

SalesTraction (POC)

Maxime ALBERT et Houssein GHANNOUM

User story : Étudiant



Inscription et profil

En tant qu'étudiant, je veux **m'enregistrer** via **email**, **Google** ou **LinkedIn** afin de créer un compte rapidement.

En tant qu'étudiant, je veux **accepter les conditions générales** afin de pouvoir accéder à la plateforme.

En tant qu'étudiant, je veux **créer mon profil** avec des infos comme **l'âge, l'université, LinkedIn, langues**, etc. pour mieux me présenter aux startups.

En tant qu'étudiant, je veux **modifier mon profil** à tout moment afin de le garder à jour.



Recherche et candidature

En tant qu'étudiant, je veux **consulter les offres** avec des **filtres** (secteur, lieu, commission, mode de travail) afin de trouver les missions qui me correspondent.

En tant qu'étudiant, je veux **consulter les détails d'une offre** (produit, commission, documents) pour décider si je veux postuler.

En tant qu'étudiant, je veux **postuler à une offre** avec mon **CV** et une **lettre de motivation** pour montrer mon intérêt.

En tant qu'étudiant, je veux **consulter les retours** à mes candidatures pour savoir où j'en suis.



Communication et suivi

En tant qu'étudiant, je veux **échanger des messages** avec les startups pour poser des questions ou discuter des missions.

En tant qu'étudiant, je veux **envoyer des messages d'expérience** ou de **feedback** à la plateforme.

En tant qu'étudiant, je veux **consulter mon historique de missions et de commissions** pour suivre mes gains.

En tant qu'étudiant, je veux **contacter les admins** pour signaler un problème ou poser une question.

User story : Startup



Création et validation du compte

En tant que startup, je veux **m'enregistrer** via email ou LinkedIn pour créer mon compte rapidement.

En tant que startup, je veux **entrer mon SIRET** et le faire **vérifier automatiquement** ou **par un admin** pour valider mon profil.

En tant que startup, je veux **créer mon profil** avec une **description** et accepter les **CGU** pour accéder à la plateforme.



Gestion des offres

En tant que startup, je veux me **connecter** à mon compte pour **accéder à mes offres**.

En tant que startup, je veux **créer une nouvelle offre** avec **formulaire, documents, commission** et **description** pour proposer une mission.

En tant que startup, je veux **consulter mes offres** en cours et les **modifier** si besoin.

En tant que startup, je veux **suivre les candidatures** à mes offres et **consulter le profil des étudiants** intéressés.



Interaction et suivi

En tant que startup, je veux **accepter une candidature** ou la **refuser** en envoyant un retour à l'étudiant.

En tant que startup, je veux **consulter** une **liste d'étudiants ouverts** aux missions pour les contacter.

En tant que startup, je veux **contacter la plateforme** pour demander une mise en avant ou du support.

En tant que startup, je veux **échanger des messages** avec les **étudiants** via une messagerie interne.

En tant que startup, je veux **voir l'historique des missions** et des **commissions générées** par mes offres.

User story : Admin

Modération

En tant qu'admin, je veux **modérer** les **discussions** ou **contenus signalés** pour garantir la bonne conduite sur la plateforme.

En tant qu'admin, je veux **voir les signalements** et y **répondre** rapidement pour résoudre les problèmes.

Analyse et support

En tant qu'admin, je veux **analyser les données** de la plateforme (offres, commissions, utilisateurs actifs) pour améliorer le service.

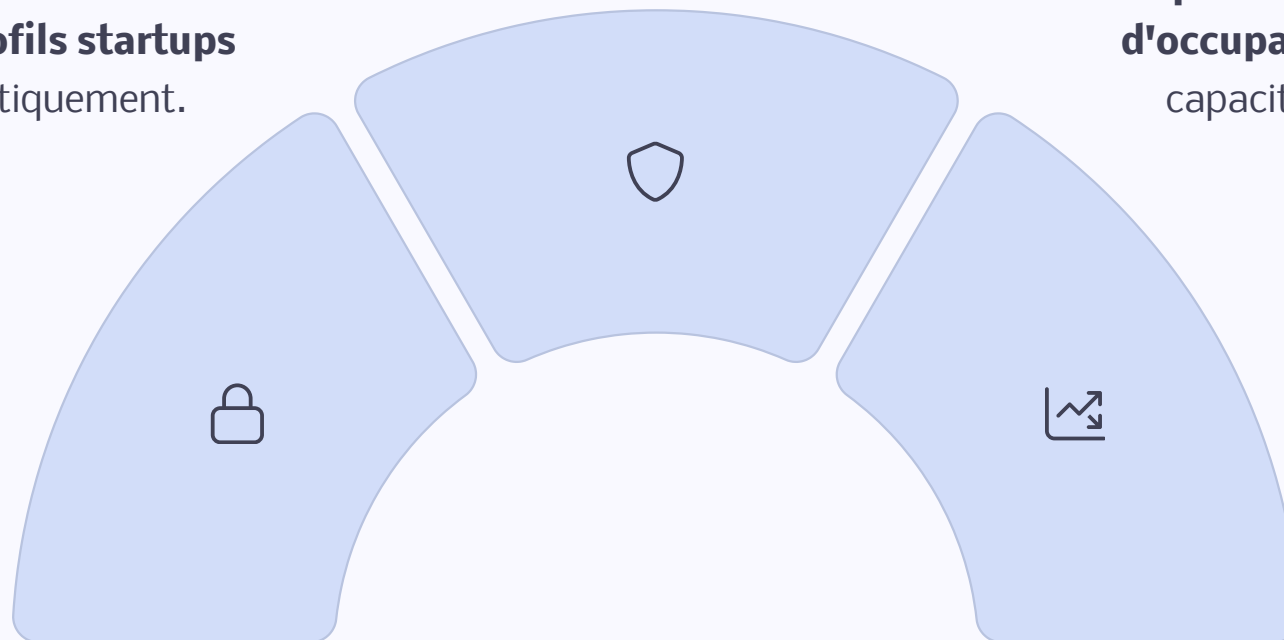
En tant qu'admin, je veux **accéder aux messages** d'aide et **répondre** aux utilisateurs pour les accompagner.

En tant qu'admin, je veux **consulter l'espace de stockage** et le **taux d'occupation** pour surveiller la capacité de la plateforme.

Accès et validation

En tant qu'admin, je veux me **connecter** pour accéder au back-office de gestion.

En tant qu'admin, je veux **valider manuellement** les **profils startups** non validés automatiquement.





Choix technique

Composant	Local (POC)	AWS Prod
Frontend	React local	AWS Amplify / S3 + CloudFront
Backend	Express local	AWS Elastic Beanstalk / Lambda
Base de données (BDD)	MariaDB local	Amazon RDS (MariaDB)
Authentification	MariaDB local	AWS Cognito
CI/CD	GitHub Actions	GitHub Actions + AWS CLI
Fichiers	Local	Amazon S3
Maquettes / UX	Figma (local/file)	Figma (réutilisé, feedback + itération)

Explication des choix

Frontend

Local (POC) : React → Framework moderne, modulaire et connu de l'équipe, facilitant le développement rapide d'interfaces interactives.

AWS Prod : AWS Amplify ou S3 + CloudFront → Déploiement connecté à GitHub, SSL auto, cache géré. En alternative, S3 + CloudFront pour de meilleures performances.

Backend

Local (POC) : Express.js → Création rapide d'API REST, avec un framework familier, permettant des itérations rapides.

AWS Prod : Elastic Beanstalk ou Lambda → Déploiement managé sans gestion serveur avec Beanstalk. Possibilité de migrer vers Lambda pour du serverless scalable.

Base de Données (BDD)

Local (POC) : MariaDB → Moteur SQL open-source compatible MySQL, bien connu de l'équipe.

AWS Prod : Amazon RDS (MariaDB) → Base managée avec sauvegardes auto, haute dispo, mise à l'échelle simple.

Explication des choix (suite)

Authentification

Local (POC) : Gestion des utilisateurs via la base de données locale.

AWS Prod : AWS Cognito → Gestion des utilisateurs, connexions, rôles, tokens, et sécurisation des accès.

CI/CD

Local (POC) : GitHub Actions → Automatisation des tests, linting, build à chaque push.

AWS Prod : GitHub Actions + AWS CLI → Déploiement automatique via workflows, intégration CLI AWS.

Gestion des fichiers

Local (POC) : Stockage local → Simplicité durant le dev sans config externe.

AWS Prod : Amazon S3 → Stockage sécurisé, durable et scalable pour les fichiers utilisateurs.

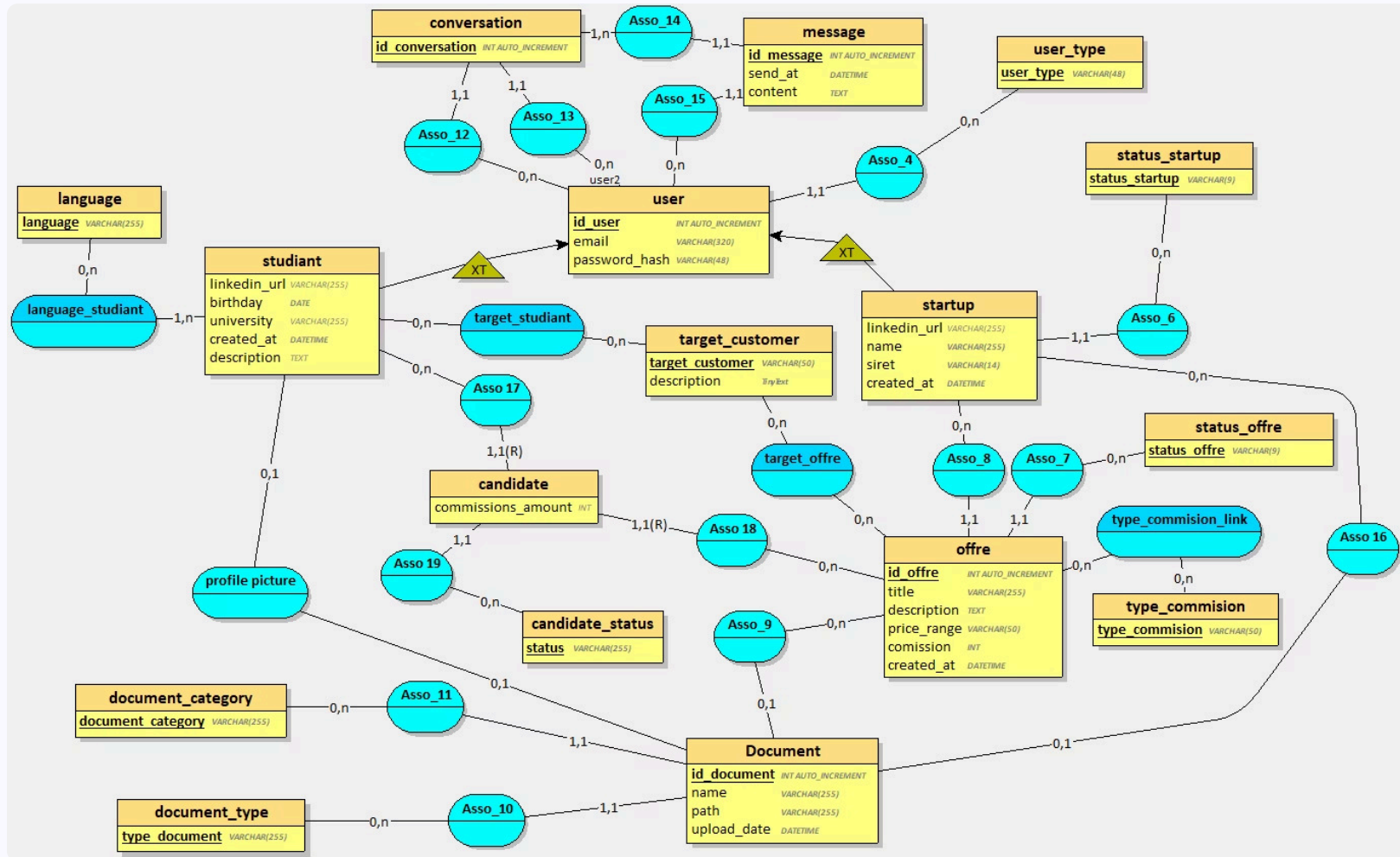
Maquettes / UX

Local (POC) : Figma (fichier local) → Création initiale des maquettes localement ou en partage simple.

AWS Prod : Figma (réutilisé) → Réutilisation des maquettes pour itérations UX avec retour utilisateur.

Modèle Conceptuel de Données (MCD)

Le Modèle Conceptuel de Données représente la structure des données et leurs relations pour notre projet. Il permet de visualiser les entités principales, leurs attributs, et les liens entre elles afin d'assurer une base solide pour la base de données.



GitHub Public & Arborescence du Projet

Accès au code

Le projet est disponible publiquement sur GitHub à l'adresse suivante :

<https://github.com/Landormi/SalesTraction>

Structure principale

- **Back/** : Backend Node.js avec API REST et base de données
- **Front/** : Frontend React, interface utilisateur
- **bdd/** : Scripts SQL et Modèle Conceptuel de Données (MCD)
- **README.md** : Documentation globale du projet

Documentation intégrée

Chaque dossier contient un README pour guider le développement local et la maintenance.