

---

# FICHE PROJET

---

# SOMMAIRE

Description du projet

Présentation technique

L'équipe et de son organisation



# PITCH DU PROJET

**Saviez-vous que depuis la crise sanitaire, 65 % des jeunes ont vu leur éducation se dégrader ?**

**Comment venir en aide à ces étudiants en difficultés dans leur formation ?**

## **Solutions existantes :**

- Sites de cours en ligne (Google Classroom, OpenClassrooms, Khan Academy,...)
- Sites de questions-réponses (StackExchange, Quora...)

## **Inconvénients :**

- Les informations ne sont pas toujours fiables
- Le temps de réponse peut être long
- Les communautés ne sont pas toujours accueillantes envers les débutants

# VALEUR AJOUTÉE PAR RAPPORT À NOS CONCURRENTS

## FLÉXIBILITÉ HORAIRE

L'étudiant répartit ses heures mensuelles à l'avance en fonction des disponibilités des professeurs et de son emploi du temps. Plus d'horaires imposés, l'étudiant peut plus facilement gérer ses études et son travail.

## QUALITÉ PÉDAGOGIQUE

Tous les professeurs inscrits reçoivent une note qui dépend : du résultat à des tests sur la/les technologie(s) pour lesquelles ils veulent être référencés + leur disponibilité sur le site.

## MATCHMAKING INTELLIGENT

Mise en relation avec des professeurs en fonction des paramètres de son choix : niveau de qualification dans une technologie (langage, CMS, Framework, ...), disponibilité sur un créneau horaire, curriculum du professeur (matchmaking projets du professeur vs mots clés entrés par l'étudiant).



# Cible



Personnes concernées : Un jeune étudiant travaillant à distance.  
Une personne qui cherche à se former de manière autodidacte (ex : reconversion...)

# Présentation technique

## FRONT-END



## BACK-END

**django**



## POURQUOI REACT ?

ReactJS produit du code propre c'est-à-dire simple à lire. Sa lecture permet de déterminer immédiatement quelles sont les fonctionnalités de l'application. Ce qui est essentiel pour la maintenance et l'expansion du projet dans le temps.

## POURQUOI DJANGO ?

### Architecture MVC :

- Un développement plus clair et efficace pour les grands projets grâce à la séparation des données de la vue (affichage) et du contrôleur (fonctions de commande).
- Meilleure maintenance car la structure standardisée permet à l'application d'être reprise par d'autres développeurs.

### Écrit en Python :

- Implémentation plus simple pour la programmation au cas où il faut inclure des algorithmes.

# L'ÉQUIPE ET SON ORGANISATION



**BALTHAZAR  
PERRIN**

Développeur Front-end



**FAHD  
EL ABD**

Développeur Python



**FLAVIEN  
MEMAIN**

Développeur Back-end



**SYOU-PHENG  
DO**

Développeur Back-end



**ZARUHI  
GYULUMYAN**

DevOps



**YASSER  
DAHY**

Développeur Back-end



**MITHULAN  
MANORANCHITHAN**

Développeur Front-end

**MERCI DE VOTRE LECTURE**