

Examen de Fin de Quadrimestre: L'Académie des Super-Héros vs Les Super-Vilains Profs

Contexte du Cas:

Au cœur de l'Académie des Super-Héros, un lieu où les étudiants sont formés pour devenir la prochaine génération de protecteurs de l'humanité, une bataille se prépare. Une série de missions critiques (examens) est sur le point de commencer, opposant les étudiants super-héros aux professeurs devenus super-vilains pour l'occasion. En tant qu'étudiant et développeur en chef au sein de l'académie, votre mission est de créer un système de gestion permettant d'organiser les affrontements, de cataloguer les compétences et de planifier stratégiquement les interventions de chaque super-héros.

Objectif de l'Examen:

Développez un système de gestion en Python qui modélise la dynamique entre super-héros et super-vilains profs, en utilisant des concepts clés de la programmation orientée objet pour représenter les caractéristiques uniques et les pouvoirs des participants, ainsi que le déroulement des missions.

Instructions Détaillées:

Partie 1: Conception UML (40 points)

1. Diagramme de Classes (20 points) :

- **Classe SuperHero** : Doit inclure des attributs pour le nom, une liste de **SuperPower**, et un historique des missions accomplis. Méthodes pour ajouter des pouvoirs, participer à des missions, et afficher les informations du héros.
- **Classe SuperPower** : Représente les pouvoirs uniques des héros, avec des attributs pour le nom du pouvoir, la description, et le niveau de puissance.
- **Classe SuperVillainProf** : Similaire à **SuperHero**, mais pour les professeurs, incluant des méthodes pour défier les héros dans des missions spécifiques.
- **Classe Mission** : Contient des informations sur les défis à relever, les participants (héros vs. vilain), et le résultat de l'affrontement.

2. Diagramme de Cas d'Utilisation (20 points) :

- Illustrer les interactions clés : "Planifier une mission", où les super-héros choisissent leurs défis ; "Entraîner un super-pouvoir", pour améliorer leurs compétences ; et "Affronter un super-vilain prof", le cœur de l'action académique.

Diagramme de séquences(20 points) :

- illustrer les interactions clés entre les super-héros, les super-vilains professeurs, et le processus de planification des missions à l'Académie des Super-Héros.

Partie 2: Développement en Python si les diagrammes sont justes (60 points)

Fondé sur la conception UML, implémentez le système avec les détails suivants :

1. Polymorphisme :

- Notamment avec la méthode **utiliser_pouvoir()**.

2. Composition :

- Modélisation de la relation entre **SuperHero** et **SuperPower**.

3. Encapsulation :

- Protection des attributs et méthodes de chaque classe.

Scénario à Simuler :

- Les super-héros planifient et exécutent plusieurs missions, affrontant les super-vilains profs dans une série d'épreuves conçues pour tester leurs limites et leur ingéniosité.