République Algérienne Démocratique et Populaire Université Abou Bakr Belkaid– Tlemcen

Faculté des Sciences Département d'Informatique

Mémoire de fin d'études pour l'obtention du diplôme de Master en Informatique

Option: Gestion logiciel

Thème

Block-Chain pour Vote Electronique

Réalisé par :

- Salah mohamed habib

Présenté le 21/ 06/2025

devant le jury composé de :

- BENTAALLAH AMINE (Examinateur)
- REMACI ZINEB YASMINA (Présidente)
- SELADJI YASSAMINE (Expert De I2E)
- Belhocine Amine (Encadreur)
- LachgueuerMohammed Abdesalem (Co-encadreur)

Année universitaire : 2025-2026

Remerciements

Au terme de ce mémoire, je remercie avant tout Dieu Tout-Puissant, pour la force, la patience et la bénédiction qu'Il m'a accordée durant toutes les étapes de ce travail.

je souhaite exprimer ma profonde reconnaissance à mes parents, qui ont toujours été un pilier essentiel dans mon parcours. Leur soutien moral, leurs sacrifices, leurs prières et leur encouragements constant ont constitué une source inestimable de motivation. Sans leur amour et leur confiance, rien de tout cela n'aurait été possible.

mes remerciements les plus sincères vont à mon encadrant, Monsieur Belhocine Amine, pour son accompagnement , sa disponibilité permanente et la qualité de ses conseils. Son exigence scientifique, sa rigueur et son soutien continu ont joué un rôle déterminant dans l'aboutissement de ce travail.

je tiens également à exprimer ma gratitude à mon co-encadrant, Monsieur Islem Lachgeur , pour sont précieux suivi, orientations pertinentes, aide constante et remarques constructives. implication et bienveillance ont grandement enrichi ce mémoire et facilité son aboutissement.

j'adresse aussi des remerciements aux membres du jury, qui me font l'honneur d'évaluer ce travail et d'y consacrer une partie de leur temps.

mes pensées reconnaissantes vont également au chef du département ainsi qu'à l'ensemble des enseignants du département d'informatique. Par leurs enseignements, leurs valeurs et leur engagement, ils ont contribué à ma formation et à l'acquisition de mes compétences.

Enfin, je remercie chaleureusement mes proches, amis et toutes les personnes qui mon encouragés, soutenus ou aidés de près comme de loin. Leur présence et leur confiance ont été un véritable moteur tout au long de cette expérience

Dédicaces

Je dédie ce modeste travail à toutes celles et ceux qui ont contribué, de près ou de loin, à mon parcours et à l'aboutissement de ce mémoire.

À mon père, dont la force tranquille et la présence rassurante ont guidé mes pas lorsque mes certitudes vacillaient. Tu as été ce pilier silencieux, toujours là, toujours solide, et pour cela, je te suis infiniment reconnaissant.

À ma mère, source inépuisable d'amour, de douceur et de courage, dont les sacrifices et la tendresse ont nourri chacun de mes progrès. Aucune page de ce mémoire ne saurait exprimer toute la gratitude que je te dois.

À ma grande famille, pour son soutien constant et pour les valeurs qui ont façonné mon chemin.

À mon encadrant, Monsieur Belhocine, pour la clarté de ses orientations, la richesse de ses connaissances, et l'accompagnement précieux qu'il m'a offert tout au long de ce travail.

À mes co-encadrants monsieur Lachgueur islem, pour son soutien constant, ca disponibilités exemplaire et ces précieux conseils .

À toutes celles et ceux qui m'ont transmis des valeurs, qui on façonné mon regard sur le monde, et qui ont contribué à faire de moi la personne que je suis.

Enfin, à mes camarades de promotion, et à toutes les personnes qui, par un geste, un mot ou une présence, ont laissé une trace bienveillante et durable sur mon parcours. Je vous dédie ces pages avec toute ma gratitude

TABLE DES MATIERES

TABLE DES FIGURES

LISTE DES TABLEAUX

TABLE DES ABRÉVIATIONS

Abréviation	Signification
ABI	Application Binary Interface
API	Application Programming Interface
BFT	Byzantine Fault Tolerance
BMC	Business Model Canvas
CORS	Cross-Origin Resource Sharing
DRE	Direct Recording Electronic
DZD	Dinar algérien
ETH	Ether, cryptomonnaie native d'Ethereum
EVM	Ethereum Virtual Machine
HTTP	HyperText Transfer Protocol
ID	Identifiant
JWT	JSON Web Token
PoA	Proof of Authority
PoS	Proof of Stake
PoW	Proof of Work
R1CS	Rank-1 Constraint System
RPC	Remote Procedure Call
SHA-256	Secure Hash Algorithm 256 bits
UI	User Interface
UML	Unified Modeling Language
WASM	WebAssembly
ZK	Zero-Knowledge
ZKP	Zero-Knowledge Proof

INTRODUCTION GENERALE

Le vote constitue un mécanisme essentiel de prise de décision collective. Il intervient dans les élections politiques, mais également dans les universités, les associations, les entreprises et les organisations qui doivent choisir des représentants, adopter une résolution ou départager plusieurs propositions. Quelle que soit l'échelle considérée, la légitimité du résultat dépend de la confiance accordée au processus elector1al.

Pendant une longue période, le bulletin papier et l'urne physique ont constitué les principaux instruments du scrutin. Cette organisation possède des qualités importantes, notamment la simplicité du principe et l'existence d'une trace matérielle. Elle exige toutefois une logistique conséquente : préparation des listes électorales, impression et distribution des bulletins, mobilisation des bureaux de vote, contrôle manuel de l'identité et dépouillement.

Le développement de l'informatique a conduit à l'apparition de solutions électroniques capables d'automatiser une partie de ces opérations. Le vote électronique peut réduire certains délais et faciliter l'accès au scrutin, mais il déplace aussi les problèmes de confiance vers le logiciel, les réseaux, les bases de données et les personnes chargées d'administrer le système. Une erreur ou une compromission informatique peut alors affecter un grand nombre de votes. [1], [2]

Les technologies de registre distribué proposent une autre approche. Au lieu de confier l'enregistrement final à une base unique, elles permettent à plusieurs nœuds de maintenir un historique commun selon des règles de consensus. Cette propriété n'élimine pas tous les risques électoraux, mais elle fournit des mécanismes utiles pour l'intégrité, la traçabilité et la vérification publique.[5], [6]

Cependant, les systèmes de vote électronique actuels manquent d'outils qui permettent à l'électeur de vérifier par lui même la prise en compte de son vote dans le résultat final ainsi que la véracité du résultat global, assurant le bon décompte de toutes les voies exprimées lors du scrutin. Les nouvelles techniques apportées par les nouvelles technologies de ces dernières années permettent de remédier à ces manquements et assurer ainsi la confiance totale des électeurs dans ce types de votes électroniques.

C’est dans cette perspective que s’inscrit ce mémoire, qui porte sur la conception et la réalisation d’un système de vote électronique sécurisé basé sur la blockchain et les preuves à divulgation nulle de connaissance. L’objectif principal est de proposer une solution permettant à un électeur autorisé de voter une seule fois, tout en préservant son anonymat et en assurant l’intégrité du résultat à travers un smart contract vérifiable.[6], [9], [10]

Pour mieux présenter ce travail, le manuscrit est organisé en plusieurs chapitres :

Le premier chapitre, intitulé « Généralités », présente l’évolution des systèmes de vote, depuis les formes traditionnelles jusqu’aux solutions électroniques. Il expose également les principales typologies du vote électronique, les exigences fondamentales d’un système de vote sécurisé, les limites des systèmes classiques, ainsi que les notions de consensus distribué appliquées au vote.

Le deuxième chapitre est consacré aux technologies utilisées dans le projet. Il présente les concepts liés à la blockchain, aux smart contracts, aux arbres de Merkle, aux fonctions de hachage cryptographiques et aux preuves à divulgation nulle de connaissance. Ce chapitre permet de comprendre les bases techniques nécessaires à la mise en œuvre du système proposé.

Le troisième chapitre présente la conception du système zk-voting. Il décrit l’architecture générale de la solution, les différents acteurs du système, le processus d’inscription des électeurs, la génération des engagements d’identité, la construction de l’arbre de Merkle, ainsi que le fonctionnement général du vote à l’aide d’une preuve cryptographique.

le quatrieme chapitre est consacré à la validation économique de ce projet, où les différents détails relatifs à cette partie son présentés, notamment les entrés et dépences financières, la tranche de clients visés dans le marché et autres

En somme, ce mémoire vise à montrer que la combinaison de la blockchain, des smart contracts et des preuves à divulgation nulle de connaissance peut contribuer à renforcer la confiance dans les systèmes de vote électronique. Le système proposé cherche à répondre aux principaux défis du vote numérique, notamment l’intégrité, la vérifiabilité, l’unicité du vote et la protection de l’identité des électeurs.

CHAPITRE I : GENERALITE

I.1. INTRODUCTION

Ce chapitre présente d'abord l'évolution historique des systèmes de vote, depuis les scrutins traditionnels jusqu'aux premières formes de vote électronique. Il expose ensuite les principales typologies de vote électronique, ainsi que les exigences fondamentales qu'un système de vote sécurisé doit respecter, comme l'intégrité, l'anonymat, le secret du vote, la vérifiabilité et la résistance à la coercition [1], [2], [13] . Enfin, il aborde les limites des systèmes électroniques classiques et introduit les notions de consensus distribué et de registre immuable, qui constituent une base importante pour les systèmes de vote modernes fondés sur la blockchain.

I.2. Historique des systèmes de vote

I.2.1. Du vote antique aux scrutins papier

Les premières formes connues de participation collective reposaient sur des mécanismes simples et directement observables : vote à main levée, dépôt d'objets dans un récipient ou expression orale du choix. Ces procédures étaient adaptées à des communautés de taille limitée, mais elles ne garantissaient pas toujours le secret du vote et pouvaient exposer les participants à la pression du groupe.

L'évolution des institutions politiques a progressivement imposé des exigences plus strictes. Le scrutin secret a été développé afin de dissocier l'identité de l'électeur de son choix. Le bulletin papier, l'isoloir et l'urne scellée ont constitué une réponse pratique à cette exigence. Le contrôle des listes électorales permet de vérifier l'éligibilité, tandis que les représentants des parties prenantes peuvent observer le dépouillement.

Le vote papier repose ainsi sur une chaîne de confiance composée de procédures, de personnes et de preuves matérielles. Sa robustesse ne vient pas d'un composant unique, mais de la combinaison de plusieurs contrôles : identification, surveillance du bureau, scellement de l'urne, procès-verbaux et recomptage possible. Cette organisation reste néanmoins coûteuse et lente lorsque le nombre d'électeurs ou la dispersion géographique augmente.[13]

Figure .I.1 : Exemple de scrutin papier

I.2.2. L'émergence du vote électronique (1970–2000)

À partir de la seconde moitié du XXe siècle, l'informatique a été introduite dans les opérations électorales. Les premiers usages concernaient principalement le comptage automatisé de bulletins et l'enregistrement direct des choix sur des machines spécialisées. L'objectif était de réduire le temps de dépouillement, d'éviter certaines erreurs de saisie et de faciliter l'organisation de scrutins complexes.

Dans les années 1990, la diffusion des réseaux informatiques et d'Internet a élargi le champ des possibilités. Le vote n'était plus nécessairement limité à une machine isolée : des systèmes de vote à distance, des kiosques connectés et des plateformes web ont commencé à être étudiés. Ces solutions promettaient une meilleure accessibilité pour les électeurs éloignés et une publication plus rapide des résultats.

Cette évolution a cependant transformé la nature des risques. Les problèmes matériels et organisationnels du vote papier ont été complétés par des risques logiciels : défaut de programmation, mauvaise configuration, compromission du système d'exploitation, attaque du réseau, altération d'une base de données ou utilisation abusive de privilèges administratifs.[3], [13]

I.2.3. Premières tentatives et controverses

Les premières expériences de vote électronique ont montré qu'un résultat rapide ne suffit pas à établir la confiance. Un système peut produire un total cohérent tout en demeurant impossible à auditer par les électeurs ou les observateurs. Cette difficulté est souvent résumée par l'expression « boîte noire » : le fonctionnement interne de la machine ou du logiciel n'est pas directement observable.[2], [13]

Une autre controverse concerne la concentration du pouvoir technique. Dans une architecture centralisée, l'administrateur du serveur, le fournisseur du logiciel ou l'opérateur de la base de données peut disposer de privilèges importants. Même en l'absence de fraude, la simple impossibilité de vérifier indépendamment les opérations peut fragiliser l'acceptation du résultat.

Ces limites ont favorisé la recherche de mécanismes vérifiables de bout en bout. L'idée centrale consiste à fournir des preuves permettant de contrôler les différentes étapes du scrutin sans révéler le contenu privé des votes. Les techniques cryptographiques modernes, notamment les engagements, les preuves d'appartenance et les preuves à divulgation nulle de connaissance, s'inscrivent dans cette démarche.

I.3. Typologie des systèmes de vote électronique

I.3.1. Vote en machine contrôlée (DRE)

Les systèmes DRE, pour Direct Recording Electronic, enregistrent directement le choix de l'électeur sous forme numérique. L'utilisateur sélectionne un candidat ou une option sur une interface, puis la machine conserve le résultat dans une mémoire interne ou le transmet à un serveur.

Ces dispositifs permettent un dépouillement rapide et peuvent intégrer des contrôles de cohérence. Ils facilitent également l'adaptation de l'interface à plusieurs langues ou à des besoins d'accessibilité. Leur principal inconvénient réside dans la difficulté de vérifier que l'enregistrement numérique correspond exactement au choix exprimé.

Lorsqu'aucune trace indépendante n'est produite, la confiance dépend fortement du logiciel et du matériel. Une erreur cachée peut être difficile à détecter. Pour cette raison, certains systèmes associent l'enregistrement électronique à une preuve papier vérifiable par l'électeur.

Figure I.2 : Exemple de machine de vote électronique DRE

I.3.2. Vote par correspondance électronique

Le vote par correspondance électronique permet à l'électeur de participer à distance au moyen d'un ordinateur ou d'un appareil connecté. Cette approche réduit les contraintes géographiques et peut être adaptée aux élections internes d'universités, d'associations ou d'entreprises.

Le vote à distance impose cependant de résoudre plusieurs problèmes simultanément. Le système doit authentifier l'électeur, protéger la communication, empêcher l'interception ou la modification du vote et préserver le secret du choix. Il doit également tenir compte du fait que le terminal utilisé n'est pas nécessairement maîtrisé par l'organisateur.

Enfin, l'absence d'environnement contrôlé augmente le risque de coercition. Une personne peut être observée ou contrainte au moment de voter. La cryptographie peut limiter certaines formes de traçabilité, mais elle ne supprime pas entièrement ce problème organisationnel.

I.3.3. Vote par kiosque Internet

Le kiosque Internet constitue une solution intermédiaire. L'électeur se rend dans un lieu supervisé, mais utilise une interface numérique reliée au système électoral. L'environnement physique peut être contrôlé, tandis que les opérations d'enregistrement et de dépouillement sont automatisées.

Cette approche permet de fournir une assistance technique et de limiter certains risques associés au terminal personnel. Elle conserve toutefois une dépendance aux postes déployés, au réseau et à l'infrastructure centrale. La sécurité doit donc couvrir à la fois le lieu de vote, les machines et les communications.

I.4. Exigences fondamentales d'un système de vote

I.4.1. Intégrité et exactitude

L'intégrité signifie qu'un vote valide doit être enregistré et compté conformément au choix exprimé. Aucun vote ne doit être ajouté, supprimé ou modifié sans autorisation. L'exactitude concerne la correspondance entre les votes valides et le résultat final.

Dans un système numérique, ces propriétés reposent sur plusieurs niveaux de contrôle : validation des entrées, authentification, protection des communications, règles d'exécution et conservation d'une trace vérifiable. Une blockchain peut renforcer l'intégrité du registre final, mais elle ne garantit pas à elle seule que toutes les données reçues avant l'enregistrement étaient correctes.

Dans zk-voting, le smart contract ne comptabilise un vote qu'après plusieurs vérifications : le scrutin doit être ouvert, le choix doit correspondre à un candidat existant, le nullifier ne doit pas avoir été utilisé et la preuve Groth16 doit être acceptée par le contrat vérificateur. Le compteur du candidat est ensuite incrémenté dans l'état du contrat.

I.4.2. Anonymat et secret du vote

L'anonymat électoral exige que l'identité du votant ne puisse pas être reliée directement à son action de vote. Le secret du bulletin porte plus précisément sur la confidentialité du choix exprimé. Ces deux notions sont liées, mais elles ne sont pas identiques.

I.4.3. Résistance à la coercition et à la vente de vote

La résistance à la coercition vise à empêcher un tiers de contraindre un électeur à voter d'une certaine manière ou d'obtenir une preuve convaincante de son choix. Cette propriété est particulièrement difficile à garantir dans un vote à distance, car l'organisateur ne contrôle pas l'environnement de l'électeur.[1], [4], [13]

I.5. Limites des systèmes de vote électronique traditionnels

I.5.1. Problème de la « boîte noire »

Un système de vote devient une « boîte noire » lorsque les participants ne peuvent pas comprendre ou vérifier comment les entrées produisent le résultat final. Le problème ne vient pas uniquement du caractère fermé du code source ; il peut également résulter d'une architecture trop complexe ou de l'absence de preuves auditables.

Dans un scrutin, la confiance ne devrait pas dépendre exclusivement d'une affirmation du fournisseur. Les règles essentielles doivent être documentées, les composants critiques examinables et les résultats accompagnés d'éléments vérifiables.

I.5.2. Risques de fraude et de manipulation

Une architecture centralisée possède généralement un serveur principal, une base de données et des comptes d'administration. La compromission de l'un de ces éléments peut permettre la modification de données, l'interruption du service ou l'accès à des informations sensibles.

Les menaces peuvent être externes, comme une attaque réseau, ou internes, comme l'abus d'un compte privilégié. La sécurité exige donc une combinaison de contrôles : séparation des rôles, journalisation, validation cryptographique, sauvegardes, surveillance et procédures organisationnelles.

I.5.3. Absence de vérification par l'électeur

Dans de nombreux systèmes classiques, l'électeur reçoit uniquement un message de confirmation. Celui-ci prouve que l'interface a affiché une validation, mais pas nécessairement que le vote a été enregistré correctement dans le résultat final.

Les systèmes vérifiables cherchent à fournir une preuve ou un reçu technique sans révéler le choix. Cette exigence doit être conciliée avec la résistance à la vente de vote : un reçu trop détaillé peut devenir une preuve transférable du choix électoral.

I.5.4. Faible participation et défiance citoyenne

La technologie peut simplifier l'accès au vote, mais elle ne produit pas automatiquement la confiance. Une interface moderne ne compense pas l'absence de transparence, une gouvernance mal définie ou des garanties de sécurité insuffisantes.

L'adoption d'un système électronique dépend de la clarté de ses règles, de la possibilité d'audit, de la qualité de l'assistance et de la capacité à expliquer les mécanismes de protection. La confiance est donc à la fois technique, institutionnelle et sociale.

I.6. Notions fondamentales sur les algorithmes de consensus

I.6.1. Définition et rôle d'un consensus distribué

Dans un réseau distribué, plusieurs nœuds conservent et mettent à jour une copie d'un même registre. Le consensus désigne l'ensemble des règles permettant à ces nœuds de s'accorder sur l'ordre et la validité des transactions, malgré les retards de communication ou le comportement incorrect de certains participants.

Le consensus ne signifie pas que tous les nœuds prennent une décision humaine sur chaque transaction. Il s'agit d'un protocole automatisé qui définit les conditions d'acceptation d'un bloc et la manière de choisir une histoire commune.

Pour un système de vote, l'intérêt principal réside dans la réduction de la dépendance à un serveur unique. Une fois la transaction incluse et finalisée selon les règles du réseau, sa modification devient beaucoup plus difficile qu'une simple mise à jour dans une base administrée par une seule entité.

I.6.2. Consensus byzantin (BFT)

Le problème des généraux byzantins illustre la difficulté d'obtenir un accord lorsque certains participants peuvent transmettre des informations contradictoires ou se comporter de manière malveillante. Un protocole tolérant aux fautes byzantines cherche à maintenir la cohérence du système tant que le nombre ou le poids des participants défaillants reste inférieur à une limite donnée.

I.6.3. Comparaison PoW, PoS et PoA

Mécanisme	Principe général	Avantages	Limites
Proof of Work (PoW)	La production de blocs repose sur un travail de calcul compétitif.	Modèle éprouvé, participation ouverte.	Consommation énergétique élevée et capacité limitée.
Proof of Stake (PoS)	Les validateurs immobilisent un actif et peuvent être pénalisés en cas de comportement incorrect.	Réduction du travail de calcul et finalité économique.	Risque de concentration du stake et complexité du protocole.
Proof of Authority (PoA)	Les blocs sont produits par un ensemble identifié de validateurs autorisés.	Débit élevé et coûts réduits.	Décentralisation limitée et dépendance à la gouvernance des validateurs.

Tableaux.I.1 : Comparaison des mécanismes de consensus PoW, PoS et PoA

I.7. impacte sur la participation electorale

L'Estonie est le premier pays au monde à avoir introduit le vote en ligne à l'échelle nationale (2005). Les données de la Commission électorale nationale montrent une progression continue de son adoption, atteignant 51% des électeurs en 2023.

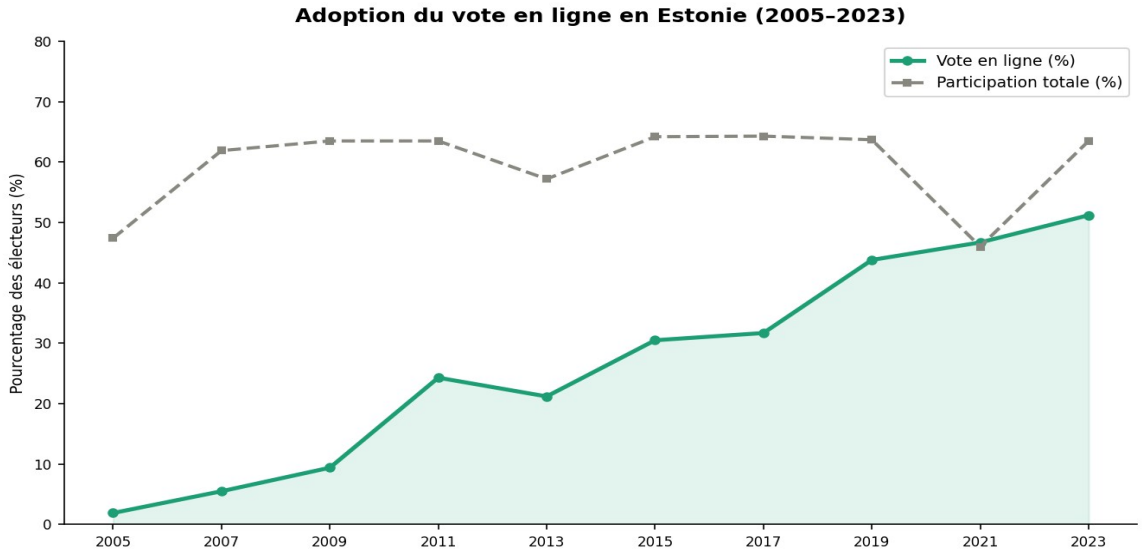

Figure.I. 3 : Adoption du vote en ligne en Estonie entre 2005 et 2023

Cette figure montre que le vote en ligne en Estonie a fortement progressé entre 2005 et 2023, passant d'environ **2 %** à plus de **51 %** des électeurs. Malgré quelques variations, la tendance générale est clairement à la hausse, ce qui montre une adoption croissante du vote numérique. La participation totale, quant à elle, reste relativement variable selon les années. [11], [12]

Conclusion

Ce chapitre a présenté l'évolution des mécanismes de vote et les principales formes de vote électronique. L'automatisation peut améliorer la rapidité et l'accessibilité, mais elle introduit des risques liés au logiciel, aux réseaux et à la concentration des privilèges.

Un système de vote sécurisé doit préserver l'intégrité, l'éligibilité, l'unicité, l'anonymat et la vérifiabilité. Ces propriétés ne sont pas garanties par un composant isolé : elles résultent de la combinaison de règles organisationnelles et de mécanismes cryptographiques.

Le consensus distribué et les registres immuables permettent de partager un état électoral commun sans dépendre d'une base de données unique. Le chapitre suivant étudie les technologies utilisées pour concrétiser cette approche dans zk-voting : blockchain, smart contracts, arbres de Merkle, fonctions de hachage et preuves zk-SNARK.

CHAPITRE II

BLOCKCHAIN, SMART CONTRACTS ET TECHNOLOGIE ZK

Introduction

La conception d'un système de vote numérique exige des mécanismes capables de combiner intégrité, contrôle d'accès et confidentialité. Une base de données classique peut enregistrer les opérations, mais elle reste administrée par une entité qui possède généralement la capacité technique de modifier son contenu. La blockchain répond à ce problème en distribuant le registre et en soumettant les mises à jour à un protocole de consensus.

Les smart contracts ajoutent une couche d'exécution programmable. Ils permettent de représenter les règles d'une élection sous forme de code : création du scrutin, définition des candidats, ouverture de la période de vote, validation des preuves et comptage des résultats. Leur caractère déterministe facilite la reproduction des mêmes contrôles par les nœuds du réseau.

Une blockchain publique est transparente par défaut. Cette propriété est utile pour l'audit, mais elle entre en tension avec l'anonymat des électeurs. Les preuves à divulgation nulle de connaissance offrent une réponse à cette difficulté : elles permettent de prouver qu'une condition est satisfaite sans révéler les données privées utilisées.[5], [6], [14]

II.1. Principes fondamentaux de la blockchain

II.1.1. Définition et architecture décentralisée

Une blockchain est un registre distribué dans lequel les participants conservent une copie d'un historique commun. Les nouvelles transactions sont regroupées, validées selon les règles du protocole, puis ajoutées à une suite de blocs liés cryptographiquement.

L'architecture décentralisée ne signifie pas nécessairement que tous les participants disposent du même pouvoir. Elle indique surtout que l'état du registre n'est pas conservé par un serveur unique. Les nœuds peuvent vérifier les données reçues et rejeter les blocs qui ne respectent pas les règles.

Dans Ethereum, les comptes soumettent des transactions qui modifient l'état du réseau. Cet état inclut les soldes, le code des contrats et leurs variables persistantes. La documentation Ethereum

décrit les smart contracts comme des programmes associés à une adresse et constitués de données et de fonctions exécutées lorsqu'ils reçoivent une transaction [5], [14] .

II.1.2. Structure d'un bloc et chaînage cryptographique

Un bloc regroupe des transactions et des informations permettant de l'intégrer à l'histoire du réseau. Il contient notamment une référence cryptographique à l'état ou au bloc précédent, ce qui crée un chaînage. Toute modification d'une donnée ancienne affecterait les empreintes dérivées et entrerait en conflit avec les copies conservées par les autres nœuds.

Les fonctions de hachage transforment une entrée de longueur variable en une empreinte de taille fixe. Une fonction cryptographique doit rendre difficile la recherche de deux entrées produisant la même empreinte et la reconstruction de l'entrée à partir du résultat.

Dans les réseaux compatibles avec Ethereum, Keccak-256 est utilisé dans plusieurs opérations, notamment pour les identifiants de fonctions, les adresses dérivées et les structures internes. Les arbres de hachage permettent également de résumer efficacement un ensemble important de données.

II.1.3. Arbre de Merkle : fonctionnement et utilité

Un arbre de Merkle est une structure hiérarchique dans laquelle chaque feuille représente une donnée et chaque nœud interne est obtenu en hachant ses enfants. Le sommet de l'arbre, appelé racine de Merkle, constitue une empreinte compacte de l'ensemble des feuilles.

Pour prouver qu'une feuille appartient à l'arbre, il n'est pas nécessaire de transmettre toutes les feuilles. Il suffit de fournir les nœuds frères rencontrés sur le chemin vers la racine, ainsi que la position gauche ou droite à chaque niveau. Le vérificateur recalcule alors les hachages successifs et compare le résultat à la racine publique.

Dans zk-voting, les feuilles sont les engagements d'identité des électeurs. Le circuit utilise un arbre de profondeur 10. Un chemin contient donc dix éléments et dix indices de position. La racine publique représente l'ensemble des électeurs inscrits pour l'élection.

Cette approche sépare l'inscription de l'électeur de son vote. Le smart contract ne stocke pas la liste complète des identités ; il conserve uniquement la racine. La preuve zk-SNARK démontre que le commitment du votant mène à cette racine sans révéler la feuille ni le chemin. [8]

II.2. Types de blockchains

II.2.1. Blockchains publiques, privées et de consortium

Type	Accès à la validation	Transparence	Gouvernance	Usage possible
Publique	Ouvert selon les règles du protocole	Élevée pour les données on-chain	Distribuée entre les participants du réseau	Scrutins nécessitant une vérification publique.
Privée	Limité à une organisation	Contrôlée par l'opérateur	Centralisée ou fortement administrée	Votes internes avec infrastructure maîtrisée.
Consortium	Réservé à plusieurs entités autorisées	Partagée entre les membres	Collective	Scrutins interinstitutionnels ou sectoriels.

TABLEAUX.II.1. type des different block-chain

Une blockchain publique offre une forte capacité d'audit externe, mais ses transactions peuvent avoir un coût variable et ses données sont largement visibles. Une blockchain privée permet un contrôle opérationnel plus direct, mais elle réduit l'indépendance de la vérification.

Le consortium constitue un compromis : plusieurs institutions exploitent les validateurs et partagent la gouvernance. La pertinence de chaque modèle dépend de la nature de l'élection, des exigences réglementaires et du niveau de confiance entre les acteurs.[5], [6], [14]

II.2.2. Choix adapté pour un système de vote

Le choix d'une blockchain pour le vote doit prendre en compte la finalité, les frais, la disponibilité, la confidentialité et la gouvernance. Un scrutin interne à faible enjeu n'a pas les mêmes contraintes qu'une élection publique nationale.

II.3. Smart contracts

II.3.1. Définition et historique

Un smart contract est un programme déployé sur une blockchain. Il possède une adresse, un état persistant et des fonctions pouvant être appelées par des transactions ou par d'autres contrats. Le terme « intelligent » ne signifie pas que le programme prend des décisions autonomes au sens de l'intelligence artificielle ; il exécute exactement les règles codées.

Ethereum a généralisé l'usage de smart contracts programmables grâce à l'Ethereum Virtual Machine. Les applications peuvent ainsi représenter des actifs, des droits, des règles de gouvernance ou des processus métier directement dans l'état de la blockchain.

II.3.2. Fonctionnement et exécution déterministe

Lorsqu'une transaction appelle une fonction, chaque nœud chargé de l'exécution doit obtenir le même résultat à partir du même état et des mêmes entrées. Cette propriété impose un environnement déterministe. Le contrat ne peut pas consulter directement une source externe non déterministe sans mécanisme intermédiaire.

L'exécution consomme du gas, une unité mesurant les ressources nécessaires. Le coût limite les boucles, le stockage et les calculs complexes. Les zk-SNARKs sont intéressants dans ce contexte car un calcul important peut être effectué hors chaîne, tandis que le contrat vérifie une preuve compacte.

II.3.3. Langages de développement : Solidity et Vyper

Solidity est le langage le plus utilisé pour développer des contrats EVM. Sa syntaxe est inspirée de langages impératifs courants et permet de définir des structures, événements, erreurs, modificateurs et fonctions. , Vyper constitue une autre option pour l'EVM. Il privilégie une syntaxe plus restreinte et la lisibilité.

II.3.4. Applications dans les systèmes de vote

Dans un système de vote, un smart contract peut définir les candidats, la période de vote et les conditions d'acceptation. Il peut également produire des événements permettant de suivre le déroulement du scrutin.

II.4. Cryptographie appliquée à la blockchain

II.4.1. Cryptographie asymétrique et signatures numériques

La cryptographie asymétrique repose sur une paire de clés : une clé privée conservée par son propriétaire et une clé publique pouvant être diffusée. Une signature numérique permet de démontrer qu'une transaction a été autorisée par le détenteur de la clé privée.

Sur une blockchain EVM, l'adresse qui envoie une transaction est dérivée d'une clé publique et la transaction est signée.

Il est important de distinguer cette identité blockchain de l'identité électorale anonyme. L'organisation utilise une adresse pour créer et administrer l'élection, tandis que l'électeur est représenté dans le circuit par un engagement et un nullifier. Le modèle évite ainsi de demander au contrat de relier directement l'adresse personnelle de l'électeur à son choix.

II.4.2. Preuves à divulgation nulle de connaissance (ZKP)

Une preuve à divulgation nulle de connaissance permet à un prouveur de convaincre un vérificateur qu'une affirmation est vraie sans révéler les données privées utilisées pour établir cette affirmation.

Trois propriétés sont généralement recherchées. La complétude signifie qu'une affirmation vrai peut être acceptée lorsque les participants suivent le protocole. La solidité signifie qu'un prouveur ne doit pas pouvoir faire accepter une affirmation fausse, sauf avec une probabilité négligeable. La propriété zero-knowledge signifie que la preuve ne révèle pas le témoin privé au-delà de la validité de l'affirmation.

II.5. Technologie zk-SNARK

II.5.1. Définition et décomposition du concept

L'acronyme zk-SNARK signifie Zero-Knowledge Succinct Non-Interactive Argument of Knowledge.

« Zero-Knowledge » indique que le témoin privé n'est pas révélé. « Succinct » signifie que la preuve est courte et que sa vérification est relativement rapide. « Non-Interactive » signifie qu'après la préparation des paramètres, le prouveur peut produire une preuve unique sans dialogue répété avec le vérificateur. Enfin, « Argument of Knowledge » indique que la validité de la preuve implique la connaissance d'un témoin satisfaisant les contraintes.

Ces propriétés sont adaptées aux blockchains, où la vérification doit être compacte et déterministe. Le calcul du témoin peut être effectué sur le navigateur ou sur un serveur, tandis que la vérification est intégrée dans un smart contract.

II.5.2. Principe de fonctionnement : Setup, Prover et Verifier

Le fonctionnement d'un système basé sur les preuves à divulgation nulle de connaissance peut être expliqué à travers trois rôles principaux : le **Setup**, le **Prover** et le **Verifier**.

La première étape est le **Setup**. Elle correspond à une phase de préparation du système. Durant cette étape, les paramètres nécessaires à la création et à la vérification des preuves sont générés. Ces paramètres permettent ensuite au système de fonctionner correctement et de garantir que les preuves produites pourront être contrôlées de manière fiable.

La deuxième étape concerne le **Prover**, c'est-à-dire le prouveur. Son rôle est de générer une preuve montrant qu'il possède certaines informations ou qu'il respecte certaines conditions, sans révéler directement ces informations. Dans le cas d'un système de vote, le prouveur peut par exemple démontrer qu'il est un électeur autorisé, sans dévoiler son identité réelle.

La troisième étape est celle du **Verifier**, ou vérificateur. Son rôle est de contrôler la validité de la preuve fournie par le prouveur. Si la preuve est correcte, le système accepte l'opération ; dans le cas contraire, elle est rejetée. Le vérificateur ne découvre pas les données privées du prouveur, mais il peut s'assurer que les règles prévues ont bien été respectées.

II.5.3. Applications classiques

Les zk-SNARKs sont employés dans les systèmes de transactions privées, les protocoles d'identité, les rollups et les applications de conformité sélective. Ils permettent par exemple de prouver qu'un utilisateur satisfait une condition sans publier la donnée complète.

Dans le vote électronique, l'affirmation peut être formulée ainsi : « je connais un secret et un nullifier dont l'engagement appartient à l'arbre des électeurs autorisés, et le nullifier hash public a été calculé correctement ». Le vérificateur apprend que l'affirmation est valide, mais il ne reçoit pas les valeurs privées.

II.6. Intégration blockchain + ZK pour le vote électronique

II.6.1. Avantages de la combinaison

La blockchain et les preuves zero-knowledge répondent à des besoins complémentaires. La blockchain fournit un état partagé, persistant et vérifiable. La preuve ZK permet de contrôler une condition privée sans inscrire les données sensibles dans cet état.

Dans zk-voting, la racine de Merkle est publique, mais les engagements et les chemins utilisés par l'électeur restent hors chaîne. Le nullifier hash est public afin que le contrat puisse interdire sa réutilisation. Le choix est également public dans la version de comptage en direct.

Le résultat est un protocole dans lequel l'éligibilité est prouvée cryptographiquement, l'unicité est imposée par l'état du contrat et le comptage est consultable publiquement.

II.6.2. Enregistrement du vote dans le smart contract

L'enregistrement du vote dans le smart contract constitue une étape essentielle du système. Lorsqu'un électeur souhaite voter, il envoie une preuve cryptographique accompagnée de son choix. Le rôle du smart contract est alors de vérifier que cette preuve est valide et que l'électeur n'a pas déjà participé au scrutin.

Avant d'accepter le vote, le smart contract effectue plusieurs contrôles. Il vérifie d'abord que la période de vote est bien ouverte. Il s'assure ensuite que l'électeur n'a pas déjà utilisé son droit de vote, afin d'empêcher le double vote. Il contrôle également que le candidat choisi existe bien dans la liste des candidats autorisés.

Si toutes ces conditions sont respectées et que la preuve est acceptée, le vote est enregistré. Le système marque alors l'électeur comme ayant déjà voté, ajoute une voix au candidat sélectionné et met à jour le nombre total de votes. Cette opération permet de garantir qu'un électeur autorisé ne peut voter qu'une seule fois, tout en préservant son anonymat.

Ainsi, le smart contract joue le rôle d'un arbitre automatique : il applique les règles du scrutin de manière transparente, rejette les votes invalides et enregistre uniquement les votes qui respectent les conditions prévues.

II.6.4. Comparaison avec une validation centralisée

Critère	Validation centralisée	Blockchain + ZK
Éligibilité	Contrôle par une base ou un serveur.	Preuve d'appartenance à une racine publique.
Double vote	Champ ou verrou dans la base.	Nullifier hash stocké dans l'état du contrat.
Identité	Souvent connue du serveur de vote.	Secret et nullifier non transmis au contrat.
Audit du résultat	Dépend des journaux de l'opérateur.	État et transactions consultables sur le réseau.
Exécution des règles	Application serveur modifiable par l'opérateur.	Code du contrat exécuté de manière déterministe.
Coût	Infrastructure serveur.	Infrastructure applicative plus frais éventuels de transactions.

TABLEAUX.II.2.Comparaison avec une validation centralisée

Ce tableau montre que l'approche blockchain + ZK réduit la dépendance à un serveur central. Elle permet de vérifier l'éligibilité sans révéler l'identité, d'empêcher le double vote avec un nullifier hash et d'auditer publiquement l'état du scrutin grâce au smart contract.

Conclusion

Ce chapitre a présenté les fondements techniques de zk-voting. La blockchain fournit un registre partagé et une exécution déterministe des règles électorales. Les smart contracts représentent les élections et assurent la vérification, l'unicité des nullifiers et le comptage des voix.

Les arbres de Merkle permettent de résumer l'ensemble des électeurs autorisés par une racine compacte. Poseidon est utilisé pour calculer les engagements et les nullifier hashes dans un circuit adapté aux contraintes arithmétiques.

Le protocole produit une preuve succincte vérifiée . Le circuit démontre la connaissance d'une identité cryptographique inscrite dans l'arbre sans révéler le secret, le nullifier ou le chemin. L'association de ces composants constitue la base cryptographique et blockchain du système zk-voting.

CHAPITRE III

CONTRIBUTION, CONCEPTION ET IMPLÉMENTATION DE ZK-VOTING

III.1. Introduction

Ce chapitre présente la contribution réalisée dans le cadre du projet zk-voting. Après l'étude du vote électronique, de la blockchain et des preuves à divulgation nulle de connaissance, il décrit les besoins du système, son architecture, les technologies retenues et le fonctionnement complet d'une élection.

zk-voting est une plateforme web permettant à une organisation de créer une élection, de définir des candidats, d'inviter des électeurs et de publier les résultats.

III.2. Présentation du projet zk-voting

III.2.1. Description générale

Le système est destiné aux organisations souhaitant organiser des scrutins numériques. L'organisation dispose d'un tableau de bord pour créer et suivre ses élections. Les électeurs utilisent un lien d'invitation et accèdent à une interface de vote.

Chaque électeur génère un secret et un nullifier. Leur combinaison produit un commitment Poseidon, inséré dans un arbre de Merkle. Lors du vote, une preuve démontre que ce commitment appartient à l'arbre sans révéler les données privées.

Le contrat reçoit la preuve, le nullifier hash et le choix du candidat. Il contrôle la preuve, l'unicité du nullifier, la validité du candidat et la période de vote avant d'incrémenter le résultat.

III.2.2. Objectifs du système

- Créer et administrer des élections depuis une interface web.
- Inviter une liste déterminée d'électeurs.
- Vérifier l'éligibilité sans transmettre l'identité au smart contract.
- Empêcher le double vote au moyen d'un nullifier hash.
- Enregistrer les votes valides sur une blockchain compatible EVM.
- Afficher les résultats de manière publique.
- Proposer une gestion de facturation destinée aux organisations
- verification indifviduelle et globale des votes

III.2.3. Acteurs du système

Acteur	Responsabilités
Organisation	Créer un compte, créer une élection, définir les candidats, importer les électeurs, envoyer les invitations, ouvrir ou fermer le vote et consulter les résultats.
Électeur	Utiliser son invitation, générer son identité cryptographique, produire une preuve et soumettre un vote.
Public	Consulter les informations et résultats publics d'une élection.

TABLAUX.III.1.Acteurs du système

III.2.4. Fonctionnalités principales

Fonctionnalité	Description
Authentification	Inscription et connexion des organisations.
Création d'élection	Saisie des informations, des candidats et de la durée.
Gestion des électeurs	Importation des adresses et création des invitations.
Arbre de Merkle	Ajout des commitments et génération des chemins d'appartenance.
Preuve zk-SNARK	Génération Groth16 à partir des artefacts Circom.
Vote blockchain	Vérification on-chain et mise à jour du compteur.
Résultats publics	Affichage du nombre de voix par candidat.
Facturation	Présentation des offres et paiement avec un service externe de facturation Stripe.

TABLAUX.III.2. Fonctionnalités principales

III.3. Analyse des besoins

III.3.1. Besoins fonctionnels

III.3.1.1. Authentification des organisations

Une organisation doit pouvoir créer un compte et accéder à des routes protégées. L'API utilise des jetons JWT pour identifier la session.

III.3.1.2. Création d'une élection

L'organisation saisit le nom, les paramètres et au moins deux candidats. L'élection est conservée dans MongoDB et peut être associée à un contrat déployé.

III.3.1.3. Gestion des électeurs

Le système importe les adresses électroniques, crée des invitations et suit le parcours de chaque électeur.

III.3.1.4. Construction de l'arbre de Merkle

Les commitments sont insérés dans l'arbre de l'élection. Le service calcule la racine et fournit les éléments du chemin.

III.3.1.5. Soumission du vote

La preuve, le nullifier hash et le choix sont transmis au backend, puis formatés pour l'appel du contrat.

III.3.1.6. Consultation des résultats

Les compteurs de candidats sont récupérés et affichés dans une page publique.

III.3.1.7. Gestion des abonnements

La plateforme présente les offres et crée une session Stripe Checkout.

III.3.2. Exigences

Besoin	Mécanismes retenus
Sécurité	JWT, Bcrypt, Helmet, CORS, rate limiting, validation des entrées et contrôles des contrats.
Confidentialité	Secret, nullifier et chemin Merkle non transmis au contrat.
Intégrité	Preuve cryptographique, nullifier unique et enregistrement on-chain.
Disponibilité	Séparation du frontend, du backend, de MongoDB et de la blockchain.
Performance	Calcul de la preuve hors chaîne et vérification succincte on-chain.
Maintenabilité	Dépôt organisé en composants, routes, services, contrats, circuits et tests.
Traçabilité	Transactions et événements produits par les contrats.

TABLAUX.III.3.Exigences

III.4.1. Interface web

Le frontend React contient les pages d'authentification, le tableau de bord, la création d'élections, le détail d'une élection, la page d'invitation, l'interface de vote, les résultats et la facturation.

III.4.2. API backend

Le backend Express gère l'authentification, les élections, les électeurs, le vote, les résultats publics, les courriels et la facturation

III.4.3. Base MongoDB

MongoDB conserve les organisations, les élections, les électeurs, les invitations et les informations applicatives nécessaires.

III.4.4. Couche cryptographique

La couche cryptographique joue un rôle essentiel dans la protection du système de vote. Elle permet de vérifier qu'un électeur est bien autorisé à participer au scrutin, sans révéler son identité réelle ni ses informations privées

III.4.5. Couche blockchain

La couche blockchain constitue la partie responsable de l'exécution et de l'enregistrement des opérations de vote. Elle permet de créer les élections, d'appliquer automatiquement les règles du scrutin et de conserver les résultats dans un registre transparent et difficile à modifier.

Dans cette couche, les smart contracts jouent un rôle central. Ils définissent les conditions nécessaires pour accepter un vote, comme l'ouverture du scrutin, la validité de la preuve et l'unicité du vote. Lorsqu'un vote respecte toutes les règles prévues, il est enregistré dans la blockchain et les résultats sont mis à jour automatiquement.

III.5. Modélisation UML

III.5.1. Diagramme de cas d'utilisation

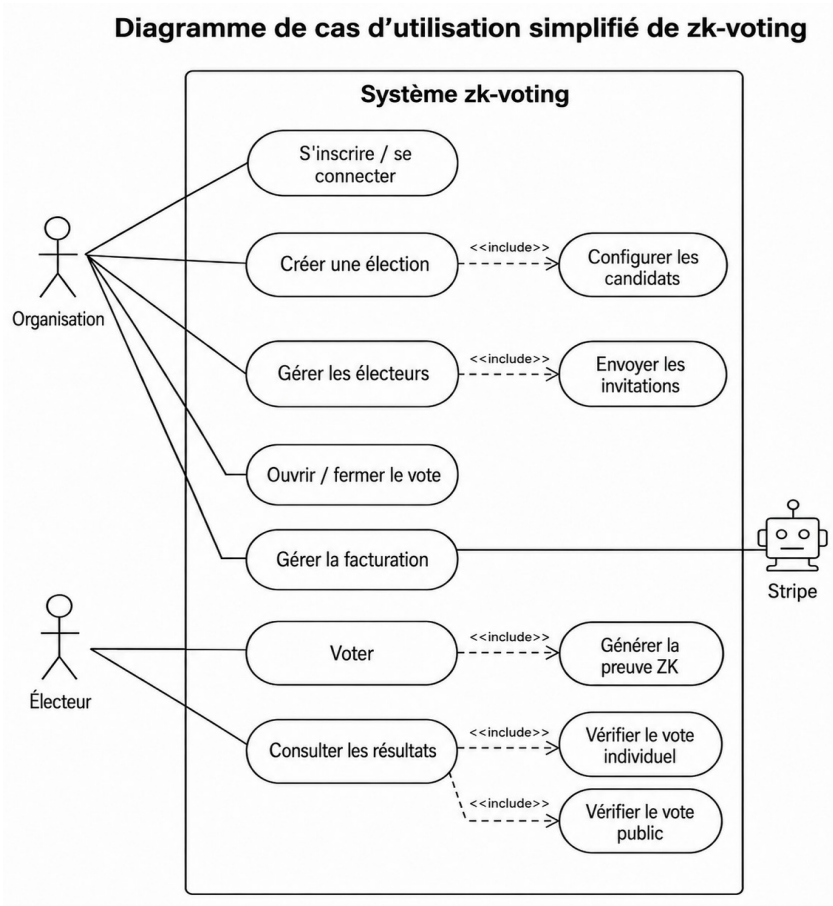

FIGURE.III.1.Diagramme de cas d'utilisation

Ce diagramme présente les principales interactions entre les acteurs et le système zk-voting. L'organisation peut créer un compte, configurer une élection, importer les électeurs, envoyer les invitations, ouvrir ou fermer le scrutin et consulter les résultats. L'électeur peut accéder à son invitation, générer son identité cryptographique, produire une preuve ZK et soumettre son vote. Le public peut consulter les résultats publiés. Le diagramme montre aussi l'intervention de services externes, notamment la blockchain pour l'enregistrement et la vérification du vote, ainsi que Stripe pour la facturation.

III.5.2. Séquence de création d'une élection

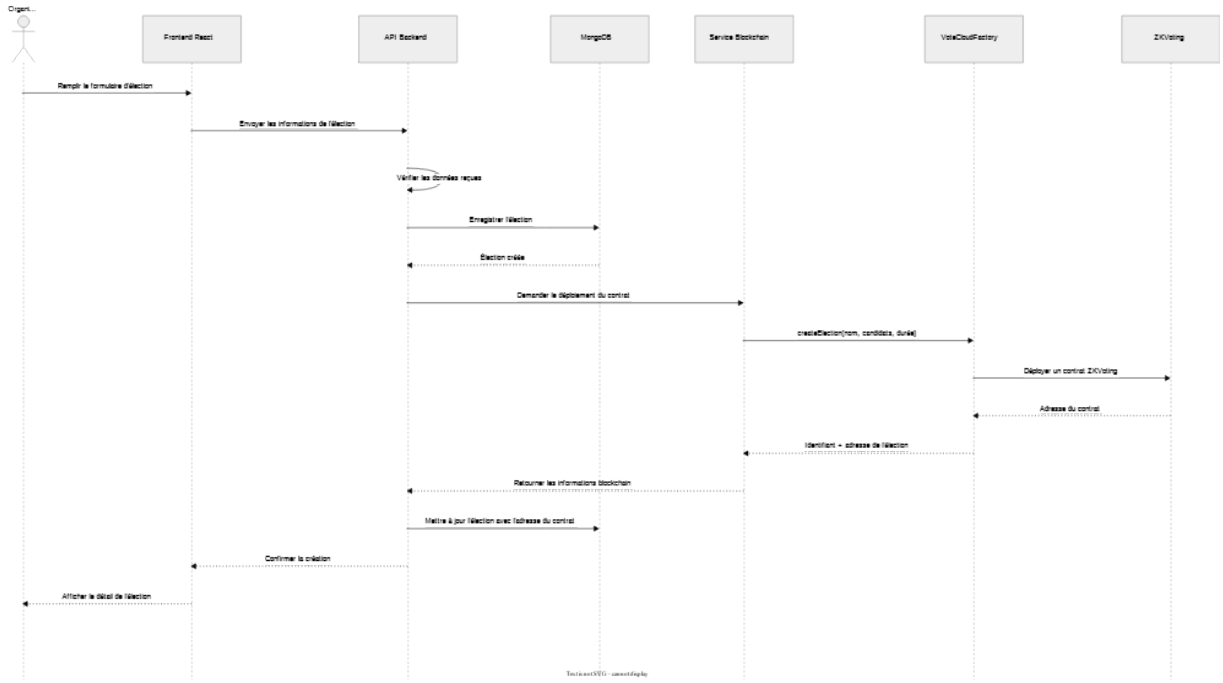

Figure.III.2 : Séquence de création d'une élection

Cette figure illustre le déroulement de la création d'une élection. L'organisation remplit le formulaire depuis l'interface React, puis les informations sont envoyées à l'API backend. Le

backend vérifie les données, enregistre l'élection dans MongoDB et demande au service blockchain de déployer le contrat associé

III.5.3. Séquence d'inscription cryptographique

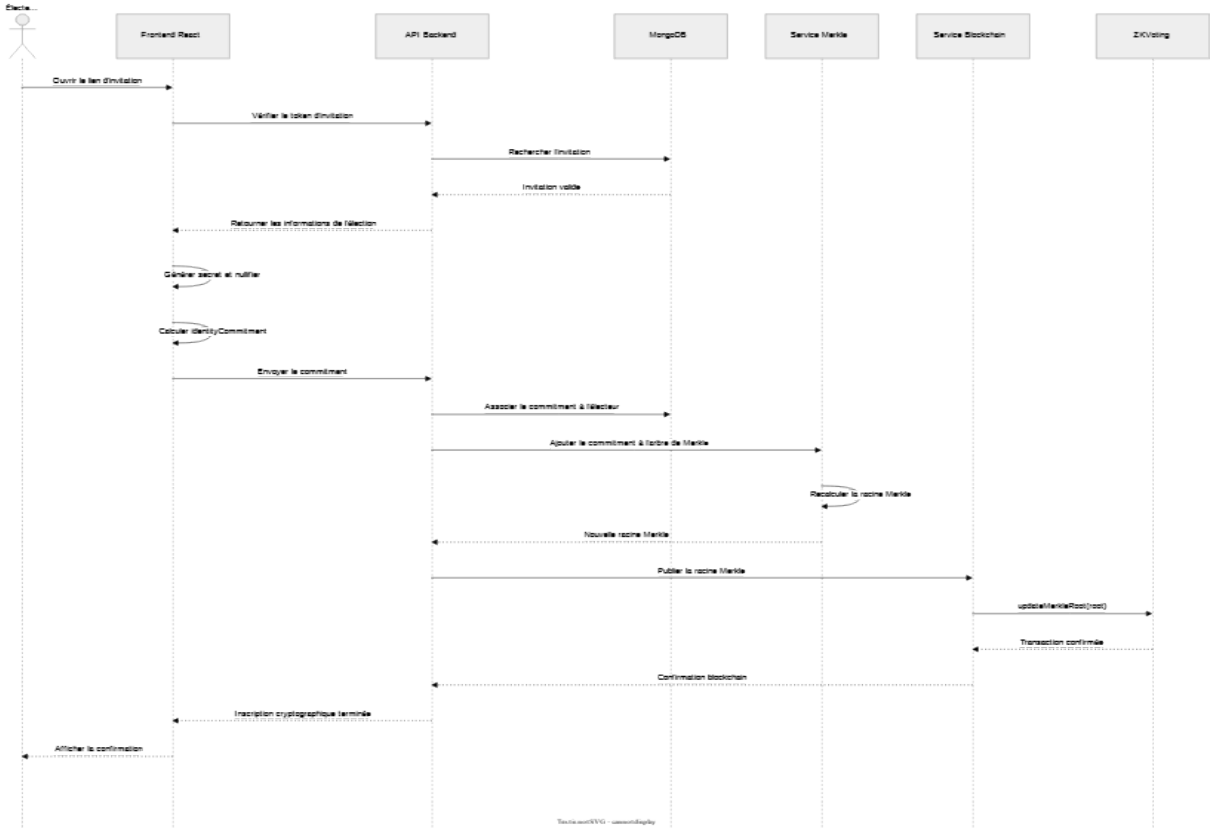

Figure.III.3 : Diagramme de séquence d'inscription cryptographique d'un électeur

Ce diagramme décrit l'étape d'inscription cryptographique d'un électeur. Après l'ouverture du lien d'invitation, le backend vérifie la validité du token et retourne les informations de l'élection

III.5.4. Séquence de soumission du vote

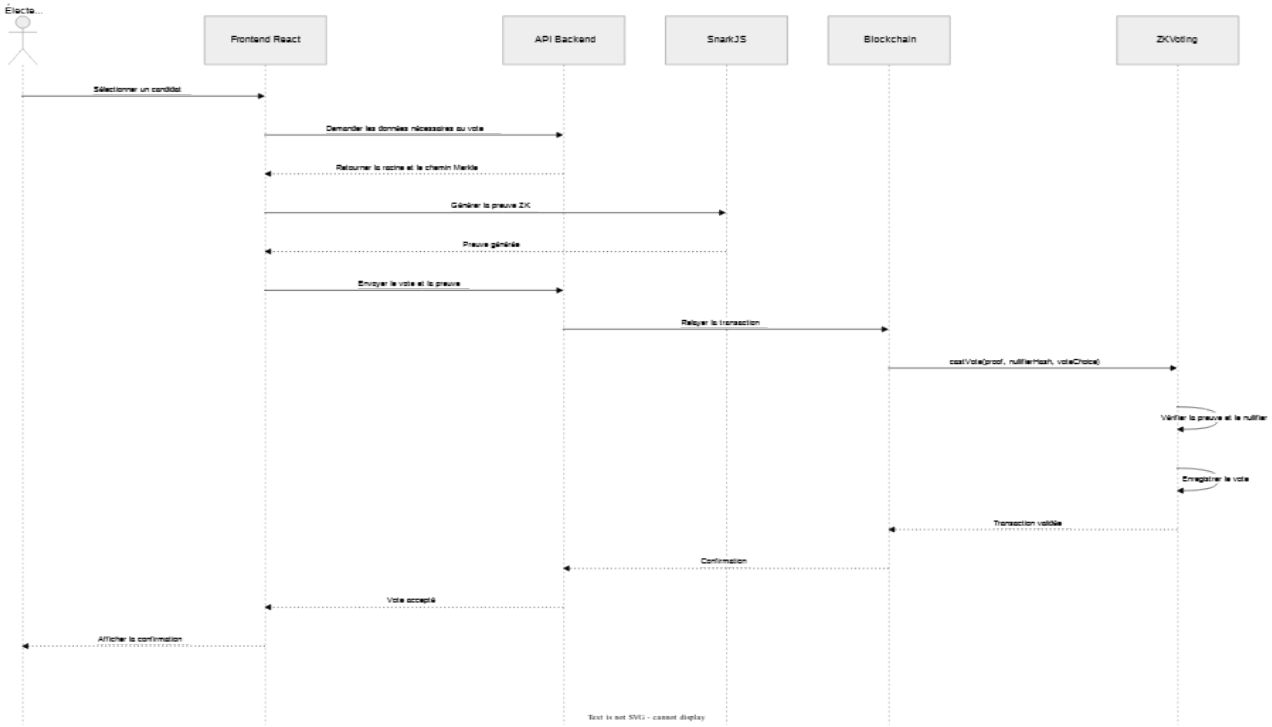

Figure.III.4 : Diagramme de séquence de soumission d'un vote

Cette figure montre les principales étapes de la soumission d'un vote. L'électeur sélectionne un candidat dans l'interface, puis le frontend demande au backend les données nécessaires à la preuve

III.6. Outils et technologies utilisés

Domaine	Technologies
Frontend	React 18, React Router, SnarkJS, CircomlibJS, Ethers.js,.
Backend	Node.js, Express.js, MongoDB, JWT, Helmet, CORS, Nodemailer et Stripe.
Blockchain	Solidity 0.8.24, Hardhat et Ethers.js.
Zero-Knowledge	Circom 2.1.6, SnarkJS 0.7.4, Groth16, Poseidon et arbre de Merkle

TABLAUX.III.5 : Outils et technologies utilisés

un tablaux qui present les technologies utilisés dans ce project

III.7. interface de l'application web

III.7.1. Inscription

L'organisation saisit les informations nécessaires à la création de son compte.

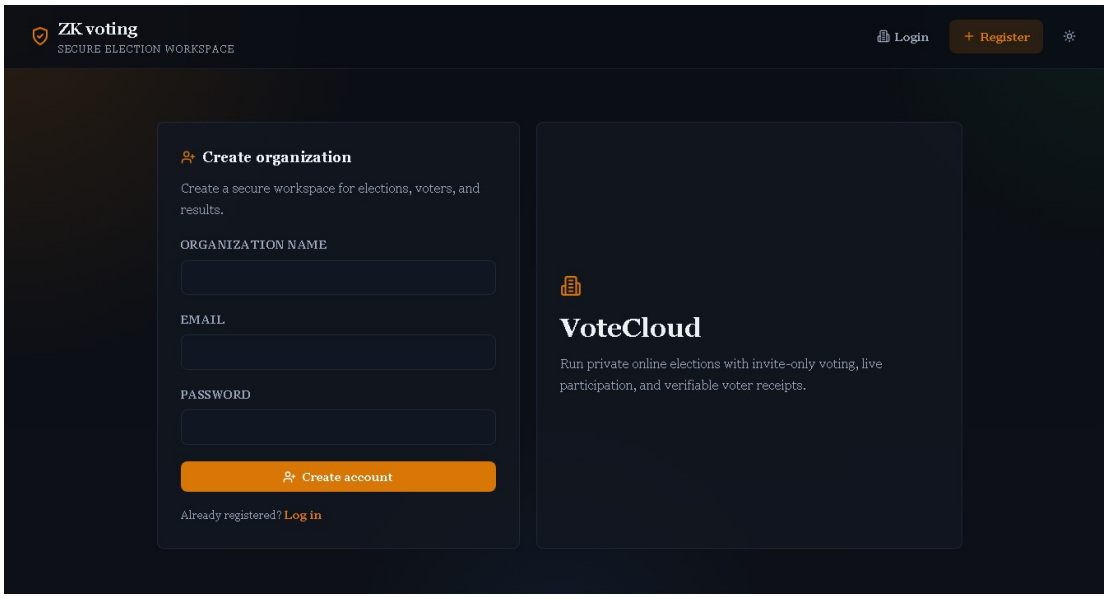

Figure.III.5 : Interface d'inscription d'une organisation

III.7.2. Connexion

La page de connexion transmet les identifiants à l'API et reçoit un jeton JWT.

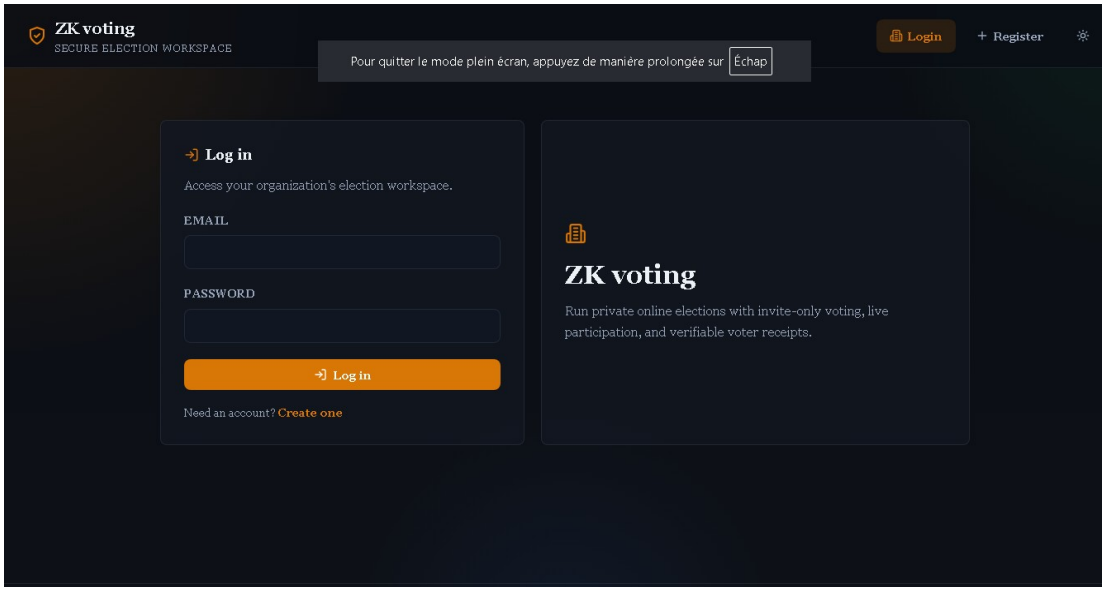

Figure.III.6 : Interface de connexion

III.7.3. Tableau de bord

Le dashboard regroupe les élections et les principales actions disponibles.

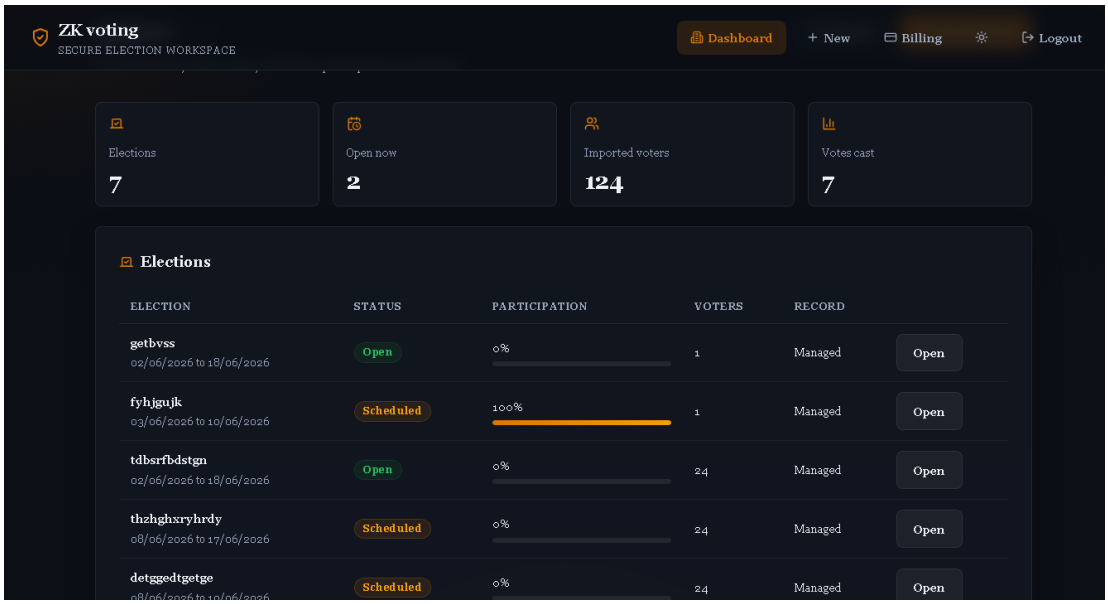

Figure.III.7 : Tableau de bord de l'organisation

III.7.4. Création d'une élection

Les formulaires permettent de définir les informations du scrutin et les candidats.

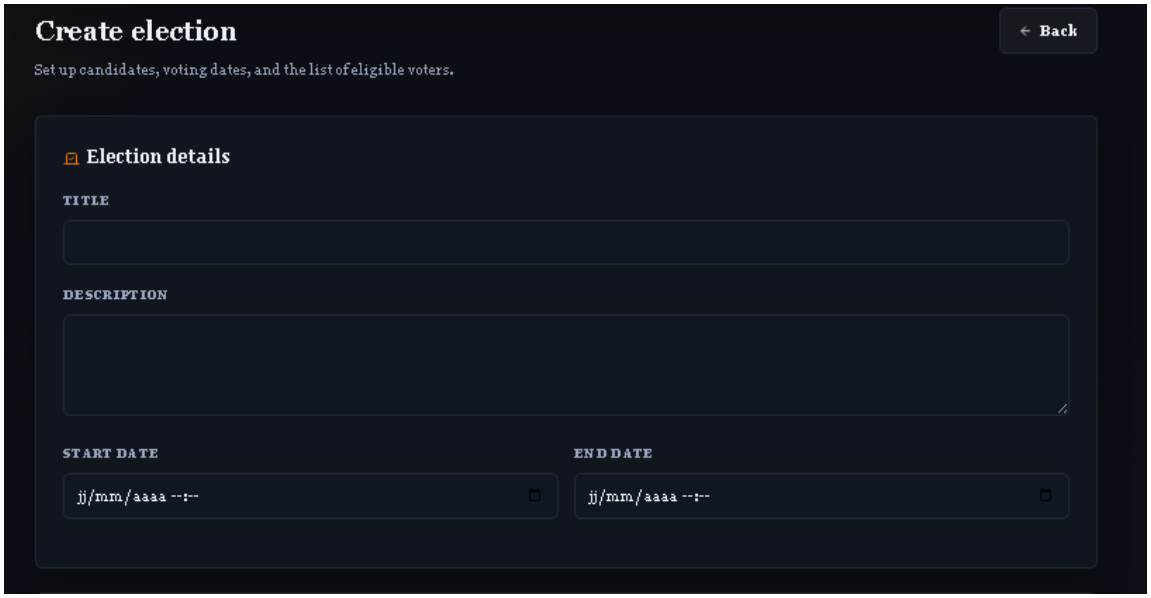

Figure.III.8 : Interface de création d'une élection

III.7.5. Configuration des candidats

La seconde étape complète la configuration des choix proposés. les candidat sont importer par une list csv des email des votant

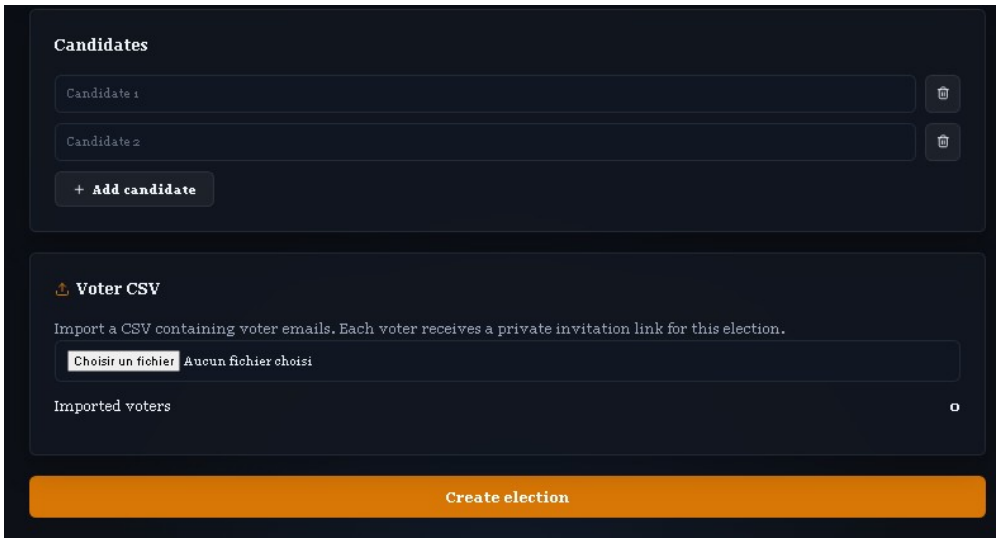

Figure.III.9 : Interface de configuration des candidats et importation des électeurs

III.7.6. Détail d'une élection

Cette page centralise l'état, les paramètres, les électeurs et les résultats.

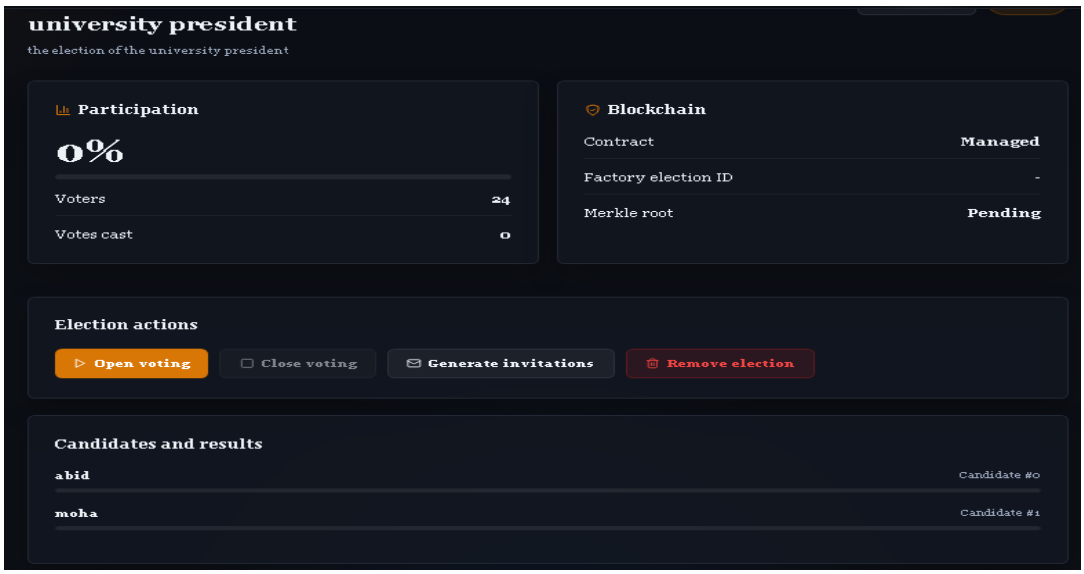

Figure.III.10 : Interface de détail d'une élection

III.7.7. Interface de vote

L’électeur sélectionne un candidat puis il génére la preuve.est enfin soumettre le vote

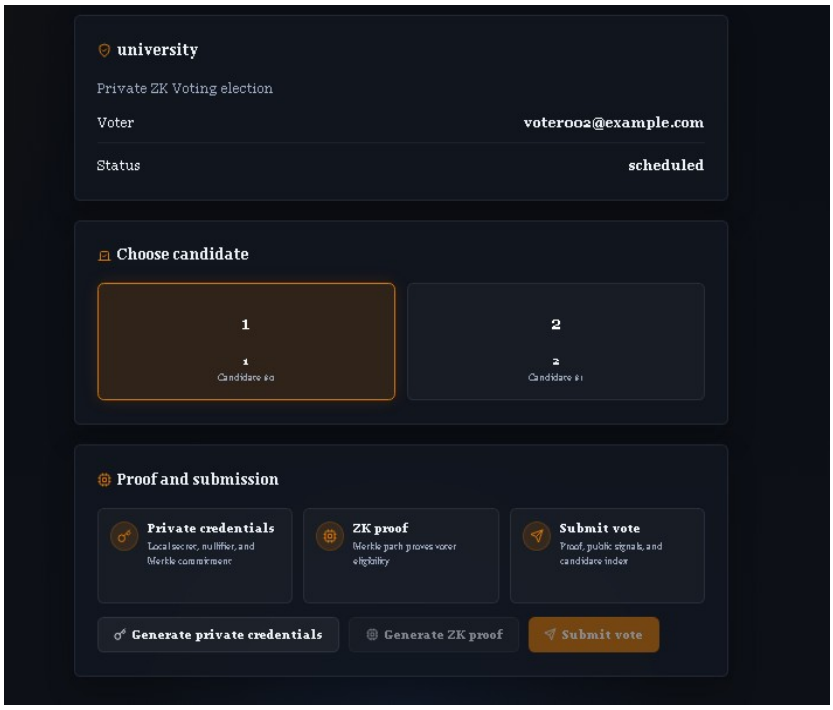

Figure.III.11 : Interface de vote

III.7.8. Résultats publics

La page affiche le nombre de voix enregistré pour chaque candidat.

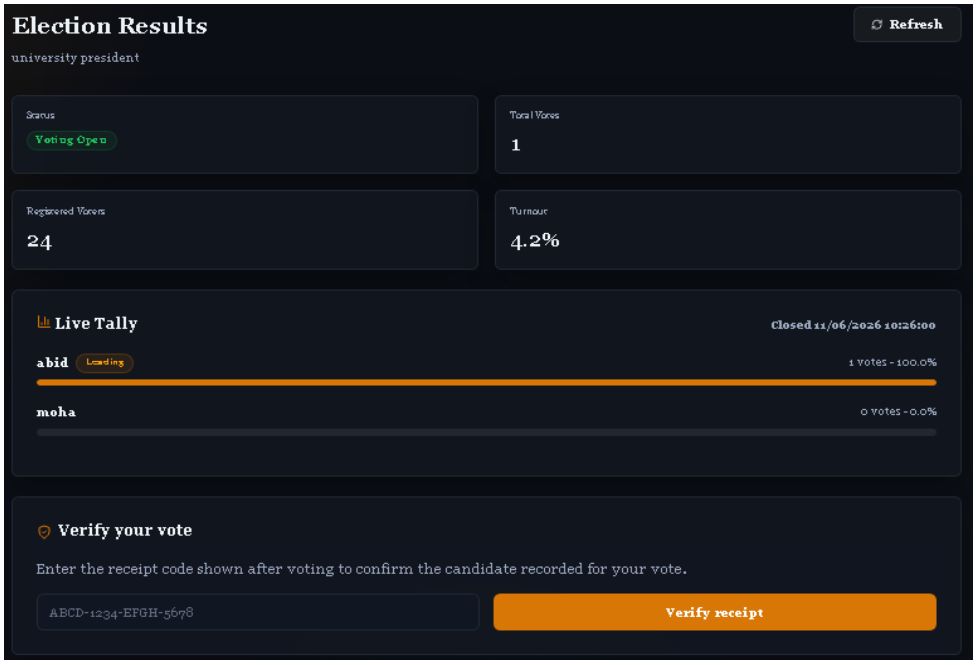

Figure.III.12 : Résultats public

III.7.9. Facturation

La page présente des formules de facturation dans la plateforme. L'organisation a le choix entre plusieurs formules qui propose plus de liberté en création d'élection.

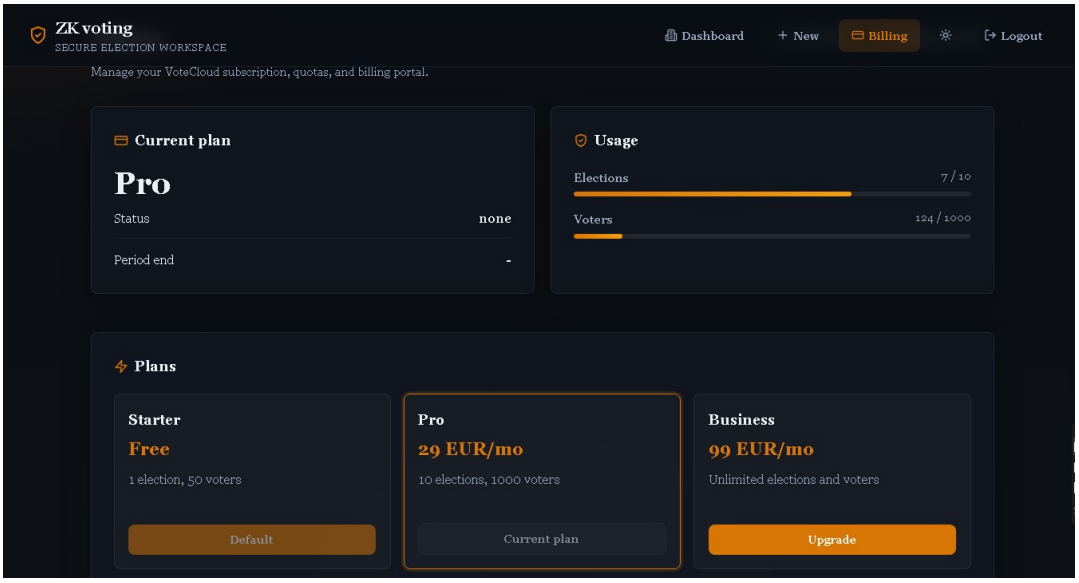

Figure.III.13 : Facturation

III.8. Vérification individuelle et vérification publique du vote

La vérification constitue une propriété essentielle dans un système de vote électronique. Elle permet de renforcer la confiance dans le processus électoral en donnant aux participants et aux observateurs la possibilité de contrôler que les opérations importantes ont bien été exécutées. Dans le cadre de zk-voting, la vérification est assurée à deux niveaux complémentaires : la vérification individuelle et la vérification publique.

III.8.1. Vérification individuelle du vote

La vérification individuelle permet à un électeur de s'assurer que son vote a bien été accepté par le système. Après la soumission du vote, l'électeur ne reçoit pas simplement une confirmation affichée par l'interface web. Son vote correspond à une transaction envoyée vers le smart contract chargé de l'élection.

Lorsque la transaction est confirmée sur la blockchain, elle produit un reçu technique. Ce reçu contient des informations telles que le hash de la transaction, l'adresse du contrat appelé,

le bloc dans lequel la transaction a été incluse et les événements émis par le contrat. Ces éléments permettent de vérifier que l'appel à la fonction de vote a bien été exécuté.

Dans zk-voting, un vote n'est accepté que si plusieurs conditions sont satisfaites. Le scrutin doit être ouvert, le candidat choisi doit être valide, le nullifier hash ne doit pas avoir été utilisé auparavant et la preuve zk-SNARK doit être validée par le contrat vérificateur. Si l'une de ces conditions échoue, la transaction est rejetée et le vote n'est pas comptabilisé.

Après un vote valide, le smart contract marque le nullifier hash comme utilisé, incrémente le compteur du candidat choisi et augmente le nombre total de votes. L'électeur peut donc vérifier que sa transaction a été acceptée sans révéler son secret, son nullifier privé ou son identité réelle. Cette approche fournit une preuve technique d'acceptation du vote tout en préservant la séparation entre l'identité du votant et l'opération enregistrée sur la blockchain.

III.8.2. Vérification publique du processus

La vérification publique, aussi appelée vérification universelle, permet à des observateurs externes de contrôler le déroulement du scrutin sans avoir accès aux données privées des électeurs. Dans un système centralisé classique, cette vérification dépend fortement des journaux du serveur ou des informations fournies par l'administrateur. Dans zk-voting, une partie importante de la vérification repose sur l'état public du smart contract.

Le contrat conserve plusieurs informations consultables : la racine de Merkle représentant les électeurs autorisés, les nullifier hashes déjà utilisés, le nombre total de votes acceptés et les résultats associés à chaque candidat. Ces données peuvent être lues par l'interface publique ou directement depuis la blockchain à travers les fonctions de lecture du contrat.

Un observateur peut donc vérifier que le nombre total de votes correspond à la somme des votes des candidats. Il peut également vérifier qu'un même nullifier hash n'a pas été utilisé plusieurs fois. Cette propriété est importante, car elle permet de contrôler publiquement la prévention du double vote sans connaître l'identité réelle des électeurs.

La vérification publique concerne également les règles d'exécution. Le smart contract applique toujours les mêmes conditions : période de vote valide, candidat existant, nullifier non dépensé et preuve cryptographique correcte. Comme ces règles sont inscrites dans le contrat, elles ne dépendent pas d'une décision manuelle du backend au moment du comptage.

III.9. Rôle de la preuve zk-SNARK dans la vérification

La preuve zk-SNARK joue un rôle central dans la vérification du vote. Elle permet à l'électeur de démontrer qu'il appartient à l'ensemble des votants autorisés sans révéler les

valeurs privées utilisées dans le circuit. Ces valeurs comprennent notamment le secret, le nullifier, les éléments du chemin de Merkle et les indices de position dans l'arbre.

Le smart contract ne vérifie pas directement l'identité civile ou l'adresse électronique de l'électeur. Il vérifie uniquement une preuve cryptographique. Cette preuve établit que l'électeur connaît une identité cryptographique valide, que l'engagement correspondant appartient à l'arbre de Merkle de l'élection et que le nullifier hash public a été correctement dérivé.

Ainsi, la vérification de l'éligibilité est réalisée sans publier les données privées du votant. Cette séparation constitue l'un des avantages principaux de l'intégration des preuves zero-knowledge dans le système de vote.

III.10. Conclusion

Ce chapitre a détaillé la réalisation de zk-voting, depuis l'analyse des besoins jusqu'à l'intégration de l'interface, du backend, du circuit et des smart contracts.

La solution sépare les données applicatives des données publiques du vote. L'éligibilité est représentée par l'arbre de Merkle, la preuve protège les entrées privées et le nullifier hash empêche un second vote.

L'architecture obtenue permet de gérer le cycle complet d'une élection depuis une application web, tout en utilisant la blockchain pour la vérification et le comptage.

CHAPITRE IV : BUSINESS MODEL CANVAS DU PROJET ZK-VOTING

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Abou Bekr Belkaid Tlemcen

Business Model Canvas

BMC

Date de dépôt :

N° de projet :

Faculté/Institut :universite de tlemcen

Département :science

Nom du projet : zk-voting

Encadrant 1 :Belhocin Amine

Encadrant 2 :

Co-encadrant 1 :

Co-encadrant 2 :

Etudiants : - Salah mohamed habib -

- -

Année universitaire : 2023/2024

IV.1.Proposition de valeur (Value Proposition)القيمة المقترحة

a. Quels problèmes résolvons-nous pour nos clients ?

Leما هي المشاكل التي نحلها لعملائنا ؟

projet zk-voting vise à résoudre plusieurs problèmes présents dans les systèmes de vote électroniques traditionnels et centralisés. Les systèmes classiques souffrent souvent d'un manque de transparence, de risques de manipulation des résultats, de problèmes de sécurité informatique et d'une faible confiance des électeurs.

Notre solution permet de garantir l'intégrité des votes grâce à la blockchain Ethereum, d'empêcher le double vote via les nullifierHash, et de protéger l'identité des électeurs grâce aux preuves cryptographiques ZK-SNARK et aux arbres de Merkle.

b. Quels besoins de nos clients satisfont nos produits ou services ?

ما هي الاحتياجات التي يلبيها منتجاتنا أو خدماتنا لعملائنا؟

Le système répond aux besoins de sécurité, de transparence et de vérification publique dans les élections numériques. Il permet également d'assurer un vote électronique sécurisé, accessible et résistant aux manipulations.

Le projet satisfait aussi les besoins des institutions souhaitant organiser des élections fiables tout en réduisant les coûts organisationnels et les risques de fraude.

c. En quoi notre offre est-elle différente de celle de nos concurrents ?

في ماذا تختلف عروضنا عن تلك التي يقدمها منافسونا؟

notre systeme aura l avantage d être plus flexible pour des utilisation a grande échelle comme des élection gouvernementale

d. Quelles est notre proposition unique de valeur ?

ما هو العرض الفريد للقيمة لدينا؟

Notre proposition de valeur consiste à fournir un système de vote électronique sécurisé, transparent et vérifiable publiquement utilisant la blockchain Ethereum et les preuves ZK-SNARK afin de garantir l'intégrité des élections tout en protégeant l'identité des électeurs.

IV.2.Segments de clients (Customer Segment) انواع العملاء :

e. Quels sont nos clients principaux?

من هم العملاء او الزبائن الرئيسيون ؟

Les principaux clients ciblés par leprojet sont :
-institutionsgouvernementales
-universités
-entreprises privées
-organisation

f. Quels sont les différents segments de clients que nous visons ?

ما هي الفئات المختلفة من العملاء التي تستهدفها؟

Nous visons les universités, les associations, les entreprises, les clubs, les syndicats et les institutions publiques ou privées. Ces clients ont besoin d'organiser régulièrement des élections, des votes internes ou des prises de décision collectives.

g. Quels sont les besoins spécifiques de chaque segment de clients?
ما هي الاحتياجات الخاصة لكل فئة من العملاء؟

Les institutions publiques recherchent principalement la transparence et l'intégrité des résultats électoraux.

Les universités et associations recherchent des solutions simples et peu coûteuses pour organiser des élections internes.

Les entreprises privées peuvent utiliser le système pour les votes administratifs, les assemblées générales ou les processus décisionnels sécurisés.

h. Comment pouvons-nous catégoriser nos clients en groupes distincts?
كيف يمكن تصنيف عملائنا الى مجموعات مختلفة؟

Nos clients peuvent être classés selon leur taille et leur usage. Les petites organisations comme les clubs et associations peuvent utiliser une formule simple. Les universités et entreprises peuvent utiliser une formule professionnelle pour plusieurs élections. Les grandes institutions peuvent choisir une formule personnalisée avec plus de support et de sécurité

IV.3.Relation avec les clients (Consumer Relationships) علاقة مع العملاء :

i. Quel type de relation chaque segment de clients attend il de nous ?
اي نوع من العلاقة يتوقعه كل فئة من العملاء منا؟

Les clients attendent une relation basée sur la confiance, l'accompagnement et la disponibilité. Les petites organisations ont besoin d'une plateforme simple et facile à utiliser. Les universités et entreprises attendent un support technique, une formation et une assistance pendant les élections

j. Comment entretenons-nous actuellement les relations avec nos clients ?
كيف نحافظ حاليًا على العلاقات مع عملائنا؟

Les relations avec les clients peuvent être entretenues à travers un support en ligne, une documentation claire, des démonstrations, des formations et un accompagnement pendant la création des élections. La plateforme peut aussi proposer un suivi après chaque élection avec des rapports et une assistance technique

k. Comment pouvons-nous améliorer ou personnaliser nos interactions avec nos clients ?
كيف يمكننا تحسين أو تخصيص تفاعلاتنا مع عملائنا؟

Les clients souhaitent principalement accéder au système à travers des plateformes numériques simples, accessibles et sécurisées. Les applications web constituent le principal canal de distribution du projet puisqu'elles permettent aux électeurs et aux administrateurs d'interagir facilement avec le système de vote. Les utilisateurs privilégient également des interfaces accessibles depuis différents appareils connectés à Internet afin de faciliter la participation aux élections.

IV.4.Canaux de distribution (Channels) قنوات التوزيع :

a- Par quels canaux nos clients veulent-ils être atteints ?
من خلال أي قنوات يفضل عملاؤنا أن يتم التواصل معهم؟

Les clients souhaitent principalement accéder au système à travers des plateformes numériques simples, accessibles et sécurisées. Les applications web constituent le principal canal de distribution du projet puisqu'elles permettent aux électeurs et aux administrateurs d'interagir facilement avec le système de vote. Les utilisateurs privilégient également des interfaces accessibles depuis différents appareils connectés à Internet afin de faciliter la participation aux élections.

b- Quels canaux sont les plus efficaces pour atteindre chaque segment de clients ?
ما هي القنوات الأكثر فعالية للوصول إلى كل فئة من العملاء؟

Les plateformes web représentent le moyen le plus efficace pour atteindre les universités, les associations et les organisations souhaitant organiser des élections numériques. Les administrations et institutions publiques privilégient généralement des interfaces sécurisées accompagnées d'outils de gestion et de supervision. Les entreprises privées peuvent également utiliser des solutions cloud permettant un accès rapide et flexible au système de vote électronique.

c- Comment pouvons-nous intégrer différents canaux pour améliorer l'expérience clients ?
كيف يمكننا دمج مختلف القنوات لتحسين تجربة العملاء؟

L'intégration de plusieurs canaux numériques permet d'améliorer considérablement l'expérience des utilisateurs en offrant une plateforme plus accessible, flexible et intuitive. Le frontend développé avec React permet aux électeurs d'interagir facilement avec le système, tandis que le backend Node.js assure la gestion des données et des communications avec la blockchain Ethereum. Cette architecture permet de connecter les services web, les APIs sécurisées et les smart contracts afin de fournir une expérience utilisateur cohérente et performante.

IV.5.Partenaires clés (Key Partnerships) : الشراكة الرئيسية

a. Qui sont nos partenaires clés ?
من هم شركاؤنا الرئيسيون؟

Les principaux partenaires du projet incluent les plateformes blockchain, les communautés Ethereum, les fournisseurs de services cloud ainsi que les laboratoires de recherche spécialisés en cybersécurité et en cryptographie. Les universités et les communautés open-source jouent également un rôle important dans le développement et l'amélioration des technologies utilisées dans le système de vote électronique.

b. Quels sont les partenariats qui nous aident à réduire les coûts, à accéder à de nouvelles ressources ou à améliorer notre proposition de valeur ?
ما هي الشراكات التي تساعدنا على خفض التكاليف أو الوصول إلى موارد جديدة أو تحسين قيمتنا المقترحة؟

Les partenariats avec les communautés open-source et les plateformes blockchain permettent de réduire les coûts de développement en donnant accès à des outils et bibliothèques déjà existants tels que Hardhat, Circom, snarkjs et Ethers.js. Les collaborations académiques et technologiques permettent également d'améliorer la sécurité du système et de bénéficier de nouvelles ressources techniques et scientifiques.

c. Comment pouvons-nous aligner nos intérêts avec ceux de nos partenaires ?
كيف يمكننا مزامنة مصالحنا مع تلك لشركائنا؟

Nous pouvons aligner nos intérêts avec ceux de nos partenaires grâce à des collaborations technologiques et académiques favorisant l'innovation et l'amélioration continue du système. Le partage des connaissances, la participation aux communautés blockchain et les projets de recherche communs permettent de renforcer la qualité technique de la plateforme tout en développant un écosystème basé sur la transparence et la sécurité.

IV.6.Activités clés (Key Activities) :الأنشطة الرئيسية

a. Quelles sont les actions principales que nous devons entreprendre pour livrer notre proposition de valeur ?

Les principales activités du projet concernent le développement des smart contracts Solidity, la création des circuits ZK-SNARK avec Circom ainsi que l'intégration entre le frontend, le backend et la blockchain Ethereum. Le système nécessite également la mise en place de mécanismes cryptographiques sécurisés permettant de vérifier les preuves de vote tout en protégeant l'identité des électeurs.

b. Quelles sont les opérations essentielles pour notre entreprise ?
ما هي العمليات الأساسية لشركتنا؟

Les opérations essentielles incluent le déploiement des smart contracts sur Ethereum Sepolia, la gestion des électeurs autorisés, la génération des preuves cryptographiques ainsi que la vérification des votes sur la blockchain. Le maintien de la sécurité de la plateforme et la surveillance des performances du système représentent également des opérations importantes pour assurer la fiabilité du service.

c. Quelles sont les activités qui créent le plus de valeur pour nos clients ?
ما هي الأنشطة التي تخلق أكبر قيمة لعملائنا؟

Les activités qui créent le plus de valeur pour les clients concernent principalement la sécurisation des élections, la prévention du double vote et la transparence des résultats. L'utilisation des preuves ZK-SNARK et des smart contracts permet d'offrir une plateforme capable de garantir l'intégrité des votes tout en protégeant l'identité des électeurs et en assurant une vérification publique des résultats électoraux.

IV.7.Ressources clés (Key resources) : الموارد الرئيسية

d. Quels sont nos actifs matériels, immatériels et humains essentiels ?
ما هي الأصول المادية وغير المادية والبشرية الأساسية لدينا؟

Les ressources essentielles du projet comprennent à la fois des ressources techniques, humaines et logicielles. Les principaux actifs technologiques incluent les smart contracts Solidity, les circuits Circom, l'infrastructure blockchain Ethereum ainsi que les serveurs backend utilisés pour la gestion des données hors blockchain. Le projet repose également sur des compétences spécialisées en développement blockchain, en cybersécurité et en cryptographie appliquée aux systèmes distribués. Les connaissances liées aux preuves ZK-SNARK, aux arbres de Merkle et aux mécanismes de consensus représentent également des ressources immatérielles importantes pour le bon fonctionnement du système.

e. Quels sont les outils, les technologies ou les partenariats dont nous avons besoin pour réussir ?

Le système nécessite plusieurs outils et technologies afin d'assurer son développement et son déploiement. Le projet utilise principalement Ethereum Sepolia comme environnement blockchain, Solidity pour les smart contracts, Hardhat pour le développement et les tests, ainsi que Ethers.js pour les interactions avec la blockchain. Les circuits cryptographiques sont développés avec Circom tandis que snarkjs et Groth16 sont utilisés pour la génération et la vérification des preuves ZK-SNARK. Le frontend est développé avec React et le backend repose sur Node.js, Express et MongoDB. Les partenariats avec les communautés open-source et les plateformes blockchain contribuent également à l'amélioration continue du système.

f. Quels sont les principaux avantages concurrentiels de nos ressources ?

ما هي المزايا التنافسية الرئيسية لمواردنا؟

Les ressources utilisées dans le projet offrent plusieurs avantages concurrentiels importants. L'intégration de la blockchain Ethereum garantit une forte transparence et une immutabilité des résultats électoraux. Les preuves ZK-SNARK permettent quant à elles de protéger l'identité des électeurs tout en maintenant une vérification publique des votes. L'utilisation des arbres de Merkle et des fonctions de hachage Poseidon améliore également l'efficacité cryptographique du système. Enfin, l'architecture hybride combinant blockchain et services off-chain permet de construire une plateforme moderne, flexible et sécurisée.

IV.8.Charges et coûts (Coste sructure) :التكاليف

g. Quels sont les coûts fixes et variables associés à notre modèle économique ?

ما هي التكاليف الثابتة والمتغيرة المرتبطة بنموذجنا الاقتصادي؟

Le développement et le maintien du système de vote électronique nécessitent plusieurs types de coûts par la dure d utilisation et variables. Les coûts concernent principalement l'infrastructure technique, l'hébergement des serveurs backend, le développement des smart contracts et la maintenance de la plateforme. Les coûts variables incluent les frais de transaction Ethereum, les dépenses liées à l'utilisation des ressources cloud ainsi que les coûts associés aux opérations de sécurité et de surveillance du système. Les opérations cryptographiques et les vérifications on-chain peuvent également générer des coûts supplémentaires liés au gas Ethereum.

Pour par exemple Basic : 20 000 DZD/mois ou Professional : 50 000 DZD/mois

h. Quels sont les coûts les plus importants pour notre entreprise ?

ما هي التكاليف الأكثر أهمية لشركتنا؟

Les coûts les plus importants concernent principalement le développement blockchain et la sécurisation du système. La création des smart contracts, l'implémentation des circuits ZK-SNARK et l'intégration des mécanismes cryptographiques nécessitent des compétences techniques avancées et un temps de développement important. Les frais de transaction Ethereum et les opérations de vérification des preuves cryptographiques représentent également des coûts significatifs, particulièrement dans les systèmes nécessitant de nombreuses opérations on-chain.

i. Comment pouvons-nous réduire les coûts ou améliorer l'efficacité de nos opérations ?

L'utilisation d'outils open-source comme Hardhat, Circom et snarkjs permet de réduire considérablement les coûts de développement. L'optimisation des smart contracts Solidity et des circuits cryptographiques contribue également à diminuer les frais de gas Ethereum et à améliorer les performances globales du système. De plus, l'automatisation de certaines opérations techniques ainsi que l'utilisation d'une architecture hybride combinant blockchain et stockage off-chain permettent d'améliorer l'efficacité tout en limitant les dépenses liées aux transactions blockchain

IV.9.Revenus (Revenue) : مصادر الدخل

j. Quels produits ou services nos clients sont-ils prêts à payer ?

ما هي المنتجات أو الخدمات التي يكون عملاؤنا على استعداد لدفع ثمنها؟

Les clients peuvent être intéressés par plusieurs services liés au système de vote électronique sécurisé. Les organisations souhaitant organiser des élections numériques peuvent payer pour l'utilisation de la plateforme, les services de déploiement et de configuration des élections, ainsi que l'assistance technique et la maintenance du système. Des services supplémentaires comme les audits de sécurité, les intégrations blockchain personnalisées et les solutions adaptées aux besoins spécifiques des institutions peuvent également représenter des sources de revenus importantes.

k. Quels sont les différents moyens par lesquels nous pouvons générer des revenus ?

ما هي الطرق المختلفة التي يمكننا من خلالها تحقيق الدخل؟

Les revenus peuvent être générés à travers plusieurs modèles économiques. Le système peut être proposé sous forme d'abonnement permettant aux organisations d'utiliser régulièrement la plateforme de vote. Il est également possible de facturer les services selon le nombre d'élections organisées ou selon le nombre d'électeurs participants. Les services de maintenance, de support technique, d'audit de sécurité et de personnalisation de la plateforme peuvent également constituer des sources de revenus complémentaires

1. Quel est notre modèle de tarification ?

ما هو نموذج التسعير لدينا؟

Le modèle de tarification peut être adapté selon les besoins des clients et la taille des organisations utilisant le système. Une approche basée sur un abonnement mensuel ou annuel peut être utilisée pour les institutions nécessitant un accès permanent à la plateforme. Il est également possible d'utiliser un modèle de paiement par élection ou par nombre d'électeurs. Les services avancés comme les audits de sécurité,

Business Model Canevas : BMC

Partenaires clés
Key Partnerships
الشراكة الرئيسية

les entreprises technologiques, les prestataires de paiement en ligne et les organisations qui utilisent ou recommandent la plateforme

Activités clés
Key Activities
الأنشطة الرئيسية

Développement et maintenance de la plateforme , gestion des élections, support client, sécurité, mises à jour et amélioration continue

Ressources clés
Key resources
الموارد الرئيسية

- API backend et base de données
- Plateforme web
- Serveurs, hébergement est outils de déploiement

Proposition de valeur
Value Proposition
القيمة المقترحة

Une plateforme de vote électronique sécurisé permettant aux organisations de créer, gérer et suivre leurs élections en ligne sans développer leur propre système

Relation clients
Consumer Relationship
علاقة مع العملاء

La relation avec les clients repose sur l'accompagnement, la formation, le support technique, la documentation et la maintenance continue de leur plateforme de vote.

Canaux de distribution
Channels
قنوات التوزيع

-site web officiel

-reseaux sociaux

-- démonstrations dans les universités

Segment client
Customer Segment
انواع العملاء

Nos clients sont les organisations qui ont besoin d'organiser régulièrement des élections ou des votes internes

Coûts
Coste sructure
التكاليف

Les coûts principaux concernent l'hébergement cloud, la maintenance, la sécurité, le support technique, les mises à jour et le développement de nouvelles fonctionnalités.

Revenus
Revenue
مصادر الدخل

Les revenus proviennent principalement des abonnements mensuels ou annuels, complétés par des frais selon le nombre d'élections, le nombre d'électeurs, la personnalisation et le support par exemple ; - Basic : 20 000 DZD/mois ou Professional : 50 000 DZD/mois

Figure.IV.1 : Business Model Canvas du projet zk-voting

Conclusion générale

Ce mémoire a présenté la conception et la réalisation de zk-voting, une plateforme de vote électronique utilisant la blockchain et les preuves à divulgation nulle de connaissance. Le travail répond à la problématique de la vérification de l'éligibilité, de l'intégrité des résultats, de la prévention du double vote ainsi que la vérification du vote individuel et global par l'électeur lui même

L'étude théorique a montré que la blockchain apporte un registre partagé et une exécution déterministe, tandis que les zk-SNARKs permettent de prouver une appartenance sans révéler les informations privées. La combinaison de ces mécanismes a été appliquée à une architecture comprenant React, Express, MongoDB, Circom, SnarkJS, Solidity et Hardhat.

Le circuit calcule un engagement Poseidon à partir du secret et du nullifier, vérifie son appartenance à un arbre de Merkle et impose la dérivation correcte du nullifier hash. Le contrat vérifie la preuve Groth16, contrôle l'unicité et met à jour le résultat du candidat.

comme perspective, on peut ajouter d'autres fonctions au système tel que la possibilité de refaire le vote d'un électeur en cas de problème technique ou la possibilité à l'électeur d'annuler son vote tant que le scrutin n'est pas cloturé

Références

[1] Benaloh, J., Rivest, R., Ryan, P. Y. A., Stark, P., Teague, V., & Vora, P., *End-to-end verifiability*, 2015.

[2] Ali, S. T., & Murray, J., *An Overview of End-to-End Verifiable Voting Systems*, 2016.

[3] Schneider, A., Meter, C., & Hagemeister, P., *Survey on Remote Electronic Voting*, 2017.

[4] Chaum, D., *Secret-Ballot Receipts: True Voter-Verifiable Elections*, 2004.

[5] Nakamoto, S., *Bitcoin: A Peer-to-Peer Electronic Cash System*, 2008.

[6] Buterin, V., *Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform*, 2014.

[7] Ethereum Foundation, *Introduction to Smart Contracts*, 2026.

[8] Merkle, R. C., *A Digital Signature Based on a Conventional Encryption Function*, 1987.

[9] Goldwasser, S., Micali, S., & Rackoff, C., *The Knowledge Complexity of Interactive Proof Systems*, 1985.

[10] Groth, J., *On the Size of Pairing-Based Non-interactive Arguments*, 2016.

[11] Clarke, D., & Martens, T., *E-voting in Estonia*, 2016.

[12] Zhang, B., Li, Z., & Willemson, J., *UC Modelling and Security Analysis of the Estonian IVXV Internet Voting System*, 2021.

[13] Karlof, C., Sastry, N., & Wagner, D., *Cryptographic Voting Protocols: A Systems Perspective*, 2005.

[14] Wood, G., *Ethereum: A Secure Decentralised Generalised Transaction Ledger*, 2014.

[15] Szabo, N., *Smart Contracts: Building Blocks for Digital Markets*, 1996.

Résumé

Le vote électronique, bien qu'il facilite l'accès au scrutin et accélère le dépouillement, soulève des défis majeurs en matière de sécurité, d'anonymat, d'intégrité et de prévention du double vote. Ce mémoire propose un système sécurisé combinant blockchain, smart contracts et preuves à divulgation nulle de connaissance (ZK-SNARKs) pour garantir la transparence, la traçabilité et l'automatisation des règles électorales, tout en permettant aux électeurs de prouver leur éligibilité sans révéler leur identité grâce à un arbre de Merkle, ainsi que la vérification du vote individuel et global. L'objectif est d'assurer l'unicité du vote, l'anonymat, la vérifiabilité et la transparence du dépouillement, renforçant ainsi la confiance dans les systèmes de vote numérique.

Mots-clés : Vote électronique, Blockchain, Smart contract, Zero-Knowledge Proof, zk-SNARK, Arbre de Merkle, Anonymat, Sécurité, Vérifiabilité.

Abstract

Electronic voting, although it facilitates access to elections and accelerates vote counting, raises major challenges in terms of security, anonymity, integrity, and the prevention of double voting. This thesis proposes a secure system combining blockchain, smart contracts, and zero-knowledge proofs (ZK-SNARKs) to ensure transparency, traceability, and the automation of electoral rules, while allowing voters to prove their eligibility without revealing their identity through a Merkle tree, as well as enabling both individual and global vote verification. The objective is to ensure vote uniqueness, anonymity, verifiability, and transparency in the counting process, thereby strengthening trust in digital voting systems.

Keywords: Electronic Voting, Blockchain, Smart Contract, Zero-Knowledge Proof, zk-SNARK, Merkle Tree, Anonymity, Security, Verifiability..

الملخص

إن التصويت الإلكتروني، رغم أنه يسهل الوصول إلى العملية الانتخابية ويسرّع عملية فرز الأصوات، يطرح تحديات كبيرة تتعلق بالأمن، والخصوصية، وسلامة البيانات، ومنع التصويت المزدوج. يقترح هذا البحث نظامًا آمنًا بهدف ضمان الشفافية، (ZK-SNARKs)، يجمع بين تقنية البلوك تشين، والعقود الذكية، وإثباتات المعرفة الصفرية وإمكانية التتبع، وأتمتة القواعد الانتخابية، مع تمكين الناخبين من إثبات أهليتهم للتصويت دون الكشف عن هويتهم، وذلك باستخدام شجرة ميركل، إضافة إلى إتاحة التحقق من التصويت على المستويين الفردي والعام. يهدف هذا النظام إلى ضمان تفرد التصويت، والحفاظ على السرية، وتحقيق قابلية التحقق وشفافية عملية الفرز، مما يعزز الثقة في أنظمة التصويت الرقمية. الكلمات المفتاحية: التصويت الإلكتروني، البلوك تشين، العقد الذكي، إثبات شجرة ميركل، الخصوصية، الأمن، قابلية التحقق ،zk-SNARK، المعرفة الصفرية.