



PPII 2021/2022

Rapport de projet de PPII

Tom BENE
Alexandre DESCAMPS
Camille MOUSSU
François MERAT

Responsables de module :
Olivier Festor
Gerarld Oster



Table des matières

1	Introduction	4
1.1	Contexte	4
1.2	Plan	4
2	État de l'art	5
2.1	Démocratie participative	5
2.2	Discussion avec un élu	5
2.3	Les produits déjà existants	6
2.3.1	La Civic Tech	6
2.3.2	Algorithmes de tris	6
2.3.3	Comparaison des différents services existants	7
2.4	Les problèmes possibles	8
3	Gestion de projet	9
3.1	Equipe de projet	9
3.2	Analyse du projet	9
3.2.1	Définition des objectifs	9
3.2.2	Analyse des risques : Matrice SWOT	10
3.3	Organisation du projet	10
3.3.1	Répartition des tâches : Matrice RACI	11
3.3.2	Gantt	11
3.4	Outils de travail	11
4	Conception et réalisation	13
4.1	La base de donnée	13
4.1.1	Structure de la base	13
4.1.2	Schéma de la base de données	14
4.1.3	Le lien entre le serveur et la base de donnée	14
4.2	Le site	15
4.2.1	Web sous Flask	15
4.2.2	Les pages HTML	15
4.2.3	Design et mise en forme	17
4.3	L'algorithme de tri par pertinence	18
4.3.1	Le principe	18
4.3.2	La complexité	18
5	Tests et performances	19

6	Bilan du projet	20
6.1	Bilan global du projet	20
6.2	Bilan du projet par membre	21
6.2.1	Tom BENE	21
6.2.2	Alexandre DESCAMPS	21
6.2.3	François MERAT	21
6.2.4	Camille MOUSSU	22
6.3	Travail réalisé	22
7	Annexe	23
7.1	Comptes-rendu des réunions	23
7.1.1	7 novembre 2021 9h30	23
7.1.2	24 novembre 2021 13h	24
7.1.3	28 novembre 2021	25
7.1.4	5 décembre 2021	26
7.1.5	22 décembre 2021 11h	27
7.1.6	27 décembre 2021 17h	28
7.2	Bibliothèques Python utilisées	29

Chapitre 1

Introduction

1.1 Contexte

Ce projet a été réalisé dans le cadre de notre 1er semestre de première année du cycle ingénieur sous statut étudiant de TELECOM Nancy.

L'objectif est de concevoir une application facilitant la démocratie participative locale en s'appuyant sur une base de données, des algorithmes de traitement avancés et être accessible via le Web.

Le travail s'est décomposé en quatre parties : la mise en place d'un site accessible à tous, d'une base de donnée cohérente, la conception d'algorithmes de tris des données et la gestion du projet en équipe.

1.2 Plan

Dans le chapitre 2, nous présentons les outils de la démocratie participative ainsi qu'une analyse des application déjà existantes sous la forme d'un état de l'art.

Dans le chapitre 3, nous présentons les éléments ainsi que les outils de gestion de projet que nous avons utilisés.

Dans le chapitre 4, nous présentons la conception de notre site, la base de donnée associée et l'algorithme réalisé. Nous analyserons aussi sa complexité.

Dans le chapitre 5 nous verrons les tests effectués et les performances de notre travail

Dans le chapitre 6, nous réalisons un bilan du projet, d'un point de vue personnel et global.

Chapitre 2

État de l’art

2.1 Démocratie participative

La démocratie participative comprend l’ensemble des actions qui incluent les citoyens aux processus de décisions politiques. Il existe plusieurs façons de l’appliquer dans le monde informatique, notamment par des applications dédiées.

Les outils de démocratie participative permettent une interaction entre les institutions et les citoyens. De la consultation citoyenne à l’implication directe des citoyens, les institutions ont besoin de différents moyens pour faire participer la population. [7]

2.2 Discussion avec un élu

Résumé :

La mairie a deux besoins fondamentaux pour la démocratie participative :

- le *push* : envoyer des informations vers les administrés
- le *pull* : faire remonter les contributions et avis vers ceux qui prennent les décisions

Compte-rendu :

On commence par donner quelques exemples de démocratie participative tels que les réunions de quartier et les difficultés qui leur sont liées :

- Programmer ces réunions
- Permettre à ceux qui ne peuvent pas venir de participer
- Préparer les réunions en avance

On aborde aussi la possibilité de diffuser le bulletin municipal via ce genre de service même si ce ne serait pas le principal intérêt d’une telle application car le bulletin peut être diffusé autrement. Finalement on aborde aussi la question de la demande des contributions.

Pour cet aspect il est mentionné l’utilisation de forums qui a été vite abandonnée car trop difficiles à modérer. En effet, la municipalité veut avoir une maîtrise du débat afin d’éviter les disputes entre voisins qui n’apportent rien à la demande de contribution. Il s’agit donc de pouvoir relire les messages avant de les reposter à la vue de tous (mais on ne modifie pas les messages pour éviter toute censure)

Le véritable problème vient quand il s’agit de faire descendre l’information. Très peu de personnes s’inscrivent sur les newsletters par mail et il n’est pas garanti qu’elles les lisent de manière régulière. L’avantage d’une application c’est la possibilité d’envoyer des notifications. C’est cette partie qui est essentielle au “push”.

On ajoute que les mairies n'ont pas forcément un budget important. On regarde ensuite l'exemple d'une commune voisine qui a voulu utiliser l'application intra-muros mais cela a été un échec probablement dû au manque de participation. On peut conclure que la cible d'une telle application sont les communes ne possédant aucune compétence informatique car il s'agit d'une solution toute faite avec potentiellement un service de maintenance compris dans le contrat.

Conclusion :

Le problème le plus important qui a été soulevé est celui des notification, il est donc important de porter notre attention sur ce sujet car il représente une partie importante de la démocratie participative.

2.3 Les produits déjà existants

2.3.1 La Civic Tech

La Civic Tech ou technologie civique représente tous les outils technologiques et procédés impliqués dans la mise en place de la démocratie participative. La Civic Tech a des acteurs différents, les associations, les collectifs, les sociétés, les ONGs et les citoyens. Elles sont généralement commandées par les institutions pour leur permettre d'améliorer la participation des citoyens dans la politique. [9], [2]

2.3.2 Algorithmes de tris

Les algorithmes de tris sont récurrents sur Internet. La pertinence est décidée grâce à des critères pondérés qui permettront donc le classement des données.

2.3.3 Comparaison des différents services existants

	Intra-Muros [6]	Decidim [4]	Maire et Citoyens [8]
Calendrier des événements	Avec ordre de priorité selon la proximité	non	carte des services de la commune et des points clés
Carte avec les points d'intérêts de la commune	oui	non	non
Annuaire	oui	non	annuaire des services
Système de signalisation	oui	possible	non
Sondages et enquêtes	sondages	sondages	sondages
Demander des propositions	non	demande de propositions	consultation citoyenne
Fil d'actualités	celui de la commune + ceux spécifique des différentes associations ainsi que des établissements scolaires	oui	oui
Informations sur la commune	Liste et informations des commerces	pages et blog	Présentation des chiffres officiels, secrets historiques, albums photos, liens utiles, présentation de l'équipe municipale, etc ...
Organisation de conférence / ateliers / rencontre	non	oui	non
Notification	oui	oui	notification push et email combiné
Espace de suggestion	non	texte participatif	boîte-à-idées
Espace d'interaction	non	commentaires	Espace de libre-échange, de discussion
Budget participatif	non	non	non
Système de vote intégré	non	oui	non
Autres fonctionnalités	non	système de tirage au sort des citoyen (pour former un jury par exemple)	système d'envoi de SMS groupés et messagerie privé pour utilisateur payant et alerte danger

	Agora [1]	Fluicity [5]	CitizenLab [3]
Calendrier des événements	oui	invitation à des événements spécifique	non
Carte avec les points d'intérêts de la commune	cartes participatives	géolocalisation des proposition sur un plan	géolocalisation des idées sur un plan
Annuaire	non	non	non
Système de signalisation	oui	oui	non
Sondages et enquêtes	sondage	enquêtes	les 2
Demander des propositions	consultation citoyenne, proposition de projet	appel aux propositions, organisation de consultation sur des projets	proposition citoyenne
Fil d'actualités	oui	oui	partage d'information
Informations sur la commune	non	non	non
Organisation de conférence / ateliers / rencontre	atelier participatif	non	non
Notification	oui	oui	oui
Espace de suggestion	oui	non	oui
Espace d'interaction	commentaires	espace de concertation pour les citoyens	non
Budget participatif	non	oui	oui
Système de vote intégré	non	non mais outil pour organisation facilité des élections (de quartier)	oui
Autres fonctionnalités	vérification de l'identité	Statistique en temps réel, espace affilié, visio-conférence	Outils de comparaison de différent scénario puis proposition de vote aux citoyens

2.4 Les problèmes possibles

Les applications de démocratie participative se heurtent à plusieurs problèmes.

D'abord, le public ne les utilise pas forcément et donc leur utilité se retrouve réduite puisqu'il faut un certain nombre de participant pour que le principe de la démocratie participative s'applique.

Ensuite, lorsque l'application est un site internet, il est difficile de garder les utilisateurs à jour. Les notifications sont envoyées soit par mail soit avec en utilisant le navigateur mais uniquement lorsque le site est ouvert. Cela ne permet parfois pas d'impliquer les citoyens de façon suffisante pour que l'application prospère.

Chapitre 3

Gestion de projet

3.1 Equipe de projet

L'équipe se compose de quatres étudiants en première année :

- BENE Tom
- DESCAMPS Alexandre
- MERAT François
- MOUSSU Camille

François a été désigné chef de projet, il a eu la responsabilité d'animer les réunions et de suivre l'avancement du projet.

L'équipe s'est réunie régulièrement sur Discord le week-end pour le suivi du projet, de la répartition de la charge de travail et de la définition des objectifs.

3.2 Analyse du projet

3.2.1 Définition des objectifs

Suite à nos recherches d'applications similaires (à l'échelle communale) et à l'établissement de notre état de l'art, nous avons opté pour une application permettant à une mairie de consulter l'avis des citoyens.

Pour commencer, la mairie et les utilisateurs peuvent consulter les articles des organisations autorisées à publier telles que les syndicats et autres associations. Puis, les utilisateurs en général peuvent voter sur les projets proposés. La mairie peut ainsi savoir ce qui plaît aux citoyens.

Pour faciliter l'accessibilité au contenu, nous mettrons en place un système de tri par pertinence qui permettra aux utilisateurs de consulter en priorité les articles susceptible de les intéresser.

3.2.2 Analyse des risques : Matrice SWOT



FIGURE 3.1 – Matrice SWOT

3.3 Organisation du projet

Le projet prend place du mois de novembre 2021 jusqu'au début du mois de janvier 2022. Nous avons décomposé les fonctionnalités en tâches et les avons réparties au fur et à mesure de la progression du projet.

3.3.1 Répartition des tâches : Matrice RACI

Nous avons mis au point une matrice RACI pour répartir les tâches équitablement.

	Tom	Camille	Alexandre	François
Gestion de projet	I	C	I	RA
Affichage page d'accueil	C	RA	R	R
Algorithme de pertinence	RA	I	R	I
Page de Recherche	C	RA	I	I
Page de Login	RA	R	I	I
Page de profil	I	I	RA	I
Recherche par mot clés	AC	R	I	I
Système de Tags	A	R	I	I
Votes	A	C	I	R
Base de Données	RA	R	R	R
Login	RA	I	I	I
Publication des projets	C	RA	I	I
CSS	C	RA	R	R

3.3.2 Gantt

A partir de la répartition des tâches de la matrice RACI, nous avons créé un diagramme de Gantt, représenté en figure 3.2.

3.4 Outils de travail

IDE :

L'ensemble de l'équipe a travaillé sur VisualStudio Code.

Partage du travail :

L'équipe a majoritairement utilisé le git fourni par l'école pour échanger le travail mais avait aussi un drive dédié pour certains fichiers de la gestion de projet et pour partager des documents lors de la mise en place du projet.

Rédaction du rapport :

Le rapport a été rédigé sur Leaf pour permettre à tout les membres de compléter leurs éléments simultanément.



Page 12

Chapitre 4

Conception et réalisation

4.1 La base de donnée

La base de donnée a été faite en utilisant sql-Alchemy. C'est un ORM permettant de simplifier la gestion de base de données et de réduire le temps de développement des fonctionnalités liées. Nous avons créé un fichier python déclarant le modèle de la base de données, puis créé un fichier en .db contenant des données de test, nous permettant de réinitialiser rapidement les données. La base de donnée est en 3ème forme normale.

4.1.1 Structure de la base

La base de donnée permet de stocker les articles, leurs tags, les votes et les utilisateurs du site. Pour cela on a utilisé plusieurs tables liées par des clés étrangères. Il y a donc :

- La table des utilisateurs : Elle permet de lister les utilisateurs avec leur nom, prénom, mail, rôle et statut (ainsi que le hash de leur mot de passe).
 - Le rôle permet de savoir l'implication de l'utilisateur dans la vie citoyenne (citoyen, syndicat, adjoint, ...)
 - Le statut permet de savoir si l'utilisateur a le droit de publier des projets ou des articles (User/Poster)
- La table des article/projets permettant de comptabiliser les votes pour et contre les projets et stockant toutes les données utiles pour l'affichage des articles/projets.
- La table des tags qui liste tous les tags disponibles pour faciliter l'accès aux tags
- La table des vote permettant de savoir si un utilisateur a voté sur un projet et quand il a voté

4.1.2 Schéma de la base de données

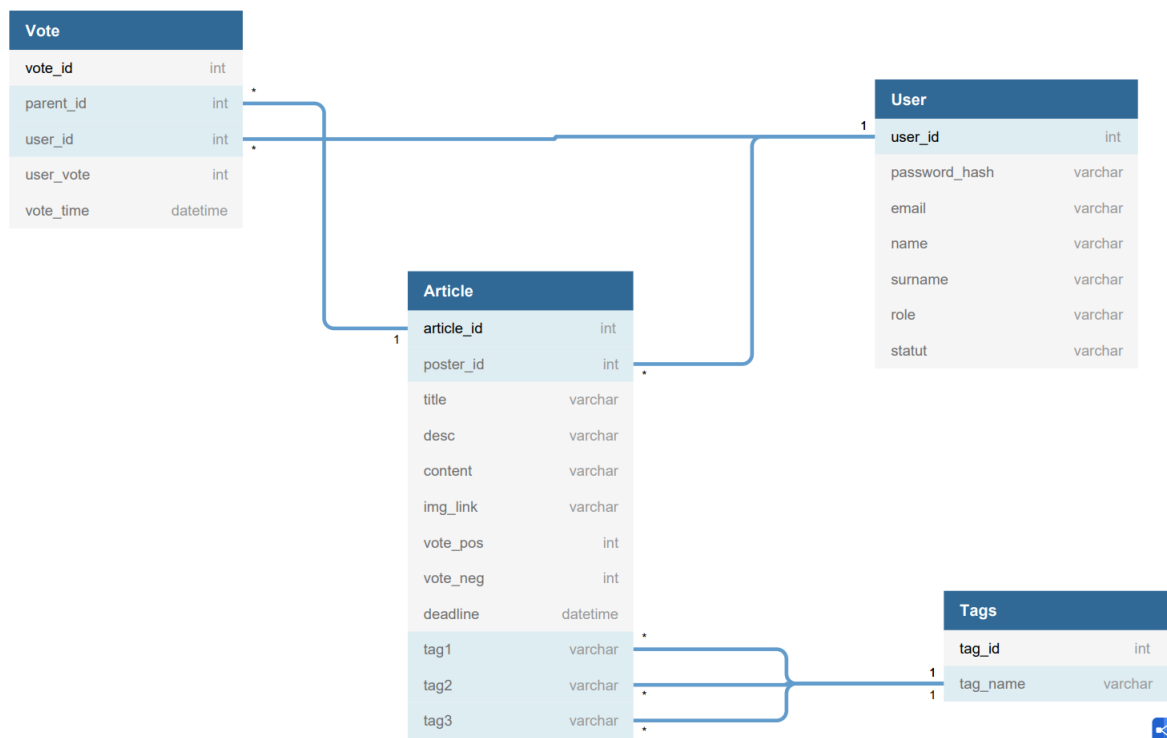


FIGURE 4.1 – Schéma relationnel

4.1.3 Le lien entre le serveur et la base de donnée

Dans le fichier app.py avec toutes les routes, on importe le module Flask-SQLAlchemy et le fichier modele.bd. Toute les requêtes sont faites avec le module. Toutes les données de test ont été entrées dans la base de données à l'aide du fichier requetesBD.py. Ainsi, les utilisateurs, les articles et les tags utilisés pour tester les fonctionnalités de l'application ont été créés dans ce fichier (on peut donc y consulter leurs propriétés).

4.2 Le site

Nous avons choisi de nommer notre site Ekklesia en référence au mot grec qui signifie assemblée.

4.2.1 Web sous Flask

Pour la mise en place du backend, on a utilisé un serveur web sous flask. Toutes les routes et les fonctions correspondantes sont stockées dans le fichier app.py.

4.2.2 Les pages HTML

Pour la mise en place du frontend de l'application on a utilisé des pages HTML mises dans un fichier template pris en compte par Flask. Cela permet au serveur d'afficher les bonne pages.

Le site permet aux utilisateurs de consulter des articles ou des projets. C'est pourquoi nous avons mis en avant les articles et leur sélection. Toutes les pages prennent appui sur et étendent la page de base.

Il y a plusieurs types de pages :

Les pages pour accéder à des fonctionnalités annexes

Page de création de compte (register) : Cette page nous permet d'ajouter un utilisateur à la base de donnée. Les utilisateurs enregistrés ont accès à plus de fonctionnalités.

Page de connexion : Cette page est proposée a chaque fois qu'un utilisateur veut effectuer des actions à l'accès restreint (publier ou voter).

Page de profil : Cette page permet au utilisateurs connectés d'accéder à leurs données et s'ils le désirent, de les modifier.

(a) Login

(b) Register

(c) Profil

FIGURE 4.2 – Les pages annexes aux principales fonctionnalités du projet

Les pages concernant les projets

Page de création de projet/article : Permet aux utilisateurs connectés d'ajouter un projet ou article à la base de donnée, avec un titre, un chapeau, le contenu de l'article, une image et jusqu'à 3 tags.

On a restreint les entrées pour que la plupart des éléments soient obligatoires pour pouvoir valider la publication.

(a) Publication d'un projet

(b) Erreur

FIGURE 4.3 – La page pour publier des projets

Page d'accueil : Propose les 3 projets les plus pertinents pour les utilisateurs connectés et les 3 plus récents pour ceux non connectés.

Page des projets : Permet aux utilisateurs de sélectionner un tag et affiche tous les articles avec ce tag. Ils peuvent être triés par pertinence ou par date

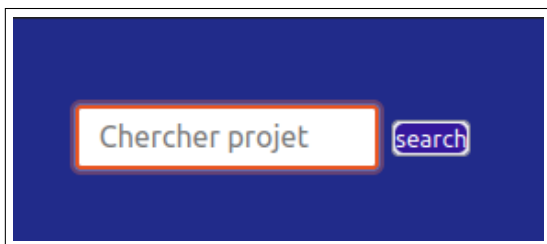
(a)

(b)

FIGURE 4.4 – Boite de recherche par tag

Affichage d'un projet : Permet aux utilisateurs de consulter un projet / article et voir le nombre de votes pour/contre celui-ci. Les utilisateurs connectés peuvent voter pour ou contre le projet.

La page de résultat de recherche : Les utilisateur sont dirigés sur cette page lorsqu'ils ont effectué une recherche dans la barre de recherche. S'il n'y a pas de résultat, l'utilisateur est envoyé sur une page l'informant de l'absence de résultats et lui proposant de consulter les autres articles.



(a) barre de recherche



(b) Pas de résultats

FIGURE 4.5 – Chercher des projets

4.2.3 Design et mise en forme

Le design a été fait en CSS dans un fichier style.css stocké dans un dossier static par convention Flask. Ce dossier sert aussi à stocker les images utilisées et télé-versées par les utilisateurs. Une image type (placeholder) est aussi stockée dans ce dossier. Elle nous a permis de vérifier le bon affichage des potentielles images télé-versées.

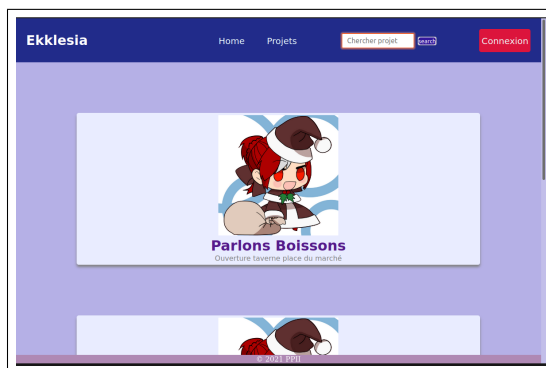


FIGURE 4.6 – Visuel du site

4.3 L'algorithme de tri par pertinence

4.3.1 Le principe

L'algorithme vise à trier les articles selon une liste de critères pondérés.

Quand l'utilisateur consulte, cherche ou vote pour un article les tags de l'article sont enregistré dans les cookies avec un coefficient d'importance différent selon l'action effectuée :

- Coef 3 : Pour les tags des articles où vote l'utilisateur
- Coef 2 : Pour les tags que recherche l'utilisateur
- Coef 1 : Pour les tags des articles qu'il visite suite à une recherche

Lorsque l'utilisateur consulte la page des projets, un score est assigné à chaque projet en fonction de ses tags et des habitudes de navigation de l'utilisateur. Plus ce score est important, plus les tags correspondant ont été visités par l'utilisateur. Les articles seront ensuite affichés par score décroissant.

4.3.2 La complexité

Cet algorithme a une complexité en $\Theta(n)$, avec n le nombre d'articles à classer.

Chapitre 5

Tests et performances

Chapitre 6

Bilan du projet

6.1 Bilan global du projet

Travail attendu	Travail réalisé
Algorithme de pertinence Suggestions en page d'accueil Cookies Système de tags Recherche lexicographique Système de vote (réaction) Stockage des données	Algorithme de pertinence Suggestion en page d'accueil Cookies persistants Liste de tag accessible pour que les utilisateurs cherchent les articles correspondants Recherche dans toutes les données de la BD des articles ceux qui contiennent les mots cherchés Système de vote Base de données contenant les utilisateurs, les posts et les tags
Travail secondaire	Travail secondaire réalisé
Login Boîte à idées Sondages	Login et possibilité de modification de profil X X

Justification de la différence :

- Les travaux secondaires n'ont pas été réalisés par manque de temps
- Certaines idées ont été trouvées et ajoutées en cours de réalisation et n'apparaissent donc pas sur la liste des attendus

	Points positifs	Points négatifs	Expérience
Gestion de projet	On a trouvé des outils pour la gestion de projet qui simplifient la mise en place des différents graphiques	Réunion parfois mal gérées surtout en début de projet	Il faut s'assurer de noter les CR à chaque réunion dans leur intégralité
Ecriture du code	L'application fonctionne et modifie la base de donnée	Parfois difficile de trouver d'où viennent les erreurs	

6.2 Bilan du projet par membre

6.2.1 Tom BENE

Points positifs	- Amélioration de mes connaissances en Web, particulièrement en front-end
Difficultés rencontrées	- Organisation difficile du à la première expérience de travail en groupe - Difficulté à former et impliquer certains membres sur les parties techniques du projet
Expérience personnelle	- J'ai affiné mes capacités en développement web et en base de données
Axes d'amélioration	- La répartition des tâches - L'accompagnement des personnes ayant moins de compétences techniques

6.2.2 Alexandre DESCAMPS

Points positifs	- Communication par Discord fluide et pratique - Soutien apporté par les autres membres du groupe
Difficultés rencontrées	- Difficultés techniques car premier maniement avancé de Flask et de SQLAlchemy - Manque d'autonomie pour les parties techniques
Expérience personnelle	- J'ai acquis énormément de compétences techniques en html/css et base de données (avec SQLAlchemy) - J'ai compris l'importance des outils de gestion de projet
Axes d'amélioration	- La gestion du temps et des deadlines a posé problème : Il aurait fallu s'atteler à la partie technique plus tôt -

6.2.3 François MERAT

Points positifs	- J'ai appris énormément sur les outils de développement - J'ai apprécié travailler avec des personnes motivées en qui je peux avoir confiance
Difficultés rencontrées	- Une faible implication de certains lors de la phase de définition qui a sans doute retardé le début du développement - Une mauvaise gestion du temps a entraîné des retards - Une absence de connaissance technique qui a demandé de maîtriser les outils avant de pouvoir travailler
Expérience personnelle	- J'ai appris l'utilisation du HTML, du CSS, de Flask et de SQLAlchemy - J'ai compris l'importance de la gestion de projet et appris l'utilisation des outils liés
Axes d'amélioration	- Avoir une meilleure gestion du temps pour éviter les retards - Être capable de définir des tâches plus réalistes et avoir une meilleure estimation du temps nécessaire à leur allouer

6.2.4 Camille MOUSSU

Points positifs	<ul style="list-style-type: none"> - J'ai apprécié travailler dans ce genre de projet et j'ai hâte de recommencer. - La communication était très régulière grâce à Discord
Difficultés rencontrées	<ul style="list-style-type: none"> - Gestion du temps très différentes entre chaque membre ce qui a causé du retard sur la progression du projet. - Une implication parfois faible de certains membres au début du projet a ralenti sa définition - Un manque d'autonomie de certain membres sur les parties techniques
Expérience personnelle	<ul style="list-style-type: none"> - J'ai amélioré mes capacités en Web, que ce soit du HTML, Flask et CSS - J'ai réalisé l'importance capitale de la gestion de projet pour le bon déroulement du projet
Axes d'amélioration	<ul style="list-style-type: none"> - La gestion du temps et des deadlines sont 2 points qui ont posé problème - Définir le projet aurait dû être fait plus rapidement et plus précisément pour pouvoir avoir plus de temps pour coder

6.3 Travail réalisé

Étapes	Tom	Alexandre	François	Camille
État de l'art	3h	5h	15h	5h
Conception de l'application				
Web	21h	18h	4h	23h
Algorithmes	10h	4h	3h	3h
Base de Données	4h	10h	2h	2h
Tests				5h
Gestion de projet	2h	2h	15h	3h
Secrétariat				
Comptes-rendus de réunion	1h	1h	5h	3h
Rédaction du rapport	12h	10h	10h	15h
TOTAL	53h	50h	54h	59h

Chapitre 7

Annexe

7.1 Comptes-rendu des réunions

7.1.1 7 novembre 2021 9h30

Présent	Absent
Camille MOUSSU Alexandre DESCAMPS Tom BENE François MERAT	Personne

Ordre du jour

1. Prise de contact avec les membres du groupe
2. Début de l'établissement de l'état de l'art
3. Établissement de la matrice SWOT
4. établissement des objectifs :
 - Envoyer des informations au citoyens
 - Recevoir des informations de citoyens
 - Faire interagir les citoyens entre eux
5. Répartition des tâches pour la phase de définition : BD et architecture du site

TODO-LIST

1. BD -> Camille
2. Algorithmes -> Tom
3. Architecture du site -> François et Alexandre

A faire avant le 23/11

7.1.2 24 novembre 2021 13h

Présent	Absent
Camille MOUSSU Alexandre DESCAMPS Tom BENE François MERAT	Personne

Ordre du jour

1. Récapitulatif de l'avancement
2. Répartition des tâches

Tâches effectuées

1. Vérification du travail effectué :
 - Structure de la base de donnée : OK
 - État de l'art : OK
 - Bilan écrit : OK
2. Utilisation de l'état de l'art pour établir les fonctionnalités de manière définitive
3. Planification pour la présentation pour la validation

TODO-LIST

1. Terminer la mise en forme de l'état de l'art (Alexandre)
2. Envoyer un mail pour la validation (Alexandre)
3. Lettre à jour la structure de la base de donnée selon les fonctionnalités définitives (Camille)
4. Faire l'architecture Web (Camille)
5. Lister les besoins en algorithmes (Tom)
6. Terminer le schéma de présentation des fonctionnalités à développer (François)
7. Faire un planning (Gantt) provisoire (François)

Avant la fin de semaine pour pouvoir le présenter à M.Festor ou M.Oster

7.1.3 28 novembre 2021

Présent	Absent
Camille MOUSSU Alexandre DESCAMPS Tom BENE François MERAT	Personne

Ordre du jour

- Informer tous les membre du groupe
- Redéfinir de nouveaux objectifs

Informar tous les membre du groupe

Après avoir rencontré M.Oster le 26/11, le groupe a du réfléchir a modifier la définition des objectifs pour proposer une application répondant à la consigne du projet.

Redéfinir de nouveaux objectifs

Tâches effectuées

1. Redéfinition de nos objectifs
 - Projet moins ambitieux
 - Focalisation sur la recherche par pertinence
2. Réduction des fonctionnalités prévues
 - Abandon de la fonctionnalité de commentaire sur les projets
 - Abandon des sondages

TODO-LIST

1. Parler a M.Festor ou M.Oster pour lui présenter les fonctionnalités prévues

A Faire avant le dernier TP de Web

7.1.4 5 décembre 2021

Présent	Absent
Camille MOUSSU Alexandre DESCAMPS Tom BENE François MERAT	Personne

Ordre du jour

1. Récapitulatif de l'avancement
2. Répartition des tâches

Récapitulatif de l'avancement

Après avoir rencontré M.Festor le 29/11, le groupe a vu ses objectifs validés et a pu commencer à travailler sur le site.

Tâches effectuées

1. Bilan des fonctionnalités
 - Publier des projets
 - Voter sur les projets
 - Tri des projets par
 - Vote
 - Tag
 - Pertinence
2. Répartition des tâches
3. Mise en place d'un Gantt

Répartition des tâches

Il s'agit de faire une répartition très générale :

Camille s'occupe du HTML et du CSS pour la structure de l'application, Tom de la gestion des cookies et des données des utilisateurs, Alexandre des critères pour le tri des données ainsi que certaine page HTML et François de la gestion de projet et du système de vote (like/dislike). De plus, le groupe doit mettre en place les tables de la base de données.

TODO-LIST

- Camille : structure HTML de base et CSS
- Tom : backend pour la gestion des cookies et des utilisateurs
- Alexandre : critères pour le tri des données et quelques pages HTML
- François : système de like et gestion de projet
- Un peu tout le monde : mise en place de tables pour la base de données

7.1.5 22 décembre 2021 11h

Présent	Absent
Camille MOUSSU Alexandre DESCAMPS Tom BENE François MERAT	Personne

Ordre du jour

1. Reprise après les partiels
2. Récapitulatif de l'avancement
3. Répartition des tâches

Reprise après les partiels

Contre-temps

Avec les partiels, l'équipe n'a pas pu avancer aussi efficacement que prévu sur les tâches affectées à la réunion précédente.

Récapitulatif de l'avancement

Tâches effectuées

- Première version du système de tag
- BD utilisable
- Système de like en cours (pas encore push sur le git)
- Publication de projet possible

Répartition des tâches

Tom se charge des cookies et d'aider Alexandre pour l'algorithme de pertinence. Il doit aussi corriger la fonction de *register*.

François doit faire la fonction de vote et débiter l'écriture du rapport avec Camille. Elle, doit améliorer le système de tag.

Alexandre doit faire la page de profil.

TODO-LIST

- Cookies (persistance serveur)
- Page de profil
- Système de like
- Algorithme de pertinence
- Écriture du rapport (CR et mise en page)
- Amélioration du système de tag
- Fonction *register* opérationnelle

A faire avant le 27/12

7.1.6 27 décembre 2021 17h

Présent	Absent
Camille MOUSSU Alexandre DESCAMPS Tom BENE François MERAT	Personne

Ordre du jour

1. Récapitulatif de l'avancement
2. Répartition des tâches

Récapitulatif de l'avancement

Contre-temps

Les fêtes de Noël ont ralenti le travail encore plus que prévu.

Ancienne TODO-LIST

- Cookies (persistance serveur) : en cours d'implémentation
- Page de profil : besoins de l'aide de Tom pour un aspect technique
- Système de like : Aucun avancement
- Algorithme de pertinence : Partie théorique ok on passe donc à l'implémentation
- Écriture du Rapport (CR et mise en page) : Début de mise en page et modèle de CR
- Amélioration du système de tag : OK
- Fonction register opérationnelle : Besoins de test

Répartition des tâches

Tom continue l'implémentation des cookies et se charge aussi de l'implémentation de l'algorithme de pertinence donnée par Alexandre.

François doit faire le système de like et avancé le compte-rendu avec Camille. Alexandre doit recevoir de l'aide de Tom pour la page de profil.

Camille propose de s'occuper de l'upload d'image ainsi que du bouton pour accéder à la page de profil. De même, elle s'occupe des message d'erreur pour le login.

TODO-LIST

- Implémentation des cookies
- Page de profil
- Système de like
- Implémentation de l'algorithme de pertinence
- Écriture du Rapport (Contenu des CR et +)
- Test de la fonction register
- Bouton profil & message d'erreur si login invalide
- Implémentation de l'upload d'image

7.2 Bibliothèques Python utilisées

- Flask
- SQLAlchemy
- Hashlib : sha256 (pour l'encodage du mot de passe)
- Datetime
- Pytest

Bibliographie

- [1] Agora. Agora. <https://www.app-agera.fr/>.
- [2] Kirsten Boehner, Carl DiSalvo. Data, design and civics : An exploratory study of civic tech. <https://dl.acm.org/doi/abs/10.1145/2858036.2858326>, 2016.
- [3] CitizenLab. CitizenLab. <https://www.citizenlab.co/fr>.
- [4] Decidim. Contact Decidim. <https://decidim.org/>.
- [5] Fluicity. Fluicity. <https://get.flui.city/>.
- [6] Intramuros. Contact IntraMuros. <https://appli-intramuros.fr/#CONTACT>.
- [7] Loïc Blondiaux. La démocratie participative : une réalité mouvante et un mouvement résistible. <https://www.vie-publique.fr/parole-dexpert/279196-la-democratie-participative-par-loic-blondiaux>, 2021.
- [8] MairesCitoyen. Fonctionnalité MairesCitoyen. <https://mairesetcitoyens.fr/fonctionnalites-application-maire>.
- [9] Wikipedia contributors. Civic technology — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Civic_technology&oldid=1060883494, 2021. [Online; accessed 5-January-2022].