|  |  |  |
| --- | --- | --- |
|  |  |  |



Modernisation du système de gestion des élections populaires

Par

RAKOTONDRANAIVO Alain Rico

Mémoire présenté  
en vue de l’obtention du grade de Licence

en Informatique

option Développement

Décembre, 2024

Jury :

Dr Olivier Robinson, président  
 M. Tahina Razafinjoelina, examinateur  
 M. Roger Dupont, encadreur professionnel

© RAKOTONDRANAIVO Alain Rico , 2024

Table des matières

**Table des matières**

[Liste des tableaux iii](#__RefHeading___Toc1377_1819560122)

[Liste des figures iv](#__RefHeading___Toc1379_1819560122)

[Glossaire v](#__RefHeading___Toc1381_1819560122)

[Avant-propos 1](#__RefHeading___Toc1383_1819560122)

[L’IT University 1](#__RefHeading___Toc1385_1819560122)

[CENI Madagascar 2](#__RefHeading___Toc1387_1819560122)

[Remerciements 4](#__RefHeading___Toc1389_1819560122)

[Introduction 5](#__RefHeading___Toc1391_1819560122)

[Présentation du projet 6](#__RefHeading___Toc1393_1819560122)

[1 Objectifs du projet 6](#__RefHeading___Toc1395_1819560122)

[2 Planning de réalisation 6](#__RefHeading___Toc1397_1819560122)

[3 Technologies utilisées 8](#__RefHeading___Toc1399_1819560122)

[a) Java 17 8](#__RefHeading___Toc1401_1819560122)

[Réalisation de l’application 9](#__RefHeading___Toc1403_1819560122)

[1 Analyse et conception 9](#__RefHeading___Toc1405_1819560122)

[a) Analyse de l'existant 9](#__RefHeading___Toc1407_1819560122)

[b) Conception de l’application 9](#__RefHeading___Toc1409_1819560122)

[2 Développement par fonctionnalité ou module 10](#__RefHeading___Toc1411_1819560122)

[a) Module ou Fonctionnalité numéro 1 10](#__RefHeading___Toc1413_1819560122)

[b) Module ou Fonctionnalité numéro 2 10](#__RefHeading___Toc1415_1819560122)

[c) Gestion des utilisateurs 10](#__RefHeading___Toc1417_1819560122)

[3 État d’Analyse et Statistiques 11](#__RefHeading___Toc1419_1819560122)

[a) État numéro 1 11](#__RefHeading___Toc1421_1819560122)

[b) État numéro 2 11](#__RefHeading___Toc1423_1819560122)

[c) Statistique numéro 1 11](#__RefHeading___Toc1425_1819560122)

[4 Problèmes rencontrés et solutions 12](#__RefHeading___Toc1427_1819560122)

[Évaluation du projet et connaissances acquises 13](#__RefHeading___Toc1429_1819560122)

[1 Bilan pour l'entreprise 13](#__RefHeading___Toc1431_1819560122)

[2 Bilan personnel 13](#__RefHeading___Toc1433_1819560122)

[3 Extension et évolution de l’application 13](#__RefHeading___Toc1435_1819560122)

[Conclusion 14](#__RefHeading___Toc1437_1819560122)

[Bibliographie 15](#__RefHeading___Toc1439_1819560122)

[Annexe i](#__RefHeading___Toc1441_1819560122)

# Liste des tableaux

[Tableau 1 : Classement TIOBE 2014 de quelques langages de programmation 8](#_Toc441148074)

# Liste des figures

[Figure 1: Part de marché des navigateurs en février 2014. 4](#_Toc441148082)

# Glossaire

**ACID :** Terme décrivant les quatres propriétés essentielles d’un transaction de base de données (Atomicité, Cohérence, Isolation, Durabilité)

**API (**Application Programming Interface**) :** Interface permettant aux applications de communiquer entre elles en échangeant des données de manière structurée.

**Back-office :** Partie d’un système informatique qui gère les processus internes de l’organisation, souvent inaccessible au grand public.

**Composant (**cadre frontal**) :** Bloc de construction fondamental qui permet aux développeurs de créer des interfaces modulaires et réutilisables.

**Framework** : Cadre de développement en français, c’est une structure logicielle qui fournit des outils, des bibliothèques, des conventions pour faciliter le développement d’applications.

**Gantt :** C’est un diagramme qui représente visuellement un planning ou un calendrier de projet.

**JPA (**Java Persistence API**) :** Une interface Java pour la communication avec les bases de données.

**JSON (**JavaScript Object Notation**) :** Un format de donnée léger et lisible, utilisé pour l’échange d’informations entre un client et un serveur, principalement dans les applications Web.

**JWT (**JSON Web Token**) :** Un standard pour sécuriser les échanges de données entre deux parties.

**HTTP (**Hypertext Transfer Protocol**)** : Protocole de communication utilisé pour transferer des informations sur le Web entre un client (comme un navigateur web) et un serveur.

**Module (**cadre frontal**) : U**nité de code organisée qui regroupe des composants, services et autres éléments liés fonctionnellement pour structurer l'application de manière modulaire.

**Oracle Spatial :** Une extension de la base de données Oracle, conçue pour le traitement des données géospatiales.

**Spring Boot :** Extension de Spring qui simplifie le déploiement d’applications en réduisant les configurations nécessaires.

**Spring Security :** Module de sécurité pour le framework Spring, offrant des fonctionnalités de protection et de contrôle d’accès pour les applications.

**SQL :** Langage de requête structuré pour interagir avec les bases de données relationnelles.

**NoSQL :** Un modèle de base de données non relationel, souvent optimisé pour la gestion de données en temps réel et une grande flexibilité de schema.

# Avant-propos

Le présent mémoire présente les résultats du travail effectué lors de mon stage de fin d’études de Licence en Informatique de l’IT University; stage effectué au Direction du Système Informatique de la CENI Madagascar durant 03 mois, de Août à Novembre 2024.

Afin de poser clairement le contexte de ce mémoire, je vais présenter succinctement d’une part l’IT University et d’autre part mon établissement d’accueil.

## L’IT University

Fondée en 2011, l’IT University (ou ITU) est une université privée, spécialisée en informatique, formant les jeunes bacheliers, de préférence scientifiques :

* En trois ans, pour l‘obtention d’une :
  + Licence, option Développement, Réseaux et Bases de Données ou Web et Design
  + Licence, option Graphic Design ou Communication Digitale
* En cinq ans, pour l’obtention d’un :
  + Master MBDS en coopération avec l’Université Côte d’Azur à Nice Sophia Antipolis – France
  + Master BIHAR en coopération avec l’ESTIA du Pays Basque - France

Étant une formation professionnalisante, l’ITU a tissé des liens forts avec ses partenaires industriels, dont l’opérateur convergent TELMA et la plupart des entreprises et institutions du secteur des TIC[[1]](#footnote-2). Ces partenaires participent effectivement à la formation par la fourniture de connexion Internet à haut débit, l’envoi de conférenciers ou par l’accueil des étudiants en stage. Beaucoup de ces partenaires recrutent aussi les sortants dès leur sortie d’École.  
D’autre part, le corps enseignant de l’ITU est constitué intégralement de spécialistes de très haut niveau et obligatoirement actifs professionnellement dans leurs domaines respectifs.

## CENI Madagascar

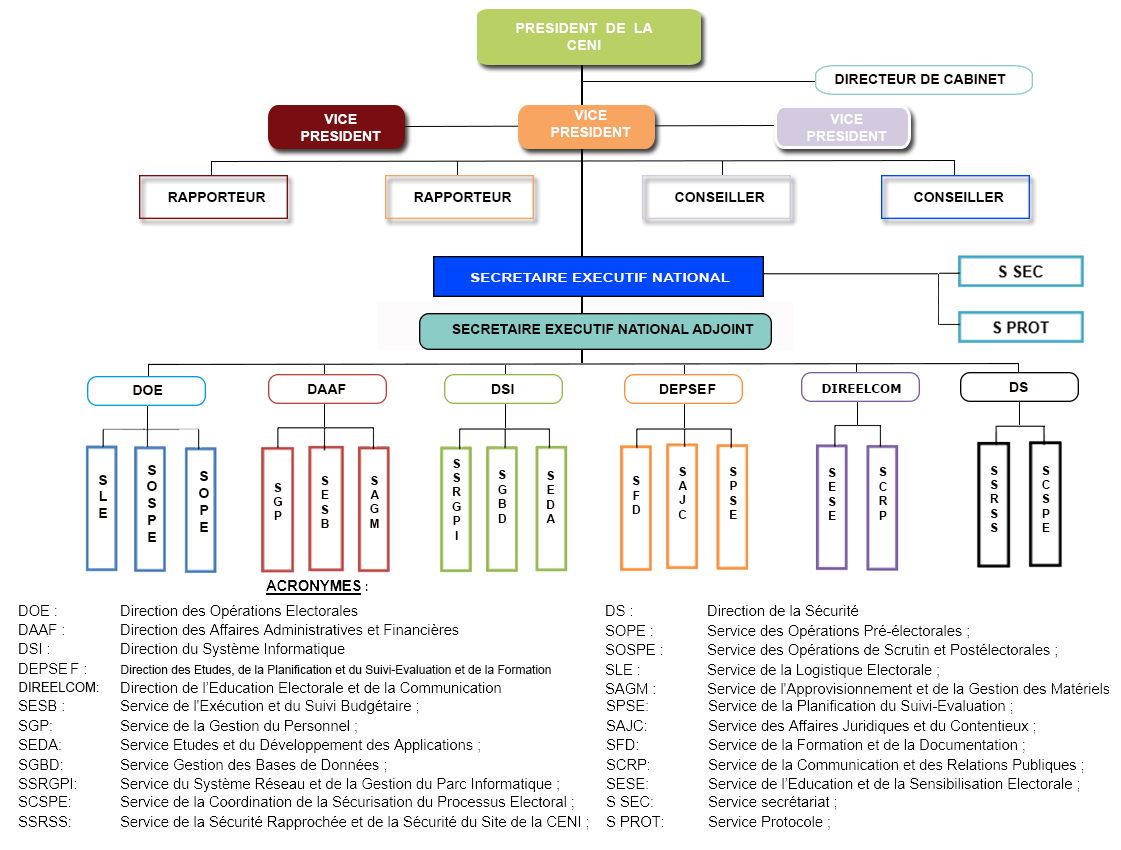
La Commission Électoral National Indépendante (CENI) de Madagascar est l’organisme officiel chargé de la gestion, de la supervision et de l’organisation des élections sur l’ensemble du territoire malgache.

La CENI a été formellement instituée par la **Loi n° 2015-020**, promulguée le **19 octobre 2015**, qui établit ses structures, ses responsabilités, et son autonomie financière et administrative.

Selon leur plan de travail pluriannuel, les principales missions de la CENI incluent :

* La préparation et la mise en place des élections : Organisation des dispositifs afin d'assurer des élections transparentes, en respectant les lois électorales.
* La surveillance des opérations électorales : Garantie de la liberté, justice et transparence tout au long du processus électoral.
* La certification des résultats : Validation des résultats électoraux reflétant la volonté des électeurs.
* La médiation en cas de litige : Résolutions des contestations liées aux élections.
* La sensibilisation des électeurs : Campagnes d'éducation civique pour informer et encourager la participation électorale.
* La formation et assistance technique : Formation des acteurs électoraux et soutien technique pour renforcer leurs compétences.
* La supervision du vote et du dépouillement : Contrôle rigoureux des opérations de vote et dépouillement.

L’organigramme de la CENI est présenté comme ceci :

Figure 1: Organigramme de la CENI.

## Remerciements

Je souhaite exprimer ma sincère reconnaissance envers tous ceux qui ont participé à la réalisation de ce mémoire.

Avant tout, je voudrais remercier mon superviseur pédagogique, le Dr RAZAFINJATOVO Heriniaina [...].

Je tiens à adresser également ma gratitude envers Mme RANAIVOASIMBOLA Vanona, mon encadreur professionnel, pour ses conseils et ses critiques qui ont influencé mes réflexions tout au long de mes recherches.

J'aimerais aussi exprimer ma reconnaissance envers la CENI pour m'avoir offert l'occasion de faire ce stage et de collaborer sur un projet d’envergure national.

Un grand merci à l’IT University ainsi qu’à tous les membres du personnel pour les connaissances transmises et les éducations que j’ai pu acquérir tout au long de mes années d’études.

Sans oublier :

* Mr RANAIVODIMBY Alberto, Directeur du Système d’Information de la Commission Électorale Nationale Indépendante ainsi que Mme RABOBA Julie Viviane, Chef du Service de la Gestion de la Base de données pour m’avoir offert leur temps lors des enquêtes réalisées.
* Toute l'équipe du DSINT (Direction des Systèmes d’Informations et des Nouvelles Technologies) de la CENI pour leur accueil chaleureux et leur collaboration pendant ces 03 mois.
* Tous les membres du personnel de la CENI.

Et enfin, mes plus sincères remerciements aux parents, la famille, les amis pour leur soutien et leurs encouragements qui ont joué un rôle crucial dans la réalisation de ce travail.

# Introduction

L’essentiel du travail que j’ai réalisé a porté sur le thème : « Modernisation du système de gestion des élections populaires ».

Ce dernier revêt une importance particulière dans le domaine de l’informatique, car il touche aux enjeux de la transparence, d’efficacité et de sécurité des processus critiques. Dans un milieu où le volume de données est déjà immense et les exigences de protection de l’information toujours plus élevées, les technologies informatiques sont devenues essentielles pour soutenir la fiabilité des systèmes électoraux.

C’est la raison pour laquelle la CENI m’a confié la responsabilité de développer un système de gestion électorale unifié, intégrant une nouvelle application mobile et back-office.

Ce projet permettra aux personnels de la CENI de faciliter la prise de décision, de gagner en efficacité et d’accélérer le processus de traitement jusqu’à la publication des résultats.

Les résultats essentiels de mon travail durant ce stage sont listés ci-après :

* Optimisation et création d’une nouvelle application mobile pour les bureaux de vote, permettant un enregistrement et une transmission directe des résultats électoraux.
* Suivi en temps réel des incidents grâce à la mise en place d’un système d’alerte.
* Unification des systèmes back-office pour centraliser la gestion des résultats électoraux, indépendamment du type de scrutin.
* Mise en place d’un tableau de bord permettant de visualiser les statistiques essentielles.

Ce mémoire est structuré en trois grandes parties :

* Dans un premier temps, la présentation du contexte et des enjeux du projet
* Ensuite, la description détaillée de la réalisation technique et des fonctionnalités
* Enfin, l’évaluation de l’impact du projet, accompagnée des connaissances acquises durant cette expérience.

# Présentation du projet

## Objectifs du projet

Le projet vise à garantir une gestion centralisée et efficace des processus électoraux. Cette initiative englobe plusieurs aspects cruciaux, tels que l’enregistrement des électeurs au niveau des bureaux de vote, la récolte et remontée des résultats, la gestion des incidents et aussi l’exportation des résultats provisoires.

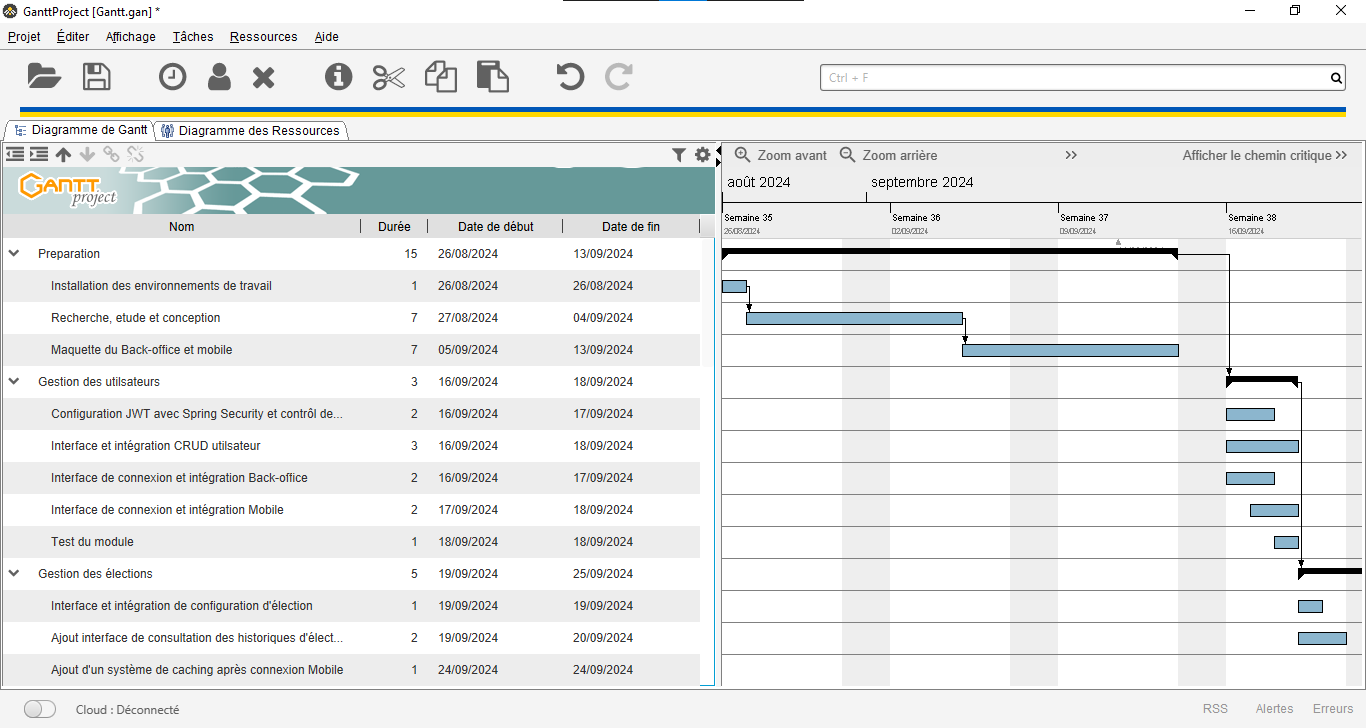
Nous aspirons également à offrir une meilleure visibilité sur les statistiques, comme le taux de couverture de l’application mobile, le taux d’abstention des électeurs et la fréquence des incidents au sein des bureaux de vote. Avec un tableau de bord dynamique, les équipes de la CENI pourront analyser aisément les données, ce qui facilitera leur prise de décision.

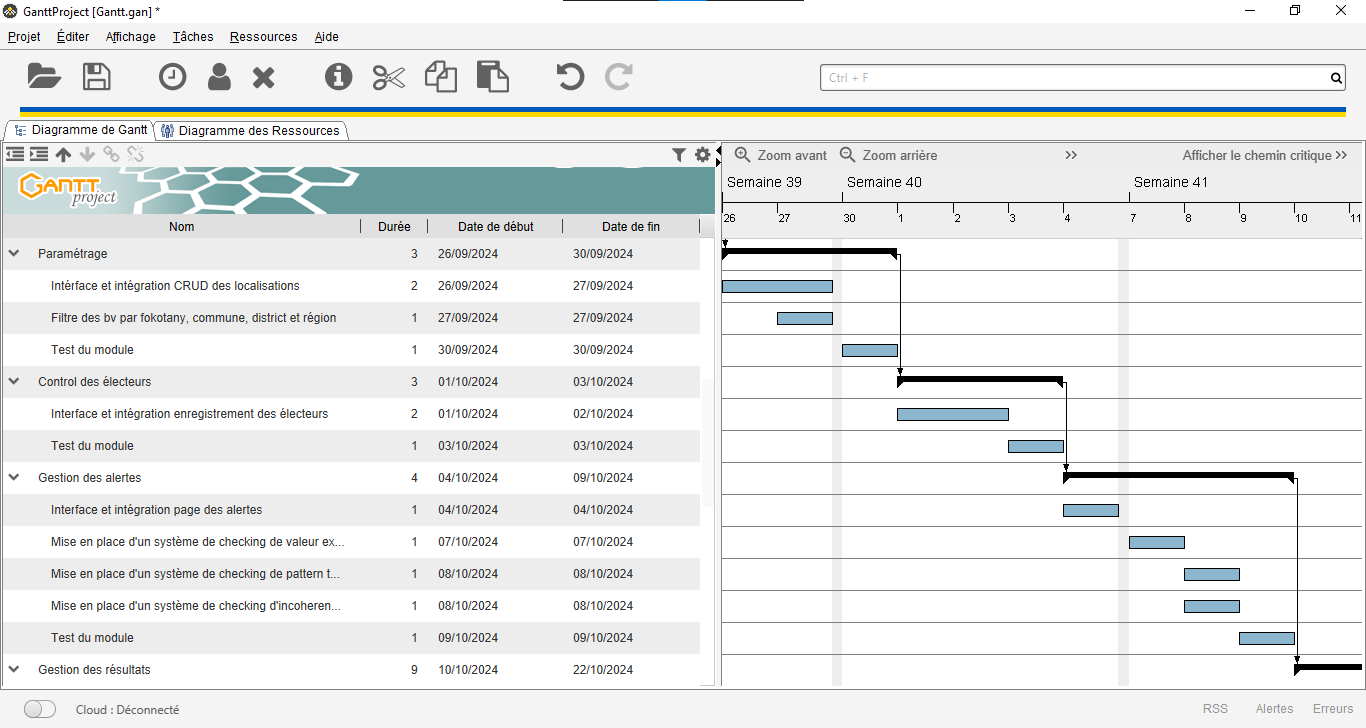
En somme, l’objectif de cette refonte est d’obtenir une solution plus moderne, efficace et optimisée par rapport à la version précédente.

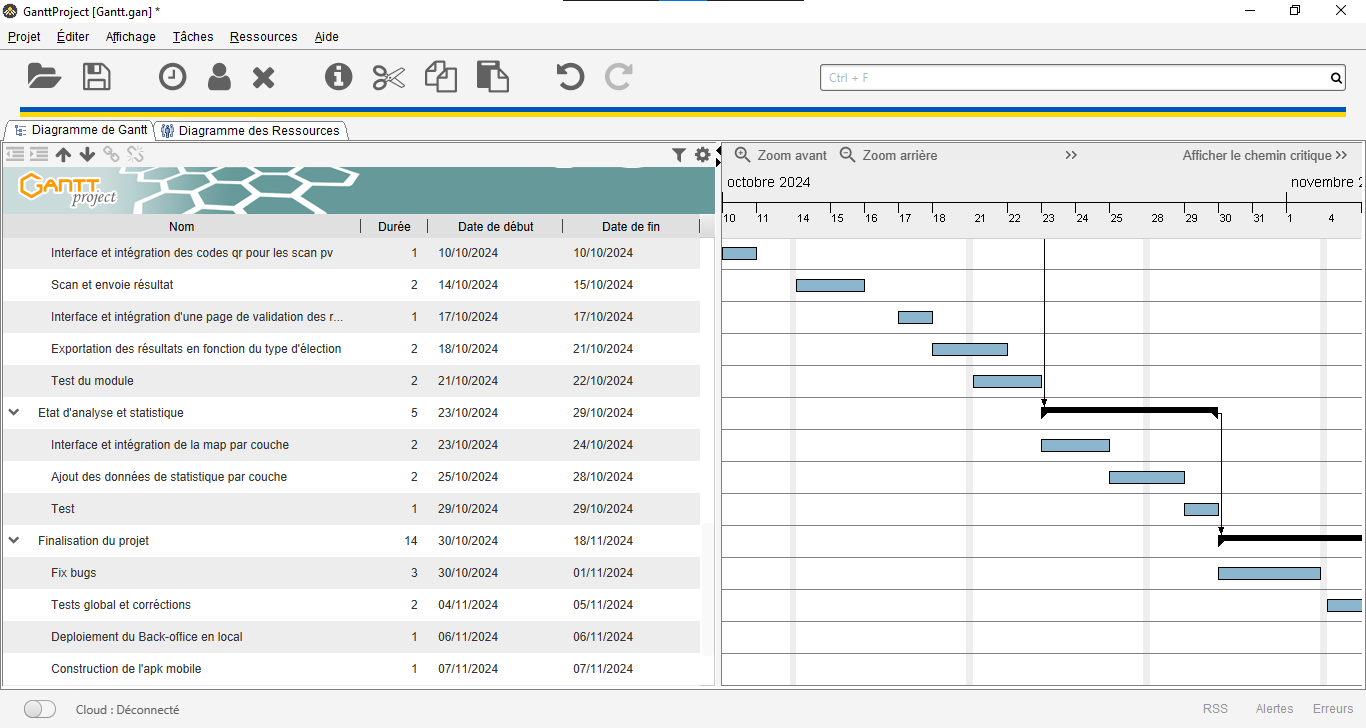
## Planning de réalisation

La planification d’un projet revêt une grande importance. Comme disait **H. Stanely Judd : ’*Un bon plan est comme une carte routière : il indique la destination finale et généralement la meilleure façon d’y arriver*’. C'est la raison pour laquelle, avant de se lancer dans l'exécution du projet, nous avons commencé par élaborer un plan détaillé. Cette étape préliminaire nous a permis de définir les objectifs et de créer un planning précis des étapes à suivre afin de mieux organiser notre temps et notre travail.**

**Le schéma suivant montre la réalisation du projet durant les semaines passées au sein de la CENI, avec le diagramme de GANTT :**

Figure 2: Diagramme de GANTT partie 1

Figure 3: Diagramme de GANTT partie 2

Figure 4: Diagramme de GANTT partie 3

## Technologies utilisées

Concernant les technologies utilisées pour le développement de notre système, nous avons utilisés :

### Spring Framework avec Spring Boot

Spring est un framework Java moderne très robuste, conçu pour simplifier le développement d’applications en entreprise. Couplé avec Spring Boot, il permet aux développeurs de se concentrer sur la logique métier plutôt que sur la configuration. Sa rapidité, sa sécurité ainsi que son écosystème riche en ressources font de Spring une option optimale pour le serveur de traitement de notre projet. Nous l’avons donc utilisés pour générer les APIs consommés par nos applications back-office Web et mobile.

Actuellement, la CENI utilise le framework Django avec le langage de programmation Python pour le développement de ses applications. Bien qu’il s’agisse d’une solution fiable, le passage à Spring pourrait offrir plusieurs avantages significatifs :

|  |  |  |
| --- | --- | --- |
| **Critères** | **Spring** | **Django** |
| Langage | Java | Python |
| Performance | Supérieure en termes de vitesse de réponse, mieux adapté aux applications d’entreprise | Performances acceptables, optimisé pour des applications moyennes |
| Sécurité | Capacité de sécurité renforcée avec Spring Security | Menace potentielle d’exploitation des interpréteurs Python |
| Traitement des requêtes HTTP | Capable de gérer plusieurs requêtes en même temps | Limiter à une seule requête à la fois |

Tableau 1: Tableau comparatif entre Spring et Django

Source : https://optymize.io/blog/django-vs-springboot-which-is-better-for-your-website/

### Oracle

Oracle est un système de gestion de base de donnée relationnelle réputé pour sa flexibilité, sa sécurité et sa performance élevée, notamment dans les applications d’entreprise. Grâce à son architecture optimisée, Oracle excelle dans le traitement de volumes importants de données. La base de données offre des fonctionnalités de haute sécurité, comme le chiffrement des données, garantissant la confidentialité et l’intégrité essentielles aux informations électorales sensibles. De plus, Oracle intègre des fonctionnalités géospatiales via Oracle Spatial, permettant de traiter des données de localisation et d’analyser la répartition géographique des données de vote.

### Angular

Angular est un framework de développement d’applications Web spécialisé dans les cadres frontaux. Il permet de créer des interfaces modernes et réactives grâce à son architecture basée sur des modules. Son système de compilation Ahead-Of-Time (AOT) optimise les performances et réduit les temps de chargement.

|  |  |  |  |
| --- | --- | --- | --- |
| **Critères** | **ReactJs** | **Angular** | **VueJs** |
| Architecture | Basée sur des composants, plus flexible | Basé sur des modules, adapté aux grandes applications | Basée sur des composants, simple à intégrer |
| Performance | Rapide et légère, mais nécessite une gestion manuelle des optimisations | Framework complet, offrant des performances solides, mais un peu lourd en raison de son architecture complète | Également léger et rapide, mais moins adapté pour les grandes applications |
| Communauté et support | De nombreuses bibliothèques et ressources disponible | Écosystème solide soutenu par Google avec une documentation complète. | Écosystème en croissance et documentation entretenue. |

Tableau 2: Tableau comparatif entre ReactJs, Angular et VueJs

Source :[https://blog.logrocket.com/angular-vs-react-vs-vue-js-comparing-performance/#optimization-techniques-angular-react-vue](https://blog.logrocket.com/angular-vs-react-vs-vue-js-comparing-performance/" \l "optimization-techniques-angular-react-vue)

### Flutter

Flutter est un framework de développement d'applications mobiles open-source qui permet de créer des applications performantes et esthétiques à partir d'une seule base de code, tant pour Android que pour iOS. Comme la CENI envisage une refonte de ses applications, Flutter est idéal dans ce contexte, offrant des interfaces utilisateurs de haute qualité. De plus, Flutter bénéficie d’une communauté solide et d’un écosystème en plein expansion, garantissant un soutien continu et des mises à jour régulières.

|  |  |
| --- | --- |
| **Classement** | **Framework** |
| **1** | Flutter |
| **2** | React Native |
| **3** | SwiftUI |
| **4** | Ionic |
| **5** | Xamarin |
| **6** | Corona SDK |
| **7** | Apache Cordova |
| **8** | Framework7 |
| **9** | Vue Native |
| **10** | Native Script |

Tableau 3: Top 10 des frameworks de développement mobiles en 2024

Source : https://www.kellton.com/kellton-tech-blog/top-mobile-app-development-frameworks

### SQLite

SQLite est un autre système de gestion de base de données relationnelle, léger et sans serveur, qui stocke les données dans un simple fichier sur le périphérique. Selon le classement publié par GeeksForGeeks le 16 Juillet 2024, SQLite se figure en deuxième place derrière Firebase. Le tableau suivant justifie le choix de ce dernier par rapport à Firebase

|  |  |  |
| --- | --- | --- |
| **Critères** | **SQLite** | **Firebase** |
| Stockage | Local (sur l’appareil de l’utilisateur), aucune dépendance réseau requise | Cloud (sur des serveurs Google), support hors-ligne partiel (nécessite la synchronisation en ligne) |
| Configuration | Simple, aucune configuration serveur requise | Intégration en ligne, nécessite des services cloud |
| Type de données | SQL, avec support de transactions et intégrité ACID | NoSQL, orienté en temps réel |
| Cas d’utilisation | Parfait pour les applications nécessitant un stockage sécurisé et local | Adapté aux applications qui demandent des fonctionnalités en temps réel (ex : chat, notification...) |

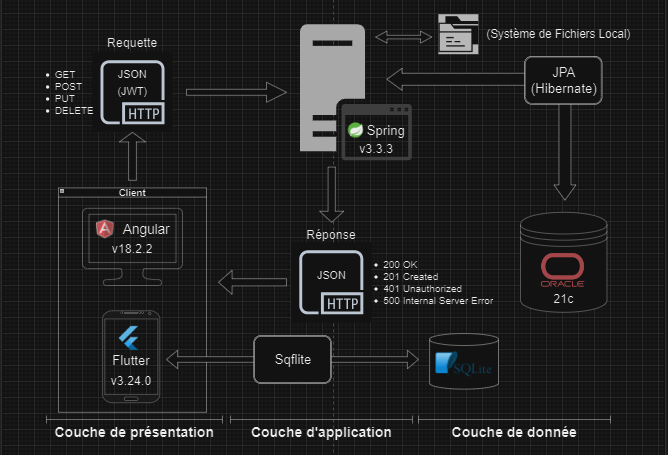
Tableau 4: Tableau comparatif entre SQLite et Firebase

Source :https://www.geeksforgeeks.org/databases-for-mobile-app-development/

## Architecture

Notre système utilise une architecture trois tiers. C’est une approche de conception pour les systèmes d’information qui sépare les différentes parties de l’application en trois couches ou niveaux logiques. Chaque couche étant responsable de tâches spécifiques, ce qui permet une meilleure organisation et maintenance du code :

* **Couche de présentation :** C’est la couche supérieur qui interagit avec les utilisateurs finaux. Elle assure la présentation des informations à l’utilisateur et de la collecte des saisies utilisateurs. Ici nous avons une application mobile (Flutter) et une application web (Angular) dans la couche.
* **Couche d’application :** Cette couche contient la logique métier et le traitement de tout le système. Elle gère les fonctionnalités principales de nos applications, telle que le traitement des demandes utilisateurs, l’exécution des opérations métiers et la gestion des données. Ici nous avons le serveur backend (Spring).
* **Couche de données :** C’est la couche inférieure responsable de la gestion et du stockage des données. Elle comprend des bases de données ou des systèmes de stockage de données où les données des applications sont stockées et récupérées. Elle interagit avec la couche d’application pour fournir et recevoir des données. Ici, nous avons les bases de données Oracle et SQLite dans la couche.

Figure 5: Architecture du système

# 

# Réalisation de l’application

## Analyse et conception

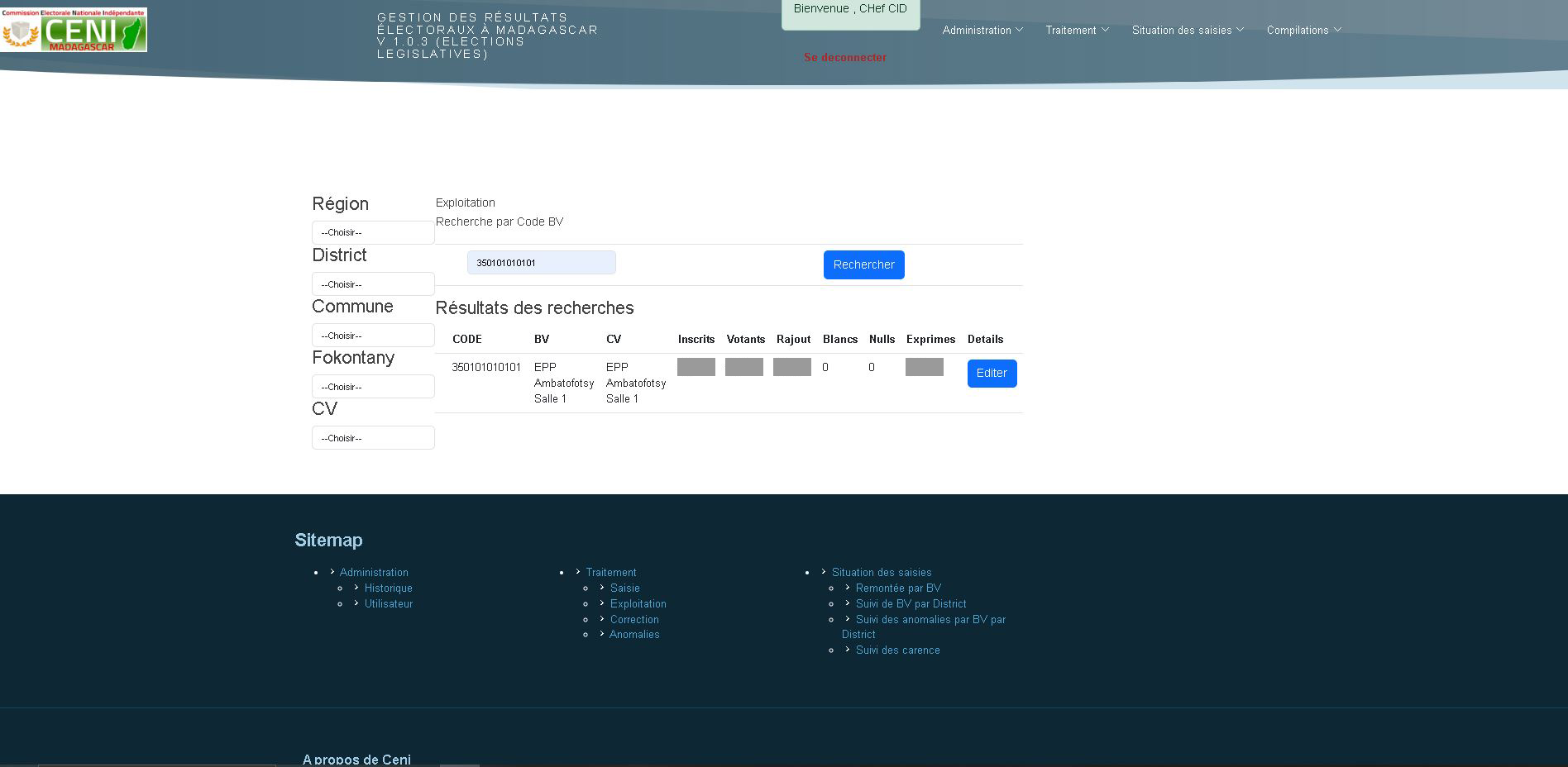
Comme le but de ce stage est de moderniser la version actuelle de l’application de gestion des résultats existante, des enquêtes ont eu lieu auprès des chefs de service de la DSINT pour discuter des points à améliorer et pour lister les fonctionnalités à implémenter.

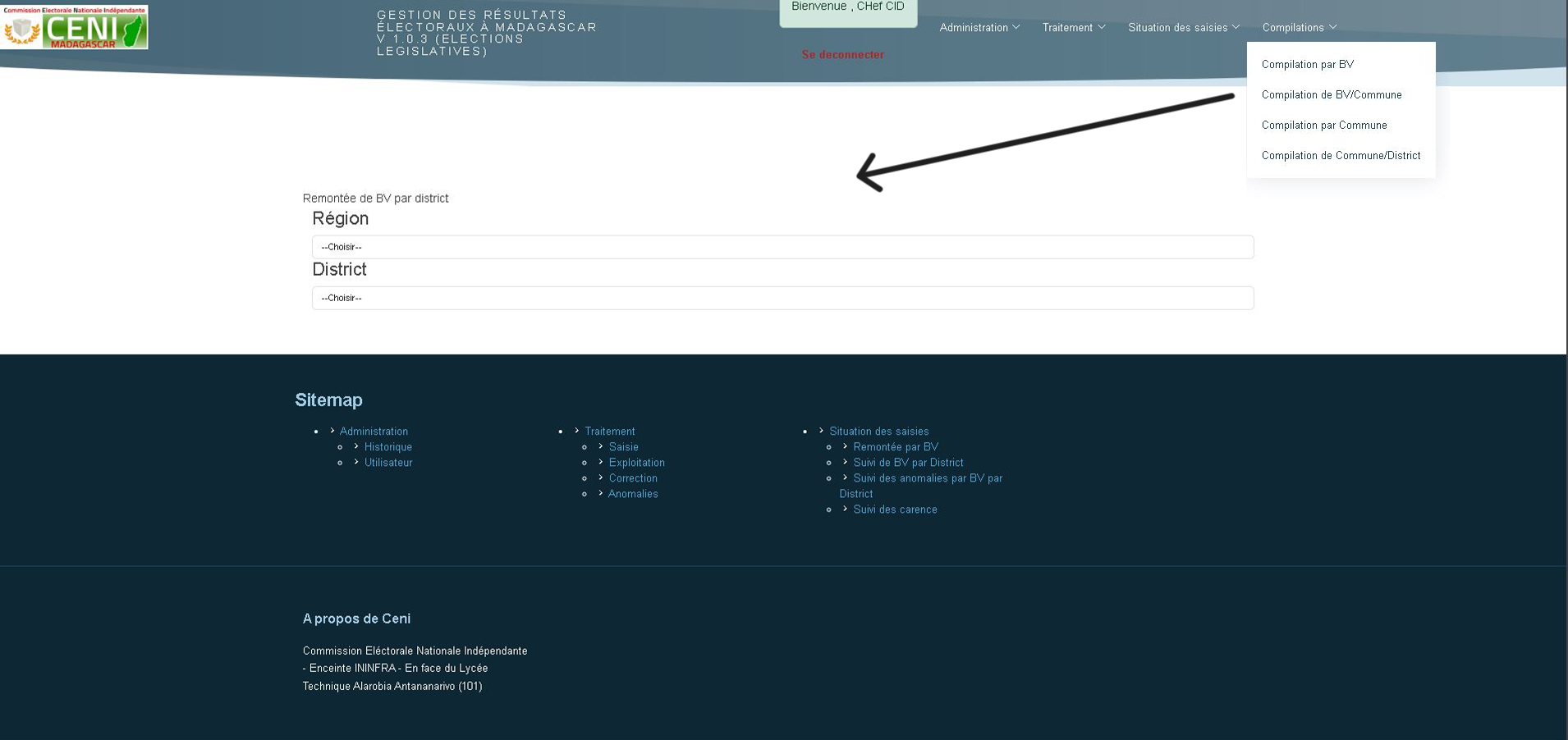
### Analyse de l'existant

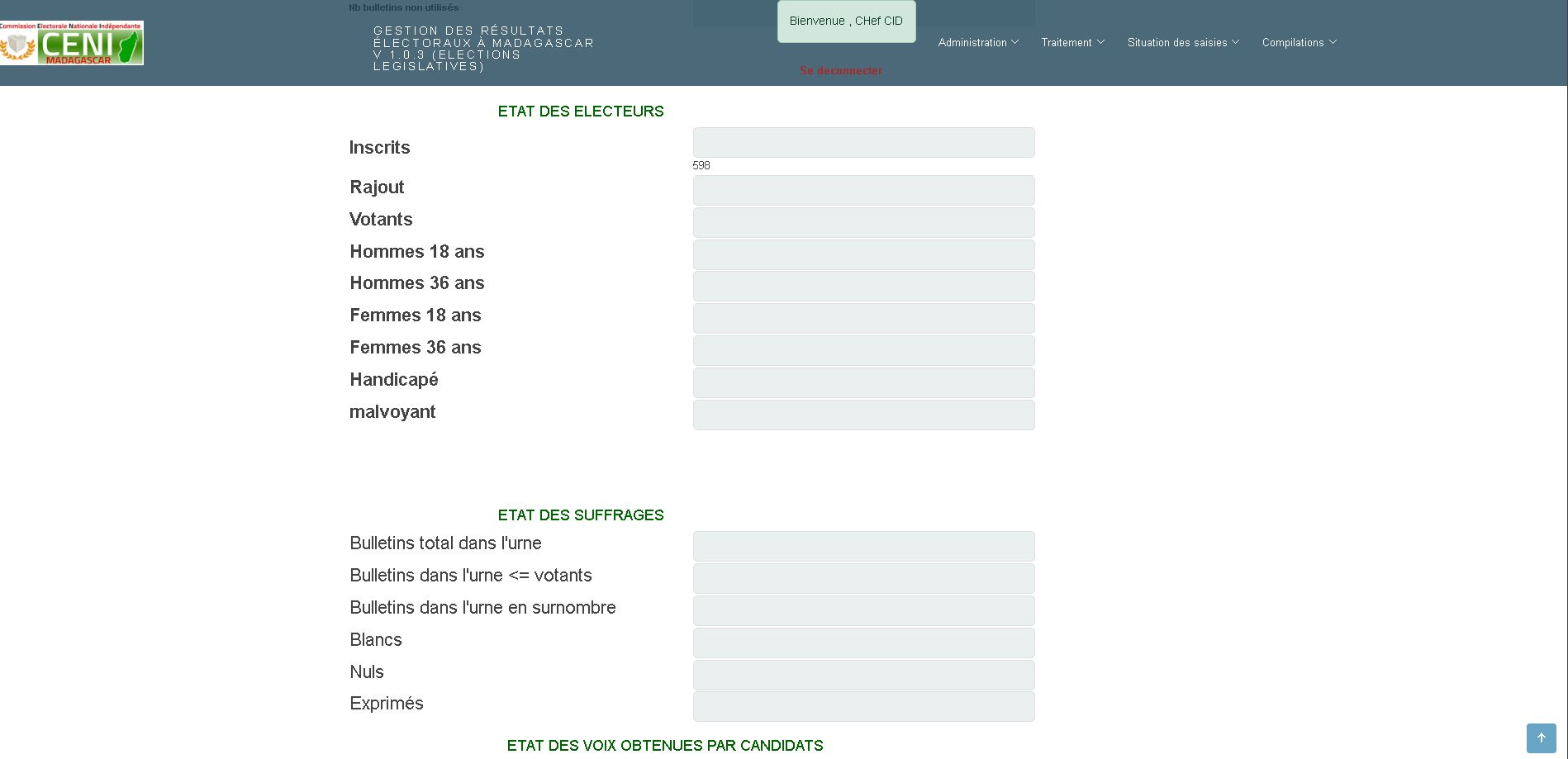
La CENI dispose actuellement de plusieurs systèmes indépendants pour gérer les opérations électorales, incluant des applications pour la gestion des électeurs et des candidats, plusieurs applications spécialisées pour gérer les résultats et plusieurs applications mobiles pour la collecte des résultats selon le type de scrutin. Toutefois, ces applications de gestion des résultats opèrent de manière isolée, sans interconnexion directe, ce qui peut limiter la coordination et la centralisation des données nécessaires.

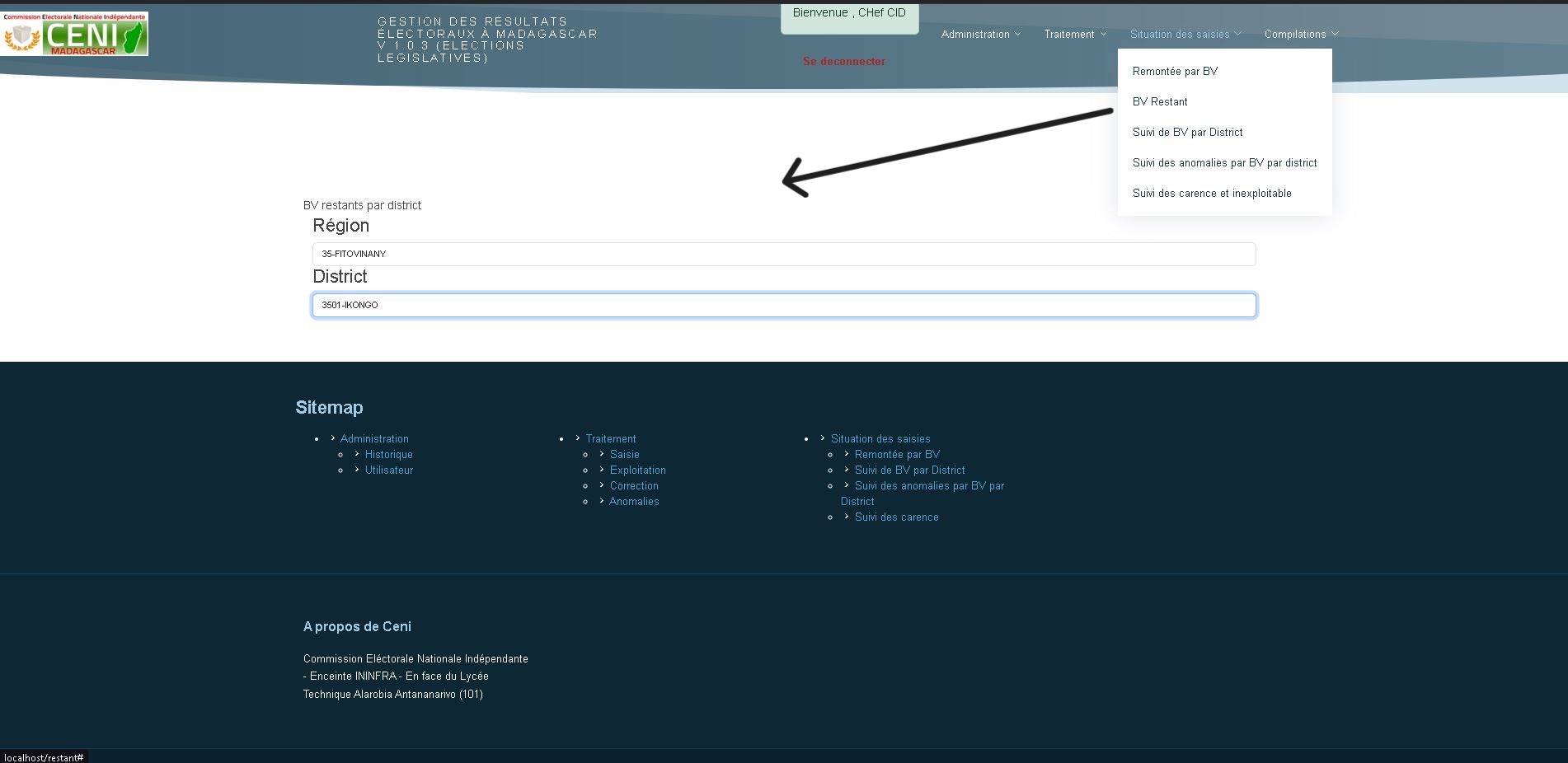
Le nouveau système proposé vise à unifier les processus de gestion des résultats indépendamment du type de scrutin, tout en intégrant les informations nécessaires provenant des systèmes de gestion des électeurs et des candidats.

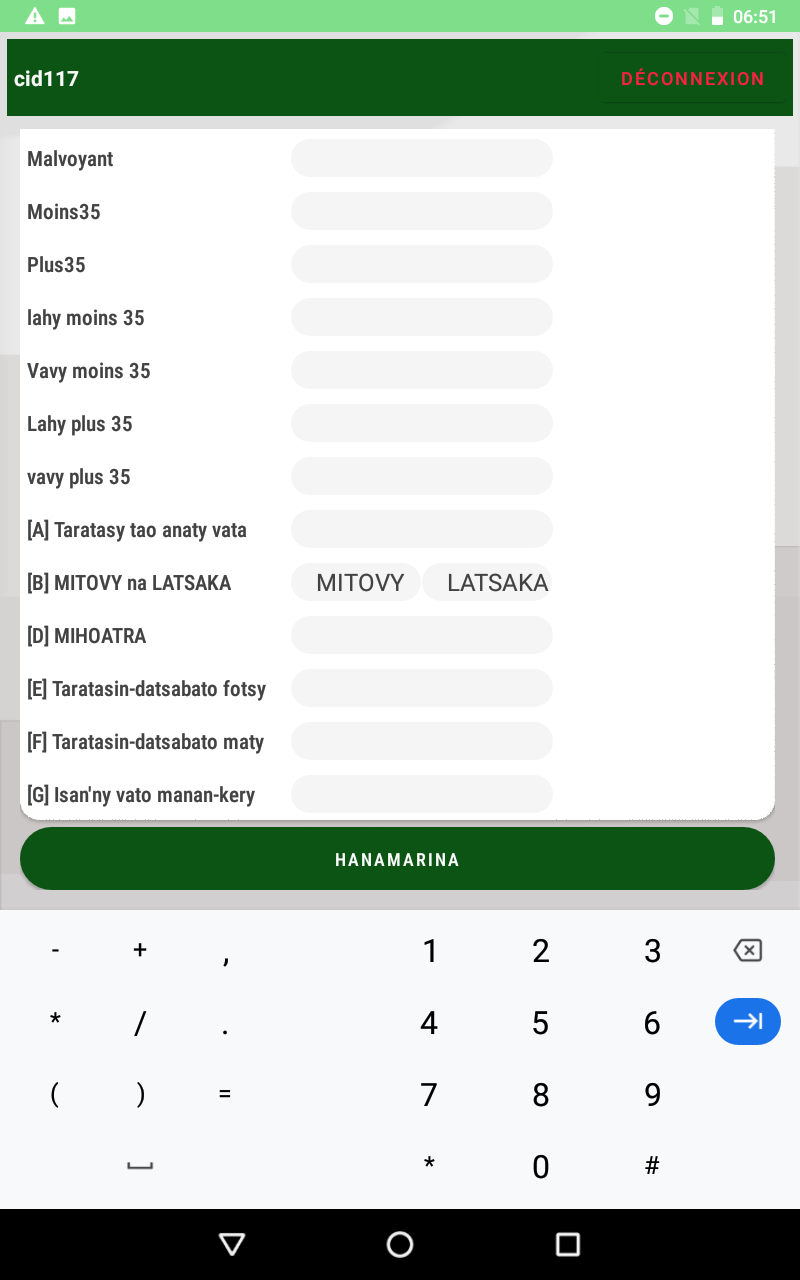
Nous pouvons voir avec les figures ci-dessous la version actuelle et en cours d’utilisation, des applications de gestion électorale :

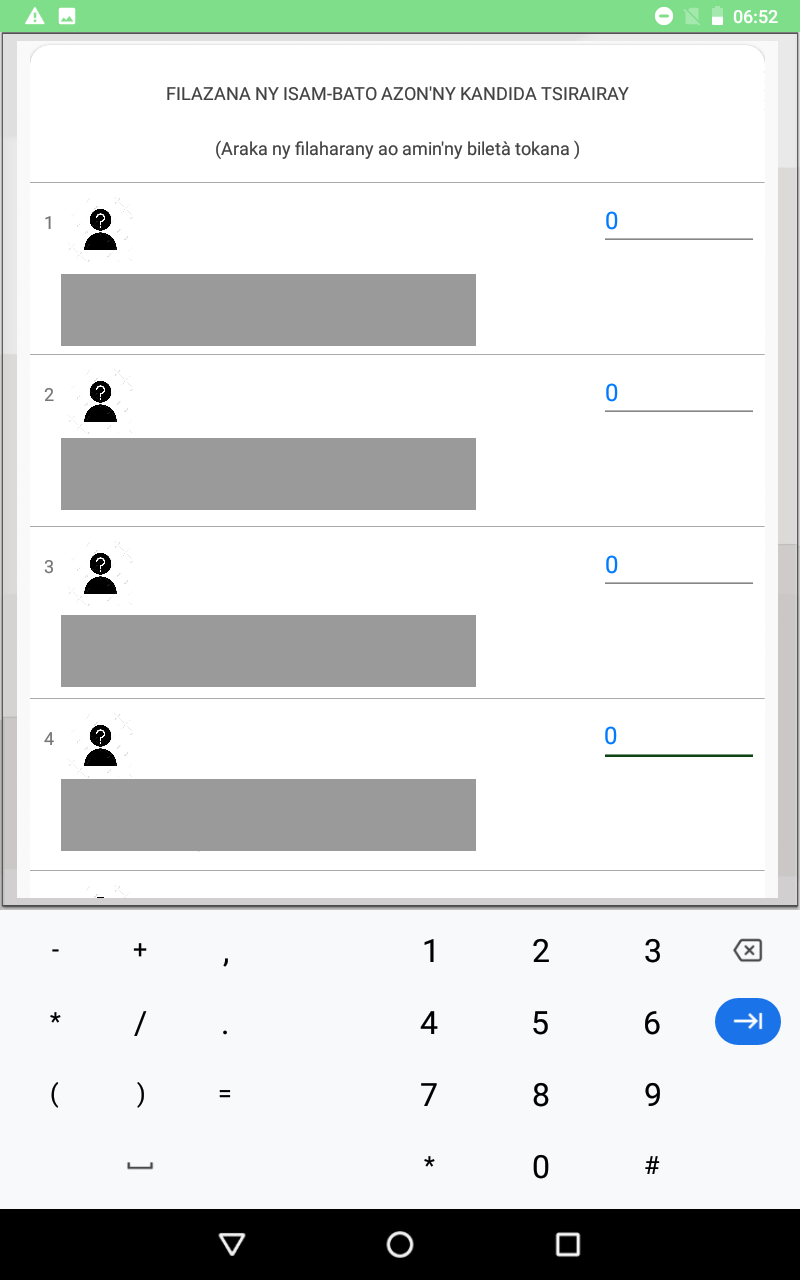
Figure 6: Page de consultation en temps réel des résultats récoltés et validés

Figure 7: Page exportation en PDF des résultats provisoires

Figure 8: Page de saisie et validation des résultats pour les opérateurs de saisie

Figure 9: Page exportation en PDF des situations de saisie

Figure 10: Interface de saisie partie 1

Figure 11: Interface de saisie partie 2

### Conception de l’application

Pourquoi doit-on faire la conception ? Pourquoi ne pas faire directement ?

Lister vos travaux, exemple :

* Nombre de tables créées
* Nombre de classes
* Nombre d’écrans
* Etc…

## Développement par fonctionnalité ou module

### Module ou Fonctionnalité numéro 1

### Module ou Fonctionnalité numéro 2

### Gestion des utilisateurs

* Gestion des utilisateurs
  + Description des fonctionnalités du module
  + Pourquoi a-t-on développé ce module pour l'entreprise
  + Scénario d'utilisation clé (1 ou 2)
    - Quel scénario ?
    - Dessin écran + explication
    - Importance
    - Comment ?
      * Pas forcément besoin d’un extrait du code source
      * Diagramme de séquence

## État d’Analyse et Statistiques

### État numéro 1

### État numéro 2

### Statistique numéro 1

Tableau 5 : Classement TIOBE 2014 de quelques langages de programmation (**exemple**)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Position Jan ‘14 | Position Jan ‘13 | Language | Share Jan ‘14 | Trend |
| 1 | 1 | Java | 26.2% | -0.6% |
| 2 | 2 | PhP | 13.2% | -1.6% |
| 3 | 6 | Python | 10.2% | +1.3% |
| 4 | 3 | C# | 9.6% | -0.4% |
| 5 | 4 | C++ | 8.9% | 0% |
| 6 | 5 | C | 8.1% | -0.2% |
| 7 | 7 | JavaScript | 7.6% | +0.3% |

Il ne s’agit pas de reproduire ce tableau, surtout si vous ne faites pas une comparaison entre les langages de programmation : ceci est un exemple de présentation de données statistiques.  
REMARQUEZ que **des données chiffrées doivent être datées et récentes**.

## Problèmes rencontrés et solutions

Il s’agit de **présenter ici les problèmes, les plus intéressants/importants**, soulevés durant le stage et les solutions que vous avez apporté.

**Il ne s’agit pas de raconter vos propres lacunes,** que vous avez dû résoudre pour réaliser le travail.

# Évaluation du projet et connaissances acquises

## Bilan pour l'entreprise

**A faire avec le contrôle de votre encadreur d’entreprise**

## Bilan personnel

N’insistez pas sur vos lacunes.

## Extension et évolution de l’application

# Conclusion

Doit inclure **au moins** 3 points :

* Les avantages et satisfactions exprimés par l’entreprise/institution d’accueil au vu de l’atteinte des objectifs fixés au début du stage,
* Vos impressions personnelles
* Une partie **perspective** d’extension du travail effectué.

**Insistez** alors sur les apports de votre travail par rapport à l’existant.

La conclusion doit tenir **obligatoirement** sur une page maximum.

# Bibliographie

Vous avez ici 3 exemples (livre, référence Web, thèse/mémoire) : veuillez suivre scrupuleusement les formats.

Un travail de mémoire comporte obligatoirement une recherche documentaire sur les problèmes et techniques abordés : **veillez à étoffer vos références pour montrer que vous avez fait correctement ce travail de recherche**.

DELLEY, A., FRANCIOLI, M., ZBINDEN, P., *Technologies d’accès aux réseaux*, Fribourg : Ecole d’ingénieurs et d’architectes de Fribourg, 2007. 220p.

Damien A., *Concevez votre réseau TCP/IP* [en ligne]. Disponible sur : https://openclassrooms.com/en/courses/6944606-concevez-votre-reseau-tcp-ip?archived-source=857447 (consultée le 31-05-2022)

GAUTHIER C. *Contribution à l'étude du fractionnement de l'aluminium libéré dans des solutions de sols forestiers : influence de la quantité et de la nature de la matière organique*. Thèse de doctorat d’université. Limoges : Université de Limoges, 2003.

# Annexe

S’il y en a : pour isoler des éléments techniques afin de laisser le mémoire plus lisible, surtout par des non spécialistes.

1. TIC : Technologies de l’Information et de la Communication [↑](#footnote-ref-2)