

Projet POO – CitySim : La ville évolutive

Un consortium de développeurs urbains souhaite créer un simulateur de ville intelligent pour évaluer la gestion de nouvelles zones urbaines. Vous êtes chargés de concevoir ce simulateur, qui doit reproduire l'évolution d'une ville à travers ses bâtiments, ses habitants, ses ressources et les événements qui peuvent s'y produire. Le simulateur fonctionnera via un menu interactif en console, avec des actions permettant de faire évoluer la ville jour après jour.

Structure générale attendue

- 1. La Ville

- Contient un nom, une population totale, un niveau de bonheur (0–100), une liste de bâtiments
- Gère l'ajout de bâtiments et l'évolution de la ville au fil du temps

- 2. Les Bâtiments

- Classe abstraite Batiment avec : nom, capacité, état (100% au départ), type (habitation, production, loisir...)
- Héritages obligatoires : Maison (augmente la capacité d'habitants), Usine (produit des ressources mais diminue le bonheur), Parc (augmente le bonheur)
- Tous les bâtiments peuvent être endommagés ou réparés

- 3. Les Habitants

- Ont un nom, un âge, une profession, un niveau de satisfaction
- Sont répartis dans les bâtiments selon la capacité
- Leur satisfaction dépend du bonheur global de la ville et de l'accès aux ressources

- 4. Les Ressources

- Eau, électricité, nourriture : consommées chaque jour
- Si la ville est en pénurie → baisse de satisfaction + effets en cascade

- 5. Les Événements

- Incendie, inondation, grève, fête locale...
- Peuvent dégrader des bâtiments ou influencer bonheur/population
- Devraient être simulés aléatoirement à chaque "jour"

- 6. Simulation & Interface

- Un menu interactif permet à l'utilisateur de :
 - Créer une ville
 - Ajouter des bâtiments
 - Répartir les habitants
 - Simuler le passage d'une journée
 - Afficher les logs quotidiens (événements, dégâts, consommation...)

Organisation en groupe

Chaque membre du groupe peut se concentrer sur un aspect :

- Architecte POO : Structure les classes, assure la cohérence
- Gestionnaire d'habitants : Implémente la classe Habitant, leur satisfaction
- Responsable bâtiments : Implémente les bâtiments et leur logique
- Gestionnaire d'événements : Gère les effets aléatoires, les logs

Livrables attendus

- Code source propre et commenté (au moins 5 classes)
- Menu interactif fonctionnel
- Fichier de simulation (log) généré
- UML général du projet (papier, image ou .drawio)