

Membre :

Diallo Seydou  
Doumbia Amadou  
Bansimba Ceulain  
Laagad El Mehdi  
Oubraim Rachid

# Le système de vote électronique

## **Introduction**

Le vote est un système de communication, celui-ci permet de partager, échanger ses idées et exprimer son opinion.

L'utilité du vote est la prise d'une décision commune/globale d'un groupe d'individus (quelque chose ou quelqu'un), ce pour quoi on vote peut être représenté par une idée, une solution ou une personne censée les incarner.

La nature et les circonstances dans lesquelles les systèmes de vote électronique (application web, boîtier électronique, smartphone) sont utilisés sont diverses.

Libres sans enjeu ou contrôlés ils sont utilisés pratiquement dans tous les domaines (administratif, économique, politique, culturel, professionnel) : sondage, meeting, réunion d'entreprise, jeu télévisé...

## **Problématique :**

Quand on parle de système de vote, des questions liées à la sécurité, la fiabilité, l'intégrité des données et la cohérence des processus électoraux sont soulevées.

Toutes ces questions proviennent du manque de transparence vis-à-vis des électeurs qui n'ont aucune visibilité sur les problèmes internes que les systèmes électoraux peuvent rencontrer.

Afin que les électeurs puissent s'assurer du bon fonctionnement et de la cohérence des processus électoraux, des services permettant l'accès à ces informations doivent être mis à leur disposition.

Un système électoral pourrait-il tenir compte des questions de transparence de ces futurs participants ?

## Spécifications fonctionnelles :

### Le Framework prendra en compte les acteurs suivants:

- **Électeur (utilisateur de base) :** l'électeur peut voter aura le droit de voter dès lors qu'il se sera inscrit sur la liste des votant pour un scrutin proposé. Ce profil offre les fonctionnalités de base, ses droits sont accessibles à tout utilisateur.
- **Organisateur :** l'organisateur a la possibilité d'organiser (créer) des élections, d'en définir les modes de scrutins ainsi que leur accessibilité, c'est-à-dire les critères auxquels les utilisateurs doivent répondre pour avoir le droit de participer (électeurs et les candidats).
- **Candidat :** Le candidat est un choix possible pour lequel les électeurs peuvent voter. Il représente une idée, une solution, une décision ou quelqu'un, dans ce dernier, le candidat peut aussi être un électeur qui a choisi de participer à une élection dans laquelle il peut se présenter.
- **Administrateurs :** l'administrateur est un super-profil dont la liste complète des droits sera à définir avant la génération de la plateforme de vote. Ce profil aura, de base, pour rôle de valider les demandes de création d'élections.

**Mode de scrutins :** Le Framework permettra de créer différents moyens de système de vote, le système de qualification de tours par un candidat sera alors mis en place par l'organisateur : Scrutin uninominal/plurinominal et majoritaire/proportionnel/mixte à n tours.

## Spécifications non fonctionnelles :

- **Fiabilité:** Le système doit pouvoir assurer à chaque acteur que sa propre voix a bien été comptabilisée et garantir l'authenticité des résultats.
- **Cohérence :** Le système doit répondre exactement aux règles prédéfinies.
- Un électeur ne peut pas voter pour un candidat participant à une élection si celle-ci ne concerne pas sa circonscription ou s'il correspond aux critères fixés par l'organisateur.
- Un électeur ne pourra pas voter plus d'une fois lors d'une même élection, il sera également impossible de modifier son choix après validation.
- Un candidat ne peut figurer qu'une seule fois dans la liste des candidats d'une élection.
- Pour qu'un candidat puisse se présenter à une élection, celui-ci devra respecter les critères fixés par l'organisateur.
- Un organisateur ne pourra pas se présenter en tant que candidat à l'élection qu'il aura organisé.
- Il devra être possible pour un électeur après la fermeture d'un scrutin de vote, de vérifier la prise en compte de son vote.

```
graph TD
    P[programmeur] -->|ImplemeVote| MG[Module de generation de Vote]
    MG -->|Service de Vote| SV[Service de Vote]
    E[Electeur] -->|Voter(NumeroUnique)| SV
    C[Candidat] -->|se presenter(Candidat)| SV
```

The diagram illustrates the functional requirements of an election system, involving three actors: User, Organisateur, and Candidat.

- User** interacts with: s'inscrire, Mettre à jour info-perso, Vérification d'un vote, Voter.
- Organisateur** interacts with: Se connecter, Définir les règles, Création d'élection, Définir Liste des participants, Définition du Mode de Scrutin.
- Candidat** interacts with: Se présenter à une élection.

**Use Cases:**

- s'inscrire
- Mettre à jour info-perso
- Vérification d'un vote
- Voter
- Se connecter
- Définir Nombre de Tour
- Définir les règles
- Définition du Mode de Scrutin
- Fermeture des Candidatures
- ouverture Candidatures
- Création d'élection
- Définir Liste des participants
- Se présenter à une élection

**Relationships:**

- Include (<<Include>>):**
  - Se connecter includes s'inscrire and Mettre à jour info-perso.
  - Définir les règles includes Vérification d'un vote.
  - Définir les règles includes Définir Nombre de Tour.
  - Définition du Mode de Scrutin includes Définir les règles.
  - Fermeture des Candidatures includes Définir Nombre de Tour and Définition du Mode de Scrutin.
  - ouverture Candidatures includes Fermeture des Candidatures.
  - Se présenter à une élection includes ouverture Candidatures.
- Extend (<<Extend>>):**
  - Mettre à jour info-perso extends s'inscrire.
  - Création d'élection extends Définir Liste des participants.

### Diagramme de classe :

