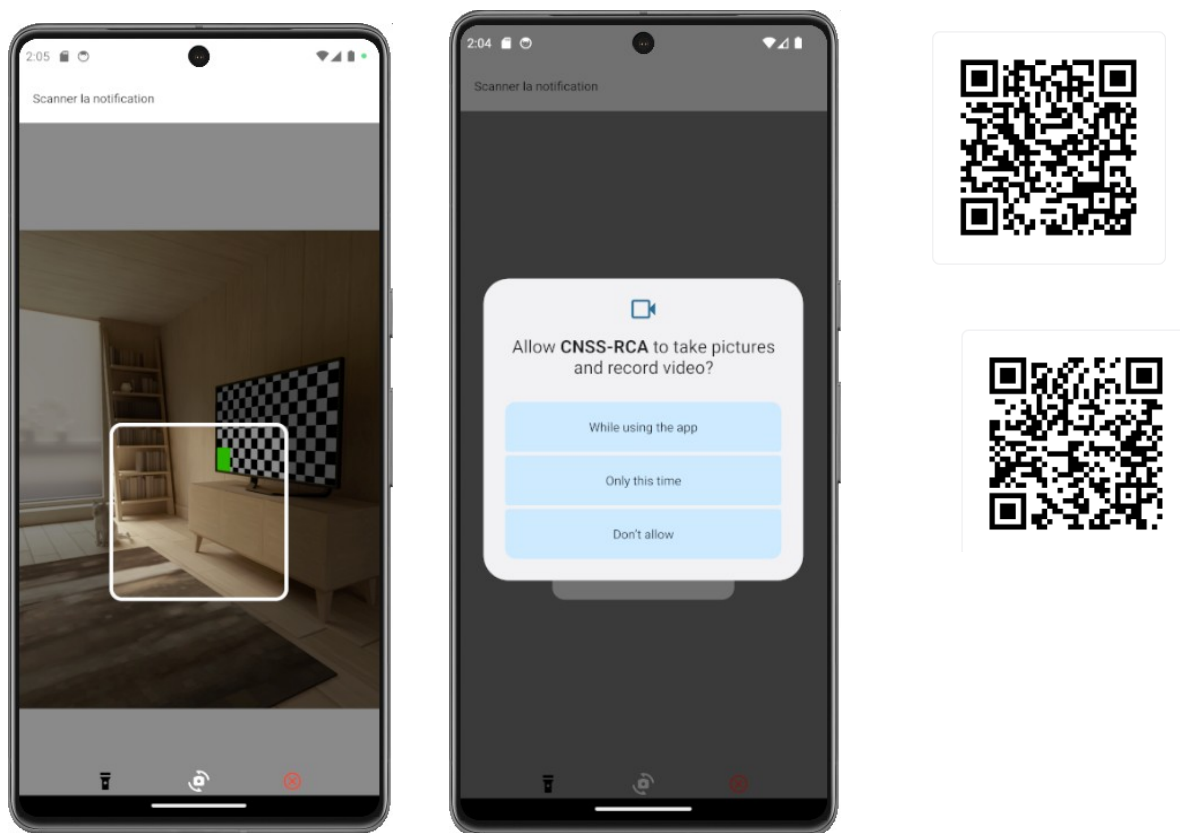


Figure 19: Scan de code QR via la camera du téléphone

### 1.14 Profil utilisateur

Il est possible pour le client de mettre à jour son profil. Lors de la création du compte utilisateur, un profil par défaut est créé.

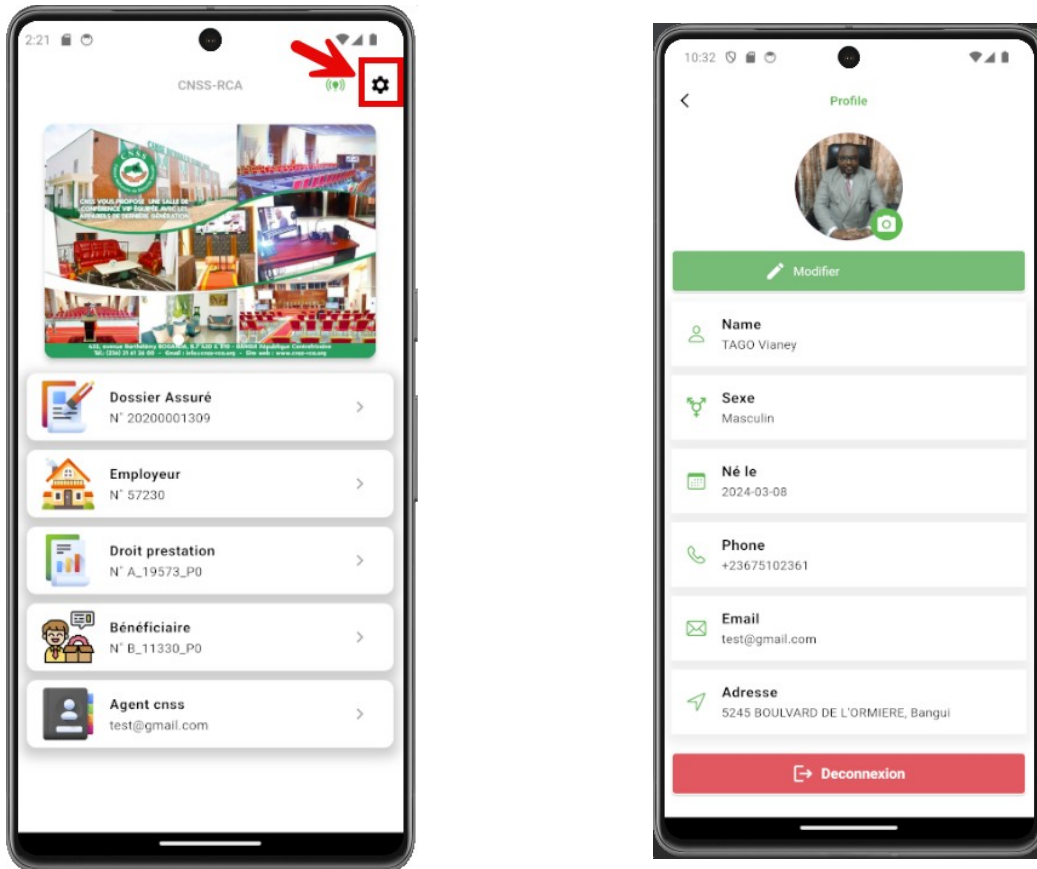


Figure 20: Consultation du profil utilisateur

Pour mettre à jour le profil, il suffit de cliquer sur le bouton 'Modifier' et de remplir le formulaire qui s'affiche à l'écran. Pour mettre à jour la photo de profil, il suffit de cliquer sur l'icône de caméra juste à côté de la photo et de suivre les options.

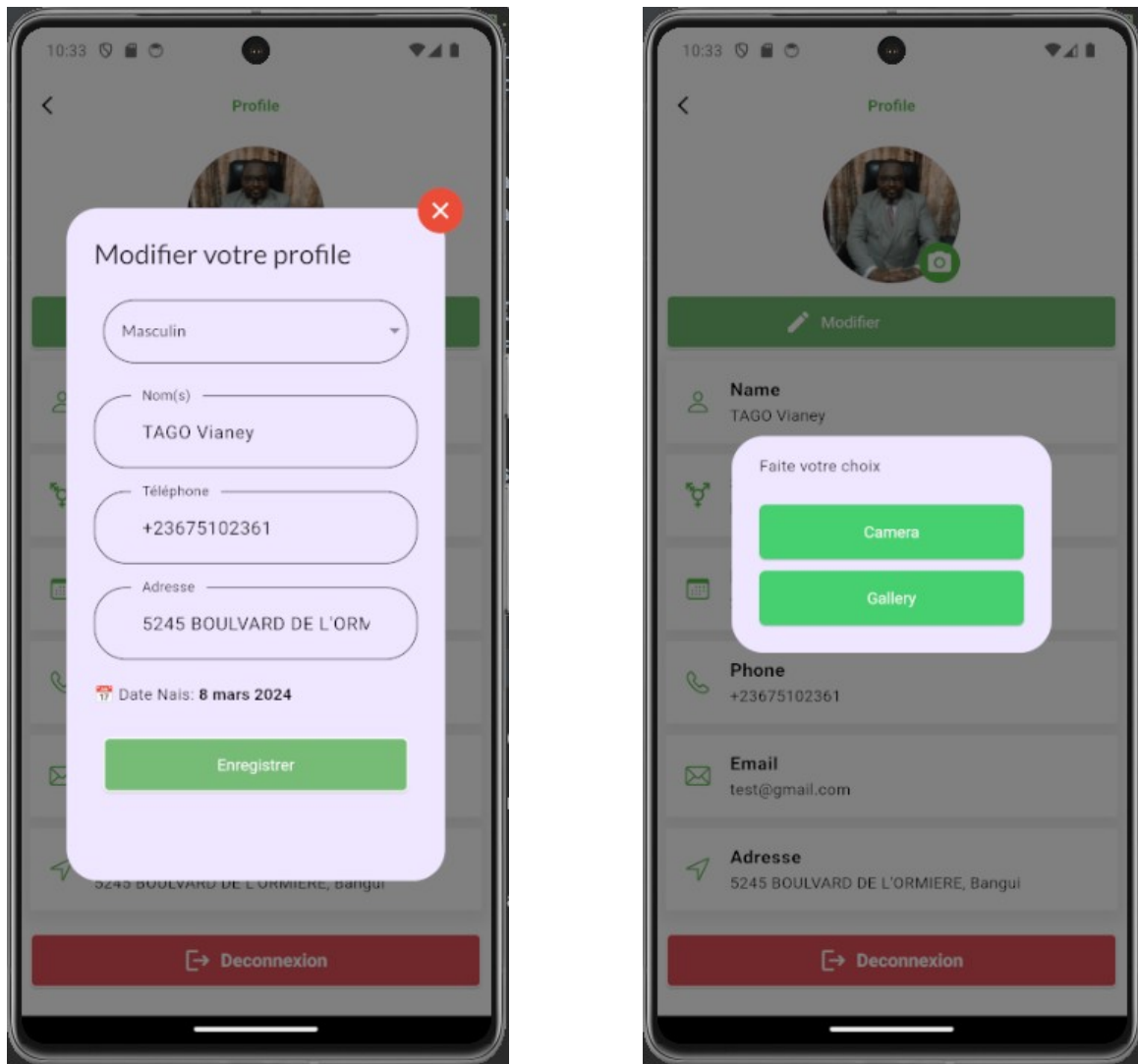


Figure 21: Modification du profil utilisateur

## VI. Conclusion

Les différents tests effectués sur cette version de l'application mobile, donnent une garantie qu'elle peut être utilisée en production. Néanmoins, il reste encore des efforts à faire sur l'infrastructure pour permettre le déploiement automatique des changements. Il est possible d'étendre cette version. Elle peut donc inclure d'autres fonctionnalités, comme l'ajout de données via les divers formulaires.