

Langage Python A3 TD8 et TD9 - Le projet

Une plateforme open source pour aider les communes à gérer les besoins (en particuliers de première nécessité) lors d'un confinement total (plutôt couvre-feu).

Analyse de besoins

La phase d'analyse de besoins s'est achevée avec la contribution de tous les groupes, un travail précieux a été fourni.

Ce travail couvre un champs large d'usage, aide alimentaire, aide sanitaire et funéraire, urgence de travaux, hygiène animale, dons, urgence médicale, personnes fragiles isolées, système de livraison, gestion de stocks, gestion de bénévoles, ..

Certains usages couvrent aussi l'aide scolaire et même des activités de divertissement et de bien être.

Certains groupes sont allées jusqu'à ma description de la plateforme et les différents onglets (fonctionnalités) qu'elle regroupe.

Ce travail fait ressortir trois grandes fonctionnalités :

- Gestion des habitants (inscription à la plateforme)
- Gestion des demandes
- Gestion du stocks et équipements de livraisons
- Gestion du personnel
- Gestion de la livraison
- Etude de performance et amélioration

Contraintes : Evaluation et contexte actuel de fin de semestre

Le contexte actuel et les difficulté lié à la fin de semestre, la volonté d'égalité pour l'évaluation de ce module (voire l'équité), nous contraint à réduire nos ambitions.

Nous définissons une base commune "minimaliste" pour l'ensemble des groupes. Nous pouvons aller plus loin sur la base de volontariat. Ce volontariat sera récompensé sous forme de bonus.

Nous nous imposons une contrainte forte, c'est de rester toujours dans le cadre du module langage Python. Nous ne traitons pas des aspects de base de données avancées. Des solutions à base des fichiers csv sont autorisées.

Le projet pour les séances 8 et 9:

Nous nous limitons aux deux premiers volets :

- La gestion des demandeurs

- La gestion des demandes (saisie et analyse, c'est à dire l'usage du citoyen et l'usage de la collectivité)

Ci-dessous la description de ces volets, tout ce qui est marqué par "Pour aller plus loin" est une option, il n'y a pas d'obligation à la développer.

1-La gestion des demandeurs

Il s'agit de l'interface permettant à un citoyen de s'inscrire sur la plateforme, plus précisément d'enregistrer son foyer, un formulaire de base : Nom, email, password, téléphone, adresse, nombre de personnes formant le foyer, leurs nom... Une attention supplémentaire est accordée à l'adresse pour assurer la livraison mais surtout dans un premier temps des visualisations spatiales.

Pour aller plus loin : Un seul compte par foyer

Toutes ces données peuvent être enregistrées dans une BD, un fichier csv, ou n'importe quel fichier en utilisant json ou pickle. (il ne faut pas s'ajouter des difficultés sql pour ceux qui en ont)

2-La gestion des demandes

Interface Citoyen: Une fois le citoyen inscrit, il peut se connecter et saisir sa demande

Il s'agit d'un formulaire regroupant une vingtaine de produits de premières nécessités, ne laisser pas le choix libre dans le formulaire, c'est à dire le citoyen doit choisir à partir d'une liste prédéfinie de produits.

Pour les besoins particuliers (allergie, demande de santé spécifique, ...) on mentionnera la présence d'un autre formulaire avec son lien mais sans le mettre en place.

Interface Collectivité : Cette interface doit assurer à la collectivité une meilleure visibilité des demandes des citoyens par le biais de plusieurs visualisations pertinentes.

Laissez s'exprimer votre créativité : le/les produit/s les plus demander. Repartition des demandes selon les quartiers de la villes (geomap), fréquences des demandes ...

Pour aller plus loin : sur la base des demandes d'un jours J, calculer le panier essentiel à préparer pour le j+1 (Apriori)

Approches techniques, modules, framework, ...

Vous êtes libres de faire le choix qui vous convient, aucun module n'est imposé. Vous pouvez utiliser le framework qui vous convient (django, flask, cherrypy, web2py, pandas, folium, plotly, ...).

N'attendez pas des solutions de la part de vos chargés de TD, c'est à vous de trouver ce qui vous convient le mieux. C'est plus à vous d'expliquer vos choix et de les justifier auprès de vos profs.

Il s'agit de Python, gratuit, open source, plein de ressources, gardez ce même esprit, vous avez le droit de tout utiliser à condition qu'il soit open source et gratuit (aucun usage des modules ou librairies commerciales n'est autorisées, même si vous ne la payez pas)

Version minime avec pénalité : Vous avez le droit de ne pas concevoir votre projet sous forme web mais par une standalone application, dans ce cas une pénalité de 5 points est appliquée (note sur 15 au lieu de 20)

Groupes de travail:

Ce travail est à rendre par groupe de 5 ou 6 étudiants une seule exception est accordée pour faire une équipe de 7 étudiants (qd il s'agit de 31 étudiants, 5 groupe de 6 étudiants ca fait 30, du coup j'accepte un groupe de 7, même si je préfère 5 groupes de 5 étudiants et le 6 ième de 6 étudiants)

Deux Deadlines:

Une première date critique est la séance 9, il faut présenter l'état d'avancement de vos travaux, cette rencontre sera notée (il s'agit d'une note préparatoire pour la note finale)

Deadline Finale: 17 mai à 23h50, vous déposez sur DVO votre solution avec un rapport décrivant vos travaux, ce qui est fait et fonctionnel, ce qui ne l'est pas, des captures d'écran seront fort utiles (un bonus à ceux qui livre une version en ligne).

Faites attentions à inclure, les modules spécifiques que vous avez employés, et à lister tous les modules utilisés. Un readme d'installation est indispensable.