

Document technique:

Projet : ProManager

Outils de gestion de projets et d'équipes

Introduction :

Ce projet vise à mettre à disposition de manager d'équipe un outil de gestion d'équipe et de projet, qui leur permettra d'assigner des personnes ressources à des projets, créer des projets, affecter des tâches aux membres des projets, faire un suivi et une évaluation des tâches après leur exécution.

I. Besoins fonctionnels :

- Gestion des utilisateurs
- Gestion des projets
- Gestion des tâches

II. Besoins non fonctionnels

- L'application doit implémenter un système d'authentification et d'autorisation.
- Les données de l'application doivent être stockées dans une base de données relationnelles.
- L'application doit être accessible via un navigateur web (Google chrome, Firefox, Safari etc.)

III. Le diagramme de cas d'utilisations

Ces diagrammes modélisent à quoi sert le système, en organisant les interactions possibles avec les acteurs.

1. Les acteurs de l'application:

Le système sera composé des acteurs suivant:

- **Utilisateur simple :** Un membre quelconque de l'équipe qui n'est ni manager ni admin, par exemple un développeur ou un assistant.
Il peut entre autres modifier les propriétés d'une tâche : soumettre, ajouter un commentaire.
- **Manager :** Le chef de projet a comme rôle principal la gestion du projet et des tâches..
Il peut créer un projet, le modifier, le supprimer, ajouter ou enlever des membres au projet. Il peut aussi créer, modifier ou encore supprimer des tâches; partager les tâches

Il peut créer, modifier ou supprimer d'autres utilisateurs, définir leur rôle à la création ou à l'activation. Il a aussi tous les droits dont disposent les managers et les simples utilisateurs.

Diagramme de cas d'utilisation pour un système de gestion de projet.

Acteurs : Employé, Manager, Admin, ProManager.

Cas d'utilisation :

- S'inscrire
- Consulter le projet
- Consulter la tâche
- Soumettre la tâche
- Evaluer la tâche
- Résoudre la tâche
- Editer le projet
- Supprimer le projet
- Gérer le projet
- Créer le projet
- créer via importation csv
- Gérer la tâche
- Créer la tâche
- Editer la tâche
- Supprimer la tâche
- Rechercher
- Rechercher l'utilisateur
- Rechercher le projet
- Rechercher la tâche
- Supprimer l'utilisateur
- Gérer l'utilisateur
- Créer un utilisateur
- Editer l'utilisateur
- Sauthentifier
- Authentification google
- Authentification github
- Authentification slack

Relations :

- Employé : S'inscrire, Consulter le projet, Consulter la tâche, Soumettre la tâche, Evaluer la tâche, Résoudre la tâche, Editer le projet, Supprimer le projet, Gérer le projet.
- Manager : Consulter le projet, Consulter la tâche, Soumettre la tâche, Evaluer la tâche, Résoudre la tâche, Editer le projet, Supprimer le projet, Gérer le projet, Créer le projet, créer via importation csv, Gérer la tâche, Créer la tâche, Editer la tâche, Supprimer la tâche, Rechercher, Rechercher l'utilisateur, Rechercher le projet, Rechercher la tâche, Supprimer l'utilisateur, Gérer l'utilisateur, Créer un utilisateur, Editer l'utilisateur.
- Admin : Consulter le projet, Consulter la tâche, Soumettre la tâche, Evaluer la tâche, Résoudre la tâche, Editer le projet, Supprimer le projet, Gérer le projet, Créer le projet, créer via importation csv, Gérer la tâche, Créer la tâche, Editer la tâche, Supprimer la tâche, Rechercher, Rechercher l'utilisateur, Rechercher le projet, Rechercher la tâche, Supprimer l'utilisateur, Gérer l'utilisateur, Créer un utilisateur, Editer l'utilisateur.
- ProManager : Consulter le projet, Consulter la tâche, Soumettre la tâche, Evaluer la tâche, Résoudre la tâche, Editer le projet, Supprimer le projet, Gérer le projet, Créer le projet, créer via importation csv, Gérer la tâche, Créer la tâche, Editer la tâche, Supprimer la tâche, Rechercher, Rechercher l'utilisateur, Rechercher le projet, Rechercher la tâche, Supprimer l'utilisateur, Gérer l'utilisateur, Créer un utilisateur, Editer l'utilisateur.

Notes :

- (Condition : Admin ou Manager)
- «include»
- «extend»

3. Actions :

Créer projets

- **ID : P1**
- **Intitulé : Créer projets**
- **Acteurs principaux : Manager (Chef de projet)**
- **Parties prenantes et intérêts :**
Le chef de projets veut un moyen de créer des projets et d'y allouer de la ressource humaine.
- **Précondition :**
L'utilisateur doit être authentifié et disposer de la permission de gérer des projets.
- **Garantie de succès :**
Un nouveau projet est créé.
- **Scénario nominal:**
 - 1- Le manager clique sur le bouton "créer projet" .
 - 2- Un formulaire de création lui est montré.
 - 3- Il remplit le formulaire et le soumet.
 - 4- Une requête est lancée à l'API web responsable de la gestion de projets.
 - 5- L'API web vérifie si l'utilisateur a le droit de créer des projets ou pas.
 - 6- L'API web transmet au service responsable des projets (ProjectService).
 - 7- Un projet est créé et un objet est retourné à l'api web qui le retransmet au client .
 - 8- Un message de succès est affiché.
- **Scénario alternatif :**
 - 3a- Le manager décide d'annuler
 - 3.b- L'utilisateur importe le projet à partir d'un fichier excel(xls ou csv), et automatiquement les champs du projet seront remplis par les données provenant de ce fichier.
- **Scénario d'exception :**
 - 5a- L'utilisateur n'a pas les droits de créer des projets, un message d'erreur est retourné.
 - 7a- Un projet ayant le même nom existe déjà, une erreur est retourné
- **Spécification particulière :**
Via l'interface web(Navigateur)

Consulter un projet

- **ID: P2**
- **Intitulé : Consulter un projet**
- **Acteurs principaux : Manager(chef de projet) et membre de l'équipe**
- **Parties prenantes et intérêts:**
Un manager ou un membre de l'équipe veut consulter un projet pour y voir les détails.
- **Précondition :**
L'utilisateur doit être authentifié et disposer les permissions de consulter des projets
- **Garantie de succès :**
L'utilisateur voit les détails du projet

- **Scénario nominal :**
 1. L'utilisateur clique sur le projet dont il veut voir les détails
 2. Une requête est lancée à l'api web responsable de la gestion de projets
 3. L'api vérifie si l'utilisateur a le droit de consulter le projet concerné ou pas.
 4. L'api transmet au service responsable des projets (ProjectService).
 5. Un projet concerné est retourné à l'api qui le retransmet au client .
 6. Un message de succès est affiché à l'écran.
- **Scénario d'exception:**
 - 3a- L'utilisateur n'a pas le droit de consulter le projet concerné, un message d'erreur est retourné.
 - 5a- Le projet sélectionné est introuvable, un message d'erreur est retourné.
- **Spécification particulière:**
Via l'interface web(navigateur)

Éditer ou modifier un projet

- **ID: P3**
- **Intitulé : Éditer un projet**
- **Acteurs principaux: Manager(Chef de projet)**
Le manager veut modifier les informations relatives à un projet donné.
- **Précondition:**
L'utilisateur doit être authentifié et disposer les permissions d'éditer un projet(MANAGER ou ADMIN)
- **Garantie de succès:**
Les informations modifiables du projet sont modifiées.
- **Scénario nominal:**
 1. L'utilisateur demande la page de modification d'un projet
 2. Il modifie certaines informations
 3. Il soumet le formulaire
 4. Une requête est lancée à l'api web responsable de la gestion de projets.
 5. L'api web transmet au service responsable des projets (ProjectService).
 6. Le projet est modifié ou une erreur est retournée
 7. Le message de succès ou d'erreur est affiché à l'écran.
- **Scénario alternatif :**
 3. Le manager décide d'annuler
- **Scénario d'exception:**
 - 3a- Les informations saisies sont incorrectes(sans requis non remplis par exemple) et les messages d'erreurs appropriés sont affichés sur le formulaire.
- **Spécification particulière:**
Via l'interface web(navigateur)

Chercher un projet

- **ID : P4**
- **Intitulé : Chercher un projet**

- **Principaux acteurs : Manager et Membre de l'équipe**
- **Parties prenantes et intérêts :** Un manager ou un membre de l'équipe veut disposer d'un moyen de rechercher de projet dans liste des projets
- **Précondition:** L'utilisateur doit être authentifié
- **Garantie de succès :**
Un projet ou plusieurs projets sont retournés.
- **Scénario nominal :**
 1. L'utilisateur entre un mot clé dans le champ de recherche et clique sur le bouton "chercher"
 2. Une requête est lancée à l'api web responsable de la gestion de projets.
 3. L'api web vérifie si l'utilisateur est authentifié
 4. L'api web transmet la requête au service responsable des projets (ProjectService).
 5. Le service retourne à l'api web une liste contenant tous les projets dont la désignation ou le statut match avec le mot clé.
 6. Cette liste est retransmise au client qui l'affiche à l'écran
- **Scénario d'exception:**
3.a- L'utilisateur n'est pas authentifié, un message d'erreur est retourné
- **Spécification particulière:**
Via l'interface web(navigateur)

Voir les statistiques d'un projet

- **ID : P5**
- **Intitulé : Voir les statistiques d'un projet**
- **Principaux acteurs : Manager**
- **Parties prenantes et intérêts :**
Un manager veut disposer d'un moyen de voir les statistiques d'un projet
- **Précondition :** L'utilisateur doit être authentifié et disposer des droits pour voir les statistiques.
- **Garantie de succès :**
Les statistiques sont retournés
- **Scénario nominal :**
 1. Le manager clique sur le bouton "voir les stats du projet"
 2. Une requête est lancée à l'api web responsable de la gestion de projets(Project Controller)
 3. L'api web vérifie si l'utilisateur a le droit de voir les statistiques ou pas
 4. L'api web transmet la requête au service responsable de la gestion (Project Service)
 5. Le service retourne un objet qui représente les statistiques du projet à l'api web qui le retransmet au client
 6. Un message de succès est affiché
- **Scénario d'exception :**
3.a- L'utilisateur n'a pas le droit de voir les statistiques du projet, un message d'erreur est retourné.

- **Spécification particulière:** Via l'interface web(navigateur)

Supprimer un projet

- **ID : P6**
- **Intitulé: Supprimer un projet**
- **Principaux acteurs: Manager(Chef de projet)**
- **Parties prenantes et intérêts:**
Un manager veut supprimer définitivement un projet de la base de données
- **Précondition:** L'utilisateur doit être authentifié et disposer les permissions (ADMIN ou MANAGER) de supprimer un projet.
- **Garantie de succès:**
Le projet concerné est supprimé
- **Scénario nominal:**
 - 1.L'utilisateur clique sur le bouton "supprimer"
 - 2.Un formulaire de confirmation lui est affiché
 - 3.Il clique sur le bouton "confirmer"
 - 4.Une requête est lancée à l'api web responsable de la gestion de projets.
 - 5.L'api web transmet les infos de la requête au service responsable des projets (ProjectService).
 - 6.Le projet est supprimé ou une erreur est retourné
 - 7..Le message de succès ou d'erreur est affiché à l'écran.
- **Scénario alternatif:**
3a- Le manager décide d'annuler
- **Scénario d'exception:**
6a- Le projet sélectionné est introuvable, un message d'erreur est retourné.
- **Spécification particulière :**
Via l'interface web(navigateur).

Lister les membre d'un projet

- **ID: P7**
- **Intitulé: Lister les membres d'un projet**
- **Principaux acteurs: Manager(Chef de projet), membre de l'équipe projet**
- **Parties prenantes et intérêts:**
Le manager ou un membre veut voir la liste des membre d'un projet
- **Précondition:**
L'utilisateur doit être authentifié et disposer les permissions (ADMIN ou MANAGER ou MEMBER) de lister les membres d'un projet.
- **Garantie de succès:**
L'utilisateur voit la liste des membre du projet concerné
- **Scénario nominal:**
 1. L'utilisateur effectue le cas P2(Cas consulter un projet)
 - 2.L'utilisateur clique sur "Liste de membres"
 - 3.Une requête est lancée à l'api web responsable de la gestion de projets
 4. L'api web vérifie si l'utilisateur à le droit ou pas de voir les membres d'un projet

5. L'api web transmet les infos de la requête au service responsable des projets (ProjectService).
 6. Il retourne la liste des membres du projet concerné à l'api qui le retransmet au client.
 7. La liste des membres est affichée à l'écran.
- **Scénario d'exception:**
4a- L'utilisateur n'a pas le droit de voir les membres d'un projet, une erreur est retournée
 - **Spécification particulière:**
Via l'interface web

Ajouter un utilisateur au projet

- **ID:P8**
- **Intitulé: Ajouter un utilisateur à un projet**
- **Principaux acteurs : Manager (Chef de projet)**
- **Partie prenantes et intérêts:**
Le manager veut ajouter un utilisateur à un projet existant.
- **Précondition:**
L'utilisateur doit être authentifié et disposer des permissions (ADMIN ou MANAGER) pour ajouter des utilisateurs aux projets.
- **Garantie de succès :**
L'utilisateur est ajouté au projet
- **Scénario nominal:**
 1. L'utilisateur effectue le cas P7 (Lister les membres d'un projet)
 2. Il clique sur le bouton "nouveau membre"
 3. Un modal est affiché invitant à choisir le nouveau membre dans la liste
 4. Le manager choisit le membre en question et clique sur le bouton "Ajouter"
 5. Une requête est lancée à l'api responsable de la gestion de projets
 6. L'api vérifie si l'utilisateur a le droit ou pas d'ajouter un membre à un projet
 7. L'api transmet les infos de la requête au service responsable des projets (ProjectService).
 8. La personne est ajoutée au projet ou une erreur est retournée
 9. Un message d'erreur ou de succès est affiché à l'écran.
- **Scénario alternatif:**
4a- Le manager décide d'annuler
- **Scénario d'exception :**
6a- L'utilisateur n'a pas le droit d'ajouter un membre à un projet, une erreur est retournée.
- **Spécification particulière:**
Via l'interface web.

Enlever un utilisateur au projet

- **ID:P9**
- **Intitulé: Enlever un utilisateur à un projet**

- **Principaux acteurs : Manager (Chef de projet)**
- **Partie prenantes et intérêts:**
Le manager veut enlever un utilisateur d'un projet.
- **Précondition:**
L'utilisateur doit être authentifié et disposer les permissions (ADMIN ou MANAGER) pour enlever des utilisateurs aux projets.
- **Garantie de succès :**
L'utilisateur est enlevé du projet
- **Scénario nominal:**
 1. L'utilisateur effectue le cas P7(Lister les membres d'un projet)
 2. Il clique sur l'icône de la suppression devant le nom du membre dans la liste
 3. Un formulaire de confirmation lui est affiché
 4. Il clique sur le bouton "confirmer"
 5. Une requête est lancée à l'api responsable de la gestion de projets
 6. L'api vérifie si l'utilisateur a le droit ou pas d'enlever un membre à un projet
 7. L'api web transmet les infos de la requête au service responsable des projets (ProjectService).
 8. L'utilisateur est enlevé du projet ou une erreur est retournée
 9. Un message d'erreur ou de succès est affiché à l'écran.
- **Scénario alternatif:**
4a- Le manager décide d'annuler
- **Scénario d'exception :**
6a- L'utilisateur n'a pas le droit d'enlever un membre à un projet, une erreur est retournée.
- **Spécification particulière:**
Via l'interface web.

Lister les tâches d'un projet

- **ID: P10**
- **Intitulé: Lister les tâches d'un projet**
- **Principaux acteurs: Manager(Chef de projet), membre de l'équipe projet**
- **Parties prenantes et intérêts:**
Le manager ou un membre veut voir la liste des tâches d'un projet
- **Précondition:**
L'utilisateur doit être authentifié et disposer les permissions (ADMIN ou MANAGER ou MEMBER) de lister les tâches d'un projet.
- **Garantie de succès:**
L'utilisateur voit la liste des tâches du projet concerné
- **Scénario nominal:**
 1. L'utilisateur effectue le cas P2(Cas consulter un projet)
 2. L'utilisateur clique sur "Liste de tâches"
 3. Une requête est lancée à l'api web responsable de la gestion de projets
 4. L'api web vérifie si l'utilisateur a le droit ou pas de voir les tâches d'un projet

5. L'api web transmet les infos de la requête au service responsable des projets (ProjectService).
 6. Il retourne la liste des tâches du projet concerné à l'api web qui le retransmet au client.
 7. La liste des tâches est affichée à l'écran.
- **Scénario d'exception:**
4a- L'utilisateur n'a pas le droit de voir les tâches d'un projet, une erreur est retournée
 - **Spécification particulière:**
Via l'interface web

Ajouter une tâche à un projet

- **ID: P11**
- **Intitulé: Ajouter une tâche**
- **Principaux acteurs: Manager(Chef de projet)**
- **Partie prenantes et intérêts:**
Un chef de projet veut ajouter une tâche à un projet
- **Précondition:**
L'utilisateur doit être authentifié et disposer les permissions (ADMIN ou MANAGER) d'ajout d'une tâche.
- **Garantie de succès:**
Une tâche est créée et ajoutée au projet
- **Scénario nominal:**
 1. L'utilisateur effectue le cas P10(Lister les tâches d'un projet)
 2. Il clique sur le bouton "Nouvelle tâche"
 3. Un formulaire de création d'une tâche lui est affiché
 4. Il rentre les informations relatives à une tâche et soumet le formulaire
 5. Une requête est lancée à l'api web responsable de la gestion de projets
 6. L'api web vérifie si l'utilisateur a le droit ou pas d'ajouter des tâches à un projet
 7. L'api web transmet au service responsable des projets (ProjectService).
 8. Ce service (ProjectService) à son tour transmet les infos de la requête au service responsable de la gestion des tâches(TaskService)
 9. TaskService retourne un objet représentant la tâche créée à ProjectService, qui retourne à son api web qui retransmet au client.
 10. Un message de succès est affiché à l'écran
- **Scénario alternatif:**
4a- Le manager décide d'annuler
- **Scénario d'exception:**
6a- L'utilisateur n'a pas le droit d'ajouter une tâche à un projet, un message d'erreur est retourné.
9a- Une erreur lors de l'ajout, un message d'erreur est retourné
- **Spécification particulière**
Via l'interface web(Navigateur)

Enlever une tâche à un projet

- **ID: P12**
 - **Intitulé: Enlever une tâche**
 - **Principaux acteurs: Manager(Chef de projet)**
 - **Partie prenantes et intérêts:**
Un chef de projet veut enlever une tâche d'un projet
 - **Précondition:**
L'utilisateur doit être authentifié et disposer les permissions (ADMIN ou MANAGER) d'enlever une tâche.
 - **Garantie de succès:**
La tâche concernée est enlevée du projet
 - **Scénario nominal:**
 1. L'utilisateur effectue le cas P10(Lister les tâches d'un projet)
 2. Il clique sur le bouton "supprimer" ou l'icône de suppression devant la tâche à supprimer.
 3. Un formulaire de confirmation lui est affiché
 4. Il clique sur le bouton "confirmer"
 5. Une requête est lancée à l'api web responsable de la gestion de projets
 6. L'api web vérifie si l'utilisateur a le droit ou pas d'enlever des tâches à un projet
 7. L'api web transmet au service responsable des projets (ProjectService).
 8. Ce service (ProjectService), à son tour transmet les infos de la requête au service responsable de la gestion des tâches(TaskService).
 9. La tâche concernée est enlevée du projet et supprimée de la base de données.
 10. Un message de succès est affiché à l'écran.
 - **Scénario alternatif:**
4a- Le manager décide d'annuler.
 - **Scénario d'exception:**
6a- L'utilisateur n'a pas le droit d'enlever une tâche à un projet, un message d'erreur est retourné.
9a- Une erreur lors de la suppression, un message d'erreur est retourné.
- Spécification particulière**
Via l'interface web(Navigateur)

S'inscrire

- **ID:U1**
- **Intitulé:S'inscrire**
- **Principaux acteurs: Manager(chef de projet), membre de l'équipe**
- **Parties prenantes et intérêts :**
Un nouveau manager ou membre quelconque de l'équipe veut s'inscrire sur l'application pour participer aux différentes activités.
- **Précondition :** Aucune
- **Garantie de succès :**
L'utilisateur est créé

- **Scénario nominal:**
 - 1.L'utilisateur demande le formulaire de création de compte
 - 2.Il entre ses informations et soumet le formulaire
 3. Une requête est lancée à l'api web responsable de la gestion des utilisateurs
 4. L'api transmet les données de la requête au service responsable de la gestion des utilisateurs.
 - 5.Le service retourne un objet représentant l'utilisateur créé
 - 6.L'api retransmet la réponse au client
 - 7.Un message de succès est affiché à l'écran.
- **Scénario alternatif :**
 - 2a - L'utilisateur décide d'annuler
- **Scénario d'exception :**
 - 5a - Erreur lors de la création de l'utilisateur(un utilisateur avec le même email existe déjà par exemple), un message est retourné
- **Spécification particulière :**

Via l'interface web(Navigateur)

S'authentifier

- **ID: U2**
- **Intitulé: S'authentifier**
- **Principaux acteurs : Manager et membre de l'équipe**
- **Parties prenantes et intérêts:**

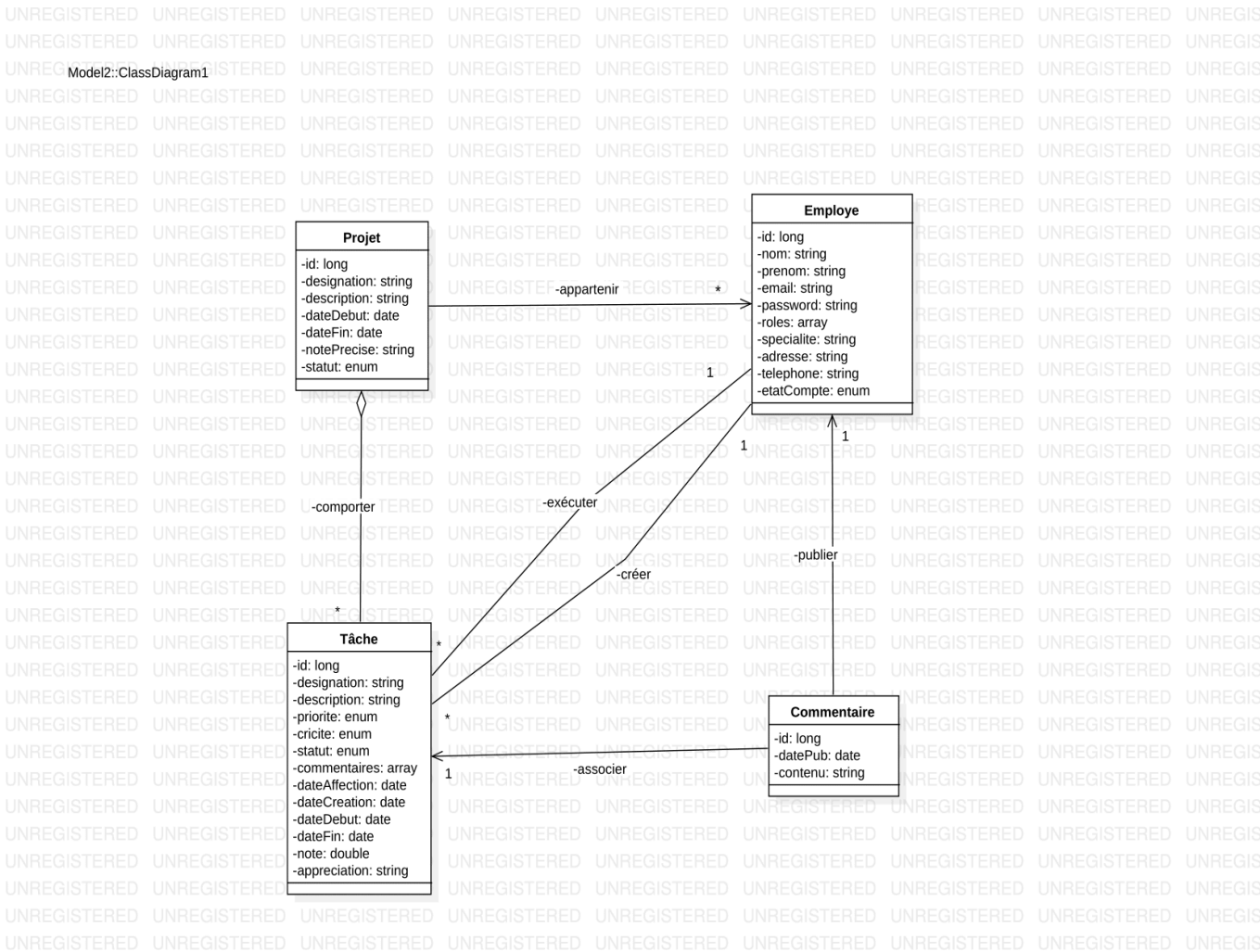
Un manager ou un membre de l'équipe veut disposer d'un accès à l'application
- **Garantie de succès:**

L'utilisateur a accès à l'application
- **Scénario nominal:**
 1. L'utilisateur demande la page d'authentification
 - 2.Il remplit le formulaire et le soumet
 3. Une requête est lancée à l'api web responsable de la gestion des utilisateurs
 4. L'api transmet les données de la requête au service responsable de la gestion des utilisateurs.
 5. Les métadonnées ou une erreur sont retournées
 - 6.Le navigateur stocke les identifiants de l'utilisateur
 7. Un message de succès ou d'erreur est affiché à l'écran et l'utilisateur est redirigé vers la page d'accueil
- **Scénario alternatif :**
 - 2a- L'utilisateur décide d'annuler l'authentification
 - 2.b - L'utilisateur indique un oubli de mot de passe
- **Scénario d'exception :**
 - 5a- Les identifiants sont incorrects, une erreur est retournée
 - 6a-Le compte de l'utilisateur n'est pas encore activé, un message invitant l'utilisateur à demander l'activation par l'admin est affiché
- **Spécification particulière:**

Via l'interface web(navigateur)

IV. Diagrammes de classes :

Les diagrammes de classes permettent de spécifier la structure et les liens entre les objets dont le système est composé : ils spécifient qui sera à l'oeuvre dans le système pour réaliser les fonctionnalités décrites par les diagrammes des cas d'utilisation.



Attribue	Description	Type	Optionnel
id	L'identifiant unique de l'employé	Long	
name	Le nom de l'employé	String	
firstname	Le prénom de l'employé	String	

email	L'adresse email de l'employé	String	
password	Le mot de passe de l'employé	String	
roles	L'ensemble des rôles des employés(Admin, Manager etc.)	String[]	
jobTitle	La spécialité ou le domaine d'expertise d'un de l'employé.	String	
address	L'adresse d'un employé	String	
telephone	Le numéro de téléphone d'un employé	String	
tasks	L'ensemble des tâches dont un employé est responsable de l'exécution.	Array	
accountState	L'état du compte d'un employé	Boolean	

Dictionnaire de donnée de l'entité "Employe"

Attribue	Description	Type	Optionnel
id	L'identifiant unique du projet		
title	La désignation du projet	String	
description	La description du projet	String	true
startDate	La date de début du projet	DateTime	
EndDate	La date de fin du projet	DateTime	
specificNote	Une note précise concernant le projet	String	true
statut	Le statut du projet	Enumeration	
members	Les membres du projets	Array	
tasks	Les tâches qui constituent le projet	Array	

Dictionnaire de donnée de l'entité "Projet"

Attribue	Description	Type	Optionnel
----------	-------------	------	-----------

id	L'identifiant unique de la tâche	Long	
title	La désignation de la tâche	String	
description	La description de la tâche	String	true
priority	Le niveau du priorité de la tâche	Enumeration	
criticality	Le niveau de criticité de la tâche	Enumeration	
statut	Le statut de la tâche	Enumeration	
creationDate	La date de création de la tâche	DateTime	
affectationDate	La date d'affectation de la tâche	DateTime	
startDate	La date de début de la tâche	DateTime	
endDate	La date de fin de la tâche	DateTime	
author	Les informations sur l'auteur de la tâche	Objet	
responsable	Les infos sur l'employé chargée d'effectuer la tâche	Objet	
project	Le projet dont appartient la tâche	Objet	
taskEvaluation	La note donnée à la tâche à l'issue de l'évaluation	double	
mark	Le commentaire d'appréciation lors de l'évaluation	String	True

Dictionnaire de donnée de l'entité "Tâche"

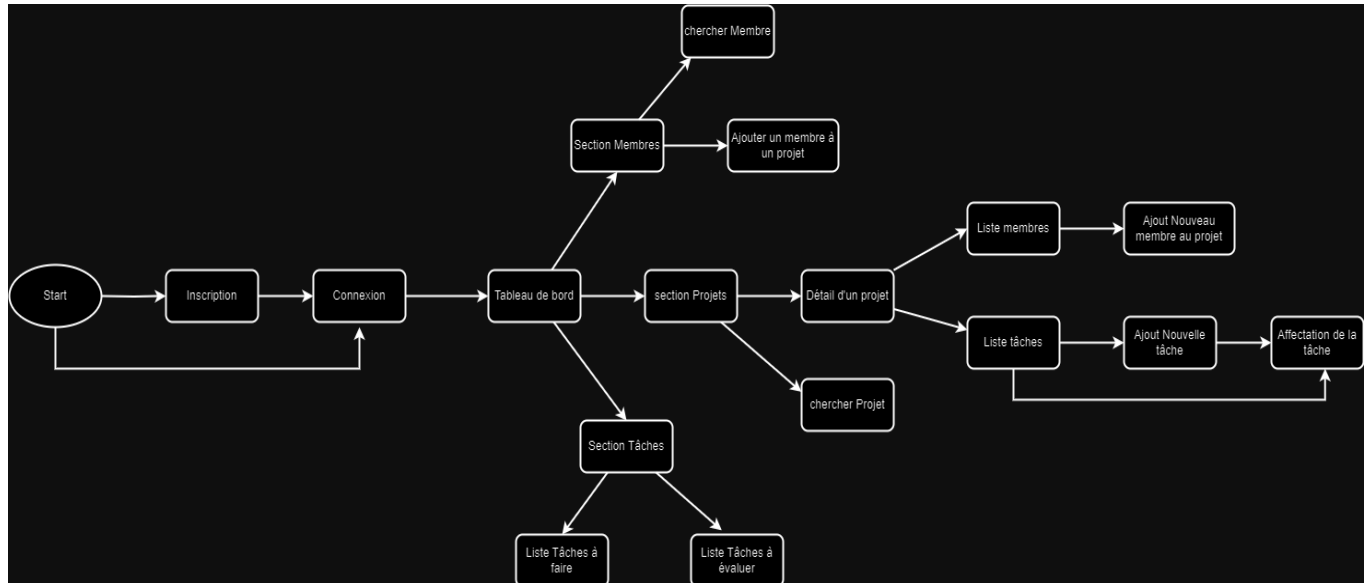
Attribue	Description	Type	Optionnel
id	L'identifiant unique du commentaire	Long	
pubDate	La date de publication du commentaire	DateTime	
content	Le contenu du commentaire	String	
author	L'utilisateur qui a fait le commentaire	Objet	
task	La tâche concernée par le commentaire	Objet	

Dictionnaire de donnée de l'entité "Commentaire"

V. User flows :

Le user flow est une représentation graphique de l'ensemble des parcours et des interactions qu'un utilisateur peut effectuer sur un site ou une application. Il permet un gain de temps important durant la réalisation d'un projet notamment sur l'étape des wireframes.

Ci-dessous se trouve le flow principal pour l'application ProManager.



VI. Diagrammes de séquence:

Les diagrammes de séquence décrivent de manière séquentielle dans le temps comment les éléments du système interagissent entre eux et avec les acteurs pour réaliser les cas d'utilisations:

- Les objets au cœur du système interagissent en s'échangeant des messages.
- Les acteurs interagissent avec le système au moyen d'IHM (Interface Homme-Machine)

VII. Outils et environnement technologique:

Ce projet est réalisé principalement avec la technologie Java qui est un langage de programmation orienté objet multiplateforme largement utilisé par les entreprises. J'ai notamment utilisé le framework Spring qui est de facto le framework n°1 des frameworks Java et aussi l'un des framework les plus utilisés actuellement par les entreprises pour la création des backends des applications.

Spring MVC

Pour ce projet j'ai choisi d'utiliser le module Spring MVC de Spring Framework qui est un choix technique solide pour plusieurs raisons. Fondamentalement, il repose sur le modèle MVC (Modèle-Vue-Contrôleur), qui divise l'application en trois composants distincts, chacun ayant son rôle précis. Le modèle représente les données métier, la vue est responsable de l'interface utilisateur et le contrôleur agit comme un médiateur entre les deux.

Ce qui rend Spring MVC particulièrement puissant, c'est son mécanisme de gestion des requêtes HTTP. Lorsqu'une requête arrive, elle est interceptée par le Front Controller, qui est le DispatcherServlet dans le cas de Spring MVC. Ce dernier consulte le mapping des URL pour déterminer le contrôleur approprié à appeler. Une fois le contrôleur identifié, il traite la requête en utilisant les annotations ou la configuration XML pour mapper les méthodes aux URL.

Ensuite, le contrôleur traite la requête en interagissant avec les services métier et en préparant les données nécessaires à l'affichage. Une fois cela fait, le contrôleur retourne la vue appropriée, généralement en utilisant le mécanisme de résolution de vues de Spring pour traduire les noms de vues logiques en vues physiques.

L'avantage clé de ce fonctionnement est sa flexibilité et son extensibilité. En utilisant des concepts tels que les intercepteurs, les convertisseurs, les résolveurs de vue, les gestionnaires d'exceptions, etc., Spring MVC permet de personnaliser pratiquement tous les aspects du cycle de vie d'une requête.

De plus, Spring MVC s'intègre harmonieusement avec d'autres fonctionnalités de Spring, telles que Spring Security pour la gestion de la sécurité, Spring Data pour l'accès aux données, et Spring Boot pour une configuration simplifiée et une mise en route rapide des applications.

En résumé, Spring MVC offre une architecture solide et bien définie pour développer des applications web Java, avec une gestion efficace des requêtes HTTP et une grande extensibilité grâce à son architecture modulaire.

