


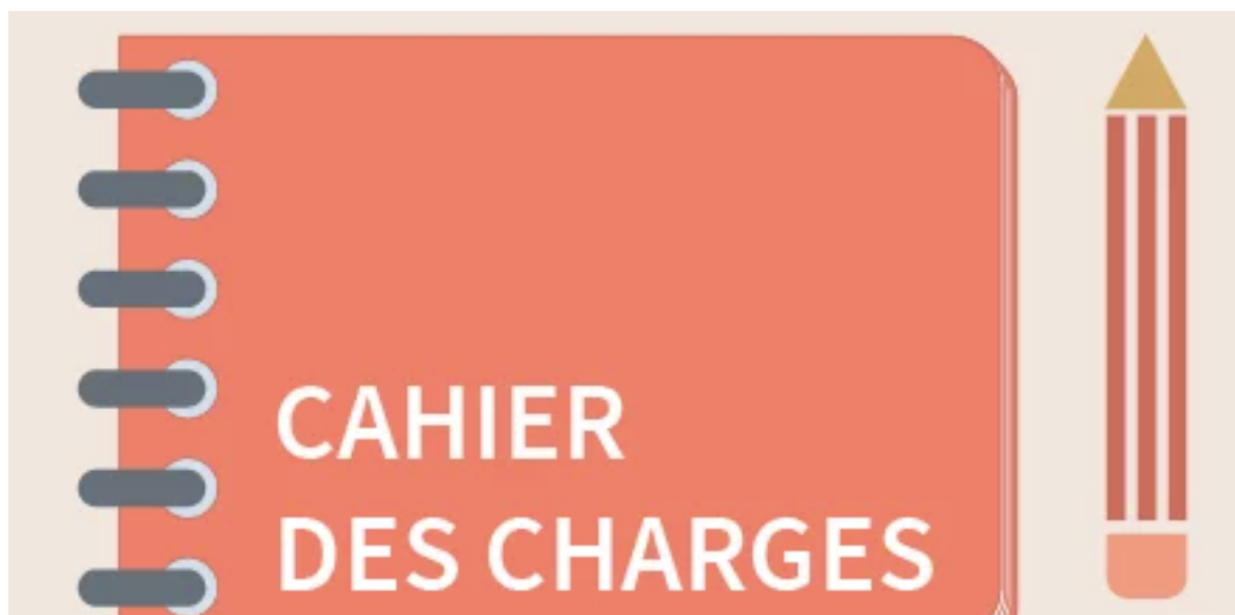
# Cahier des charges fonctionnel



Votre plateforme scolaire de référence. Simplifiez votre expérience éducative avec INTRA-TAMC, l'outil conçu spécialement pour les étudiants et enseignants.

Connectez-vous rapidement et en toute sécurité

 Sign in with Google



## Introduction

### A. Contexte du projet

Le projet Intranet vise à fournir une plateforme centralisée pour les étudiants et l'équipe pédagogique d'un centre de formation. Cet outil a pour but de faciliter la gestion des projets, des formations, ainsi que le suivi des compétences et des performances des étudiants.

Définition des besoins et objectifs

- Offrir aux étudiants une plateforme pour gérer leurs projets et suivre leurs compétences.
- Fournir à l'équipe pédagogique un outil pour gérer les promotions, étudiants, formations, projets, et avoir des statistiques pertinentes.
- Assurer une évolutivité pour répondre aux besoins futurs, y compris la possibilité de transformation en SaaS.

Profil utilisateur·trice

- Étudiants
- Équipe pédagogique (responsables, accompagnateurs, intervenants externes)

### B. Description fonctionnelle du site

Arborescence

- Page d'accueil
- Projets
- Compétences
- Profil étudiant
- Gestion des promotions
- Gestion des étudiants
- Gestion des unités de formation
- Gestion des projets pédagogiques
- Statistiques

### C. Objectifs du document

Ce document vise à définir clairement les exigences fonctionnelles, à identifier les utilisateurs cibles, à présenter les scénarios d'utilisation, et à offrir une vision claire des interactions et des fonctionnalités de l'application.

### D. Portée du projet

La refonte de l'intranet s'étalera probablement sur plusieurs mois ou années, avec plusieurs versions prévues. Cela comprend la création de fonctionnalités dédiées pour les étudiants et l'équipe pédagogique.

### E. Public cible

Étudiants de l'organisme de formation

Équipe pédagogique, incluant les responsables, les accompagnateurs et les intervenants externes

## F. Technologies

### Front-end

- React.js
- Redux
- Bootstrap

### Back-end

- Node.js
- Express.js
- MariaDB

### Navigateurs compatibles

- Google Chrome
- Mozilla Firefox
- Safari
- Microsoft Edge

L'intranet sera développé en utilisant des technologies modernes, bien adaptées à la création d'applications web évolutives. Les détails technologiques seront définis ultérieurement.

## 2. L'équipe

### L'Équipe de direction de INTRA-TAMC Agency

Rencontrez notre équipe de direction



**LEE MILLER**  
Fondatrice



**Chaima Aicheche**  
Chef de Projet

## L'Équipe Scrum de INTRA-TAMC Agency

Rencontrez notre équipe Scrum



**Noa Andrieux**  
product owner



**Antoine Auclair**  
Lead Développeur



**Meriem Barka**  
Scrum Master

## L'Équipe de Développeur de INTRA-TAMC Agency

Rencontrez notre équipe de développeur



**Alizéa Massé**  
Développeuse



**Thomas Garcia**  
Développeur



**Camilla Martine**  
Conceptrice



**Tchèssi Pre**  
Concepteur

#### 4. Les méthodes envisageables

Pourquoi la méthode Scrum ?

Nous avons préféré utiliser Scrum pour gérer notre projet d'intranet, car il combine les avantages dont nous avons besoin. Notre équipe a pleinement adopté les quatre piliers de Scrum : le sprint planning, le daily scrum, le sprint review et la rétrospective.

La taille réduite de notre équipe a rendu la flexibilité et la collaboration rapides de Scrum adaptées à notre situation. De plus, cette méthode favorise l'amélioration continue, ce qui est crucial étant donné que notre projet d'intranet évolue constamment.

Scrum nous aide à simplifier les défis en découpant le projet en tâches gérables et en mettant l'accent sur ce qui est le plus important. Étant donné que notre intranet est amené à évoluer, Scrum s'adapte parfaitement à ces changements.

En utilisant Scrum, nous pouvons fournir régulièrement des versions fonctionnelles de l'intranet, ce qui facilite son adoption par nos utilisateurs.

La clarté des rôles au sein de l'équipe a également été importante, car elle favorise la responsabilisation et une meilleure répartition des tâches. Scrum permet une gestion fluide des tâches, tout en fournissant une structure claire pour la planification et la gestion des itérations.

Les autres méthodes

"Pourquoi ne les avons-nous pas choisies ?"

Cascade (Waterfall) : Nous avons écarté la méthode en cascade en raison de sa nature séquentielle, qui exige que chaque phase soit achevée avant de passer à la suivante. Dans notre projet où les besoins et les exigences évoluent fréquemment, une approche plus itérative et flexible comme Scrum est nécessaire pour assurer une livraison plus rapide et une adaptation plus aisée aux changements.

Méthode en V : Nous avons choisi de ne pas utiliser le modèle en V car il est plus adapté aux projets où les exigences sont clairement définies dès le début. Dans notre cas, la nature évolutive du projet nécessite une approche plus flexible qui peut s'adapter aux changements rapides et aux retours des parties prenantes.

Kanban pur : Bien que Kanban soit efficace pour la gestion des flux de travail, il peut parfois manquer de la structure et de la planification requises pour les projets plus complexes. Comme notre projet nécessite un équilibre entre une planification structurée et une gestion flexible des tâches, Scrum s'est avéré être un choix plus adapté, car il intègre des éléments des deux méthodologies.

#### 4. Besoins utilisateurs

##### A. User stories

---

##### Côté étudiants

---

##### Accueil

En tant que...	Je veux pouvoir...	Afin de...	Effort	Sprint
Étudiant	Se connecter avec mon compte Google La Plateforme	Accéder à l'intranet	3	1
Étudiant	M'inscrire au projet	Participer au projet	4	1
Étudiant	Rendre un projet	Soumettre mon travail	4	2
Étudiant	Consulter les projets disponibles	Voir les projets que je peux réaliser	3	1
Étudiant	Renseigner mes informations personnelles	Mettre à jour mon profil	2	1
Étudiant	Consulter mes compétences	Visualiser mes progrès	3	2

---

##### Côté équipe pédagogique

---

## Gestion des promotions

En tant que...	Je veux pouvoir...	Afin de...	Effort	Sprint
Responsable	Ajouter une promotion	Intégrer une nouvelle classe	5	1
Responsable	Modifier une promotion	Mettre à jour les informations	4	2
Responsable	Récupérer une promotion	Voir les détails d'une classe	3	1
Responsable	Supprimer une promotion	Retirer une classe terminée	5	2

## Gestion des étudiants

En tant que...	Je veux pouvoir...	Afin de...	Effort	Sprint
Responsable	Ajouter un étudiant	Intégrer un nouveau membre	4	1
Responsable	Modifier un étudiant	Mettre à jour les informations	4	2

Responsable	Récupérer un étudiant	Voir le profil d'un étudiant	3	1
Responsable	Supprimer un étudiant	Retirer un étudiant sortant	5	2

## Gestion des projets

En tant que...	Je veux pouvoir...	Afin de...	Effort	Sprint
Responsable	Ajouter un projet	Proposer un nouveau travail	5	1
Responsable	Modifier un projet	Actualiser les détails	4	2
Responsable	Récupérer un projet	Consulter un projet	3	1
Responsable	Supprimer un projet	Retirer un projet terminé	5	2

## Gestion des unités de formation

En tant que...	Je veux pouvoir...	Afin de...	Effort	Sprint
----------------	--------------------	------------	--------	--------



Responsable	Ajouter une unité de formation	Intégrer une nouvelle matière	5	2
Responsable	Modifier une unité de formation	Mettre à jour les informations	4	3
Responsable	Récupérer une unité de formation	Voir les détails d'une matière	3	2
Responsable	Supprimer une unité de formation	Retirer une matière obsolète	5	3

#### Suivi et validation des projets

En tant que...	Je veux pouvoir...	Afin de...	Effort	Sprint
Accompagnateur	Créer et valider des groupes pour les projets	Organiser le travail	4	2
Responsable	Suivre les projets en cours et rendus	Assurer le bon déroulement	3	2
Responsable	Suivre les temps de log des étudiants	Contrôler l'activité	4	3

## Statistiques

En tant que...	Je veux pouvoir...	Afin de...	Effort	Sprint
Responsable	Avoir des statistiques sur les promotions	Mesurer les performances	5	3

## B. Personas

Étudiant : Personne inscrite dans l'organisme pour suivre une formation.

Responsable : Membre de l'équipe pédagogique gérant les promotions et les étudiants.

Accompagnateur : Membre de l'équipe pédagogique qui intervient auprès des étudiants sans gérer les promotions.

Intervenant Externe : Expert externe intervenant pour des sessions ou des projets spécifiques.

## C. Liste des routes

### Front-end

- /accueil
- /projets
- /compétences
- /profil
- /gestion-promotions
- /gestion-etudiants
- /gestion-formations
- /gestion-projets
- /statistiques

### Back-end

- /api/projets
- /api/compétences
- /api/étudiants
- /api/promotions
- /api/formations
- /api/stats

## D. Scénarios d'utilisation

Un étudiant se connecte via Google La Plateforme et s'inscrit à un projet.

Un responsable ajoute une nouvelle promotion dans l'intranet.

Un accompagnateur valide les groupes de projets.

## E. Exigences fonctionnelles

Authentification via Google La Plateforme pour les étudiants

Fonctionnalités CRUD (Créer, Lire, Mettre à jour, Supprimer) pour les promotions, étudiants, unités de formation et projets pour l'équipe pédagogique

Suivi des projets en cours et rendus

Statistiques sur les promotions

## 5. Spécifications fonctionnelles

### A. Vue d'ensemble de l'app

L'application comprend deux sections principales : une pour les étudiants et une autre pour l'équipe pédagogique. Chaque section aura des sous-sections dédiées aux différentes fonctionnalités.

### B. Détail des écrans

Écran de connexion : Avec intégration Google La Plateforme

Dashboard Étudiant : Vue d'ensemble des projets, compétences, etc.

Dashboard Équipe Pédagogique : Vue d'ensemble des promotions, étudiants, statistiques, etc.

### C. Interactions utilisateur

Les étudiants peuvent s'inscrire et rendre des projets

L'équipe pédagogique peut ajouter, modifier, ou supprimer des informations

### D. Flux de navigation

Les utilisateurs se connectent et sont dirigés vers leur dashboard respectif, où ils peuvent naviguer vers les différentes fonctionnalités.

## E. Fonctionnalités

### MVP - Minimum Viable Product

- Connexion via le compte Google "La Plateforme" pour les étudiants.
- Inscription au projet pour les étudiants.
- Rendre un projet.
- Consultation des projets disponibles.
- Renseignement des informations personnelles par les étudiants.
- Ajout/Modification/Suppression d'une promotion par l'équipe pédagogique.
- Ajout/Modification/Suppression d'un étudiant par l'équipe pédagogique.
- Ajout/Modification/Suppression d'une unité de formation par l'équipe pédagogique.
- Suivi des projets en cours et rendus par l'équipe pédagogique.

## F. Contraintes et conception

L'application doit être sécurisée, en particulier les informations personnelles des étudiants. Elle doit être évolutive pour permettre des mises à jour futures et éventuellement la transformation en un produit SaaS.

L'interface doit être intuitive et user-friendly pour tous les utilisateurs.

## 6. Base de données

### A. Méthode Merise

Schéma Conceptuel de la Base de Données (MCD) :

Utilisateur

C'est l'entité centrale du MCD.

Elle est liée à plusieurs autres entités par différentes relations.

Attributs : CodeUtilisateur, Nom, Prénom, CodeRole, CodePromotion, CompteGoogleLaPlateforme.

Un utilisateur peut avoir plusieurs compétences, logs de temps, et plusieurs liens externes.

Compétence

Représente les compétences qu'un utilisateur peut posséder.

Attributs : CodeCompétence, Nom, Niveau.

La relation POSSEDE relie un utilisateur à une compétence, indiquant qu'un utilisateur peut avoir plusieurs compétences.

LogTemps

Représente les logs de temps d'un utilisateur.

Attributs : CodeLogTemps, PremierLog, DernierLog, Temps, CodeUtilisateur.

La relation ENREGISTRE montre qu'un utilisateur peut avoir plusieurs logs de temps.

Permission

Attributs : CodePermission, Nom.

Représente les permissions qu'un rôle peut avoir.

### Rôle

Représente les différents rôles qu'un utilisateur peut avoir (par exemple, responsable, accompagnateur, intervenant externe).

Attributs : CodeRole, Nom.

La relation POSSEDE\_UN\_ROLE indique qu'un utilisateur peut avoir un rôle.

### LiensExternes

Contient des liens vers des profils ou des plateformes externes liés à un utilisateur.

Attributs : CodeLiensExternes, Github, LinkedIn, Plesk, PortFolio, Cv, CodeUtilisateur.

La relation A\_DES\_LIENS lie un utilisateur à ses liens externes.

### Promotion

Représente les différentes promotions d'étudiants.

Attributs : CodePromotion, Nom, Type, DateDebut, DateFin.

La relation APPARTIENT\_A montre qu'un utilisateur peut appartenir à une promotion.

La relation GERE peut indiquer que certains utilisateurs (par exemple, les responsables) gèrent certaines promotions.

### Projet

Représente les projets sur lesquels les étudiants peuvent travailler.

Attributs : CodeProjet, Nom, Description, Pdf, DateDebut, DateFin, CodeUnite.

La relation ASSOCIE\_A lie un projet à un groupe.

La relation CATEGORISE\_DANS lie un projet à une unité de formation.

### Groupe

Représente les différents groupes d'étudiants travaillant sur un projet.

Attributs : CodeGroupe, Nom, CodeProjet.

La relation COMPOSE\_DE lie un groupe à plusieurs utilisateurs.

### Unité

Représente les différentes unités de formation.

Attributs : CodeUnite, Nom, Description.

Pas directement relié à d'autres entités par une relation dans ce schéma, mais connecté à Projet via la relation CATEGORISE\_DANS.

Globalement, ce MCD donne un aperçu des différents composants du système, de leurs attributs et de leurs relations mutuelles. Il constitue la base pour la conception des bases de données et la mise en œuvre du système.

Modèle Logique de Données (MLD) :

- Le MLD représente la structure de la base de données sous forme de tables et de colonnes.
- Les tables principales incluent "Utilisateur", "Compétence", "LogTemps", "Permission", "Rôle", "LiensExternes", "Promotion", "Projet", "Groupe", et "Unité".
- Des clés primaires (PK) sont définies pour chaque table pour identifier de manière unique chaque enregistrement.
- Les clés étrangères (FK) sont utilisées pour établir des relations entre les tables principales.
- Des tables associatives sont utilisées pour gérer les relations many-to-many, comme "Utilisateur\_Competence" pour les compétences associées aux utilisateurs.

Le schéma conceptuel de la base de données décrit les entités, attributs et relations du système, tandis que le modèle logique de données détaille la structure de la base de données sous forme de tables. Les clés primaires et étrangères sont utilisées pour maintenir la cohérence des données et faciliter la gestion des informations. Le modèle global offre une vue complète de la manière dont les utilisateurs, les compétences, les rôles, les permissions, les projets et d'autres éléments du système sont liés, formant ainsi la base solide pour la conception et la mise en œuvre de la base de données.

#### B. Méthode UML

Utilisateur :

Représente une entité qui a plusieurs attributs tels que l'ID, le nom, le prénom, et des références à des rôles et des promotions.

Chaque utilisateur a des liens externes associés, tels que Github, LinkedIn, etc., indiqués par la relation entre Utilisateur et LiensExternes.

Les utilisateurs ont également des journaux de temps, indiqués par la relation entre Utilisateur et LogTemps.

Les utilisateurs sont associés à des compétences spécifiques, comme le montre la relation entre Utilisateur et Competence.

Les utilisateurs peuvent avoir un ou plusieurs rôles et ces rôles peuvent déterminer les permissions qui leur sont accordées. C'est représenté par la relation entre Utilisateur, Role et Permission.

Les utilisateurs sont associés à des projets, principalement via le groupe auquel ils appartiennent. C'est-à-dire qu'un utilisateur appartient à un groupe et ce groupe travaille sur un projet spécifique.

Projet :

Représente une entité avec des attributs comme l'ID, le nom, la description, le lien vers un document PDF, et les dates de début et de fin.

Les projets sont associés à des unités spécifiques, indiquées par la relation entre Projet et Unite.

Les projets sont gérés ou réalisés par des groupes, comme le montre la relation entre Projet et Groupe.

Groupe :

Représente une entité qui a un ID, un nom, et une référence à un projet spécifique.

Les utilisateurs peuvent appartenir à un ou plusieurs groupes, ce qui est représenté par la relation entre Utilisateur et Groupe. Ainsi, un utilisateur peut travailler sur plusieurs projets s'il appartient à plusieurs groupes.

Role, Permission, et Competence :

Ces entités définissent respectivement les rôles des utilisateurs, les permissions associées à ces rôles et les compétences des utilisateurs.

Les relations entre ces entités et l'utilisateur déterminent les compétences d'un utilisateur, son rôle et les permissions qui lui sont accordées.

LiensExternes et LogTemps :

LiensExternes contient des liens vers des plateformes externes pour un utilisateur.

LogTemps contient des journaux de temps indiquant quand un utilisateur s'est connecté pour la première fois, la dernière fois, et le temps qu'il a passé.

Promotion :

Représente une entité avec des attributs comme le nom, le type, et les dates de début et de fin. Elle est associée à Utilisateur, indiquant à quelle promotion un utilisateur appartient.

En résumé, le diagramme UML décrit un système où les utilisateurs peuvent avoir différents rôles, compétences et permissions. Ils peuvent être associés à différents projets via les groupes auxquels ils appartiennent. Chaque projet est associé à une unité et est géré par un groupe. Les utilisateurs ont également des liens externes et des journaux de temps associés à eux.

Cardinalités

Réalisation : flèches pointillés

agrégation : diamant vide

composition : diamant plein

cardinalités : relations

Réalisation :

La réalisation est un type de relation en UML qui décrit une relation entre deux éléments, où l'un réalise (implémente) le comportement spécifié par l'autre. Elle est généralement utilisée pour représenter la relation entre une interface et la classe qui l'implémente. La flèche de réalisation est une ligne en pointillés avec une flèche ouverte pointant vers l'interface.

## Agrégation :

L'agrégation est une relation de type "ensemble-partie" entre deux classes où une classe (le tout) peut regrouper d'autres classes (les parties), mais les deux peuvent également exister indépendamment. Elle est représentée par une ligne avec un losange vide du côté du tout.

## Composition :

La composition est une forme plus forte d'agrégation, où la partie ne peut pas exister sans le tout. Si le tout est supprimé, la partie est également supprimée. Elle est représentée par une ligne avec un losange noir du côté du tout.

## Graphisme et ergonomie

### Contraintes visuelles

- Respecter la charte graphique du centre de formation.
- Interface intuitive et responsive.

### Solutions ergonomiques

- Menu de navigation clair et accessible.
- Affichage adaptatif en fonction du rôle de l'utilisateur.

### Solutions techniques

- Utilisation de CSS Flexbox et Grid pour le design responsive.
- Implémentation de React Hooks pour une meilleure gestion de l'état de l'application.

## Annexes:

Charte graphique

Maquettes et wireframes

MCD

MLD

UML

Autres ressources



## Annexe : charte graphique

#FFFFFF

Background Color

#051050

Background skill card

#7F56D9

Title font color

**fontStyle:**  
**Roboto Mono Bold**

#000000

Font color

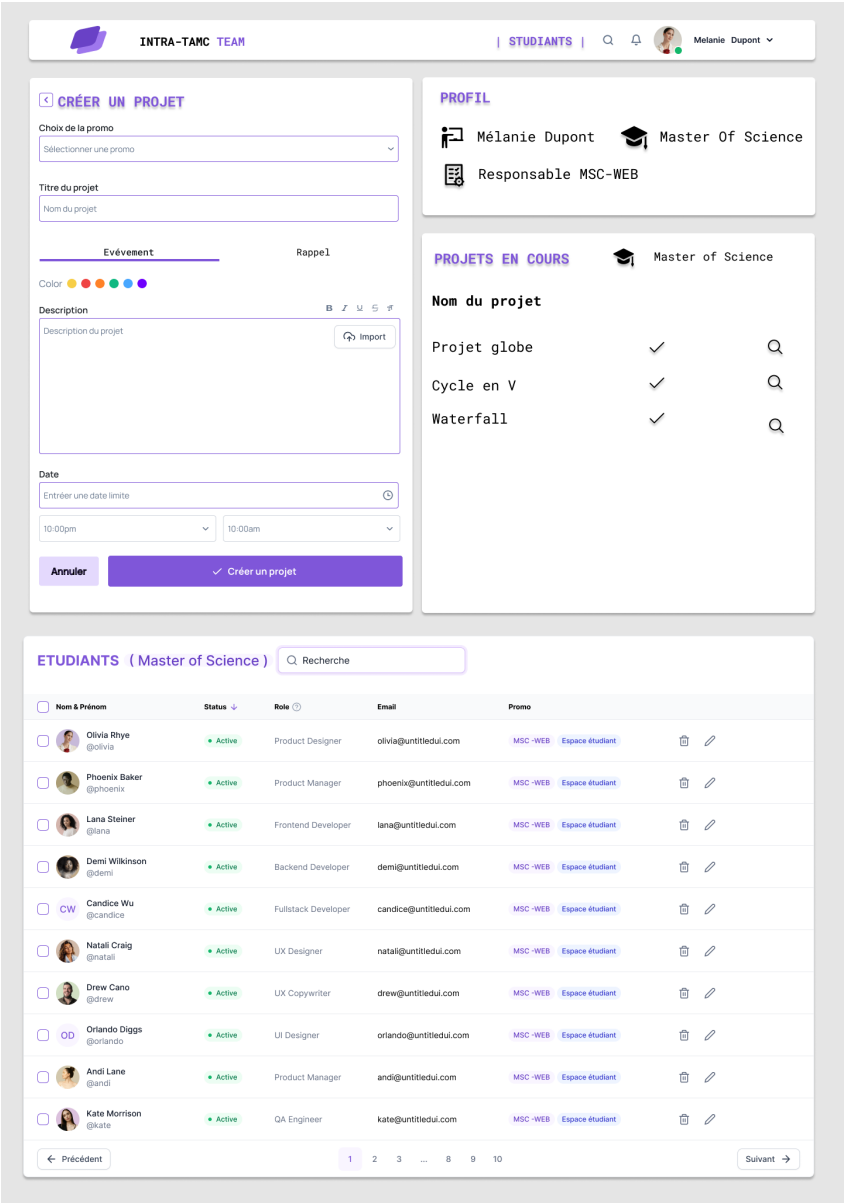
fontStyle:  
Roboto Mono Regular



INTRA-TAMC

Logo de l'entreprise

Annexe : Maquettes et wareframe



INTRA-TAMC

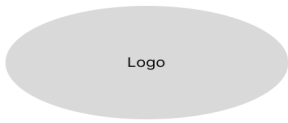
Votre plateforme scolaire de référence. Simplifiez votre expérience éducative avec INTRA-TAMC, l'outil conçu spécialement pour les étudiants et enseignants.

Connectez-vous rapidement et en toute sécurité

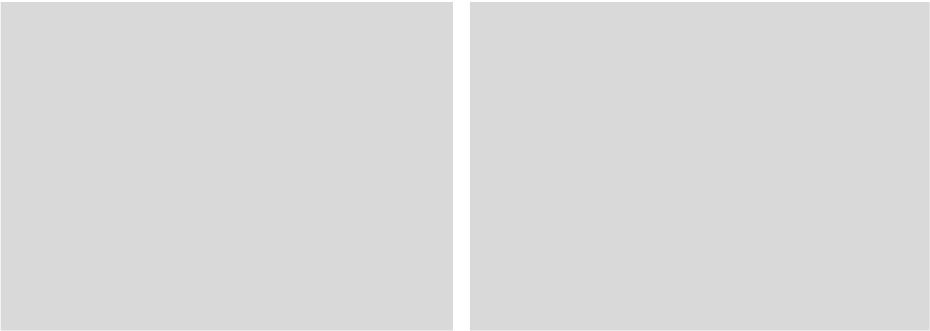
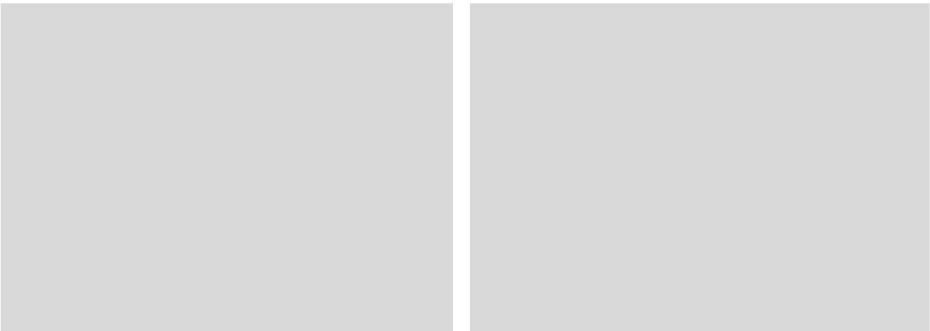
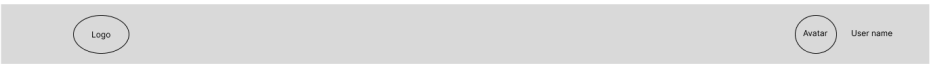
Sign in with Google

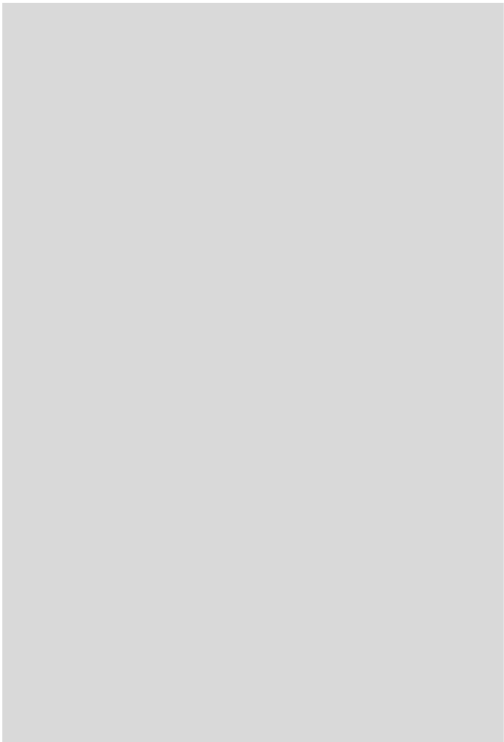




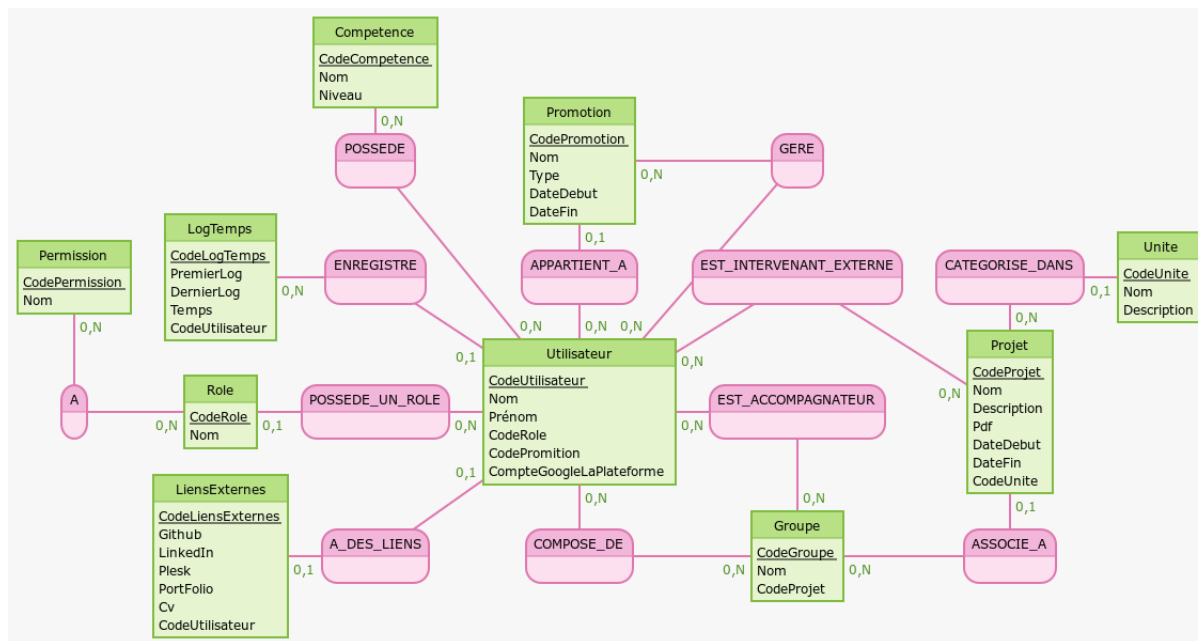


Clarity gives you the blocks & components you need to create a truly professional website, landing page or admin panel for your SaaS.

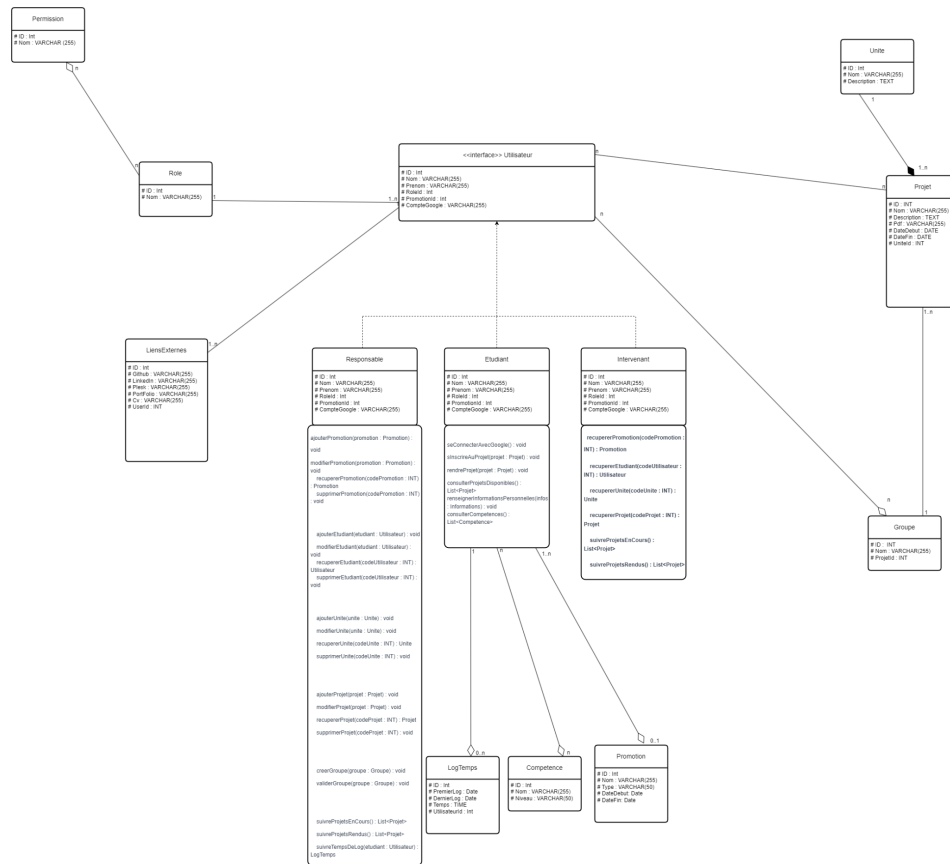




## Annexe : Schema database MCD



## Annexe : Schema database UML



## annexe MLD :

<b>TABLE Utilisateur</b> ( Id PK, Nom VARCHAR, Prenom VARCHAR, RoleId FK REFERENCES Role(Id), PromotionId FK REFERENCES Promotion(Id), CompGoogleLaPlateforme VARCHAR )	<b>TABLE Competence</b> ( Id PK, Nom VARCHAR, Niveau INT )	<b>TABLE LogTemps</b> ( Id PK, PremierLog DATETIME, DernierLog DATETIME, Temps INT, CodeUtilisateur FK REFERENCES Utilisateur(Id) )	<b>TABLE Permission</b> ( Id PK, Nom VARCHAR )	<b>TABLE Role</b> ( Id PK, Nom VARCHAR )	<b>TABLE LiensExternes</b> ( Id PK, Github VARCHAR, LinkedIn VARCHAR, Plesk VARCHAR, Portfolio VARCHAR, Cv VARCHAR, CodeUtilisateur FK REFERENCES Utilisateur(Id) )	<b>TABLE Promotion</b> ( Id PK, Nom VARCHAR, Type VARCHAR, DateDebut DATE, DateFin DATE )	<b>TABLE Unite</b> ( Id PK, Nom VARCHAR, Description VARCHAR )
<b>TABLE Projet</b> ( Id PK, Nom VARCHAR, Description VARCHAR, Pdt VARCHAR, DateDebut DATE, DateFin DATE, UniteId FK REFERENCES Unite(Id) )	<b>TABLE Groupe</b> ( Id PK, Nom VARCHAR, CodeProjet FK REFERENCES Projet(Id) )	<b>TABLE Utilisateur_Competence</b> ( UtilisateurId FK REFERENCES Utilisateur(Id), CompetenceId FK REFERENCES Competence(Id), PK (UtilisateurId, CompetenceId) )	<b>TABLE Utilisateur_Promotion</b> ( UtilisateurId FK REFERENCES Utilisateur(Id), PromotionId FK REFERENCES Promotion(Id), PK (UtilisateurId, PromotionId) )	<b>TABLE Utilisateur_Groupe</b> ( UtilisateurId FK REFERENCES Utilisateur(Id), GroupeId FK REFERENCES Groupe(Id), PK (UtilisateurId, GroupeId) )	<b>TABLE Utilisateur_Projet</b> ( UtilisateurId FK REFERENCES Utilisateur(Id), ProjetId FK REFERENCES Projet(Id), PK (UtilisateurId, ProjetId) )	<b>TABLE Projet_Unite</b> ( ProjetId FK REFERENCES Projet(Id), UniteId FK REFERENCES Unite(Id), PK (ProjetId, UniteId) )	<b>TABLE Permission_Role</b> ( PermissionId FK REFERENCES Permission(Id), RoleId FK REFERENCES Role(Id), PK (PermissionId, RoleId) )