

Projet non alternant

Equipe

- **Chef de projet** : Yahaya
- **Devops** : Vitalii
- **Front end** : Vitalii
- **Back end** : Ambroise & Yahaya

Nom du projet

- ☐ Exploreasy
- ☐ Tourista
- ☐ Turisimu
- ☐ Voluptuaria

Contexte du projet

Application mobile et web, dans le but de centraliser les différents services d'aide au tourisme au sein d'une application.

Projets existants

- Tripadvisor
- Google Maps
- ...

Besoins

- Centraliser les services existants d'aide au tourisme
- Proposer de la gestion de l'expérience du touriste

Résultats attendus :

- Après les deux premières semaines (**Bloc 1**) :
 - maquette graphique de l'application
 - base de données
 - documentation de l'Api
- Le 25/10/24 (**Bloc 2**) :
 - liaison avec les Apis de réseaux sociaux
 - récupération de la météo et de ses risques d'une zone
 - recherche de spots dans un lieu et risques associés
- Le 23/11/24 (**Version 1 minimale**)
 - front end disponible
 - login implémenté
 - recommandation simple
 - parcours de tourisme simple

-
- L'avant-dernière semaine de projet (**Version 1 acceptable**)

Contraintes

Dépendances aux Apis

Apis payantes au bout d'un quota de requêtes.

- Instagram
- Google Maps
- OpenWeatherMap)

Limitation au niveau de certains réseaux sociaux

Certains réseaux sociaux ne proposent pas l'accès aux données considérées comme sensibles.

Ressources nécessaires

- Données provenant des réseaux sociaux
- Données provenant des Apis

Fonctionnalités du projet

- Certaines fonctionnalités sont disponibles sans création d'un compte
- Rechercher rapidement dans un endroit sous forme de (cartes / liste), les lieux de :
 - repos
 - visites
 - pour manger
 - loisirs
 - ...
- Voir les zones à risque basé sur la météo
- Evaluer / Voir les évaluations d'un lieu
- Recommandation de lieu à visiter basé sur :
 - Avis Google
 - Note de lieu
 - Distance
 - Type de lieu fréquemment recherché et / ou visité
 - Profil d'intérêt construit à partir
 - d'une similarité entre utilisateurs
 - distance
- Création de parcours de visite basé sur :
 - date de début et de fin de la visite
 - budget
 - même critères que la recommandation

Planification

Tâches	Membre(s)	Date de début	Date de fin
Design graphique de l'application	Vitalii	17/09/24	20/09/24
Définition de la base de données	Yahaya	17/09/24	27/09/24
Définition de l'api	Ambroise	17/09/24	20/09/24
Mise en place des différents environnements de développement	Vitalii	23/09/24	24/09/24
Définition des patterns front et back end	Ambroise & Yahaya	25/09/24	27/09/24
Intégration de la maquette figma	Vitalii	21/10/24	23/11/24
Intégration des appels d'Api au back end	Vitalii	21/10/24	23/11/24
Implémentation des classes de gestion de procédure	Yahaya	21/10/24	21/10/24
Implémentation des méthodes de vérification de risques	Ambroise	21/10/24	21/10/24
Implémentation de la recherche rapide de lieu en intégrant les risques météo	Yahaya	21/10/24	22/10/24
Implémentation des classes et interfaces de description de réseaux sociaux	Yahaya & Ambroise	22/10/24	22/10/24
Implémentation des classes de liaison aux réseaux sociaux	Yahaya & Ambroise	23/10/24	25/10/24
Implémentation du login utilisateur et liaison aux réseaux sociaux	Yahaya & Ambroise	18/11/24	19/11/24
Implémentation des classes (abstraite / interface) de gestion de recommandation	Yahaya	19/11/24	19/11/24
Implémenter la recommandation simple basée sur les notes et lieux similaires	Ambroise	19/11/24	21/11/24
Implémenter la construction de parcours simple basés sur les critères définis ainsi que les notes et lieux similaires	Yahaya	19/11/24	22/11/24
Implémentation des classes de récupération des lieux liés à un réseau social (construction du profil)	Ambroise	25/11/24	26/11/24
Implémenter la procédure de détection de similarité entre utilisateur (construction du profil)	Ambroise & Yahaya	26/11/24	--/--/2-
Implémenter la recommandation basée sur la similarité utilisateur	Ambroise & Yahaya	26/11/24	--/--/2-
Implémenter la construction de parcours basée sur la similarité utilisateur	Ambroise & Yahaya	26/11/24	--/--/2-

Génération des documentations (mentions, conditions ...)

Nota bene :

Nous nous laissons une semaine de marge pour la rédaction et pour les potentielles erreurs.

Technologies

- **Design graphique** : Figma
- **Front end** : Flutter Le choix de flutter est dû à :
 - Sa facilité d'apprentissage
 - Le développeur front-end connaît déjà la technologie
 - Multiplateforme et OS à la base d'un code
- **Back end** :
 - Organisé sous forme d'API REST
 - Adonis JS
 - Apis majeures utilisées :
 - [Instagram Api Graph](#) pour l'accès aux données instagram (meta données des publications)
 - [OpenWeatherMap](#) pour l'interaction avec la météo
 - [Google Map Places Api](#) pour les recherches de lieux
- **Base de données** :
 - MongoDB Le choix du type de base de données se base sur la quantité potentielle des données et la non nécessité d'un model relationnel. Le NoSql et les outils associés sont donc plus appropriés.

Outils de gestion / communication

Les liens ne sont accessibles qu'aux encadrants du projet.

- **Versionnage du projet** : [Github](#)
- **Gestion des tâches** : [Github projet](#)
- **Communication dans le groupe** : [Teams](#)