



Projet PPII

Civic'TN

Ridel Julien
Dénès François
Trabelsi Youssef

Responsables : Festor Olivier et Oster Gérald



Table des matières

1	Introduction au sujet	3
1.1	Contexte du projet	3
2	Etat de l’art	4
2.1	Critères de comparaison des applications	4
2.2	Comparaison des offres du marché	4
3	Gestion de projet	5
3.1	Équipe de projet	5
3.2	Définition du projet	5
3.3	Analyse des risques : Matrice SWOT	6
3.4	Organisation du projet : WBS	7
3.5	Répartition des tâches : Matrice RACI	8
3.6	Diagramme de Gantt	8
3.7	Comptes rendus des réunions	9
4	Développement de l’application	18
4.1	Partie Web	18
4.2	Partie Base de données	25
4.3	Partie algorithmie : Analyse des commentaires	26
4.4	Complexité algorithmique	28
5	Bilan du projet	29
5.1	Bilan global du projet	29
5.2	Bilan du projet par membre	29
6	Annexes	33

1 Introduction au sujet

1.1 Contexte du projet

Ce projet a eu lieu dans le courant de la première année du cycle ingénieur de Télécom Nancy. Il fait partie du module PPII (Projet Pluridisciplinaire d'informatique intégrative) supervisé par M.Festor Olivier et M.Oster G  rald.

L'objectif est de concevoir une application qui rentre dans le cadre de la d  mocratie participative, qui semble se d  mocratiser aupr  s de nos   lus Fran  ais.

L'application    d  velopper devra contenir 3 grands axes, qui sont :

- Une partie Web. Cette partie sera d  velopp  e    l'aide des langages HTML, CSS et enfin la biblioth  que Flask du langage Python.
- Une partie base de donn  es. Cette partie a   t   premi  rement d  velopp  e en utilisant SQLite3 mais ce syst  me a par la suite   t   remplac   par le syst  me postgresql.
- Une partie Algorithmique. Cette partie sera d  velopp  e en utilisant le langage Python.

2 Etat de l'art

Afin d'obtenir une idée plus précise de ce qu'il était attendu d'une application de démocratie participative, nous avons fait une étude de ce qui était proposé sur le marché. Nous avons alors sélectionné plusieurs applications que nous avons comparé selon plusieurs critères définis ci-dessous.

2.1 Critères de comparaison des applications

- Impact géographique de l'application : Découvrir la diversité des zones géographiques qui pouvait être impactée par de telles applications.
- Ergonomie : L'ergonomie d'une application destinée à tous est primordiale pour nous. En effet, ce genre de projet doit pouvoir être utilisé par des personnes n'ayant pas une grande affinité avec tout ce qui touche à internet.
- Nombre de villes ayant adopté le service : Regarder le nombre de communes utilisant un service peut être un bon indicateur sur son efficacité auprès des citoyens.
- Services proposés : Ce critère nous permet de déterminer les points essentiels que doit avoir notre application et aussi, il peut nous donner certaines idées à apporter à notre conception.
- Payante ou non : Déterminer si les applications gratuites permettaient d'offrir les mêmes services que pour une application devant être achetée.

2.2 Comparaison des offres du marché

Un tableau est présent en annexe (Annexe 1) regroupant plusieurs applications (7) disponibles sur le marché afin d'avoir une vision globale et comparative de ce qui peut être fait dans le domaine de la démocratie participative. Ce tableau est seulement présent en annexe afin de permettre une lecture plus simple de celui à cause de son format.

Certaines applications étant payante, il a été difficile de pouvoir pleinement apercevoir l'application dans son ensemble, surtout pour la partie ergonomie. Afin de palier à cela, nous avons essayé de trouver des captures d'écrans de ces applications pour pouvoir évaluer son design. Le site du fabricant permettait aussi de pouvoir spéculer sur l'ergonomie de leur application de part les fonctionnalités présentes et la facilité de navigation.

3 Gestion de projet

3.1 Équipe de projet

L'équipe de projet est composée des trois étudiants suivants :

- Ridet Julien
- Trabelsi Youssef
- Dénès François

L'équipe de projet s'est réunie tout au long de la réalisation du projet à TELECOM Nancy puis sur l'application Discord lors des sessions de réunions lors des vacances. L'équipe est composée d'uniquement trois membres. En effet, un quatrième membre du groupe devait participer à ce projet, mais il n'a finalement pas été présent, ce qui a impacté le travail effectué par les membres restants du projet.

3.2 Définition du projet

Nous avons décidé de nous orienter vers une application permettant aux citoyens et à l'administration de soumettre des idées à un vote. Suivant le résultat, la Mairie pourra prendre des dispositions afin de mettre en place l'idée abordée. Sur chacune des propositions, un espace commentaire sera proposé afin de permettre aux utilisateurs de débattre sur un sujet et donner leurs avis. Leurs commentaires seront analysés à l'aide d'un algorithme afin d'avoir un aperçu de l'engouement autour d'une proposition. Pour se concentrer sur l'essentiel et ne pas perdre de temps, nous avons décidé de ne pas ajouter de système de gestion de comptes.

Certaines fonctionnalités ont été imaginées mais ne seront sûrement pas présentes lors de la présentation au jury, comme par exemple un système de modération des posts et des commentaires, un système permettant de consulter les profils des autres utilisateurs afin de pouvoir consulter leurs avis et leurs posts.

3.3 Analyse des risques : Matrice SWOT

Voici ci-dessous notre matrice SWOT. Nous avons isolé comme force les connaissances acquises par François au cours de sa formation en IUT, ainsi que la créativité générale du groupe. Nous avons décerné des faiblesses dans notre projet. Tout d'abord une connaissance faible de certains langages, impliquant un temps de formation à considérer qui ralentissant notre progression, ainsi que la perte d'un membre de notre groupe, faisant de notre groupe un groupe constitué d'uniquement trois personnes. Nous n'avons pas soulevé d'opportunités, en revanche, la période d'examens et de fêtes réduisent notre productivité sur une période. La présence de problèmes techniques est également un point à envisager.



Figure 1: Matrice SWOT

3.4 Organisation du projet : WBS

Afin de pouvoir mieux cerner tout ce qui doit être fait pour finir notre application, nous avons décidé de découper le projet en lots. Ces lots ont été répertoriés dans le schéma WBS ci-dessous. Une partie développée de ce WBS se trouve en annexe (Annexe 2).

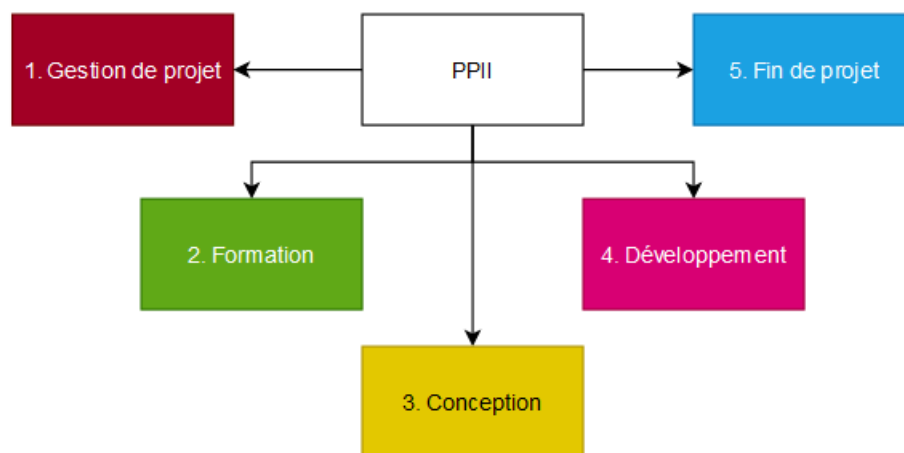


Figure 2: WBS

- Gestion de projet : Tâches qui concernent la réalisation des éléments en rapport avec la gestion de projet, comme les différents diagramme ou les comptes rendus.
- Formation : Tâches qui concernent la formation sur les différents langages de programmation.
- Conception : Tâches qui concernent le préambule du projet. Que ce soit la réalisation des maquettes ou la conception de la base de données nécessaires.
- Développement : Tâches qui concernent le développement de notre application
- Fin de projet : Tâches qui concernent la phase de rédaction du rapport ainsi que la soutenance du projet.

3.5 Répartition des tâches : Matrice RACI

La matrice RACI de la partie développement de notre WBS peut être retrouvée en annexe (Annexe 3). En effet, les phases de définition du projet et de clôture du projet sont réalisées en groupe, et ne font donc pas l'objet d'une répartition. On peut tout de même noter que Julien Ridet a été chargé de la préparation et de la rédaction des comptes-rendus des réunions.

Pour procéder à la répartition des tâches, une réunion a été faite par les membres du groupe afin de les répartir selon les capacités de chacun et leurs envies. François Dénès a, de par son expérience acquise lors de sa formation en IUT, la responsabilité de vérifier le code produit par chacun et d'apporter son aide aux autres membres du groupe pour résoudre les problèmes liés à l'aspect programmation.

3.6 Diagramme de Gantt

Afin de pouvoir consulter facilement le diagramme de Gantt relatif à ce projet, une section en annexe (Annexe 4) est dédiée afin de rendre sa lecture plus facile. Il est important de repérer quelques points sur ce diagramme :

- Lors de la phase de gestion de projet, il y a un faible nombre de tâches en parallèle afin de spécifier le fait que tout le groupe travaille ensemble et en même temps sur les mêmes tâches.
- La période des examens, à elle aussi était prévue, ce qui est représenté par une réduction des tâches attribuées lors de cette période.
- Dernier point, il est important de spécifier que cette répartition des tâches est prévisionnelle. Celle-ci peut être amenée à varier pendant le développement de l'application en fonction de l'avancement de chacun.

3.7 Comptes rendus des réunions

3.7.1 16 Novembre 2021

Comptes rendus réunions n°1

Participants : Ridet Julien - Youssef Trabelsi

Durée : 2h30

Objectifs de la séance :

Brainstorming autour du thème et recherche d'applications déjà existantes :

- Brainstorming autour du thème
 - L'application devra permettre aux citoyens de proposer des idées qui seront soumises à une évaluation de la part de leurs concitoyens.
 - * Chaque propositions sera validées ou non. Si elle est validée, cette idée est soumise au conseil municipal qui pourra décider soit d'agir directement, soit de consulter ses citoyens, soit de ne pas étudier le projet, je jugeant non pertinent.
 - L'application devra permettre au conseil municipal de consulter les citoyens sur divers thèmes, par le biais d'une question, sur le même principe du référendum.
 - L'application devra permettre aux utilisateurs de réaliser des sondages et des pétitions.
 - Une fonctionnalité "bonus" à été aussi abordée. Les citoyens pourraient signaler des problèmes dans leur ville comme par exemple, signaler un dépôt sauvage d'ordures.
 - Une partie de l'application serait présente pour afficher les actualités de la ville ainsi que les propositions ayant affecté la vie de la ville.
 - Certaines idées de designs ont été apportés mais rien de concluant pour l'instant.
 - Chaque propositions/idées etc. . . aura un système de like/dislike comme les réseaux sociaux ainsi qu'une partie commentaire afin d'ouvrir des débats entre citoyens (bien sûr, un système de signalement devra sûrement être mis en place afin d'éviter les débordements et les conflits, ce qui serait contre productif).

- Recherche approfondie sur le Web d'applications déjà existante :
 - Une recherche approfondie sur le Web à permis de mettre en évidence différentes applications de CivicTech sur le marché.
 - Début de la conception de l'État de l'Art en comparant toutes les applications retenues avec des critères qu'il nous reste encore à pleinement identifier.

3.7.2 17 Novembre 2021

Comptes rendus réunions n°2

Participants : Ridel Julien - Youssef Trabelsi - Dénès François

Durée : 2h

Objectifs de la séance :

Définir clairement nos envies pour l'application souhaitée, mettre en place un début de base de donnée et créer un style graphique pour notre application.

- Définition de notre application:
Plusieurs idées ont été abordées. On à terminé par se mettre d'accord sur une idée globale de l'application :
 - L'application devra permettre aux citoyens de proposer des idées qui seront soumises à une évaluation de la part de leurs concitoyens.
 - Plusieurs idées ont été échangées pour au final converger vers une idée commune.
 - Nous avons donc décidé de nous orienter vers une application permettant aux citoyens et à l'administration de soumettre des idées à un vote. Suivant le résultat, la Mairie pourra prendre des dispositions afin de mettre en place l'idée abordée.
 - Nous avons aussi déterminé un ensemble d'idées avec un ordre de priorité afin de pouvoir rendre en priorité les éléments principaux de l'application tout en nous laissant des possibilités d'ajout "bonus" comme par exemple, un système de commentaires sous chaque proposition afin de donner la possibilité de débattre.

- Début de base de données :
 - Après avoir clairement explicité nos envies sur l'application, nous avons commencé à proposer une première ébauche de base de données.
 - Parler de base de données, à ce stade est un peu trop avancé. Nous avons surtout explicité les parties importantes qu'il faudra implémenter dans une base de donnée afin d'assurer le bon fonctionnement de notre application.
- Création du style graphique :
 - Après un long moment à expliciter un schéma bref sur une probable base de données, nous avons décidé de commencer une recherche de style graphique pour conclure notre réunion.
 - Quelques schémas des principales pages ont été dessiné ce qui nous a donné un avis sur la complexité d'écriture des programmes que nous devons faire afin de suivre le mieux possible nos schémas.

3.7.3 23 Novembre 2021

Comptes rendus réunions n°3

Participants : Ridel Julien - Youssef Trabelsi - Dénès François

Durée : 15 min

Objectifs de la séance :

Procéder à la présentation de notre idée ainsi qu'à sa validation auprès de M. Olivier Festor.

- Contenu de la réunion :

Notre application à été présenté avec l'aide de nos différentes maquettes. Et certains points essentiels ont été abordés à propos de notre future application.

 - La partie Login (mot de passe et compte) doit être mise de côté afin de se concentrer sur le développement de l'application et surtout sur la partie gestion de projet.

- L'idée d'un espace commentaire sur chaque post afin de lancer des débats à été appréciée. M. Festor a ainsi proposé l'idée de réaliser des algorithmes d'analyse de commentaires afin d'obtenir des informations sur les différentes réactions à propos d'un post. Ces algorithmes permettront de trier les posts afin de les classer par popularité.
- Conclusion de la réunion :
 - A la fin de cette réunion, notre idée a été validée et nous sommes sortis satisfait de cette entrevue car elle nous a permis d'obtenir des idées supplémentaires afin de réaliser l'application souhaitée et aussi, cette réunion nous a permis de savoir exactement sur quel point se concentrer. Nous tenons à remercier M. Festor pour le temps qu'il nous a accordé et sur les idées qu'il nous a apporté.

3.7.4 25 Novembre 2021

Comptes rendus réunions n°4

Participants : Ridel Julien - Youssef Trabelsi - Dénès François

Durée : 15 min

Objectifs de la séance :

Procéder au découpage de notre projet en lots selon le système WBS et début de mise en place d'une matrice RACI.

Contenu de la réunion :

Nous nous sommes réunis afin de découper notre projet en lots. Avec l'aide du site internet draw.io, nous avons pu créer un arbre représentant l'ensemble des tâches à réaliser lors de notre projet. Puis, à l'aide de cet arbre, nous avons commencé à réaliser une matrice RACI afin de répartir les tâches entre les personnes du groupe. Cette arbre est présent en Annexe (Annexe 2 WBS)

3.7.5 1 Décembre 2021

Comptes rendus réunions n°5

Participants : Ridel Julien - Youssef Trabelsi - Dénès François

Durée : 2h

Objectifs de la séance :

Résumer les tâches déjà effectuées et mettre en place une matrice RACI.

- Résumé des tâches déjà effectuées :
 - Tâches déjà réalisées :
 - * Mise en accord des idées et décision de la marche à suivre (Faite par le groupe)
 - * Prévisions de ce qui peut être fait dans le temps imparti en prenant en compte les capacités du groupe (Faite par François Dénès)
 - * Les premières maquettes visuelles de l'application (Faite par Julien Ridel)
 - * Mise en place d'un début de schéma relationnel sur papier qu'il faudrait améliorer et mettre au format draw.io (Faite par Youssef Trabelsi)
 - * Validation de l'idée auprès de M. Festor
- Mise en place de la matrice RACI :
 - Nous avons réalisé une matrice RACI en essayant de prendre en compte les capacités propres de toutes les personnes du groupe ainsi que leurs préférences sur les différentes tâches à réaliser. Cette matrice a été réalisée en gardant à l'esprit que notre groupe serait flexible sur l'assignation des tâches. Ce qui veut dire que si une personne a une idée pour réaliser une tâche d'un autre, il peut la faire mais doit obtenir l'accord de la personne en charge de cette tâche. Cette matrice RACI se trouve en Annexe (Annexe 3).

3.7.6 6 décembre 2021

Comptes rendus réunions n°6

Participants : Ridel Julien - Youssef Trabelsi - Dénès François

Durée : 4h

Objectifs de la séance :

Utiliser la séance de TP du lundi 06/12/2021 pour mettre en place les bases de notre application afin de commencer le développement.

- Réalisations de Julien Ridel :
 - Création d'une page très basique en HTML avec la mise en place d'éléments de style. La partie Flask n'a pas été abordée lors de cette séance. Le style de la page a été défini afin de donner une charte graphique à suivre pour tout le groupe. Un début de mise en place d'un futur bouton pour ajouter des posts sur notre application a été lui aussi ajouté.
- Réalisations de Youssef Trabelsi :
 - Création d'un premier schéma relationnel afin de prévoir la future base de données qu'il faudra implémenter à notre application.
- Réalisations de François Dénès :
 - Aide pour certains aspects de style de la page HTML.
 - Aide pour la réalisation du schéma relationnel puis vérification.
 - Recherche pour la mise en place d'un système permettant la création de fenêtre pop up sur notre page principale afin d'éviter une redirection sur une nouvelle page lors de la consultation d'un post.
- Conclusion de fin de séance :
 - A la fin de cette séance, nous nous sommes mis d'accord sur le fait que lors des jours suivant, le développement allait être léger. En effet, nous avons voulu prendre du temps pour les partiels, quitte à devoir redoubler d'efforts par la suite. Mais ceci n'était pas obligatoire, il est possible de commencer les recherches pour le développement de notre application.

3.7.7 Réunions d'avancements

Comptes rendus réunions n°7

Participants : Ridel Julien - Youssef Trabelsi - Dénès François

- Objectifs de ce compte rendu : Ce document a pour but de répertorier les différentes discussions opérées par les membres du groupe afin de témoigner de l'avancement du projet et des différentes difficultés rencontrées tout au long du développement de l'application.
- Point du 21 décembre 2021 par Ridel Julien :

Lors de cette journée, la mise en place des éléments primordiaux de Flask a été effectuée (Templates, static, App.py). Du temps a été pris pour faire le lien entre HTML et CSS. Après cela, un début de page pour permettre la création des posts a été codée sans certains éléments, comme la possibilité de télécharger des images sur l'application et le style CSS de l'ensemble. Tout semblait fonctionner, il était possible de voir les posts créés directement sur la page principale de notre site. A ce moment là, il était impossible de consulter le contenu des posts. Pour faire fonctionner le tout, une première base de données a été mise en place en utilisant SQLite3.
- Point du 22 décembre 2021 par Ridel Julien :

Les éléments codés la veille ont été repris afin de rajouter des options et tout a été push sur le répertoire git.
- Point du 23 décembre 2021 par Ridel Julien :

Lors de cette journée, tout ce qui a été fait la veille a été repris. Le style des différentes pages a été mis en place. Un éditeur de texte "complet" ainsi que la fonctionnalité d'ajout d'image a été ajoutés à l'éditeur de posts. Les images à ce moment là, doivent être ajoutées via un lien Web. Enfin, un début de page pour consulter le contenu des posts a été ajoutée. Le style a pris beaucoup de temps afin de pouvoir agencer tous les éléments du post, mais à ce moment là, le contenu du post n'était pas pleinement consultable.

- Point du 24 décembre 2021 par Dénès François :

Les grands axes du contrôleur ont été mis en place en utilisant Flask. Les routes vers les pages commencées et prévues ont été mises en place, ainsi qu'un certain nombre de fonctions utilisataires, permettant par exemple de trier et filtrer les posts selon les critères définis. L'implémentation du schéma relationnel en SQL au travers de la création de tables a également été commencé au cours des semaines et jours précédents.

- Point du 27 décembre 2021 par Dénès François :

L'affichage des informations des posts au sein d'une page dédiée a été implémenté. Les informations propres au post telles que son titre, son contenu ou encore son auteur ont été ajoutés dans un premier temps. Les informations plus complexes telles que les avis et les commentaires seront ajoutés par la suite car plus compliquées à mettre en oeuvre et nécessitant encore une réflexion quant à la façon d'implémenter ces fonctionnalités. La première version de la base de donnée a été complétée, avec l'ajout de vues simplifiant l'accès et la modification des données depuis la partie Web. La base de donnée pourra encore éventuellement être légèrement modifiée si le besoin s'en fait sentir pour faciliter l'intégration d'une fonctionnalité.

- Point du 28 décembre 2021 par Ridel Julien, Dénès François et réunion entre Dénès François et Trabelsi Youssef :

- Ridel Julien a commencé l'ajout des différentes réunions effectuées. Les notes prises lors de ces réunions ont dû être organisées ainsi que codée en LaTeX. Seulement 2 ont été ajoutées, les autres suivront.
- Dénès François a continué l'affichage des posts, avec l'implémentation de l'affichage des avis donnés sur le commentaire, et l'ajout ou la modification de ceux-ci par l'utilisateur.
- Lors de la réunion entre Dénès François et Trabelsi Youssef (Ridel Julien ne pouvait être présent à cause d'un impératif), une présentation de l'avancement du projet a été effectuée. Les membres du groupe ont par la suite exprimé les tâches sur lesquelles ils étaient en train de travailler, et les tâches les plus importantes à avancer, afin d'avoir une vision globale sur l'avancement du projet et sur ce qui restait à faire.

- Point du 30 décembre 2021 par Dénès François et Trabelsi Youssef :
 - Youssef Trabelsi a commencé à travailler sur la fonction d'analyse des commentaires. Tout d'abord, il a implémenté une fonction qui permet la correction (grammaticale) automatique des commentaires afin de les analyser. Mais il s'est avéré que cette tâche n'était pas si simple à exécuter à cause de la possibilité d'avoir un très grand nombre de fautes. Il a donc supposé que les commentaires soient grammaticalement corrects. Également, il n'a pas trouvé ce qu'il voulait sur internet pour avoir une liste connue des mots positifs ou négatifs. D'après lui, la recherche d'un dictionnaire de ce type était difficile car c'est un produit qui doit être acheté.
 - Il a alors pensé à regrouper tous les adjectifs, noms, verbes positifs et négatifs les plus utilisés et utiliser les règles de la langue française. (Par exemple le suffixe -ard permet souvent de construire des termes péjoratifs comme chauffard)
 - Il a aussi pensé aux négations dans sa fonction pour éviter la fausse interprétation du commentaire.
 - Pour cela il a tout d'abord créé une fonction prenant tous les mots du commentaire et les analyses individuellement, puis une fonction analysant le commentaire dans son ensemble.
 - Dénès François a ajouté les fonctions de connexion et déconnexion. Les filtres et tris développés précédemment ont également été ajoutés à la page d'accueil affichant la liste des posts. L'ajout des commentaires et de leurs réponses vont être ajoutés au cours des jours suivant.
- Point du 31 décembre 2021 par Ridet Julien : Ajout du document contenant tous les différents points d'avancements du développement ainsi qu'un début de document pour le rapport de projet.
- Point du 1 décembre 2021 par toute l'équipe : L'équipe de projet s'est réunie pour faire une réunion afin de faire le point sur ce qui a été fait et sur ce qu'il restait à faire. A la fin de cette réunion, l'équipe a pu constater que l'application était majoritairement opérationnelle. Les tâches restante à faire vont sûrement être implémentées dans les jours qui suivent.
- Point du 3 janvier 2022 par Trabelsi Youssef :

Trabelsi Youssef a ajouté un début de page permettant de visionner le profil de l'utilisateur connecté.

4 Développement de l'application

Dans cette partie, l'ensemble des moyens et procédés utilisés lors du développement de notre application vont être présentés.

4.1 Partie Web

Dans cette partie, nous allons aborder toute la partie WEB de notre application. L'ensemble de l'application a été élaborée en utilisant les langages html et css ainsi que la bibliothèque Flask de Python.

Nous allons donc présenter la différence entre nos maquettes de début de recherche d'idées et ce qui est présent à la fin du développement.

Page d'accueil de notre application :

Maquette :

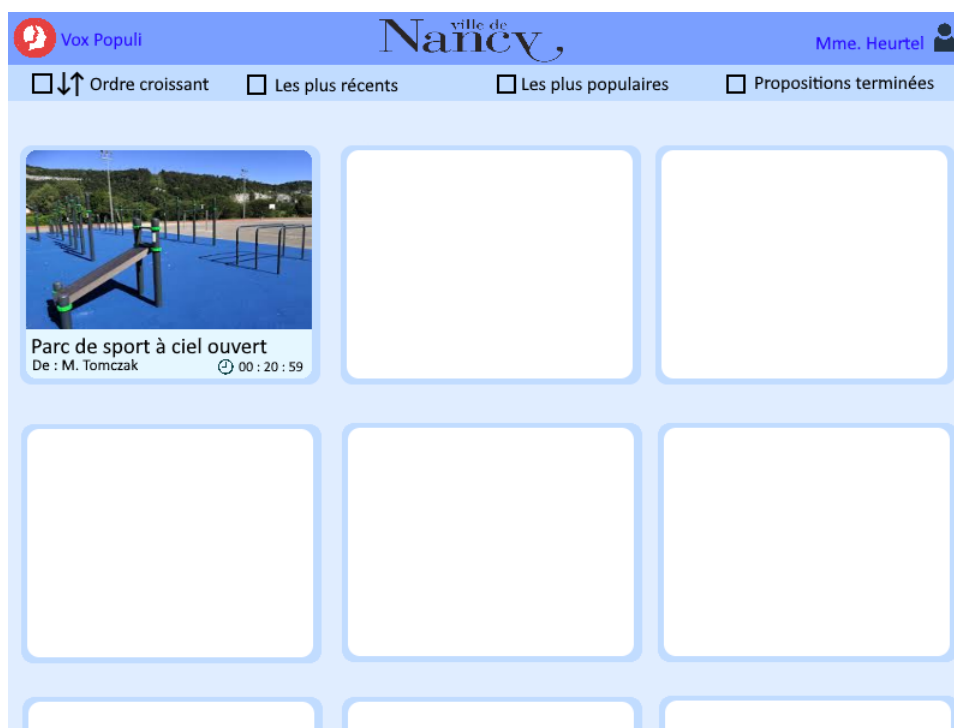


Figure 3: Maquette page principale

Résultat final :

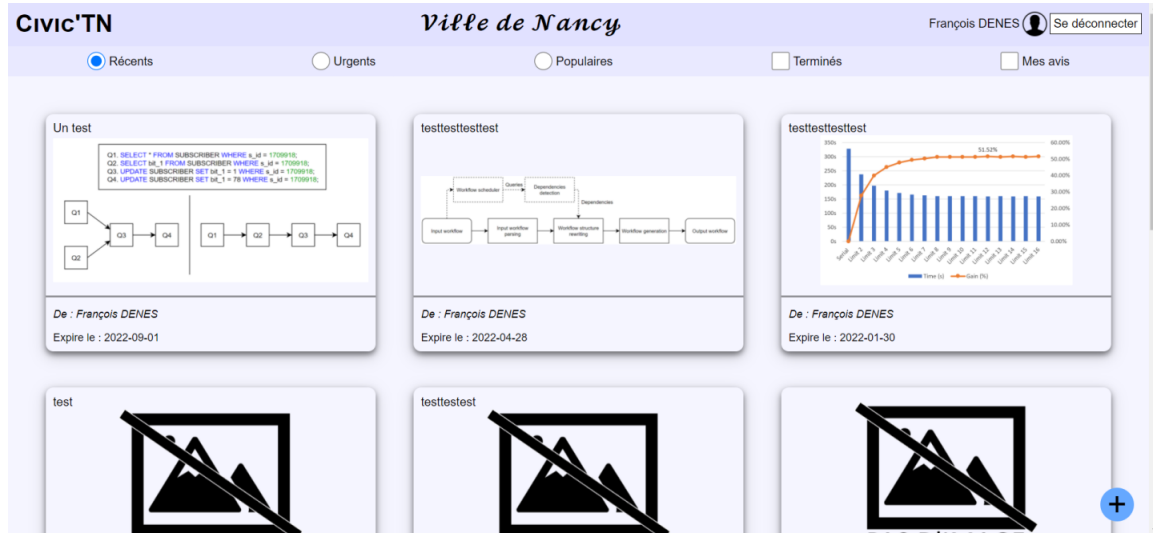


Figure 4: Résultat page principale

Tout d'abord, il est bon de remarquer que le résultat du design final est assez proche de la maquette.

Cette page a plusieurs fonctionnalités :

- Premièrement, cette page est l'accueil de notre site. C'est d'ici que l'utilisateur aura accès à toutes les fonctionnalités de l'application.
- Pour l'affichage des posts, une association entre requêtes SQL et un système de grille a été effectuée afin de pouvoir consulter tous les posts existants sur le site.
- Afin d'offrir une fonctionnalité supplémentaire aux utilisateurs, un système de filtre a été mis en place afin de faciliter la recherche de certains types de posts, comme par exemple, les plus récents, les plus appréciés, ...
- Cette page permet aussi de consulter son profil. Pour éviter de perdre du temps sur le système de connexion, nous avons décidé de créer seulement deux comptes tests. Le premier est un compte d'administrateur et le second un compte d'utilisateur.
- Enfin, par le biais d'un bouton fixe situé en bas à droite de l'écran, l'utilisateur peut accéder à une page permettant de créer son propre post et de le publier sur la plateforme.

Page de consultation des posts :

Maquette :



Figure 5: Maquette post

Résultat final :

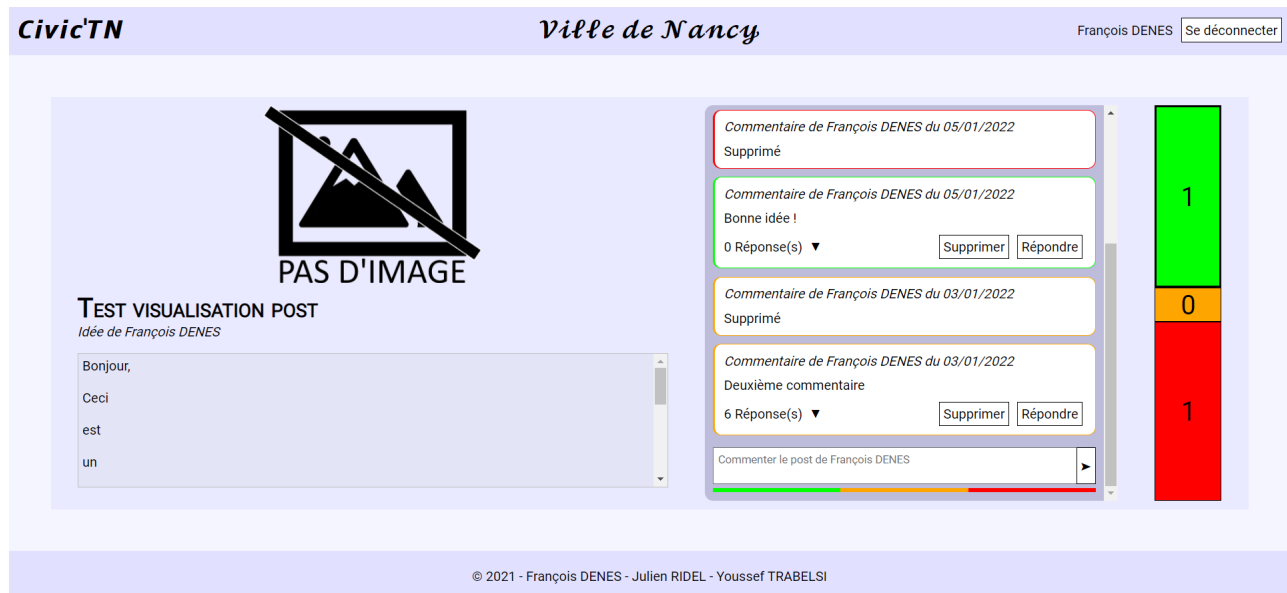


Figure 6: Résultat post

L'idée de départ pour visualiser les posts était de faire un système de pop-up permettant de rester sur la page d'accueil principale. Malheureusement, face à la difficulté pour implémenter ce système, l'idée a été abandonnée et a été remplacée par une page annexe qui s'ouvre une fois avoir cliqué sur un post. Sur cette page, on peut trouver :

- Sur la partie gauche, on retrouve toutes les informations relatives au post : Son image, son titre, son auteur ainsi que son contenu. Ces informations sont obtenues à l'aide d'une requête SQL qui prend exactement les bonnes informations.
- Sur la partie droite, on retrouve la partie commentaires et débats ainsi qu'une représentation des avis laissés par les utilisateurs. Encore une fois, ces informations sont récupérés par le biais de requêtes SQL. Celles-ci sont transmises à un algorithme d'analyse qui est développé par la suite afin d'ajouter un code couleur à chaque commentaires pour spécifier son caractère (Positif/Neutre/Négatif).

On peut donc remarquer une certaine différence entre la maquette et le résultat final, ce qui peut être une amélioration comme avec le code couleur qui a été ajouté au cours du développement.

Page de profil :

Pour cette partie, aucune maquette n'a été réalisée. En effet, cette fonctionnalité n'était pas une priorité lors de la conception de notre idée. Cette fonctionnalité a été ajoutée lors de la conception afin de permettre d'introduire quelques fonctionnalités. Par exemple, sans être connecté, l'utilisateur n'a pas accès à la page pour créer un post. D'autre part, le profil administratif a droit à certaines fonctionnalités permettant de gérer l'application, comme par exemple la suppression des posts et pourrait avoir la possibilité de supprimer certains commentaires afin de mettre en place un système de modération. Actuellement, ces pages ne sont pas réellement complètes. En effet, elles ont été ajoutées de façon sommaire afin d'avoir une vision de tout ce qui pourrait être présent dans le futur sur notre application.

Voici le résultat final des différentes pages de profil :

Profil utilisateur :



Figure 7: Page de profil utilisateur

Profil administratif :



Figure 8: Page de profil de l'administration

L'idéal, ce serait que dans le futur l'utilisateur puisse modifier certaines informations sur son profil afin d'ajouter un système de personnalisation.

Page de création de post :

Pour ce dernier point de la partie Web de notre application, aucune maquette n'a été réalisée au préalable car le design de cette page était encore à déterminer en fonction des autres fonctionnalités. Voici le résultat final :

The screenshot shows a web interface for creating a proposal. At the top, there is a header bar with 'CivicTN' on the left, 'Ville de Nancy' in the center, and 'François DENES' with a 'Se déconnecter' button on the right. The main heading is 'Création de votre proposition'. Below this, there are several input fields: 'Titre' with a placeholder 'Titre de votre proposition', 'Auteur' with the value 'François DENES', 'Image' with a 'Choisir un fichier' button and the text 'Aucun fichier choisi', and 'Date de fin' with a date picker showing 'jj/mm/aaaa'. A large text area for 'Description de votre proposition' is on the right. At the bottom of the form is a 'Soumettre' button. The footer contains the copyright notice '© 2021 - François DENES - Julien RIDEL - Youssef TRABELSI'.

Figure 9: Création de post

Afin de récupérer les différentes informations que l'utilisateur veut apporter à son post, un système de formulaire a été utilisé. Sur cette page, l'auteur peut renseigner le titre, le descriptif, une image ainsi que la période pendant laquelle sa proposition est soumise aux votes. Toutes ces informations étant ajoutées à la base de données du site afin d'actualiser la page d'accueil au moment de la validation de la création du post.

4.2 Partie Base de données

Cette partie présentera la base de donnée utilisée dans notre application, depuis sa conception jusqu'à son implémentation.

4.2.1 Schéma relationnel

Le schéma relationnel, qui figure en Annexe 5, présente les entités de notre base de donnée ainsi que leurs relation.

L'élément central est l'entité Utilisateur, représentant l'ensemble des utilisateurs de l'application. Cet entité possède deux attributs, un identifiant, utilisateur_id, et une adresse, communs aux deux entités dérivées de celle-ci, Particulier et Administration. La première, Particulier est composée de son identifiant, particulier_id, d'un nom et d'un prénom comme attributs, tandis que la seconde, Administration, possède également un identifiant, administration_id, une dénomination, un numéro de téléphone, un email et un site internet.

Le deuxième élément de grande importance est l'entité Post, qui représente les posts partagés par les utilisateurs. Un Post ne peut être posté que par un seul utilisateur, mais un utilisateur a la liberté de poster autant qu'il le souhaite. Un Post est notamment composé d'un titre, et d'un texte, ainsi que d'une date de début et de fin. Les utilisateurs peuvent laisser des avis sur les posts qu'ils souhaitent.

Les utilisateurs peuvent commenter les posts publiés, ainsi que répondre à des commentaires déjà postés. On retrouve donc une entité Commentaire, comportant le texte du commentaire et la date à laquelle il a été posté. Les entités Premier et Réponse héritent de Commentaire. La première entité désigne les commentaires effectués directement sur un post, tandis que la deuxième désigne les commentaires effectués en réponse à un commentaire. En effet, nous avons choisi de laisser aux utilisateurs la possibilité d'échanger en réponse à un commentaire, mais pas de répondre à une réponse, afin de limiter l'imbrication des réponses et la perte de lisibilité. A noter qu'un utilisateur est libre de poster plusieurs commentaires sur un même post, ainsi que de répondre plusieurs fois à un même commentaire.

Le détail des attributs présents dans les entités du schéma relationnel peuvent retrouvés dans le dictionnaire de données présent en annexe 6.

4.2.2 Implémentation

Cette partie s'attachera à détailler l'implémentation de la base de donnée dans le cadre de notre projet.

La base de données a été implémentée en PostgreSQL. Nous avons fait ce choix pour deux raisons. Tout d'abord, comparé à SQLite, utilisé dans le cadre des travaux pratiques de web, PostgreSQL a l'avantage de fournir d'avantages de fonctionnalités, qui nous ont été utiles lors du développement de notre application. Ensuite, PostgreSQL a été déjà utilisé au cours des travaux pratiques de Base de données, et nous avons donc déjà une expérience sur ce DBMS.

L'implémentation des tables de la base de donnée a suivi le schéma relationnel, ce qui a donné un total de huit tables, la relation "Donne avis" s'étant traduite par l'ajout d'une table supplémentaire ayant pour clé primaire deux clés étrangères, l'identifiant d'un utilisateur et l'identifiant d'un post. Une colonne supplémentaire permet de conserver l'avis donné par l'utilisateur.

La principale différence a été l'ajout de vues, pour chacune des tables. Ce choix a été fait afin de simplifier les requêtes effectuées sur la base de donnée. Ainsi la vue "Particuliers" permet d'accéder facilement aux informations de la table "Particulier", mais également aux informations complémentaires contenues dans la table "Utilisateur" dont elle hérite. De la même manière, l'accès aux informations des administrations, des premiers commentaires et de leurs réponses s'en trouvent simplifié. L'utilisation de vue permet aussi l'intégration d'attribut calculés, tels que le nombre d'avis sur un post, ou encore déterminer si le post a dépassé sa date de fin. Enfin les vues permettent de grandement simplifier la gestion des ajouts de nouvelles données dans les tables, en faisant une seule insertion sur la table, qui s'occupe de rediriger vers les différentes tables concernées les informations.

4.3 Partie algorithmie : Analyse des commentaires

Dans cette partie, nous allons aborder l'algorithme qui permet d'analyser l'ensemble des commentaires d'un post afin d'en déduire si celui-ci provoque un engouement auprès des utilisateurs de la plateforme.

Cette algorithme utilise plusieurs fonctions pour permettre l'analyse des commentaires. Nous allons donc les détailler ci-dessous, ainsi que tous les éléments permettant le bon fonctionnement de cet algorithme.

Il est important de préciser que toutes les fonctions suivantes sont de complexité linéaire.

- Fonction 1 : Find-words :

Cette fonction prend un commentaire en paramètre et permet de séparer tous les mots le composant afin de les incorporer dans une liste. Tous ces mots étant mis dans un même format, ici en minuscule. Cette fonction permet donc de garder seulement les mots contenus dans le commentaire en retirant la ponctuation.

Prenons un exemple, utilisons le commentaire suivant :

”Cette phrase, est un test pour la fonction.”

La fonction Find-words renverra alors :

[’cette’, ’phrase’, ’est’, ’un’, ’test’, ’pour’, ’la’, ’fonction’]

Ce découpage sera utile pour la suite de notre algorithme afin de faciliter la détection des mots positifs ou négatifs.

- Fonction 2 : Analyse :

Cette fonction prend un mot en argument et détermine si celui-ci est à connotation positive, négative ou neutre. Cette fonction possède aussi comme autres arguments une liste des mots positifs et une des mots négatifs négatifs. Ces deux derniers arguments ont été ajoutés pour faciliter l’amélioration continue des dictionnaires.

Cette fonction permet aussi de faire des tests dans le cas ou le mot à analyser n’est ni dans la liste négative ni dans la liste positive.

Ces tests exploitent la structure des mots dans la langue française en regardant les suffixes et les préfixes qui donnent généralement le sens du mot. La fonction renvoie 1 (positif), 0 (neutre) ou -1 (négatif) en fonction du mot analysé.

Par exemple, si on utilise cette fonction sur les mots ”insuffisant” et ”chauf-fards”, la fonction retournera la valeur -1.

Les dictionnaires utilisés lors du développement de l’application sont présents en Annexe (Annexe 5).

- **Fonction 3 : Avis :**

Afin de pouvoir pleinement procéder à l'analyse des commentaires, cette dernière fonction va utiliser les deux premières. Cette fonction prend un commentaire et renvoie s'il est positif, négatif ou neutre. Il prend de même les deux dictionnaires en argument pour les mêmes raisons qu'évoquées précédemment. La différence entre la fonction analyse et avis est que la fonction Analyse est conçue pour un seul mot et la fonction Avis pour un commentaire.

Tout d'abord, celle-ci découpe le commentaire puis effectue une analyse de chaque mots et prend aussi en compte la tournure négative de la phrase.

Dans un second temps, les valeurs attribuées à chacun des mots sont additionnées afin de savoir si le commentaire analysé est plutôt positif, neutre ou négatif.

Exemple : le retour de la fonction avis pour le commentaire: "je crois que cette solution n'est pas efficace car on ne peut pas prévoir les retards" est -3.

Cette valeur est ensuite envoyée à notre application, ce qui permet d'exprimer l'engouement d'un post.

4.4 Complexité algorithmique

Après avoir réalisé l'ensemble de l'application, un test de complexité des fonctions s'est imposé. Il s'est avéré que l'ensemble des fonctions permettant le bon fonctionnement de l'application sont de complexité linéaire. Il est aussi important de garder à l'esprit que leur performances dépendent du code mais également des requêtes SQL qui y sont exécutées. Les résultats obtenus des tests de performance des fonctions d'analyse de commentaires de l'application sont en annexe 7, avec pour abscisse le nombre de mots et pour ordonnée le temps en secondes.

5 Bilan du projet

5.1 Bilan global du projet

Au lancement de notre projet, nous étions conscient que plusieurs fonctionnalités n'allaient pas être présente lors de la livraison de l'application. Nous avons été heureux de voir que la plupart des fonctionnalités voulues et importantes à nos yeux sont présentes dans le prototype final.

Le temps a peut-être joué contre nous vers la fin de la période de développement mais nous avons fait l'essentiel pour que notre idée soit menée à bien.

Ajouts envisagés si l'application venait à être développée :

- Ajout de tags sur les posts afin d'ajouter plus de filtres de recherche.
- Ajout d'un système de notifications et de recommandations afin d'orienter les posts qui sont visible en haut de liste suivant l'utilisateur connecté.
- Ajout de plusieurs statuts sur les posts les plus intéressants afin d'avoir un aperçu de l'avancement de sa réalisation par l'équipe administrative. Exemple : " En projet " et " Réalisé "
- Ajout d'un système de modération complet
- Ajout de la personnalisation des profils.

5.2 Bilan du projet par membre

5.2.1 Ridet Julien

- Points positifs :
 - Un travail de groupe est toujours une bonne chose pour en apprendre sur soi.
 - Découverte du déroulement d'un projet de ce type.
 - Meilleure compréhension de certaines notions découverte lors du MOOC.

- Difficultés rencontrées :
 - Difficultés sur certaines parties liant Flask et base de données.
 - Problèmes de communications entre les membres du groupe.
 - Les révisions des partiels ont réduit le nombre de réunions et le temps de développement.
- Expériences personnelle :
 - Amélioration des capacités à coder en Flask, LaTeX, html et css.
 - Adaptation envers les personnes du groupe améliorée afin de procéder au bon déroulement du projet.
- Axes d'amélioration :
 - L'organisation personnelle et avec les autres membres du groupe.
 - Amélioration dans la clarté des productions (le code notamment).
 - Améliorer sa communication lors de problèmes au sein du groupe.
 - Améliorer la gestion du temps.
 - Prendre de l'avance sur l'avancement du projet.

5.2.2 Dénès François

- Points positifs :
 - Un résultat obtenu fidèle aux objectifs fixés en début de projet.
 - Une bonne expérience de travail avec les membres de ce groupe.
- Difficultés rencontrées :
 - Garder un espace de travail propre et adapté au travail de groupe, nécessitant la reformulation et l'amélioration du code produit par les autres membres moins expérimentés dans le domaine.
 - Un large temps accordé à la définition du projet et la préparation des examens, suivis d'une période fêtes a rendu le rythme à maintenir lors des autres phases un peu trop soutenu par moments.

- Expériences personnelle :
 - Découverte de la librairie python Flask pour réaliser des applications Web.
 - Expérience supplémentaire acquise en gestion de projet.
- Axes d'amélioration :
 - Augmenter la fréquence des commentaires dans le code dès le commencement du développement afin de faciliter la lecture par les autres membres du groupe.

5.2.3 Trabelsi Youssef

- Points positifs :
 - Découverte du processus de création d'application.
 - Application des connaissances en gestion de projet acquises lors du MOOC de gestion de projet.
 - Motivation pour réaliser d'autres projets scolaires ou professionnels.
- Difficultés rencontrées :
 - Difficultés à appréhender le langage html.
 - Difficultés pour réaliser le début de l'algorithme permettant l'analyse des commentaires.
- Expériences personnelle :
 - Amélioration en communication en langue Française.
 - Progression dans la compréhension de certains langages comme html ou Flask.
 - Réalisation de tests unitaires en Python.
- Axes d'amélioration :
 - Communication avec le groupe.
 - L'organisation personnelle.
 - S'améliorer dans le codage en LaTeX.

Table des illustrations

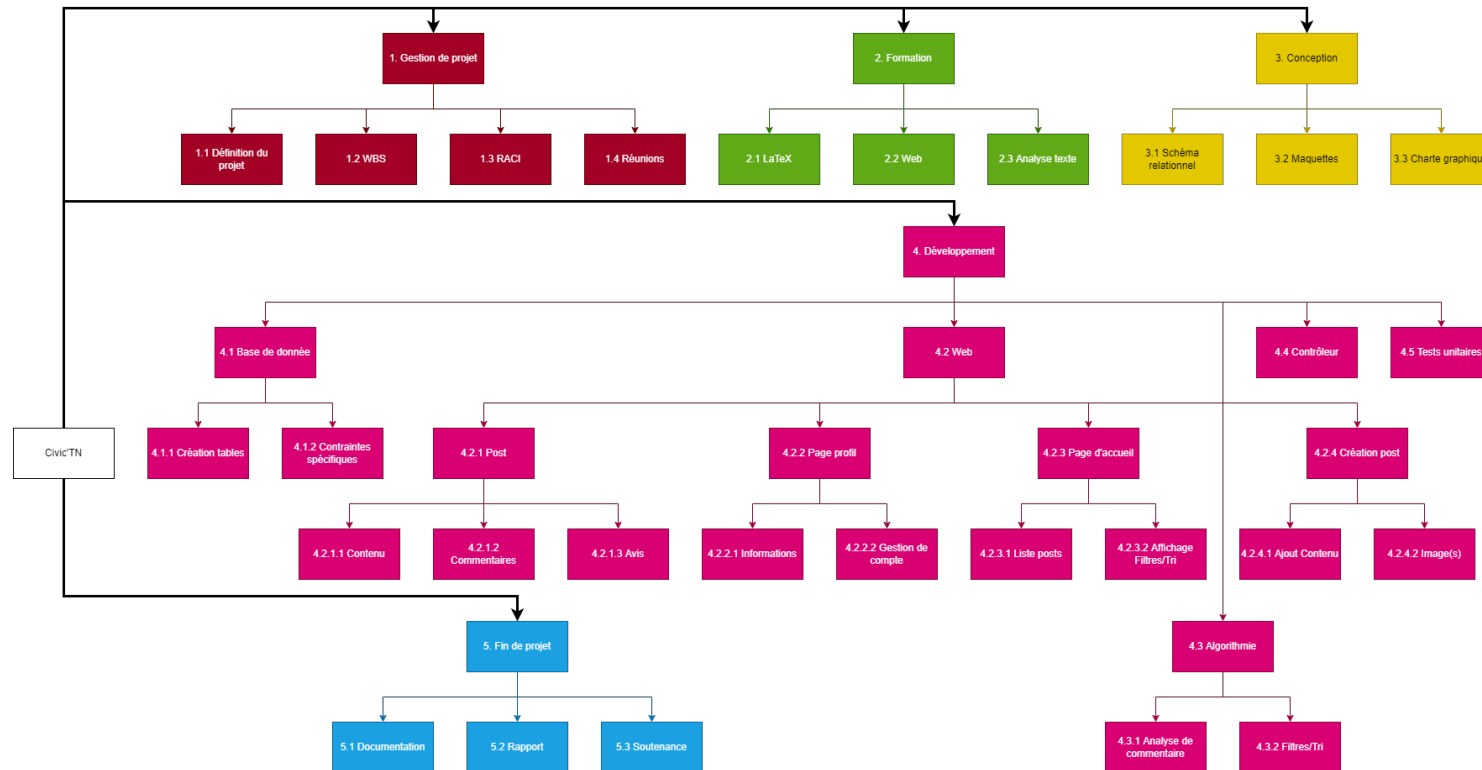
1	Matrice SWOT	6
2	WBS	7
3	Maquette page principale	18
4	Résultat page principale	19
5	Maquette post	20
6	Résultat post	21
7	Page de profil utilisateur	22
8	Page de profil de l'administration	23
9	Création de post	24

6 Annexes

Annexe 1 : Tableau comparatif de l'état de l'art

Applications	Déployabilité de l'application	Ergonomie	Ville employant le service	Services proposés	Payante ou non
FluiCity	Local	Assez facile d'accès	+de 100 villes et + de 500 consultations citoyennes lancées	-Budget participatif -Concentration -Projets -Election -Signalement -Enquêtes	Payante
Citizenlab	Local	Difficile à prendre en main pour des personnes peu familières avec l'informatique	+ de 300 collectivités locales	- Proposition d'idées - Budget participatif - Sondages - Enquêtes - Informations	Payante
We Sign It	Mondial	Facile d'accès et claire	Les pétitions des citoyens peuvent impacter leur ville. Ce n'est pas la ville qui initie l'utilisation de cette plateforme	-Pétitions	Gratuite Avec système de dons
Change.org	Mondial	Facile d'accès et claire	Ce n'est pas la ville qui initie l'utilisation de cette plateforme	- Pétitions	Gratuite Avec système de dons
Tell My City	Local	Très Facile d'accès et d'utilisation	Aucune information précise sur le nombre de commune utilisant l'application	- Actualités - Idées - Consulter les citoyens	Gratuite
Pop Vox	Local	Facile d'accès et simple d'utilisation	+ de 20 villes dont certaines importante en France comme Biarritz	- Budget Participatif - Boîte à idées - commissions - Alertes - Consultations	Gratuite pour le citoyen
Cocoriko	Local	Facile d'accès	Aucune information	- Sondages - Commentaires - Rapports détaillés - Idées	Payante

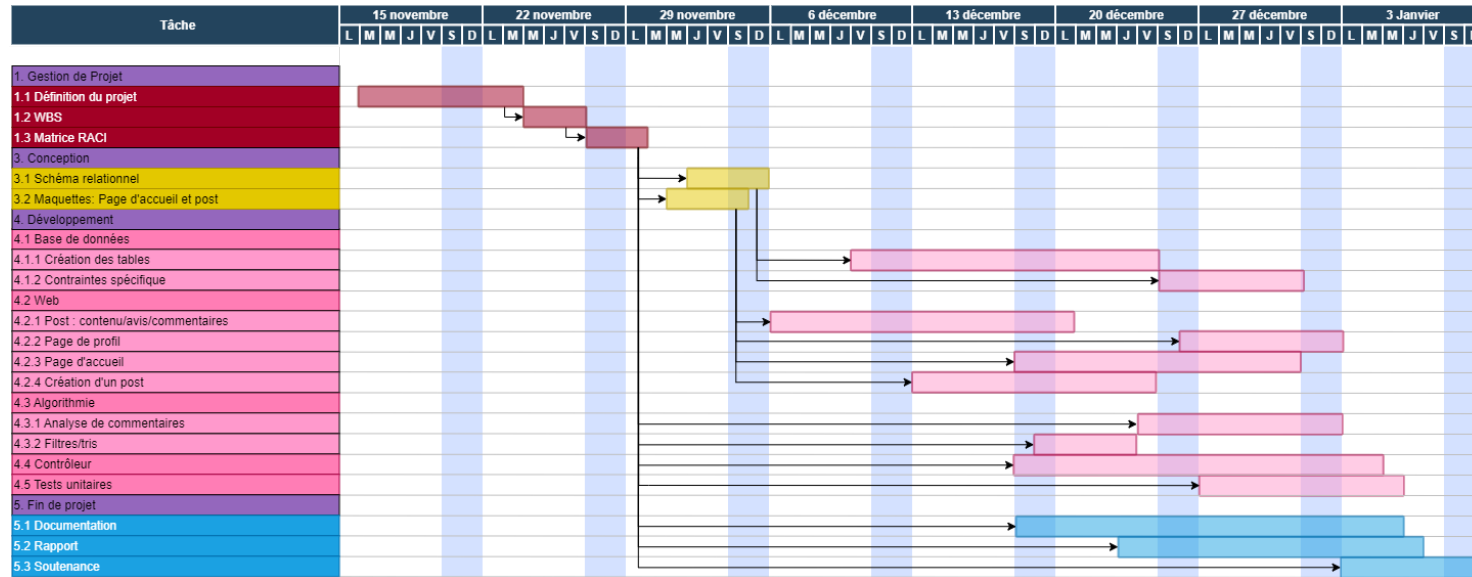
Annexe 2 : Work Breakdown Structure



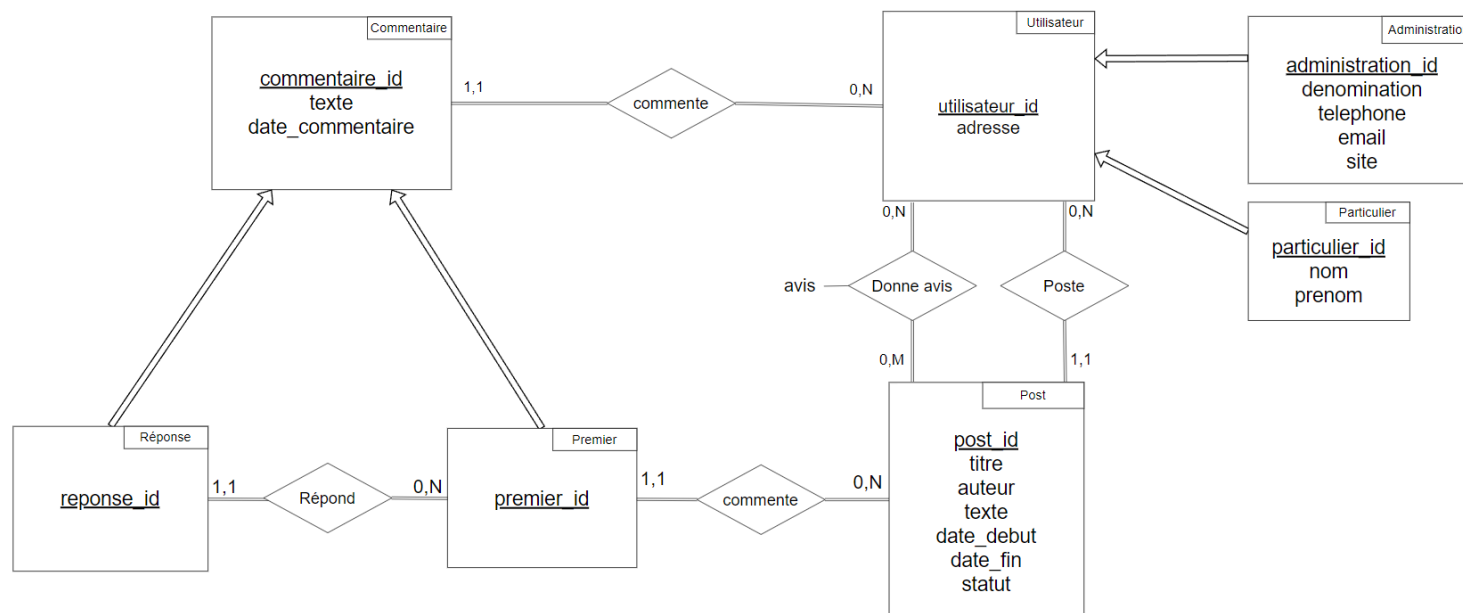
Annexe 3 : Matrice RACI (I : informé, A : Autorité, R : Réalise)

	Ridel Julien	Dénès François	Trabelsi Youssef
Base de données 4.1	—	—	—
Création tables 4.1.1	I	A	R
Contraintes spécifiques 4.1.2	I	R/A	I
WEB 4.2	—	—	—
POSTS 4.2.1	—	—	—
Contenu 4.2.1.1	R	R	I
Commentaires 4.2.1.2	I	R	R
Avis 4.2.1.3	I	A	R
Page Profil 4.2.2	—	—	—
Informations 4.2.2.1	I	R/A	R
Gestion de compte 4.2.2.2	I	R/A	I
Page d'accueil 4.2.3	—	—	—
Liste posts 4.2.3.1	R	R/A	I
Filtres/Tri 4.2.3.2	R	A	I
Création post 4.2.4	—	—	—
Contenu 4.2.4.1	R	R/A	I
Image(s) 4.2.4.2	R/A	I	I
Algorithmie 4.3	—	—	—
Analyse de commentaire 4.3.1	I	A	R
Filtres/Tri 4.3.2	R	A	I
Contrôleur 4.4	I	R/A	I
Tests unitaires 4.5	I	R/A	I

Annexe 4 : Diagramme de Gantt



Annexe 5 : Schéma relationnel

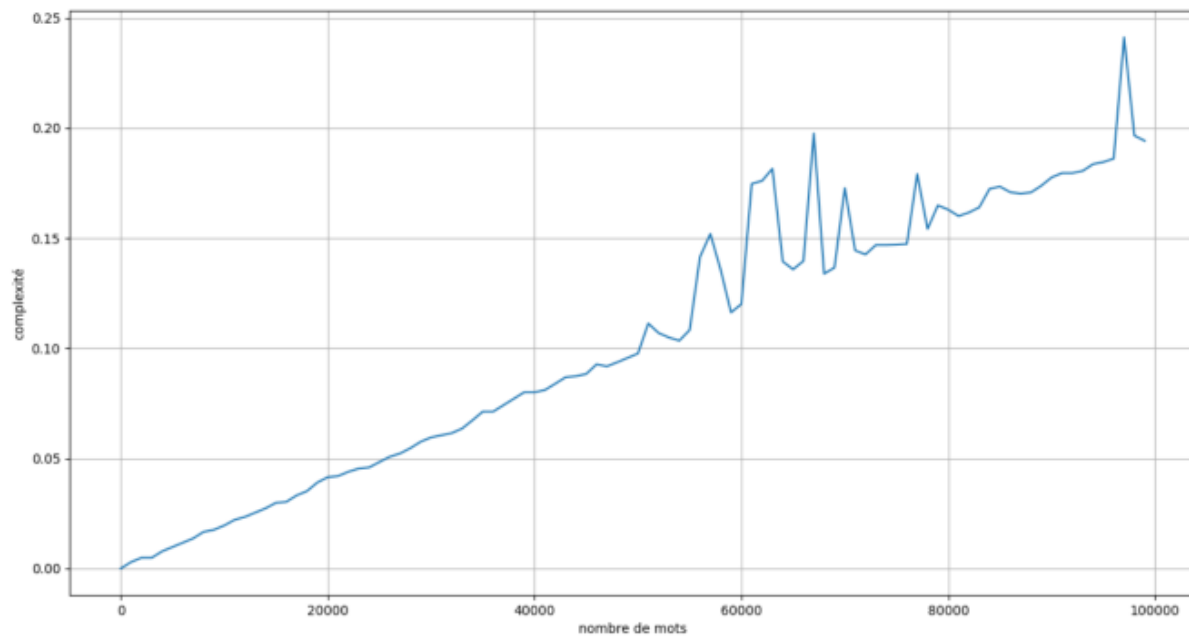


Annexe 6 : Dictionnaire de données

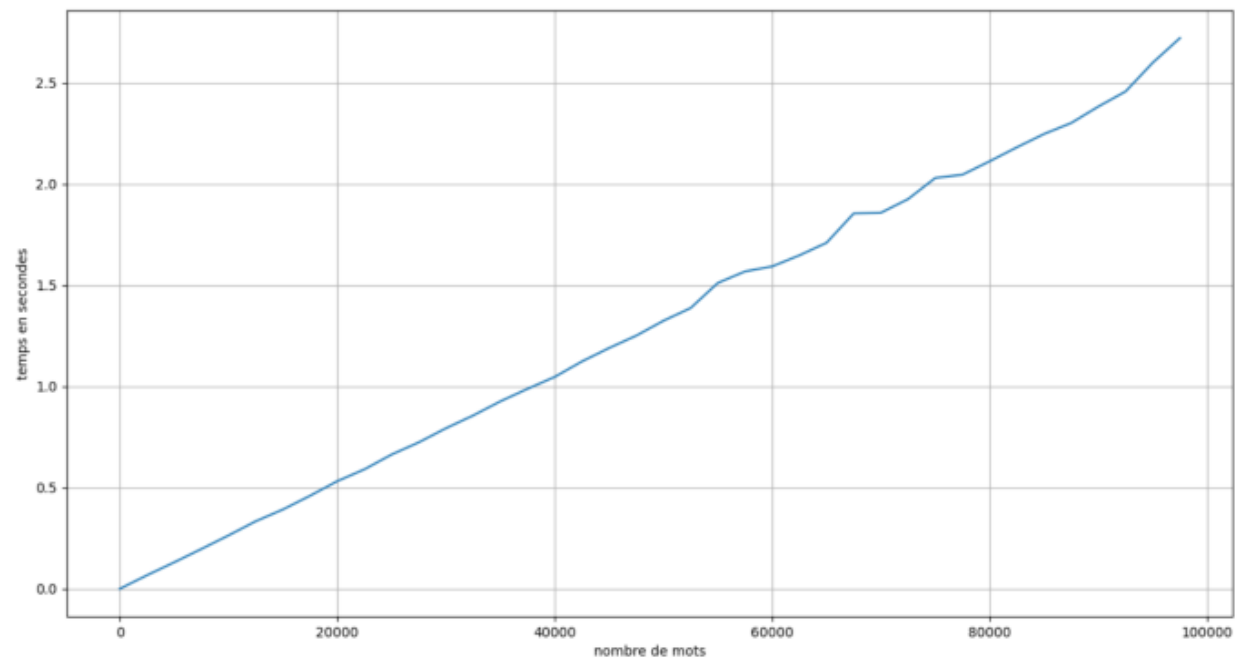
Nom d'attribut	Définition	Atomique	Domaine	Monovalué	Obligatoire	Calculé
utilisateur_id	L'attribut qui définit un utilisateur: un particulier ou une administration	Oui	Integer	Oui	Oui	Non
adresse	L'adresse de l'utilisateur	Oui	Varchar	Oui	Oui	Non
administration_id	L'attribut qui définit une administration	Oui	Integer	Oui	Oui	Non
denomination	Le nom de l'administration	Oui	Varchar	Oui	Oui	Non
telephone	Le numéro de téléphone de l'administration	Oui	Varchar	Oui	Oui	Non
email	L'adresse mail de l'administration	Oui	Varchar	Oui	Oui	Non
Site	Le site web de l'administration	Oui	Varchar	Oui	Oui	Non
particulier_id	L'attribut qui définit un particulier	Oui	Integer	Oui	Oui	Non
nom	Le nom du particulier	Oui	Varchar	Oui	Oui	Non
prénom	Le prénom du particulier	Oui	Varchar	Oui	Oui	Non
avis	L'avis de l'utilisateur sur un post	Oui	{"Pour","Contre","Indifférent"}	Oui	Oui	Non

Nom d'attribut	Définition	Atomique	Domaine	Monovalué	Obligatoire	Calculé
post_id	L'attribut qui définit un post	Oui	Integer	Oui	Oui	Non
titre	Le titre du post	Oui	Varchar	Oui	Oui	Non
auteur	Identifiant de l'auteur post	Oui	Integer	Oui	Oui	Non
texte	Le texte du post	Oui	Varchar	Oui	Oui	Non
date_debut	La date à laquelle le post a été posté	Oui	Varchar	Oui	Oui	Non
date_fin	La date à laquelle le post se termine	Oui	Varchar	Oui	Oui	Non
statut	Le statut du post: indique si la date de fin est passée	Oui	Booléen	Oui	Oui	Oui
commentaire_id	L'attribut qui définit un commentaire	Oui	Integer	Oui	Oui	Non
texte	Le texte commentaire	Oui	Varchar	Oui	Oui	Non
date_commentaire	La date de soumission du commentaire	Oui	Varchar	Oui	Oui	Non
premier_id	L'attribut qui définit le commentaire d'un post	Oui	Integer	Oui	Oui	Non
reponse_id	L'attribut qui définit la réponse à un commentaire	Oui	Integer	Oui	Oui	Non

Annexe 7 : Courbes des tests de performance effectués



Complexité de la fonction Find-words



Complexité de la fonction Avis