



Projet Pluridisciplinaire 1A 21/22  
Module : CS54

## Démocratie participative

Matéo DEMANGEON  
Nicolas FRACHE  
Louis VERMEL  
Clément WENGER



# Table des matières

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Contexte . . . . .	4
<b>2</b>	<b>État de l'art</b>	<b>5</b>
2.1	Objectif et démarche . . . . .	5
2.2	Définitions . . . . .	5
2.3	Tableau synthétique de l'analyse de l'existant . . . . .	5
2.4	Bilan . . . . .	7
<b>3</b>	<b>Charte de projet</b>	<b>8</b>
3.1	Cadrage - présentation du concept . . . . .	8
3.1.1	Étude d'opportunité . . . . .	8
3.1.2	Présentation du concept d'application . . . . .	8
3.2	Description fonctionnelle . . . . .	9
3.2.1	Liste des fonctionnalités du système . . . . .	9
3.2.2	Diagramme de cas d'utilisation . . . . .	9
<b>4</b>	<b>Gestion de projet</b>	<b>11</b>
4.1	Organisation générale du projet . . . . .	11
4.1.1	Organisation . . . . .	11
4.1.2	Découpage en lots de travail . . . . .	11
4.1.3	Analyse de l'environnement - matrice SWOT . . . . .	12
4.1.4	Diagramme de Gantt . . . . .	12
4.2	Outils de travail . . . . .	13
4.2.1	Gestionnaire de versions . . . . .	13
4.2.2	Outils de développements . . . . .	13
4.2.3	Rédaction du présent rapport . . . . .	13
<b>5</b>	<b>Fonctionnement et structure de l'application</b>	<b>14</b>
5.1	Base de données . . . . .	14
5.1.1	Schéma de la base de données . . . . .	14
5.1.2	Implémentation du schéma . . . . .	14
5.1.3	Utilisation avec Python . . . . .	15
5.2	Structure de l'application . . . . .	16
5.3	Serveur web et déploiement . . . . .	17

5.3.1	Flask . . . . .	17
5.3.2	Système de template . . . . .	17
5.3.3	Déploiement sur un serveur . . . . .	18
<b>6</b>	<b>Tests</b>	<b>19</b>
6.1	Démarche de tests . . . . .	19
6.2	Exemples . . . . .	20
<b>7</b>	<b>Conclusion</b>	<b>21</b>
7.1	Bilan sur le livrable . . . . .	21
7.2	Bilan personnel . . . . .	21
7.2.1	Points positifs . . . . .	21
7.2.2	Difficultés rencontrées . . . . .	21
<b>8</b>	<b>Annexes</b>	<b>22</b>
.1	Configuration du serveur . . . . .	22
.1.1	Configuration d'Apache . . . . .	22
.1.2	Configuration en Python . . . . .	23
.2	Galerie . . . . .	24
.2.1	Page d'accueil . . . . .	24
.2.2	Page d'une conférence . . . . .	25
.2.3	Page de création de questionnaire . . . . .	26
.3	Comptes-rendus des réunions . . . . .	26

# Chapitre 1

## Introduction

### 1.1 Contexte

2022 sera une année électorale majeure durant laquelle les français seront invités à élire leur président ainsi que leurs représentants à l'assemblée Nationale. On constate cependant un désengagement important des français de la vie politique. En effet, pour une part croissante d'entre eux, un vote tous les cinq ans n'est pas une participation suffisante pour se sentir au coeur de la vie démocratique.

Le concept de démocratie participative, né il y a plus de 50 ans, est une solution avancée pour une meilleur écoute de la volonté des citoyens. Il vise à dynamiser l'engagement et renforcer l'impact des citoyens dans la prise de décision et dans le développement de la société à toutes les échelles : locale, régionale et nationale.

Le référendum est le premier moyen qui vient à l'esprit quand on parle de démocratie participative, mais il est toutefois limité car son organisation est longue et coûteuse. Renforcer le rôle des citoyens dans les prises de décisions politiques et sociales impose donc la mise en place de procédures plus simples, plus rapides et donc plus adaptées à l'échelle locale.

Ainsi le numérique semble particulièrement indiqué pour cet enjeu de société. Que ce soit pour sa facilité d'utilisation, ou pour la logistique qui s'en retrouve grandement simplifiée.

De nombreux logiciels dédiés, regroupés sous l'appellation Civic Tech, offrent ainsi toute une gamme de services (consultations citoyennes, pétitions, espaces de discussion, d'idéation, de co-création, etc) sous diverses formes et licences (applications propriétaires, logiciels open sources, ...).

L'objectif de ce projet est ainsi de concevoir et implémenter une application permettant de contribuer activement à la démocratie participative.

## Chapitre 2

# État de l’art

### 2.1 Objectif et démarche

La première étape à réaliser dans le cadre de notre projet est une étude de l’existant dans le domaine de la Civic Tech et de la démocratie participative. C’est une étape obligatoire avant tout projet qui se veut innovant afin de comprendre au mieux le cadre et les attentes du sujet. Cela permet également de nourrir nos réflexions autour de notre application.

### 2.2 Définitions

Commençons par définir les termes de démocratie participative et Civic Tech.

La démocratie participative vise à promouvoir l’implication et la participation des citoyens dans les prises de décision. Les autorités politiques invitent donc les citoyens à exprimer leurs avis ou faire des propositions.

La Civic Tech ou technologie civique désigne toutes les initiatives et technologies mise en place pour faciliter la démocratie participative, en permettant aux citoyens de jouer un rôle plus important dans les prises de décision.

### 2.3 Tableau synthétique de l’analyse de l’existant

Le tableau ci-dessous a pour vocation de synthétiser et de structurer nos recherches sur les applications existantes. Pour cela, nous allons nous servir d’un certain nombre de critères qui seront regroupés dans le tableau de la section suivante :

- Nom de l’application et visuel,
- Échelle de l’application (village, ville, région...),
- Type d’application (en quelques mots),
- Description :
  - Fonctionnalités principales,
  - Points d’intérêts particuliersj
  - Limites.

Nom	Échelle	Type	Fonctionnalités d'intérêt
Fluicity	Collectivité locale/région (Thionville, Wallonie)	Plateforme	L'organisation passe commande à Fluicity pour développer son application. <ul style="list-style-type: none"> <li>• Sondages/votes,</li> <li>• Consultations sur des sujets précis,</li> <li>• Enquêtes sous formes de questionnaires.</li> </ul>
DansMaRue	Locale (ville de Paris)	Application et site internet	<ul style="list-style-type: none"> <li>• Permet aux citoyens d'effectuer des signalements d'anomalie détaillés (nature, lieu, photo, niveau de priorité),</li> <li>• Suivi des signalements,</li> <li>• Signalements remontés aux services compétents.</li> </ul>
Demodyne	De locale à nationale	Site internet	<ul style="list-style-type: none"> <li>• Soumettre des propositions aux institutions gouvernementales,</li> <li>• Trier ces propositions par priorité.</li> </ul>
Projet Arcadie	Nationale (France)	Site internet	Base de données sur les parlementaires français : <ul style="list-style-type: none"> <li>• Fiche personnelle pour chaque parlementaire,</li> <li>• Informations complémentaires (mandats cumulés, profession...),</li> <li>• Possibilité de faire des recherches croisées complexes avec une interface simple.</li> </ul>
Nancy - Budget participatif	Locale (ville de Nancy)	Site internet	<ul style="list-style-type: none"> <li>• Proposition d'aménagements par les citoyens,</li> <li>• Vote populaire pour les meilleures propositions,</li> <li>• Communications sur les résultats de réunions techniques intégrant les différents acteurs pour l'attribution précise des budgets.</li> </ul> <p>Note : Techniquement conçu avec Decidim (voir point suivant).</p>
Decidum	Toute organisation (à l'origine la ville de Barcelone)	Plateforme	En se servant du code source de Decidim les organisations peuvent créer des applications comprenant notamment les fonctionnalités suivantes : <ul style="list-style-type: none"> <li>• Budget participatif (exemple de Nancy),</li> <li>• Consultation citoyenne,</li> <li>• Planification stratégique.</li> </ul>
Crossiety	Locale / tout type de groupe (Villes suisses ou allemandes)	Site internet et Application mobile	L'idée est de proposer "une place du village" virtuelle. <ul style="list-style-type: none"> <li>• Organisation d'événements,</li> <li>• Proposition d'initiatives,</li> <li>• Chat local et groupes de discussion,</li> <li>• Marché en ligne et Aide de quartier.</li> </ul> <p>Aussi adapté aux groupes de travail en entreprise ou à l'école.</p>

Change.org	Internationale, nationale ou locale	Site internet	Site international indépendant permettant de lancer et voter pour des pétitions. <ul style="list-style-type: none"> <li>• Chacun (particulier ou association) peut lancer sa pétition.</li> <li>• Permet d'interagir avec la presse et les décideurs et de fixer des objectifs à atteindre.</li> <li>• Permet de s'opposer à ou de soutenir des projets locaux.</li> </ul>
------------	-------------------------------------	---------------	--

## 2.4 Bilan

On constate que la majorité des projets de notre analyse sont des sites internet destinés à des localités. Ces sites ont pour point commun de permettre aux individus ou aux associations de faire des propositions, des sondages ou des pétitions. Un système de notation permet ensuite aux autorités de discerner lesquels sont les plus plébiscités.

Une autre partie des applications consistent en des plateformes récoltant et mettant à disposition de tous des informations utiles au débat démocratique.

Ainsi cet état de l'art oriente nos réflexions principalement autour de ces deux aspects de la démocratie participative. En particulier, notre attention s'est portée sur la complexité apparente de mise en place des solutions de consultation des populations.

Il nous semble probable qu'un élu de village peu habitué au support numérique et ayant peu de moyens, peine à franchir le pas avec les solutions actuelles.

## Chapitre 3

# Charte de projet

### 3.1 Cadrage - présentation du concept

#### 3.1.1 Étude d’opportunité

Au cours de nos recherches nous avons découvert le concept de conférence des citoyens. C’est une méthode de consultation des habitants d’une commune qui repose sur le tirage au sort d’un nombre réduit d’entre eux (généralement une vingtaine). Les tirés au sort sont alors formés sur le sujet de la consultation et leur avis sert à l’établissement d’un consensus. Sauf justification, la mairie a alors l’obligation de suivre le résultat de la consultation.

En France, il y a eu un nombre relativement réduit de conférences de citoyens à l’échelle nationale, mais on peut citer par exemple celle de 1998 sur les organismes génétiquement modifiés, ou encore celle autour de Cigéo, le centre de stockage géologique de Bure, en 2013.

L’objectif de notre application est donc d’organiser et de gérer ces conférences de citoyens pour une ville donnée, que ce soit du côté des citoyens qui seront ou non tirés au sort, ou du côté des organisateurs de la consultation. L’accent doit être mis sur la simplicité d’utilisation pour des personnes peu à l’aise avec les outils numériques.

#### 3.1.2 Présentation du concept d’application

Voici une courte présentation du fonctionnement de notre concept d’application que nous élaborons avec différents outils de gestion de projet par la suite.

Pour organiser le tirage au sort, l’application a besoin d’une base de données avec la liste de tous les citoyens et à minima leur nom. Notre application se destine à un village ou une petite ville ce qui rend plus facile la gestion de cette base. Un administrateur peut gérer la base de données en ajoutant ou supprimant un citoyen au besoin. Des identifiants de connexion pour tous les comptes sont générés automatiquement.

Quand un citoyen est tiré au sort son mot de passe lui est transmis par voie postale, il peut alors accéder à l’application et compléter son profil (âge, situation professionnelle, études...). Et bien sûr, changer son mot de passe.



L'utilisateur a alors accès à un espace qui présente la problématique et les enjeux de la conférence des citoyens, ainsi qu'à la liste des formations et des séances d'échanges qui sont organisés.

Finalement, l'utilisateur peut répondre à une série de questionnaire autour de la question initiale depuis le site web.

À l'issue du processus, une fiche librement accessible présente l'ensemble du déroulement de la conférence de citoyens et également les réalisations concrètes de la mairie qui vont la suivre.

## 3.2 Description fonctionnelle

### 3.2.1 Liste des fonctionnalités du système

Gestion de compte :

- Connexion à un compte
- Modification du profil

Utilisation :

- Du point de vue de la mairie / de l'instance décisionnaire :
  - Création de questionnaires avec différents types de réponses possibles
  - Envoi des questionnaires à un nombre donné de citoyens de la base tirés au sort (avec éventuellement des paramètres sur leur profil)
  - Consulter les résultats aux questionnaires
- Du point de vue du citoyen :
  - Première connexion via des identifiants fournis par la mairie / l'instance décisionnaire
  - Possibilité de compléter son profil
  - Réponse aux éventuels questionnaires reçus de la mairie / l'instance décisionnaire
  - Possibilité de consulter les thèmes abordés (même ceux pour lesquels on a pas été interrogés)

### 3.2.2 Diagramme de cas d'utilisation

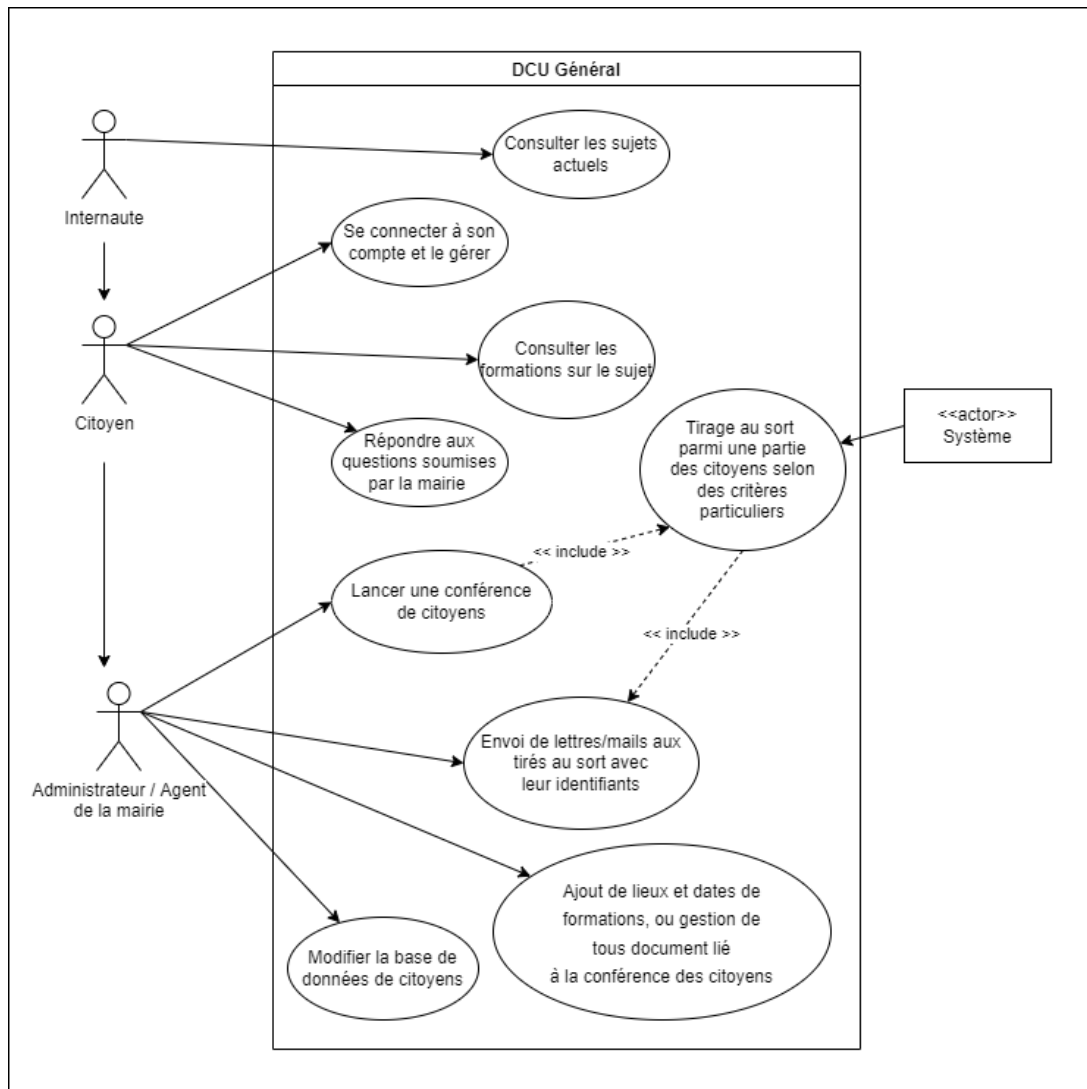


FIGURE 3.1 – Diagramme de cas d'utilisation général initial

## Chapitre 4

# Gestion de projet

### 4.1 Organisation générale du projet

#### 4.1.1 Organisation

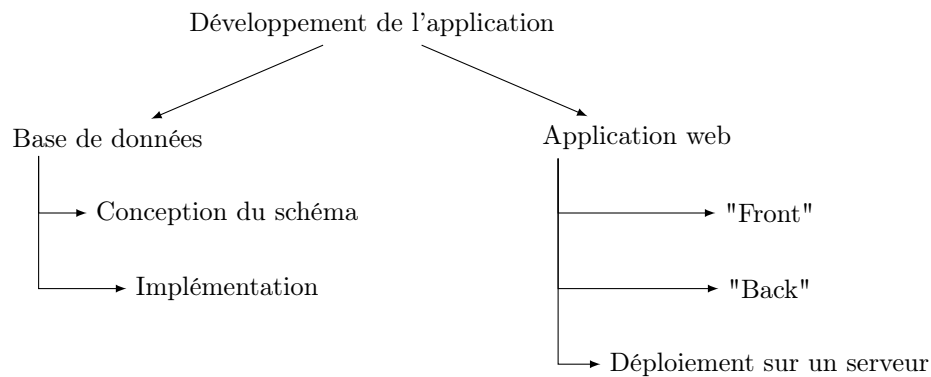
Au vue des délais restreints pour la réalisation de notre livrable, nous avons décidé d’opter pour la méthode AGILE. En effet, celle-ci permet entre autres de découper le projet en itérations (sprint) à l’issue desquelles nous aurons un livrable fonctionnel (même si incomplet dans l’état).

Nous avons choisi des sprints particulièrement courts d’une semaine. À la fin de celles-ci, nous faisons un bilan sur l’accomplissement des objectifs prévus et nous définissons les objectifs de la semaine suivante.

Des réunions ont été tenues régulièrement et leurs comptes-rendus sont disponibles en annexe (voir .3).

#### 4.1.2 Découpage en lots de travail

Nous avons dans un premier temps découpé notre projet en une WBS (Work Breakdown Structure) permettant de définir les principaux lots de travail de l’application qui sont pour l’instant très généraux :



#### 4.1.3 Analyse de l'environnement - matrice SWOT

La matrice SWOT permet d'évaluer la pertinence et l'environnement de notre projet en identifiant les points positifs et négatifs qui s'imposent à nous.

	Positif	Négatif
Interne	Compétences dans le langage Python, Connaissances dans la gestion de base de données	Faible connaissance concernant Flask, nécessité de se former, Difficultés à anticiper le déroulement temporel du projet par manque d'expérience
Externe	Application innovante de par sa simplicité d'utilisation, Engouement croissant pour une démocratie plus participative	Délais très restreint pour l'ensemble du projet

#### 4.1.4 Diagramme de Gantt

Le diagramme de Gantt (fig. 4.1) permet une représentation synthétique et visuelle de la répartition dans le temps des principaux lots de travail du projet.

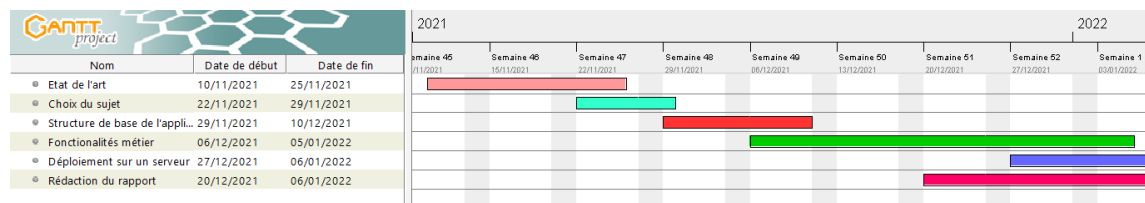


FIGURE 4.1 – Diagramme de Gantt

## 4.2 Outils de travail

### 4.2.1 Gestionnaire de versions

Afin de pouvoir collaborer efficacement dans le développement de l'application, l'équipe s'est servie du gestionnaire de version Git. Le dépôt Git est hébergé sur un serveur GitLab privé fourni à l'équipe projet. Les documents de conception et de gestion de projet ont quant à eux été déposés dans un dossier partagé Google Drive afin de permettre leur édition en collaboration simultanée. Ils ont été par la suite transférés au dépôt Git.

### 4.2.2 Outils de développements

Dans le cadre du développement de l'application Python, l'équipe projet a opté pour l'environnement de développement *Pycharm*. C'est un outil payant de l'éditeur JetBrains spécialisé pour le langage Python.

Pour faciliter l'utilisation de Git, l'équipe a utilisé l'interface graphique *GitKraken*. Ainsi que le logiciel gratuit *Beyond Compare 3* pour la résolution des conflits.

### 4.2.3 Rédaction du présent rapport

La rédaction du rapport en  $\text{\LaTeX}$  a été effectuée par l'intermédiaire d'Overleaf. Ce système permet la collaboration en temps réel pour l'écriture des documents utilisant ce langage.

## Chapitre 5

# Fonctionnement et structure de l'application

### 5.1 Base de données

#### 5.1.1 Schéma de la base de données

La base de données relationnelle permet d'enregistrer toutes les données nécessaires au fonctionnement de l'application de manière structurée. La conception de son schéma <sup>1</sup> nécessite une bonne assimilation des besoins fonctionnels de l'application.

Ci-dessous une représentation schématique du schéma de la base de données que nous avons choisi pour enregistrer les données de notre application (fig. 5.1).

On remarque que le schéma de la base est principalement structuré autour du système de questionnaire. En effet une relation **Questionnaire** est naturellement associée à une liste de **Questions** numérotées par ordre d'apparition dans le questionnaire. Dans le cas des questions à choix multiple on associe alors la question à une liste de relations **ChoixQCM**.

Dans un second temps on s'intéresse à la gestion des réponses à ces questions. Pour représenter une réponse à un type de question on choisit ainsi d'associer à un utilisateur soit une question soit un choix de réponse à l'aide d'une association binaire ou ternaire.

#### 5.1.2 Implémentation du schéma

Le système de gestion de base de données (SGBD) est le système qui manipule cette base de données. Il en existe un nombre important. Parmi les principaux on peut citer Oracle, MySQL ou encore PostgreSQL.

Mais pour ce projet, nous avons choisi SQLite. Celui-ci a la particularité de contenir la base de données sous la forme d'un unique fichier. L'absence de système d'authentification et la simplicité de manipulation de la base permet ainsi de simplifier les opérations dans le cadre d'un projet de

---

1. Le schéma de la base de données désigne l'ensemble des contraintes qui l'organisent d'un point de vue structurel. Le schéma est donc indépendant du système choisi pour implémenter la base de données.

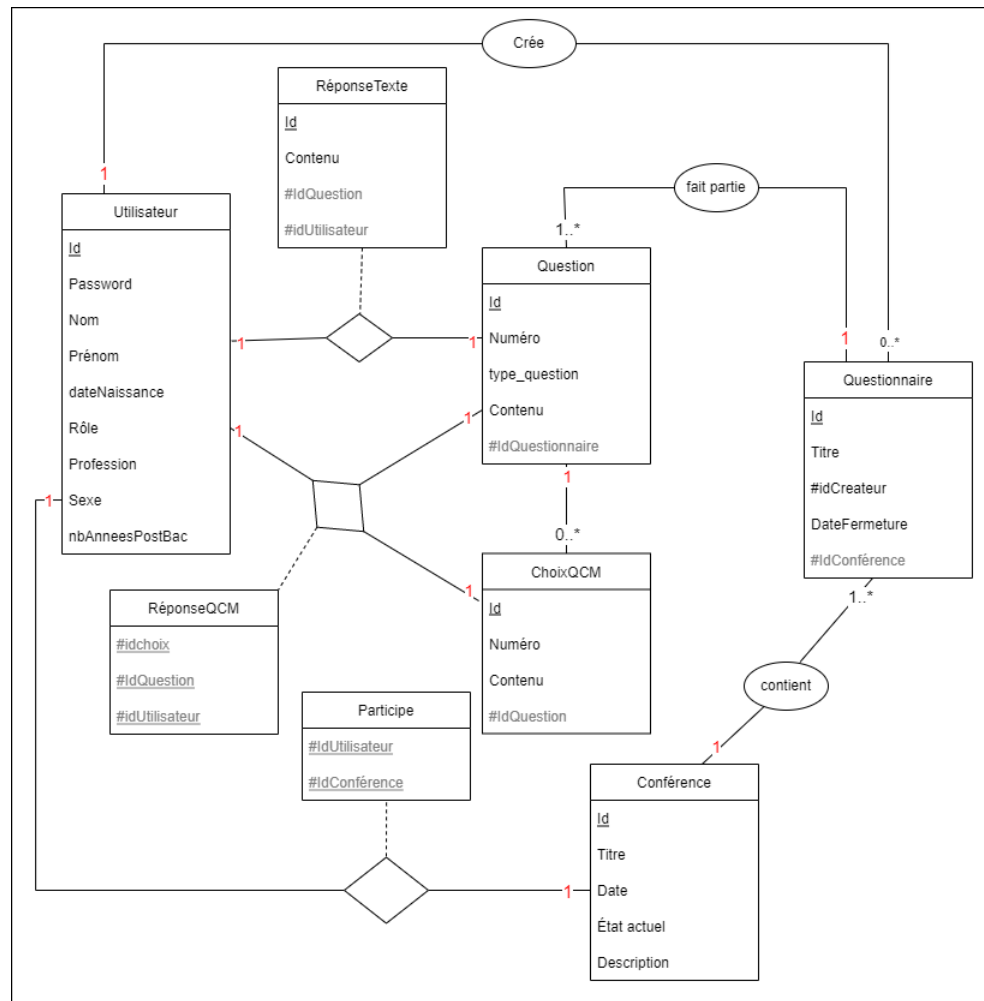


FIGURE 5.1 – Diagramme du schéma de la base de données

petite taille ou d'un prototype. Toutefois ce SGBD limite les possibilités de mise à l'échelle et impose donc un changement en cas de mise en production en conditions réelles.

### 5.1.3 Utilisation avec Python

Pour interfacer l'application Python avec la base de données, il est possible d'utiliser un ORM (object relationnal mapper). C'est un système qui ajoute un niveau d'abstraction supplémentaire en représentant un tuple de la base par un objet du langage et en permettant ainsi sa manipulation au travers de méthodes prédéfinies.

Pour notre application c'est l'ORM *SQLAlchemy* que nous avons utilisé, et plus particulièrement sa variante *Flask-SQLAlchemy* optimisée pour Flask. Par conséquent il a fallu écrire les classes

représentant les tables de la base. L'utilisation de la bibliothèque Python *Flask sqlacodegen* nous a permis d'automatiser partiellement cette tâche en convertissant automatiquement le schéma de la base de données.

Afin de pouvoir générer un grand nombre de citoyens de manière aléatoire et automatique, nous avons opté pour la bibliothèque *Faker*. Celle-ci permet en effet de tirer aléatoirement un attribut (nom, prénom, profession ou encore mot de passe) parmi un large répertoire. De plus, nous avons configuré la bibliothèque pour obtenir des résultats en langue française.

## 5.2 Structure de l'application

La structure de l'application a été conçue comme une arborescence partant de la page d'accueil (fig. 5.2). Ainsi sur la page */conference* on a accès à la liste des conférences de citoyens existantes et on a également la possibilité d'en créer une nouvelle à condition que l'utilisateur ait les privilèges administrateur. De la même manière, depuis la page d'une conférence de citoyens, la liste des questionnaires permet de les visualiser, d'y répondre et de voir les résultats.

Des captures d'écrans des page d'accueil, de conférence et de création de questionnaire sont disponibles en annexe .2 pour avoir une idée de l'identité visuelle de l'application.

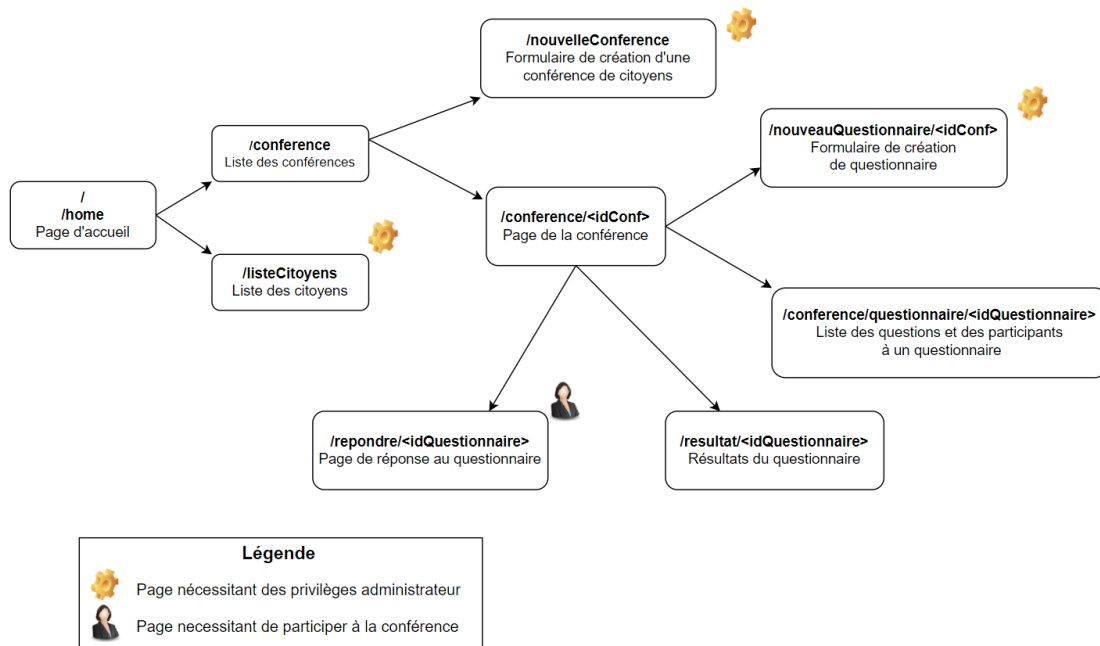


FIGURE 5.2 – Diagramme d'arborescence du site



## 5.3 Serveur web et déploiement

### 5.3.1 Flask

Nous avons développé le serveur de notre application à l'aide du framework open-source *Flask*. Celui-ci a la particularité d'être léger par rapport à la plupart des autres outils de développement web qui nécessitent des configurations extensives.

Ainsi, pour chacune des routes imaginées dans la structure de l'application (voir section 5.2), nous avons créé une fonction. Cette fonction a accès à la requête HTTP et à ses paramètres, le contenu d'un formulaire par exemple, et retourne le contenu HTML correspondant à la page qui sera chargée par le navigateur.

### 5.3.2 Système de template

La génération du HTML se fait à l'aide du moteur de rendu de template *jinja2*. Aussi ce moteur permet d'intégrer des variables et de la logique dans le template HTML, ce qui est essentiel au bon fonctionnement de notre application.

The screenshot shows a web application interface for 'Conf-Cit'. The page title is 'Conférence de citoyens: Mobilité dans le centre-ville'. It contains a description of the conference, a list of associated questionnaires, and a table of participants. The table has columns for Numéro, Nom, Prénom, Sexe, Profession actuelle, and Date de Naissance. The participants listed are Denis Thierry, Humbert Hortense, Bourdon Margeux, Schmitt Céline, Didier Benoît, and Moulin Victor. Colored arrows point to different parts of the page, indicating the template files used: header.html (green), base.html (red), conference.html (pink), and data\_tab.html (yellow).

Numéro	Nom	Prénom	Sexe	Profession actuelle	Date de Naissance
216	Denis	Thierry	H	administrateur de spectacle	1980-02-16
226	Humbert	Hortense	H	photographe	1982-06-09
727	Bourdon	Margeux	F	géotechnicien	1963-02-03
171	Schmitt	Céline	H	serrurier dépanneur dépanneuse	1946-06-07
317	Didier	Benoît	H	faconnier des industries graphiques	1963-11-12
908	Moulin	Victor	H	coordonnateur d'études cliniques	1980-06-30

FIGURE 5.3 – Représentation de l'implication des différents fichiers de template pour le chargement de la route `/conference/<id>`

*Jinja2* comprend un système d'héritage et d'inclusion de templates permettant d'éviter toute redondance. La page représentée sur la figure 5.3 illustre notre utilisation de ces concepts dans l'application :

- Le template *conference.html* hérite de *base.html*, donc à son chargement il se retrouve englobé dans ce dernier.

- Le template *base.html* inclut quant à lui le template correspondant à l'entête du site (*header.html*). Ainsi qu'un champ central dont le contenu est défini par le template enfant, ici *conference.html*.
- Enfin, la liste des participants dans le template *conference.html* est obtenue avec une inclusion du template *data\_tab.html* qui est utilisé à plusieurs reprises dans l'application.

### 5.3.3 Déploiement sur un serveur

Pour déployer notre application sur un serveur web publique nous avons mis à contribution le VPS<sup>2</sup>, détenu par un des membres de l'équipe de projet (Nicolas).

Pour configurer le serveur nous avons utilisé le logiciel libre *Apache HTTP Server*. En effet, celui-ci permet de configurer l'écoute sur les ports associés aux requêtes HTTP (80 et 443) et de choisir l'action à effectuer avec la requête en fonction, notamment, de sa partie *host*<sup>3</sup>. L'annexe .1.1 présente l'ensemble du fichier de configuration du site Apache.

De plus afin de réaliser l'interface entre le serveur web et notre application Python, nous utilisons naturellement WSGI (*Web Server Gateway Interface*). Sa configuration nécessite l'ajout d'un fichier *nom\_projet.wsgi* à notre application qui régit le déclenchement de l'application. Ce fichier est disponible en annexe .1.2.

Finalement, au moment de la conclusion de ce projet, notre application est disponible et fonctionnelle à l'adresse suivante : [www.confcit.xyz](http://www.confcit.xyz).

---

2. Un VPS, de l'anglais *virtual private server* ou *serveur privé virtuel* est un système d'hébergement courant. On parle de serveur virtuel car l'hébergeur ne met pas à disposition un serveur dédié proprement parler, mais plutôt une machine virtuelle utilisant une partie des ressources du serveur physique.

3. L'*host* d'une requête HTTP est le champ du header qui indique l'adresse ip d'origine de la requête.

# Chapitre 6

## Tests

### 6.1 Démarche de tests

Nous avons réalisé différents tests dans le but de vérifier le bon fonctionnement de certains éléments de l'application. Pour cela, nous avons utilisé Pytest, l'outil de test par excellence de l'environnement de Python.

Ces tests se divisent en deux catégories : les tests unitaires et les tests fonctionnels.

- Les tests unitaires permettent de vérifier qu'une méthode particulière se comporte bien comme attendu.
- Les test fonctionnels, quant à eux, simulent une utilisation standard de l'application pour tester une fonctionnalité en conditions réelles. Dans le cas de notre application Flask, cela se traduit principalement par l'utilisation d'un mock<sup>1</sup> simulant l'envoi des requêtes HTTP par un utilisateur.

Pour pouvoir réaliser des tests sur la base de données, il est nécessaire de supprimer toute les données qu'elle contient à la fin de chaque test. Dans un soucis de factorisation du code nous nous sommes servis d'une fonctionnalité de Pytest conçue dans ce sens, les *fixtures*. Ainsi les instructions de nettoyage de la base de données sont automatisées pour chaque test.

---

1. Un mock (littéralement *bouchon*) désigne, dans le cadre de la programmation objet, un objet dont la fonction est de reproduire le comportement d'un pan de l'application sur lequel le système testé n'a normalement pas le contrôle. Ici cela correspond à l'émission de requêtes HTTP vers l'application qui est normalement effectué par le client.

## 6.2 Exemples

Exemple de test fonctionnel testant si l'application répond correctement à une requête HTTP GET sur la route */home* (vérification du code de réponse HTTP et de la présence du mot "Bienvenue" sur la page) :

```
def test_home_get():
    with app.test_client() as test_client:
        response = test_client.get('/home')
        assert response.status_code == 200
        assert b"Bienvenue" in response.data
```

Exemple de test unitaire très basique vérifiant que la base contient bien un nouvel utilisateur suite à la création d'une instance de la classe Utilisateur (voir section 5.1.3) :

```
def test_utilisateur_nouveau():
    db.session.add(Utilisateur(prenom="prenom", nom="nom"))
    db.session.commit()
    assert Utilisateur.query.count() == 1
    assert Utilisateur.query.first().nom == "nom"
    assert Utilisateur.query.first().prenom == "prenom"
```

# Chapitre 7

## Conclusion

### 7.1 Bilan sur le livrable

Finalement, notre projet répond globalement à la charte de projet initiale (voir la liste initiale des fonctionnalités prévues 3.2.1). Nous avons cependant omis certaines fonctionnalités annexes comme une gestion plus poussées des connexions des utilisateurs ou encore la possibilité de personnaliser son profil, soit par manque de temps, soit car on les a finalement jugé peu pertinentes. Nous n'avons également pas pu implémenter le système de formation par manque de temps.

### 7.2 Bilan personnel

#### 7.2.1 Points positifs

Ce projet nous a poussé à compléter nos compétences dans différents domaines, du développement web jusqu'à la manipulation de base de données en passant par la gestion de projet. Nous sommes globalement satisfaits du livrable final, qui bien qu'incomplet, remplit les principaux objectifs et missions définis dans la charte de projet rédigée en début de travail.

#### 7.2.2 Difficultés rencontrées

Le manque d'expérience du groupe dans les domaines abordés a rendu difficile la planification des différentes tâches, et l'estimation de leur durée. La gestion du projet s'est ainsi révélée complexe. Nous avons également rencontrés quelques difficultés organisationnelles quant au partage des tâches de l'équipe projet.

# Chapitre 8

## Annexes

### .1 Configuration du serveur

#### .1.1 Configuration d'Apache

```
# Configuration d'un hôte virtuel sur le port 80 (http non sécurisé)
<VirtualHost confcit.xyz:80>
    # La configuration de ce site s'applique si la valeur host du header de
    # la requête http correspond à un des deux formats suivant (l'adresse du DNS
    # redirigeant ses requêtes vers ce serveur) :
    ServerName confcit.xyz
    ServerAlias www.confcit.xyz

    # Adresse mail s'affichant en cas d'erreur et avec un whois
    ServerAdmin nicolas.frache@telecomnancy.eu

    # Chemin du fichier python .wsgi qui lie le serveur à l'application Python
    WSGIScriptAlias / /home/nicolas/conf-cit/project1-grp_ell/src/flaskProject/flaskProject.wsgi

    # Autorisation d'accès aux fichiers de l'application
    <Directory /home/nicolas/conf-cit/project1-grp_ell/src/flaskProject/>
        Order allow,deny
        Allow from all
        Options -Indexes
    </Directory>

    # Autorisation d'accès aux fichiers static mais pas à l'affichage des listes de fichiers
    Alias /static /home/nicolas/conf-cit/project1-grp_ell/src/flaskProject/flaskProject/project/static
    <Directory /home/nicolas/conf-cit/project1-grp_ell/src/flaskProject/flaskProject/project/static/>
        Order allow,deny
        Allow from all
        Options -Indexes
    </Directory>

    # Redirection des erreurs dans les fichiers de logs ci-dessous
    ErrorLog /home/nicolas/conf-cit/project1-grp_ell/src/flaskProject/logs/error.log
    LogLevel warn
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

FIGURE 1 – Fichier de configuration */etc/apache2/sites-available/002-conf-cit.conf*

## .1.2 Configuration en Python

```
#!/usr/bin/python
import sys
import logging
import os

# Configuration du path de Python au chemin du projet
logging.basicConfig(stream=sys.stderr)
sys.path.insert(0, "/home/nicolas/conf-cit/project1-grp_e11/src/flaskProject/")

# On importe l'application Flask qui comporte toute les routes et la création de l'app
from flaskProject import app as application
```

FIGURE 2 – Fichier WSGI servant d’interface entre Apache et l’application Python  
*/home/nicolas/conf-cit/project1-grp\_e11/src/flaskProject/flaskProject.wsgi*

## .2 Galerie

Dans cette partie, nous présentons quelques captures d'écran des principales pages du site que nous jugeons représentatives de l'application.

### .2.1 Page d'accueil



FIGURE 3 – Page d'accueil de l'application. Route : / ou /home



## .2.2 Page d'une conférence

Conf-Cit

Connecté en tant que : Nicolas

### Conférence de citoyens: Mobilité dans le centre-ville

**Description :**

Alors que le nombre de cyclistes et de piétons est croissant, cette consultation a pour but de déterminer la place que les habitants souhaitent laisser aux véhicules motorisés dans le centre-ville.

**Questionnaires associés :**

Titre questionnaire	Date de fermeture		
Limitations de vitesse pour les véhicules motorisés.	07/01/2022 23:34	Voir résultats	Répondre
Investissement dans de nouvelles infrastructures cyclables	08/01/2022 16:27	Voir résultats	Répondre

Créer un questionnaire associé à cette conférence

**Participants à la conférence :**

Numéro	Nom	Prénom	Sexe	Profession actuelle	Date de Naissance
216	Denis	Thierry	H	administrateur de spectacle	1980-02-16
226	Humbert	Hortense	H	photographe	1982-06-09
727	Bourdon	Margaux	F	géotechnicien	1963-02-03
171	Schmitt	Céline	H	serrurier dépanneur dépanneuse	1946-06-07
317	Didier	Benoît	H	façonnier des industries graphiques	1963-11-12
908	Moulin	Victor	H	coordonnateur d'études cliniques	1980-06-30

FIGURE 4 – Exemple d'une page de conférence. Route : `/conference/<idConference>`

## .2.3 Page de création de questionnaire

**Conf-Cit** Connecté en tant que : Nicolas

### Création d'un questionnaire

Pour créer un questionnaire associé à la conférence "Mobilité dans le centre-ville" ajoutez autant de questions que vous le désirez ci-dessous puis validez.

**Titre du questionnaire :**

Développement des vélos à assistance électrique

**Question #1**

**Intitulé de la question**

Pensez-vous que la municipalité devrait mettre à disposition des riverains des vélos à assistance électrique ?

**Sélection du type de réponse**

☐ Réponse texte ☒ Réponse à choix multiples

- Oui
- Non
- Ne s'exprime pas

+ -

**Question #2**

**Intitulé de la question**

Si la ville devait offrir une prime à l'achat de vélos à assistance électrique, quel serait pour vous son montant idéal en euros ? Inscrivez "0" si vous pensez que cette prime n'a pas lieu d'être.

**Sélection du type de réponse**

☒ Réponse texte ☐ Réponse à choix multiples

**Question #3**

**Intitulé de la question**

Avez-vous des remarques supplémentaires ?

**Sélection du type de réponse**

☒ Réponse texte ☐ Réponse à choix multiples

Retirer question Ajouter question

Valider le questionnaire

FIGURE 5 – Exemple de page de création questionnaire. Route : `/nouveauQuestionnaire/<idConference>`

## .3 Comptes-rendus des réunions

Au cours de notre projet, nous avons organisé des réunions, souvent pour faire un point sur les travaux terminés et établir de nouvelles tâches. Ces comptes-rendus sont disponibles ci-dessous.

# Projet Pluridisciplinaire groupe E11 -

## Compte rendu n°01

<b>Motif / type de réunion:</b> Mise en place et réflexions pour l'orientation du projet	<b>Lieu:</b> Locaux de TELECOM NANCY
<b>Présents :</b> Ensemble de l'équipe-projet : <ul style="list-style-type: none"><li>• DEMANGEON Matéo,</li><li>• FRACHE Nicolas,</li><li>• VERMEL Louis,</li><li>• WENGER Clément.</li></ul>	<b>Date:</b> 08/11/2021 <b>Heure de début :</b> 13h <b>Durée:</b> 1h

### Ordre du jour

1. Mise en commun du parcours et des compétences de chacun,
2. Réflexion pour le concept de l'application,
3. Organisation du projet.

### Informations échangées

- Chaque membre a partagé son parcours,
- Bilan de nos recherches respectives sur le sujet de la démocratie participative. La grande majorité de l'existant rentre dans une de ces trois catégories :
  - Système de vote,
  - Système de consultation, proposition citoyenne,
  - Application basée autour de l'open data.

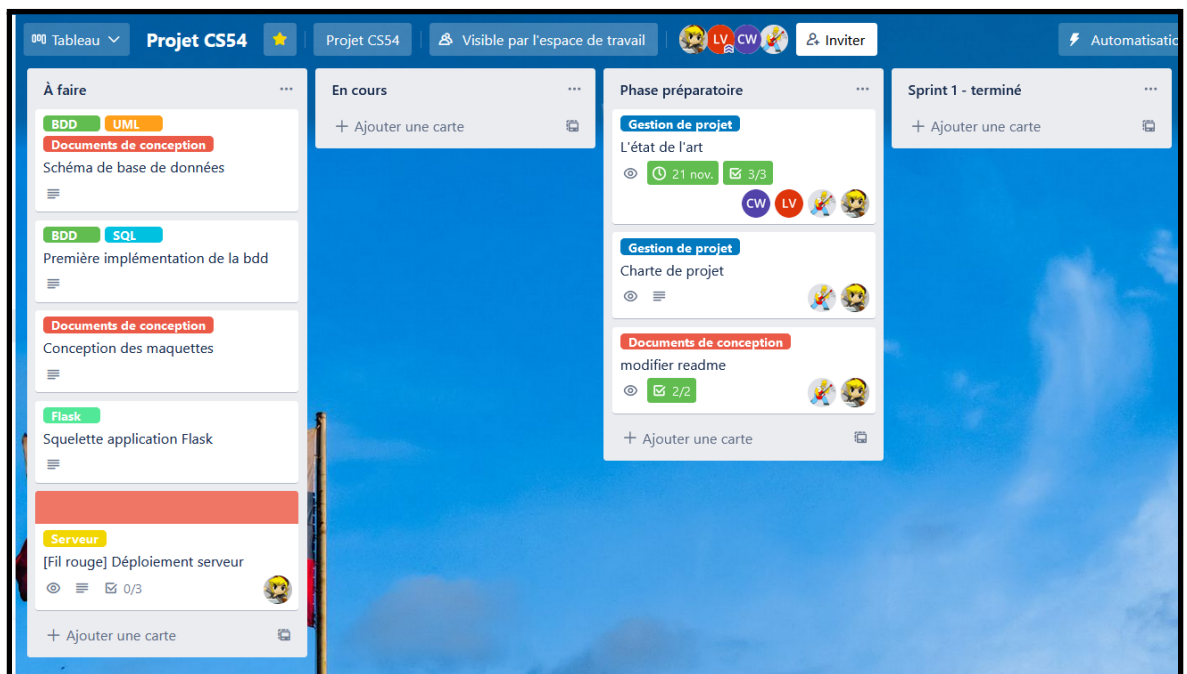
Malheureusement nous ne sommes pas parvenus à trouver un concept qui nous semble suffisamment innovant et intéressant au cours de cette réunion. C'est donc évidemment notre première priorité.

## Décisions

- Utilisation d'un trello (outils de gestion de tâche "ticket") pour s'organiser et avoir du recul sur l'avancement du projet en permanence. [Lien du trello](#).
- Création d'un serveur Discord, un outil qui permet d'échanger aussi bien à l'écrit qu'à l'oral avec différents canaux de communication.
- Sonder nos entourage et continuer de réfléchir afin de pouvoir nous réunir une seconde fois avec des actions plus concrètes.

## Documents additionnels :

1 - Instantané du tableau Trello du projet :



## Projet Pluridisciplinaire groupe E11 - Compte rendu n°02

<b>Motif / type de réunion:</b> Lancement du projet	<b>Lieu:</b> Locaux de TELECOM NANCY
<b>Présents :</b> Ensemble de l'équipe-projet : <ul style="list-style-type: none"><li>• DEMANGEON Matéo,</li><li>• FRACHE Nicolas,</li><li>• VERMEL Louis,</li><li>• WENGER Clément.</li></ul>	<b>Date:</b> 15/11/2021 <b>Heure de début :</b> 18h <b>Durée:</b> 1h

### Ordre du jour

1. Choix final de concept
2. Mise en place des processus de gestion de projet

### Informations échangées

- Partage et explications du concept de conférence des citoyens,
- Nos expériences respectives en gestion de projet pour le choix des outils.

### Décisions

- Axer notre application autour du principe de conférence des citoyens.
- Commencer la rédaction de la charte de projet pour notamment mettre à l'écrit les principales fonctionnalités.
- Utilisation de documents Google partagés pour la rédaction de l'ensemble des documents de gestion de projet et de conception.
- Utilisation régulière d'un tableau Trello pour répertorier les tâches restantes et avoir une vue synthétique de ce sur quoi chacun travail.

### Actions à suivre / Todo list

Description	Responsable	Délai	Livrable	Validé par
Rédaction charte de projet	-	Avant le 29/11/2021	Pdf du document	-
Faire valider le concept par l'équipe enseignante	-	Avant le 29/11/2021	-	Monsieur Festor

# Projet Pluridisciplinaire groupe E11 - Compte rendu n°03

<b>Motif / type de réunion:</b> Validation du concept d'application	<b>Lieu:</b> TELECOM NANCY
<b>Présents:</b> Ensemble de l'équipe-projet : <ul style="list-style-type: none"><li>• DEMANGEON Matéo,</li><li>• FRACHE Nicolas,</li><li>• VERMEL Louis,</li><li>• WENGER Clément,</li></ul> Enseignant : <ul style="list-style-type: none"><li>• FESTOR Olivier.</li></ul>	<b>Date:</b> 29/11/2021 <b>Heure de début :</b> 16h <b>Durée:</b> 20 min

## Ordre du jour

1. Point avec l'équipe pédagogique concernant notre choix de projet.

## Informations échangées

- Partage du concept de notre application et des fonctionnalités métier.
- Choix techniques en découlant.

## Actions à suivre / Todo list

Description	Délai	Livrable
Découpage et planifications des différentes tâches des tâches	05/12/2021	-
Terminer les étapes du 1er sprint	05/12/2021	-Schéma de la base de donnée -Première implémentation de la bdd -Maquettes -Squelette de l'application Flask

# Projet Pluridisciplinaire groupe E11 -

## Compte rendu n°04

<b>Motif / type de réunion:</b> Passage du 1er au 2e sprint	<b>Lieu:</b> Conversation Discord
<b>Présents :</b> Ensemble de l'équipe-projet : <ul style="list-style-type: none"><li>• DEMANGEON Matéo,</li><li>• FRACHE Nicolas,</li><li>• VERMEL Louis,</li><li>• WENGER Clément.</li></ul>	<b>Date:</b> 05/12/2021 <b>Heure de début :</b> 20h <b>Durée:</b> 1h

### Ordre du jour

1. Point sur les premiers éléments terminés
2. Liste des prochains points à traiter

### Informations échangées

- Difficultés rencontrées
- Prochains éléments à aborder

### Décisions

- Utiliser SQLite au lieu de MySQL

### Actions à suivre / Todo list

Description	Délai	Livrable
Terminer les étapes du 2e sprint	Avant le 16/12/2021	Fichiers du code source

## Projet Pluridisciplinaire groupe E11 - Compte rendu n°05

<b>Motif / type de réunion:</b> Passage du 2e au 3e sprint	<b>Lieu:</b> Conversation Discord
<b>Présents :</b> Ensemble de l'équipe-projet : <ul style="list-style-type: none"><li>• DEMANGEON Matéo,</li><li>• FRACHE Nicolas,</li><li>• VERMEL Louis,</li><li>• WENGER Clément.</li></ul>	<b>Date:</b> 16/12/2021 <b>Heure de début :</b> 18h <b>Durée:</b> 1h

### Ordre du jour

1. Est-il judicieux d'utiliser SQLAlchemy
2. Point sur les premiers éléments terminés
3. Liste des prochains points à traiter

### Informations échangées

- Difficultés rencontrées
- Prochains éléments à aborder

### Décisions

- Utilisation de SQLAlchemy pour manipuler la base de données simplement

### Actions à suivre / Todo list

Description	Délai	Livrable
Se familiariser avec l'utilisation de SQLAlchemy (formation rapide <a href="https://courses.prettyprinted.com/courses/1016334/lectures/21156762">https://courses.prettyprinted.com/courses/1016334/lectures/21156762</a> )	Avant le 29/12/2021	-
Se familiariser avec la syntaxe Latex en vue de la rédaction du rapport	Avant le 29/12/2021	-
Entamer l'écriture des tests pytest	Avant le 29/12/2021	-



# Projet Pluridisciplinaire groupe E11 -

## Compte rendu n°06

<b>Motif / type de réunion:</b> Finaliser le programme et la rédaction du rapport	<b>Lieu:</b> Conversation Discord
<b>Présents :</b> Ensemble de l'équipe-projet : <ul style="list-style-type: none"><li>• DEMANGEON Matéo,</li><li>• FRACHE Nicolas,</li><li>• VERMEL Louis,</li><li>• WENGER Clément.</li></ul>	<b>Date:</b> 03/01/2022 <b>Heure de début :</b> 18h30 <b>Durée:</b> 1h

### Ordre du jour

1. Bilan sur les derniers ajustements
2. Point sur la rédaction du rapport

### Informations échangées

- Questions sur le fonctionnement de la syntaxe Latex

### Décisions

- Décisions sur la structure du rapport

### Actions à suivre / Todo list

Description	Délai	Livrable
Finaliser le code (derniers ajouts et vérification)	Avant le 06/01/2022	Fichiers du code source
Terminer le rapport	Avant le 06/01/2022	Fichier du rapport
Ajouter plus de tests	Avant le 06/01/2022	Fichiers du code source