



RAPPORT APO

Modélisation de systèmes électoraux et dynamiques de vote



Lien GIT du projet : https://github.com/caaam213/APO_projet

Lien GIT pour l'interface graphique :
https://github.com/caaam213/APO_Interface_graphique

2021 ~ 2022

AARAB NABIL, MERAOUI CAMELIA, REHAB HOURIA

1 - Répartition des tâches et organisation du travail

1 - Répartition des tâches

Le travail a été découpé de manière équitable. Camélia et Houria ont fait le choix de travailler ensemble tandis que Nabil a travaillé de manière un peu plus isolée.

Travail réalisé par Camélia & Houria

1.1/ Réalisation du diagramme de cas d'utilisation, de séquence et d'activité

1.2/ Codage de Scrutins

1.3/Interface textuelle et graphique

1.4/Build.xml

1.5/Evolution des opinions

1.6/Spécialisation et fichier de configuration

Travail réalisé par Nabil

2.1/ Réalisation du diagramme de classe

2.2/ Codage de Scrutins

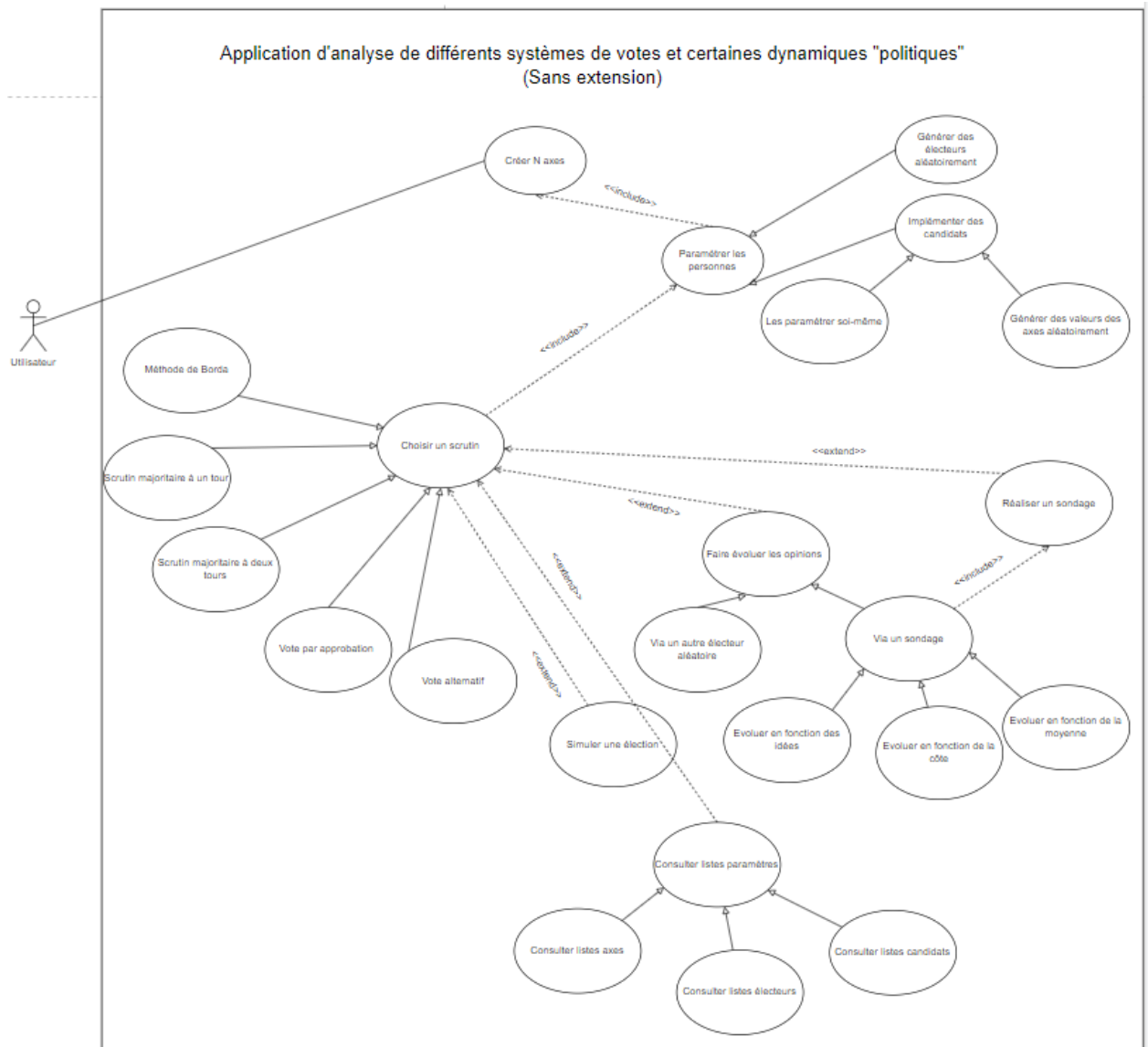
2.3/ Extension Sauvegarde Excel

2.4/ Extension d'Analyse du mode de Scrutin

2 - Organisation du travail et déroulé temporel

- Avant de commencer à coder, il fallait d'abord réaliser la phase de conception pour nous permettre de nous donner un point de départ et une bonne organisation.
- Après avoir réalisé ces diagrammes, nous avons configuré notre projet sur GitHub directement et ne plus devoir s'en soucier après.
- Nous avons ensuite créé nos packages et les classes paramètres ainsi que le scrutin UnTour pour faire nos premiers tests.
- Comme nous avons décidé de faire l'extension des tests unitaires, il fallait l'implémenter dès le début pour nous permettre de faire des tests plus approfondis. Nous avons fait le bon choix car cette extension nous a permis de gagner du temps et de nous rassurer sur nos différentes fonctions.
- Nous avons ensuite codé la possibilité de faire évoluer les opinions en fonction de différents critères et l'implémentation des autres scrutins.
- Après avoir codé toutes ces fonctionnalités, nous avons mis en place une interface textuelle qui va nous permettre de faire des tests à une plus grande échelle mais aussi pour rendre notre application utilisable.
- Nous avons ensuite ajouté toutes les extensions proposées sauf l'interface graphique pour l'analyse des modes de scrutin que nous avons trouvés assez floue à réaliser.
- Pour conclure, nous avons testé toutes les possibilités et nous avons ajoutés les dernières corrections nécessaires.

2 - - Diagramme de cas d'utilisation



3 - Scénario de cas d'utilisation (Sans interface graphique)

Scénario nominal

1 - L'utilisateur va choisir comment il va paramétrer les paramètres

Lorsque l'utilisateur va ouvrir l'application, cette dernière va demander à l'utilisateur de choisir la manière dont il veut paramétrer les axes et les personnes parmi les deux suivantes :

1. Prendre les paramètres déjà existant sur le fichier de configuration
2. Créer lui-même de nouveaux paramètres

3 - L'utilisateur choisi ceux sauvegardés dans le fichier de configuration

Passage à l'étape 4

4 - L'utilisateur choisi de paramétrer lui-même l'application

3.1- L'utilisateur va créer N axes

L'application va demander le nombre d'axes à créer puis il devra saisir le nom des N axes.

3.2 - L'utilisateur va paramétrer les personnes

Après avoir paramétrer les axes, l'application va d'abord demander à l'utilisateur de générer des électeurs de manière aléatoire. L'application va lui demander de saisir le nombre d'électeur à générer.

Pour les candidats, l'application va demander de saisir le nombre de candidats et leurs noms. Pour le paramétrage, l'utilisateur peut :

- Soit générer les valeurs des axes de façon aléatoires
- Soit saisir lui-même les valeurs de chaque axe de chaque candidat

5 - Choisir son mode de scrutin

L'utilisateur va choisir le mode de scrutin parmi les cinq proposés :

- Scrutin majoritaire à un tour
- Scrutin majoritaire à deux tours
- Vote par approbation
- Vote alternatif
- Méthode de Borda

5 - Affichage d'un menu des fonctionnalités de l'application

Après avoir fait cette étape, un menu va s'afficher sur la console contenant dix options :

1. Réaliser un sondage
2. Evoluer des opinions
3. Simuler une élection
4. Analyser l'influence des modes de scrutin
5. Consulter liste électeurs
6. Consulter liste candidats
7. Consulter liste axes
8. Sauvegarder les résultats
9. Sauvegarder les paramètres dans le fichier de configuration
10. Quitter l'application

5.1 – L'utilisateur peut réaliser un sondage

Pour réaliser un sondage, l'utilisateur peut choisir un pourcentage de la population à interroger. Il peut choisir un pourcentage compris entre 20% et 80% (Pas pertinent d'interroger un trop petit échantillon ou un trop grand échantillon dans le cas d'un sondage). Une liste de candidats va être affichée par ordre du candidat le plus favorable au candidat le moins favorable. Le résultat du sondage va être sauvegardé. Si l'utilisateur choisi de faire un autre sondage, le résultat du précédent sondage va être sauvegardé.

5.2 – L'utilisateur peut faire évoluer des opinions

Il peut choisir parmi deux méthodes :

Via un autre électeur : Pour chaque électeur, l'application va lui attribuer un électeur ou un candidat. Si un électeur lui est attribué, celui-ci doit être de la même position géographique que lui. Si l'électeur est proche des préférences d'un autre électeur ou du candidat choisi, il va légèrement modifier ses préférences pour se rapprocher de ceux de son interacteur. Dans le cas contraire, les préférences de l'électeur vont être éloignée.

Via un sondage : Un sondage doit être réalisé au préalable. Si celui-ci n'a pas été réalisé, on va momentanément à l'étape 4.1 pour réaliser un sondage. Les préférences de l'électeur vont être modifiées grâce au sondage en 3 manières que l'utilisateur pourra choisir :

- Par discussion
- Par les idées
- Par la côte
- Par la moyenne

Après cela, on peut afficher les évolutions réalisées et retourner à l'étape 5 ou l'étape 10.

5.3 – L'utilisateur peut réaliser une simulation d'élection

L'utilisateur peut choisir son mode de scrutin parmi les cinq proposés dans l'énoncé. Les pourcentages de vote des candidats et le candidat élu seront affichés sur la console. En cas d'égalité, on fait évoluer les opinions par discussion et on refait une simulation jusqu'à obtenir un gagnant. Retour à l'étape 5 ou l'étape 10.

5.4 – L'utilisateur peut analyser l'influence du mode de scrutin

L'utilisateur peut voir l'influence des modes de scrutin. Retour à l'étape 5 ou l'étape 10.

5.5 – Consulter liste électeurs

La liste des électeurs va être affichée avec leurs informations. Retour à l'étape 5 ou l'étape 10.

5.6 – Consulter liste candidats

Idem pour les candidats. Retour à l'étape 5 ou l'étape 10.

5.7 – Consulter liste axes

Idem pour les axes. Retour à l'étape 5 ou l'étape 10.

5.8 – Sauvegarder les résultats

Les résultats sont sauvegardés dans un fichier Excel. Retour à l'étape 5 ou l'étape 10.

5.9 – Sauvegarder les paramètres dans le fichier de configuration

Les paramètres seront sauvegardés dans le fichier de configuration et va écraser automatiquement les anciens paramètres.

5.10 – Quitter l'application

Fin. Tous les paramètres et résultats ne seront pas sauvegardés en dehors du fichier de configuration et du fichier Excel.

Cas d'erreur

1-A Le fichier de configuration est introuvable ou vide

Afficher un message d'erreur et passage à l'étape 3.

3-1-A L'utilisateur saisi moins d'un axe ou ne saisit pas le nom d'un des axes

Affichage d'un message d'erreur et boucler jusqu'à ce que tout soit bon.

3-2-A L'utilisateur saisi moins de deux électeurs/candidats ou ne saisit pas le nom d'un des candidats

Affichage d'un message d'erreur et boucler jusqu'à ce que tout soit bon.

3-2-B L'utilisateur saisi une valeur d'un axe inférieur à 0 ou supérieur à 1

Affichage d'un message d'erreur et boucler jusqu'à ce que tout soit bon.

4-A L'utilisateur saisi une valeur inférieure à 1 et supérieure à 5 pour le choix des scrutins

Affichage d'un message d'erreur et boucler jusqu'à ce que tout soit bon.

5-A L'utilisateur va choisir un nombre non compris entre 1 et 8

Un message d'erreur va s'afficher et on redemande une saisie.

5-1-A L'utilisateur va saisir un nombre inférieur à 20 ou supérieur à 80.

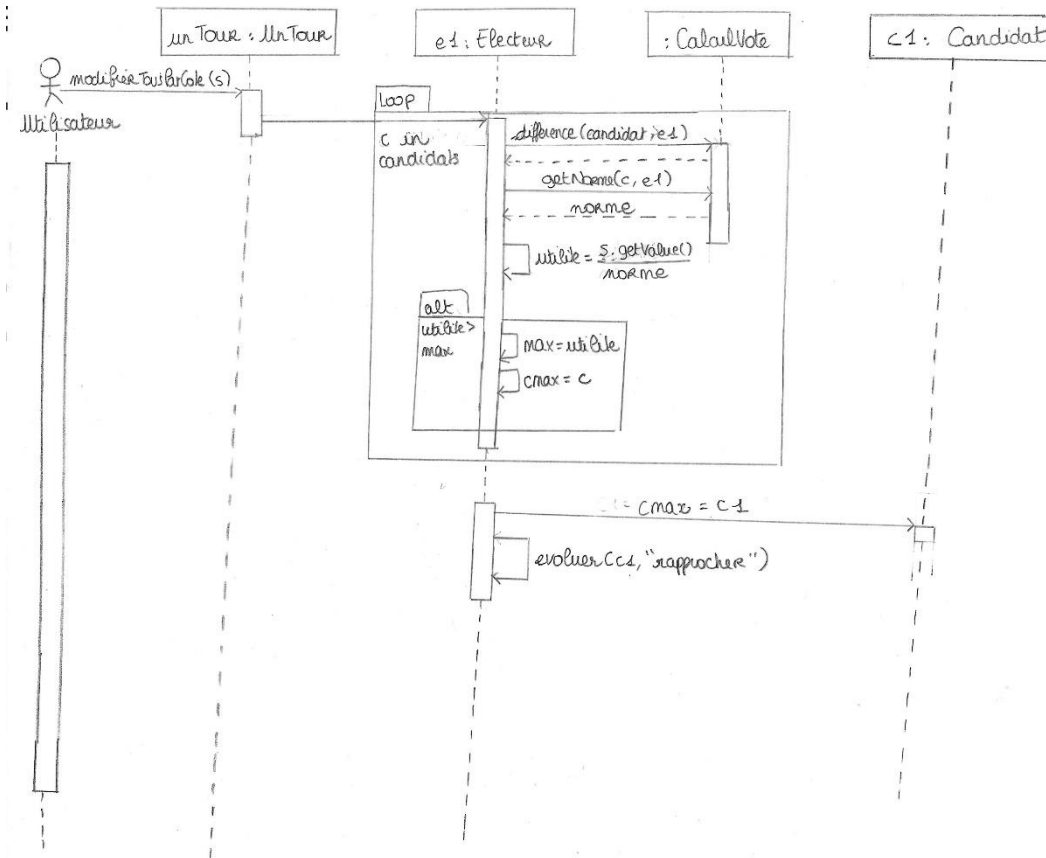
Afficher un message d'erreur et boucler jusqu'à que le nombre soit bon.

4 - Diagramme de classe

Diagramme de classe disponible sur le fichier « diagramme de classe.vpd »

5 - Diagramme de séquence

L'utilisateur veut faire évoluer les opinions par la côte à partir d'un sondage réalisé sur un vote en un tour.



6 - Justification des choix faits

1^{er} choix : Choisir un scrutin qui restera fixe jusqu'à la fermeture du fichier

Nous avons choisi de procéder ainsi car c'est plus simple à gérer au niveau de l'interface textuelle. Si l'utilisateur souhaite garder les mêmes paramètres mais qu'il souhaite changer de scrutin, il suffit qu'il sauvegarde les paramètres dans le fichier de configuration et ensuite, il relance l'application.

2^{ème} choix : Gestion des pourcentages dans le cas des scrutins suivant : DeuxTours et Alternatif

Comme se sont deux scrutins qui nécessitent de faire plusieurs tours, nous avons décidé de garder uniquement les résultats finaux et de mettre à 0% les candidats éliminés.

3^{ème} choix : Gestion des égalités

Lorsqu'il y a une égalité, nous avons choisi de procéder de la façon suivante :

- 1 - On prélève les candidats en première place
- 2 - On fait évoluer les opinions des électeurs par discussions
- 3 - On relance une simulation
- 4 - On répète ces opérations jusqu'à avoir un gagnant.

Ainsi, nous avons choisi cette solution du fait que c'est une de celle qui ressemble le plus à ce qu'on peut trouver dans la réalité.

4^{ème} choix : Utilisation de LinkedHashMap pour sauvegarder les résultats

Au lieu de créer une classe Sondage pour stocker nos résultats, nous avons voulu sauvegarder nos résultats dans une LinkedHashMap et les stocker dans scrutin. Nous avons préféré procéder ainsi du fait que nous considérons que pour réaliser un sondage, il suffit de réaliser une simulation d'élection à un pourcentage choisi de la population.

5^{ème} choix : Gestion des abstentions

Pour les abstentions, nous avons choisi de comparer les axes de chaque électeur avec tous les candidats pour extraire le candidat avec la norme la plus faible. Si la norme de l'électeur et du candidat est supérieure à $\sqrt{\text{nombre d'axes}} / \text{nombre d'axes}$, cela sera considéré comme une abstention

6^{ème} choix : Choix de l'équation pour la méthode évoluer de Electeur

Nous avons choisi de rapprocher ou d'éloigner les opinions de deux personnes d'un décalage de la moyenne des deux valeurs afin que la valeur de l'électeur ne dépasse pas celle de l'autre électeur ou du candidat.

7^{ème} choix : Choix de l'équation pour la méthode RapprocherUtilite de Electeur

Idem mais aussi pour rapprocher de façon proportionnelle pour chaque candidat.

7 - Ce que ce projet nous a apporté

- Une meilleure organisation et gestion des délais
- Des compétences solidifiées en JAVA et UML
- Découvertes d'extensions utiles notamment Git
- Un projet professionnel qui nous permettra de nous mettre en avant lors de notre recherche de stage et pour après.