

# Rapport Individuel – Projet de Jeu RPG 2D (Projet long TOB)

ZENBI Othmane

Mai 2025

## Contexte

Cette partie vise à créer un système d'inventaire pour le jeu vidéo. J'ai principalement développé les classes `Item`, `Weapon` et `Grenade`, tandis que Khalil s'est chargé des autres éléments (`HealItem`, `Inventory`, etc.).

*Version primaire présentée à l'équipe pour anticiper des modifications.*

## Classes développées

1. **Item (abstraite)** : Base de tous les objets de l'inventaire (`id`, `name`, `rarity`). Fournit les getters et `toString()`. Conçue pour l'héritage et la standardisation via une énumération de rareté.

2. **Weapon (hérite de Item)** : Modélise les armes avec gestion des munitions (`damage`, `maxAmmo`, `currentAmmo`) et accessoires (`attachments`). Méthodes principales : `fire()`, `reload()`, `addAttachment()`.

3. **Grenade (hérite de Item)** : Représente des grenades à usage unique (`damage`, `radius`). Structure simple pour une utilisation directe.

## Choix techniques

- **Héritage** : Évite la redondance et facilite l'extension.
- **Encapsulation** : Attributs privés, accès via getters.
- **Extensibilité** : Ajout d'accessoires sans modifier la classe `Weapon`.

## Perspectives d'amélioration

- Ajouter des types de munitions et une durabilité aux armes.
- Effets spéciaux pour les grenades.
- Optimiser avec des `Streams` Java et ajouter la sauvegarde de l'inventaire.

## Conclusion

Cette version de base, conforme aux bonnes pratiques POO, servira de support pour les retours et l'intégration avec les modules de Khalil. Prochaines étapes : revue de code, tests unitaires, adaptation aux besoins du projet.

*Suggestions et retours sont les bienvenus pour améliorer la flexibilité et la performance.*