

REPUBLIQUE DU CAMEROUN

Paix – Travail – Patrie

UNIVERSITE DE DOUALA

ECOLE NATIONALE SUPERIEURE
POLYTECHNIQUE DE DOUALA

B.P. 2701 Douala

Tél. (237) 697 542 240

Site web: www.enspd-udo.cm



REPUBLIC OF CAMEROON

Peace – Work – Fatherland

THE UNIVERSITY OF DOUALA

NATIONAL HIGHER POLYTECHNIC
SCHOOL OF DOUALA

P.O. Box :2701 Douala

Phone :(237) 697 542 240

Email : contact@enspd-udo.cm



INGENIERIE LOGICIEL

**THÈME : PLATEFORME DE COMPTAGE DE VOTE LORS D'UNE
ELECTION : VOTECOUNT**

REDIGER PAR:

NOMS	MATRICULE
FOUEJIO NGUEFACK NILIMA JODIE	24G01092
NONGNING LELE STEVE JORDAN	24G01122
TCHAGNOUO NKAMDEM JOVILE BERTHOL	24G01131
TEZEM YEMETIO TEDDY MIGUEL	24G01133
TIEMOUO DJIOLIO DORCAS	24G01134

SUPERVISER PAR :

DR IHONOCK

**ANNEE ACADEMIQUE :
2024/2025**

REMERCIEMENTS

Ce travail n'aurait pu être réalisé sans l'aide, les conseils très précieux et les encouragements de nombreuses personnes. Nos remerciements vont à l'endroit de :

Monsieur le directeur de l'école national supérieur polytechnique M MOUANGUE Ruben Martin pour les efforts qu'il ne cesse de déployer et les conseils qu'il ne cesse de prodiguer

DR IHONOCK professeur d'ingénierie logiciel pour son enseignement de qualité sans lequel nous ne pouvions réaliser ce travail

A tous nos camarades de la promotion GIT4 pour leurs aides multiformes

Sommaire

REMERCIEMENTS	2
Sommaire	3
LISTE DES TABLEAUX.....	4
LISTE DES FIGURES.....	5
RESUME	6
ABSTRACT.....	7
INTRODUCTION GENERAL.....	8
Partie 1 : ETUDE DE FAISABILITE.....	9
A- Analyse SWOT.....	9
B- HIERARCHISATION DES BESOINS	12
1- DEFINITION	12
2- APPLICATION AVEC L'ANALYSE MOSCOW	13
Partie 2 : PHASE TECHNIQUE.....	20
A- CAHIER DE CHARGE.....	20
I- CONTEXTE	20
II- OBJECTIFS DU PROJET	21
IV- EXPRESSION DES BESOINS	22
V- ESTIMATION DU COÛT DU PROJET	23
VI- Planification du projet.....	27
VII- contraintes	27
VIII- livrables	28
Chapitre 2 : DOSSIER D'ANALYSE.....	28
I- Etude et critique de l'existant	29
II- Problématique	30
IV- Choix Et Présentation De La Démarche D'analyse	30
V- PRESENTATION DES DIAGRAMMES INTERVENANTS	36
• Présentation	42
CONCLUSION GENERAL	45
TABLE DES MATIERES	47

LISTE DES TABLEAUX

Tableau 1: Ressources humaines	23
Tableau 2: Ressources materielles.....	25
Tableau 3: cout total	26
Tableau 4:Formalisme diagramme de UseCase	37
Tableau 5: formalisme du sequence diagram	39

LISTE DES FIGURES

Figure 1: Diagramme de use case.....	38
Figure 2: Diagramme de sequence pour connexion	41
Figure 3:Diagramme de sequence processus de vote	42

RESUME

Le présent rapport rend compte de la réalisation du projet VOTE COUNT, dédié à la conception et à la mise en œuvre d'une plateforme de comptage de votes. Ce projet a été mené dans le but de proposer une solution informatique capable d'optimiser le processus de dépouillement, en améliorant la précision des résultats, la rapidité de traitement et la transparence des opérations. La démarche adoptée commence par l'analyse des systèmes existants de gestion des votes, mettant en évidence leurs insuffisances, notamment en matière de fiabilité, de sécurité et de centralisation des données. À partir de cette étude, les exigences fonctionnelles et non fonctionnelles du système ont été définies. La phase de conception repose sur la modélisation du système à l'aide de diagrammes UML, permettant de structurer l'architecture et les interactions entre les différents acteurs. La réalisation de l'application s'appuie sur des technologies adaptées et est suivie d'une phase de tests visant à valider le bon fonctionnement du système. Le rapport présente également une estimation du budget nécessaire à la mise en œuvre du projet. En conclusion, une analyse critique des résultats obtenus est proposée, accompagnée des limites du système et des perspectives d'évolution. Ce projet académique illustre l'application pratique des connaissances théoriques et met en évidence l'apport des solutions numériques dans l'amélioration des processus électoraux.

ABSTRACT

This report presents the realization of the VOTE COUNT project, which focuses on the design and implementation of a vote-counting platform. The objective of this project is to propose an information system capable of optimizing the vote-counting process by improving result accuracy, processing speed, and operational transparency. The adopted approach begins with an analysis of existing vote management systems, highlighting their limitations in terms of reliability, security, and data centralization. Based on this study, the functional and non-functional requirements of the system were defined. The design phase relies on system modeling using UML diagrams, which help structure the architecture and describe interactions between the various actors involved. The implementation of the application is based on appropriate technologies and is followed by a testing phase aimed at validating the system's proper functioning. The report also includes an estimation of the budget required for the project's implementation. Finally, a critical analysis of the results obtained is presented, along with the system's limitations and perspectives for future improvements. This academic project demonstrates the practical application of theoretical knowledge and highlights the contribution of digital solutions to the improvement of electoral processes.

INTRODUCTION GENERAL

Dans un monde où la transparence et la fiabilité des processus électoraux sont devenues des enjeux cruciaux, la modernisation du vote s'impose comme une nécessité. Les méthodes traditionnelles de comptage manuel des votes, bien qu'ayant longtemps servi à organiser les élections, présentent des limites notables : elles sont lentes, exposées aux erreurs humaines et difficiles à auditer de manière efficace. Ces contraintes peuvent compromettre la crédibilité des scrutins et engendrer des contestations qui affectent la confiance des citoyens dans les institutions. C'est dans ce contexte que ce projet académique s'inscrit, visant la conception et le développement d'une plateforme de comptage automatique des votes. Cette application a pour objectif de traiter et de compiler les votes de manière fiable et sécurisée, tout en affichant les résultats en temps réel, offrant ainsi une transparence immédiate pour les acteurs électoraux et les observateurs. Elle constitue également un outil pédagogique qui permet de mettre en pratique des compétences en développement logiciel, gestion de bases de données et sécurité informatique. Le présent rapport décrit les différentes étapes du projet, depuis l'analyse des besoins et la conception jusqu'au développement, aux tests et à l'évaluation des performances de l'application. Une attention particulière est portée sur la précision du comptage, la rapidité de traitement des données et la convivialité de l'interface utilisateur, afin de proposer un outil complet et fonctionnel. L'application développée se veut une solution moderne et efficace pour accompagner le processus électoral, tout en illustrant concrètement l'application de concepts théoriques enseignés dans le cadre académique. Enfin, ce projet permet également de réfléchir aux enjeux de sécurité, de confidentialité et de fiabilité des données dans les systèmes de vote électroniques, qui sont essentiels pour garantir la légitimité des résultats et renforcer la confiance des citoyens dans le processus démocratique.

Partie 1 : ETUDE DE FAISABILITE

A- Analyse SWOT

L'analyse SWOT est un outil stratégique utilisé pour identifier les forces, faiblesses, opportunités et menaces d'un projet, d'une entreprise ou d'une initiative. Elle permet de :

- Forces (Strengths) : Identifier les atouts internes qui peuvent être exploités pour atteindre les objectifs.
- Faiblesses (Weaknesses) : Reconnaître les limitations internes qui pourraient freiner la réussite.
- Opportunités (Opportunities) : Identifier les éléments externes favorables qui peuvent être utilisés pour le développement.
- Menaces (Threats) : Déceler les facteurs externes qui pourraient nuire au succès du projet.

L'analyse SWOT aide à élaborer des stratégies en capitalisant sur les forces et les opportunités tout en atténuant les faiblesses et les menaces.

ANALYSE SWOT	
INTERNES	<u>FORCES (STENGTHS)</u>
	<p>-Automatisation du comptage des votes : L'application permet de réduire les erreurs humaines liées au dépouillement manuel et d'améliorer la fiabilité des résultats.</p> <p>- Rapidité de traitement des résultats : Le système accélère la consolidation et la publication des résultats, limitant ainsi les délais souvent observés dans les processus traditionnels.</p> <p>- Centralisation et traçabilité des données : Les résultats sont regroupés dans une base de données centralisée, facilitant la vérification,</p>

l'audit et la transparence du processus électoral.

- Adaptabilité aux contextes locaux : La conception modulaire du système permet son ajustement aux réalités institutionnelles et organisationnelles propres aux pays africains.

FAIBLESSES (WEAKNESSES)

- Dépendance aux infrastructures technologiques : Le fonctionnement du système nécessite une alimentation électrique stable, un accès à Internet et des équipements informatiques, qui peuvent être insuffisants dans certaines zones rurales.

- Niveau variable de compétences numériques : Certains acteurs électoraux peuvent rencontrer des difficultés d'utilisation de l'application en raison d'un manque de formation ou de maîtrise des outils numériques.

- Coût initial de mise en œuvre : Le déploiement du système, incluant l'achat de matériel et la formation des utilisateurs, peut représenter une contrainte financière pour les administrations à budget limité.

- Risque de pannes techniques : Les défaillances matérielles ou logicielles peuvent perturber le processus de comptage

	si des mécanismes de secours ne sont pas prévus.
EXTERNES	<p><u>OPPORTUNITES</u></p> <ul style="list-style-type: none"> - Dynamique de transformation numérique en Afrique : De nombreux pays africains s'engagent dans la modernisation de leurs systèmes administratifs, ce qui favorise l'adoption de solutions numériques électorales. - Intérêt pour la transparence et la bonne gouvernance : Les institutions publiques et les partenaires internationaux soutiennent les projets visant à renforcer la crédibilité des élections. - Expansion des réseaux mobiles et des technologies numériques : La généralisation progressive des technologies mobiles facilite l'intégration et l'utilisation d'applications informatiques. - Possibilités de partenariats et de financements : Le projet peut bénéficier de l'appui d'organisations internationales, d'ONG ou de programmes de soutien à la démocratie et à l'innovation.
	<p><u>MENACES (THREATS)</u></p> <ul style="list-style-type: none"> - Sensibilité du contexte politique : Les tensions politiques et les contestations électorales peuvent limiter l'acceptation ou

	<p>l'utilisation d'un système de comptage numérique.</p> <ul style="list-style-type: none"> - Risques liés à la cybersécurité : Les tentatives de piratage, de fraude ou de manipulation des données représentent une menace pour l'intégrité du système. - Contraintes juridiques et réglementaires : L'absence ou l'évolution des cadres légaux peut restreindre l'utilisation des technologies numériques dans les processus électoraux. - Fracture numérique persistante : Les inégalités d'accès aux technologies entre zones urbaines et rurales peuvent affecter l'équité et l'efficacité du système
--	--

B- HIERARCHISATION DES BESOINS

1- DEFINITION

La hiérarchisation des besoins d'un projet est un processus qui consiste à classer et prioriser les besoins identifiés en fonction de leur importance, de leur urgence et de leur faisabilité. Cela permet de s'assurer que les ressources (temps, budget, personnel) sont allouées de manière efficace pour répondre aux besoins les plus critiques et d'atteindre les objectifs du projet.

i- Objectifs de la Hiérarchisation

- Clarté : Aider les équipes à comprendre les priorités et à se concentrer sur les éléments essentiels du projet.
- Efficacité : Maximiser l'utilisation des ressources en ciblant les besoins qui auront le plus grand impact.

- Gestion des attentes : Communiquer clairement aux parties prenantes quelles fonctionnalités ou besoins seront traités en premier.

ii- Étapes de la Hiérarchisation des Besoins

- Identification des Besoins : Recueillir toutes les exigences et attentes des utilisateurs et des parties prenantes.
- Évaluation des Besoins : Analyser chaque besoin en termes d'importance, d'urgence et de faisabilité.
- Classement : Classer les besoins selon les critères définis, souvent à l'aide de méthodes comme la matrice de priorisation ou le diagramme de Kano.
- Validation : Soumettre la hiérarchisation aux parties prenantes pour validation et ajustements nécessaires.

iii- Méthodes de Hiérarchisation

- Matrice de Priorisation : Évalue les besoins sur plusieurs critères et les classe en fonction de leurs scores.
- Diagramme de Kano : Classe les besoins en fonction de leur impact sur la satisfaction des utilisateurs (besoins essentiels, performants, et indifférents).
- Analyse MoSCoW : Classe les besoins en quatre catégories : Must have (indispensables), Should have (souhaitables), Could have (facultatifs), Won't have (non prioritaires).

Pour notre projet **conception et réalisation d'une application de comptage de vote**, nous allons utiliser l'analyse de MOSCOW

2- APPLICATION AVEC L'ANALYSE MOSCOW

L'analyse MoSCoW est une méthode populaire pour classer les besoins d'un projet en quatre catégories : **Must have**, **Should have**, **Could have**, et **Won't have**.

i- Catégories de l'Analyse MoSCoW

- **Must have (Indispensables)** : Ces besoins sont critiques pour le succès du projet. Sans eux, la plateforme ne pourra pas fonctionner correctement.
- **Should have (Souhaitables)** : Ces besoins sont importants, mais la plateforme peut fonctionner sans les fonctionnalités associées. Leur absence aurait un impact modéré sur l'expérience utilisateur.
- **Could have (Facultatifs)** : Ces besoins sont des améliorations qui peuvent être mises en œuvre si les ressources le permettent. Ils ajoutent de la valeur, mais ne sont pas essentiels.
- **Won't have (Non prioritaires)** : Ces besoins ne seront pas inclus dans la version actuelle du projet. Ils peuvent être considérés pour des versions futures.

ii- Identification des besoins

1. Authentification des électeurs : Garantir que seuls les électeurs autorisés puissent accéder au système de vote.

Fonctionnalités :

- Connexion sécurisée (identifiant + mot de passe ou code secret)
- Vérification de l'identité avant accès au vote
- Interface conviviale pour naviguer.

2. Garantie du vote unique : Empêcher toute tentative de vote multiple afin d'assurer l'équité du scrutin.

Fonctionnalités :

- Vérification automatique de l'état de vote de l'électeur
- Blocage du système après validation du vote
- Enregistrement horodaté du vote.

3. Comptage automatique des votes : calcul immédiat et fiable des voix exprimées pour chaque candidat ou option.

Fonctionnalités : Enregistrement automatique des choix de vote

- Enregistrement automatique des choix de vote
- Incrémentation instantanée des voix par candidat
- Calcul automatique des totaux

4. Affichage des résultats en temps réel : mise à jour instantanée des résultats dès qu'un vote est enregistré.

Fonctionnalités :

- Mise à jour dynamique des résultats
- Tableau de bord de suivi des votes
- Affichage du nombre de votants et des scores en direct

5. Sécurisation des données : protection des votes et des informations sensibles contre toute altération ou accès non autorisé.

Fonctionnalités :

- Chiffrement des données de vote
- Protection de la base de données
- Options de personnalisation (thèmes, polices, etc.).

6. Gestion des rôles utilisateurs : organisation claire des responsabilités et des droits d'accès.

Fonctionnalités :

- Création de profils (administrateur, électeur, Responsable région)
- Attribution des droits d'accès
- Interfaces spécifiques selon le rôle

7. Statistiques électorales : affichage des pourcentages, taux de participation et nombre total de votants.

Fonctionnalités :

- Calcul automatique des pourcentages

- Affichage du taux de participation
- Synthèse statistique des résultats.

8. Historique et traçabilité des opérations : conservation des actions et résultats pour audit et vérification.

Fonctionnalités :

- Journalisation des actions (connexion, vote, administration)
- Historique des opérations
- Consultation des logs par l'administrateur.

9. Ergonomie et facilité d'utilisation : faciliter l'utilisation du système par tous les profils d'utilisateurs.

Fonctionnalités :

- Interface graphique claire
- Navigation simplifiée
- Messages de confirmation et d'erreur explicites

10. Exportation des résultats : faciliter l'analyse et l'archivage des données.

Fonctionnalités :

- Génération de fichiers PDF
- Export des données en Excel ou CSV
- Téléchargement des rapports.

11. Visualisation avancée : Permet de suivre la participation des étudiants et des enseignants aux cours, aux sessions en direct et aux activités de la plateforme.

Fonctionnalités :

- Graphiques de résultats
- Diagrammes comparatifs

- Tableaux interactifs

12. Vote biométrique :

Fonctionnalités :

- Reconnaissance d'empreintes
- Scan facial ou iris

13. Déploiement à l'échelle nationale

Fonctionnalités :

- Interconnexion multi-serveurs nationaux
- Synchronisation avec des bases institutionnelles

L'absence de cette fonctionnalité entraînerait une falsification des résultats. Elle constitue donc un élément fondamental du système de comptage automatique.

BESOINS		CATEGORIE	JUSTIFICATION
Authentification des électeurs	des	Must Have	Conditionne l'accès au système de vote. Sans mécanisme

		d'authentification, il est impossible de garantir que seuls les électeurs autorisés participent au scrutin, ce qui remettrait en cause la légitimité des résultats.
Garantie du vote unique	Must Have	Permet d'assurer l'équité électorale. L'absence de cette fonctionnalité entraînerait une falsification des résultats. Elle constitue donc un élément fondamental du système de comptage automatique.
Comptage automatique des votes	Must Have	Représente l'objectif principal du projet. Elle est indispensable au fonctionnement du système.
Affichage des résultats en temps réel	Must Have	Il garantit la transparence et la rapidité dans la diffusion des résultats
Sécurisation des données	Must Have	La protection des données de vote est critique afin d'éviter toute manipulation, perte ou accès non autorisé.
Gestion des rôles utilisateurs	Should Have	Améliore l'organisation et la sécurité du système. Bien qu'importante pour une bonne administration, l'application peut fonctionner de manière

		basique sans cette fonctionnalité.
Statistiques électorales	Should Have	Enrichissent l'analyse des résultats et facilitent leur interprétation. Cependant, elles ne sont pas strictement nécessaires au comptage et à l'affichage des votes, ce qui explique leur priorité secondaire.
Historique et traçabilité des opérations	Should Have	Permet de contrôler et vérifier le déroulement du scrutin. Bien qu'elle renforce la fiabilité et la confiance, son absence n'empêche pas le fonctionnement principal du système.
Ergonomie et facilité d'utilisation	Should Have	Améliore l'expérience utilisateur et réduit les erreurs de manipulation. Toutefois, même avec une interface simple, le système peut remplir ses fonctions essentielles
Exportation des résultats	Could Have	Pas nécessaire pour la première version de la plateforme, mais peut être envisagée plus tard.

Visualisation avancée	Could Have	Les graphiques améliorent la lisibilité des résultats mais ne sont pas nécessaires au fonctionnement du système de base. Ils constituent un enrichissement fonctionnel optionnel.
Vote biométrique	Won't Have	La mise en œuvre de technologies biométriques nécessite des équipements spécialisés, des coûts élevés et un cadre légal strict. Ces contraintes dépassent les objectifs et les moyens d'un projet académique.
Déploiement à l'échelle nationale	Won't Have	Ce besoin est volontairement exclu afin de maintenir un périmètre réaliste et pédagogique.

Partie 2 : PHASE TECHNIQUE

A- CAHIER DE CHARGE

Le cahier des charges est un document contractuel établi entre le maître d'œuvre et le maître d'ouvrage qui étale les besoins du client. Il étudie et présente avec exactitude les exigences formulées par les utilisateurs en ce qui concerne le projet, son déroulement et les résultats attendus.

I- CONTEXTE

Le processus électoral constitue un élément fondamental de la démocratie, car il garantit l'expression libre et transparente de la volonté des citoyens. Cependant, dans de nombreux contextes, notamment en Afrique, le comptage des votes repose encore largement sur des

méthodes manuelles, souvent longues, sujettes à des erreurs et sources de contestations. Ces insuffisances affectent la crédibilité des résultats et la confiance des citoyens dans le processus électoral. L'évolution des technologies de l'information offre aujourd'hui des solutions capables de moderniser et de sécuriser les opérations électorales. La mise en place d'un système informatisé de comptage des votes permettrait d'améliorer la rapidité, la fiabilité et la transparence du dépouillement. C'est dans cette perspective que s'inscrit le projet VOTE COUNT, dont la réalisation est justifiée par la nécessité de proposer une solution numérique adaptée aux réalités locales, tout en tenant compte des contraintes techniques, organisationnelles et sécuritaires.

II- OBJECTIFS DU PROJET

Le projet VOTE COUNT vise à répondre aux insuffisances observées dans les méthodes traditionnelles de comptage des votes à travers le développement d'une application informatique fiable et sécurisée.

1- Objectif général

L'objectif général du projet est de concevoir et de développer une plateforme de comptage de votes permettant d'améliorer la fiabilité, la rapidité et la transparence du processus de dépouillement et de centralisation des résultats électoraux.

2- Objectifs spécifiques

- Automatiser le processus de comptage des votes afin de réduire les erreurs humaines.
- Faciliter la centralisation et la consolidation des résultats issus de plusieurs bureaux de vote.
- Assurer la traçabilité et la sécurisation des données électorales.
- Mettre à disposition des acteurs électoraux un outil simple, intuitif et efficace.
- Contribuer à la modernisation du processus électoral par l'usage des technologies numériques.

IV- EXPRESSION DES BESOINS

1- Besoins fonctionnel

Gestion du vote :

- Authentification sécurisée des électeurs (login/mot de passe, code secret ou OTP).
- Création et gestion des scrutins (dates, candidats, règles de vote).
- Sélection unique des candidats ou propositions par électeur.
- Garantie du vote unique pour chaque électeur.

Comptage et affichage :

- Comptage automatique des votes validés.
- Affichage des résultats en temps réel.
- Statistiques détaillées : pourcentage de votes, nombre de votants, taux de participation.
- Historique des votes pour audit ou consultation ultérieure.

Administration :

- Gestion des rôles et droits des utilisateurs (administrateur, superviseur, électeur).
- Export des résultats au format PDF, Excel ou CSV.
- Notifications pour informer les utilisateurs des mises à jour ou rappels.

2- Besoins non fonctionnels

La plate-forme finale devra répondre aux caractéristiques d'un bon logiciel informatique à savoir :

Sécurité :

- Chiffrement des données sensibles (votes et informations personnelles).
- Traçabilité et journalisation des actions pour audit et contrôle.
- Détection et alertes en cas d'anomalies ou tentative de fraude.

Performances :

- Traitement rapide des votes pour permettre l'affichage en temps réel.
- Capacité à gérer un grand nombre d'électeurs simultanément.

Ergonomie et accessibilité :

- Interface utilisateur intuitive et facile à utiliser.
- Compatibilité avec différents appareils (ordinateurs, tablettes, smartphones).

Maintenabilité :

- Code structuré et documenté pour faciliter les mises à jour futures.
- Possibilité d'ajouter de nouvelles fonctionnalités sans perturber le système existant

V- ESTIMATION DU COÛT DU PROJET

Il est toujours judicieux et très prudent de faire un chiffrage du projet avant le lancement de celui-ci. Ce chiffrage ne doit pas être très exhaustif et pas moins non plus car les risques doivent être aussi pris en compte. En sommes le cout du projet revient à la comptabilisation des ressources humaines, matérielles et logiciels.

1- Ressources humaines

Le tableau ci-dessous consacre les ressources humaines nécessaires :

Tableau 1: Ressources humaines

Fonction	Quantité	Description	CU/jour	en Jours	Total en FCFA
FCFA					
Analyste	03		1 50 000	30	4 500 000
Concepteur			FCFA		FCFA
Développeur	04	Il a pour rôle de produire une	40.000FCFA	30	4 800 000
		application sur la base de l'analyse produite par les analystes.			FCFA

CONCEPTION D'UNE PLATEFORME DE COMPTAGE DE VOTES

Testeur	01	Il effectuera les tests unitaires afin de valider la solution obtenue.	30 000 FCFA	8	240 000 FCFA
Designer	01	Il sera chargé du design des différentes interfaces de notre application	45 000 FCFA	15	675 000 FCFA
Total					10 215 000 FCFA

3- Ressources logicielles

Tableau 3:ressources logicielles

Logiciel	Fonction	Prix (FCFA)
Microsoft Windows 11 Pro	Système d'exploitation	165 000 FCFA
Anti virus AVG	Protection de la machine	2000 FCFA
Angular JS	Permet de mettre en place un serveur web local	Gratuit
Sybase Power AMC	Atelier de génie logiciel permettant la modélisation des applications	Gratuit
GANTT Project	Logiciel de planification	Gratuit

CONCEPTION D'UNE PLATEFORME DE COMPTAGE DE VOTES

SpringBoot	SpringBoot est un environnement de développement intégré plus adapté pour le codage dans le langage java	Version Gratuite
Visual Studio Code	Editeur de code, multiplateforme, supposant plusieurs langages et plusieurs outils de programmation.	Gratuit
Microsoft Office professional 2021	Logiciel de traitement des texte	328 400 FCFA
Microsoft Edge	Navigateur Web, Interface de connexion à internet	Gratuit
Total	495400 FCFA	

4- ressources matérielles

Le tableau ci-dessous résume le matériel indispensable au déploiement de notre solution

Tableau 2: Ressources matérielles

Nom des matériels	Quantité	Prix unitaire en franc CFA	Prix total (FCFA)
-------------------	----------	----------------------------	-------------------

CONCEPTION D'UNE PLATEFORME DE COMPTAGE DE VOTES

Ordinateur portable DELL Processeur Intel® Core™ i5 RAM 8 Go, disque dur 1 T	03	550 000	1650 000
CD ROM	01	550	550
Disque Dur EXTERNE TOSHIBA 1 TB USB 3.0	01	62 675	62 675
Rame de papier format A4	01	4500	4500
Modem wifi 4G lite	02	69900	139800
Stylos Schneider	04	1000	4000
Total			1 861 525 FCFA

5- coût total du projet

Le tableau ci-dessous résume le coût total indispensable au déploiement de la solution.

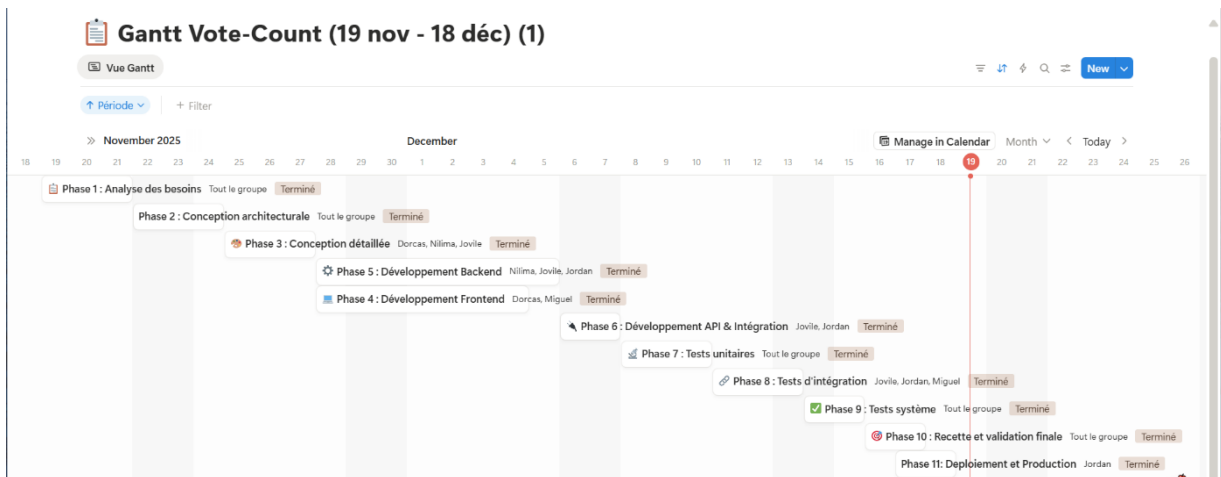
Tableau 3: cout total

Ressources	Montant en franc CFA
Ressources humaines	10 215 000 FCFA

Ressources matérielles	1 861 525 FCFA
Ressources logicielles	495 400 FCFA
Imprévus (10 %)	1 510 000 FCFA
Total	14 081 925 FCFA

VI- Planification du projet

Cette étape nous permettra de représenter la répartition des tâches sur des intervalles de temps bien définis. Pour cette planification, nous ferons recours au logiciel Gantt Project. Nous obtiendrons donc un chronogramme de Gantt qui représentera visuellement l'état d'avancement de nos différentes tâches.



VII- contraintes

Une contrainte est une condition à respecter par les parties prenantes pour la bonne marche du projet ; pour mener à bien ce projet, les contraintes suivantes s'imposent :

➤ Coût

Nous devons respecter le prix fixé dans l'étude financière et éviter une surestimation et une sous-estimation de ce prix estimé à 14 081 925 FCFA;

➤ Délai

Le projet devra respecter l'intervalle de temps défini de commun accord par les maîtres d'œuvre et d'ouvrage qui devra s'étendre sur une période allant du 24 novembre 2025 au 19 décembre 2025.

➤ **Qualité**

Après le prix et le délai fixé, nous devons produire un site web de bonne qualité répondant aux besoins exprimés et devant permettre un usage facile aux utilisateurs. L'application devra être très performante, extensible et évolutive.

➤ **Techniques**

Application web responsive compatible avec tous les navigateurs

Base de données relationnelle : MySQL

Langages recommandés : JAVA, HTML, CSS, JAVASCRIPT

VIII- livrables

Un livrable est un composant matérialisant un résultat mesurable ou vérifiable qui émane de l'achèvement d'une prestation de réalisation à la direction des systèmes d'information (DSI).

Les livrables attendues pour ce projet sont :

- un rapport physique comportant le dossier d'analyse de conception et de réalisation
- une maquette

Chapitre 2 : DOSSIER D'ANALYSE

Le dossier d'analyse a pour but d'affiner le choix de la méthode à utiliser tout au long du projet. Il permet dans la même lancée de faire comme son nom l'indique une analyse détaillée du projet à mettre en œuvre.

I- Etude et critique de l'existant

1- Description de l'existant

Le vote se déroule dans des bureaux de vote physiques où les électeurs se rendent le jour du scrutin. Chaque bureau est composé d'un président, d'assesseurs et de scrutateurs chargés de superviser le bon déroulement des opérations. Les principales étapes sont :

Ouverture du bureau de vote par les responsables électoraux à l'heure prévue.

- Accueil de l'électeur dans le bureau de vote.
- Vérification de l'identité de l'électeur à l'aide d'une pièce officielle et contrôle de son inscription sur la liste électorale.
- Émargement de l'électeur sur la liste afin de confirmer qu'il n'a pas déjà voté.
- Remise du bulletin de vote à l'électeur.
- Passage de l'électeur dans l'isoloir pour exprimer son choix en toute confidentialité.
- Dépôt du bulletin de vote dans l'urne prévue à cet effet.
- Clôture du vote à l'heure réglementaire.
- Ouverture de l'urne et dépouillement des bulletins.
- Comptage des voix et consignation des résultats dans un procès-verbal.
- Transmission des résultats aux autorités compétentes pour centralisation et publication.

2- Critique et limites de l'existant

Le système de vote actuel présente plusieurs insuffisances :

- Faible transparence du processus
- Absence de résultats en temps réel
- Dépendance excessive aux documents papier
- Coûts logistiques élevés
- Difficulté d'audit et de vérification a posteriori

II- Problématique

Bien que le vote soit un pilier essentiel de la démocratie, le processus électoral actuel repose principalement sur des méthodes manuelles, caractérisées par des lenteurs, des erreurs de comptage et des difficultés dans la centralisation des résultats. Ces insuffisances nuisent à la transparence du scrutin, retardent la publication des résultats et augmentent les risques de contestation. Face à ces limites, la conception et la mise en œuvre d'un système informatisé de comptage des votes s'avèrent nécessaires afin d'automatiser le traitement des résultats, garantir leur fiabilité et leur traçabilité, et permettre une diffusion plus rapide et plus crédible des résultats électoraux.

IV- Choix Et Présentation De La Démarche D'analyse

De l'analyse comparative réalisée dans le tableau ci-dessus, nous choisissons **UML** pour modéliser notre système d'information car ce langage de modélisation unifié, est un langage de modélisation graphique conçu pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet.



Figure 4: logo UML

UML est :

Un langage pseudo formel qui **cadre l'analyse** en offrant les différentes vues complémentaires d'un système, guidant ainsi l'utilisation des concepts objets ;

Un langage formel et **normalisé** qui donne un gain de précision et encourage l'utilisation d'outils ;

Un support de communication qui facilite la compréhension des représentations abstraites complexes.

Il a pour objectifs : de représenter le système tout entier, de prendre en compte le facteur d'échelle c'est-à-dire la grandeur du projet, de créer un langage de modélisation à la fois utilisable par les humains et les machines.

UML s'articule maintenant autour de quatorze (14) diagrammes différents, depuis sa version 2.3. Ces diagrammes sont dépendants hiérarchiquement et se complètent, de façon à permettre la modélisation d'un projet tout au long de son cycle de vie. Ils sont regroupés en deux grandes vues :

Les vues statiques ou structurelles Ces

vues rassemblent :

Le diagramme de classe (class Diagram) : représentation des classes intervenant dans le système ;

Le diagramme d'objet (Object Diagram) : représentation des instances de classes (objet) utilisées dans le système ;

Le diagramme de composant (component Diagram) : représentation des composants du système d'un point de vue physique, tels qu'ils sont mis en œuvre (fichiers, bibliothèques, bases de données) ;

Le diagramme de déploiement (deployment Diagram) : représentation des éléments matériels (ordinateurs, périphériques, réseaux, systèmes de stockages...) et la manière dont les composants du système sont répartis sur ces éléments matériels et interagissent entre eux ;

Le diagramme de structure composite (composite structure Diagram) : représentation sous forme de boîte blanche les relations entre composants d'une classe (depuis UML 2.x) ;

Le diagramme de paquetages (package Diagram) : représentation des dépendances entre les paquets (un paquet étant un conteneur logique permettant de regrouper et d'organiser les éléments dans le modèle UML), c'est-à-dire entre les ensembles de définitions ;

Diagramme de profils : permet de spécialiser, de personnaliser pour un domaine particulier un méta modèle de référence d'UML.

Les vues dynamiques ou d'interactions Dans

les vues dynamiques nous avons :

Le diagramme de séquence (séquence Diagram) : représentation de façon séquentielle du déroulement des traitements et des interactions entre les éléments du système et/ou de ses acteurs ;

Le diagramme de communication ou de collaboration : permet de mettre en évidence les échanges de messages entre objets ;

Le diagramme global d'interactions (interaction overview Diagram) : représentation des enchaînements possibles entre les scénarios préalablement identifiés sous forme de diagrammes de séquences (variante du diagramme d'activité depuis UML 2.x) ;

Le diagramme de temps (timing Diagram) : représentation des variations d'une donnée au cours du temps ;

Le diagramme d'activité (activity Diagram) : représentation sous forme de flux ou d'enchaînement d'activités le comportement du système ou de ses composants ;

Le diagramme de cas d'utilisation (use-case Diagram) : représentation des possibilités d'interaction entre le système et les acteurs (intervenants extérieurs au système), c'est-à-dire de toutes les fonctionnalités que doit fournir le système ;

Le diagramme d'état transition (state machine Diagram) : représentation sous de machine a états finis le comportement du système ou de ses composants ;

UML n'étant pas une méthode mais un langage formel et normalisé, pour développer des solutions, on a besoin de lui associer un processus de développement parmi lesquels UP, RUP, 2TUP, XUP, Extrême programming, etc... pour ne citer que ceux-là.

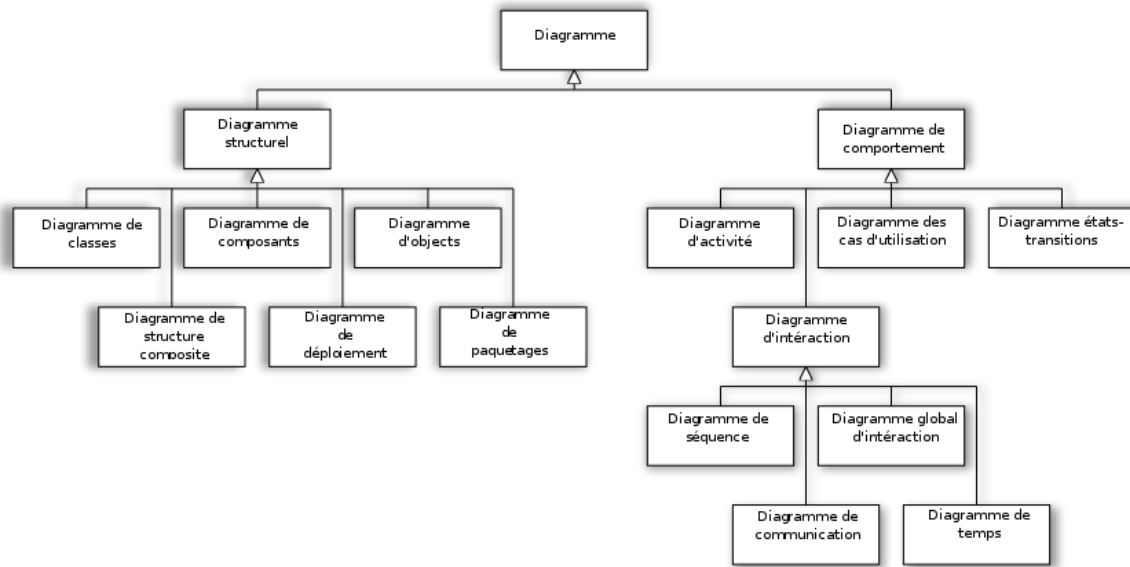


Figure 5: ensemble de diagramme d'UML

❖ CHOIX DU PROCESSUS DE DEVELOPPEMENT

Un processus définit une séquence d'étapes qui concourent à l'obtention d'un système logiciel ou à l'évolution d'un système existant. L'objectif d'un processus de développement est de produire des solutions de qualité qui répondent aux besoins des utilisateurs dans les délais et les coûts prévisibles.

Présentation du processus en V

Le cycle en V est une méthode de gestion de projet séquentielle, héritée du modèle en cascade, qui structure le projet en deux phases en forme de V : une descente pour la **conception et le développement** (analyse des besoins, spécifications techniques, programmation) et une remontée pour la **validation et les tests** (tests unitaires, intégration, recette). Chaque étape de conception a sa phase de validation correspondante, assurant que chaque partie construite est vérifiée et validée selon les exigences initiales, le bas du V étant la réalisation concrète.

Les phases du Cycle en V

- Côté gauche (Descente) : Conception et Spécification
 - Analyse des besoins : Définition des exigences du client.
 - Spécifications fonctionnelles : Proposition des fonctionnalités.
 - Spécifications techniques : Détail des aspects techniques (matériels, logiciels).

-Conceptions détaillées : Descriptions fines des composants (ex: schémas de BDD).

- Bas du V : Réalisation
 - Programmation/Codage : Implémentation des solutions.
- Côté droit (Remontée) : Tests et Validation
 - Tests unitaires : Vérification des blocs de code individuels.
 - Tests d'intégration : Vérification du fonctionnement des modules assemblés.
 - Tests de validation : Test de conformité globale du système.
 - Recette et mise en production : Livraison et approbation finale par le client

Ainsi donc, les avantages que nous présente le processus en V sont :

- **Clarté et structure** : Approche linéaire avec étapes distinctes (analyse, conception, développement, test) qui réduit la confusion et aligne les parties prenantes.
- **Qualité proactive** : Les tests (unitaires, intégration, validation) sont planifiés en parallèle de la conception, permettant de trouver et corriger les erreurs tôt.
- **Traçabilité** : Lien direct entre les exigences initiales et les phases de test correspondantes, assurant le respect des besoins.
- **Réduction des coûts** : Identifier les bugs tôt évite des corrections coûteuses en fin de projet.
- **Gestion des risques** : La vision en amont permet une meilleure anticipation et maîtrise des risques.
- **Documentation** : Chaque phase génère une documentation détaillée, facilitant le suivi et la maintenance

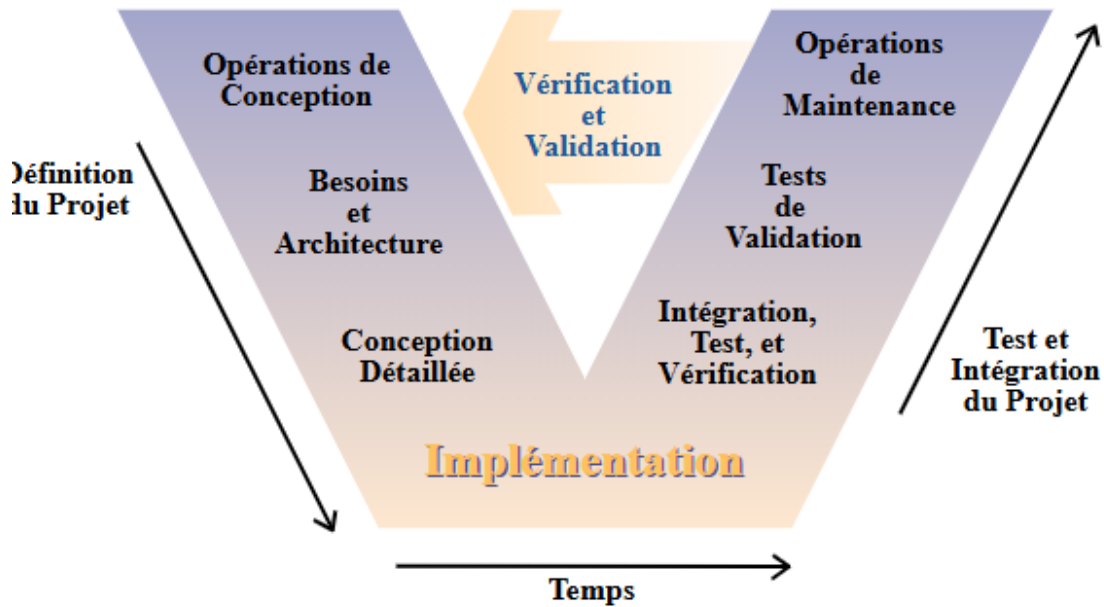


Figure 5: processus en V

Décrivons la forme Y du processus 2TUP :

La branche gauche (fonctionnelle) comporte :

- ♣ La capture des besoins fonctionnels : qui produit un modèle des besoins focalisés sur le métier des utilisateurs. Elle qualifie au plus tôt le risque de produire un système inadapté aux utilisateurs. De son côté, la maîtrise d'œuvre consolide les spécifications et en vérifie la cohérence et l'exhaustivité;

- ♣ L'analyse : qui consiste à étudier précisément la spécification fonctionnelle de manière à obtenir une idée de ce que va réaliser le système en termes de métier. Les résultats de l'analyse ne dépendent d'aucune technologie particulière ;

La branche droite (technique) comporte :

- ♣ La capture des besoins techniques : qui recense toutes les contraintes et les choix dimensionnant la conception du système. Les outils et les matériels sélectionnés ainsi que la prise en compte de contraintes d'intégration avec l'existant conditionnent généralement des prérequis d'architecture technique ;

- ♣ Les architectures matérielles et logicielles : constituant de la conception générique, qui définit ensuite les composants nécessaires à la construction de l'architecture technique. Cette conception est complètement indépendante des aspects fonctionnels. Elle a pour

objectif d'uniformiser et de réutiliser les mêmes mécanismes pour tout un système. L'architecture technique construit le squelette du système informatique et écarte la

plupart des risques de niveau technique. L'importance de sa réussite est telle qu'il est conseillé de réaliser un prototype pour assurer sa validité ; La branche du milieu (conception) comporte :

- ♣ La conception préliminaire : qui représente une étape délicate, car elle intègre le modèle d'analyse dans l'architecture technique de manière à tracer la cartographie des composants du système à développer ;
- ♣ La conception détaillée : qui étudie ensuite comment réaliser chaque composant, l'étape de codage, qui produit ces composants et teste au fur et à mesure les unités de code réalisées ; l'étape de recette, qui consiste enfin à valider les fonctions du système développé ;
- ♣ Codage, tests et recette : qui renvoie à la phase de programmation des fonctionnalités au fur et à mesure ; et aussi, à la validation des fonctions du système développé.

V- PRESENTATION DES DIAGRAMMES INTERVENANTS

1- Diagramme de USE CASE


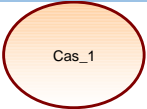

❖ Présentation


Les diagrammes de cas d'utilisation identifient les fonctionnalités fournies par le système (les cas d'utilisations), les utilisateurs qui interagissent avec le système (acteurs) et les interactions entre ces derniers. C'est une description de l'ensemble des opérations que l'utilisateur pourra effectuer dans le système.

Les cas d'utilisations sont représentés par une ellipse contenant le nom du cas d'utilisation. Un acteur et un cas d'utilisation sont mis en relation par une association représentée par une ligne.

❖ Formalisme

Tableau 4: Formalisme diagramme de UseCase

Représentation	Composition	Description
 Acteur_1	Acteur	Représente un rôle joué par une personne ou une chose qui interagit avec le système. (La même personne physique peut donc être représentée par plusieurs acteurs en fonction des rôles qu'elle joue).
 Cas_1	Cas d'utilisation (use case)	Correspond à un objectif du système, motivé par un besoin d'un ou plusieurs acteurs ou tout simplement une fonctionnalité du système
 Système	Périmètre système	Le système représente l'espace étudié ; c'est dans celui-ci que l'acteur effectue ses actions par conséquent c'est dans ce dernier que se trouve les cas d'utilisations. Il est représenté par un cadre avec le nom du système.

	<p>Relation</p>	<p>Elle exprime l'interaction existant entre un acteur et un cas d'utilisation</p> <p>Il existe 3 types de relations entre cas d'utilisation :</p> <p>La relation de généralisation</p> <p>La relation d'extension</p> <p>La relation d'inclusion</p>
---	-----------------	---

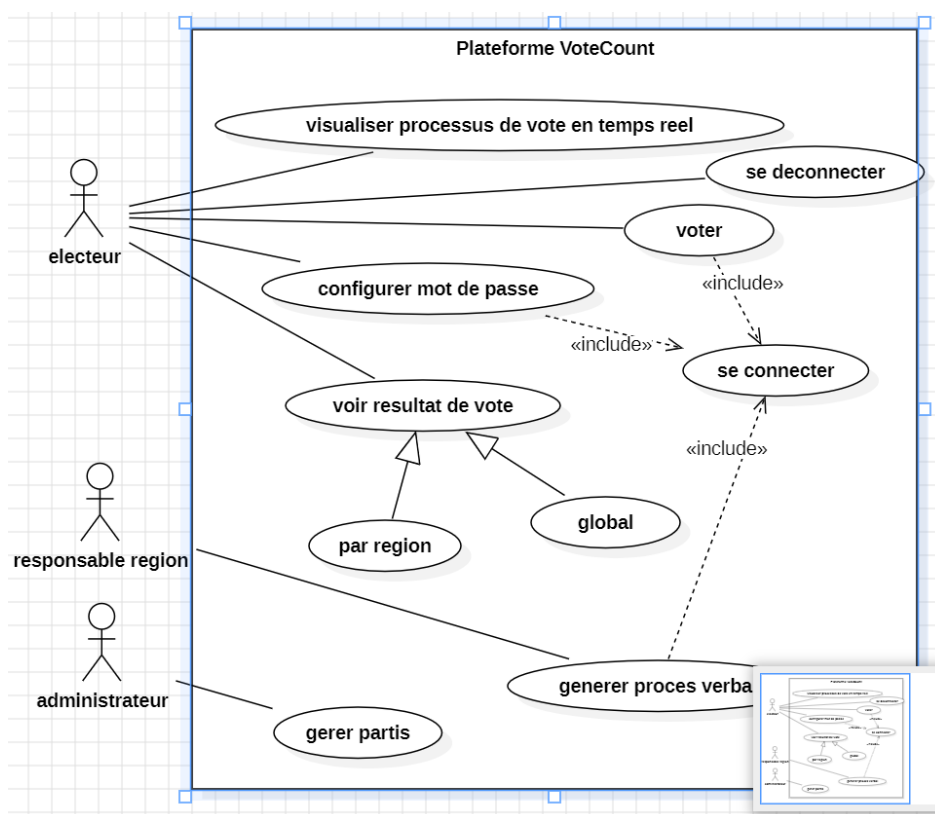


Figure 1: Diagramme de use case

2- Diagramme de séquence


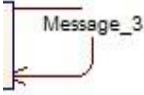

❖ Présentation

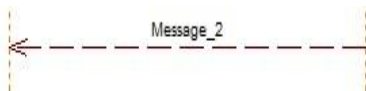
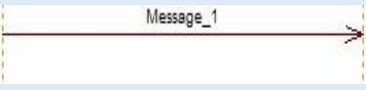

Les diagrammes de séquence sont des représentations graphiques des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation UML. Le diagramme de séquences permet de décrire les interactions entre les objets au sein d'un diagramme de cas d'utilisation.

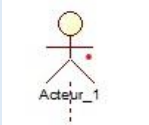

❖ Formalisme

Le diagramme de séquence énumère les objets horizontalement et le temps verticalement. Il modélise l'exécutable des différents modèles en fonction du temps. Dans ce diagramme, les objets et les acteurs sont énumérés en colonnes avec leurs lignes de vie verticales indiquant la durée de vie de l'objet.

Tableau 5: formalisme du sequence diagram

Représentation	Composition	Description
	Ligne de vie	Ligne verticale qui identifie l'existence de l'objet par rapport au temps
	Message réflexif	C'est lorsqu'un objet ou un acteur s'auto envoie un message
	Activation	Boite rectangulaire sur une ligne de vie marquant la portée d'une action.
	Message retour	Réponse d'un objet suite à un message. Il marque la fin d'une activation.

		
	Message asynchrone	Contrairement au message synchrone l'envoi d'un message asynchrone ne met pas en attente le déroulement de l'activation jusqu'au message de retour.
	Objet	Ce sont les instances de classe.

	Acteur	Personne qui interagit et communique avec le système.
	Message synchrone	L'envoi d'un message synchrone met en attente le déroulement de l'activation jusqu'à la réception d'une réponse provenant de l'objet/l'acteur à qui le message a été envoyé.

• Quelques diagrammes de séquence de notre système

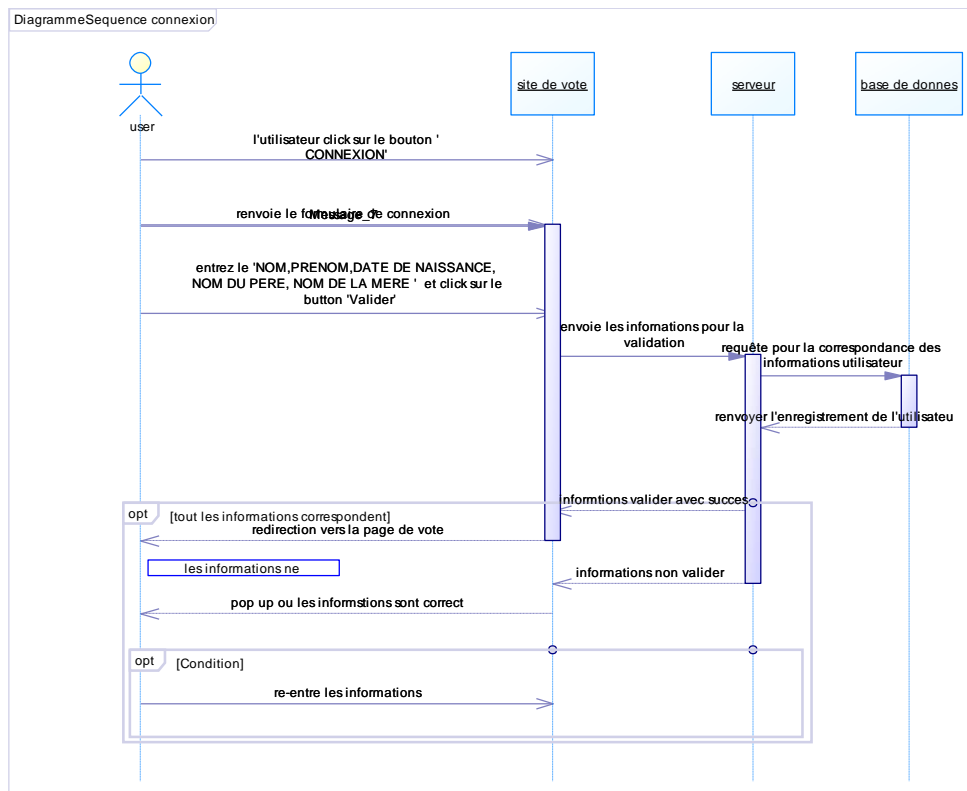


Figure 2: Diagramme de sequence pour connexion

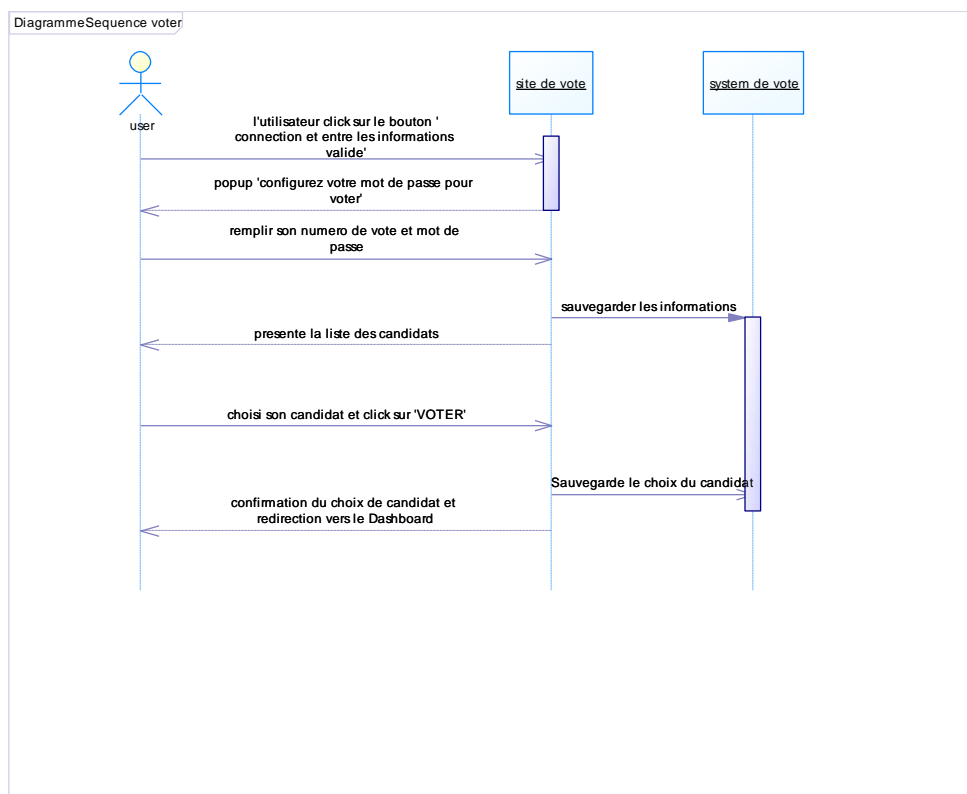


Figure 3:Diagramme de sequence processus de vote

3- Diagramme de classe

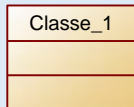

• Présentation

Le diagramme de classe permet de représenter l'ensemble des informations finalisées qui sont gérées par le domaine. Ces informations sont structurées, c'est-à-dire qu'elles sont regroupées dans des classes. Le diagramme met en évidence d'éventuelles relations entre ces classes. Contrairement au diagramme de cas d'utilisation qui montre le système du point de vue des acteurs, lui il montre la structure interne. Il permet de fournir une représentation abstraite des objets du système qui vont interagir pour réaliser les cas d'utilisations.

• Formalisme

Les éléments constitutifs du diagramme de classe sont :

Tableau 13 : Formalisme du diagramme de classe

Élément du diagramme	Description	Représentation												
La classe	Est un ensemble d'attribut et de méthode regroupés au sein d'une entité ou encore une classe est une description abstraite (condensée) d'un ensemble d'objets du domaine de l'application : elle définit leur structure, leur comportement et leurs relations.													
La multiplicité	Définit le nombre d'instances de l'association pour une instance de la classe.	<table><tr><td>1</td><td>Un et un seul</td></tr><tr><td>0..1</td><td>Zéro ou un</td></tr><tr><td>N ou *</td><td>N (entier naturel)</td></tr><tr><td>M..N</td><td>De M à N (entiers naturels)</td></tr><tr><td>0..*</td><td>De zéros à plusieurs</td></tr><tr><td>1..*</td><td>De 1 à plusieurs</td></tr></table>	1	Un et un seul	0..1	Zéro ou un	N ou *	N (entier naturel)	M..N	De M à N (entiers naturels)	0..*	De zéros à plusieurs	1..*	De 1 à plusieurs
1	Un et un seul													
0..1	Zéro ou un													
N ou *	N (entier naturel)													
M..N	De M à N (entiers naturels)													
0..*	De zéros à plusieurs													
1..*	De 1 à plusieurs													
L'association	C'est la relation générique entre deux classes													




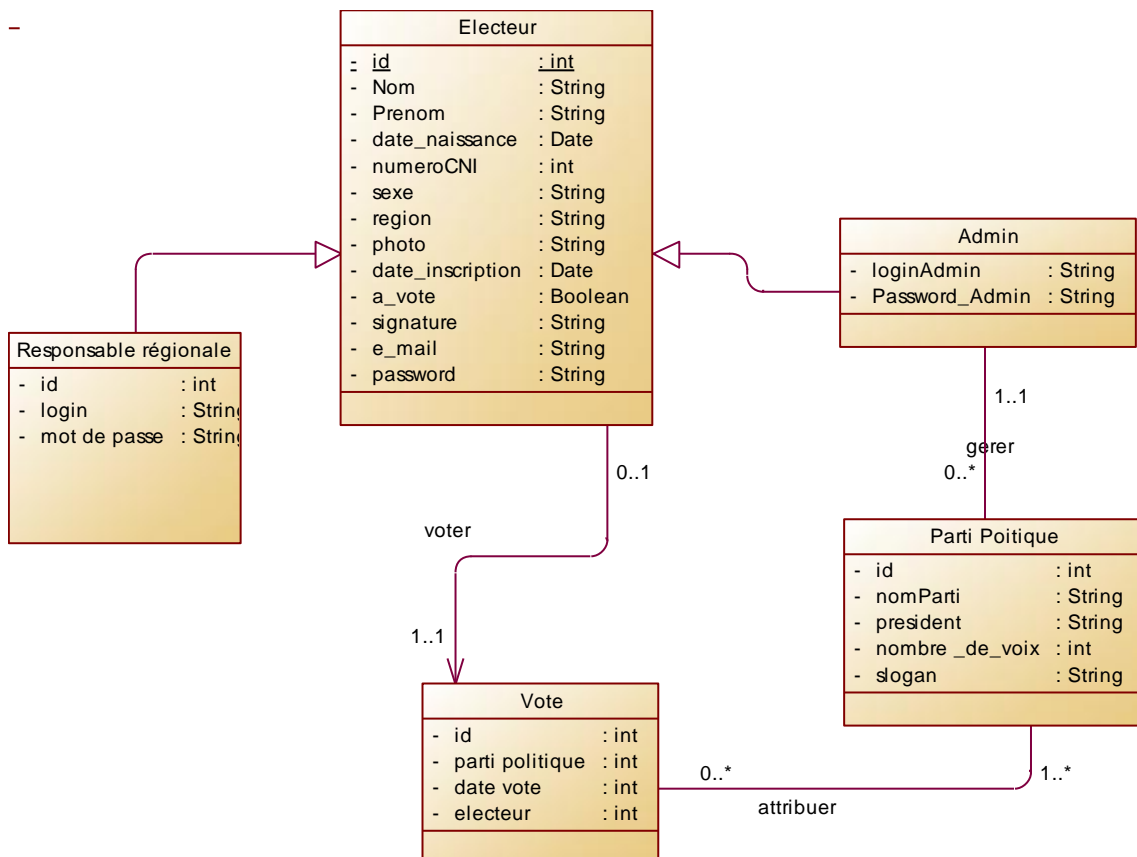
La généralisation	Traduit le concept d'héritage. Est utilisée pour faire le lien entre un (ou plusieurs) objet(s).	
L'agrégation	C'est une relation particulière qui attribue à l'une des classes le rôle d'agrégat et à l'autre classe le rôle d'agrégé.	
La composition	Est un cas particulier d'agrégation. Ici les classes composantes ne peuvent exister sans leur composite	

Tableau 14:diagramme de classe de la solution



CONCLUSION GENERAL

Au terme de ce projet académique, il apparaît clairement que la conception et le développement d'une plateforme de comptage automatique des votes constituent une réponse pertinente aux limites des méthodes traditionnelles de dépouillement. En s'appuyant sur une approche méthodologique rigoureuse, fondée sur le cycle de vie en V, le projet a permis de structurer l'ensemble du processus de développement, depuis l'expression des besoins jusqu'à la validation finale du système, tout en garantissant la cohérence entre les exigences initiales et la solution réalisée.

La plateforme développée répond aux objectifs fixés en assurant un comptage fiable, rapide et sécurisé des votes, tout en offrant une visualisation des résultats en temps réel. Les différentes phases d'analyse et de conception, appuyées par la modélisation UML, ont permis de définir une architecture claire et évolutive, facilitant ainsi la maintenabilité et la compréhension du système. La phase de réalisation, complétée par des tests unitaires, d'intégration et de validation, a contribué à renforcer la qualité du logiciel et à réduire les risques d'erreurs fonctionnelles ou techniques.

Au-delà de l'aspect technique, ce projet a également constitué une expérience pédagogique enrichissante, permettant de mettre en pratique des connaissances théoriques en ingénierie logicielle, en gestion de bases de données et en sécurité informatique. Il a favorisé le développement de compétences essentielles telles que l'analyse des besoins, le travail collaboratif, la gestion de projet et la prise en compte des contraintes réelles liées aux systèmes critiques.

Toutefois, bien que la solution proposée réponde aux exigences du cadre académique, certaines limites subsistent, notamment en matière de déploiement à grande échelle, d'intégration de mécanismes de sécurité avancés et de prise en compte des cadres juridiques spécifiques aux systèmes de vote électronique. Ces limites ouvrent des perspectives d'évolution intéressantes, telles que l'amélioration des mécanismes de sécurisation des données, l'intégration de solutions d'authentification renforcée ou encore l'adaptation du système à des contextes électoraux plus complexes.

En définitive, ce projet illustre l'apport des solutions numériques dans l'amélioration des processus électoraux et met en évidence l'importance d'une démarche méthodologique rigoureuse pour garantir la fiabilité et la légitimité des systèmes de vote électroniques. Il constitue ainsi une base solide pour de futurs travaux de recherche ou de développement dans le domaine des systèmes électoraux informatisés.

TABLE DES MATIERES

REMERCIEMENTS	1
Sommaire	3
LISTE DES TABLEAUX.....	4
LISTE DES FIGURES.....	5
RESUME	6
ABSTRACT.....	7
INTRODUCTION GENERAL.....	8
Partie 1 : ETUDE DE FAISABILITE.....	9
A- Analyse SWOT.....	9
B- HIERARCHISATION DES BESOINS	12
1- DEFINITION	12
i- Objectifs de la Hiérarchisation	12
ii- Étapes de la Hiérarchisation des Besoins	13
iii- Méthodes de Hiérarchisation.....	13
2- APPLICATION AVEC L'ANALYSE MOSCOW	13
i- Catégories de l'Analyse MoSCoW.....	13
ii- Identification des besoins	14
Partie 2 : PHASE TECHNIQUE.....	20
A- CAHIER DE CHARGE.....	20
I- CONTEXTE	20
II- OBJECTIFS DU PROJET	21
1- Objectif général.....	21
L'objectif général du projet est de concevoir et de développer une plateforme de comptage de votes permettant d'améliorer la fiabilité, la rapidité et la transparence du processus de dépouillement et de centralisation des résultats électoraux.....	21

2- Objectifs spécifiques	21
IV- EXPRESSION DES BESOINS	22
1- Besoins fonctionnel	22
2- Besoins non fonctionnels	22
V- ESTIMATION DU COÛT DU PROJET	23
1- Ressources humaines	23
3- Ressources logicielles	24
4- ressources matérielles	25
5- coût total du projet	26
VI- Planification du projet.....	27
VII- contraintes	27
VIII- livrables	28
Chapitre 2 : DOSSIER D'ANALYSE	28
I- Etude et critique de l'existant	29
1- Description de l'existant	29
2- Critique et limites de l'existant	29
II- Problématique	30
IV- Choix Et Présentation De La Démarche D'analyse	30
V- PRESENTATION DES DIAGRAMMES INTERVENANTS	36
1- Diagramme de USE CASE	36
□ Présentation	36
2- Diagramme de séquence	39
3- Diagramme de classe	42
• Présentation	42
CONCLUSION GENERAL	45
TABLE DES MATIERES	47

