

Conclusions de l'implémentation de la Team Rocket

L'implémentation de IPokemonFactory fournie par la Team Rocket suit les principes de création d'un Pokémon avec leurs caractéristiques comme l'index, le nom, les statistiques ... Les valeurs des statistiques sont générées aléatoirement dans un intervalle, ce qui permet de bien diversifier la création des Pokémon. Cependant, cette implémentation comporte aussi quelques points faibles comme lorsque l'index fourni ne correspond à aucun Pokémon valide, le nom du Pokémon par défaut pourrait ne pas convenir.

Comparaison avec mon implémentation

Tandis que la Team Rocket génère les valeurs aléatoires, l'implémentation que j'ai développée s'appuie davantage sur les métadonnées des Pokémon, ce qui garantit une cohérence entre les Pokémon de même type. De plus, la valeur d'IV est calculée en fonction des autres données.

Une autre différence est le cas où les paramètres ne sont pas valides. Lorsque l'index n'existe pas dans la table, la Team Rocket retourne un nom par défaut et fixe les valeurs attaque, défense et stamina à 100, ce qui peut conduire à des incohérences. Dans mon implémentation, j'avais alors choisis de lever une exception pour signaler l'erreur.

Tests ajoutés

La logique d'implémentation étant différente, j'ai dû modifier mes tests existants. Ensuite, afin de couvrir les défauts non pris en compte, j'ai ajouté les tests suivants :

- Test des index invalides : Vérifier que l'implémentation retourne un Pokémon avec des valeurs par défaut appropriées lorsque l'index est invalide.
- Test de la génération aléatoire des statistiques : Vérifier que les statistiques générées sont toujours dans des intervalles valides (0-100).