**UNIVERSITÉ DU QUÉBEC EN OUTAOUAIS**

Département d’Informatique et Ingénierie

INF1163 : Modélisation et conception orientée objet

Enseignant : **ABD-ALI JAMAL**

Devoir

**Développement d’un système nommé TimeLog qui offre une assistance automatisée de calcul de temps, de salaires, et de contrôle de budget des projets**

Réalisé par :

**MAKOUET MOUNYICHE NJIEMOUN, Valerie**

**NYAMI NDJEUKOU DJOKOMENI, Annie Christelle**

Été 2024

Table des matières

[Introduction 3](#_Toc171560471)

[Objectifs 4](#_Toc171560472)

[Exigences Fonctionnelles 5](#_Toc171560473)

[Exigences Non Fonctionnelles 6](#_Toc171560474)

[Description Complète des Cas d'Utilisation 7](#_Toc171560475)

[Cas d'utilisation : Débuter une activité 9](#_Toc171560476)

[Scenario alternatif 10](#_Toc171560477)

[Cas d'utilisation : Terminer une activité 10](#_Toc171560478)

[Scenario alternatif 11](#_Toc171560479)

[Conclusion 11](#_Toc171560480)

[Annexes 12](#_Toc171560481)

# Introduction

Le système dénommé TimeLog est un logiciel conçu par la compagnie de développement de logiciel X pour répondre à des besoins spécifiques de gestion efficace des projets et des employés en permettant notamment aux employés d’enregistrer leurs heures de travail et aux administrateurs de gérer les projets et de générer les rapports. Ce système offre une assistance automatisée de calcul de temps, de salaires et de contrôle de budget de ses projets, qui permettra ainsi à ladite compagnie de rationaliser ses processus de gestion des projets, d’améliorer la productivité des employés et de prendre des décisions éclairées basées sur des données précises et fiables.

La première phase de ce projet, vise à déterminer notre compréhension du système TimeLog et à mettre en place une base solide qui va servir de point de départ de la réalisation des autres phases du projet.

# Objectifs

Ces objectifs visent à créer un système complet et flexible qui optimisera la gestion des ressources et des projets de l'entreprise, tout en simplifiant les processus de suivi du temps et de calcul des salaires.

Les principaux objectifs du logiciel sont les suivants :

1. Automatisation du suivi du temps :
   * Permettre aux employés d'enregistrer les horaires de début et de fin de leurs activités de travail.
   * Associer automatiquement chaque activité à un projet et une discipline spécifiques.
2. Gestion des projets:
   * Maintenir une base de données des projets en cours avec leurs caractéristiques (dates de début et de fin, budgets par discipline).
   * Fournir des rapports d'état pour chaque projet et pour tous les projets.
3. Gestion des ressources humaines :
   * Gérer les informations des employés, notamment leurs taux horaires et leur historique d'emploi.
   * Limiter le nombre de projets auxquels un employé peut être assigné simultanément.
4. Calcul automatisé des salaires :

* Calculer les salaires bruts et nets des employés en fonction des heures travaillées.
* Générer des talons de paie pour les dernières 26 périodes de paie.

1. Contrôle budgétaire :

* Suivre l'avancement des projets en comparant les heures travaillées au budget fourni pour chaque discipline.
* Etablir des estimations de l'avancement total et par discipline pour chaque projet.

1. Sécurité et gestion des accès :

* Offrir différents niveaux d'accès pour les employés et les administrateurs.
* Permettre aux administrateurs de gérer les utilisateurs, les projets et les paramètres du système.

1. Flexibilité et personnalisation :
   * Permettre la modification des paramètres du système, comme le nombre maximal de projets par employé (NPE).
   * Offrir la possibilité de personnaliser la liste des disciplines de travail.
2. Persistance des données :
   * Sauvegarder toutes les informations dans des fichiers texte au format JSON pour assurer la pérennité des données.
3. Interfaçage avec d'autres systèmes :
   * Préparer l'intégration future avec un système de paie en fournissant les données nécessaires au format requis.
4. Reporting :
   * Générer divers rapports, notamment sur l'état des projets, les salaires des employés et les coûts totaux pour l'entreprise.

# Exigences Fonctionnelles

Les exigences fonctionnelles définissent ce que le système doit faire. Voici les principales exigences fonctionnelles pour le système TimeLog :

* **Gestion des Projets**
  + L'admin doit pouvoir créer, modifier et supprimer des projets.
  + Chaque projet doit inclure un nom, un numéro d'identification, une date de début, une date de fin et des heures budgétées pour chaque discipline.
* **Gestion des Employés**
  + L'admin doit pouvoir créer, modifier et supprimer des employés.
  + Chaque employé doit inclure un nom, un numéro d'identification (ID), une date d'embauche, une date de départ (optionnelle), un numéro de sécurité sociale, un poste, un taux horaire de base et un taux horaire pour les heures supplémentaires.
* **Assignation des Employés aux Projets**
  + L'admin doit pouvoir assigner des employés à des projets spécifiques.
  + L'admin doit pouvoir définir le nombre maximum de projets (NPE) auxquels un employé peut être assigné.
* **Gestion des Disciplines**
  + L'admin doit pouvoir gérer (ajouter, modifier, supprimer) la liste des disciplines de travail.
  + Les disciplines par défaut sont : design1 (haut niveau), design2 (détaillé), implémentation, test et déploiement.
* **Authentification des Employés**
  + Les employés et admin doivent pouvoir se connecter au système en fournissant leur nom d'utilisateur et leur mot de passe.
  + L'admin doit pouvoir modifier les noms d'utilisateur et les ID des employés ainsi que les siens.
* **Gestion des Activités des Employés**
  + Les employés doivent pouvoir signaler le début et la fin d'une activité.
  + Une seule activité est permise à la fois par employé.
  + L'heure et la date de début et de fin de chaque activité doivent être enregistrées.
* **Consultation des Heures Travaillées**
  + Les employés doivent pouvoir consulter le nombre d'heures travaillées sur une période donnée.
* **Consultation des Talons de Paie**
  + Les employés doivent pouvoir consulter leur talon de paie pour les 26 dernières périodes de paie.
* **Génération de Rapports**
  + Le système doit pouvoir générer des rapports d'état pour chaque projet, indiquant le nombre d'heures travaillées par discipline et le pourcentage d'avancement.
  + Le système doit pouvoir générer un rapport global de l'état de tous les projets.
  + Le système doit pouvoir générer un rapport des valeurs (en salaire) des heures travaillées par employé pour une période donnée.
  + Le système doit pouvoir générer un rapport des salaires bruts et nets de tous les employés.
* **Persistance des Données**
  + Le système doit sauvegarder les informations sur des fichiers texte au format JSON.

# Exigences Non Fonctionnelles

Les exigences non fonctionnelles définissent les critères de qualité du système. Voici les principales exigences non fonctionnelles pour le système TimeLog :

* **Performance**
  + Le système doit être capable de gérer simultanément les connexions de multiples employés sans dégradation significative de la performance.
  + Le système doit répondre aux actions des utilisateurs en moins de 2 secondes.
* **Sécurité**
  + Les données des employés et des projets doivent être protégées contre les accès non autorisés.
  + Les mots de passe doivent être stockés de manière sécurisée.
* **Fiabilité**
  + Le système doit être disponible et opérationnel 99,9 % du temps.
  + Le système doit sauvegarder automatiquement les données toutes les heures pour éviter toute perte de données.
* **Maintenabilité**
  + Le code source du système doit être bien documenté pour faciliter la maintenance et les mises à jour.
  + Le système doit être modulaire pour permettre des modifications ou extensions futures sans impact majeur sur l'ensemble du système.
* **Utilisabilité**
  + L'interface en ligne de commande doit être intuitive et facile à utiliser, même pour les utilisateurs non techniques.
  + Les messages d'erreur doivent être clairs et informatifs pour aider les utilisateurs à résoudre les problèmes.
* **Compatibilité**
  + Le système doit être compatible avec Java 15.
  + Le système doit pouvoir être installé et fonctionner sur les principaux systèmes d'exploitation (Windows, MacOs, Linux).
* **Scalabilité**
  + Le système doit pouvoir évoluer pour gérer un nombre croissant d'employés et de projets sans dégradation des performances.

Ces exigences fonctionnelles et non fonctionnelles permettront de guider le développement et l'implémentation du système TimeLog pour répondre aux besoins de l'entreprise et des utilisateurs finaux.

# Description Complète des Cas d'Utilisation

* **Gérer Projets**
  + **Acteur** : Admin
  + **Description** : Créer, modifier et supprimer des projets.
  + **Actions** :
    - Créer un nouveau projet
    - Modifier un projet existant
    - Supprimer un projet
* **Gérer Employés**
  + **Acteur** : Admin
  + **Description** : Créer, modifier et supprimer des employés.
  + **Actions** :
    - Ajouter un nouvel employé
    - Modifier les informations d'un employé
    - Supprimer un employé
* **Assigner Employés aux Projets**
  + **Acteur** : Admin
  + **Description** : Assigner des employés à des projets spécifiques.
  + **Actions** :
    - Assigner un employé à un projet
    - Retirer un employé d'un projet
* **Modifier Paramètres Système**
  + **Acteur** : Admin
  + **Description** : Modifier les paramètres du système.
  + **Actions** :
    - Changer les paramètres du système
* **Se Connecter**
  + **Acteur** : Employé, Admin
  + **Description** : Se connecter au système.
  + **Actions** :
    - Entrer le nom d'utilisateur et mot de passe
    - Authentification de l'utilisateur
* **Débuter une Activité**
  + **Acteur** : Employé
  + **Description** : Signaler le début d'une activité.
  + **Actions** :
    - Sélectionner le projet
    - Choisir la discipline de travail
    - Enregistrer l'heure et la date de début
* **Terminer une Activité**
  + **Acteur** : Employé
  + **Description** : Signaler la fin d'une activité.
  + **Actions** :
    - Enregistrer l'heure et la date de fin
* **Consulter Heures Travaillées**
  + **Acteur** : Employé
  + **Description** : Consulter le nombre d'heures travaillées.
  + **Actions** :
    - Sélectionner la période
    - Afficher les heures travaillées
* **Consulter Talon de Paie**
  + **Acteur** : Employé, Admin
  + **Description** : Consulter son talon de paie.
  + **Actions** :
    - Afficher le talon de paie
* **Générer Rapport d'État de Projet**
  + **Acteur** : Employé, Admin
  + **Description** : Générer un rapport indiquant le nombre d'heures travaillées et l'avancement par discipline pour un projet spécifique.
  + **Actions** :
    - Sélectionner le projet
    - Afficher le rapport d'état
* **Générer Rapport Global**
  + **Acteur** : Employé, Admin
  + **Description** : Générer un rapport global de l'état de tous les projets.
  + **Actions** :
    - Afficher le rapport global
* **Générer Rapport des Salaires**
  + **Acteur** : Admin
  + **Description** : Générer un rapport des salaires bruts et nets de tous les employés.
  + **Actions** :
    - Afficher le rapport des salaires
* **Persister les Données**
  + **Acteur** : Admin
  + **Description** : Sauvegarder les informations sur des fichiers texte au format JSON.
  + **Actions** :
    - Enregistrer les données dans un fichier JSON
* **Gestion des Disciplines**
  + **Acteur** : Admin
  + **Description** : Gérer la liste des disciplines de travail.
  + **Actions** :
    - Ajouter, modifier ou supprimer des disciplines
* **Assigner NPE (Nombre de Projets Employé)**
  + **Acteur** : Admin
  + **Description** : Définir le nombre maximum de projets auxquels un employé peut être assigné.
  + **Actions** :
    - Définir la valeur de NPE
* **Générer Rapport Employé**
  + **Acteur** : Employé, Admin
  + **Description** : Générer un rapport des heures travaillées et des salaires pour une période donnée.
  + **Actions** :
    - Sélectionner la période
    - Afficher le rapport de l'employé

## Cas d'utilisation : Débuter une activité

**Acteur principal :**

* Employé

**Partie prenante et intérêts :**

* **Employé** : Veut enregistrer le début de son activité afin de suivre ses heures de travail.

**Pré-conditions :**

* L'employé est authentifié et connecté au système.
* L'employé est assigné à au moins un projet.

**Post-condition (succès garanti) :**

* Le début de l'activité est enregistré avec la date et l'heure actuelles.

**Post-condition (échec garanti) :**

* L'activité n'est pas enregistrée et un message d'erreur est affiché.

|  |  |
| --- | --- |
| Action des Acteurs | Réponse du système |
| 1)L'employé se connecte au système en fournissant son nom d'utilisateur et son mot de passe | 2)Système vérifie les informations de connexion et authentifie l'employé. |
| 4)L'employé sélectionne l'option "Débuter une activité" dans le menu. | 3)Le système afficher les options à employer |
| 6) L'employé sélectionne un projet | 5) Système affiche une liste des projets auxquels l'employé est assigné. |
| 8) L'employé sélectionne une discipline de travail | 7) Système affiche une liste des disciplines de travail (design1, design2, implémentation, test, déploiement). |
|  | 9) Système enregistre la date et l'heure actuelles comme heure de début de l'activité. |
|  | 10) Système affiche un message de confirmation indiquant que l'activité a débuté avec succès. |

## Scenario alternatif

8)a Aucune discipline sélectionnée

8)a 1. **L'employé ne sélectionne pas de discipline et tente de continuer.**

1.2 Système affiche un message d'erreur demandant de sélectionner une discipline.

8)a 2 L’employésélectionne une discipline de travail.

2.1 Retour à l'étape 8 du scénario principal.

## Cas d'utilisation : Terminer une activité

**Acteur principal :**

* Employé

**Partie prenante et intérêts :**

* **Employé** : Veut enregistrer la fin de son activité pour suivre ses heures de travail.

**Pré-conditions :**

* L'employé est authentifié et connecté au système.
* L'employé a une activité en cours.

**Post-condition (succès garanti) :**

* La fin de l'activité est enregistrée avec la date et l'heure actuelles.

**Post-condition (échec garanti) :**

* L'activité n'est pas enregistrée et un message d'erreur est affiché.

|  |  |
| --- | --- |
| Action des Acteurs | Réponse du système |
| 1)L'employé sélectionne l'option "Terminer une activité" dans le menu. | 2) Le système affiche les informations de l'activité en cours. |
| 4)L'employé confirme la fin de l'activité. | 3)Le système offre une option permettant à l’employé de confirmer la fin de son activité |
|  | 5) Le système enregistre la date et l'heure actuelles comme heure de fin de l'activité. |
|  | 6) Le système affiche un message de confirmation indiquant que l'activité a été terminée avec succès. |
|  |  |
|  |  |
|  |  |

## Scenario alternatif

1 a : Aucune activité en cours

1a1. L’employé tente de terminer une activité alors qu’il n’y a pas d’activité en cours.

2. Le système affiche un message d'erreur indiquant qu'aucune activité n'est en cours.

1a2 L’employé retourne au menu principal

# Conclusion

Cette première phase est très importante pour le développement du système TimeLog car elle pose les bases essentielles de l’ensemble du projet. Les fonctionnalités majeures et les interactions entre les différents utilisateurs et ce système, ont été clairement définies. Cette phase nous permet d’avoir une vue d’ensemble de sa structure, ainsi que des besoins et des exigences fonctionnelles et non fonctionnelles. La réussite de cette phase d’analyse permettra d’aborder les phases de conception et d’implémentation dans de meilleures conditions, qui permettront que le système TimeLog réponde efficacement aux besoins pour lesquels il sera mis en place.

# Annexes

Une image contenant texte, capture d’écran, diagramme, Police

Description générée automatiquement