

# S.A.É. 3 : Analyse, Conception et Développement d'une Application Web de Gestion Électorale

## 1 Description du projet :

L'élection est un processus ancien et universel qui accompagne l'histoire de la civilisation depuis ses origines. Elle permet de désigner des représentants ou de prendre une décision commune, et joue un rôle central dans de nombreux domaines, qu'ils soient politiques, culturels, sportifs ou associatifs.



Arc de l'élection du 13<sup>ème</sup> président dans l'anime *Hunter X Hunter* : un exemple de la dimension stratégique que peut prendre un scrutin.

Cependant, bien que largement utilisé, le vote papier présente plusieurs limites. Lors de scrutins de grande ampleur, des erreurs humaines peuvent survenir lors du dépouillement, et des risques de fraude ne peuvent être écartés. Ces contraintes soulignent la nécessité de recourir à des systèmes modernes et fiables, capables d'automatiser et de sécuriser l'ensemble du processus électoral, de l'inscription des électeurs jusqu'à la proclamation des résultats.

Ce projet a pour objectif de remplacer le vote papier par une application web assurant la gestion complète d'une élection. Il devra être réalisé en équipe, selon une démarche itérative, et conduire à la conception et au développement d'une solution fiable, sécurisée et ergonomique.

Chaque groupe choisira un contexte spécifique. Par exemple :

- **politique** : élection municipale, référendum ;
- **scolaire** : élection de délégués de classe ou de représentants étudiants ;
- **sportif** : Ballon d'Or, équipe de l'année ;
- **culturel ou médiatique** : prix littéraire, festival de cinéma, Miss World ;
- **etc.**

Dans chaque contexte, plusieurs modes de scrutin pourront être étudiés et mis en œuvre (majorité simple, majorité absolue à deux tours, proportionnelle, vote pondéré, etc.). Chaque choix entraînera des besoins spécifiques et un cahier des charges adapté.

### Objectifs :

- Clarifier et formaliser le besoin, en identifiant les attentes et contraintes du contexte choisi.
- Étudier l'existant : solutions de vote en ligne, méthodes utilisées, contraintes réglementaires et organisationnelles du domaine choisi.
- Concevoir et développer une application reposant sur un serveur et une base de données.

- Implémenter les fonctionnalités essentielles : gestion des électeurs, gestion des candidats, unicité et sécurité du vote, publication automatique des résultats.
- Suivre une démarche de développement itérative, intégrant des phases de conception, de prototypage, de tests et d'améliorations successives.

**Contraintes :**

- Respect strict de l'anonymat : séparation entre l'identité des électeurs et les bulletins. On ne doit pas pouvoir savoir qui a voté pour qui.
- Unicité du vote : un électeur ne peut voter qu'une seule fois et un vote soumis est irréversible.
- Gestion temporelle du scrutin : l'application doit respecter les différentes phases de l'élection. Avant l'ouverture du scrutin, le vote est impossible ; durant la période de vote, les électeurs peuvent voter une seule fois ; après la clôture, le système doit fournir les résultats agrégés et anonymes.
- Résultats visibles uniquement à la clôture du scrutin : aucun résultat partiel ne doit être disponible, y compris pour l'administrateur.
- Interface utilisateur claire, simple et ergonomique.
- Qualité du code : lisibilité, documentation, tests fonctionnels et jeux d'essais systématiques.

**Fonctionnalités complémentaires :**

- Analyse des résultats lorsque le contexte le permet : représentations graphiques.
- Interactions : commentaires publics et outils collaboratifs tels qu'un chat ou un forum.

**Consignes :**

- Le travail doit être réalisé en binôme.
- Le projet doit être déposé sur un dépôt Git, avec des messages de commit pertinents, faisant apparaître les différentes étapes du travail et une réelle collaboration.
- Vous ajouterez à votre dépôt vos enseignants des ressources : R3.01 Développement web et R3.04 Qualité de développement.
- Les livrables doivent être présentés sous forme de rapports PDF rédigés, avec un texte explicatif clair et structuré. Les textes générés par l'IA ne sont pas acceptés.
- Les délais de remise doivent être respectés.
- Les notions vues dans les ressources mobilisées doivent être appliquées dans le projet.

**Ressources mobilisées :**

- R3.01 Développement web (2h TD, 5h TP) : M. Micard, M. Rositi
- R3.02 Développement efficace (2h TD, 2h TP) : M. Assayakh
- R3.03 Analyse (2h TD, 2h TP) : M. Assayakh
- R3.04 Qualité de développement (3h TD, 6h TP) : M. Assayakh
- R3.05 Programmation système (3h TD, 3h TP) : M. Dieudonné, M. Bourjij
- R3.06 Architecture réseaux (2h TD, 2h TP) : M. Dieudonné
- R3.07 SQL et programmation (3h TD, 4h TP) : M. Beaudeau
- R3.10 Management des SI (6h TD, 2h TP) : M. Davin
- R3.11 Droit, contrats et numérique (16h TD) : Mme Prinz
- R3.12 Anglais (4h TD, 2h TP) : Mme Caspary
- R3.13 Communication professionnelle (4h TD, 2h TP) : Mme Doennlen

## 2 Pré-planning des rendus

Date	Livrables	Ressources concernées
15/10/25	Cahier des charges fonctionnel	R3.10 Management des SI (1.1)
26/10/25	Premiers diagrammes UML	R3.03 Analyse (2.1)
	MCD/MLD	R3.07 SQL et programmation (3)
22/11/25	Dossier d'analyse et de conception	R3.03 Analyse (2.2)
	MVP 1	R3.01 Développement web (4.1)
		R3.02 Développement efficace (5.1)
12/12/25	MVP 2	R3.04 Qualité de développement (6.1)
		R3.01 Développement web (4.2)
		R3.02 Développement efficace (5.2)
18/01/26	Rapport technique de sécurisation	R3.04 Qualité de développement (6.2)
	Rapport final de la réalisation	R3.05 Programmation système (7)
		R3.06 Architecture réseaux (8)
		R3.01 Développement web (4.3)
	Rapport d'analyse de la documentation et de l'approche qualité	R3.02 Développement efficace (5.3)
	Bilan du projet et perspectives	R3.04 Qualité de développement (6.3)
	Présentation de l'application (affiche ou vidéo)	R3.10 Management des SI (1.2)
23/01/26	Soutenance finale	R3.12 Anglais (9)
		R3.13 Communication professionnelle (10)
		R3.11 Droit, contrats et num. (11)
		Toutes les ressources (12)

### 3 Évaluation

Chaque livrable est évalué dans les ressources concernées (les numéros de notes apparaissent dans les cercles). La note finale est calculée comme suit :

$$\text{Note finale} = \frac{2 \cdot C1 + C2 + 2 \cdot C3 + C4 + 2 \cdot C5 + 3 \cdot C6 + \textcircled{12}}{11}$$

<b>Compétence 1</b> : Partir des exigences et aller jusqu'à une application complète.	$C1 = \frac{1}{2}(\textcircled{4} + \textcircled{6})$
<b>Compétence 2</b> : Sélectionner les algorithmes adéquats pour répondre à un problème donné.	$C2 = \textcircled{5}$
<b>Compétence 3</b> : Déployer des services dans une architecture réseau.	$C3 = \frac{1}{2}(\textcircled{7} + \textcircled{8})$
<b>Compétence 4</b> : Optimiser une base de données, interagir avec une application et mettre en œuvre la sécurité.	$C4 = \textcircled{3}$
<b>Compétence 5</b> : Appliquer une démarche de suivi de projet en fonction des besoins métiers des clients et des utilisateurs.	$C5 = \frac{1}{2}(\textcircled{1} + \textcircled{2})$
<b>Compétence 6</b> : Situer son rôle et ses missions au sein d'une équipe informatique.	$C6 = \frac{1}{3}(\textcircled{9} + \textcircled{10} + \textcircled{11})$

*Bon courage !*