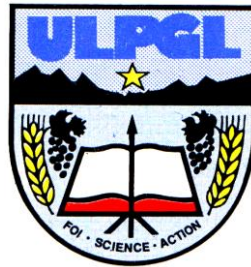


UNIVERSITE LIBRE DES PAYS DES GRANDS LACS
FACULTE DE SCIENCES ET TECHNOLOGIES
APPLIQUEES

DEPARTEMENT DE GENIE ELECTRIQUE ET INFORMATIQUE



BP. 368 GOMA

www.ulpgl.net

**CONCEPTION ET IMPLEMENTATION D'UN SYSTEME
DE VOTE EN LIGNE INTEGRANT LA TECHNOLOGIE
BLOCKCHAIN POUR RENFORCER LA SECURITE ET LA
TRANSPERANCE ELECTORALE EN RDC**

Par MUGISHO CIRINDYE Frederick

Travail présenté en vue de l'obtention du Diplôme
d'ingénieur civil en génie électrique et informatique

Option : Génie Informatique

Directeur : Prof. Dr. BARAKA MUSHAGE Olivier

Encadreur : Msc. Ir KRAME KADURA David

ANNEE ACADEMIQUE 2023 - 2024

Epigraphe

« La confiance est le problème fondamental des systèmes de vote conventionnels. »

Satoshi Nakamoto

Dédicace

A mes parents CIRINDYE KALENDE Chiza et NTAKWINJA Lucie

Remerciements

Nous remercions notre Dieu, le maître des temps et des circonstances, pour la protection à notre égard au cours de notre travail. Que l'honneur et la gloire lui soient rendus.

Nous remercions tout le corps administratif et académique de l'Université Libre des Pays des Grands Lacs pour son encadrement moral et intellectuel.

Nous remercions notre Directeur, le Prof. BARAKA MUSHAGE Olivier et notre Encadreur Msc. Ir KRAME KADURA David pour l'accompagnement afin que nous puissions réaliser ce travail.

Nous remercions nos parents, CIRINDYE KALENDE Chiza et NTAKWINJA Lucie, pour leur encadrement moral et intellectuel de qualité et leur soutien financier.

Nos remerciements à notre grande sœur bien-aimée Gorette CIRINDYE et son époux pour leur soutien tant moral que matériel.

Nos remerciements pour tous ceux qui nous ont soutenus de près ou de loin, d'une manière ou d'une autre.

MUGISHO CIRINDYE Frederick

Résumé

Ce mémoire porte sur la conception et la mise en place d'un système de vote en ligne qui utilise la technologie blockchain pour améliorer la sécurité et la transparence des élections en République Démocratique du Congo. L'objectif principal est de résoudre les problèmes actuels, tels que la manipulation des résultats, les fraudes et le manque de clarté dans le processus électoral. Pour y parvenir, nous avons choisi la blockchain, une technologie qui garantit l'intégrité des votes, empêche toute modification des données, tout en assurant l'anonymat des électeurs et la vérifiabilité des résultats. Le système développé permet de gérer les scrutins, d'enregistrer les électeurs et les candidats, et permet un vote à distance sécurisé. Les résultats montrent que l'utilisation de la blockchain renforce la confiance des électeurs et améliore considérablement la sécurité des élections. En outre, ce système facilite la participation des citoyens, y compris ceux des zones éloignées, tout en réduisant les coûts liés à l'organisation des élections. Au final, ce projet vise à moderniser et rendre plus crédible le processus électoral en RDC.

Mot clés : Système, Vote, Blockchain, Sécurité et Transparence.

Abstract

This report focuses on the design and implementation of an online voting system that uses blockchain technology to improve the security and transparency of elections in the Democratic Republic of the Congo. The main objective is to address current issues such as vote manipulation, fraud, and the lack of clarity in the electoral process. To achieve this, we have chosen blockchain, a technology that ensures the integrity of votes, prevents data modification, while guaranteeing voter anonymity and the verifiability of results. The developed system allows for the management of ballots, the registration of voters and candidates, and enables secure remote voting. The results show that the use of blockchain strengthens voter confidence and significantly improves the security of elections. Furthermore, this system facilitates citizen participation, including those from remote areas, while reducing the costs associated with organizing elections. Ultimately, this project aims to modernize and enhance the credibility of the electoral process in the DRC.

Key words: vote, system, blockchain, security, transparency.

Table des matières

Epigraphe	i
Dédicace.....	ii
Remerciements.....	iii
Résumé.....	iv
Abstract	v
Table des matières	vi
Liste des sigles et abréviations.....	x
Liste des tableaux.....	xi
Liste des figures	xii
0. Introduction générale	1
0.1. Contextes.....	1
0.2. Problématique.....	1
0.3. Questions de recherche.....	2
0.4. Hypothèses de travail	2
0.5. Choix et Motivation du Sujet	3
0.6. Objectifs de recherche	4
0.6.1. Objectif général.....	4
0.6.2. Les objectifs opérationnels/spécifiques	4
0.7. Méthodologie et délimitation du travail	4
0.7.1. Méthodes et techniques.....	4
0.7.2. Délimitation du travail	5
0.8. Subdivision du travail.....	5

Chapitre 1 . Etat de l’art du vote en ligne et de la technologie Blockchain	7
1.1 Introduction	7
1.2 Concepts de base de la recherche	7
1.2.1. Vote.....	7
1.3 Les différents moyens de vote.....	11
1.3.1 Vote à main levée.....	12
1.3.2 Vote papier	12
1.3.3 Vote par correspondance.....	12
1.3.4 Vote par procuration	12
1.3.5 Vote électronique	13
1.3.6 Vote en ligne	13
1.4 Analyse de la procédure actuelle des élections en RDC	13
1.4.1 Présentation de l’existant	13
1.4.2 Critique de l’existant.....	15
1.5 La sécurité informatique.....	16
1.6 Technologie blockchain	18
1.6.1 Définitions la blockchain	18
1.6.2 Concepts de base dans le domaine de la blockchain.....	19
1.6.3 Fonctionnement de la blockchain	25
1.6.4 Caractéristiques de la blockchain.....	28
1.6.5 Types de blockchain	29
1.6.6 Protocoles de blockchain	30
1.6.7 Applications de la Blockchain	32
1.6.8 Cadre du projet.....	33
1.6.9 Solution proposée.....	34
1.7 Conclusion partielle.....	35
Chapitre 2 . Conception de l’architecture logicielle	36
2.1 Introduction	36
2.2 Intégration de la Technologie Blockchain	36

2.2.1	Déploiement du scrutin	36
2.2.2	Processus de Vote	37
2.2.3	Vérification des Votes.....	37
2.2.4	Fermeture du scrutin	37
2.3	Présentation des acteurs	37
2.4	Spécifications des besoins	38
2.4.1	Spécifications fonctionnelles	38
2.4.2	Spécifications non fonctionnelles	40
2.5	Types de vote pris en charge par le système	41
2.6	Présentation des cas d'utilisation	42
2.6.1	Diagramme de cas d'utilisation	42
2.6.2	Documentation de différents cas d'utilisation	45
2.7	Diagramme de classe.....	58
2.8	Architecture globale	66
2.9	Conclusion partielle.....	67
Chapitre 3 . Implémentation du système et intégration de la technologie Blockchain		68
3.1	Introduction	68
3.2	Environnement de travail	68
3.2.1	Environnement matériel.....	68
3.2.2	Environnements logiciels.....	69
3.2.3	Technologie utilisée	71
3.3	Présentation des Interfaces	73
3.3.1	Accueil	73
3.3.2	Interface d'authentification.....	74
3.3.3	Créer un compte	75
3.3.4	Tableau de bord.....	76
3.3.5	Interface scrutins	77
3.3.6	Interface tableau de bord d'un scrutin	78

3.3.7	Interface types de candidature	79
3.3.8	Interface candidats du scrutin	80
3.3.9	Interface électeurs du scrutin	81
3.3.10	Interface charger les électeurs.....	82
3.3.11	Interface votes du scrutin	83
3.3.12	Interface résultats du scrutin	83
3.3.13	Interfaces utilisateurs	84
3.3.14	Interface chercher scrutin.....	85
3.3.15	Interface sélectionner candidat pour vote	86
3.3.16	Interface de visualisation du bulletin	86
3.3.17	Interface envoyer le bulletin	87
3.4	Présentation des tests.....	88
3.4.1	Test de performance sous charge	88
3.4.2	Test de résistance à une attaque DDoS	89
3.4.3	Test de brute forcing	90
3.5	Justification de la supériorité de la solution basée sur Blockchain.....	91
3.6	Conclusion partielle.....	91
	Conclusion générale.....	92
	Bibliographie.....	95
	ANNEXES	101
	Quelques autres interfaces de notre système	102
	Quelques parties du code	106
	Autres figures.....	110

Liste des sigles et abréviations

CENI	Commission Electeur Nationale Indépendante
RDC	République Démocratique du Congo
UML	Unified Modeling Language
UP	Unified Process

Liste des tableaux

Tableau 1 : Documentation du cas d'utilisation « S'authentifier »	45
Tableau 2 : Documentation du cas d'utilisation « Créer un compte ».....	46
Tableau 3 : Documentation du cas d'utilisation « Créer un scrutin ».....	47
Tableau 4 : Documentation du cas d'utilisation « Ajouter un type de candidature ».....	48
Tableau 5 : Documentation du cas d'utilisation « Ajouter un candidat ».....	49
Tableau 6 : Documentation du cas d'utilisation « Ajouter un électeur ».....	50
Tableau 7 : Documentation du cas d'utilisation « Ajouter une proposition ou décision »	51
Tableau 8 : Documentation du cas d'utilisation « Déployer le scrutin ».....	52
Tableau 9 : Documentation du cas d'utilisation « Chercher un scrutin »	54
Tableau 10 : Documentation du cas d'utilisation « Voter »	55
Tableau 11 : Documentation du cas d'utilisation « Publier les résultats du scrutin »	56
Tableau 12 : Documentation du cas d'utilisation « Ajouter un utilisateur »	57
Tableau 13 : Dictionnaire de donnée	59

Liste des figures

Figure 1 : Registre de la blockchain	18
Figure 2 : Les blocs de la Blockchain avec les hashes	24
Figure 3 : Etapes de fonctionnement la Blockchain	25
Figure 4 : Système centralisé et système décentralisé	28
Figure 5 : Schéma présentatif du cadre de projet.....	33
Figure 6 : Diagramme de cas d'utilisation d'un administrateur	43
Figure 7 : Diagramme de cas d'utilisation d'un électeur et un internaute	44
Figure 8 : Diagramme de classe du système.....	65
Figure 9 : Architecture globale	66
Figure 10 : Visual Studio Code (VS Code)	69
Figure 11 : Remix Ethereum IDE	70
Figure 12 : Ganache	71
Figure 13 : Accueil.....	74
Figure 14 : Interface d'authentification	75
Figure 15 : Créer un compte	76
Figure 16 : Tableau de bord	77
Figure 17 : Interface scrutins	78
Figure 18 : Interface tableau de bord d'un scrutin.....	79
Figure 19 : Interface types de candidature	80
Figure 20 : Interface candidats du scrutin.....	81
Figure 21 : Interface électeurs du scrutin.....	82
Figure 22 : Interface charger les électeurs	83
Figure 23 : Interface votes du scrutin	83
Figure 24 : Interface résultats du scrutin.....	84
Figure 25 : Interfaces utilisateurs.....	85

Figure 26 : Interface chercher scrutin	85
Figure 27 : Interface sélectionner candidat pour vote.....	86
Figure 28 : Interface de visualisation du bulletin.....	87
Figure 29 : Interface envoyer le bulletin.....	88
Figure 30 : Test de performance sous charge	89
Figure 31 : Test de résistance à une attaque DDoS	90

0. Introduction générale

0.1. Contextes

Au fur et à mesure que la technologie progresse, elle s'impose dans presque tous les aspects de notre vie quotidienne. Et avec l'avènement de l'ère numérique, les méthodes traditionnelles de gestion sont devenues obsolètes et nécessitent une révision ou une amélioration pour s'adapter aux nouvelles technologies de l'information et de la communication. C'est particulièrement pertinent dans le contexte de la démocratie représentative qui est un régime dans lequel le pouvoir est attribué par le peuple à des représentants élus qui l'exercent en son nom [1], où on observe une transition vers des systèmes de vote en ligne.

0.2. Problématique

Alors que la technologie est omniprésente dans diverses sphères de notre vie, les systèmes de vote, particulièrement en République Démocratique du Congo (RDC), sont toujours entravés par des processus manuels obsolètes.

Voici quelques problèmes que les applications de vote actuelles peuvent renfermer :

- ❖ Manipulation des votes : Les systèmes de vote électronique actuels peuvent être vulnérables aux attaques informatiques, permettant la modification ou la suppression des votes.
- ❖ Intégrité des données : La possibilité de manipuler les bases de données de vote sans détection peut compromettre l'intégrité des résultats électoraux.
- ❖ Violation de la confidentialité des électeurs : Les systèmes de vote électronique peuvent parfois exposer les choix des électeurs, compromettant ainsi le principe du vote secret.
- ❖ Absence de vérifiabilité : Les électeurs et les observateurs indépendants peuvent ne pas être en mesure de vérifier que les votes ont été correctement enregistrés et comptés.

La blockchain, une technologie innovante, a le potentiel de renforcer la sécurité et la transparence des systèmes de vote en ligne. Cependant, son application et son implémentation restent un défi.

0.3. Questions de recherche

Les questions suivantes vont faire l'objet de notre travail de recherche :

1. De quels moyens avons-nous besoin pour créer un système de vote en ligne robuste utilisant la technologie blockchain ?
2. Quelle blockchain choisir pour développer notre solution et pourquoi ce choix ?
3. Quels sont les défis liés à l'implémentation d'un système de vote en ligne basé sur la blockchain et comment les surmonter ?

Ce travail de recherche vise à répondre à ces questions.

0.4. Hypothèses de travail

1. Un système de vote en ligne basé sur la blockchain nécessiterait une infrastructure technologique robuste, incluant des serveurs sécurisés et une connectivité Internet fiable, afin d'assurer la disponibilité et la sécurité des votes.
2. Le choix de la blockchain Ethereum, avec ses capacités de smart contract solides et sa large communauté de développeurs, serait optimal pour notre système de vote en ligne.
3. Notre système devrait relever des défis de scalabilité pour gérer un grand volume de transactions simultanées, assurer la sécurité des données contre les cyberattaques et s'intégrer avec des infrastructures réseau limitées.
4. Pour le développement de notre système, nous devons mettre en place un smart contract et un backend sécurisé, concevoir une interface utilisateur intuitive, garantir une infrastructure réseau fiable et évolutive, et intégrer des mécanismes de vérification et de validation des votes

0.5. Choix et Motivation du Sujet

Avec la montée des cyber-attaques et des fraudes électorales, le développement d'une plateforme de vote sécurisée et transparente est une priorité. C'est pourquoi la conception et l'implémentation d'un système de vote en ligne utilisant la technologie Blockchain en RDC est un sujet pertinent et d'actualité.

Les avantages qui suivent nous motivent à choisir ce sujet, notamment :

- ❖ **Avantage d'ordre scientifique** : Il offre une opportunité de mettre en pratique les concepts de génie logiciel et de sécurité informatique dans un cas réel, ce qui enrichit la recherche scientifique dans ces domaines.
- ❖ **Avantage d'ordre économique** : La transition vers un système de vote en ligne intégrant la technologie Blockchain peut réduire les coûts associés à l'organisation d'élections, en réduisant le besoin de matériel physique et de personnel supplémentaire pour la supervision des bureaux de vote.
- ❖ **Avantage d'ordre démocratique** : En renforçant la sécurité, la transparence et l'intégrité du processus électoral, ce système contribue à garantir des élections libres, justes et crédibles, renforçant ainsi la démocratie et la confiance des citoyens dans le système politique.
- ❖ **Avantage d'accessibilité** : La facilité d'accès au vote en ligne permet à un plus grand nombre de citoyens, y compris ceux vivant dans des zones reculées ou ayant des difficultés de mobilité, de participer au processus électoral.
- ❖ **Avantage de modernisation** : L'adoption de cette technologie innovante montre l'engagement de la RDC envers des élections modernes et transparentes, renforçant ainsi son image sur la scène internationale et sa crédibilité en tant que démocratie émergente.

0.6. Objectifs de recherche

0.6.1. Objectif général

L'objectif principal de cette recherche est de concevoir et d'implémenter un système de vote en ligne utilisant la technologie blockchain pour renforcer la sécurité et la transparence des élections en République Démocratique du Congo.

0.6.2. Les objectifs opérationnels/spécifiques

Plus spécifiquement, pour réaliser l'objectif global de ce travail, nous comptons :

- ❖ Analyser le processus de vote actuel en RDC pour identifier ses insuffisances et failles.
- ❖ Étudier le potentiel de la technologie blockchain dans le renforcement de la sécurité et de la transparence des systèmes de vote en ligne.
- ❖ Concevoir un système de vote en ligne basé sur la blockchain pour les élections en RDC.
- ❖ Implémentation du backend
- ❖ Développer le smart contract de vote
- ❖ Implémenter le frontend du système
- ❖ Intégrer le smart contract dans notre backend
- ❖ Intégrer l'API dans notre frontend
- ❖ Évaluer les avantages et les défis de l'implémentation de ce système

0.7. Méthodologie et délimitation du travail

0.7.1. Méthodes et techniques

Pour atteindre nos objectifs de recherche et concevoir efficacement notre système de vote en ligne basé sur la technologie blockchain, nous avons employé une combinaison de méthodes et techniques :

- ❖ **Méthode analytique** : cette méthode a été employée pour analyser minutieusement les procédures de vote existant, ce qui a facilité la conception du nouveau système tout en préservant largement sa structure originelle.

- ❖ **Le processus Unifié (UP : Unified Process)** : Cette méthode permet l'implémentation et le développement majoritairement utilisées dans le développement informatique. Ce processus utilise le langage UML qui est sa partie intégrante.
- ❖ **La technique expérimentale** : cette technique a été appliquée au processus précédemment décrit afin de tester et évaluer de manière systématique les différentes modifications apportées aux procédures de vote, en vue d'optimiser la conception du nouveau système sans altérer substantiellement sa structure de base.
- ❖ **La technique d'analyse documentaire** : grâce à cette technique, nous avons pu mener des investigations approfondies sur le système en place, ainsi que sur la manière d'intégrer efficacement le nouveau système en exploitant la technologie blockchain. Cette approche nous a permis de mieux comprendre les exigences et les implications de l'adoption de la blockchain, tout en identifiant les meilleures pratiques pour son intégration réussie dans notre environnement.

0.7.2. Délimitation du travail

Notre travail se concentre principalement sur l'implémentation d'un système de vote en ligne intégrant la technologie blockchain pour les élections en République Démocratique du Congo. Nous nous limiterons à la gestion des élections présidentielles, législatives et les votes de propositions (où un électeur accepte, refuse ou s'abstienne).

0.8. Subdivision du travail

Hormis l'introduction et la conclusion générales ce travail est constitué :

- ❖ **Le Chapitre 1 porte sur l'Etat de l'art du vote en ligne et de la technologie Blockchain** : Dans ce chapitre nous aborderons les bases du système de vote traditionnel, mettant en lumière ses processus et ses défis, tout en présentant les principes fondamentaux de la technologie blockchain.
- ❖ **Le Chapitre 2 porte sur la Conception de l'architecture logicielle** : Dans ce chapitre nous allons nous concentrer sur la conception de l'architecture logicielle et nous

détaillerons les choix d'architecture, les composants clés et les flux de données pour assurer la robustesse et la sécurité du système.

- ❖ **Le Chapitre 3 porte sur l'Implémentation du système et intégration de la technologie Blockchain** : Dans ce chapitre nous allons détailler les étapes de développement (les outils utilisés), d'intégration et de déploiement du système, en mettant l'accent sur les défis techniques rencontrés et les solutions adoptées.

Chapitre 1. Etat de l'art du vote en ligne et de la technologie Blockchain

1.1 Introduction

Dans ce chapitre, nous allons présenter quelques généralités concernant les concepts de base, les méthodes et les outils nécessaires pour la réalisation de notre système.

C'est dans cette partie que nous présenterons aussi le système actuellement utilisé par la RDC en y apportant quelques critiques ; nous présenterons aussi nos propositions et une analyse du système futur.

1.2 Concepts de base de la recherche

Dans cette partie nous allons nous focaliser sur les concepts fondamentaux sur lesquels se forme notre sujet de recherche

1.2.1. Vote

Le vote est un pilier central des démocraties modernes, donnant à chaque citoyen la possibilité de s'impliquer directement dans la gouvernance de son pays. Ce processus est crucial pour assurer la légitimité et la représentativité des institutions politiques. Il existe différents systèmes électoraux et modalités de vote, chacun ayant ses propres caractéristiques. Cette section examine les divers types de systèmes électoraux, les différentes modalités de vote, l'importance du vote, ainsi que les défis qui y sont associés.

A. Les modalités de vote

Le vote se fait selon plusieurs modalités dont les plus importantes sont [2] :

- ❖ Le vote à main levée qui est caractérisé par le fait que les votants émettent leurs avis pour ou contre en levant la main.

- ❖ Le vote en se mettant debout qui est caractérisé par le fait que les votants émettent leurs avis pour ou contre en se mettant debout.
- ❖ Le vote par acclamation qui est caractérisé par le fait que les votants émettent leurs avis pour ou contre par acclamation.
- ❖ Le vote par boules ou bulletins qui est caractérisé par le fait que les votants émettent leurs avis pour ou contre par des boules qui éventuellement peuvent avoir des couleurs différentes ou par les bulletins qui, comme les boules, peuvent avoir des couleurs différentes.

B. Quelques concepts du contexte de vote

❖ **Election**

Une élection est un processus formel de prise de décision par lequel une population choisit des individus pour occuper des fonctions publiques. Les élections sont un élément fondamental des systèmes démocratiques, permettant aux citoyens de sélectionner leurs représentants et d'influencer les politiques et la gouvernance de leur pays ou région. [3]

❖ **Scrutin**

Le scrutin peut être défini comme l'ensemble des opérations entourant le vote. [3]

❖ **Candidat**

Un candidat est une personne qui se présente lors d'une élection. [3]

❖ **Candidature**

La candidature est l'action de se porter candidat. [3]

❖ **Campagne électorale**

La campagne électorale est un ensemble d'activités menées par un candidat ou un parti politique pendant la période électorale en vue de recueillir des votes le jour du scrutin. [3]

❖ **Suffrage universel**

Le suffrage universel est un suffrage où le droit de vote est reconnu à pratiquement tous les citoyens, sous réserve de quelques conditions objectives jugées non discriminatoires. [3]

❖ **Droit de vote**

Le droit de vote est le droit de participer au choix de représentants ou à une décision en votant lors d'une élection ou d'un référendum. [3]

❖ **Partis politiques**

Les partis politiques sont des organisations qui promeuvent une certaine vision politique et qui présentent des candidats aux élections en vue de gouverner ou de participer à l'exercice du pouvoir. [3]

❖ **Électeurs**

Les électeurs sont des personnes qui participent à une élection en exprimant leur voix. [3]

❖ **Fraude électorale**

La fraude électorale est une infraction aux lois électorales commise par une personne et pouvant entraîner la perte de certains droits électoraux pour une période déterminée. [3]

❖ **Bureau de vote**

Un bureau de vote est un espace aménagé pour le déroulement d'un vote, comprenant notamment une table où prend place le personnel électoral, une urne et, à proximité, un isoloir. [3]

❖ **Urne**

Une urne est une boîte munie d'une fente, dans laquelle les votants déposent leur bulletin de vote. [3]

❖ **Isoloir**

Un isoloir est un dispositif de séparation physique destiné à préserver le secret lorsqu'un électeur exerce son droit de vote. [3]

❖ **Bulletin de vote**

Un bulletin de vote est un document par lequel un électeur exprime son choix à l'égard des candidats en lice ou d'une question qui lui est posée. [3]

❖ **Abstention**

L'abstention est le fait de ne pas participer au vote. [3]

❖ **Suffrage**

Le suffrage est un vote exprimant un choix et pouvant être comptabilisé en faveur d'un candidat ou d'une option lors d'une élection ou d'une consultation. [3]

❖ **Vote blanc**

Le vote blanc est un vote n'exprimant aucun choix et par lequel un électeur manifeste son rejet d'une offre politique donnée. [3]

❖ **Dépouillement des votes**

Le dépouillement des votes est l'opération débutant après la clôture du scrutin, qui consiste à procéder au dénombrement des votes dans le but de compiler les résultats d'une élection. [3]

C. Types de scrutin

❖ Scrutin majoritaire

Le scrutin majoritaire permet l'élection de celui ou de ceux qui ont obtenu le plus de voix. Quand il s'agit d'attribuer un siège, on dit que le scrutin est uninominal. Quand plusieurs sièges sont à pourvoir, on parle de scrutin plurinominal. [4]

❖ Scrutin proportionnel

Les sièges sont attribués en proportion du nombre de voix obtenues par chaque liste de candidats, favorisant une représentation proportionnelle des différents partis politiques. [4]

❖ Scrutin mixte

Il intègre à la fois des éléments de scrutin majoritaire et proportionnel, ce système permet d'élire à la fois des candidats individuels et des représentants selon des quotas de voix pour les partis. [4]

1.3 Les différents moyens de vote

Dans les processus démocratiques, les moyens de vote sont divers pour permettre aux citoyens de s'exprimer. Chacune de ces méthodes présente ses propres particularités en ce qui concerne la confidentialité, la sécurité et l'accès. Chacune de ces méthodes, qu'il s'agisse du vote traditionnel à bulletin papier ou des systèmes numériques contemporains, vise à répondre aux attentes des électeurs tout en garantissant l'intégrité des résultats. Nous allons mettre l'accent sur le vote en ligne dans notre étude. [5]

Voici une liste de quelques modes de scrutins :

1.3.1 Vote à main levée

Le vote à main levée implique d'exprimer un choix de manière publique en levant la main. Son utilisation est fréquente lors de réunions ou d'assemblées afin de prendre des décisions immédiates. Néanmoins, il n'est pas confidentiel et peut avoir un impact sur les électeurs en raison d'un effet de groupe.

1.3.2 Vote papier

La méthode traditionnelle de vote papier consiste à faire cocher un bulletin qui est ensuite déposé dans une urne. Le dépouillement manuel ou électronique est possible avec cette méthode, cependant, il peut être coûteux et lent, nécessitant des ressources humaines importantes pour la gestion et le dépouillement.

1.3.3 Vote par correspondance

Les électeurs peuvent recevoir leur bulletin de vote par courrier et le renvoyer une fois qu'il a été complété. Il est fréquemment employé pour les individus incapables de se déplacer. Malgré sa praticité et sa facilité d'accès, il y a des risques de retard dans la réception des bulletins, de perte de courrier ou de fraude électorale.

1.3.4 Vote par procuration

Le vote par procuration est une forme de vote où un électeur peut confier son vote à une autre personne en cas de difficulté à se rendre au bureau de vote. Cette approche est utile pour les individus souffrant de maladies, absents ou handicapés, mais elle repose sur la confiance, ce qui peut engendrer des problèmes de fiabilité ou de fraude si la procuration n'est pas correctement dirigée.

1.3.5 Vote électronique

Le vote numérique consiste à utiliser des machines ou des systèmes numériques pour enregistrer les suffrages. Il est possible d'effectuer le vote sur place dans des bureaux de vote équipés ou à distance à l'aide de terminaux. Les bénéfices comprennent la facilité de traitement des résultats et une diminution des erreurs commises par les individus. Cependant, cela suscite des inquiétudes concernant la protection des données, les attaques cybernétiques et la transparence.

1.3.6 Vote en ligne

Le scrutin en ligne offre aux électeurs la possibilité de voter à travers une plateforme numérique accessible en ligne. Ce système présente une grande souplesse et offre la possibilité de participer à distance, notamment pour les expatriés ou ceux qui ont des contraintes de temps. Néanmoins, il est très exposé à des problèmes de sécurité, comme les attaques informatiques, la manipulation des résultats et la protection des données personnelles, ce qui demande des mesures de sécurité renforcées afin de garantir la confiance du public. [6]

1.4 Analyse de la procédure actuelle des élections en RDC

1.4.1 Présentation de l'existant

Le processus électoral en RDC se fait en 5 étapes dont : [7]

❖ Identification et enrôlement des électeurs

La première étape cruciale du processus électoral est l'identification et l'enrôlement des électeurs. Cela commence par la vérification des conditions requises pour chaque électeur, telles que l'âge, la nationalité et la résidence. Une fois ces critères confirmés, l'électeur est enregistré sur les listes électorales et se voit remettre une carte d'électeur, qui lui permettra de participer au scrutin.

❖ **Campagne électorale**

Pendant la campagne électorale, les candidats et les partis politiques mettent en avant leurs programmes et leurs projets de société pour convaincre les électeurs de voter pour eux. C'est un moment où ils organisent des rassemblements publics, des débats, et utilisent divers médias comme la radio, la télévision et les réseaux sociaux pour partager leurs idées et leurs visions. Ils rencontrent également les électeurs directement sur le terrain pour comprendre leurs préoccupations et leurs aspirations. Le but ultime est de gagner le soutien populaire en démontrant leur capacité à répondre aux besoins de la société et à résoudre ses problèmes.

❖ **Votes**

L'élection proprement dite permet aux électeurs de se rendre dans les bureaux de vote pour exprimer leur choix parmi les candidats en lice. C'est un moment où chaque citoyen exerce son droit démocratique en présentant sa carte d'électeur, recevant un bulletin de vote, marquant ses préférences dans l'isoloir, et déposant finalement son bulletin dans l'urne.

❖ **Dépouillement des votes**

À la fermeture du bureau de vote, les votes sont effectivement comptabilisés pour chaque candidat. Sous la supervision des témoins désignés par les candidats, les scrutateurs ou le personnel électoral ouvrent les urnes scellées et commencent à compter les bulletins de vote un par un. Les témoins de chaque candidat sont présents pour assurer la transparence du processus et vérifier que chaque vote est correctement attribué au candidat choisi par l'électeur.

❖ **Proclamation des résultats**

Une fois que les votes sont dépouillés, la CENI procède à la publication des résultats préliminaires des candidats. Ensuite, on attend les réclamations des candidats mécontents des résultats préliminaires. Après avoir examiné toutes les réclamations et assuré la résolution des problèmes éventuels, la CENI publie alors les résultats finaux. Ces résultats finaux représentent le verdict définitif du processus électoral et sont généralement rendus publics pour assurer une totale transparence.

1.4.2 Critique de l'existant

Après avoir présenté et décrit les différentes étapes du vote en RDC, nous pouvons établir un diagnostic des failles potentielles qui peuvent survenir à chaque étape :

❖ Identification et enrôlement des électeurs

Nous pouvons remarquer qu'à chaque élection, les électeurs doivent être enregistrés de nouveau, ce qui entraîne un risque de fraudes telles que les inscriptions multiples. En effet, les informations d'enrôlement des électeurs ne sont pas partagées instantanément avec les autres centres d'enrôlement, rendant le système vulnérable aux doublons et aux irrégularités.

❖ Votes

Certains électeurs peuvent éprouver des difficultés à se rendre aux bureaux de vote, notamment en raison de l'éloignement ou de l'absence de moyens de transport. Le jour du scrutin, divers obstacles tels que la fraude électorale, les pressions sur les électeurs, l'insuffisance de matériel électoral ou des erreurs dans les listes peuvent entraver le bon déroulement du vote.

❖ **Dépouillement des votes**

Un dépouillement fastidieux et non transparent, avec des erreurs de comptage et un manque de supervision adéquate, peut susciter des doutes et des contestations, compromettant ainsi la confiance des électeurs dans les résultats.

❖ **Proclamation des résultats**

Nous pouvons observer des retards dans la publication des résultats, ce qui peut générer des tensions, des frustrations parmi les électeurs et des soupçons de manipulation des résultats, compromettant ainsi la crédibilité et la transparence du processus électoral.

1.5 La sécurité informatique

La sécurité informatique c'est l'ensemble des moyens mis en œuvre pour réduire la vulnérabilité d'un système contre les menaces accidentelles ou intentionnelles. L'objectif de la sécurité informatique est d'assurer que les ressources matérielles et/ou logicielles d'un parc informatique sont uniquement utilisées dans le cadre prévu et par des personnes autorisées. Il convient d'identifier les exigences fondamentales en sécurité informatique, qui caractérisent ce à quoi s'attendent les utilisateurs de systèmes informatiques au regard de la sécurité : [8]

❖ **La confidentialité**

Seules les personnes habilitées doivent avoir accès aux données. Toute interception ne doit pas être en mesure d'aboutir, les données doivent être cryptées, seuls les acteurs de la transaction possèdent la clé de compréhension.

❖ **L'intégrité**

Il faut garantir à chaque instant que les données qui circulent sont bien celles que l'on croit, qu'il n'y a pas eu d'altération (volontaire ou non) au cours de la communication. L'intégrité des données doit valider l'intégralité des données, leur précision, l'authenticité et la validité.

❖ **La disponibilité**

Il faut s'assurer du bon fonctionnement du système, de l'accès à un service et aux ressources à n'importe quel moment. La disponibilité d'un équipement se mesure en divisant la durée durant laquelle cet équipement est opérationnel par la durée durant laquelle il aurait dû être opérationnel.

❖ **La non-répudiation**

Une transaction ne peut être niée par aucun des correspondants. La non-répudiation de l'origine et de la réception des données prouve que les données ont bien été reçues. Cela se fait par le biais de certificats numériques grâce à une clé privée.

❖ **L'authentification**

Elle limite l'accès aux personnes autorisées. Il faut s'assurer de l'identité d'un utilisateur avant l'échange de données. Bref, on mesure la sécurité d'un système entier à la sécurité du maillon le plus faible. Ainsi, si tout un système est sécurisé techniquement mais que le facteur humain, souvent mis en cause, est défaillant, c'est toute la sécurité du système qui est remise en cause.

1.6 Technologie blockchain

1.6.1 Définitions la blockchain

Sur cette illustration nous présentons un registre de la blockchain [9]

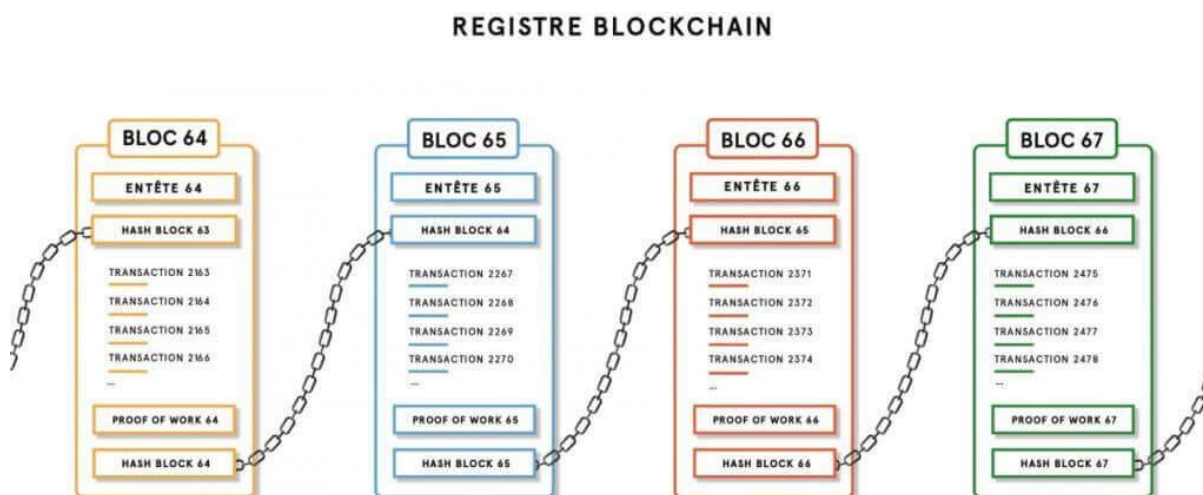


Figure 1 : Registre de la blockchain

Une blockchain permet de stocker des données de manière décentralisée et sécurisée. Les données sont répliquées au sein d'un réseau d'utilisateurs n'ayant pas besoin de se faire confiance directement entre eux, cette confiance émanant directement de la manière dont fonctionne le système. Pour les monnaies électroniques, cette base de données décentralisée représente le montant de cryptomonnaie que possède chaque utilisateur. Pendant que certains développeurs essayaient de rendre les cryptomonnaies plus fiables, plus anonymes, moins énergivores, d'autres ont vu dans la blockchain une technologie capable de faire plus, ou pour le moins, de faire autre chose. [10]

Un registre partagé et immuable qui facilite le processus d'enregistrement des transactions et de suivi des actifs dans un réseau d'entreprise. [11]

Une base de données distribuée et décentralisée qui enregistre des transactions de manière sécurisée et immuable. Elle est composée de blocs qui sont liés entre eux par des fonctions cryptographiques. [12]

1.6.2 Concepts de base dans le domaine de la blockchain

❖ **Block** : est un ensemble horodaté et validé de transactions. Ces transactions sont généralement des transferts de cryptomonnaies, mais peuvent aussi constituer des interactions avec des contrats ou contenir des données arbitraires. Les blocs sont liés les uns aux autres de sorte à former une structure linéaire appelée la chaîne de blocs, ou blockchain. [13]

Les blocs sont construits et chaînés selon des règles précises, appelées règles de consensus, qui sont définies par le réseau. Par exemple, les blocs ont généralement une taille limitée par ces règles pour empêcher les attaques par déni de service (DDoS) et pour faire en sorte que le réseau reste décentralisé. [13]

❖ **La Chaîne**

Un hash qui relie un bloc à un autre, les enchainant mathématiquement ensemble. C'est l'un des concepts de la blockchain le plus difficile à comprendre. C'est aussi la magie qui colle des blocs ensemble et leur permet de créer une confiance mathématique. Le hash dans la blockchain est créé à partir des données qui se trouvaient dans le bloc précédent. Le hash est une empreinte digitale de ces données qui verrouille les blocs dans l'ordre et dans le temps. [8]

❖ **Le réseau**

Le réseau est composé de « nœud plein » (full nodes). Songez-y comme à un ordinateur exécutant un algorithme sécurisant le réseau. Chaque nœud contient un enregistrement complet de toutes les transactions qui ont déjà été enregistrées dans cette blockchain. [8]

- ❖ **Nœud** : est une composante essentielle au bon fonctionnement d'une blockchain. Il s'agit d'un ordinateur participant au réseau blockchain, jouant un rôle crucial dans la sécurité, la stabilité et la validation des transactions.

Un nœud ou Node est un ordinateur qui participe au réseau blockchain en validant et en relayant des transactions. Les nœuds maintiennent une copie complète de la blockchain. [14]

La plupart des blockchains fonctionnent en mettant en commun les capacités de plusieurs types de nœuds, il existe différents types des nœuds blockchain dont nous pouvons citer :

- **Full node, ou nœud complet** : Selon la documentation de Bitcoin Core par Satoshi Nakamoto, « **Un nœud complet** est un programme qui valide entièrement les transactions et les blocs. Presque tous les nœuds complets soutiennent également le réseau en acceptant les transactions et les blocs d'autres nœuds complets, en validant ces transactions et ces blocs, puis en les relayant à d'autres nœuds complets. » [14]

Le défaut principal de ces nœuds est la mémoire nécessaire pour assurer leur bon fonctionnement, ces derniers devant stocker de grandes quantités d'informations

- **Les nœuds réduits (pruned nodes)** : Un pruned node est un nœud qui possède les mêmes propriétés qu'un nœud complet, à ceci près qu'il ne possède pas l'historique de la blockchain. Par conséquent, il est donc beaucoup plus léger.

Ainsi, un nœud réduit ne conserve que les composantes essentielles d'un nœud afin d'être en mesure d'effectuer les différents processus de vérification au sein de la blockchain. Et pour ce faire, il ne conserve qu'une

partie conséquemment réduite de l'historique de la blockchain concernée.

[14]

- **Les nœuds légers (light nodes) :** Les light nodes sont des versions atrophiées des nœuds complets dans la mesure où ils ne conservent qu'une partie extrêmement réduite de l'historique de la blockchain, à tel point qu'ils sont plus légers qu'un nœud réduit.

Toutefois, étant donné qu'ils ne stockent qu'une partie infime de la blockchain, les nœuds légers ne sont pas capables d'assurer la sécurité du réseau à eux seuls, c'est pourquoi ils sont eux-mêmes connectés aux nœuds complets afin d'obtenir les données complètes des blocs. [14]

- **Les nœuds archiveurs (archive nodes) :** Une archive node, ou nœud archiveur, bénéficie des mêmes capacités qu'un nœud complet, mais son utilisation sera privilégiée dans des situations où il sera nécessaire de remonter particulièrement loin dans l'historique d'une blockchain. Ce type de nœud est par exemple utilisé pour consulter des données au-delà du 128e bloc en partant du dernier validé sur la blockchain Ethereum, qui est d'ailleurs celle qui fait le plus appel à ce type de nœuds.

Un nœud archiveur aura une taille naturellement conséquente dans la mesure où il doit héberger la totalité de l'historique de la blockchain. Notons également que les archive nodes ne sont pas nécessairement tenus de participer au processus de validation d'une blockchain, ils peuvent simplement être utilisés en tant que bibliothèque à consulter. [14]

- **Les mining nodes :** Les nœuds de minage, utilisés par exemple pour le minage de Bitcoin, sont principalement utilisés pour résoudre les équations ultra complexes inhérentes à la création de nouveaux blocs. Ils sont interconnectés et sont à dissocier des nœuds classiques. Bien qu'ils en aient

la possibilité, ils ne conservent que rarement l'historique de la blockchain, puisque cela n'influe pas sur leur tâche de travail.

Le minage étant une activité onéreuse en termes de temps et de matériel, les participants au bon fonctionnement du processus de minage sont susceptibles d'être éligibles à des récompenses, contrairement aux individus faisant tourner des nœuds complets (dans le cas du Bitcoin). [14]

- ❖ **Gaz** : une unité qui représente le temps de travail informatique nécessaire à l'utilisation d'un smart contract sur le réseau Ethereum. Cela représente le prix à payer pour pouvoir utiliser le smart contract. C'est l'équivalent des fees de Bitcoin. Les gas servent à rétribuer les mineurs. [15]

Il fait aussi référence au frais d'une transaction sur la Blockchain

- ❖ **Contrats intelligents (Smart Contracts)** : Les contrats intelligents sont des programmes numériques stockés sur une blockchain qui s'exécutent automatiquement lorsque des conditions prédéfinies sont remplies. Ils permettent d'automatiser des accords sans intermédiaire, garantissant une exécution rapide, sécurisée et transparente. Principalement utilisés sur des blockchains comme Ethereum, ces contrats sont écrits dans des langages de programmation comme Solidity et permettent de développer des applications décentralisées dans divers domaines, tels que la finance, la logistique ou le vote en ligne. [16]

Chaque nœud du réseau blockchain conserve une copie du contrat et exécute son code lors des transactions, garantissant ainsi que les termes du contrat sont respectés sans nécessiter d'autorité centrale. Cela permet aux contrats intelligents de fonctionner de manière sécurisée et fiable, même dans des contextes financiers complexes. [9]

❖ **Fonction de hachage** : Une fonction de hachage $H(x)$ est une fonction qui mappe une entrée de longueur arbitraire x sur une sortie de longueur fixe appelé hash. Elles sont irréversibles.

Impossible de trouver :

- ✓ Entrée x telle que $H(x) = h$ (un hachage spécifique)
- ✓ Entrée y tel que $y \neq x$ et $H(y) = H(x)$, avec un x donné.
- ✓ Couples d'entrées $\{x, y\}$ s.t. $H(x) = H(y)$ (résistance à la collision)

Propriétés

- ✓ Déterminisme : La même entrée produit toujours la même sortie.
- ✓ Résistance aux collisions : Il est difficile de trouver deux entrées différentes qui produisent la même sortie.
- ✓ Diffusion : Un petit changement dans l'entrée (même un seul bit) entraîne un changement radical dans la sortie. [17]

Dans la blockchain, la fonction de hachage génère une chaîne alphanumérique unique pour chaque transaction, agissant comme une "empreinte digitale". Cette valeur de hachage est essentielle pour garantir l'intégrité des données, car toute modification entraîne un hachage différent, assurant l'immuabilité. [17]

Dans la blockchain, le hachage assure l'inaltérabilité et l'authenticité des transactions. Lorsqu'une transaction est initiée, un algorithme de hachage sécurisé produit une « **signature numérique** ». Chaque bloc contient les détails de la transaction et le hachage du bloc précédent, créant une chaîne de blocs interconnectés qui renforce la sécurité et la fiabilité du système. [10]

Sur cette illustration nous présentons les blocs de la Blockchain avec les hashes [14]

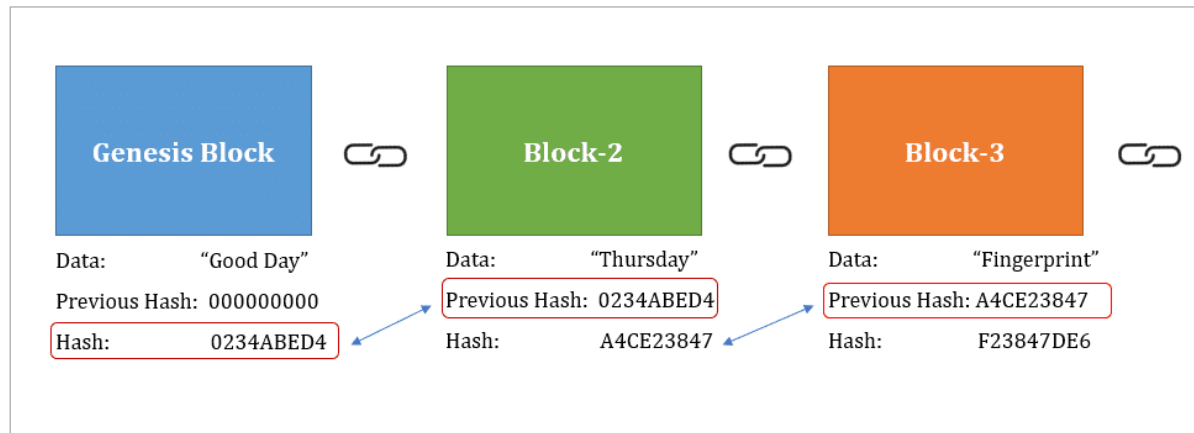


Figure 2 : Les blocs de la Blockchain avec les hashes

- ❖ **Consensus** : Un mécanisme de consensus permet d'arriver à un accord commun entre différents participants. Dans le cadre d'une blockchain, il garantit la stabilité du système malgré l'indisponibilité ou le mauvais fonctionnement d'un ou plusieurs de ses nœuds et assure que les nœuds arrivent au même état. [18]

Les mécanismes de consensus les plus connus sont :

- **Preuve de travail (Proof of Work - PoW)** : nécessite une quantité suffisante d'énergie électrique pour être dépensée. Cela limite le nombre de propositions de nouveaux blocs et favorise l'émergence d'un consensus. Aujourd'hui, il est appliqué sur le réseau Bitcoin. [18]
- **Preuve d'enjeu (Proof of Stake - PoS)** : les participants doivent investir dans la cryptomonnaie native de la blockchain et en verrouiller une quantité importante pour une certaine durée. Une variante spécifique, la preuve d'enjeu déléguée (DPoS), a été implémentée dans la version 2.0 d'Ethereum. [18]

- **Preuve d'autorité (Proof of Authority - PoA)** : un petit nombre de nœuds ont le pouvoir de participer à l'élection des nouveaux blocs. [18]

❖ **Transactions** : les transactions en chaîne font référence à une transaction effectuée sur un réseau blockchain du début à la fin. Une fois vérifiée, la transaction est enregistrée dans le grand livre public d'un réseau blockchain.

1.6.3 Fonctionnement de la blockchain

Sur cette illustration nous présentons les étapes de fonctionnement la Blockchain [31]

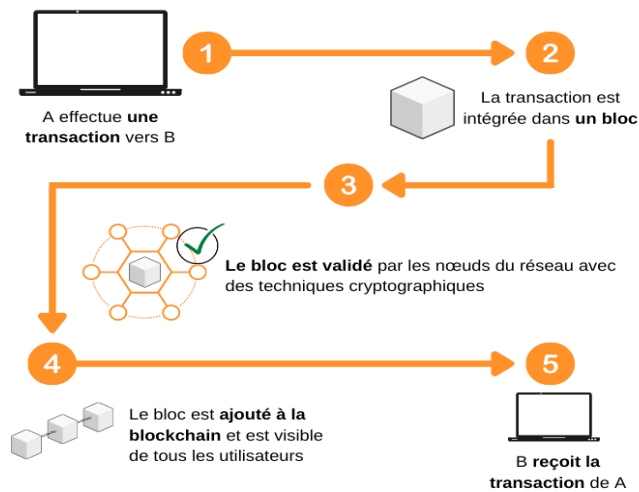


Figure 3 : Etapes de fonctionnement la Blockchain

❖ Principes Fondamentaux

La blockchain est une technologie de registre distribué qui utilise la cryptographie pour assurer des fonctions essentielles. Elle peut être programmée pour des applications spécifiques, mais son fonctionnement repose principalement sur : [19]

- La synthèse des informations stockées
- La lisibilité de l'information uniquement par des destinataires spécifiques
- L'enregistrement des informations de manière immuable et vérifiable

❖ **Processus d'une transaction Blockchain**

Lorsqu'une transaction est effectuée sur la blockchain, il se passe quelques opérations :

- Initiation de la transaction
- Hachage de la transaction : Les données de la transaction sont soumises à un algorithme de hachage, générant une signature numérique unique.
- Enregistrement dans le registre distribué : La transaction est enregistrée dans le grand registre distribué (la blockchain).

❖ **Validation et enregistrement des transactions**

Pour enregistrer et valider une transaction, la blockchain utilise des mécanismes de consensus. Un exemple concret est le Proof-of-Work (PoW) utilisé par Bitcoin. Ce que fait le Proof-of-Work

- **Examen des transactions** : Les nœuds Bitcoin examinent les nouvelles transactions introduites dans le réseau, rejettent les transactions illicites et envoient les transactions légitimes aux mineurs.
- **Sélection du mineur** : Les mineurs utilisent des appareils informatiques puissants pour résoudre un puzzle cryptographique complexe. Le premier mineur qui résout le puzzle gagne le droit d'ajouter les transactions au prochain bloc de la blockchain.
- **Ajout au bloc** : Le mineur enregistre le nouveau bloc de manière cryptographique et immuable, transmet son hachage aux nœuds, et tout le réseau vérifie l'exactitude des transactions et met à jour chaque copie du registre.

- **Preuve de Travail** : Le hachage obtenu par le mineur constitue la « preuve de travail ».

❖ **Récompense pour les mineurs**

Le mineur reçoit des frais de transaction et de nouveaux bitcoins comme récompense pour son travail, d'où le terme "minage", car il s'agit d'extraire de nouveaux bitcoins grâce à un travail intensif.

❖ **Sécurité du réseau**

Le mécanisme de consensus oblige les nœuds à se mettre d'accord sur les informations enregistrées à chaque étape. Les mineurs ont un intérêt direct au bon fonctionnement du processus, ce qui protège le réseau contre les intentions malveillantes.

1.6.4 Caractéristiques de la blockchain

❖ Décentralisation

Contrairement aux systèmes centralisés, la blockchain fonctionne sur un réseau décentralisé où chaque participant (ou nœud) possède une copie complète du registre avec une communication **Peer-to-Peer (P2P)**. [20]

Sur cette illustration nous présentons le Système centralisé et système décentralisé [32]

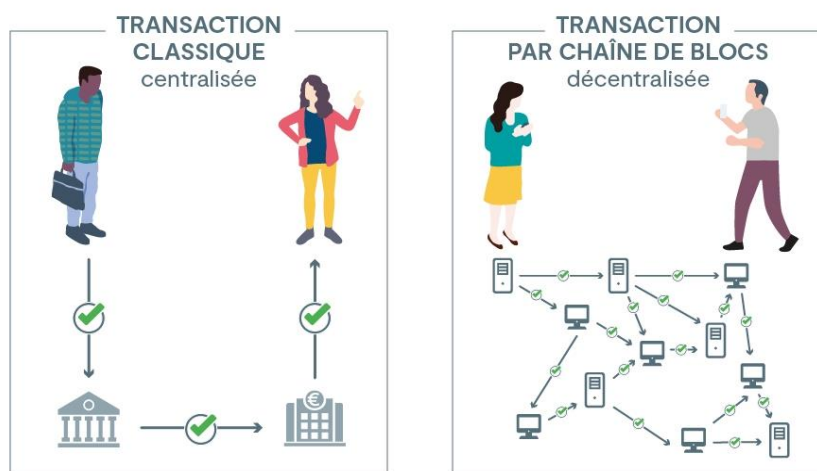


Figure 4 : Système centralisé et système décentralisé

Il n'y a pas de point de contrôle central, ce qui réduit les risques de corruption et de manipulation des données.

❖ Immutabilité

Une fois qu'une transaction est enregistrée dans un bloc et ajoutée à la chaîne, elle ne peut plus être modifiée ni supprimée. Cette caractéristique rend la blockchain particulièrement efficace pour prévenir la fraude et garantir l'intégrité des données.

❖ **Sécurité**

Les transactions sont sécurisées par des algorithmes cryptographiques avancés qui garantissent leur confidentialité et leur intégrité. Les nœuds du réseau doivent parvenir à un consensus pour valider les transactions, ce qui empêche les transactions frauduleuses.

❖ **Transparence**

Sur les blockchains publiques, toutes les transactions sont visibles par tous les participants, ce qui garantit une transparence totale. Chaque transaction peut être tracée depuis son origine, ce qui facilite les audits et les vérifications.

❖ **Programmable**

La blockchain permet la création de **contrats intelligents** auto-exécutables avec des conditions définies, automatisant ainsi de nombreuses procédures et réduisant les besoins en intermédiaires.

❖ **Anonymat et pseudonymat**

Bien que les transactions soient transparentes, les participants peuvent rester anonymes ou pseudonymes, préservant ainsi leur confidentialité.

❖ **Résistance aux pannes**

La nature décentralisée de la blockchain la rend hautement résiliente aux pannes, car le registre est répliqué sur de nombreux nœuds.

1.6.5 Types de blockchain

❖ **Blockchain publique.**

Un réseau blockchain public, ou sans autorisation, est un réseau auquel tout le monde peut participer sans restriction. La plupart des types de crypto-monnaies

fonctionnent sur une blockchain publique régie par des règles ou des algorithmes de consensus. [11]

❖ **Blockchain autorisée ou privée.**

Une blockchain privée, ou autorisée, permet aux entreprises de définir des contrôles sur qui peut accéder aux données de la blockchain. Seuls les utilisateurs disposant d'autorisations peuvent accéder à des ensembles de données spécifiques. Oracle Blockchain Platform est une blockchain autorisée. [11]

❖ **Blockchain fédérée ou de consortium.**

Un réseau de blockchain où le processus de consensus (processus de réflexion) est étroitement contrôlé par un ensemble présélectionné de nœuds ou par un nombre présélectionné de parties prenantes. [11]

1.6.6 Protocoles de blockchain

Les protocoles blockchain définissent les règles et standards qui régissent le fonctionnement d'une blockchain. Ils incluent les mécanismes de consensus, les structures de données, et les méthodes de communication entre les nœuds. Voici une présentation des quelques principaux protocoles utilisés dans les blockchains :

❖ **Bitcoin (BTC)**

Bitcoin est la première et la plus célèbre des cryptomonnaies, créée en 2009 par une personne ou un groupe de personnes utilisant le pseudonyme Satoshi Nakamoto. Il s'agit d'un système de paiement pair-à-pair décentralisé qui permet aux utilisateurs d'envoyer et de recevoir des bitcoins sans avoir besoin d'un intermédiaire comme une banque. [21] [22]

❖ **Ethereum (ETH)**

Ethereum est un protocole ouvert et décentralisé offrant à n'importe qui la possibilité d'y déployer des applications sur sa blockchain. Conçu comme une solution open source et décentralisée, Ethereum a été créé dans un but bien précis : décloisonner le Web et offrir des alternatives aux applications et plateformes aujourd'hui sous la coupelle des GAFAM et des multinationales. Bien qu'il désigne avant tout la blockchain, le terme Ethereum est aussi couramment utilisé pour désigner l'Ether (ETH), la cryptomonnaie native du réseau. [23] [24]

❖ **Binance Smart Chain (BSC)**

La Build N Build (BNB) Chain est un réseau blockchain distribué au sein duquel les développeurs et les innovateurs peuvent concevoir des applications décentralisées (DApps) pour accompagner la transition vers le Web3. [2]

❖ **Cardano (ADA)**

Lancée en 2015, Cardano se présente comme étant « le premier projet blockchain développé selon une approche scientifique. ». Alors que la somme récoltée lors de son ICO atteint les 62 millions de dollars, la première version de la blockchain Cardano a été officiellement lancée en septembre 2017, soit après deux années de recherche. [25]

❖ **Hyperledger Fabric**

Hyperledger Fabric, projet open source de la Linux Foundation, est le cadre modulaire et la norme de facto pour les plateformes de blockchain d'entreprise. Destiné à développer des applications d'entreprise et des solutions sectorielles, l'architecture modulaire ouverte utilise des composants plug-and-play pour répondre à un large éventail de cas d'utilisation. [26]

1.6.7 Applications de la Blockchain

Les applications de la blockchain couvrent de nombreux secteurs économiques. Parmi ceux-ci, on retrouve. [27]

- ❖ **Finance décentralisée (DeFi)** : DeFi est l'un des cas d'utilisation les plus prometteurs de la blockchain, permettant la création et l'échange de produits financiers décentralisés sans avoir recours aux intermédiaires traditionnels tels que les banques et les courtiers.
- ❖ **La logistique** : Particulièrement utilisée dans le secteur agroalimentaire, elle permet aux entreprises d'avoir une vision complète de leur chaîne d'approvisionnement. Du producteur à la livraison en magasin, toutes les données sont enregistrées au fur à mesure de la production. Les informations étant par définition infalsifiables, les clients bénéficient d'une transparence totale sur les produits qu'ils s'apprêtent à acheter. À la clé : une meilleure traçabilité et une relation de confiance avec le consommateur retrouvée.
- ❖ **La santé** : Regrouper toutes ses données médicales à un seul endroit ? C'est le concept mis en place par l'Estonie. Grâce à un système de blockchain, les patients choisissent d'autoriser ou non l'accès à leurs données de santé. Perte d'informations et confidentialité non respectée : le secteur médical pourrait être amené à dire adieu à des problématiques récurrentes.
- ❖ **L'assurance** : Avec cette technologie informatique, les sociétés d'assurance seraient susceptibles de gérer les indemnisations de leurs clients grâce à des processus automatisés, immédiats et anonymes. Sans aucune intervention humaine, le contrat intelligent est capable de détecter le délai éventuel d'un vol et de déclencher un paiement de compensation à l'assuré.

- ❖ **L'administratif** : Registre numérique infalsifiable, la blockchain se révèle être un outil formidable pour une meilleure gestion des données et des documents officiels. Transactions immobilières plus rapides, classification des titres de propriété, conservation de papiers originaux types diplômes ou acte d'état civil, certification originale d'une œuvre... Un moyen efficace d'éviter les erreurs, réduire les délais et donc réaliser des économies.

1.6.8 Cadre du projet

Dans le cadre de ce travail, nous allons concevoir et implémenter un système de vote en ligne sécurisé par la Blockchain. Figure 5 nous présente l'architecture qui sera utilisé dans notre système

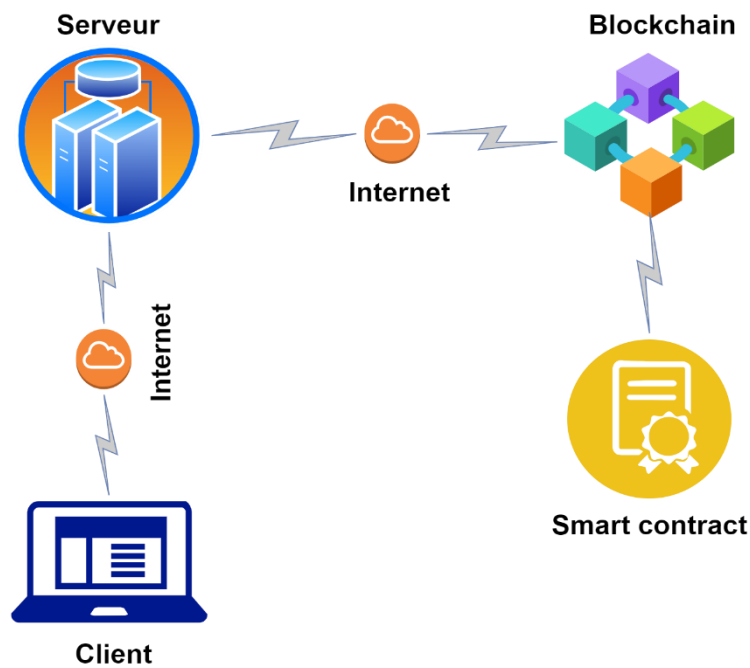


Figure 5 : Schéma présentatif du cadre de projet

1.6.9 Solution proposée

Afin de pallier les défaillances soulevées au point « Critique de l'existant », nous proposons la mise en place d'un système de vote en ligne sécurisé par la technologie Blockchain. Pour ce faire, nous optons pour le développement d'une application web et mobile dont l'objectif principal est de faciliter et sécuriser les élections en RDC.

Les caractéristiques de l'application :

- ❖ **Organisation des élections** : L'application doit permettre de planifier et gérer les élections, y compris la création et la gestion des listes de candidats.
- ❖ **Enregistrement des électeurs** : L'application doit permettre un enrôlement des électeurs plus efficace et sécurisé, réduisant les risques de doublons et de fraudes grâce à l'utilisation de la Blockchain pour enregistrer en partageant instantanément les données et en utilisant les signatures numériques d'identité
- ❖ **Vote à distance** : L'application doit permettre aux électeurs disposant d'un smartphone ou d'un ordinateur de voter depuis chez eux, garantissant ainsi une plus grande accessibilité, en particulier pour ceux vivant dans des zones éloignées ou ayant des contraintes de déplacement.
- ❖ **Sécurisation du vote** : Utiliser la technologie Blockchain pour sécuriser chaque étape du vote, de l'enregistrement de l'électeur à la soumission et au comptage des bulletins de vote, assurant ainsi une transparence totale et une immutabilité des données.
- ❖ **Publication rapide des résultats** : Les votes seront visibles en temps réel sur l'application, ce qui peut améliorer la transparence et être considéré comme des résultats préliminaires. L'application doit également permettre la publication des résultats finaux des élections dans un court délai après la clôture du scrutin, réduisant ainsi les tensions et les suspicions liées aux retards et garantissant la crédibilité du processus électoral.

Cette solution vise à moderniser le processus électoral en RDC, à accroître la participation des électeurs, à renforcer la transparence et à garantir des élections libres et équitables.

1.7 Conclusion partielle

Dans ce chapitre nous avons présenté les différents concepts autour du thème des élections et de la technologie Blockchain, ainsi que les potentielles applications de la Blockchain. Nous avons également souligné les insuffisances du processus actuel utilisé par la RDC pour organiser les élections, accompagnées d'une proposition de solution.

Dans le chapitre suivant, nous allons aborder l'étude conceptuelle de la solution retenue.

Chapitre 2. Conception de l'architecture logicielle

2.1 Introduction

L'étude du système existant a révélé ses points forts ainsi que ses faiblesses. Ce chapitre se concentre sur deux aspects principaux. D'abord, nous présenterons les spécifications des besoins pour identifier les contraintes à prendre en compte lors de la conception de la solution.

Dans le cadre de notre projet, nous adopterons une approche basée sur le Processus Unifié (UP) et l'utilisation d'UML, en structurant notre démarche en plusieurs phases. Tout d'abord, des diagrammes de cas d'utilisation seront créés pour définir les interactions entre les acteurs (utilisateurs ou systèmes) et le système, afin de décrire les fonctionnalités principales du système du point de vue des utilisateurs. Ensuite, des diagrammes de classes seront utilisés pour modéliser la structure des données, en représentant les objets du système ainsi que les relations qui existent entre eux.

2.2 Intégration de la Technologie Blockchain

Dans notre projet, l'intégration de la blockchain est cruciale pour résoudre les défis liés à la transparence et la sécurité des élections. Notre objectif est de créer un système dans lequel chaque électeur doit avoir confiance. Dans cette section, nous allons montrer comment la blockchain est intégré dans notre système afin de répondre aux objectifs précédemment définis.

La liaison de notre système avec la blockchain peut être décrit à plusieurs étapes clés :

2.2.1 Déploiement du scrutin

Le déploiement du scrutin est la première étape où le processus de vote est lancé et mis à la disposition des électeurs. A cette étape notre système se connecte à la blockchain et enregistre toutes les informations (Informations générales du scrutin, les électeurs, les candidats) et les configurations (la période de vote, type de vote, ...) liées au scrutin afin de s'assurer qu'aucune

information ou paramètre ne peuvent être modifiés une fois le scrutin lancé. Après cette étape aucun nouvel électeur ou candidat ne peut être enregistré.

2.2.2 Processus de Vote

Lorsque le scrutin est déployé, les électeurs peuvent participer au processus de vote en ligne via une interface utilisateur. Quand l'électeur demande à voter, le système lui demande son jeton de vote unique pour authentifier son vote. Une fois que l'électeur soumet son vote le système utilise la blockchain pour enregistrer ce dernier sous forme de transaction unique. Chaque vote est alors validé par les nœuds du réseau via un mécanisme de consensus avant d'être ajouté à la chaîne. Cette étape garantit que chaque vote est enregistré de manière immuable et ne peut pas être modifié ou supprimé

2.2.3 Vérification des Votes

Une fois qu'un électeur a voté, un hash de transaction est généré et affiché à l'électeur. Cet hash permet à l'électeur de vérifier, que son vote a bien été prise en compte, sans révéler son contenu exact, assurant ainsi l'anonymat.

2.2.4 Fermeture du scrutin

À la fin de la période de vote, le processus de fermeture du scrutin est automatiquement déclenché. À ce stade, aucun nouveau vote ne peut être soumis, et le système de blockchain clôture l'enregistrement des transactions liées au vote. Tous les votes enregistrés jusqu'à ce moment restent accessibles pour vérification. Après la clôture du scrutin seules les vues seront autorisées.

2.3 Présentation des acteurs

Avant d'étudier les cas d'utilisation, nous présentons les acteurs intervenant dans notre champ d'étude. Ces acteurs se répartissent en deux catégories : les acteurs primaires, qui interagissent directement avec le système, et les acteurs secondaires, qui jouent un rôle plus passif ou indirect.

- ❖ **Acteurs primaires** : Ces acteurs participent activement au processus électoral et interagissent directement avec le système.
 - ✓ **Administrateur** : Responsable de la gestion des votes, il crée, supervise et administre le processus électoral. Il enregistre les électeurs et les candidats, et veille au bon déroulement des opérations.
 - ✓ **Électeur** : Utilisateur habilité à voter, il génère et valide son bulletin de vote à l'aide d'un jeton, puis consulte les résultats.

- ❖ **Acteurs secondaires** : Ces acteurs n'interviennent pas directement dans le processus électoral, mais ils jouent un rôle dans l'écosystème du système.
 - ✓ **Internaute** : Utilisateur qui accède aux informations générales sur les votes et aux résultats, sans participer activement au vote.
 - ✓ **Blockchain** : Infrastructure technologique assurant la décentralisation et la sécurisation du système en distribuant les données sur un réseau d'ordinateurs.

2.4 Spécifications des besoins

Dans cette partie, nous allons nous concentrer sur les besoins des utilisateurs en abordant les **spécifications fonctionnelles** et **non fonctionnelles**, afin de concevoir une application répondant aux attentes définies.

2.4.1 Spécifications fonctionnelles

Nous devons mettre en place un système qui va permettre à l'utilisateur de faire ce qui suit :

2.4.1.1 Gestion du compte :

Cette section décrit les fonctionnalités liées à la gestion des comptes des utilisateurs :

- ❖ Création d'un compte

- ❖ Connexion à la plateforme.
- ❖ Récupération du compte en cas de mot de passe oublié.
- ❖ Ajouter un administrateur.
- ❖ Mettre à jour les informations d'un administrateur.
- ❖ Modifier son profil.
- ❖ Modifier son mot de passe.

2.4.1.2 Gestion des scrutins :

Cette section couvre les fonctionnalités liées à l'organisation et à la gestion des scrutins :

- ❖ Créer un nouveau scrutin.
- ❖ Afficher les informations du scrutin
- ❖ Ajouter des types de candidatures pour le scrutin.
- ❖ Enregistrer les candidats pour un scrutin.
- ❖ Enregistrer les électeurs pour un scrutin.
- ❖ Déployer le scrutin.
- ❖ Publier les résultats du scrutin.

2.4.1.3 Gestion des jetons et accès au scrutin :

Cette section présente les fonctionnalités relatives à la distribution des jetons et à l'accès aux scrutins :

- ❖ Envoyer des jetons aux électeurs : Cette fonctionnalité permet au système d'envoyer des jetons uniques à chaque électeur enregistré pour un scrutin spécifique. Chaque jeton

contient un code unique, généré automatiquement par le système, qui permet à l'électeur d'accéder au scrutin et de voter. Cette génération automatique des jetons pour chaque électeur garantit ainsi la sécurité et l'intégrité du vote.

- ❖ Envoyer le code du scrutin aux candidats pour leur permettre de consulter les résultats.

2.4.1.4 Gestion des votes :

Cette section décrit les actions que les électeurs peuvent entreprendre lors du vote :

- ❖ Accéder au scrutin en entrant son code.
- ❖ Créer un bulletin.
- ❖ Ajouter des choix dans le bulletin.
- ❖ Valider le bulletin ou ses choix.

2.4.1.5 Gestion des résultats :

Enfin, cette section traite des fonctionnalités liées à la publication et à la consultation des résultats :

- ❖ Publier les résultats du scrutin
- ❖ Consulter les résultats via le code du scrutin

2.4.2 Spécifications non fonctionnelles

Les besoins non fonctionnels jouent un rôle crucial, car ils influencent indirectement le résultat final et l'expérience utilisateur. Il est donc impératif que notre application les prenne en compte pour garantir un système électoral fiable, sécurisé et performant.

- ❖ **Fiabilité** : L'application doit assurer un enregistrement correct et inaltérable de chaque vote sur la blockchain. Pour cela, nous utilisons un mécanisme de validation garantissant qu'aucun vote ne soit perdu ou modifié après son enregistrement.

- ❖ **Gestion des erreurs** : Un système de gestion des erreurs est mis en place pour détecter et signaler tout problème. Des messages clairs et explicites guideront les utilisateurs en cas d'échec d'une opération, facilitant ainsi une résolution rapide.
- ❖ **Ergonomie** : L'interface est conçue pour être intuitive et accessible, même aux utilisateurs non technophiles. Des instructions simples, des icônes compréhensibles et un processus de vote guidé garantissent une expérience fluide.
- ❖ **Efficacité** : Le processus de vote et la consultation des résultats doivent être optimisés pour minimiser le nombre d'étapes. Une navigation simplifiée et des actions rapides permettent aux électeurs d'accomplir leur tâche en quelques clics.
- ❖ **Sécurité** : La protection des données et l'intégrité du vote sont assurées par des mécanismes de chiffrement et d'authentification sécurisée via un jeton unique. La blockchain garantit l'immutabilité des votes, empêchant toute manipulation ou fraude électorale.
- ❖ **Performance** : L'architecture de l'application est conçue pour supporter un grand nombre d'électeurs votant simultanément. L'optimisation des requêtes et l'usage de la blockchain permettent un traitement rapide des votes et l'affichage instantané des résultats.

2.5 Types de vote pris en charge par le système

Dans notre système nous allons prendre en compte différents types de vote dont :

- ❖ **Le vote binaire** : Les électeurs choisissent entre deux options opposées, telles que « **Pour** » ou « **Contre** » une proposition ou une décision.
- ❖ **Le vote uninominal** : Chaque électeur sélectionne un seul candidat parmi ceux en compétition, ce qui est courant dans les élections pour des postes uniques.

- ❖ **Le vote plurinominal** : Les électeurs votent pour plusieurs candidats dans un même scrutin, en respectant le nombre de choix autorisé. Il est utilisé dans les élections impliquant plusieurs postes à pourvoir.
- ❖ **Le vote plurinominal suivant les règles de la CENI** : Ce type de vote est comme le précédent mais celui-ci suit les directives spécifiques établies par la CENI. [28]

2.6 Présentation des cas d'utilisation

Cette section vise principalement à identifier les principaux acteurs et les cas d'utilisation. Nous allons créer les diagrammes de cas d'utilisation pour décrire en détail les besoins du client final et préciser le comportement attendu du système à développer.

2.6.1 Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation offre une vue d'ensemble du comportement fonctionnel du système en montrant les interactions entre les acteurs et le système de manière abstraite. Dans ce diagramme, les acteurs interagissent avec les cas d'utilisation (use cases) pour réaliser des objectifs spécifiques.

❖ Diagramme de cas d'utilisation d'un administrateur

Figure 6 représente les besoins fonctionnels d'un administrateur



Figure 6 : Diagramme de cas d'utilisation d'un administrateur

❖ Diagramme de cas d'utilisation d'un électeur et un internaute

Figure 7 représente les besoins fonctionnels d'un électeur et un internaute

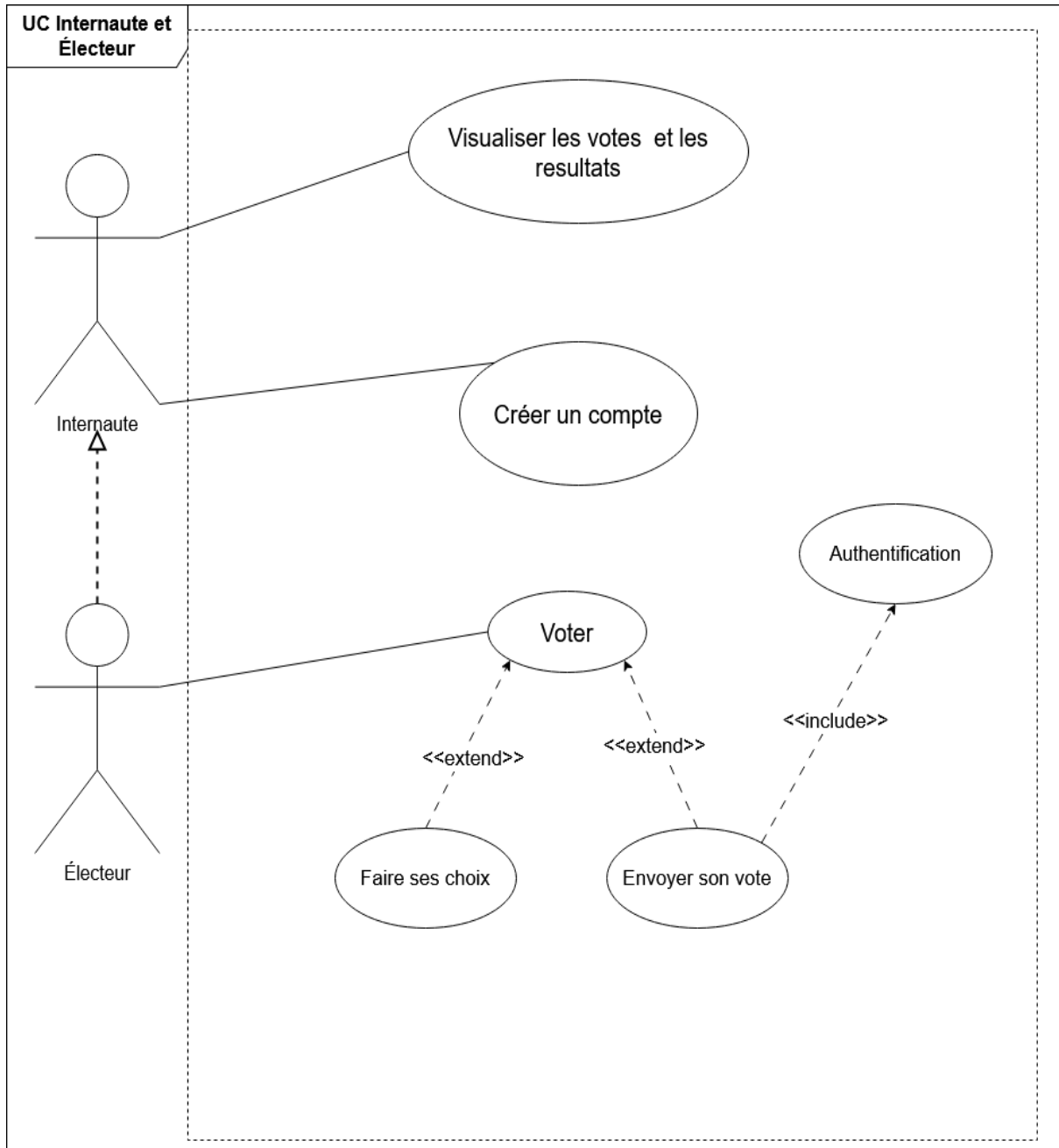


Figure 7 : Diagramme de cas d'utilisation d'un électeur et un internaute

2.6.2 Documentation de différents cas d'utilisation

2.6.2.1 Documentation du cas d'utilisation « S'authentifier »

L'organisateur de vote doit s'authentifier pour pouvoir accéder à la plateforme

Tableau 1 : Documentation du cas d'utilisation « S'authentifier »

UC : S'authentifier
ID : 1
Acteur : Administrateur, Electeur
Précondition : l'acteur doit avoir un compte sur la plateforme
Enchaînement principal : <ol style="list-style-type: none">1. L'acteur demande l'accès au système2. Le système affiche le formulaire d'authentification3. L'acteur complète et valide le formulaire4. Le système vérifie les données saisies par l'acteur<ul style="list-style-type: none">• Si les données saisies sont valides<ul style="list-style-type: none">➤ Le système affiche l'interface d'accueil• Si les données saisies sont invalides<ul style="list-style-type: none">➤ Le système affiche un message d'erreur.
Post condition : L'utilisateur est connecté

2.6.2.2 Documentation du cas d'utilisation « Créer un compte »

L'internaute doit pouvoir créer un compte d'organisation pour qu'il ait l'accès à la plateforme et organiser des scrutins

Tableau 2 : Documentation du cas d'utilisation « Créer un compte »

UC : Créer un compte
ID : 2
Acteur : Internaute
Précondition :
Enchaînement principal : <ol style="list-style-type: none">1. Le cas d'utilisation démarre lorsque l'acteur clic sur le bouton créer un compte2. Le système affiche un formulaire de création de compte3. L'acteur complète et valide le formulaire4. Le système vérifie les données saisies par l'acteur<ul style="list-style-type: none">• Si les données saisies sont valides<ul style="list-style-type: none">➤ Le système ajoute un compte et envoie les paramètres de connexion à l'adresse mail de l'acteur.• Si les données saisies sont invalides<ul style="list-style-type: none">➤ Le système affiche un message d'erreur
Post condition : Augmentation du nombre d'utilisateurs dans le système

2.6.2.3 Documentation du cas d'utilisation « Créer un scrutin »

L'organisateur de scrutin doit pouvoir créer un scrutin dans le système pour organiser le vote

Tableau 3 : Documentation du cas d'utilisation « Créer un scrutin »

UC : Créer un scrutin
ID : 3
Acteur : Administrateur
Précondition : l'acteur est connecté
Enchaînement principal : <ol style="list-style-type: none">1. Le cas d'utilisation démarre lorsque l'acteur clic sur le bouton créer un scrutin2. Le système affiche un formulaire de création de scrutin3. L'acteur complète et valide le formulaire4. Le système vérifie les données saisies par l'acteur<ul style="list-style-type: none">• Si les données saisies sont valides<ul style="list-style-type: none">➤ Le système ajoute un scrutin et renvoie l'utilisateur sur page de gestion du scrutin• Si les données saisies sont invalides<ul style="list-style-type: none">➤ Le système affiche un message d'erreur
Post condition : Augmentation du nombre des scrutins dans le système

2.6.2.4 Documentation du cas d'utilisation « Ajouter un type de candidature »

L'organisateur de scrutin doit pouvoir ajouter un type de candidature pour un scrutin, mais ce cas d'utilisation est seulement disponible pour tous les types de vote citer au 2.3 sauf le type de vote binaire.

Tableau 4 : Documentation du cas d'utilisation « Ajouter un type de candidature »

UC : Ajouter un type de candidature
ID : 4
Acteur : Administrateur
Précondition : l'acteur est connecté
Enchaînement principal : <ol style="list-style-type: none">1. Le cas d'utilisation commence quand l'acteur clique sur bouton ajouter un type de candidature du menu types candidature de la gestion du scrutin2. Le système affiche un formulaire que l'acteur doit compléter pour ajouter un type de candidature.3. L'acteur complète le formulaire avec les données demande4. Le système vérifie les données que l'acteur à saisies<ul style="list-style-type: none">• Si les données sont valides<ul style="list-style-type: none">• Le système ajoute un type de candidature et renvoie l'utilisateur sur la page de types de candidature.• Si les données sont invalides

➤ Le système affiche un message d'erreur a l'acteur
Post condition : Augmentation du nombre de type de candidature pour le scrutin dans le système

2.6.2.5 Documentation du cas d'utilisation « Ajouter un candidat »

L'organisateur de scrutin doit pouvoir ajouter un candidat pour un scrutin dans le système, mais ce cas d'utilisation est seulement disponible pour tous les types de vote citer au point 2.3 sauf le type de vote binaire.

Tableau 5 : Documentation du cas d'utilisation « Ajouter un candidat »

UC : Ajouter un candidat
ID : 5
Acteur : Administrateur
Précondition : l'acteur est connecté
<p>Enchaînement principal :</p> <ol style="list-style-type: none"> 1. Le cas d'utilisation commence quand l'acteur clique sur bouton ajouter un candidat du menu candidat de la gestion du scrutin 2. Le système affiche un formulaire que l'acteur doit compléter pour ajouter un candidat. 3. L'acteur complète le formulaire avec les données demande 4. Le système vérifie les données que l'acteur à saisies

<ul style="list-style-type: none"> • Si les données sont valides <ul style="list-style-type: none"> • Le système ajoute un candidat dans le système et renvoie l'acteur sur la page des candidats • Si les données sont invalides <ul style="list-style-type: none"> ➤ Le système affiche un message d'erreur a l'acteur
Post condition : Augmentation du nombre de candidats pour le scrutin dans le système

2.6.2.6 Documentation du cas d'utilisation « Ajouter un électeur »

L'organisateur de scrutin doit pouvoir ajouter un électeur qui est autorisé à participer et le système va lui envoyer un jeton de vote unique.

Tableau 6 : Documentation du cas d'utilisation « Ajouter un électeur »

UC : Ajouter un électeur
ID : 6
Acteur : Administrateur
Précondition : l'acteur est connecté
<p>Enchaînement principal :</p> <ol style="list-style-type: none"> 1. Le cas d'utilisation commence quand l'acteur clique sur bouton ajouter un électeur du menu électeur de la gestion du scrutin 2. Le système affiche un formulaire que l'acteur doit compléter pour ajouter un électeur. 3. L'acteur complète le formulaire avec les données demande

<p>4. Le système vérifie les données que l'acteur à saisies</p> <ul style="list-style-type: none"> • Si les données sont valides <ul style="list-style-type: none"> • Le système ajoute un candidat dans le système et renvoie l'acteur sur la page des électeurs • Si les données sont invalides <ul style="list-style-type: none"> ➤ Le système affiche un message d'erreur a l'acteur
<p>Post condition : Augmentation du nombre des électeurs pour le scrutin dans le système</p>

2.6.2.7 Documentation du cas d'utilisation « Ajouter une proposition ou décision »

L'organisateur de scrutin doit pouvoir ajouter une proposition ou décision à voter dans un scrutin du type binaire.

Tableau 7 : Documentation du cas d'utilisation « Ajouter une proposition ou décision »

UC : Ajouter une proposition ou décision
ID : 7
Acteur : Administrateur
Précondition : l'acteur est connecté
<p>Enchaînement principal :</p> <ol style="list-style-type: none"> 1. Le cas d'utilisation commence quand l'acteur clique sur bouton ajouter une proposition du menu Proposition de la gestion du scrutin

<ol style="list-style-type: none"> 2. Le système affiche un formulaire que l'acteur doit compléter pour ajouter une proposition qui doit être voté. 3. L'acteur complète le formulaire avec les données demande 4. Le système vérifie les données que l'acteur à saisies <ul style="list-style-type: none"> • Si les données sont valides <ul style="list-style-type: none"> • Le système ajoute un candidat dans le système et renvoie l'acteur sur la page de proposition • Si les données sont invalides <ul style="list-style-type: none"> ➤ Le système affiche un message d'erreur a l'acteur
<p>Post condition : Augmentation du nombre des propositions dans le système</p>

2.6.2.8 Documentation du cas d'utilisation « Déployer le scrutin »

L'organisation de scrutin, après avoir créé le scrutin doit pour déployer le scrutin sur la blockchain et le rendre disponible pour que les électeurs puissent votés et les internautes suivent le déroulement.

Tableau 8 : Documentation du cas d'utilisation « Déployer le scrutin »

UC : Déployer le scrutin
ID : 8
Acteur : Administrateur
Précondition : l'acteur est connecté

Enchaînement principal :

1. Le cas d'utilisation commence quand l'acteur clique sur le bouton **déployer** dans la gestion du scrutin
2. Le système procède au déploiement du scrutin en effectuant ces tâches :
 - Enregistrer les candidats et les électeurs participant au scrutin dans la blockchain afin de garantir l'intégrité des données et d'éviter toute modification non autorisée.
 - Envoyer un mail aux candidats contenant un message qui confirme que leur candidature a été prise en compte et le code du scrutin pour suivre le déroulement de vote
 - Envoyer un mail aux électeurs contenant un message disant qu'ils sont invités à participer au scrutin, le code du scrutin pour y accéder et un jeton de vote à usage unique
 - Calculer le hash (empreinte numérique) du scrutin en utilisant les informations générales du scrutin, ainsi que les électeurs et les candidats enregistrés pendant son déploiement, afin de garantir qu'aucune modification ne puisse être effectuée, notamment l'ajout d'un candidat, l'ajout d'un électeur ou le changement des informations du scrutin.
3. Le système affiche un message à l'acteur
 - Si le processus de déploiement très bien passé
 - Le système affiche un message à l'acteur lui informant que le déploiement
 - Si toutes les tâches du déploiement n'ont pas été bien effectués

<ul style="list-style-type: none"> ➤ Le système affiche un message d'erreur lui disant ce qui s'est mal passé
Post condition : Le scrutin est déployé

2.6.2.9 Documentation du cas d'utilisation « Chercher un scrutin »

L'internaute doit pouvoir chercher un scrutin sur la plateforme en utilisant son code

Tableau 9 : Documentation du cas d'utilisation « Chercher un scrutin »

UC : Chercher un scrutin
ID : 9
Acteur : Internaute
Précondition : L'acteur possède un code de scrutin
<p>Enchaînement principal :</p> <ol style="list-style-type: none"> 1. Le cas d'utilisation commence quand l'acteur clique sur bouton chercher un scrutin 2. Le système affiche un formulaire demande à l'acteur le code du scrutin qu'il cherche 3. Le système vérifie si le scrutin existe dans le système <ul style="list-style-type: none"> • Si le scrutin existe <ul style="list-style-type: none"> ➤ Le système affiche à l'acteur la page concernant les informations générales du scrutin avec la possibilité de voir les candidats ou la proposition qu'on peut choisir, les votes, voter, les résultats finals • Si le scrutin n'existe pas

<ul style="list-style-type: none"> Le système affiche à l'acteur un message disant que « le scrutin avec code n'existe pas »
Post condition : L'acteur accède aux informations du scrutin

2.6.2.10 Documentation du cas d'utilisation « Voter »

L'électeur doit pouvoir avoir la possibilité de voter

Tableau 10 : Documentation du cas d'utilisation « Voter »

UC : Voter
ID : 10
Acteur : Electeur
Précondition : l'acteur possède le code du scrutin et le jeton de vote
<p>Enchaînement principal :</p> <ol style="list-style-type: none"> Le cas d'utilisation commence quand l'acteur clique sur le bouton voter Le système affiche les candidats ou la proposition pour le scrutin L'acteur fait son choix et demande à valider son vote Le système demande à l'acteur son jeton de vote L'acteur saisie son jeton de vote Le système vérifie le vote du l'acteur (choix et jeton) <ul style="list-style-type: none"> Si les données sont valides :

<ul style="list-style-type: none"> • Le système enregistre le vote de l'acteur et lui renvoie un message de succès • Si les données sont invalides <ul style="list-style-type: none"> • Le système affiche un message d'erreur a l'acteur.
Post condition : Augmentation du nombre de vote pour le scrutin

2.6.2.11 Documentation du cas d'utilisation « Publier les résultats du scrutin »

L'organisateur de scrutin doit pouvoir publier les résultats du scrutin

Tableau 11 : Documentation du cas d'utilisation « Publier les résultats du scrutin »

UC : Publier les résultats du scrutin
ID : 11
Acteur : Administrateur
Précondition : l'acteur possède le code du scrutin et le jeton de vote
<p>Enchaînement principal :</p> <ol style="list-style-type: none"> 1. Le cas d'utilisation commence quand l'acteur clique sur le bouton publier 2. Le système vérifie si tous les votes ont été enregistré la Blockchain <ul style="list-style-type: none"> • S'il y a des votes qui n'ont pas été enregistré la Blockchain <ul style="list-style-type: none"> ➤ Le système procède à l'enregistrement de tous les votes

<ol style="list-style-type: none"> 3. Le système calcul le hash du scrutin pour éviter toutes modification après publication du scrutin 4. Le système ferme le scrutin. 5. Le système affiche un message à l'acteur disant que les résultats ont été publiés
<p>Post condition : Les résultats du scrutin sont publiés et le scrutin est fermé</p>

2.6.2.12 Documentation du cas d'utilisation « Ajouter un utilisateur »

L'organisateur de scrutin doit pouvoir ajouter un utilisateur dans son compte pour l'aider gérer les scrutins.

Tableau 12 : Documentation du cas d'utilisation « Ajouter un utilisateur »

UC : Ajouter un utilisateur
ID : 12
Acteur : Administrateur
Précondition : l'acteur est connecté
<p>Enchaînement principal :</p> <ol style="list-style-type: none"> 1. Le cas d'utilisation commence quand l'acteur clique sur le bouton nouvel utilisateur 2. Le système affiche un formulaire que l'acteur doit compléter pour ajouter un utilisateur 3. L'acteur complète le formulaire avec les données demande 4. Le système vérifie les données que l'acteur à saisies

- Si les données saisies sont valides
 - Le système ajoute l'utilisateur dans le système
 - Le système envoie un mail à l'utilisateur ajouté avec ses paramètres de connexion
- Si les données sont invalides
 - Les systèmes affiche un message d'erreur a l'acteur

Post condition : Augmentation du nombre d'utilisateurs dans le système

2.7 Diagramme de classe

Dans cette partie, il s'agit de mettre en évidence les données nécessaires afin de pouvoir concevoir la structure de la base de données. C'est dans ce sens que nous allons formaliser la structure et la signification des informations perçues dans l'intérêt de notre travail. Nous allons donc réaliser un diagramme de classe qui comporte les concepts suivants :

- ❖ **Classe** : Représente un objet ou une entité avec des caractéristiques.
- ❖ **Association** : Modélisation d'une relation entre deux entités
- ❖ **Cardinalités** : modélisation des participations minimales et maximales d'une entité a une relation.
- ❖ **Propriétés** : Ce sont les détails qui décrivent une entité ou une relation.
- ❖ **Identifiant** : Un attribut unique qui distingue chaque entité, comme un numéro d'étudiant.

Tout d'abord présentons les données qui figureront dans le modèle conceptuel et cela dans un dictionnaire de données.

Tableau 13 : Dictionnaire de donnée

N°	Classe	Codification	Libelle
1	EntiteOrganisateur	ID	Identifiant unique de l'entité organisateur de vote
		nom	Nom de l'entité organisateur de vote
		logo	Logo de l'entité organisateur de vote
		description	Description de l'entité organisateur de vote
2	Administrateur	ID	Identifiant unique de l'entité administrateur
		matricule	Matricule de l'administrateur
		nomComple	Nom complet de l'administrateur
		sexe	Sexe de l'administrateur
		telephone	Numéro de téléphone de l'administrateur
		photoProfil	Photo de profil de l'administrateur

		isValid	Un drapeau qui indique que l'administrateur est validé
3	Utilisateur	ID	Identifiant unique de l'utilisateur
		username	Nom d'utilisateur
		email	Adresse mail de l'utilisateur
		password	Hash du mot de passe de l'utilisateur
		lastLogin	Date de dernière connexion de l'utilisateur
		isActif	Un drapeau qui indique que l'utilisateur est actif et peut accéder au système
4	RolesAdmin	ID	Identifiant unique du rôle
		role	La désignation du rôle utilisateur
		isActif	Un drapeau qui indique que le rôle est actif
5	Scrutin	ID	Identifiant unique du scrutin
		code	Code du scrutin
		intitule	Intitulé du scrutin
		categorie	Catégorie du scrutin

		dateDebut	Date de début du scrutin
		dateFin	Date de fin du scrutin
		statut	Statut qui indique l'évolution du scrutin
		resultatPublier	Drapeau qui indique que les résultats sont publiés
		deployer	Drapeau qui indique que le scrutin est déjà déployé sur la blockchain et qu'on peut commencer le vote
		hashScrutin	Empreinte numérique du scrutin
6	Electeur	ID	Identifiant unique de l'électeur
		nomComplet	Nom complet de l'électeur
		email	Adresse mail de l'électeur
		telephone	Numéro de téléphone de l'électeur
		deployer	Un drapeau qui indique que l'électeur est déployé sur le scrutin et peut voter
7	TypeCandidature	ID	Identifiant unique du type de candidature
		candidature	Intitule de la candidature
		nombreSiege	Nombre de siège vaquant

8	Candidat	ID	Identifiant unique du candidat
		numero	Numéro du candidat dans le scrutin
		nomCompleat	Nom complet du candidat
		email	Adresse mail du candidat
		telephone	Numéro de téléphone du candidat
		photoProfil	Photo de profil du candidat
		deployer	Drapeau qui indique que le candidat est déployé sur le scrutin et peut être voter
9	TokenElecteur	ID	Identifiant unique du jeton de l'électeur
		tokenHash	Hash du jeton unique de vote de l'électeur
		isUse	Drapeau qui indique que le jeton est déjà utilisé
10	BulletinVote	ID	Identifiant unique du bulletin de vote de l'électeur
		hashVote	Empreinte numéro des choix de l'électeur
		deployer	Drapeau qui indique que le bulletin est enregistré sur la Blockchain

11	ChoixElecteur	ID	Identifiant unique du choix de l'électeur
		candidat	Candidat choisi par l'électeur dans son bulletin de vote
12	PropositionDecision	ID	Identifiant unique de la proposition
		proposition	Proposition que l'électeur devrait voter
		deployer	Drapeau qui indique que la proposition est enregistrée sur la blockchain
		hash	Empreinte numérique des votes de la proposition ou la décision
13	VoteProposition	ID	Identifiant unique du vote de la proposition
		choix	Choix de l'électeur sur la proposition
14	Province	ID	Identifiant unique de la province
		nom	Nom de la province
		nombreCirconscription	Nombre des circonscriptions de la province
15	Circonscription	ID	Identifiant unique de la circonscription

		code	Code de la circonscription
		nom	Nom de la circonscription

Maintenant nous pouvons présenter notre « Diagramme de classe » relatif au vote en RDC, qui est donné à la Figure 8.

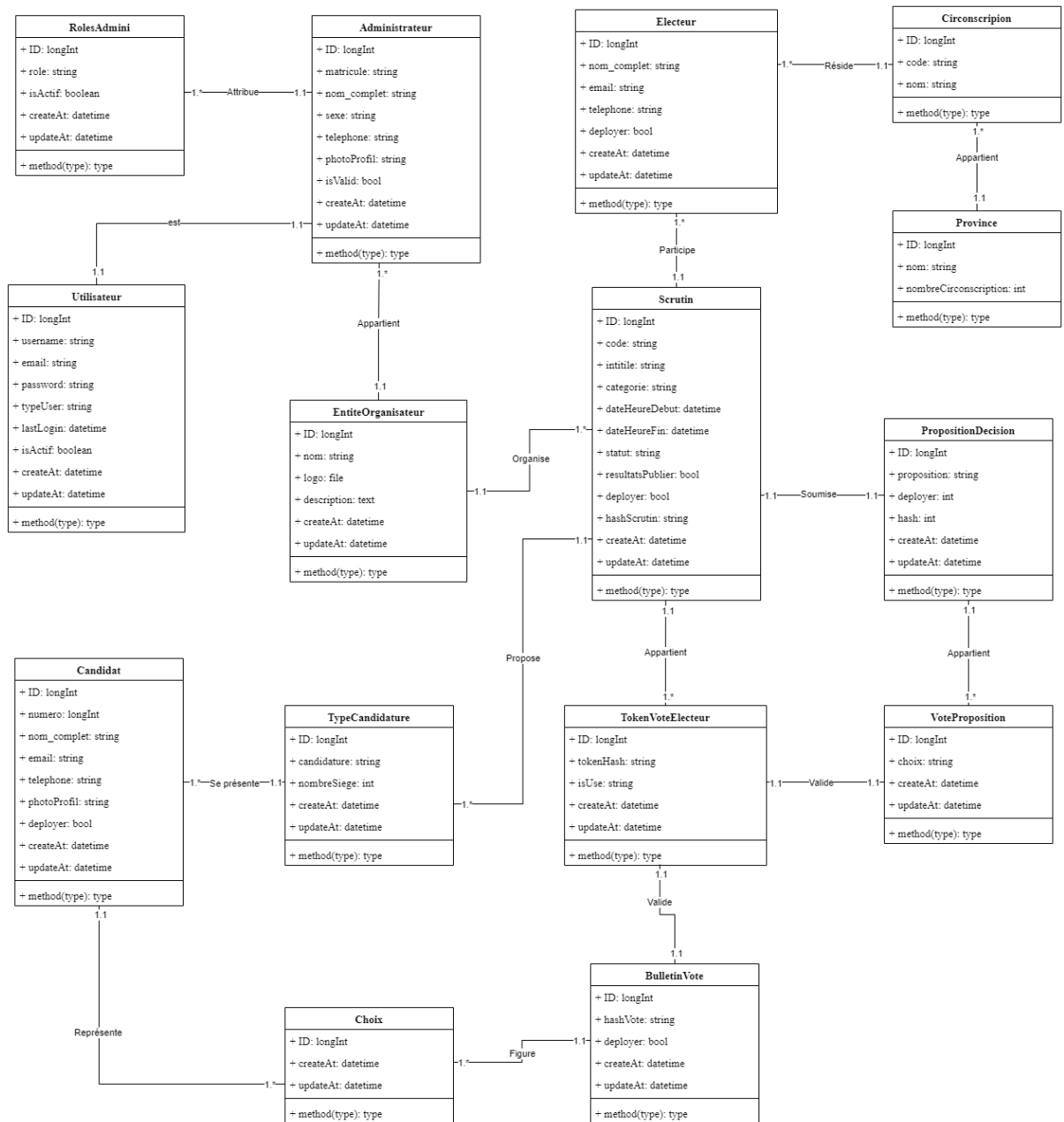


Figure 8 : Diagramme de classe du système

2.8 Architecture globale

L'architecture suivante illustre comment notre système interagira avec les utilisateurs du point de vue logiciel. Le client, à travers notre application web, envoie une demande (se connecter, voter, etc.) au serveur via Internet. Le serveur vérifie alors la disponibilité de la ressource demandée. Si la ressource est présente, il renvoie directement la réponse au client. Dans le cas contraire, si la ressource se trouve sur la blockchain, il l'interroge ; cette dernière traite la demande et renvoie les informations au serveur, qui transmet ensuite ces données au client.

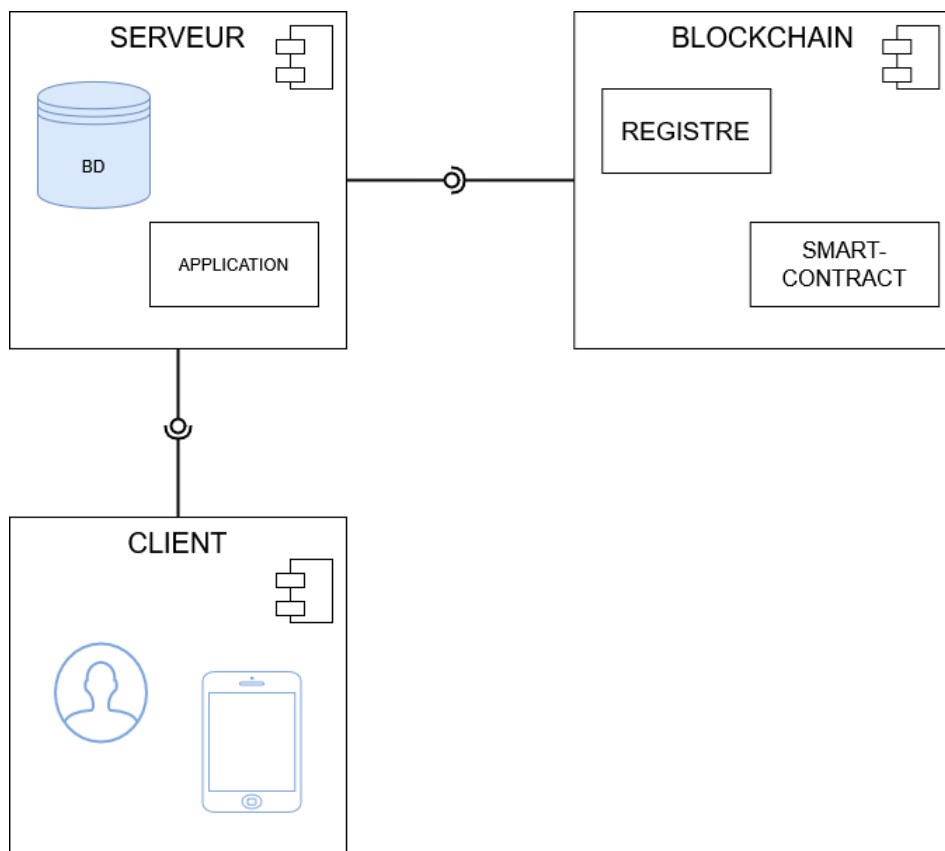


Figure 9 : Architecture globale du système

2.9 Conclusion partielle

Dans cette section, nous avons mis l'accent sur la présentation de la solution conceptuelle adoptée. Nous avons présenté les diagrammes « **de cas d'utilisation et de classes** » qui nous ont permis de représenter les besoins du système. De plus nous avons montré comment la blockchain est intégrée dans notre projet. Les résultats de cette analyse seront utilisés dans la phase d'implémentation en s'appuyant sur les technologies et outils qui seront détaillés dans le chapitre suivant.

Chapitre 3. Implémentation du système et intégration de la technologie Blockchain

3.1 Introduction

Après l'analyse et la conception de notre système, nous entamons une phase décisive qui concrétise les choix réalisés lors des étapes précédentes. Ce chapitre est consacré à la présentation des résultats obtenus à travers le développement de la solution retenue ainsi que les environnements matériels, logiciels et outils utilisés. Nous allons également présenter les tests réalisés sur le système, notamment des tests de performance sous charge, de résistance aux attaques DDoS, ainsi que des tests de brute forcing. Ces tests visent à démontrer la fiabilité, la sécurité et la robustesse de notre solution.

3.2 Environnement de travail

3.2.1 Environnement matériel

L'équipement utilisé pour la réalisation de ce projet est un ordinateur portable ASUS VivoBook 15, fonctionnant sous le système d'exploitation Linux avec la distribution Ubuntu 22.04. Ce modèle est doté d'un processeur 11th Gen Intel(R) Core(TM) i5-1135G7, qui fonctionne à une fréquence de base de 2,40 GHz et peut atteindre 2,42 GHz lors de pics de performance. Il est également équipé de 20,0 Go de mémoire RAM, dont 19,7 Go sont utilisables, et d'un espace disque de 500 Go, offrant une capacité de stockage suffisante pour les besoins du projet.

3.2.2 Environnements logiciels

3.2.2.1 Visual Studio Code (VS Code)

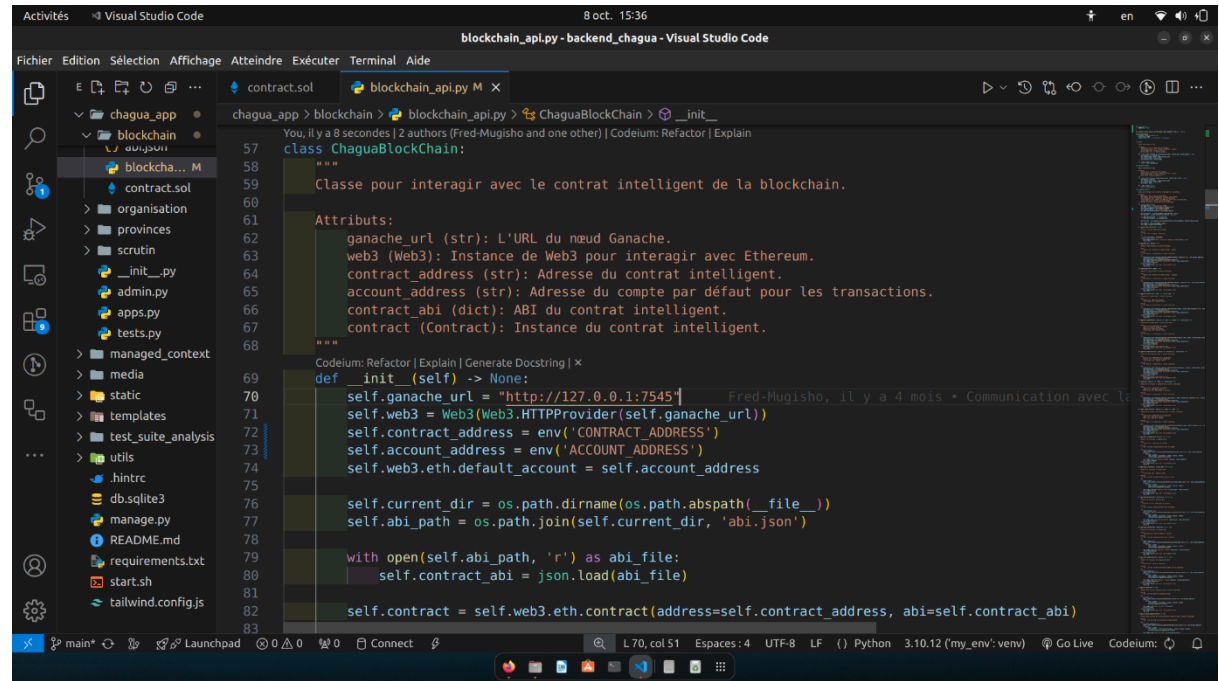


Figure 10 : Visual Studio Code (VS Code)

Visual Studio Code, illustré sur la figure 10, est un éditeur de code source léger mais puissant, conçu pour fonctionner sur les systèmes d'exploitation Windows, MacOS et Linux. Il offre un support intégré pour les langages JavaScript, TypeScript et Node.js, tout en disposant d'un écosystème riche en extensions qui permettent de travailler avec d'autres langages et environnements d'exécution tels que C++, Python, PHP et autres langages de programmation. Cet éditeur est apprécié pour sa flexibilité, sa rapidité et ses fonctionnalités avancées, ce qui en fait un choix populaire parmi les développeurs. [29]

Ce logiciel nous a permis d'écrire le code source de notre projet de manière efficace et organisée, facilitant ainsi le développement et la gestion de notre application. Grâce à ses fonctionnalités telles que la complétion automatique, le débogage intégré et la gestion des versions, Visual Studio Code a grandement contribué à optimiser notre flux de travail et à améliorer la qualité du code produit.

3.2.2.2 Remix Ethereum

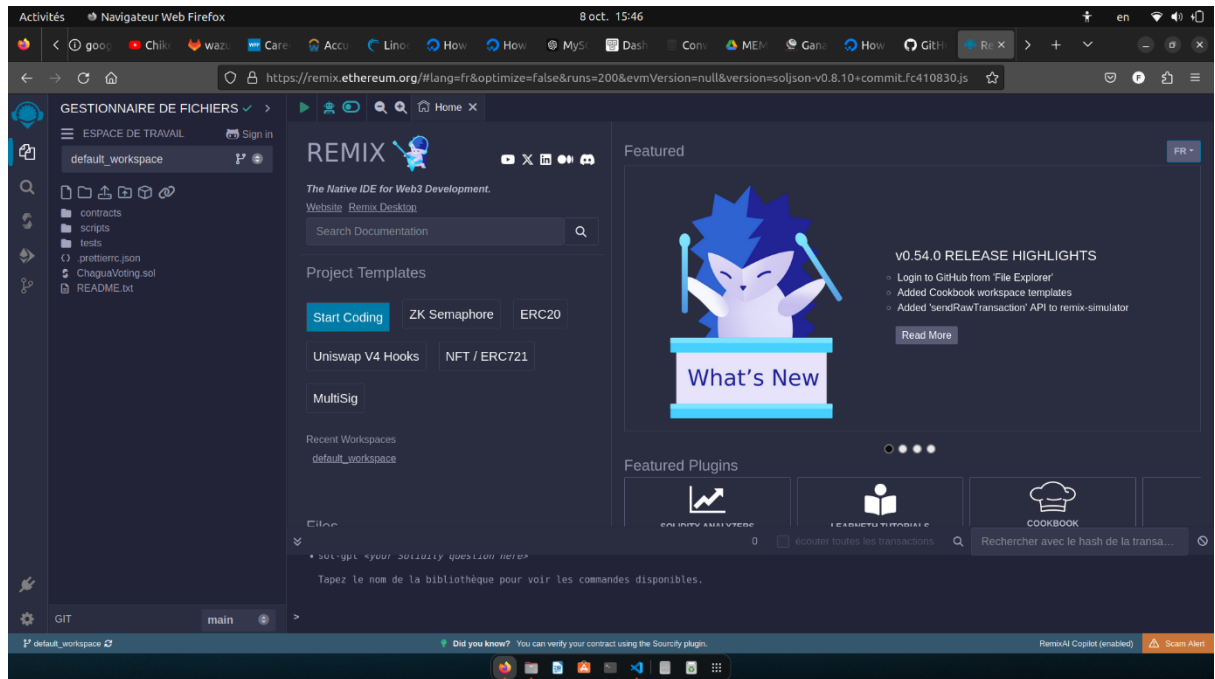


Figure 11 : Remix Ethereum IDE

Remix Ethereum est un environnement de développement intégré (IDE) très prisé pour le développement de contrats intelligents sur la blockchain Ethereum. Il est particulièrement apprécié pour sa convivialité et sa flexibilité, permettant aux développeurs de créer, tester et déployer des contrats intelligents de manière efficace. [30]

Remix fonctionne entièrement dans le navigateur, ce qui facilite l'accès aux fonctionnalités sans nécessiter d'installation complexe. Il offre une interface intuitive où les développeurs peuvent écrire du code en Solidity, le langage de programmation des contrats intelligents. Parmi ses nombreuses fonctionnalités, on trouve la compilation automatique du code, le débogage, et des outils d'analyse de sécurité qui aident à identifier les vulnérabilités potentielles dans le code. [31]

3.2.2.3 Ganache

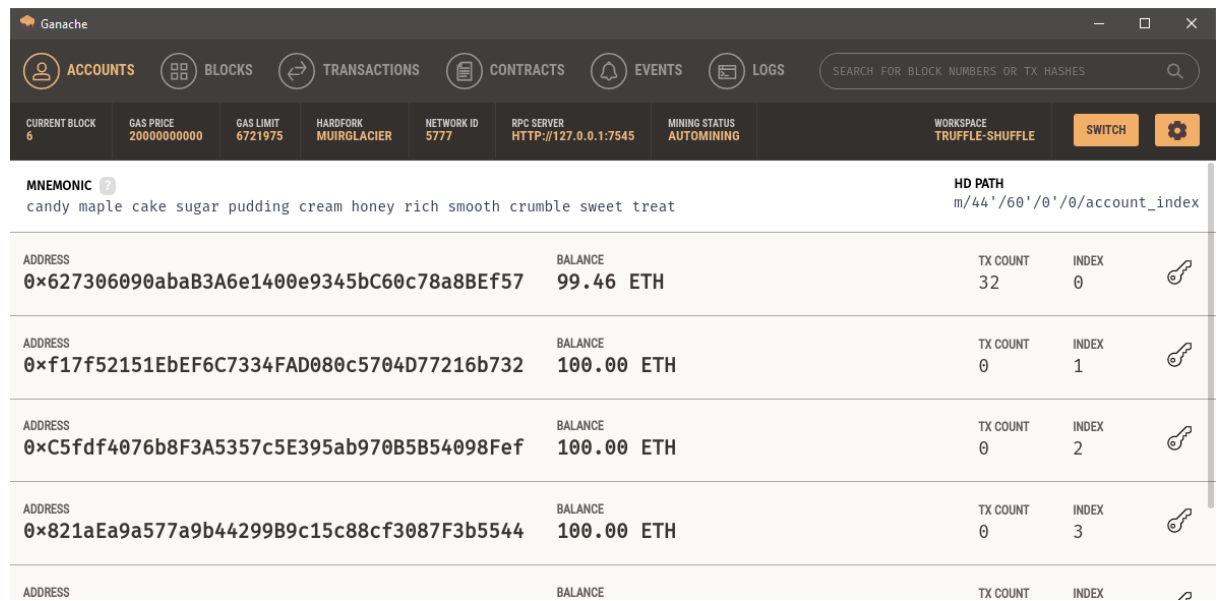


Figure 12 : Ganache

Ganache est un outil puissant de la Truffle Suite permettant de simuler une blockchain Ethereum localement. Il facilite le développement, le test et le débogage de contrats intelligents en fournissant un environnement contrôlé. Grâce à **Ganache**, les développeurs peuvent interagir avec des blocs, des transactions et des comptes simulés pour tester des DApps (*Decentralized Apps*) avant leur déploiement sur le réseau principal. [32]

3.2.3 Technologie utilisée

3.2.3.1 Python

Python est un langage de programmation interprété, orienté objet et de haut niveau, développé en 1991 par Guido van Rossum. Il est populaire pour sa simplicité et sa syntaxe claire, et ses capacités à intervenir dans divers domaines dont le développement web, l'automatisation des tâches, l'analyse de données et l'intelligence artificielle, Python bénéficie d'un écosystème riche en bibliothèques open source. [33]

Dans notre projet, nous l'avons utilisé pour le développement du backend et la communication avec la blockchain grâce à la bibliothèque web3.py.

web3.py est une bibliothèque **Python** permettant d'interagir avec la blockchain Ethereum. Elle offre des fonctionnalités pour la gestion de contrats intelligents, l'envoi de transactions et l'interaction avec des nœuds Ethereum. Cette bibliothèque facilite l'intégration des applications **Python** avec des fonctionnalités de la blockchain, telles que la lecture de données de contrats, l'exécution de transactions et la gestion des comptes [34]

3.2.3.2 Django

Django est un framework **Python** de haut niveau, permettant un développement rapide de sites internet, sécurisés, et maintenables. Créé par des développeurs expérimentés, Django prend en charge la plupart des tracas du développement web, vous pouvez donc vous concentrer sur l'écriture de votre application sans avoir besoin de réinventer la roue. Il est gratuit, open source, a une communauté active, une bonne documentation, et plusieurs options pour du support. [35]

3.2.3.3 Celery et Redis

Celery est une bibliothèque **Python** qui permet d'exécuter des tâches en arrière-plan, sans bloquer le programme principal. Elle est souvent utilisée avec **Redis** (REmote DIctionary Server), qui sert à stocker temporairement les tâches en mémoire pour qu'elles soient traitées rapidement. [36] [37]

Dans notre projet, nous avons utilisé cette combinaison pour rendre plus fluide le lancement des scrutins et automatiser l'envoi des e-mails aux électeurs et candidats. Cela a permis d'optimiser les processus et d'améliorer l'efficacité du système.

3.2.3.4 Solidity

Solidity est un langage de programmation orienté objet, similaire à JavaScript, Python ou C++. Il est principalement utilisé pour développer des contrats intelligents (smart-contract) sur la blockchain Ethereum, mais il fonctionne également sur d'autres blockchains compatibles comme Tomochain. Pour faciliter l'apprentissage des développeurs, Solidity emprunte des concepts à d'autres langages de programmation, ce qui permet une adoption rapide, même pour les novices. [38]

3.2.3.5 Tailwind CSS

Tailwind CSS est un framework CSS utilitaire qui permet de concevoir des interfaces utilisateur de manière rapide et efficace. Contrairement aux frameworks traditionnels, il offre des classes prédéfinies pour appliquer des styles directement dans une page HTML, ce qui favorise une approche personnalisée du design sans avoir à écrire beaucoup de CSS. Il est apprécié pour sa flexibilité et sa capacité à créer des designs modernes tout en minimisant le besoin d’avoir des feuilles de style séparées. [39]

3.3 Présentation des Interfaces

3.3.1 Accueil

La Figure 13 présente la page d'accueil de notre système, nommé CHAGUA. L'internaute a la possibilité de se connecter ou de rechercher un scrutin. Pour accéder à la page de connexion, il suffit de cliquer sur le bouton « **Se connecter** ». De même, pour rechercher un scrutin, l'utilisateur doit cliquer sur le bouton « **Chercher un scrutin** ».

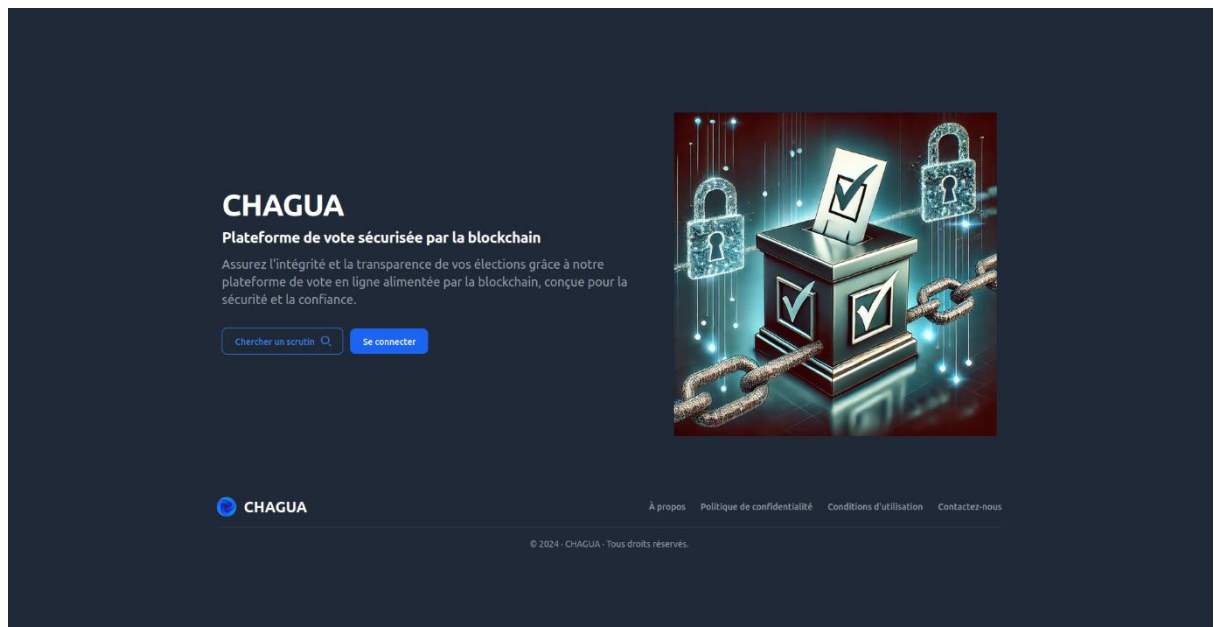


Figure 13 : Accueil

3.3.2 Interface d'authentification

La Figure 14 présente l'interface d'authentification du système. Cette interface permet à l'organisateur de scrutin de saisir ses identifiants afin d'accéder au système. Elle comprend un champ pour l'adresse email et un autre pour le mot de passe, ainsi qu'un bouton « **Se connecter** » pour valider les informations saisies. L'interface inclut également un bouton « **Créer un compte** », permettant aux nouveaux utilisateurs de s'inscrire, ainsi qu'un lien « **Mot de passe oublié** » pour récupérer leurs identifiants en cas d'oubli.

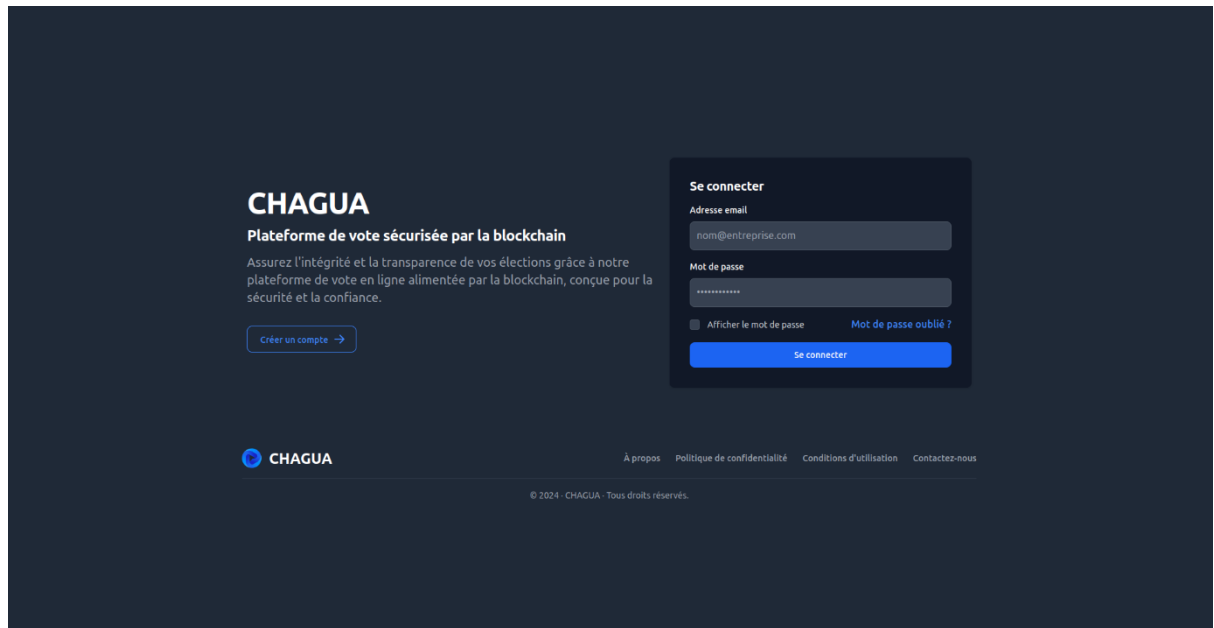


Figure 14 : Interface d'authentification

3.3.3 Créer un compte

La Figure 15 présente l'interface de création d'un compte organisateur du système. Cette interface permet aux internautes souhaitant devenir organisateurs de vote de saisir leurs informations personnelles. Les données requises incluent le nom complet, l'adresse e-mail, le sexe, le numéro de téléphone, une photo de profil, le nom de l'organisation, une description de celle-ci et le logo de l'organisation. Un bouton "Valider" est disponible pour soumettre les informations fournies. Une fois la validation effectuée, le système enverra un mot de passe à l'utilisateur par e-mail.

The image shows a registration form for CHAGUA titled "CHAGUA : Nouveau compte organisation". The form is set against a dark blue background. It contains the following fields and elements:

- Nom complet :** A text input field.
- Sexe :** A dropdown menu with "Masculin" selected.
- Email :** A text input field.
- Téléphone :** A text input field.
- Photo de profil :** A file upload area with a "Parcourir..." button and the text "Aucun fichier sélectionné."
- Nom de l'organisation :** A text input field.
- Description de votre organisation :** A text area.
- Logo de l'organisation :** A file upload area with a "Parcourir..." button and the text "Aucun fichier sélectionné."

At the bottom of the form, there are two buttons: "Annuler" (red) and "Valider" (blue). Below the form, the CHAGUA logo is displayed on the left, and a footer contains the text "À propos", "Politique de confidentialité", "Conditions d'utilisation", and "Contactez-nous". At the very bottom, it says "© 2024 - CHAGUA - Tous droits réservés."

Figure 15 : Créer un compte

3.3.4 Tableau de bord

La Figure 16 présente le tableau de bord de notre système. Cette interface permet à l'organisateur de scrutin de visualiser les différents scrutins qui ont été organisés récemment, en les catégorisant selon leur statut : "En cours de déroulement", "En attente de lancement" et "Récemment terminés". On peut également observer des statistiques de répartition des scrutins à travers des cartes colorées. En cliquant sur un scrutin, l'utilisateur est redirigé vers le tableau de bord de ce scrutin. De plus, cette interface inclut un menu latéral gauche qui permet à l'utilisateur de naviguer facilement dans le système.

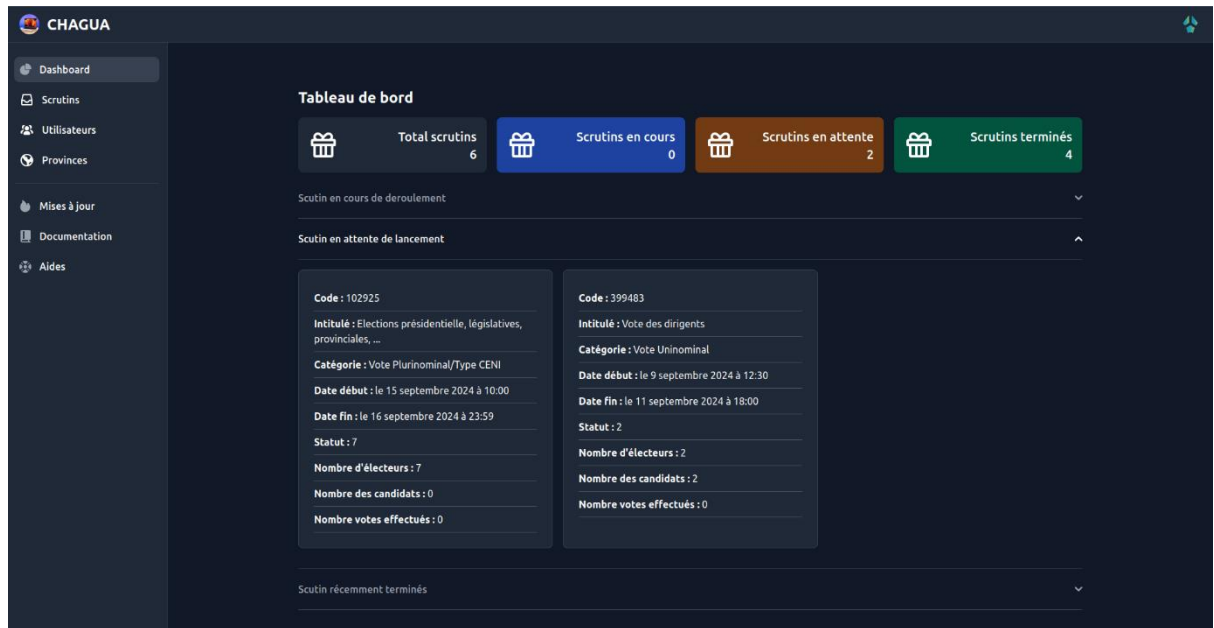


Figure 16 : Tableau de bord

3.3.5 Interface scrutins

La Figure 17 présente l'interface des scrutins de notre système. Cette interface permet à l'organisateur de scrutin de visualiser tous les scrutins associés à son compte. Pour chaque scrutin, un indicateur coloré (la blanche désigne un scrutin en attente, la jaune désigne le scrutin en cours et la verte désigne le scrutin terminé) en haut à droite de la carte indique son statut. En cliquant sur un scrutin, l'utilisateur est redirigé vers le tableau de bord de ce scrutin. Les significations de ces indicateurs sont fournies à la fin de la présentation des scrutins. De plus, l'interface inclut un bouton « Nouveau scrutin » qui permet à l'utilisateur de créer un nouveau scrutin.

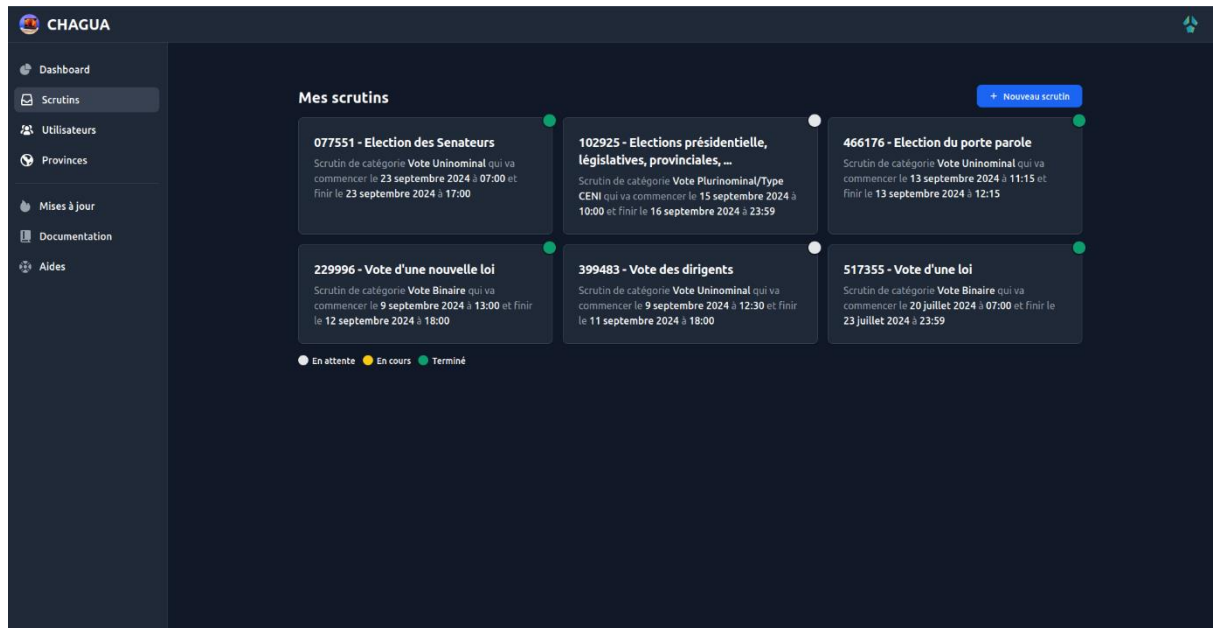


Figure 17 : Interface scrutins

3.3.6 Interface tableau de bord d'un scrutin

La Figure 18 présente l'interface du tableau de bord d'un scrutin de notre système. Cette interface affiche à l'organisateur de scrutin les informations générales et les paramètres du scrutin. Elle comprend deux boutons : le premier, « **Modifier** », permet de modifier certaines informations ou paramètres du scrutin, mais n'est disponible que lorsque le scrutin est en attente. Le second, « **Lancer le scrutin** », permet de démarrer le scrutin, mais cette opération est irréversible. Il est donc crucial de s'assurer que tout est bien configuré (types de candidature complétés, candidats ajoutés, électeurs chargés) avant de procéder. En cliquant sur ce bouton, le système déploie également le scrutin sur la blockchain, enregistrant tous les éléments du scrutin pour garantir son immutabilité. De plus, il envoie les jetons de vote aux électeurs et communique aux candidats le code du scrutin afin qu'ils puissent suivre son déroulement.

Hormis la section *Général* du menu, cette interface comprend également un menu avec les sections suivantes : **Types de candidature, Candidats, Électeurs, Votes, et Résultats**. Ces sections permettent de configurer les autres composants du scrutin. Nous allons présenter ces menus dans les prochaines interfaces.



Figure 18 : Interface tableau de bord d'un scrutin

3.3.7 Interface types de candidature

La Figure 19 présente l'interface des types de candidature du scrutin de notre système. Cette interface liste les types de candidature disponibles pour le scrutin, permettant ainsi aux candidats de postuler à ces différents types. Nous pouvons également remarquer deux boutons sur l'interface : le premier, « Nouvelle candidature », permet d'ajouter un nouveau type de candidature, tandis que le second, « Modifier », permet de modifier un type de candidature spécifique dans la liste. De plus, une section dédiée fournit une note concernant les types de candidature, qu'il est important de ne pas ignorer.



Figure 19 : Interface types de candidature

3.3.8 Interface candidats du scrutin

La Figure 20 présente l'interface des candidats du scrutin dans notre système. Cette interface affiche une liste des candidats déjà enregistrés pour le scrutin.

Elle comprend également trois boutons :

- ❖ **Nouveau candidat** : permet d'ajouter un nouveau candidat au scrutin,
- ❖ **Modifier** : permet de modifier les informations d'un candidat spécifique,
- ❖ **Supprimer** : permet de retirer un candidat du scrutin

CANDIDAT	NUMÉRO	CANDIDATURE	CIRCONSCRIPTION	VOTES
BAHATI KALUKA bahati@gmail.com 243970928194	102925@945	Député national	Ville de Goma	5 votes
BAHATI KALUKA bahati@gmail.com 243970928194	102925@683	Député provincial	Ville de Goma	1 vote
Bahati Kalume bahatikalume@chagua.io 243970123456	102925@573	Député national	Ville de Goma	3 votes
Bahati Kalume bahatikalume@chagua.io 243970123456	102925@548	Député provincial	Ville de Goma	5 votes
Banza Kalema banzakalema@chagua.io 243970990011	102925@240	Député national	Ville de Goma	1 vote
Banza Kalema banzakalema@chagua.io 243970990011	102925@281	Député provincial	Ville de Goma	1 vote
Beya Mulumba beyamulumba@chagua.io 243970567890	102925@239	Député national	Ville de Bukavu	2 votes

Figure 20 : Interface candidats du scrutin

3.3.9 Interface électeurs du scrutin

La Figure 21 présente l'interface des électeurs du scrutin dans notre système. Cette interface permet à l'organisateur de scrutin de consulter la liste complète des électeurs enregistrés pour le scrutin.

Elle comprend également deux boutons :

- ❖ **Charger les électeurs** : permet de charger les électeurs dans le scrutin en utilisant un fichier Excel modèle que l'utilisateur doit d'abord télécharger et compléter,
- ❖ **Supprimer** : permet de retirer un électeur spécifique du scrutin.

CHAGUA

Dashboard scrutins "Elections présidentielle, législatives, provinciales, ..."

Général Types de candidature Candidats **Electeurs** Votes Résultats

Rechercher... [Charger les électeurs](#)

N° D'IDENTIFICATION	NOM COMPLET	EMAIL	TÉLÉPHONE	CIRCONSCRIPTION	ACTIONS
A829388493	BAHATI KALUKA	bahati@gmail.com	243970928194	Ville de Goma	Supprimer
A829388500	Bahati Kalume	bahatikalume@chagua.io	243970123456	Ville de Goma	Supprimer
A829388509	Banza Kalema	banzakalema@chagua.io	243970990011	Ville de Goma	Supprimer
A829388516	Beya Mulumba	beyamulumba@chagua.io	243970567890	Ville de Bukavu	Supprimer
A829388515	Bintu Mumpasi	bintumumpasi@chagua.io	243970345678	Ville de Bukavu	Supprimer
A829388526	Dikembe Kadima	dikembekadima@chagua.io	243970223567	Ville de Béni	Supprimer
A829388494	ISHARA BAGALWA	ishara@gmail.com	243970928195	Lubero	Supprimer
A829388514	Ilunga Mavinga	ilungamavinga@chagua.io	243970432109	Ville de Bukavu	Supprimer
A829388497	Isabelle Kabuya	isabelle.kabuya@gmail.com	243970928198	Ville de Bukavu	Supprimer
A829388520	Kadiombo Lumumba	kadiombolumumba@chagua.io	243970912345	Kinshasa I	Supprimer

Figure 21 : Interface électeurs du scrutin

3.3.10 Interface charger les électeurs

La Figure 22 illustre l'interface de chargement des électeurs dans notre système. Sur cette interface, vous avez la possibilité de télécharger un fichier Excel modèle en cliquant sur un lien, ce fichier servant de base pour structurer les informations des électeurs à importer. Une fois le fichier complété, vous pouvez le charger directement dans le système. En cliquant sur le bouton « Charger », le système procède à l'importation des électeurs tout en vérifiant si chaque électeur est déjà inscrit dans la base de données.

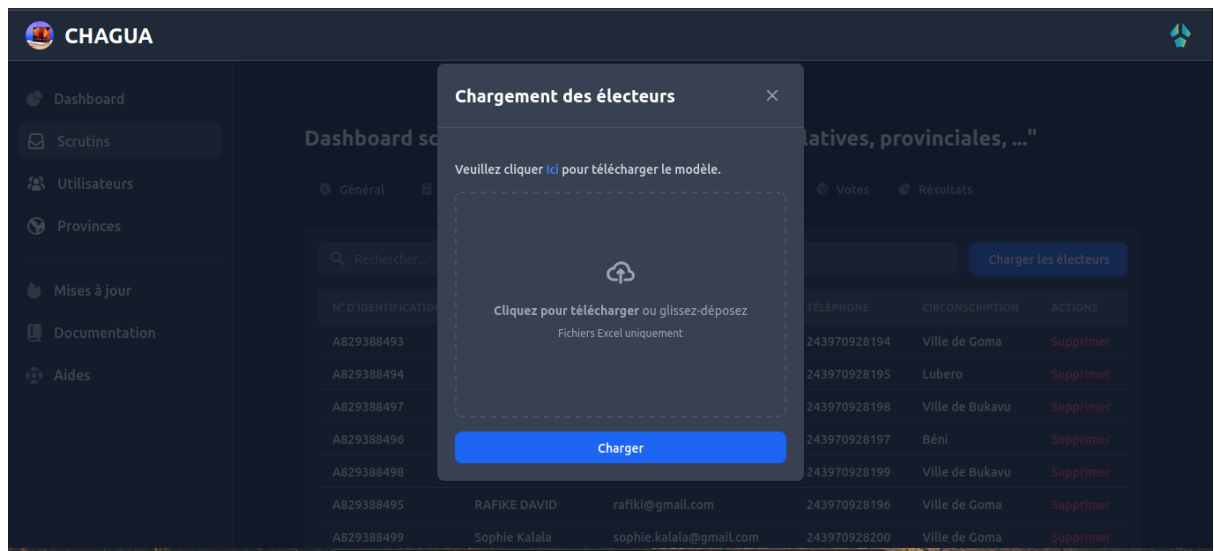


Figure 22 : Interface charger les électeurs

3.3.11 Interface votes du scrutin

La Figure 23 présente l'interface des votes du scrutin de notre système. Sur cette interface, vous pouvez consulter la liste de tous les votes enregistrés dans le scrutin.

The screenshot shows the CHAGUA application interface with the "Votes" section selected. A table displays the following data:

CANDIDAT	ELECTEUR	HASH VOTE	DATE & HEURE
102925@789 RAFIKE DAVID	78d4420fa30e150fc4fa6047...	5d47e928f075fa5647cf9e9...	9 octobre 2024 10:08
102925@398 Lucas Mbala	78d4420fa30e150fc4fa6047...	e679a58fa2d04ce1836e175d...	9 octobre 2024 10:08
102925@320 Kibala Lukusa	78d4420fa30e150fc4fa6047...	ae9fc00b72e440901267543...	9 octobre 2024 10:08
102925@789 RAFIKE DAVID	8323a781c5ab6741537f3035...	e722fa203e17d0428b2bc2c2...	9 octobre 2024 10:07
102925@417 Katanga Yav	8323a781c5ab6741537f3035...	b5c1a1ab00ef161354bcf2ae...	9 octobre 2024 10:07
102925@563 Kanku Matondo	8323a781c5ab6741537f3035...	2d2a38eab8cad1e65a86ef33...	9 octobre 2024 10:07
102925@396 Tshisekedi Kanku	1b61d8d548f181b58621661d...	011cd14c1a67a5eacdc5611c...	9 octobre 2024 10:06
102925@226 Ilunga Mavinga	1b61d8d548f181b58621661d...	18f05905330e466323b12407...	9 octobre 2024 10:06
102925@46 ISHARA BAGALWA	1b61d8d548f181b58621661d...	7b233fa4f572415054444774...	9 octobre 2024 10:06
102925@589 Sophie Kalala	db3a1e858512845edf006ab...	77e9047b45325fba538e64b...	9 octobre 2024 10:05
102925@341 Bintu Mumpasi	db3a1e858512845edf006ab...	4c9a86cf98f6357536e6332...	9 octobre 2024 10:05
102925@239 Beya Mulumba	db3a1e858512845edf006ab...	9ec4ac8163b362d7759a4c8f...	9 octobre 2024 10:05
102925@148 Ndaya Mayombo	f247fbc8d5b80e50a3795f56...	00530e2a19dfee4bcf78afa6...	9 octobre 2024 10:05
102925@548 Bahati Kalume	f247fbc8d5b80e50a3795f56...	ae94a480922154509f4ad23d...	9 octobre 2024 10:05
102925@945 BAHATI KALUKA	f247fbc8d5b80e50a3795f56...	b5a9f2355ba0b10501622a03...	9 octobre 2024 10:05
102925@789 RAFIKE DAVID	46d2aa30829632c450cd837...	96ea904b35588bdcc49ec31f...	9 octobre 2024 10:04

Figure 23 : Interface votes du scrutin

3.3.12 Interface résultats du scrutin

La Figure 24 présente l'interface des résultats du scrutin de notre système. Sur cette interface, vous pourrez consulter les résultats du scrutin une fois publiés. Vous remarquerez que les

résultats sont présentés en sections correspondant aux différents types de candidatures du scrutin.

Président	Député national		
	CANDIDAT	NUMÉRO	VOTES
Député na...	BAHATI KALUKA bahati@gmail.com	102925@945	5 votes (13,89%)
Député pr...	Bintu Mumpasi bintumumpasi@chagua.io	102925@981	3 votes (8,33%)
	Tshibanda Kayembe tshibandakayembe@chagua.io	102925@696	1 vote (2,78%)
	ISHARA BAGALWA ishara@gmail.com	102925@46	1 vote (2,78%)
	Dikembe Kadima dikembekadima@chagua.io	102925@935	1 vote (2,78%)

Figure 24 : Interface résultats du scrutin

3.3.13 Interfaces utilisateurs

La Figure 25 présente l'interface des utilisateurs de notre système. Cette interface permet à l'organisateur de scrutin de visualiser la liste de tous les utilisateurs associés à son compte organisation. En cliquant sur le bouton « Détails », le système vous redirige vers l'interface de détails de l'utilisateur sélectionné, où vous pouvez effectuer d'autres actions sur le compte, telles que mettre à jour ses informations ou changer le rôle de l'électeur. De plus, l'interface inclut un bouton « Nouvel utilisateur », permettant à l'administrateur d'ajouter un nouvel utilisateur.

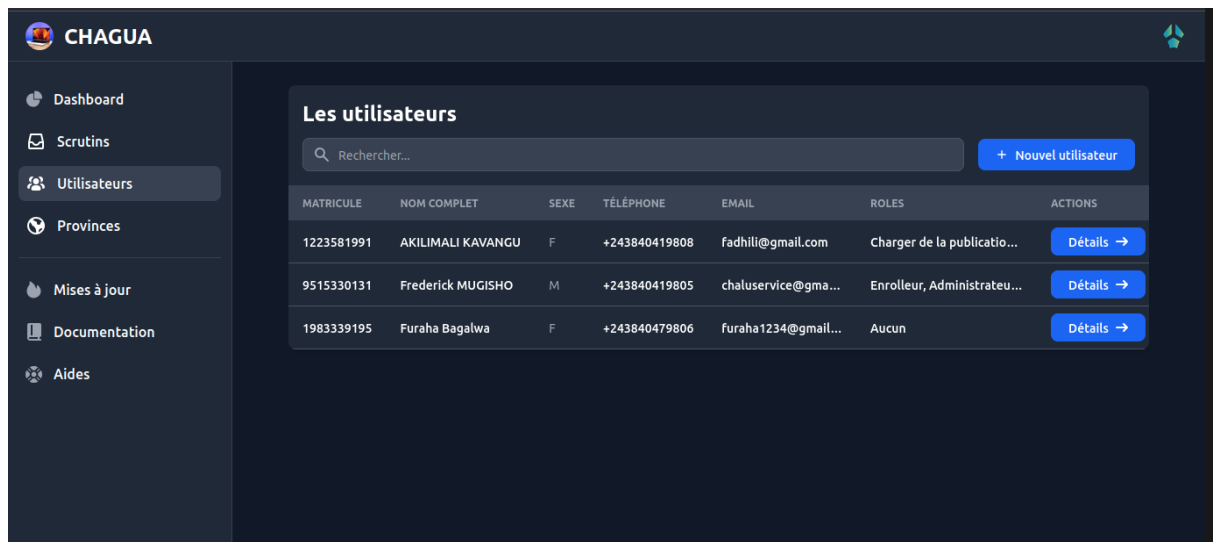


Figure 25 : Interfaces utilisateurs

3.3.14 Interface chercher scrutin

La Figure 26 présente l'interface de recherche du scrutin de notre projet. Sur cette interface, vous devez simplement entrer le code du scrutin reçu de l'organisateur. Le système vérifiera si le scrutin est accessible et s'il existe, puis redirigera l'internaute vers une interface affichant les détails du scrutin. Vous y trouverez les informations générales, la liste des candidats, les votes déjà enregistrés, ainsi que les résultats. Un bouton est également disponible pour accéder à la page de vote où vous pourrez exprimer votre choix.

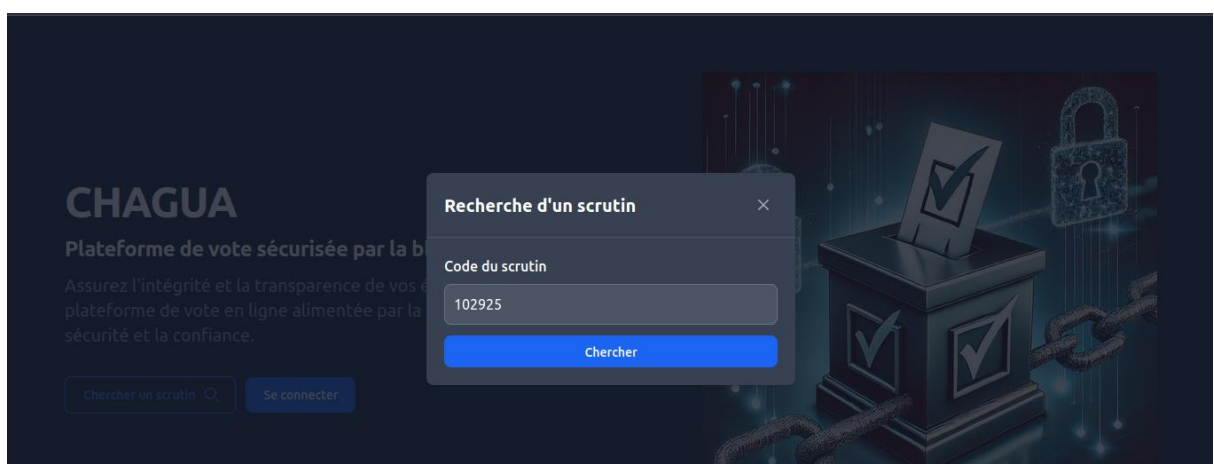


Figure 26 : Interface chercher scrutin

3.3.15 Interface sélectionner candidat pour vote

La Figure 27 présente l'interface de sélection des candidats pour le vote dans notre système. Cette interface permet à l'électeur de choisir son candidat. Lorsqu'un candidat est sélectionné, son indicateur de statut devient vert, et il est ajouté au bulletin de vote. Il est important de noter que l'électeur ne peut choisir qu'un seul candidat par type de candidature.

L'interface comprend également deux boutons : le premier, « **Sélectionner** », permet d'ajouter le candidat sur bulletin de vote, et le second, « **Supprimer** », permet de retirer le candidat du bulletin de vote de l'électeur.

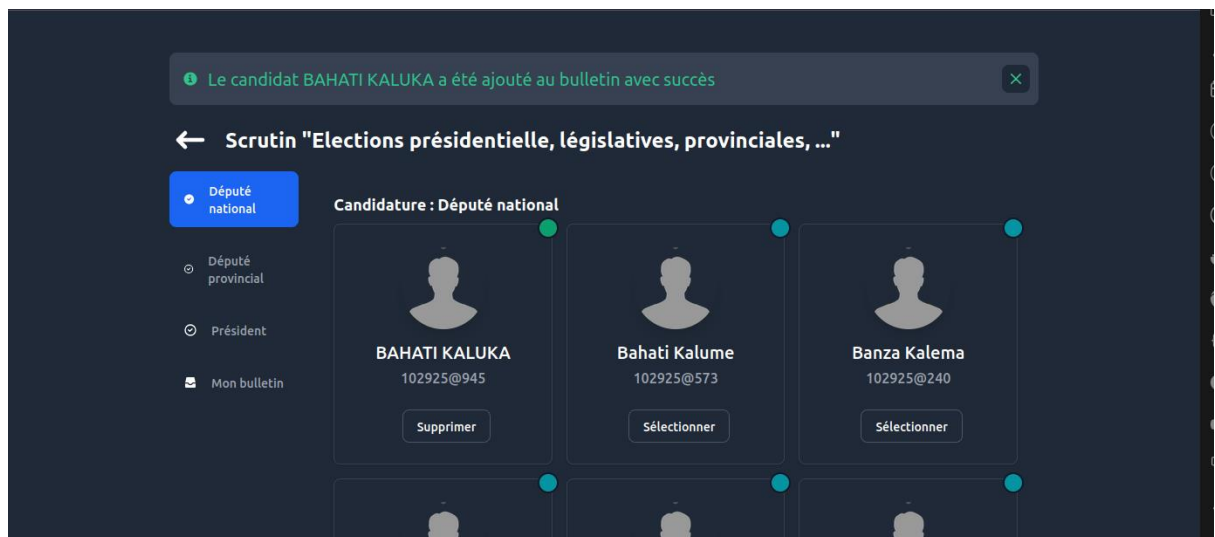


Figure 27 : Interface sélectionner candidat pour vote

3.3.16 Interface de visualisation du bulletin

La Figure 28 présente l'interface de visualisation du bulletin de notre système. Cette interface permet à l'électeur de visualiser ses choix de candidats avant d'envoyer son bulletin pour enregistrement dans la blockchain via le bouton « Envoyer le bulletin ». Si l'électeur a commis une erreur, il peut revenir pour modifier ses choix avant d'envoyer son bulletin.

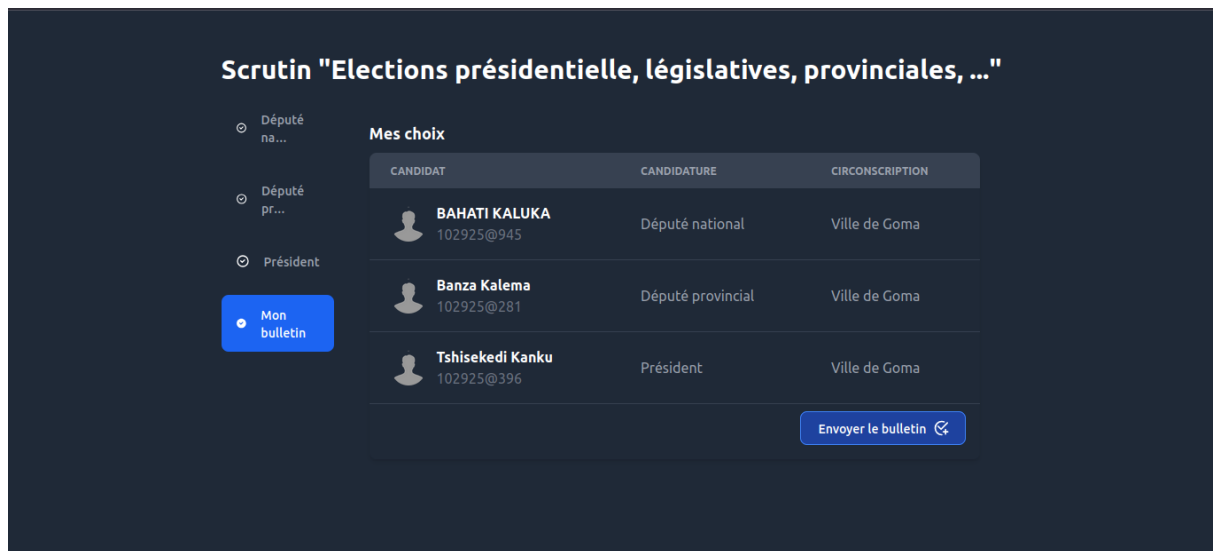


Figure 28 : Interface de visualisation du bulletin

3.3.17 Interface envoyer le bulletin

La Figure 29 présente l'interface envoyer le bulletin de notre système. Sur cette interface, l'électeur doit simplement saisir le jeton de vote que vous avez reçu de l'organisateur du scrutin et cliquer sur le bouton « Confirmer » pour envoyer votre bulletin. Le système vérifiera bien entendu la validité du jeton que vous avez saisi avant d'enregistrer votre bulletin. Une fois que vous avez envoyé votre bulletin, le système vous enverra un e-mail de confirmation pour vous informer que votre vote a bien été enregistré.

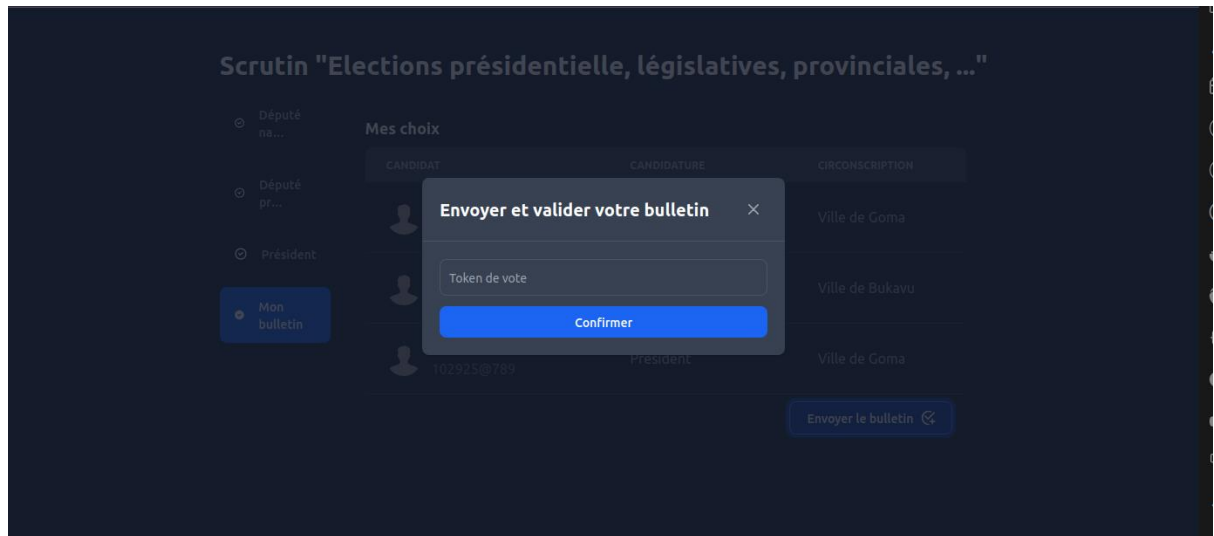


Figure 29 : Interface envoyer le bulletin

3.4 Présentation des tests

Dans cette section, nous détaillerons les différents tests effectués pour évaluer la robustesse, la sécurité et la performance de notre solution basée sur la blockchain. Ces tests ont pour objectif de montrer la fiabilité et l'efficacité de notre système par rapport à la solution classique.

3.4.1 Test de performance sous charge

Un test de charge consiste à effectuer un test permettant de mesurer la performance d'un système en fonction de la charge d'utilisateurs simultanées. [40]

La Figure 30 illustre les résultats d'un test de charge effectué avec **Locust** (un outil open-source de test de charge qui permet de simuler un grand nombre d'utilisateurs interagissant avec un site web ou une application). Locust est conçu pour évaluer la performance et la stabilité des systèmes sous diverses charges, en fournissant une interface simple et intuitive pour la configuration et l'exécution des tests [41]). Lors de ce test, 1000 utilisateurs ont été simulés pour analyser les performances de deux éléments essentiels : la page d'accueil (GET /) et la page de connexion (GET /login). Au total, 9464 requêtes ont été traitées avec un débit global de 54,8 requêtes par seconde. Le système a démontré une grande stabilité, puisqu'aucune erreur n'a été enregistrée tout au long du processus.

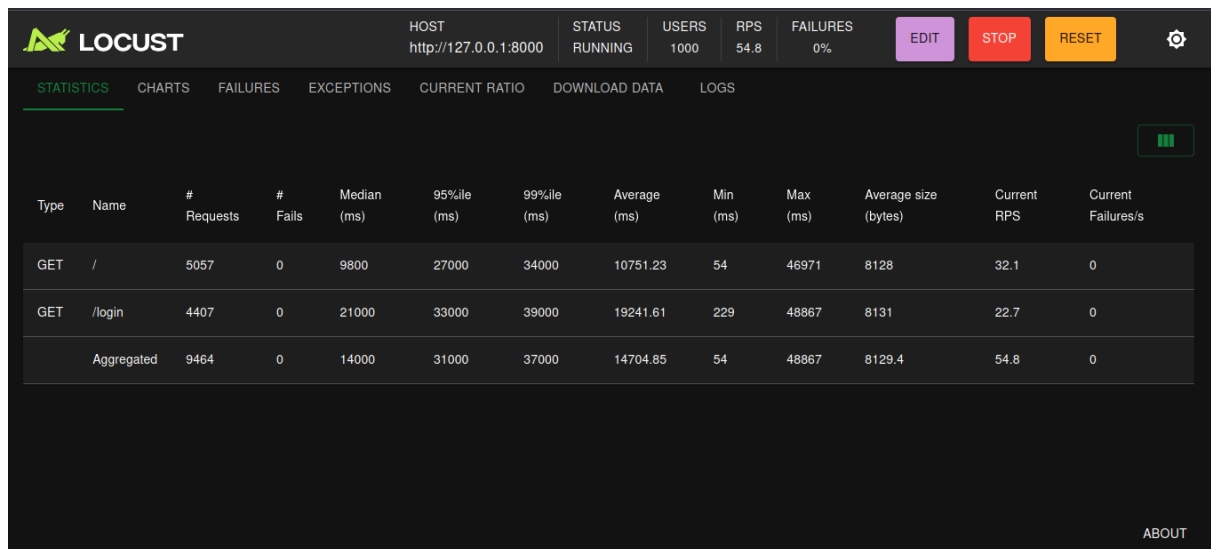


Figure 30 : Test de performance sous charge

3.4.2 Test de résistance à une attaque DDoS

Une attaque DDoS (Distributed Denial of Service) est une cyberattaque qui consiste à submerger une ressource en ligne, comme un site web ou un serveur, avec un volume massif de requêtes provenant de multiples sources, dans le but de rendre la ressource indisponible pour les utilisateurs légitimes. [42]

La Figure 31 montre comment notre système réagit face à une attaque DDoS. On peut voir que lorsque quelqu'un envoie un trop grand nombre de requêtes en très peu de temps, le système intervient pour bloquer cet utilisateur. Par exemple, les erreurs HTTP 429 ("Trop de requêtes") indiquent que le système a bien détecté et limité les utilisateurs qui tentent de submerger le serveur avec trop de demandes. D'autres erreurs, comme "ConnectionResetError" ou "ConnectionRefusedError", montrent que le système a également coupé ou refusé certaines connexions, protégeant ainsi nos ressources.

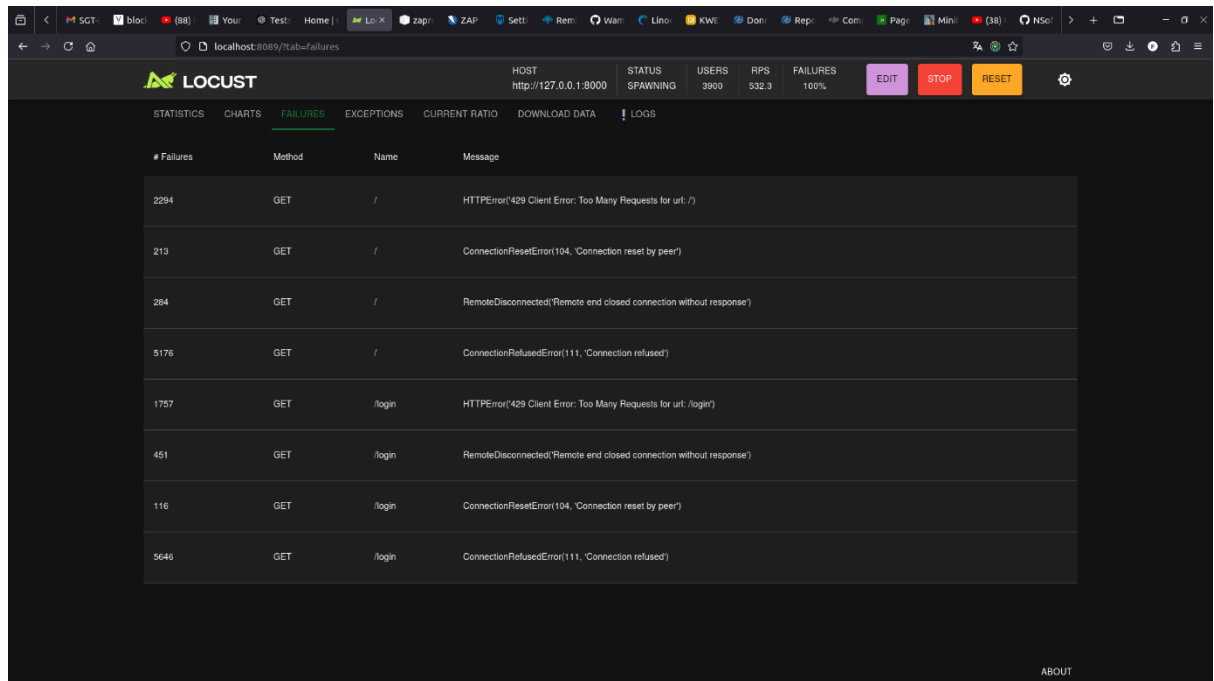


Figure 31 : Test de résistance à une attaque DDoS

3.4.3 Test de brute forcing

Une attaque par force brute consiste à utiliser une méthode d'essai-erreur systématique afin de deviner des identifiants, des clés de chiffrement ou de localiser une page web protégée. Les attaquants testent toutes les combinaisons possibles jusqu'à trouver la bonne, exploitant ainsi des vulnérabilités liées à la sécurité des authentifications. [43]

Nous avons également effectué un test de brute forcing pour évaluer la résistance de notre système face aux tentatives d'accès non autorisées. Dans le but de protéger les comptes utilisateurs, nous avons mis en place un mécanisme de verrouillage automatique qui désactive l'accès après cinq tentatives de connexion échouées. Cette mesure a été soigneusement conçue comme une protection supplémentaire pour éviter tout accès non autorisé et garantir la sécurité des données du système.

3.5 Justification de la supériorité de la solution basée sur Blockchain

Du point de vue sécuritaire, notre système se distingue des systèmes traditionnels grâce à la blockchain, qui offre une immutabilité des données, une transparence et une vérifiabilité, renforçant ainsi la confiance des utilisateurs. De plus, il présente une meilleure résilience face aux attaques potentielles, telles que les attaques DDoS, les tentatives de brute forcing, ainsi que les menaces courantes des applications web (injections SQL, XSS, CSRF, etc.). Sa structure décentralisée le rend moins vulnérable aux perturbations causées par un trafic malveillant. Les tests réalisés démontrent également une protection efficace contre les accès non autorisés, grâce à des mécanismes tels que le verrouillage de compte après plusieurs tentatives échouées.

3.6 Conclusion partielle

Dans ce chapitre, nous nous sommes concentrés sur la présentation des résultats, des outils et des technologies utilisés pour mettre en pratique l'ensemble des éléments abordés au cours de l'étude et de la conception de ce projet. Nous avons présenté nos résultats à travers différentes figures qui illustrent les interfaces que nous avons réalisées. De plus, nous avons effectué plusieurs tests, notamment des tests de performance sous charge, des tests de résistance aux attaques DDoS et des tests de brute forcing.

Ces évaluations ont montré la robustesse et la sécurité de notre système. Bien qu'elles indiquent des résultats prometteurs, un test de falsification serait nécessaire pour établir définitivement les avantages de notre solution basée sur la blockchain par rapport aux systèmes traditionnels. Cependant, en considérant la blockchain comme une entité de confiance et fiable, nous sommes convaincus que notre système est protégé contre la falsification des données.

Conclusion générale

Notre projet de fin d'études a consisté à concevoir et mettre en place un système de vote en ligne basé sur la technologie Blockchain afin de garantir la sécurité et la transparence des élections en RDC. Notre travail a été divisé en trois chapitres principaux et une introduction générale.

De par les résultats obtenus dans notre travail, nous avons eu les réponses à nos questions de recherche :

1. De quels moyens avons-nous besoin pour créer un système de vote en ligne robuste utilisant la technologie blockchain ?

Selon nos observations au cours de cette recherche, la réalisation d'un système de vote en ligne efficace nécessite une infrastructure technologique solide, incluant des serveurs sécurisés, une connectivité Internet fiable et des protocoles de sécurité avancés. Il est également crucial de choisir des technologies de développement appropriées pour garantir la robustesse et la performance du système.

2. Quelle blockchain choisir pour développer notre solution et pourquoi ce choix ?

Nous avons opté pour la blockchain Ethereum grâce à ses fonctionnalités de smart contracts, qui rendent possibles l'automatisation des processus de vote. De plus, sa vaste communauté de développeurs assure un bon niveau de support et favorise l'innovation. Ce choix s'est révélé idéal pour répondre aux besoins de notre projet.

3. Quels sont les défis à l'implémentation d'un système de vote en ligne basé sur la blockchain et comment les surmonter ?

Parmi les principaux défis rencontrés, la scalabilité pour gérer un grand volume de transactions est essentielle. Pour surmonter ces obstacles, il est crucial d'optimiser les processus de vote tout en mettant en place un équilibrage de charge sur le serveur de

production. De plus, assurer la sécurité des données contre les cyberattaques est fondamental. Cela peut être réalisé grâce à des audits réguliers et à l'implémentation de mécanismes de sécurité robustes.

4. Comment implémenter un système de vote en ligne basé sur la blockchain ?

Nous avons commencé par créer notre smart contract, chargé de gérer le processus de vote. Ensuite, nous avons mis en place un backend fiable pour assurer la communication entre notre application et la blockchain, tout en intégrant des mécanismes de vérification et de validation des votes afin de garantir la transparence et la fiabilité de notre système. Par la suite, nous avons développé le frontend et l'avons intégré au backend. Enfin, nous avons effectué des tests pour nous assurer du bon fonctionnement de l'ensemble du système.

Notre travail s'est déroulé en trois parties principales : l'analyse, la conception et l'implémentation. Pour l'implémentation, nous avons choisi le framework web Django pour le backend, utilisant le langage de programmation Python. Pour le frontend, nous avons intégré HTML et CSS avec Tailwind. Le développement du smart contract a été réalisé en Solidity, et nous avons sélectionné la blockchain Ethereum comme protocole pour garantir la sécurité et la transparence de notre système.

Enfin, étant donné que le domaine de l'informatique est extrêmement vaste et en constante évolution, notamment en ce qui concerne la sécurité informatique, nous ne pouvons pas prétendre avoir épuisé ce sujet. C'est pourquoi nous encourageons d'autres chercheurs et développeurs à contribuer au projet afin de combler nos insuffisances et imperfections. Voici quelques pistes d'amélioration :

- ❖ **Optimiser les performances** : Nous avons posé les bases, mais il reste du travail à faire pour rendre le système plus rapide, surtout dans les environnements à grande échelle. Une étude approfondie des algorithmes pourrait permettre de réduire les temps de traitement et d'améliorer la fluidité globale.

- ❖ **Étendre la compatibilité** : À ce stade, notre système n'est pas compatible avec toutes les plateformes. Il serait bénéfique de l'adapter aux appareils comme les mobiles, desktop, afin d'élargir son accessibilité.
- ❖ **Renforcer la sécurité** : L'intégration de technologies comme la **cryptographie homomorphique**, qui permet de réaliser des calculs sur des données chiffrées sans les déchiffrer, renforcerait la sécurité du processus d'organisation du scrutin. Cela apporterait une protection supplémentaire contre les attaques potentielles.
- ❖ **Mener des tests de sécurité approfondis** : Des tests de sécurité supplémentaires sont nécessaires, car ceux réalisés jusqu'à présent ne prouvent pas directement la robustesse de la technologie choisie. Ils confirment uniquement la robustesse de la plateforme. Ces nouveaux tests permettront de prouver la sécurité spécifique de notre solution

Bibliographie

- [1 A. Goujon, «CAIRN.INFO,» 28 Avril 2022. [En ligne]. Available:
] <https://www.cairn.info/les-democraties--9782200287276-page-23.htm>. [Accès le 9 Juin 2024].
- [2 Binance, «Binance et la BNB Chain : quelle est la différence ?,» Binance, 21 decembre
] 2024. [En ligne]. Available: <https://www.binance.com/fr/blog/ecosystem/binance-et-la-bnb-chain%C2%A0-quelle-est-la-diff%C3%A9rence%C2%A0-6186229776062522437>. [Accès le 1 juillet 2024].
- [3 Élections Québec, «Vocabulaire des élections,» [En ligne]. Available:
] <https://www.electionsquebec.qc.ca/comprendre/comprendre-la-democratie-et-les-elections/vocabulaire-des-elections/>. [Accès le 2024 juin 27].
- [4 V. publique, «Quels sont les différents modes de scrutin ?,» 19 9 2022. [En ligne].
] Available: <https://www.vie-publique.fr/fiches/23948-quels-sont-les-differents-modes-de-scrutin>. [Accès le 2 10 2024].
- [5 MINISTERE DE L'EUROPE ET DES AFFAIRES ETRANGERES, «Modalités de vote,»
] [En ligne]. Available: <https://www.diplomatie.gouv.fr/fr/services-aux-francais/elections-a-l-etranger/modalites-de-vote/>. [Accès le 1 10 2024].
- [6 NDI, «Internet Voting,» NDI, 17 12 2013. [En ligne]. Available: <https://www.ndi.org/e-voting-guide/internet-voting>. [Accès le 28 9 2024].
- [7 CENI, «Commission Electorale Nationale Indépendante,» [En ligne]. Available:
] <https://www.ceni.cd/>. [Accès le 2 juillet 2024].
- [8 H. ISMAIL, «Utilisation de Blockchain pour la sécurisation des données ERPs,»
] Université Mohamed Khider – BISKRA, Alger, 2022.
- [9 coinjournal, «Comment fonctionne la blockchain ?,» [En ligne]. Available:
] <https://coinjournal.net/fr/crypto-monnaies/apprendre/blockchain/>. [Accès le 1 juillet 2024].

- [1 S. Tessier, *Fonctionnement de la blockchain et son intérêt pour le*, Bordeaux: dumas, 2019. 0]
- [1 Oracle, «Qu'est-ce que Blockchain ?», [En ligne]. Available: <https://www.oracle.com/ca-1 fr/blockchain/what-is-blockchain/>. [Accès le 27 juin 2024].
- [1 IBM, «Qu'est-ce que la blockchain ?», [En ligne]. Available: <https://www.ibm.com/fr-2 fr/topics/blockchain>. [Accès le 27 juin 2024].
- [1 L. Lars, «Qu'est-ce qu'un bloc dans la technologie blockchain ?», *Cryptoast*, 27 juin 2022. 3] [En ligne]. Available: <https://cryptoast.fr/bloc-blockchain-crypto-explication/>. [Accès le 27 juin 2024].
- [1 M. Prué, «Qu'est-ce qu'un nœud dans l'univers de la blockchain ?», *Cryptoast*, 30 janvier 4] 2023. [En ligne]. Available: <https://cryptoast.fr/noeud-cryptomonnaies/>. [Accès le 27 juin 2024].
- [1 G. Barbey, «Tout ce qu'il faut savoir sur la cryptomonnaie et la blockchain», *heidi*, 31 août 5] 2021. [En ligne]. Available: <https://www.heidi.news/cyber/tout-ce-qu-il-faut-savoir-sur-la-cryptomonnaie-et-la-blockchain>. [Accès le 1 Juillet 2024].
- [1 IBM, «Que sont les contrats intelligents sur la blockchain ?», IBM, [En ligne]. Available: 6] <https://www.ibm.com/fr-fr/topics/smart-contracts>. [Accès le 27 juin 2024].
- [1 metlabs, «Qu'est-ce qu'un hachage dans la blockchain ? Concepts et caractéristiques de 7] base», metlabs, [En ligne]. Available: <https://metlabs.io/fr/quest-ce-quun-hachage-dans-la-blockchain-concepts-et-caracteristiques-de-base/>. [Accès le 27 juin 2024].
- [1 C. L. J.-H. M. M. R. Olivier Depierre, «Centre de droit bancaire et financier», *Centre de 8] droit bancaire et financier*, 17 septembre 2022. [En ligne]. Available: <https://cdbf.ch/lexique/consensus/>. [Accès le 27 juin 2024].
- [1 Young Platform academy, «La signification de la blockchain : ce qu'elle est et comment 9] elle fonctionne», Young Platform academy, 22 mai 2023. [En ligne]. Available: <https://academy.youngplatform.com/fr/blockchain/blockchain-qu-est-signification-definition-fonctionnement/>. [Accès le 15 juin 2024].

- [2 M. Rachid, «Réalisation d'un système de vote en ligne avec la blockchain,» Université 0] Saad Dahleb Blida, Algérie, 2021.
- [2 Bitcoin, Bitcoin, [En ligne]. Available: <https://bitcoin.org/fr/>. [Accès le 1 juillet 2024].
1]
- [2 T. Bourbotte, «Qu'est-ce que le Bitcoin (BTC),» cryptoast, 19 avril 2024. [En ligne].
2] Available: <https://cryptoast.fr/bitcoin/>. [Accès le 2024 juillet 2024].
- [2 Ethereum, «Qu'est-ce qu'Ethereum,» Ethereum, [En ligne]. Available:
3] <https://ethereum.org/fr/learn/>. [Accès le 1 juillet 2024].
- [2 T. Bourbotte, «C'est quoi Ethereum,» cryptoast, 31 mai 2024. [En ligne]. Available:
4] <https://cryptoast.fr/fiche-ethereum/>. [Accès le 1 juillet 2024].
- [2 T. Bourbotte, «Qu'est-ce que Cardano (ADA) ?,» 9 août 2023. [En ligne]. Available:
5] <https://cryptoast.fr/fiche-cardano/>. [Accès le 1 juillet 2024].
- [2 IBM, «Qu'est-ce que Hyperledger Fabric ?,» IBM, [En ligne]. Available:
6] <https://www.ibm.com/fr-fr/topics/hyperledger>. [Accès le 1 juillet 2024].
- [2 Talend, «Qu'est-ce que la technologie Blockchain et à quoi sert-elle ?,» Talend, [En ligne].
7] Available: <https://www.talend.com/fr/resources/blockchain/>. [Accès le 1 juillet 2024].
- [2 CENI, «ceni.cd,» Ceni, 15 6 2023. [En ligne]. Available: <https://www.ceni.cd/cadre-legal/loi-ndeg23025-du-15-juin-2023-portant-repartition-des-sieges-par-circonscription>.
8] [Accès le 2024 9 20].
- [2 Microsoft, «Visual Studio Code,» Microsoft, [En ligne]. Available:
9] <https://visualstudio.microsoft.com/fr/#:~:text=Visual%20Studio%20Code%20est%20un,pour%20Windows%2C%20macOS%20et%20Linux..> [Accès le 8 10 2024].
- [3 Tokize, «Qu'est-ce que Remix Ethereum ?,» Tokize, 22 6 2023. [En ligne]. Available:
0] <https://www.tokize.com/fr/remix-ethereum-smart-contracts/#h-qu-est-ce-que-remix-ethereum>. [Accès le 8 10 2024].
- [3 Remix, «REMIX IDE Documentation,» Remix, [En ligne]. Available: <https://remix-project.org/?lang=fr>. [Accès le 8 10 2024].
1]

- [3 Truffle, «Ganache One Click Blockchain,» [En ligne]. Available:
2] <https://archive.trufflesuite.com/ganache/>. [Accès le 8 10 2024].
- [3 AWS, «Qu'est-ce que Python ?,» AWS, [En ligne]. Available:
3] <https://aws.amazon.com/fr/what-is/python/>. [Accès le 8 10 2024].
- [3 Ethereum Foundation, «What is web3.py,» [En ligne]. Available:
4] <https://web3py.readthedocs.io/en/stable/>. [Accès le 8 10 2024].
- [3 Mozilla, «Introduction à Django,» Mozilla, [En ligne]. Available:
5] https://developer.mozilla.org/fr/docs/Learn/Server-side/Django/Introduction#quest_ce_que_django. [Accès le 8 10 2024].
- [3 C. DELEUZE, «Distribuer des tâches en Python avec Celery,» 25 05 2021. [En ligne].
6] Available: <https://www.pythoniste.fr/python/celery/distribuer-des-taches-en-python-avec-celery/>. [Accès le 08 10 2024].
- [3 AWS, «Qu'est-ce que Redis ?,» AWS, [En ligne]. Available:
7] <https://aws.amazon.com/fr/elasticache/what-is-redis/>. [Accès le 08 10 2024].
- [3 Cryptoast, «Qu'est ce que Solidity ?,» Cryptoast, 13 9 2024. [En ligne]. Available:
8] <https://cryptoast.fr/quest-ce-que-solidity-definition-explication/>. [Accès le 9 10 2024].
- [3 Tailwind CSS, «Get started with Tailwind CSS,» [En ligne]. Available:
9] <https://tailwindcss.com/docs/installation>. [Accès le 08 10 2024].
- [4 Syloe, «Test de charge,» [En ligne]. Available: <https://www.syloe.com/glossaire/effectuer-un-test-de-charge/>. [Accès le 20 10 2024].
- [4 Locust, «What is Locust?,» [En ligne]. Available: <https://docs.locust.io/en/stable/what-is-locust.html>. [Accès le 14 10 2024].
- [4 Kaspersky, «Qu'est-ce qu'une attaque DDoS ?,» [En ligne]. Available:
2] <https://www.kaspersky.fr/resource-center/threats/ddos-attacks>. [Accès le 20 10 2024].
- [4 Kaspersky, «Attaque par force brute,» [En ligne]. Available:
3] <https://www.kaspersky.fr/resource-center/definitions/brute-force-attack>. [Accès le 20 10 2024].

- [4 JUnit.org. [En ligne]. Available: <http://www.junit.org>. [Accès le 2 Janvier 2019].
4]
- [4 Coinbase, «Qu'est-ce qu'un contrat intelligent ?», [En ligne]. Available:
5] <https://www.coinbase.com/fr/learn/crypto-basics/what-is-a-smart-contract>. [Accès le 27
juin 2024].
- [4 academy youngplatform, «La signification de la blockchain,» academy youngplatform, 22
6] mai 2023. [En ligne]. Available:
[https://academy.youngplatform.com/fr/blockchain/blockchain-qu-est-signification-
definition-fonctionnement/](https://academy.youngplatform.com/fr/blockchain/blockchain-qu-est-signification-definition-fonctionnement/). [Accès le 1 juillet 2024].
- [4 linksconsulting, «Blockchain : Ce Qu'il Faut Comprendre,» linksconsulting, [En ligne].
7] Available: <http://www.linksconsulting.fr/la-blockchain-ce-quil-faut-comprendre/>. [Accès
le 1 juillet 2024].
- [4 M. PIGNEL, «LA TECHNOLOGIE BLOCKCHAIN : Une opportunité pour l'économie
8] sociale ?,» POUR LA SOLIDARITÉ, 2019.
- [4 IDH, 19 08 2024. [En ligne]. Available: [https://idh-grenoble.fr/2024/08/19/comprendre-le-
vote-un-acte-citoyen-fondamental/](https://idh-grenoble.fr/2024/08/19/comprendre-le-
9] vote-un-acte-citoyen-fondamental/). [Accès le 18 09 2024].
- [5 B. Infos, «Qu'est-ce qu'une chaîne de blocs (blockchain) ?», 12 avril 2022. [En ligne].
0] Available: [https://www.economie.gouv.fr/entreprises/blockchain-definition-avantage-
utilisation-application](https://www.economie.gouv.fr/entreprises/blockchain-definition-avantage-
utilisation-application). [Accès le 1 juillet 2024].
- [5 V. publique, «Quels sont les différents modes de scrutin ?», 19 9 2022. [En ligne].
1] Available: [https://www.vie-publique.fr/fiches/23948-quels-sont-les-differents-modes-de-
scrutin](https://www.vie-publique.fr/fiches/23948-quels-sont-les-differents-modes-de-
scrutin). [Accès le 3 10 2024].
- [5 N. Verliac, «Quel est la différence entre informations et données ?», ostraca, 2021 fevrier
2] 15. [En ligne]. Available: [https://blog.ostraca.fr/blog/difference-entre-informations-et-
donnees/](https://blog.ostraca.fr/blog/difference-entre-informations-et-
donnees/). [Accès le 2024 juin 27].
- [5 Larousse, «Larousse,» [En ligne]. Available:
3] <https://www.larousse.fr/dictionnaires/francais/vote/82548>. [Accès le 23 Juin 2024].

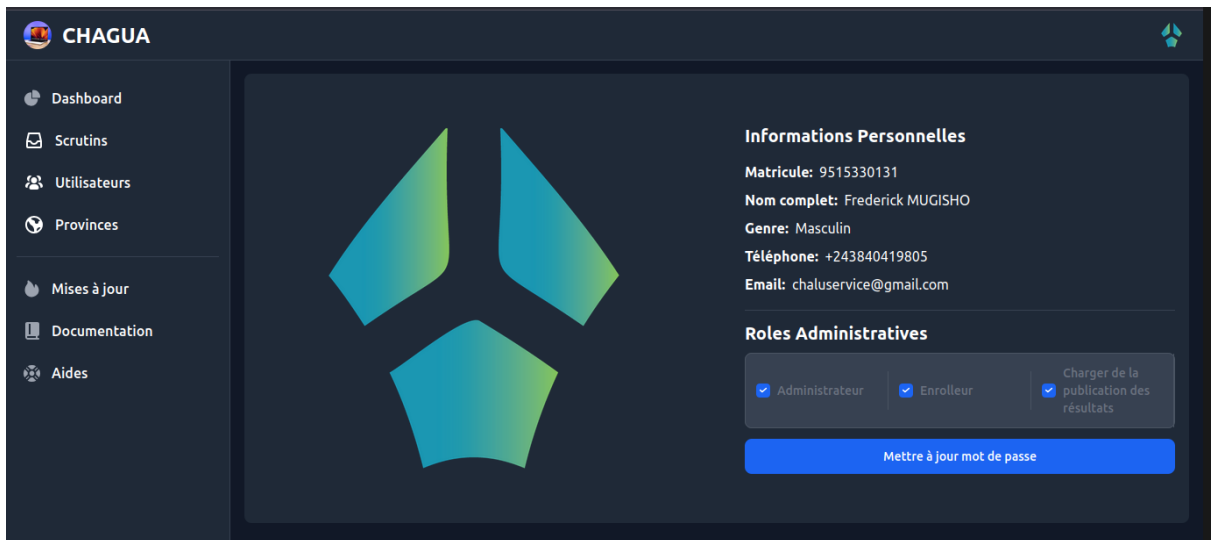
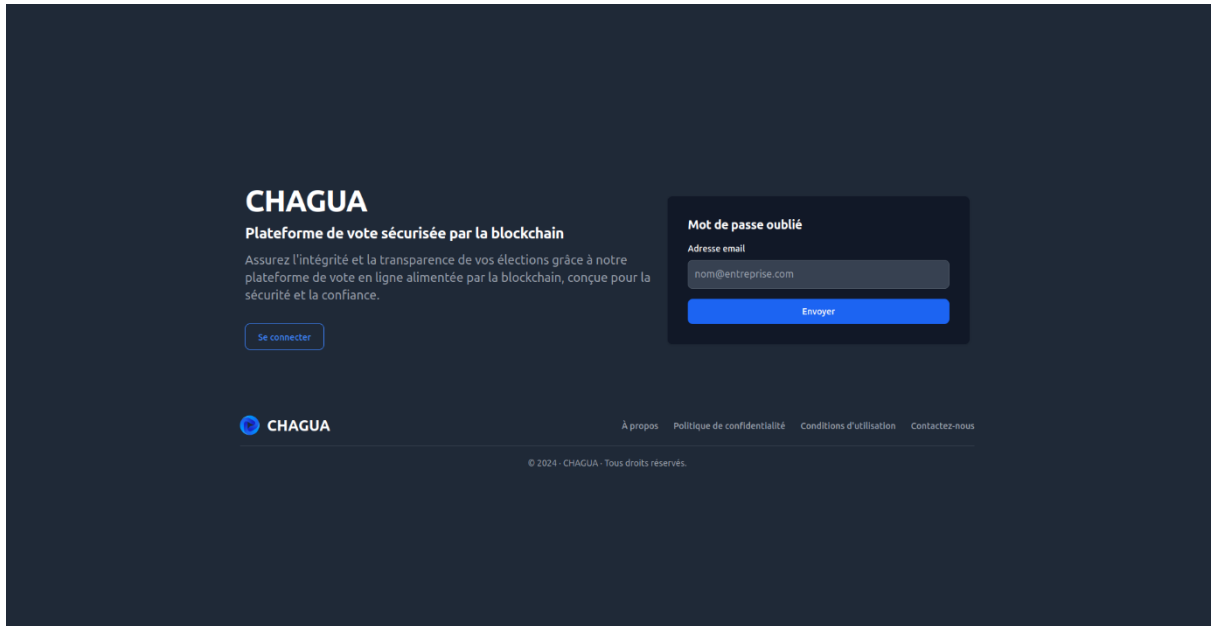
- [5 M. SARMIENTO, «LA TECHNOLOGIE BLOCKCHAIN : UNE OPPORTUNITÉ
4] POUR L'ATTEINTE,» UNIVERSITÉ DU QUÉBEC À MONTRÉAL, MONTRÉAL,
2021.
- [5 M. Toumirt, «En quoi consiste un système d'information ?», PayFit, 2021 avril 21. [En
5] ligne]. Available: <https://payfit.com/fr/fiches-pratiques/systeme-information/>. [Accès le
2024 juin 27].
- [5 N. B. Jean-bosco, «ELECTIONS ET PARTICIPATION POLITIQUE EN REPUBLIQUE
6] DEMOCRATIQUE DU CONGO. Regard sur la ville de Lubumbashi,» 2018. [En ligne].
Available: <https://www.memoireonline.com/12/21/12581/Elections-et-participation-politique-en-rpublique-dmocratique-du-Congo-Cas-des-lections.html>. [Accès le 2 juillet
2024].
- [5 IBM, «Application du chiffrement homomorphe pour la sécurité et la confidentialité,»
7] IBM, 21 06 2024. [En ligne]. Available:
<https://www.ibm.com/docs/fr/watsonx/saas?topic=learning-applying-encryption>. [Accès
le 10 10 2024].

ANNEXES

Quelques autres interfaces de notre système

The screenshot shows the 'Nouvel utilisateur' (New user) form in the CHAGUA system. The form is located in the main content area, with a sidebar on the left containing navigation links: Dashboard, Scrutins, Utilisateurs (selected), Provinces, Mises à jour, Documentation, and Aides. The form fields are: 'Nom complet' (text input), 'Sexe' (dropdown menu with 'Masculin' selected), 'Email' (text input), 'Téléphone' (text input), and 'Photo de profil' (file upload area with a 'Parcourir...' button and the text 'Aucun fichier sélectionné'). At the bottom of the form, there are two buttons: 'Annuler' (red) and 'Valider' (blue).

The screenshot shows the 'Dashboard scrutin' for 'Vote d'une nouvelle loi' in the CHAGUA system. The dashboard is located in the main content area, with a sidebar on the left containing navigation links: Dashboard, Scrutins (selected), Utilisateurs, Provinces, Mises à jour, Documentation, and Aides. The dashboard title is 'Dashboard scrutin "Vote d'une nouvelle loi"'. Below the title, there are four tabs: 'Général' (selected), 'Proposition', 'Electeurs', and 'Votes'. The main content area displays the following information: 'Code: 229996', 'Intitulé: Vote d'une nouvelle loi', 'Catégorie: Vote Binaire', 'Date et heure de début: le 9 septembre 2024 à 13:00', 'Date et heure de fin: le 12 septembre 2024 à 18:00', 'Statut: Terminé', and 'Hash: 7dd4a2a750e252d4b3e486896b63979782508f64b2ab8556b3639017cdf26'. At the bottom right of the dashboard, there is a blue button labeled 'Publier les resultats'.



Détails du scrutin "Elections présidentielle, législatives, provinciales, ..."

● Général 👤 Candidats 🗳️ Votes 📊 Résultats

[Voter](#)

Aucune données trouvées pour affichage

À propos Politique de confidentialité Conditions d'utilisation Contactez-nous

© 2024 - CHAGUA - Tous droits réservés.

- 🏠 Dashboard
- 📄 Scrutins
- 👤 Utilisateurs
- 🌐 Provinces
- 🔔 Mises à jour
- 📖 Documentation
- 🛠️ Aides

Nom: CENI RDC

Description: It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for 'lorem ipsum' will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).

Date de création: le 14 juillet 2024 03:27

Date de modification: le 15 septembre 2024 21:39

[Mettre à jour](#)

- 🏠 Dashboard
- 📄 Scrutins
- 👤 Utilisateurs
- 🌐 Provinces
- 🔔 Mises à jour
- 📖 Documentation
- 🛠️ Aides

Dashboard scrutin "Vote d'une nouvelle loi"

● Général 👤 Proposition 👤 Electeurs 🗳️ Votes

Proposition: Loi N°17/024 portant sur les droits de l'enfant

Votes Pour: 1 soit 50,0%

Votes Contre: 0 soit 0,0%

Votes Abstension: 1 soit 50,0%

Transaction Hash : 46d8ebd9b7594ed6083ae689f69dfd0ff66cdfaaff3c2e6a55ffaa338a6ece9f

CHAGUA

- Dashboard
- Scrutins
- Utilisateurs
- Provinces**
- Mises à jour
- Documentation
- Aides

Les provinces de la RDC

- Bas-Uélé (7 circonscriptions)
 - BU001 | Aketi
 - BU002 | Ango
 - BU003 | Bambesa
 - BU004 | Bondo
 - BU005 | Buta
 - BU006 | Poko
 - BU007 | Ville de Buta
- Equateur (8 circonscriptions)

CHAGUA

- Dashboard
- Scrutins**
- Utilisateurs
- Provinces
- Mises à jour
- Documentation
- Aides

Dashboard scrutin "Elections présidentielle, législatives, provinciales, ..."

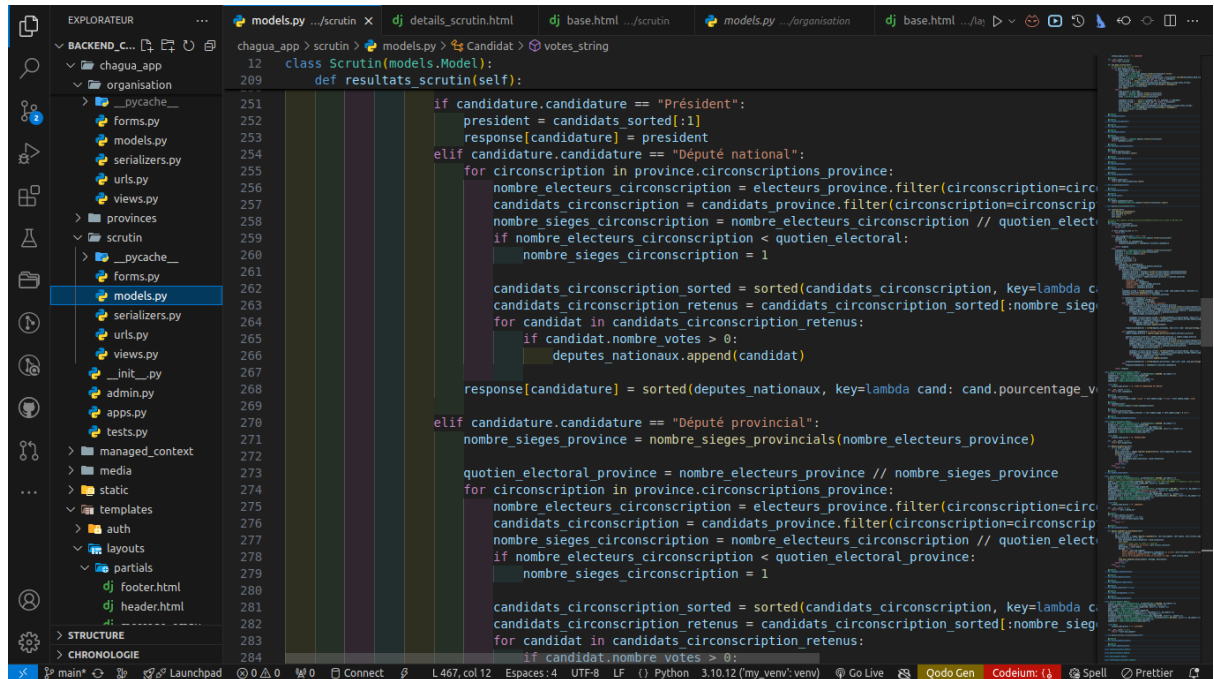
[Général](#)
[Types de candidature](#)
[Candidats](#)
[Electeurs](#)
[Votes](#)
[Résultats](#)

102925

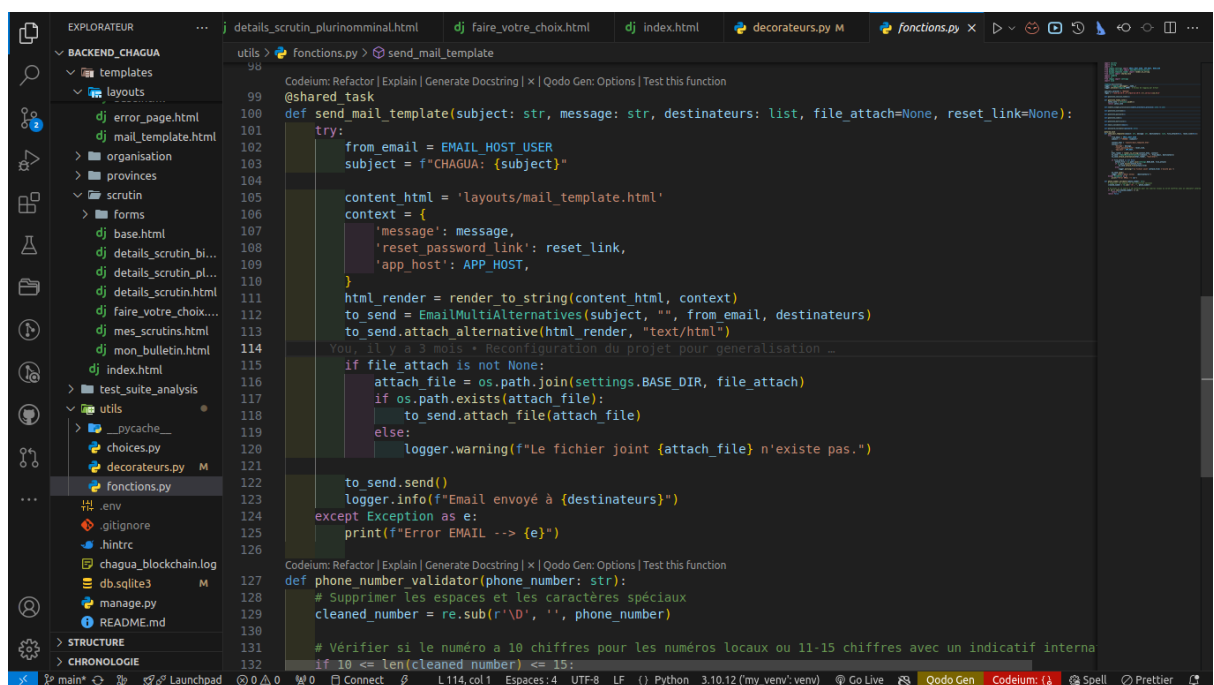
Les resultats ne sont pas disponible pour l'instant

Les resultats seront disponibles lors que le scrutin sera terminé

Quelques parties du code



```
class Scrutin(models.Model):
    def results_scrutin(self):
        if candidature.candidature == "Président":
            president = candidats_sorted[:1]
            response[candidature] = president
        elif candidature.candidature == "Député national":
            for circonscription in province.circonscriptions_province:
                nombre_electeurs_circonscription = electeurs_province.filter(circonscription=circonscription)
                candidats_circonscription = candidats_province.filter(circonscription=circonscription)
                nombre_sieges_circonscription = nombre_electeurs_circonscription // quotient_electoral
                if nombre_electeurs_circonscription < quotient_electoral:
                    nombre_sieges_circonscription = 1
            candidats_circonscription_sorted = sorted(candidats_circonscription, key=lambda c: c.nombre_sieges_circonscription)
            candidats_circonscription_retenus = candidats_circonscription_sorted[:nombre_sieges_circonscription]
            for candidat in candidats_circonscription_retenus:
                if candidat.nombre_votes > 0:
                    deputes_nationaux.append(candidat)
            response[candidature] = sorted(deputes_nationaux, key=lambda cand: cand.pourcentage_votes)
        elif candidature.candidature == "Député provincial":
            nombre_sieges_province = nombre_sieges_province(nombre_electeurs_province)
            quotient_electoral_province = nombre_electeurs_province // nombre_sieges_province
            for circonscription in province.circonscriptions_province:
                nombre_electeurs_circonscription = electeurs_province.filter(circonscription=circonscription)
                candidats_circonscription = candidats_province.filter(circonscription=circonscription)
                nombre_sieges_circonscription = nombre_electeurs_circonscription // quotient_electoral_province
                if nombre_electeurs_circonscription < quotient_electoral_province:
                    nombre_sieges_circonscription = 1
            candidats_circonscription_sorted = sorted(candidats_circonscription, key=lambda c: c.nombre_sieges_circonscription)
            candidats_circonscription_retenus = candidats_circonscription_sorted[:nombre_sieges_circonscription]
            for candidat in candidats_circonscription_retenus:
                if candidat.nombre_votes > 0:
```



```
@shared_task
def send_mail_template(subject: str, message: str, destinataires: list, file_attach=None, reset_link=None):
    try:
        from_email = EMAIL_HOST_USER
        subject = f"CHAGUA: {subject}"
        content_html = 'layouts/mail_template.html'
        context = {
            'message': message,
            'reset password link': reset_link,
            'app_host': APP_HOST,
        }
        html_render = render_to_string(content_html, context)
        to_send = EmailMultiAlternatives(subject, "", from_email, destinataires)
        to_send.attach_alternative(html_render, "text/html")
        # You, si y a 2 mois + Reconfiguration du projet pour generalisation ...
        if file_attach is not None:
            attach_file = os.path.join(settings.BASE_DIR, file_attach)
            if os.path.exists(attach_file):
                to_send.attach_file(attach_file)
            else:
                logger.warning(f"Le fichier joint {attach_file} n'existe pas.")
        to_send.send()
        logger.info(f"Email envoyé à {destinataires}")
    except Exception as e:
        print(f"Error EMAIL -> {e}")

def phone_number_validator(phone_number: str):
    # Supprimer les espaces et les caractères spéciaux
    cleaned_number = re.sub(r'\D+', '', phone_number)
    # Vérifier si le numéro a 10 chiffres pour les numéros locaux ou 11-15 chiffres avec un indicatif interna
    if 10 <= len(cleaned_number) <= 15:
```

```

EXPLOREUR  ... tml .../layouts  dj details_scrutin_plurinominal.html  dj faire_votre_choix.html  dj index.html  decorateurs.py M X
BACKEND_CHAGUA  templates  layouts  dj_error_page.html  dj_mail_template.html  organisation  provinces  scrutin  Forms  dj_base.html  dj_details_scrutin_bi...  dj_details_scrutin_pl...  dj_faire_votre_choix...  dj_mes_scrutins.html  dj_mon_bullein.html  dj_index.html  test_suite_analysis  utils  _pycache_  choices.py  decorateurs.py M  fonctions.py  env  gitignore  .hintrc  chagua_blockchain.log  db.sqlite3 M  manage.py  README.md  STRUCTURE  CHRONOLOGIE
utils > decorateurs.py > ...
14 logging.basicConfig(level=logging.INFO,
15                     format='%(asctime)s - %(levelname)s - %(message)s')
16 )
17 logger = logging.getLogger(__name__)
18 logger.setLevel(logging.INFO) # Niveau de logging par défaut
19
Codeium: Refactor | Explain | Generate Docstring | X | Qodo Gen: Options | Test this function
20 def admin_orga_required(view_func, redirect_url="account:user_redirector"):
Codeium: Refactor | Explain | Generate Docstring | X
21 @functools.wraps(view_func)
22 def is_membre_orga(request, *args, **kwargs):
23     try:
24         user_account = AdminOrga.objects.filter(user_account=request.user, validation=True)
25         if user_account.exists():
26             return view_func(request, *args, **kwargs)
27
28         messages.error(request, "Vous n'etes pas autorisé à accéder à l'interface demandée 5455")
29         return redirect(redirect_url)
30     except Exception as e:
31         messages.error(request, f"Vous n'etes pas autorisé à accéder à l'interface demandée {e}")
32         return redirect(redirect_url)
33     return is_membre_orga
34
Codeium: Refactor | Explain | Generate Docstring | X | Qodo Gen: Options | Test this function
35 def super_admin_required(view_func, redirect_url="account:user_redirector"):
Codeium: Refactor | Explain | Generate Docstring | X
36 @functools.wraps(view_func)
37 def is_super_admin(request, *args, **kwargs):
38     try:
39         user_account = AdminOrga.objects.filter(user_account=request.user, validation=True)
40         if user_account.exists():
41             user = user_account.first()
42             if "1" in user.roles:
43                 return view_func(request, *args, **kwargs)
44
45         messages.error(request, "Vous n'etes pas autorisé à accéder à l'interface demandée")
46         return redirect(redirect_url)
47     except Exception as e:
48         messages.error(request, f"Vous n'etes pas autorisé à accéder à l'interface demandée {e}")
49         return redirect(redirect_url)
50     return is_super_admin
51
main*  Launchpad  0 0 0  Connect  L 18, col 62  Espaces: 4  UTF-8  LF  Python 3.10.12 (my_venv:venv)  Go Live  Qodo Gen  Codeium: (...)  Spell  Prettier

```

```

EXPLOREUR  ... .py .../scrutin  chagua_blockchain.log  views.py .../scrutin  models.py  settings.py  dj_details_scrutin.html x  dj b
BACKEND_C...  templates > scrutin > dj_details_scrutin.html
You, hier | 1 author (You)
1 {% extends 'layouts/base.html' %}
2 {% load static %}
3
4 {% block title %}Dashboard scrutin ({{ scrutin.intitile }}){% endblock title %}
5
6 {% block content %}
7
8 <section class="bg-gray-50 dark:bg-gray-900 py-3 sm:py-5 lg:px-12">
9 <section class="bg-gray-50 py-8 antialiased dark:bg-gray-900 md:py-12">
10 <div class="mx-auto max-w-screen-xl px-4 2xl:px-0">
11 <!-- Heading & Filters -->
12 <div class="mb-4 items-end justify-between space-y-4 sm:flex sm:space-y-0 md:mb-8">
13 <div>
14 <h2 class="mt-3 text-xl font-sembold text-gray-900 dark:text-white sm:text-2xl">Dash
15 </div>
16 <div class="mb-4 border-b border-gray-200 dark:border-gray-700">
17 <ul class="flex flex-wrap -mb-px text-sm font-medium text-center" id="default-styled-tab"
18 </ul>
19 </div>
20 <div id="default-styled-tab-content">
21 <div class="hidden p-4 rounded-lg bg-gray-50 dark:bg-gray-800 id="styled-general" role="
22 </div>
23 <div class="hidden p-4 rounded-lg bg-gray-50 dark:bg-gray-800 id="styled-candidats" role
24 </div>
25 <div class="hidden p-4 rounded-lg bg-gray-50 dark:bg-gray-800 id="styled-electeurs" role
26 </div>
27 <div class="hidden p-4 rounded-lg bg-gray-50 dark:bg-gray-800 id="styled-votes" role="ta
28 </div>
29 <div class="hidden p-4 rounded-lg bg-gray-50 dark:bg-gray-800 id="styled-resultats"
30 </div>
31 <div class="md:flex justify-center items-center">
32 <section class="bg-white dark:bg-gray-900 flex rounded-lg">
33
main*  Launchpad  0 0 0  Connect  L 613, Col 142 (16 sélectionné)  Espaces: 4  UTF-8  LF  Django Template  Go Live  Qodo Gen  Codeium: (...)  Spell  Prettier

```

```

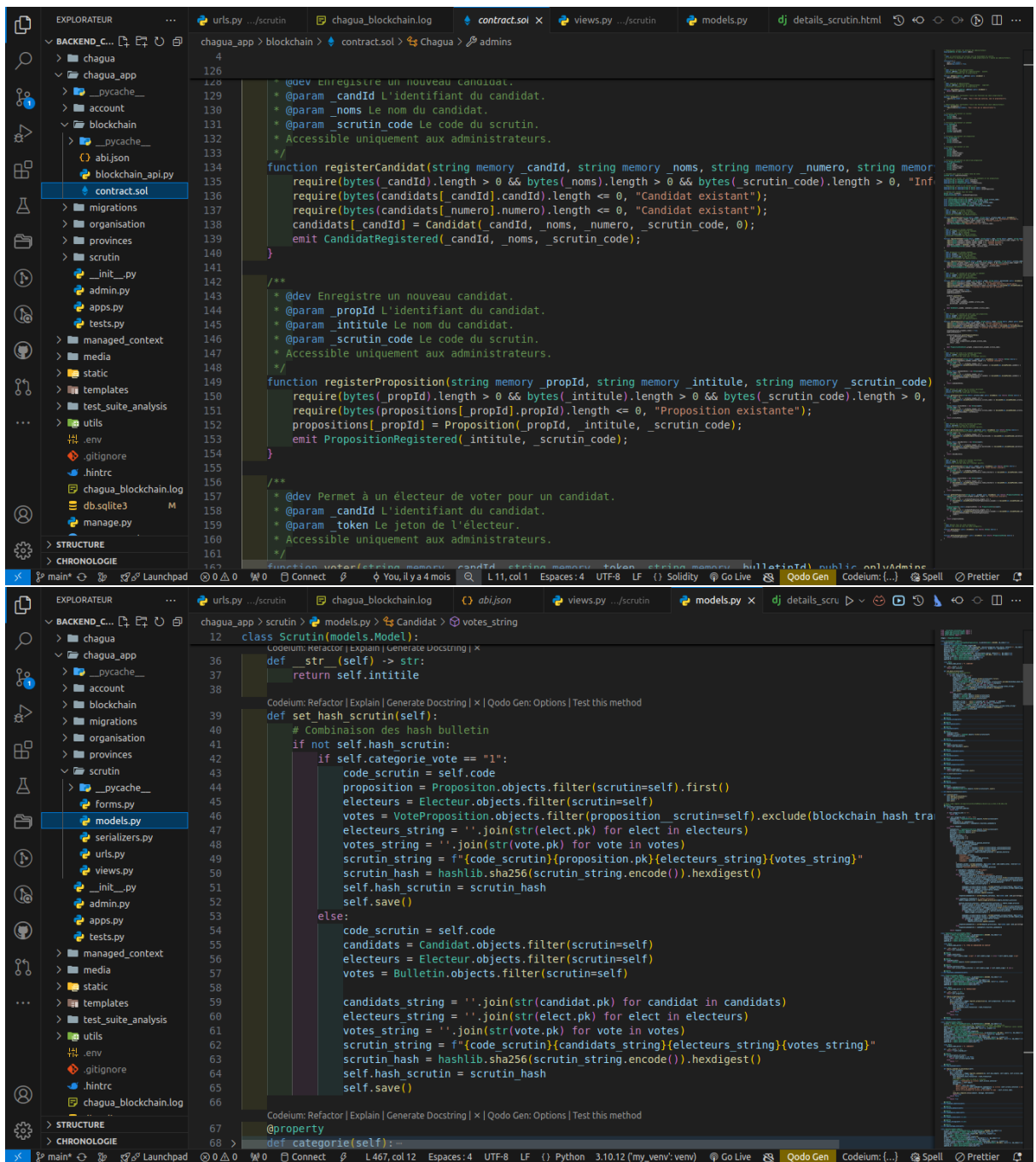
chagua_app > scrutin > urls.py > ...
You, hier | 1 author (You)
1 from django.urls import path
2 from . import views
3
4 app name = 'scrutin'
5 urlpatterns = []
6
7 path('', views.mes_scrutins, name="mes_scrutins"),
8 path('details_scrutin/<str:code>/', views.details_scrutin, name="details_scrutin"),
9 path('nouveau_scrutin/', views.nouveau_scrutin, name="nouveau_scrutin"),
10 path('modifier_scrutin/<str:code>/', views.modifier_scrutin, name="modifier_scrutin"),
11 path('nouvelle_proposition/<str:code>/', views.nouvelle_proposition, name="nouvelle proposition"),
12 path('modifier_proposition/<int:id>/', views.modifier_proposition, name="modifier proposition"),
13 path('modele_fichier_electeur/<str:code>/', views.modele_fichier_electeur, name="modele_fichier_electeur"),
14 path('charger_electeurs_scrutin/<str:code>/', views.charger_electeurs_scrutin, name="charger_electeurs_sc"),
15 path('delete_electeur_scrutin/<int:id>/', views.delete_electeur_scrutin, name="delete_electeur_scrutin"),
16 path('recherche_scrutin/', views.recherche_scrutin, name="recherche_scrutin"),
17 path('details_scrutin_binaire/<str:code>/', views.details_scrutin_binaire, name="details_scrutin_binaire"),
18 path('details_scrutin_plurinominal/<str:code>/', views.details_scrutin_plurinominal, name="details_scr"),
19 path('lancer_scrutin/<str:code>/', views.lancer_scrutin, name="lancer_scrutin"),
20 path('cloturer_scrutin/<str:code>/', views.cloturer_scrutin, name="cloturer_scrutin"),
21 path('publier_resultats_scrutin/<str:code>/', views.publier_resultats_scrutin, name="publier_resultats_sc"),
22 path('add_type_candidature/<str:code>/', views.add_type_candidature, name="add_type_candidature"),
23 path('modifier_type_candidature/<int:id>/', views.modifier_type_candidature, name="modifier_type_candida"),
24 path('add_candidat/<str:code>/', views.add_candidat, name="add_candidat"),
25 path('charger_candidats_scrutin/<str:code>/', views.charger_candidats_scrutin, name="charger_candidats_sc"),
26 path('modifier_candidat/<int:id>/', views.modifier_candidat, name="modifier_candidat"),
27 path('supprimer_candidat/<int:id>/', views.supprimer_candidat, name="supprimer_candidat"),
28 path('faite_votre_choix/<str:code>/', views.faites_votre_choix, name="faites_votre_choix"),
29 path('faite_votre_choix/<int:candidature_id>/', views.faites_votre_choix, name="faites_votre_choix"),
30 path('ajouter_candidat/<int:id>/', views.ajouter_candidat, name="ajouter_candidat"),
31 path('mon_bulletin/<str:code>/', views.mon_bulletin, name="mon_bulletin"),

```

```

chagua > settings.py > ...
153 STATIC_URL = 'static/'
154 STATIC_ROOT = os.path.join(BASE_DIR, 'staticfiles')
155 STATICFILES_DIRS = [
156     BASE_DIR / 'static',
157 ]
158
159 LOGIN_REDIRECT_URL = 'account:user_redirector'
160 LOGIN_URL = 'account:login'
161
162 MEDIA_URL = '/media/'
163 MEDIA_ROOT = BASE_DIR / 'media'
164
165 EMAIL_HOST = env('EMAIL_HOST')
166 EMAIL_HOST_USER = env('EMAIL_HOST_USER')
167 EMAIL_HOST_PASSWORD = env('EMAIL_HOST_PASSWORD')
168 EMAIL_PORT = env('EMAIL_PORT')
169 EMAIL_USE_TLS = env('EMAIL_USE_TLS')
170
171 ADDRESS_ETHEREUM = env('ADDRESS_ETHEREUM')
172 SECRET_KEY_BC = env('SECRET_KEY_BC')
173
174 # ----- PARAMETRES DE SECURITE DJANGO -----#
175 if not DEVELOPEMENT:
176     SESSION_COOKIE_SECURE = True
177     SECURE_SSL_REDIRECT = True
178     SECURE_HSTS_SECONDS = 15768000
179     SECURE_HSTS_INCLUDE_SUBDOMAINS = True
180     SECURE_HSTS_PRELOAD = True
181     SECURE_CONTENT_TYPE_NOSNIFF = True
182     SECURE_BROWSER_XSS_FILTER = True
183     SESSION_COOKIE_HTTPONLY = True
184
185 DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'
186
187 CELERY_BROKER_URL = 'redis://localhost:6379/0' # Utilisation de Redis
188 CELERY_ACCEPT_CONTENT = ['json']

```



Autres figures

Ganache

ACCOUNTS BLOCKS TRANSACTIONS CONTRACTS EVENTS LOGS

SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK: 57 GAS PRICE: 20000000000 GAS LIMIT: 6721975 HARDHORN MERGE NETWORK ID: 5777 RPC URLS: HTTP://127.0.0.1:7545 MINIMALISTIC AUTOMINING WORKSPACE: CHAGUA SWITCH

TX HASH	FROM ADDRESS	TO CONTRACT ADDRESS	GAS USED	VALUE
0x382e845c91208475797cd040cc502063cce351343ac820e6b2d3fef8bdb74b42	0x528ae1980947935fa312284896c143a21d73c1dc	0x064eDd0934103e37254A0b30e2044b4D2A0bc02	259227	0
0xdb1fce4320fdc6773c320835f645e3efb96df08d659ba7fbc44f11483f0af24	0x528ae1980947935fa312284896c143a21d73c1dc	0x064eDd0934103e37254A0b30e2044b4D2A0bc02	120794	0
0xe3464249fec8a2d1bd4d4990899ebd618a850a938ab6d2fbaffa508e1476bd	0x528ae1980947935fa312284896c143a21d73c1dc	0x064eDd0934103e37254A0b30e2044b4D2A0bc02	120794	0
0x3e50ad2c56a8d06f6c0cd49bf82792d2fee26459829c5565ab40e274a12a57d	0x528ae1980947935fa312284896c143a21d73c1dc	0x064eDd0934103e37254A0b30e2044b4D2A0bc02	129783	0
0x3d3b1f8be638ad764e7b7647852bdc647ded3ccd679724481e85e9f9fd86cd26	0x528ae1980947935fa312284896c143a21d73c1dc	0x064eDd0934103e37254A0b30e2044b4D2A0bc02	129783	0
0x88325ba50a9e6f8dcd2780be13185fd56df864625ebced033eaba9a9a791a11	0x528ae1980947935fa312284896c143a21d73c1dc	0x064eDd0934103e37254A0b30e2044b4D2A0bc02	242115	0
0x78f7c6a5b6b3a09b97e215e6bbd7c7285fa361bb3b01cfa8e2a5b08517936727				

Ganache

WORKSPACES

- chagua (ethereum)
- diplome-certification (ethereum)

QUICKSTART ETHereum

NEW WORKSPACE ETHereum

```
Activités Terminal 9 oct. 11:13 en 100%
frederick@frederick: ~/Projets/blockchain/voting_blockchain/backend_chagua

87b53c30ba86b2452396efa1496d6986994e17adc728afb634efd100f35394a7
[2024-10-09 09:13:12,489: INFO/MainProcess] Task utils.fonctions.send_mail_template[ee96f7b5-2d54-4e9f-8e6d-fba19ad247ba] received
61a5bb9452f911a735402890fa4b08a5356662c1a8ed33017339a0d1207921d8
61a5bb9452f911a735402890fa4b08a5356662c1a8ed33017339a0d1207921d8
[2024-10-09 09:13:12,761: INFO/MainProcess] Task utils.fonctions.send_mail_template[2a22da05-57b7-43cf-b802-2dcbee98142e] received
65a6f566c56c9e66a4b6545447e571f2d9463f19a531f461329e864259c4447
65a6f566c56c9e66a4b6545447e571f2d9463f19a531f461329e864259c4447
[2024-10-09 09:13:12,994: INFO/MainProcess] Task utils.fonctions.send_mail_template[5bd4bf3d-aed7-45da-aed0-81809b5da9f4] received
9486ddccb90c1546683c2e79b3f4f0b1a7635847fb39dd9f3b5924f5933ccd6
9486ddccb90c1546683c2e79b3f4f0b1a7635847fb39dd9f3b5924f5933ccd6
[2024-10-09 09:13:13,212: INFO/MainProcess] Task utils.fonctions.send_mail_template[3909c8e2-a31b-4444-abee-9fb0b34fc59a] received
a867bb5f5ac4fbd5ed7f0bfd32f80f1c080441105f22bf2359a6da93c8d9376a
a867bb5f5ac4fbd5ed7f0bfd32f80f1c080441105f22bf2359a6da93c8d9376a
[2024-10-09 09:13:13,450: INFO/MainProcess] Task utils.fonctions.send_mail_template[7f006e3d-546e-4355-a39c-0e7266ca6d95] received
e5c90c04d9ab8670f291ed7162e941dbac261ee537b4db518dd05e5a507f9f40
e5c90c04d9ab8670f291ed7162e941dbac261ee537b4db518dd05e5a507f9f40
[2024-10-09 09:13:13,698: INFO/MainProcess] Task utils.fonctions.send_mail_template[96288a3e-1b59-459b-b1fe-8359bb3bc16c] received
cf39c5823f10ce78573c764d18247aed434e8c78e825510b08efd11b95622b4c
cf39c5823f10ce78573c764d18247aed434e8c78e825510b08efd11b95622b4c
[2024-10-09 09:13:13,961: INFO/MainProcess] Task utils.fonctions.send_mail_template[2924b708-5ab7-495f-b25f-3b5609d18f36] received
d9853ea42ef331e432f2ab3d4f64fdac70529311adfcff65e8de1c6c7d7092e3
d9853ea42ef331e432f2ab3d4f64fdac70529311adfcff65e8de1c6c7d7092e3
[2024-10-09 09:13:14,212: INFO/MainProcess] Task utils.fonctions.send_mail_template[e909421b-a3a0-4a32-865b-523017247a81] received
8a8dae7338a4157fd8a410260ae5e0c63cad5f4f3bb28d88bb17baca3b09a17a
8a8dae7338a4157fd8a410260ae5e0c63cad5f4f3bb28d88bb17baca3b09a17a
[2024-10-09 09:13:14,463: INFO/MainProcess] Task utils.fonctions.send_mail_template[15b50980-0359-42f7-8e0f-876ebbcfa89e] received
562b33f3ba109c82451ad7a2850cca26e16cb9834e9918e3f2f2a207625d9d16
562b33f3ba109c82451ad7a2850cca26e16cb9834e9918e3f2f2a207625d9d16
[2024-10-09 09:13:14,697: INFO/MainProcess] Task utils.fonctions.send_mail_template[93e2b742-a9ca-44c0-9d2d-537923e042a4] received
cce1500f2976a58a011bfb3ec4e6450a29b88584c1507872245bed813b3c29b3
cce1500f2976a58a011bfb3ec4e6450a29b88584c1507872245bed813b3c29b3
[2024-10-09 09:13:14,940: INFO/MainProcess] Task utils.fonctions.send_mail_template[97f0cd2d-d623-4356-b3dd-aa8c9b40d7db] received
820cfc68fde93527c0e3c4077577352af9058bf30793f6e6e5336bd3ec33e745
820cfc68fde93527c0e3c4077577352af9058bf30793f6e6e5336bd3ec33e745
[2024-10-09 09:13:15,168: INFO/MainProcess] Task utils.fonctions.send_mail_template[0a36f084-bef7-45c7-ba71-0dcf103de37a] received
0f069b12866a2171ecb9811063a9da85934cc6fea6c50b7372dad89a35e7a016
```

