



TELECOM Nancy

Projet PPII

Démocratie Participative

Poiron Léa
Peronnin Edgar
Laurent Noé-Laurent
Galkowski Anna

Responsables de module :
Festor Olivier
Oster Gérald



Table des matières

1	Introduction	4
1.1	Contexte du projet	4
1.2	Organisation du document	4
1.3	Présentation du projet	4
2	État de l'art	6
2.1	Civic Tech et Démocratie Participative	6
2.2	Concrètement	6
2.3	Tableau comparatif	6
3	Base de données	8
3.1	Tables	8
3.2	Forme normale de la base de données	10
4	Web	11
4.1	Documentation des différentes routes	11
5	Algorithmique	13
5.1	Documentation et complexité	13
6	Tests	16
7	Gestion de projet	20
7.1	Equipe de projet	20
7.2	Analyse du projet	20
7.2.1	Définition des objectifs	20
7.2.2	Analyse des risques : Matrice SWOT	21
7.3	Organisation du projet	21
7.3.1	Jalons du projet : GANTT	21
7.3.2	Répartition des tâches : Matrice RACI	22
7.4	Réunions	22
8	Bilan du projet	23
8.1	Bilan global	23
8.2	Bilan individuel	23
8.3	Temps de travail	25
9	Bibliographie	26
10	Annexes	27
10.1	Comptes rendus des réunions	28
10.1.1	6 novembre 2021	28
10.1.2	16 novembre 2021	29
10.1.3	23 novembre 2021	30
10.1.4	30 novembre 2021	31
10.1.5	7 décembre 2021	32
10.1.6	17 décembre 2021	33
10.1.7	23 décembre 2021	34

10.1.8	27 décembre 2021	35
10.1.9	30 décembre 2021	36
10.1.10	03 janvier 2022	37
10.1.11	05 janvier 2022	38

Chapitre 1

Introduction

1.1 Contexte du projet

Ce projet a été réalisé dans le cadre du PPII (Projet Pluridisciplinaire d’Informatique Intégrative) au programme de la première année de la formation initiale sous statut étudiant (FISE) du cycle ingénieur de TELECOM Nancy.

Le but est de concevoir et d’implémenter une application Web qui s’inscrit dans le thème de la démocratie participative. Elle doit pouvoir faciliter la démocratie participative locale.

La création de cette application a été réalisée en mobilisant nos connaissances en matière de gestion de projet, d’algorithmique, de bases de données et de développement Web.

1.2 Organisation du document

Le chapitre suivant est un état de l’art qui présente les différents sites et applications déjà existantes dans le domaine de la démocratie participative.

Le chapitre 3 est consacré à la conception de la base de données nécessaire au fonctionnement de notre application, le chapitre 4 à la partie développement Web et le chapitre 5 à l’algorithmique.

Le chapitre 6 présente les test réalisés sur les fonctions en python.

Le chapitre 7 est dédié à la gestion de projet. Nous y présenterons les différents outils utilisés pour le mener à bien.

Enfin dans un dernier chapitre nous allons effectuer un bilan général puis individuel du projet.

Certains document complémentaires concernant la gestion de projet seront disponibles en annexes.

1.3 Présentation du projet

Notre application Web, InterACTION, permet aux élus locaux de questionner les citoyens de manière directe en leurs soumettant des référendums sur une période choisie et en collecter les réponses. InterACTION permet également de récolter des informations (exemple : âge, catégorie socioprofessionnelle) sur les votants grâce à un système de login dans le but de déterminer le profil des votants et celui des non-votants. Cela permettrait par exemple aux élus d’identifier les populations qui ne parti-

cupent pas pour les sensibiliser à la prise de décision.

Chapitre 2

État de l'art

2.1 Civic Tech et Démocratie Participative

La Civic Tech représente l'ensemble des outils, procédés et techniques permettant de renforcer l'investissement des citoyens dans la vie politique de leur société ou communauté. La Civic Tech se développe depuis le milieu des années 2000 et l'avancée technologique ne peut qu'accroître ce développement.

La démocratie participative se repose sur la participation des citoyens dans l'exercice du pouvoir. Elle vise d'abord à embellir l'aménagement du territoire et tend à s'intéresser à l'environnement. Le concept de démocratie participative a vu le jour à la fin des années 60, suite à la remise en question de la démocratie représentative, démocratie dans laquelle seul un groupe de personnes décide pour tous.

La Civic Tech apparaît donc comme un moyen de développer la démocratie participative sur le plan technologique.

2.2 Concrètement

Dans les faits, la Civic Tech se trouve sous forme de logiciels, d'applications ou de sites internet permettant d'améliorer un plan de la démocratie participative. Ces différents supports peuvent viser uniquement une certaine catégories de citoyens, ou au contraire, essayer de regrouper le plus de personnes possibles. Par ailleurs, l'utilisation de ces outils peut être ponctuelle (au moment des élections par exemple) ou continue (dans le cas d'un fil d'information ...). L'échelle de l'outil peut être locale, départementale, régionale, nationale ou même mondiale. Enfin, si la majorité de ces applications sont gratuites pour pouvoir toucher le maximum de personnes, certaines sont payantes (généralement pour une partie de la population comme les collectivités).

Sont alors disponibles des applications de vote à différentes échelles, des applications d'informations sur l'actualité de la politique, des cartes interactives permettant de trouver diverses informations autour de chez soi, et beaucoup d'autre. En effet, la multitude de facteurs et de sujets permet de créer et de développer un grand nombre d'outils.

2.3 Tableau comparatif

Dans le tableau comparatif de la page suivante, nous pouvons constater que la plupart des sites et applications de la Civic Tech sont gratuits, continus, et tout public. Beaucoup des outils étudiés sont à l'échelle nationale. Par ailleurs nous pouvons voir que les thèmes abordés sont très diversifiés.

Nom	But	Public ciblé	Ponctuel ou Continu	Web ou Appli	Année de création	Echelle	Gratuit Payant
Voxe.org	Informatif		Continu	Web	2014	Nationale	Gratuit
La Boussole Electorale	Orienter vers un parti politique	Tout public	Continu	Web	2014	Mondiale	Gratuit
Agree	Vote en live durant les débats	Tout public	Ponctuel	Appli iOS	Avant 2015	Nationale	Gratuit
ConsultVox	Informatif Appel à idées Signaler un dysfonctionnement Donner son avis Débattre	Tout public avec des catégories	Continu		2015	National, Locale pour les questionnaires	Payant pour les collectivités, Gratuit pour les citoyens
Cmarue	Proposer l'installation de nouveaux commerçants	Tout public	Continu	Web		Nationale	Gratuit
Access-Man	Accès aux sites webs pour tous (handicapés)	Personnes handicapées	Continu	Web		Nationale	Gratuit
Purpoz	Pouvoir de co-construire des lois	Tout public	Continu	Web		Nationale	Gratuit
i-boycott	Création de campagne de boycott	Tout public	Continu	Web (campagnes) Appli de boycott	2015	National	Gratuit

Chapitre 3

Base de données

3.1 Tables

utilisateur

user_id	nom	prenom	date_naissance	sexe	cat_socio_pro	ville	dep	region	parent	email	mdp

clef primaire : user_id

user_id (entier) : chiffre unique attribué à l'utilisateur, monovalué et obligatoire

nom (texte) : nom de l'utilisateur, monovalué et obligatoire

prenom (texte) : prénom de l'utilisateur, monovalué et obligatoire

date_naissance (date au format MM/JJ/AAAA) : date de naissance de l'utilisateur, monovalué et obligatoire

sexe (texte) : sexe de l'utilisateur, monovalué et obligatoire

cat_socio_pro (texte) : catégorie socio-professionnelle de l'utilisateur, monovalué et obligatoire

ville (texte) : ville de résidence de l'utilisateur, monovalué et obligatoire

dep (texte) : numéro du département de résidence de l'utilisateur, monovalué et obligatoire

region (texte) : région de résidence de l'utilisateur, monovalué et obligatoire

parent (texte : oui ou non) : l'utilisateur est-il parent ou non, monovalué et obligatoire

email (texte) : email de l'utilisateur

mdp (texte) : mot de passe choisi par l'utilisateur pour accéder au site

elu

elu_id	nom	prenom	role	parti	ville	dep	region	email	mdp

clef primaire : elu_id

élu_id (entier) : chiffre unique attribué à l'élu, monovalué et obligatoire
 nom (texte) : nom de l'élu, monovalué et obligatoire, monovalué et obligatoire
 prenom (texte) : prénom de l'élu, monovalué et obligatoire
 role (texte) : rôle de l'élu, monovalué et obligatoire
 parti (texte) : parti politique auquel appartient l'élu, monovalué et obligatoire
 ville (texte) : ville de résidence de l'élu, monovalué et obligatoire
 dep (texte) : numéro du département de résidence de l'élu, monovalué et obligatoire
 region (texte) : région de résidence de l'élu, monovalué et obligatoire
 email (texte) : email de l'élu, monovalué et obligatoire
 mdp (texte) : mot de passe choisit par l'élu pour accéder au site, monovalué et obligatoire

referendum

ref_id	categorie1	categorie2	ville	dep	region	enonce	presentation	debut	fin	createur	titre

clef primaire : ref_id

ref_id (entier) : chiffre unique attribué à l'élu, monovalué et obligatoire
 categorie1 (texte) : catégorie à laquelle appartient le référendum, monovalué
 categorie2 (texte) : deuxième catégorie à laquelle appartient le référendum, monovalué
 ville (texte) : ville pour laquelle le référendum est soumis, monovalué et obligatoire
 dep (texte) : département de la ville pour laquelle le référendum est soumis, monovalué et obligatoire
 region (texte) : région de la ville pour laquelle le référendum est soumis, monovalué et obligatoire
 enonce (texte) : énoncé du référendum ; question fermée, monovalué et obligatoire
 presentation (texte) : présentation du contexte et informations importantes à propos du référendum, monovalué et obligatoire
 debut (date) : date à partir de laquelle le référendum est soumis aux citoyens, monovalué et obligatoire
 fin (date) : date à partir de laquelle le référendum n'est plus ouvert aux citoyens, monovalué et obligatoire
 createur (entier) : chiffre unique attribué à l'élu, monovalué et obligatoire
 titre (texte) : titre donné au référendum, monovalué et obligatoire

votes

ref_id	user_id	vote

clef primaire : ref_id, _id

ref_id (entier) : chiffre unique attribué à l'élu, monovalué et obligatoire clé étrangère
 user_id (entier) : chiffre unique attribué à l'utilisateur, monovalué et obligatoire clé étrangère
 vote (texte : oui ou non) : vote soumis par l'utilisateur

3.2 Forme normale de la base de données

Rappel

Une relation est en 1NF si chacun de ses attributs est atomique (non composé) et mono-valué.

Une relation R munie d'une clé primaire est en 2NF si

- elle est en 1NF
- tout attribut n'appartenant pas à la clé ne dépend pas d'une partie de la clé (DF partielle)

Une relation munie d'une clé primaire est en 3NF si

- elle est en 2NF
- tout attribut n'appartenant pas à la clé ne dépend pas d'un autre attribut n'appartenant pas à la clé

utilisateur

utilisateur(user_id, nom, prenom, date_naissance, sexe, cat_socio_pro, ville, dep, region, parent, email, mdp)

Chacun des attributs est atomique et monovalué, la relation utilisateur est donc en 1NF.

De plus, tout attribut n'appartenant pas à la clé ne dépend pas d'une partie de la clé, la relation utilisateur est donc en 2NF.

Cependant, la relation utilisateur n'est pas en 3NF car l'attribut région dépend de l'attribut département.

elu

elu(elu_id, nom, prenom, role, parti, ville, dep, region, email, mdp)

Chacun des attributs est atomique et monovalué, la relation élu est donc en 1NF.

De plus, tout attribut n'appartenant pas à la clé ne dépend pas d'une partie de la clé, la relation élu est donc en 2NF.

Cependant, la relation élu n'est pas en 3NF car l'attribut région dépend de l'attribut département.

referendum

referendum(ref_id, categorie1, categorie2, ville, dep, region, enonce, presentation, debut, fin, createur, titre)

Chacun des attributs est atomique et monovalué, la relation referendum est donc en 1NF.

De plus, tout attribut n'appartenant pas à la clé ne dépend pas d'une partie de la clé, la relation referendum est donc en 2NF.

Cependant, la relation referendum n'est pas en 3NF car l'attribut région dépend de l'attribut département.

vote

vote(ref_id, user_id, vote)

Chacun des attributs est atomique et monovalué, la relation vote est donc en 1NF.

De plus, tout attribut n'appartenant pas à la clé ne dépend pas d'une partie de la clé, la relation vote est donc en 2NF.

Finalement, tout attribut n'appartenant pas à la clé ne dépend pas d'un autre attribut n'appartenant pas à la clé. En effet, seul un attribut n'appartient pas à la clé, la relation vote est donc en 3NF.

Le fait que la plupart des tables ne soient pas en 3NF ne pose aucun problème au bon fonctionnement de l'application. Nous avons donc choisi de ne pas modifier la base de données.

Chapitre 4

Web

4.1 Documentation des différentes routes

Route	Commentaire
/	Redirige vers la route 'login'
login	Amène sur la page de login que l'on soit élu ou citoyen
deconnect	Permet de se déconnecter, renvoie à la route 'login'
voteoui/int :ref_id	Route amenant sur une page s'affichant après un vote 'Oui', si aucun vote n'a déjà été enregistré pour cet utilisateur, elle enregistre le vote et informe que le vote a bien été pris en compte sinon elle informe sur cette situation, elle permet aussi de retourner à l'accueil
votenon/int :ref_id	Route amenant sur une page s'affichant après un vote 'Non', si aucun vote n'a déjà été enregistré pour cet utilisateur, elle enregistre le vote et informe que le vote a bien été pris en compte sinon elle informe sur cette situation, elle permet aussi de retourner à l'accueil
inscriptcit	Page permettant au citoyen de s'inscrire sur le site en renseignant diverses informations
inscriptelu	Page permettant à l'élu de s'inscrire sur le site en renseignant diverses informations
registre	Route permettant de vérifier si toutes les informations nécessaires à l'inscription d'un citoyen sont bien remplies. Si tout est bien rempli, un message le notifiera. Sinon, un message d'erreur s'affichera.
registerc	Route permettant de vérifier si toutes les informations nécessaires à l'inscription d'un élu sont bien remplies. Si tout est bien rempli, un message le notifiera. Sinon, un message d'erreur s'affichera.
logincit	Route amenant sur la page de connexion pour les citoyens
logincitdeux	Route récupérant les données de 'logincit', les testant et redirige vers une page en fonction de la correspondance.
accueil_c/string :cat	Route de la page des référendums disponibles pour le citoyen
loginelu	Route amenant vers la page de connexion pour les élus

Route	Commentaire
logineludeux	Route récupérant les données de 'loginelu', les testant et redirige vers une page en fonction de la correspondance.
accueil_e/string :cat	Route de la page des référendums disponibles et du lien permettant de créer un référendum
accueil	Route redirigeant vers la page de login que l'on soit élu ou citoyen, où que l'on soit dans le site
resultatselu/int :ref	Route de la page des résultats d'un référendum (pourcentage oui/non plus les graphiques sur différentes caractéristiques)
resultatscit/int :ref	Route de la page des résultats d'un référendum (pourcentage oui/non seulement)
creationreferendum	Route de la page permettant de créer un référendum
refcree	Route amenant vers la page informant que le référendum a bien été créé et qui permet de retourner à l'accueil
referendum/int :ref_id	Route redirigeant vers la page du référendum souhaité
filtrecat	Route permettant de filtrer les référendums selon différentes catégories
filtreech	Route permettant de filtrer les référendums selon l'échelle
apropos	Route de la page contenant des informations sur le site
contact	Route amenant sur la page des contacts des développeurs
contactco	Route amenant sur la page des contacts des développeurs une fois connecté
credits	Route amenant sur les crédits du site
creditsco	Route amenant sur les crédits du site une fois connecté

Chapitre 5

Algorithmique

5.1 Documentation et complexité

Programme	Entrée	Sortie	Complexité
separeidtitre	liste de tuples dont le premier élément du tuple est l'id et le second le titre du référencum	liste de listes de taille 2 dont le premier élément de la sous-liste l'id et le second le titre du référencum	$O(n)$
pourcentage	2 entiers nombre de votes "oui" nombre de votes "non"	2 réels Pourcentage de votes "oui" Pourcentage de votes "non"	$O(1)$
graphe_sociopro	2 listes de longueur 10 liste dont le n-ième élément est le nombre de "oui" pour la n-ième catégorie liste dont le n-ième élément est le nombre de "non" pour la n-ième catégorie	Histogramme des votes en fonction de la catégorie socio-professionnelle	$O(n)$
retire_zero	liste d'entiers	liste dont on à retiré les 0 et les valeurs négatives	$O(n)$
catembert	liste de longueur 10 liste dont le n-ième élément est le nombre de vote pour la n-ième catégorie	Diagramme camembert donnant la répartition des votes en fonction de la catégorie socio-professionnelle	$O(n)$
max_indice	liste	Indice du plus grand élément de la liste	$O(n)$
calcul_age	chaîne de caractère : date au format AAAA/MM/JJ	Entier : âge	$O(1)$
age	liste de tuples contenant la date de naissance du votant ainsi que son vote	Histogramme des votes en fonction de la tranche d'âge	$O(n)$
camemb_age	liste de dates	Diagramme camembert donnant la répartition des votes en fonction de la tranche d'âge	$O(n)$

Programme	Entrée	Sortie	Complexité
parembert	liste de longueur 2 liste dont le n-ième élément est le nombre de vote pour la n-ième catégorie	Diagramme camembert donnant la répartition des votes en fonction de la situation familiale	$O(n)$
sexe	2 listes de longueur 3 liste dont le n-ième élément est le nombre de "oui" pour la n-ième catégorie liste dont le n-ième élément est le nombre de "non" pour la n-ième catégorie	Histogramme des votes en fonction du sexe	$O(n)$
camembert_s	liste de longueur 3 liste dont le n-ième élément est le nombre de vote pour la n-ième catégorie	Diagramme camembert donnant la répartition des votes en fonction du sexe	$O(n)$
traitement_sociopro	liste de tuples dont le premier élément est la catégorie socio-professionnelle de l'individu et le second son vote	2 listes : yes et no le n-ième élément de oui est le nombre de vote oui pour la n-ième catégorie socio-professionnelle de même pour no avec les votes non	$O(n)$
traitement_catembert	liste des catégories socio-professionnelles des votants	liste dont le n-ième élément est le nombre de votant pour la n-ième catégorie socio-professionnelle	$O(n)$
traitement_parents	liste de tuples dont le premier élément est la situation parentale de l'individu et le second son vote	2 listes : oui et non liste dont le premier élément est le nombre de parents ayant voté oui, le second le nombre de personnes sans enfant ayant voté oui de même pour non avec les votes non	$O(n)$
traitement_parembert	liste des situations familiales des votants	liste dont le premier élément est le nombre de votants parents et le deuxième le nombre de votants pas parents	$O(n)$
traitement_sexe	liste de tuples dont le premier élément est le sexe de l'individu et le second son vote	2 listes : oui et non le premier élément de oui est le nombre d'hommes ayant voté oui, le second le nombre de femmes ayant voté oui et le troisième le nombre d'individu "neutre" ayant voté oui de même pour non avec les votes non	$O(n)$

Programme	Entrée	Sortie	Complexité
traitement_ camembert_s	liste des sexes de votants	liste dont le 1er élément est le nombre de votants hommes, le second le nombre de votants femmes et le dernier le nombre de votants neutres	$O(n)$
parents	2 listes de longueur 2 liste dont le 1er élément est le nombre de "oui" chez les parents et le deuxième est le nombre de "oui" chez les personnes sans enfants liste dont le 1er élément est le nombre de "non" chez les parents et le deuxième est le nombre de "non" chez les personnes sans enfants	Histogramme des votes en fonction de la situation familiale	$O(n)$

Chapitre 6

Tests

Méthode

Les tests des algorithmes ont été réalisés à l'aide de pytest.

1. `separeidtitre`

Tests :

- Avec une liste classique, la fonction renvoie bien le bon résultat.
- Avec une liste contenant un élément qui n'est pas un tuple, la fonction ne le prends pas en compte.
- Avec une liste vide, la fonction renvoie une liste vide.

2. `pourcentage`

Tests :

- Avec 1 "oui" et 4 "non" : on a bien 20% et 80% comme résultats.
- Avec 1 "oui" et 1 "non" : on a bien 50% pour chaque.
- Avec 0 réponses : la fonction renvoie bien (0,0).
- Lorsqu'une des valeurs entrées est "None" : la fonction renvoie bien (0,0).

3. `retire_zero`

Tests :

- Avec une liste comportant 1 zéro : celui-ci est bien retiré.
- Avec une liste comportant 1 valeur négative : celle-ci est bien retirée.
- Avec une liste comportant uniquement des zéros : la fonction retourne une liste vide.
- Avec une liste vide : la fonction renvoie la même liste.
- Avec une liste ne possédant pas de zéro : la fonction renvoie la même liste.

5. `max_indice`

Tests :

- Avec une liste ayant 1 élément maximal : le bon indice est renvoyé.
- Avec une liste d'éléments identiques ou ayant plusieurs éléments maximaux : la fonction renvoie bien l'indice du premier élément ayant la valeur maximale.
- Avec une liste vide : la fonction renvoie bien None.

6. calcul_age

Tests :

- Avec un anniversaire déjà passé dans l'année : le bon âge est renvoyé.
- Avec un anniversaire qui arrive dans l'année : le bon âge est également renvoyé.
- Avec une date de naissance située dans le futur : la fonction renvoie bien 0.

7. traitement_sociopro

Tests :

- Avec une liste vide en entrée, la fonction retourne deux listes composées de 10 zéros.
- Avec une liste classique, la fonction retourne deux listes correctes.
- Avec une liste comportant une catégorie socioprofessionnelle non existante, la liste retournée ne comporte pas son vote.
- Avec une liste comportant un vote qui n'est ni "Oui" ni "Non", la liste retournée ne comporte pas non plus son vote.

8. traitement_catembert

Tests :

- Avec une liste classique, la fonction retourne une liste correcte.
- Avec une liste vide, la fonction retourne une liste remplie de zéros.
- Avec une liste qui contient une catégorie socioprofessionnelle qui n'existe pas, celle-ci n'est pas comptée dans la liste retournée.

9. traitement_parents

Tests :

- Avec une liste vide, la fonction retourne deux listes comprenant des zéros.
- Avec une liste classique, la fonction retourne deux listes correctes.
- Avec une liste qui contient un vote non existant, la fonction ne compte pas ce vote.
- Avec une liste qui contient quelque chose de différent de "Parents" ou "Pas parents", la fonction ne le compte pas.

10. traitement_parembert

Tests :

- Avec une liste classique, la fonction retourne une liste correcte.
- Avec une liste possédant un élément différent de ceux possibles, la fonction ne le compte pas.
- Avec une liste vide, la fonction retourne une liste remplie de zéros.

11. traitement_sexe

Tests :

- Avec une liste classique, la fonction retourne un tuple de listes correctes.
- Avec une liste possédant un élément différent de ceux possibles, la fonction ne le compte pas.
- Avec une liste vide, la fonction retourne une liste de tuples remplis de zéros.

12. traitement_camembert_s

Tests :

- Avec une liste classique, la fonction retourne une liste correcte.
- Avec une liste possédant un tuple avec un élément différent de ceux possibles, la fonction ne le compte pas.

- Avec une liste vide, la fonction retourne une liste remplie de zéros.

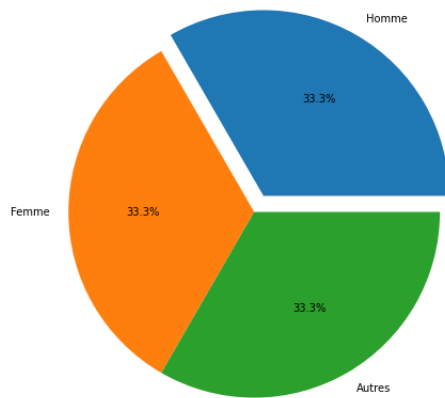
13. camembert_s

Tests :

- Avec une liste de 0

Aucune donnée disponible pour ce référendum

- Avec la liste [1,1,1]



- Avec la liste [1,1]
"la longueur de la liste est incorrecte"

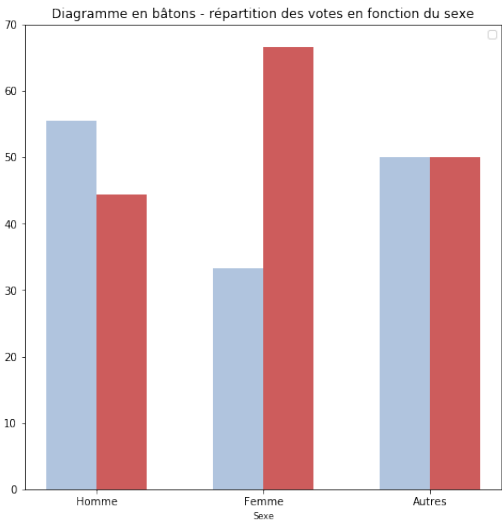
14. sexe

Tests :

- Avec une liste de 0

Aucune donnée disponible pour ce référendum

- Avec les listes [5,3,1],[4,6,1]



— Avec les listes [1,1],[1,1]
"la longueur de la liste est incorrecte"

Chapitre 7

Gestion de projet

7.1 Equipe de projet

L'équipe de projet se compose de 4 étudiants en première année :

- Galkowski Anna
- Laurent Noé-Laurent
- Peronnin Edgar
- Poiron Léa

Léa est désignée chef de projet, elle a la responsabilité d'animer les réunions et de suivre l'avancée du projet.

L'équipe s'est réunie une fois par semaine minimum à partir du choix de l'application à développer afin de définir les objectifs et de se répartir le travail. Durant les vacances scolaires ces réunions ont eu lieu par le biais de la plateforme Discord.

Les documents relatifs aux projet ont principalement été rédigés sur le serveur leaf de l'école. Cependant le groupe a également accès à un drive contenant les autres documents.

7.2 Analyse du projet

7.2.1 Définition des objectifs

Les objectifs ont été définis à l'aide de la méthode SMART :

S	Spécifique	L'objectif est simple et précis
M	Mesurable	L'objectif est vérifiable de manière qualitative ou quantitative
A	Accepté	L'objectif doit être défini avec la personne qui le réalise et non imposé
R	Réaliste	L'objectif doit être motivant sans se mettre en situation d'échec
T	Temporellement défini	L'objectif doit être inscrit dans le temps, avec une échéance précise

7.2.2 Analyse des risques : Matrice SWOT

Nous avons résumé les risques et les avantages de notre projet grâce à une matrice SWOT (Strength, Weaknesses, Opportunities, Threats), présentée en figure 7.1.

	Positif	Négatif
Interne	-Grande motivation de l'ensemble de l'équipe, -Les statistiques en algorithmie seront bien gérées	-Aucune expérience en web chez tous les membres du projet
Externe	-Possibilité de demander de l'aide aux professeurs face aux difficultés	-Temps limité (-6 semaines)

FIGURE 7.1 – Matrice SWOT

7.3 Organisation du projet

Le projet se déroule du mois de novembre 2021 jusqu'au début du mois de janvier 2022.

7.3.1 Jalons du projet : GANTT

Nous avons réalisé un diagramme de GANTT afin d'organiser notre travail. C dernier est présenté en figure 7.2.

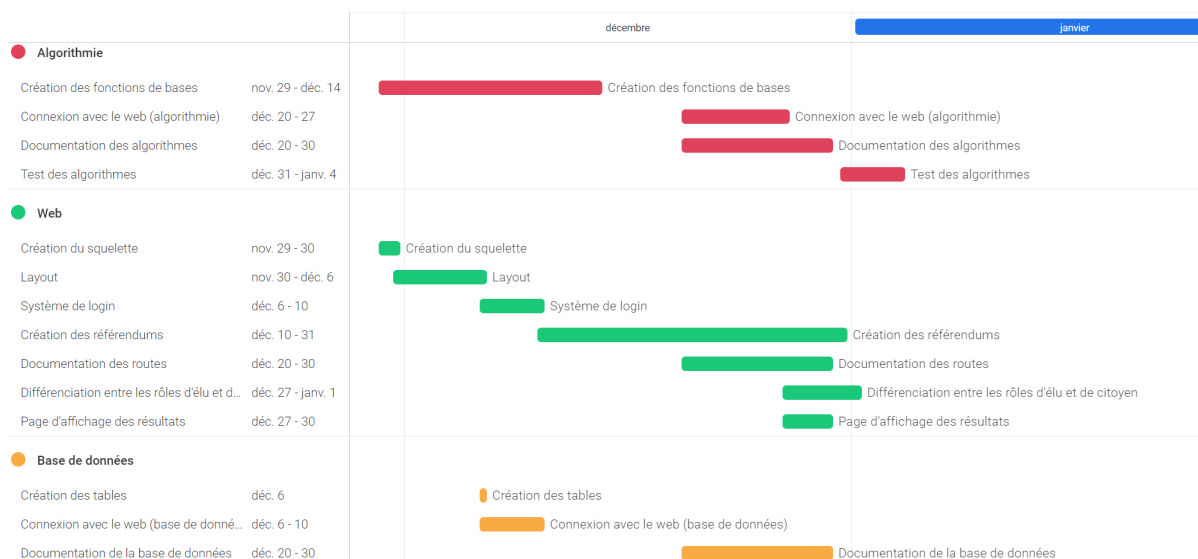


FIGURE 7.2 – GANTT

7.3.2 Répartition des tâches : Matrice RACI

Membres	Anna Galkowski	Noé-Laurent Laurent	Edgar Peronnin	Léa Poiron
Algo : Fonctions de bases	C I	C I	R	R A
Web : Création du squelette	R	R	A C I	C I
BD : Création des tables	C I	A C I	C I	R
Web : Système d'inscription avec un rôle	R	R	C I	A C I
Connexion Web / BD	A C I	R	C I	R
Web : Traitement des erreurs de saisie	C I	C I	A C I	R
Web : Système de login	A C I	R	C I	C I
Web : Création/Réponse de Référendum	R	A C I	C I	C I
Web : Pages pour les référendums	A C I	C I	C I	R
Web : Proposer uniquement les référendums correspondant à l'utilisateur connecté	A C I	C I	R	R
Web : Layout	R	R A	A C I	C I
Connexion Web / Algo	C I	R	A C I	C I
Page de consultation des résultats	C I	R	R	A C I
Test des algorithmes	R	R A	R	R
Documentation des algorithmes	C I	A C I	C I	R
Documentation de routes	C I	C I	R	A C I
Documentation de la BD	C I	C I	A C I	R

7.4 Réunions

Les comptes rendus des réunions sont présentés en annexe.

Chapitre 8

Bilan du projet

8.1 Bilan global

	Points positifs	Difficultés
Partie Base de Données	La structure est assez simple et permet de répondre à toutes les demandes de l'application	Certaines contraintes n'ont été détectées que lors de la conception
Partie Web	Prise en main assez rapide du langage html, facilitée par Visual Studio Code	C'est un aspect nouveau pour tous les membres du groupe d'où un manque d'expérience. Le démarrage a donc été assez lent.
Partie Algorithmique	Algorithmes plutôt simples à écrire	Affichage des statistiques calculées parfois compliqué
Partie Gestion de Projet	Réunions régulières	La planification du projet s'est faite en grande partie au fur et à mesure

8.2 Bilan individuel

Léa Poiron

Points positifs	Equipe motivée et persévérante
Difficultés rencontrées	Horaires de travail pour ne pas être sur Git simultanément
Expériences personnelles	Travail d'équipe agréable et entraide Développement des compétences en Web, base de données gestion de projet
Axes d'amélioration	Mieux gérer la gestion de projet dès le début

Edgar Peronnin

Points positifs	La réalisation de ce projet, de sa naissance à sa conception fut très épanouissant.
Difficultés rencontrées	Quelques difficultés pour obtenir les visuels souhaités pour les graphiques.
Expériences personnelles	J'ai pu développer mes compétences en algorithmique, découvrir le web et l'utilisation de sqlite3.
Axes d'amélioration	Être plus efficace sur les tâches a priori plus faciles. Aussi, essayer de mieux anticiper les différents cas possibles en entrée d'un algorithme.

Noé-Laurent Laurent

Points positifs	L'avancement du projet a globalement bien avancé et la communication était bonne.
Difficultés rencontrées	Le travail simultané était compliqué avec Git.
Expériences personnelles	Devoir suivre un planning et des tâches fixes était nouveau pour moi et cela m'a permis d'avoir un avancement plus clair dans le projet. Meilleure compréhension du Web et bonne efficacité sur le développement.
Axes d'amélioration	Mieux détailler chaque étape au début du projet aurait pu contribuer à une meilleur fluidité du projet.

Anna Galkowski

Points positifs	Il y avait un vrai esprit d'entraide dans l'équipe
Difficultés rencontrées	Obtention d'un nouveau PC fonctionnel tard dans l'avancement du projet et travailler simultanément sur Git n'était pas possible
Expériences personnelles	J'ai pu m'améliorer en Web et en algorithmique.
Axes d'amélioration	Mieux gérer le temps pour réaliser des tâches pour éviter une quantité de travail trop importante sur la fin du projet.

8.3 Temps de travail

Semaines	Anna Galkowski	Noé-Laurent Laurent	Edgar Peronnin	Léa Poiron
du 01/11 au 07/11	1h30	1h30	1h30	1h30
du 15/11 au 21/11	2h	2h	2h	2h
du 22/11 au 28/11	2h	2h	2h	2h
du 29/11 au 05/12	2h	7h	2h	3h30
du 06/12 au 12/12	4h	7h	5h	6h
du 13/12 au 19/12	0h	3h	0h	1h
du 20/12 au 26/12	2h30	3h45	4h	9h
du 27/12 au 02/01	8h45	3h30	9h45	5h15
du 03/01 au 06/01	17h30	17h	14h45	13h30
Total	40h15	46h30	41h	43h45

Chapitre 9

Bibliographie

Etat de l'art

- <http://civict techno.fr/civic-theque/>
- <https://www.voxe.org/>
- <https://boussoleelectorale.com/>
- <https://apps.apple.com/us/app/agree/id1547610115>
- <https://www.consultvox.co/>
- <https://experience.cmarue.fr/>
- <https://www.access-man.com/en/home/>
- <https://purpoz.com/>
- <https://www.i-boycott.org/>

Développement

- <https://docs.python.org/fr/3/>
- <https://stackoverflow.com/>

Chapitre 10

Annexes

10.1 Comptes rendus des réunions

10.1.1 6 novembre 2021

Heure	Durée	Présentiel ou distanciel	Présent	Absent
14h30	1h30	Présentiel	Anna Galkowski, Noé-Laurent Laurent, Edgar Peronnin, Léa Poiron (secrétaire)	

Ordre du jour

- Choix du chef de projet
- Réalisation de l'état de l'art
- Brainstorming autour de la notion de "Démocratie de proximité"

Tâches réalisées

- Choix de Léa comme chef de projet
- Etat de l'art (presque terminé)

Brainstorming

- Eléments à définir :
 - public ciblé
 - échelle
- Idée : Création d'une application destinée aux ados pour en apprendre plus sur le fonctionnement de la démocratie à l'échelle locale grâce à des jeux.
- Problèmes soulevés :
 - temps de réalisation des jeux
 - intérêt moindre des jeunes pour ce genre d'application

A faire lors de la prochaine séance

- Réfléchir à des idées plus pertinentes
- Terminer l'état de l'art

10.1.2 16 novembre 2021

Heure	Durée	Présentiel ou distanciel	Présent	Absent
16h00	2h	Présentiel	Anna Galkowski, Noé-Laurent Laurent, Edgar Peronnin, Léa Poiron (secrétaire)	

Ordre du jour

- Finir l'état de l'art
- Choix d'un unique projet d'application web

Tâches réalisées

- Réflexion sur la réalisation d'une application permettant aux élus de consulter leurs administrés sous forme de 'référéndum'. Les résultats seront ensuite mis sous forme de graphiques représentant les votes en fonction de paramètres variés (ex : âge, sexe, catégorie socioprofessionnelle)
- Début de la rédaction du document de présentation du projet

A accomplir lors de la prochaine séance

- Finir la rédaction du document de présentation du projet

10.1.3 23 novembre 2021

Heure	Durée	Présentiel ou distanciel	Présent	Absent
17h00	30min	Présentiel	Anna Galkowski, Noé-Laurent Laurent, Edgar Peronnin, Léa Poirion (secrétaire)	

Ordre du jour

- Réflexion sur le sujet
- Structure de la base de données
- Structure de la page Web

Réflexion sur le sujet

- 1ère idée
Application qui prend vos centres d'intérêt et vos idées et vous renvoie une liste d'actions, de rôles, d'associations, de manifestations...
- 2ème idée
Application web permettant aux élus locaux de questionner les citoyens de manière directe en leurs soumettant des référendums sur une période choisie et en collecter les réponses. Le site web permettra également de récolter des informations (exemple : âge, catégorie socio-professionnelle) sur les votants grâce à un système de login dans le but de déterminer le profil des votants et celui des non-votants. Cela permettrait aux élus d'identifier les populations qui ne participent pas pour les sensibiliser à la prise de décision.

Tâches réalisées

Tout ce qui a été accompli par la suite concerne la deuxième idée évoquée ci-dessus.

- Réalisation de la structure de la base de données
- Réalisation de croquis représentant la future page web du point de vue de l'élus et du point de vue d'un habitant

10.1.4 30 novembre 2021

Heure	Durée	Présentiel ou distanciel	Présent	Absent
16h30	2h	Distanciel	Anna Galkowski, Noé-Laurent Laurent, Edgar Peronnin, Léa Poiron (secrétaire)	

Ordre du jour

- Qu'a t-on fait depuis la dernière réunion ?
- Construire les différents éléments de la gestion de projet
- Définir les tâches à accomplir pour la prochaine réunion

Dernière réunion

- Finaliser les derniers comptes-rendus
- Première approche de l'interface web
- Réflexion sur la gestion de projet

Gestion de projet

- 1ère version de la charte de projet
- Création du cahier des charges
- Création de la matrice SWOT
- Création des lots de travail
- Création de la matrice RACI
- Création du Gantt

Tâches pour la prochaine réunion

- Création des fonctions de base en algorithmie (Edgar)
- Création des tables de la base de données (Léa)
- Construction du squelette de la page web (Noé et Anna)

10.1.5 7 décembre 2021

Heure	Durée	Présentiel ou distanciel	Présent	Absent
13h00	15min	Présentiel	Anna Galkowski, Noé-Laurent Laurent, Edgar Peronnin, Léa Poiron (secrétaire)	

Ordre du jour

- Bilan du travail réalisé
- Difficultés rencontrées
- Travail à faire avant la prochaine réunion

Tâches réalisées en autonomie

- Création de la base de données (Léa)
- Premières pages pour le site Web avec un début de layout (Noé et Anna)
- Programmes de traitement des données (Edgar)

Tâches réalisées lors de la séance de travail du 6 décembre

- Créer la page d'inscription liée à la BD et les tests associés (Noé et Léa)
- Créer une page d'erreur qui énonce le problème de saisi (Léa)
- Créer la page de création de référendum (Anna)
- Avancer les programmes de traitement des données (Edgar)

Difficultés rencontrées

- Panne de machine virtuelle (Edgar)
- Affichage des graphiques en python (Edgar)
- Utilisation de CSS (Noé)

A faire

- Travail de vérification de ce qui a été écrit (cf matrice RACI)
- Finir les fonctions de traitement des données (Edgar et Noé)
- Créer le lien de retour à l'accueil (Léa)
- Supprimer les proposition lors de la connexion et mettre des points au lieu des caractères du mdp (Léa)
- Créer un layout pour les 2 premières pages (Anna)

10.1.6 17 décembre 2021

Heure	Durée	Présentiel ou distanciel	Présent	Absent
17h00	30min	Présentiel	Noé-Laurent Laurent, Léa Poiron (secrétaire)	Anna Galkowski, Edgar Peronnin

Absents : Edgar Peronnin (train) et Anna Galkowski

Ordre du jour

- Bilan du travail réalisé
- Difficultés rencontrées
- Travail à faire avant la prochaine réunion

Travail réalisé

- Travail de vérification
- Finir les fonctions de traitement des données (Edgar et Noé)
- Créer le lien de retour à l'accueil
- Supprimer les proposition lors de la connexion et mettre des points au lieu des caractères du mdp (Léa)

Difficultés rencontrées

- Affichage des graphiques en python

Travail à faire avant la prochaine réunion

- Travail de relecture de ce qui a été fait (cf matrice RACI)
- Mettre à jour le WBS (Anna)
- Finir la charte de projet (Léa)
- Créer un layout pour les 2 premières pages (Anna)
- Finir la partie algorithmique (Edgar et Noé)
- Finir les pages correspondants aux référendums (Léa)
- Afficher les données à la connexion (Léa)
- Relier la partie algorithmique au site web (Noé)
- Page d'affichage des résultats (Noé)
- Rectifier la base de données (Noé)

10.1.7 23 décembre 2021

Heure	Durée	Présentiel ou distanciel	Présent	Absent
16h00	30min	Distanciel	Anna Galkowski, Noé-Laurent Laurent, Edgar Peronnin, Léa Poiron (secrétaire)	

Ordre du jour

- Bilan du travail réalisé
- Difficultés rencontrées
- Travail à faire avant la prochaine réunion

Bilan du travail réalisé

- Travail de vérification
- Layout pour les 2 premières pages
- Nouvelle version de la charte de projet
- Avancée sur la partie Algorithmique
- Avancée sur les pages de référendum
- Affichage du nom et prénom à la connexion
- Page d’affichage des résultats
- Insertion d’algorithmique
- Rectifier la base de données

Difficultés rencontrées

- Panne d’ordinateur

Travail à faire avant la prochaine réunion

- Travail de relecture (cf matrice RACI)
- Mettre à jour le WBS (Léa)
- Finir les pages de référendum (Léa)
- Verifier les données rentrées à la création d’un référendum (Léa)
- Relire et compléter la charte de projet (Noé, Anna, Edgar)
- Finir de lier algorithmique et web (Edgar)
- Commencer le rapport de projet (Anna)
- Lier les catégories au référendums (Noé)

10.1.8 27 décembre 2021

Heure	Durée	ou distanciel	Présent	Absent
16h00	30min	Distanciel	Anna Galkowski, Noé-Laurent Laurent, Edgar Peronnin, Léa Poiron (secrétaire)	

Ordre du jour

- Bilan du travail réalisé
- Difficultés rencontrées
- Travail à faire avant la prochaine réunion

Bilan du travail réalisé

- Vérifier les données rentrées à la création d'un référendum
- Entrer les données de référendums dans la bd
- Lier les catégories au référendums
- Layout premières pages
- Première approche du rapport de projet

Difficultés rencontrées

- Panne de VM pendant une journée (Anna)

Travail à faire avant la prochaine réunion

- Travail de relecture
- Ajouter une table à la base de données (Edgar)
- Finir de lier algorithmique et web (Edgar)
- Retravailler l'affichage des résultats (Noé)
- Documentation des routes (Edgar)
- Documentation des algorithmes (Léa)
- Verifier les entrées dans les algorithmes (Léa)
- Test des algorithmes (Anna)
- Continuer le rapport de projet (Anna)
- Créer le fichier requirement (Noé)

10.1.9 30 décembre 2021

Heure	Durée	Présentiel ou distanciel	Présent	Absent
15h00	30min	Distanciel	Anna Galkowski, Noé-Laurent Laurent, Edgar Peronnin, Léa Poiron (secrétaire)	

Ordre du jour

- Bilan du travail réalisé
- Difficultés rencontrées
- Travail à faire avant la prochaine réunion

Bilan du travail réalisé

- Travail de relecture
- Ajouter une table à la base de données
- Finir de lier algorithmique et web
- Retravailler l’affichage des résultats
- Documentation des routes
- Documentation des algorithmes
- Verifier les entrées dans les algorithmes
- Début de test des algorithmes
- Commencer le rapport de projet
- Faire le fichier requirement
- Choix du nom de l’application

Difficultés rencontrées

- aucune

Travail à faire avant la prochaine réunion

- Travail de relecture (cf matrice RACI)
- Rectifier la base de données (Anna)
- Documentation de la base de données (Léa)
- Finir les tests des algorithmes (Anna)
- Rectification du travail entre web et algo (Edgar)
- Dissociation élu - citoyen en Web (Noé)
- Rapport de projet (Tout le monde)

10.1.10 03 janvier 2022

Heure	Durée	Présentiel ou distanciel	Présent	Absent
9h10	30min	Présentiel	Anna Galkowski, Noé-Laurent Laurent, Edgar Peronnin, Léa Poiron (secrétaire)	

Ordre du jour

- Bilan du travail réalisé
- Difficultés rencontrées
- Travail à faire avant la prochaine réunion

Bilan du travail réalisé

- Travail de relecture
- Rectifier la base de données
- Documentation de la base de données (presque terminé)
- Avancée sur les tests des algorithmes
- Rectification du travail entre web et algo
- Dissociation élu - citoyen en Web
- Avancée sur le rapport de projet

Difficultés rencontrées

- Connexion dans le train (Léa)

Travail à faire avant la prochaine réunion

- Travail de relecture (cf Matrice RACI)
- Finir le rapport de projet (Tout le monde)
- Dissocier les régions (Léa et Noé)
- Finir les tests des algorithmes (Anna)
- Tester chacun un algorithme (Léa, Edgar, Noé)
- Remplir la base de données (Edgar)

10.1.11 05 janvier 2022

Heure	Durée	Présentiel ou distanciel	Présent	Absent
10h10	15min	Présentiel	Anna Galkowski, Noé-Laurent Laurent, Edgar Peronnin, Léa Poiron (secrétaire)	

Ordre du jour

- Bilan du travail réalisé
- Difficultés rencontrées
- Travail à faire avant la prochaine réunion

Bilan du travail réalisé

- Travail de relecture (cf Matrice RACI)
- Finir le rapport de projet (Tout le monde)
- Dissocier les régions (Léa et Noé)
- Finir les tests des algorithmes (Anna)
- Tester chacun un algorithme (Léa, Edgar, Noé)
- Remplir la base de données (Edgar)

Difficultés rencontrées

- Modification de la base de données non enregistrée

Travail à faire avant le rendu

- Relecture du rapport (Tout le monde)
- Création d'un référendum pour la démonstration (Noé)
- Relecture de tout les documents de gestion de projet (Léa)
- Vérifier l'absence de bug (Tout le monde)