

Rapport de Projet

Hackathon Manager

Application Java de Gestion de Hackathons

Présenté par :

Chahine Ikram

Karboubi Sami

Mekyassi Malak

El-Hamdaoui Marouane

Sous la direction de :

M. Zakrani

École Nationale Supérieure d'Arts et Métiers

Département Informatique

IAGI, CI-1

2024-2025

Remerciements

Nous tenons à exprimer notre sincère gratitude envers notre professeur d'UML pour son engagement exceptionnel et son dévouement dans la transmission des connaissances tout au long de ce semestre. Sa pédagogie claire et structurée nous a permis d'appréhender avec aisance les concepts complexes de la modélisation orientée objet.

Nous sommes particulièrement reconnaissants pour sa patience et sa disponibilité constante lors des séances de travaux pratiques, ainsi que pour ses conseils précieux qui ont significativement contribué à la réalisation de ce projet. Son expertise et sa passion pour l'enseignement de l'UML ont été une source d'inspiration et ont grandement facilité notre compréhension des diagrammes et des principes de modélisation.

Les méthodes d'apprentissage interactives qu'il a mises en place ont créé un environnement propice à l'assimilation des connaissances et au développement de nos compétences en conception logicielle. Sa capacité à illustrer les concepts théoriques par des exemples concrets a rendu la matière plus accessible et applicable dans un contexte professionnel.

Nous souhaitons également souligner l'impact positif de son encadrement rigoureux et bienveillant sur notre progression, et lui témoigner notre profonde reconnaissance pour nous avoir guidés avec excellence dans l'univers de la modélisation UML.

Résumé

Ce rapport présente le développement d'une application Java console nommée "Hackathon Manager", conçue pour faciliter l'organisation, la gestion et l'évaluation de hackathons. L'application repose sur une architecture orientée objet rigoureuse et intègre une intelligence artificielle via l'API Ollama pour automatiser l'évaluation des projets. Le système assure la persistance des données via des fichiers texte structurés et offre une interface en ligne de commande intuitive. Cette solution propose une approche centralisée pour gérer les participants, jurys, projets, équipes et évaluations, tout en maintenant une architecture modulaire et évolutive.

Mots-clés : Java, UML, Intelligence Artificielle, Gestion de Hackathon, Architecture Orientée Objet, API Ollama

1 Introduction

Le projet "Hackathon Manager" est une application Java en mode console visant à faciliter l'organisation, la gestion et l'évaluation de hackathons. Elle repose sur une architecture orientée objet rigoureuse et une persistance basée sur des fichiers textes. L'application intègre également une intelligence artificielle via une API (Ollama) pour générer automatiquement des évaluations objectives et structurées.

Dans ce rapport, nous détaillons les différents aspects du projet : les cas d'utilisation principaux, les modèles UML (cas d'utilisation, séquence, activité, états, packages), ainsi que les objectifs, les fonctionnalités et les perspectives d'évolution.

2 Objectifs du projet

- Proposer une solution informatique centralisée pour gérer les entités d'un hackathon : participants, jurys, projets, équipes et évaluations.
- Automatiser l'évaluation des projets grâce à l'intégration d'un module IA.
- Assurer la persistance des données dans un format simple et exploitable (fichiers textes).
- Offrir une interface en ligne de commande intuitive pour la gestion complète de l'événement.

3 Cas d'utilisation principal

Titre : Gérer une équipe et son projet

Acteurs principaux :

- Utilisateur (organisateur)
- Système Hackathon Manager

Scénario nominal :

1. L'utilisateur saisit le nom de l'équipe.
2. Il indique le nombre de participants (entre 2 et 3).
3. Pour chaque participant, il saisit les informations : nom, email, rôle.
4. L'utilisateur entre le titre et la description du projet associé.
5. Le système crée l'équipe, y affecte les participants, enregistre le projet et les relie.
6. Confirmation affichée.

Scénario alternatif :

- Si un email de participant existe déjà, un message d'erreur est affiché et une nouvelle saisie est demandée.
- Si le projet existe déjà, une nouvelle saisie est exigée.

Scénario d'échec :

- Si le nombre de participants est invalide (<2 ou >3), un message bloque la création.
- Si les informations saisies sont vides ou mal formatées, l'action échoue et est relancée.

4 Diagrammes UML

4.1 Diagramme de Cas d'Utilisation

Un acteur principal "Organisateur" interagit avec le système Hackathon Manager pour :

- Ajouter un participant
- Créer une équipe
- Affecter des participants
- Créer un projet
- Associer un projet à une équipe
- Évaluer un projet
- Générer un rapport

4.2 Diagrammes de Séquence

a) Création d'une équipe avec projet (acteur : Organisateur)

- L'organisateur saisit les données.
- Le système vérifie la validité.
- Le système crée les entités et les relie.

b) Évaluation d'un projet (acteur : Organisateur, secondaire : IA via API)

- L'organisateur choisit un projet.
- Le système envoie une requête à l'IA.
- L'IA retourne une évaluation.
- Le système sauvegarde l'évaluation et l'affiche.

4.3 Diagramme d'État-Transition (Projet)

États du projet :

1. Créé
2. Affecté à une équipe
3. Évalué (par jury ou IA)
4. Inclus dans le rapport final

Transitions déclenchées par : affectation, évaluation, génération de rapport.

4.4 Diagramme d'Activité (Création d'une équipe avec projet)

Début → Saisie équipe → Saisie participants → Vérification → Saisie projet → Association → Confirmation → Fin

4.5 Diagramme de Séquence (Boîte blanche – méthode associer-ProjetEquipe)

- Appel : `associerProjetEquipe(projet, equipe)`
- Contrôle : vérifier qu’aucun projet n’est déjà associé
- Action : `projet.setEquipe(equipe)`, `equipe.setProjet(projet)`

4.6 Diagramme de Package

- **model** : contient les entités (Hackathon, Projet, Participant, Jury...)
- **service** : logique métier (validation, IA, persistance)
- **exception** : classes d’exception personnalisées
- **utils** : outils génériques (ConsoleUtils)

5 Fonctionnalités principales

- Ajout/suppression de participants, jurys, équipes et projets
- Affectation d’un projet à une équipe
- Évaluation d’un projet par un jury ou par l’IA
- Génération d’un rapport final avec classement
- Persistance des données (fichiers texte structurés)

6 Persistance des données

Le système utilise des fichiers `.txt` pour enregistrer toutes les entités. Chaque entité est sauvegardée dans un format délimité pour faciliter la lecture et le rechargement. Des stratégies robustes (try/catch, validateurs) permettent de gérer les erreurs et éviter les corruptions.

7 Intégration de l’IA

L’intégration de l’intelligence artificielle repose sur l’utilisation d’Ollama, via une requête POST contenant un prompt structuré. L’IA retourne une évaluation textuelle contenant une note globale et un commentaire. Cette réponse est parsée et stockée comme une évaluation standard.

8 Test et validation

Des cas de test manuels ont été réalisés via l’interface console. Les principaux scénarios d’utilisation ont été couverts :

- Création d’équipe
- Affectation de projet
- Évaluation IA et jury

— Génération du rapport

Les fichiers générés ont été vérifiés manuellement pour s'assurer de la persistance correcte des données.

9 Conclusion et perspectives

Le projet Hackathon Manager propose une solution robuste et modulaire pour gérer les hackathons de bout en bout. L'architecture orientée objet assure une évolutivité, tandis que l'interface console reste simple à utiliser.

Perspectives :

- Développement d'une interface graphique (JavaFX ou Web)
- Ajout de fonctionnalités IA plus poussées (suggestion d'équipe, auto-complétion...)
- Passage à une base de données pour plus de robustesse
- Déploiement sur un environnement cloud ou intranet

A Annexes

A.1 Glossaire

Hackathon Événement où des développeurs, designers et autres experts collaborent intensivement sur des projets pendant une période limitée.

Équipe Groupe de 2 à 3 participants travaillant ensemble sur un projet.

Jury Personne chargée d'évaluer les projets selon des critères définis.

Évaluation Note et commentaire attribués à un projet par un jury ou l'IA.

Ollama Plateforme d'intelligence artificielle utilisée pour l'évaluation automatisée.

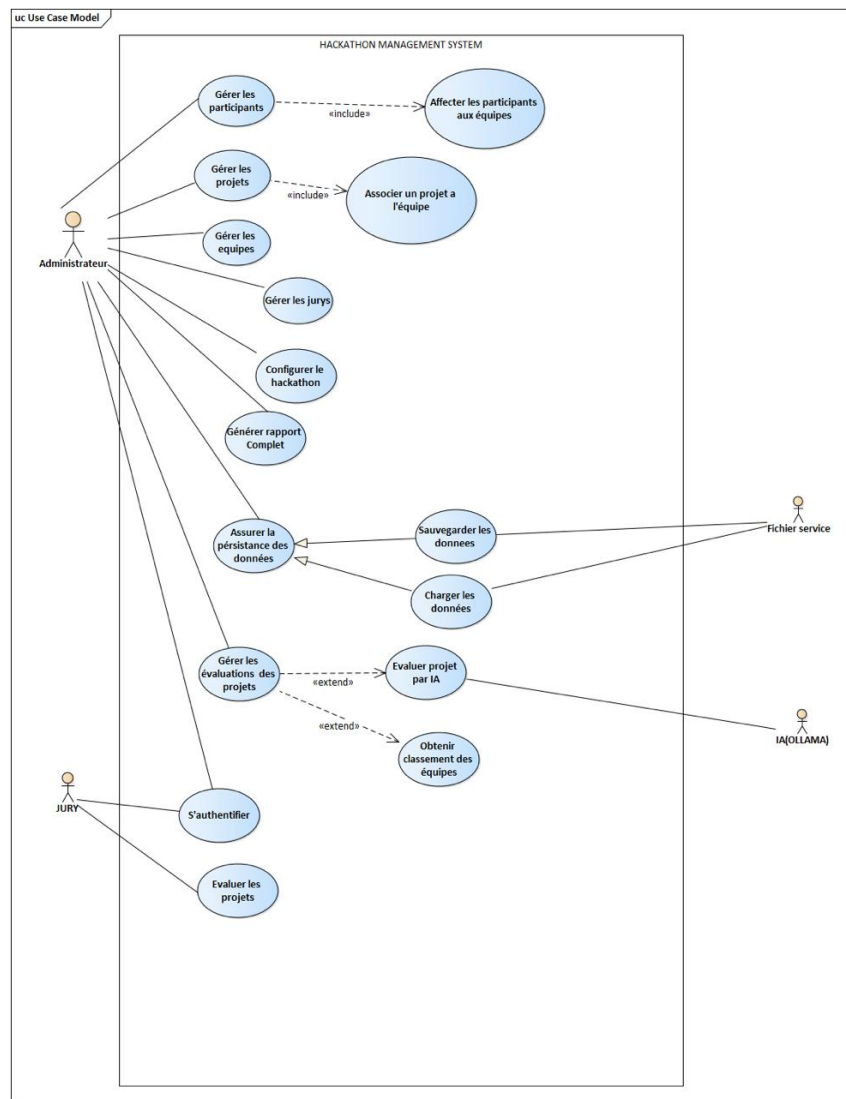


FIGURE 1 – Diagramme de cas d'utilisation du système

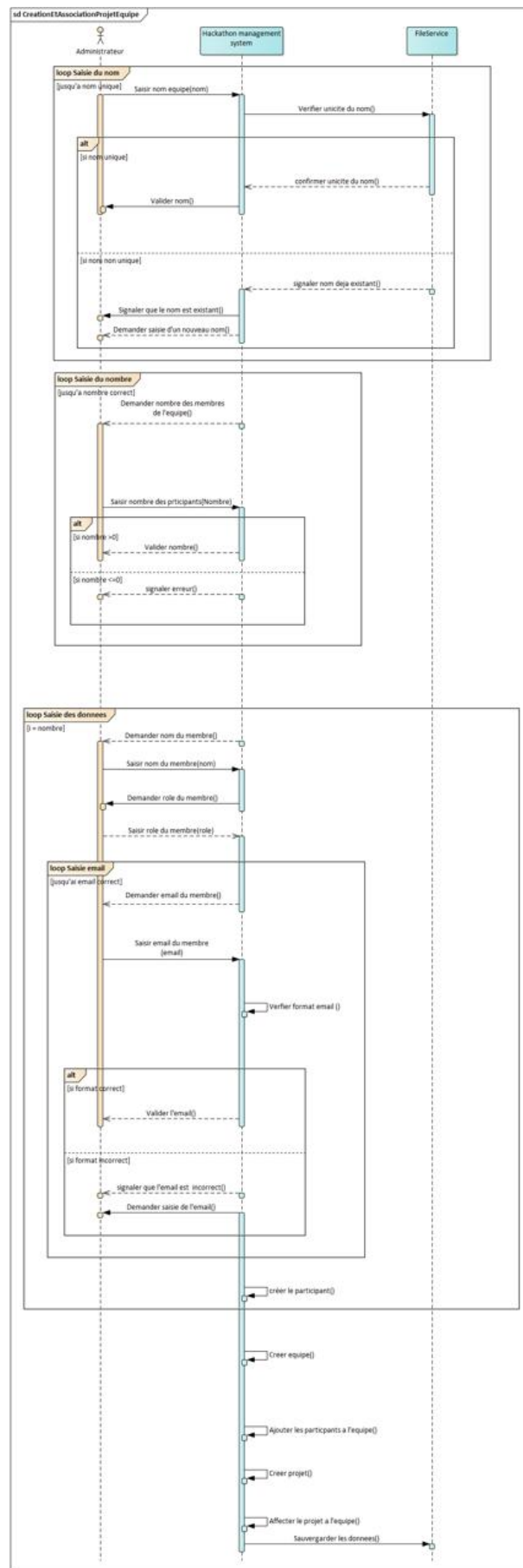


FIGURE 2 – Diagramme de séquence : Création d'une équipe avec projet

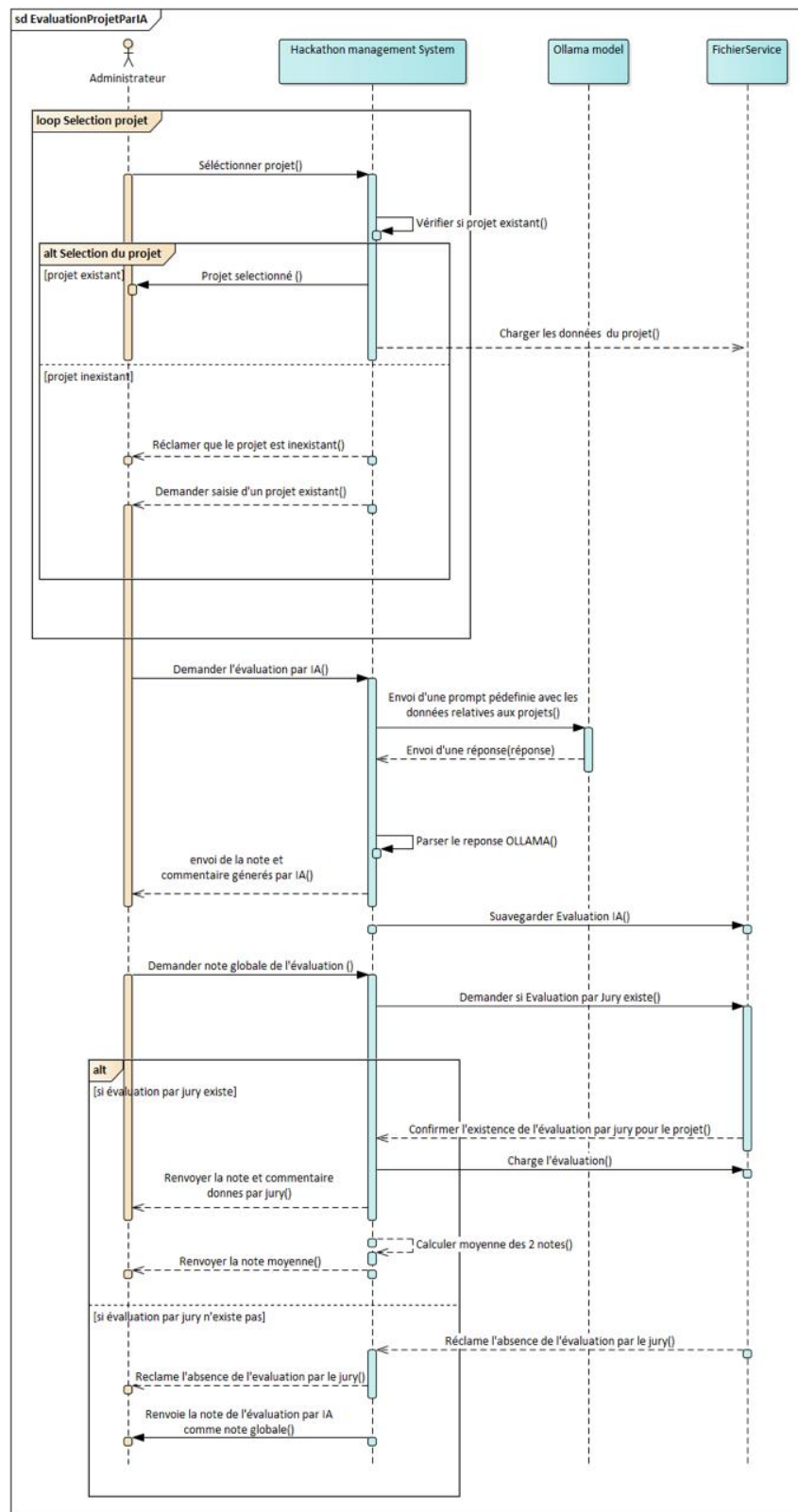


FIGURE 3 – Diagramme de séquence : Évaluation d'un projet par IA

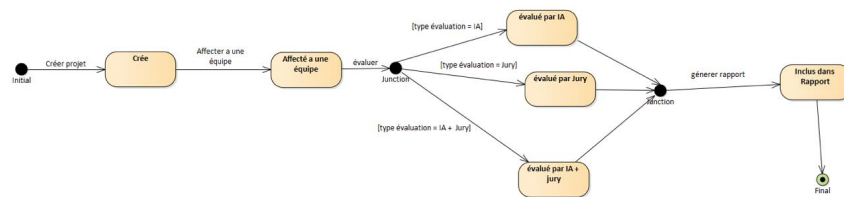


FIGURE 4 – Diagramme d'état-transition du cycle de vie d'un projet

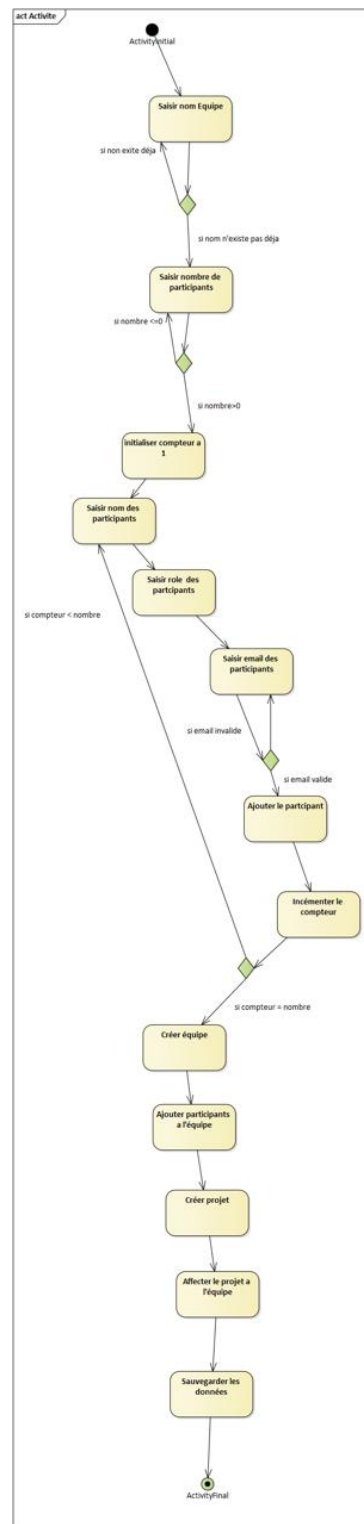


FIGURE 5 – Diagramme d'activité : Création d'une équipe avec projet

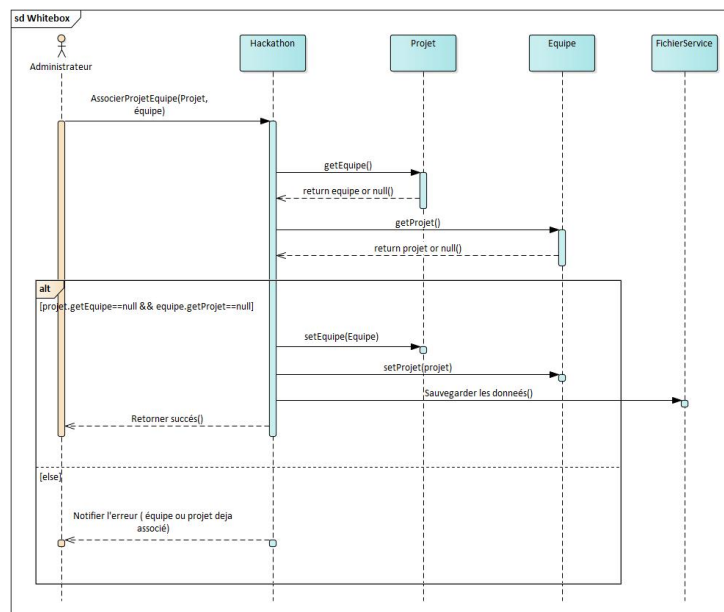


FIGURE 6 – Diagramme de séquence (boîte blanche) : méthode associerProjetEquipe

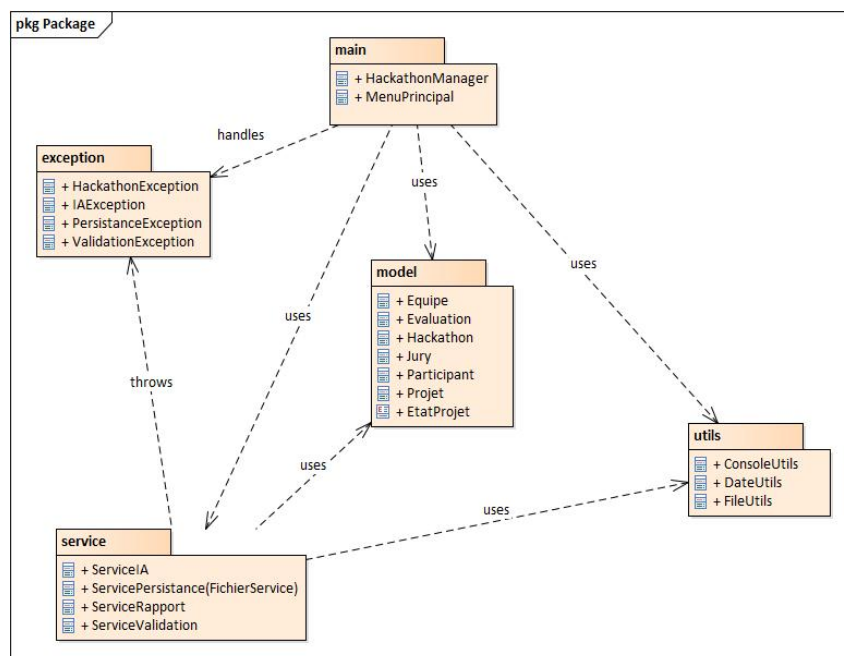


FIGURE 7 – Diagramme de packages du système