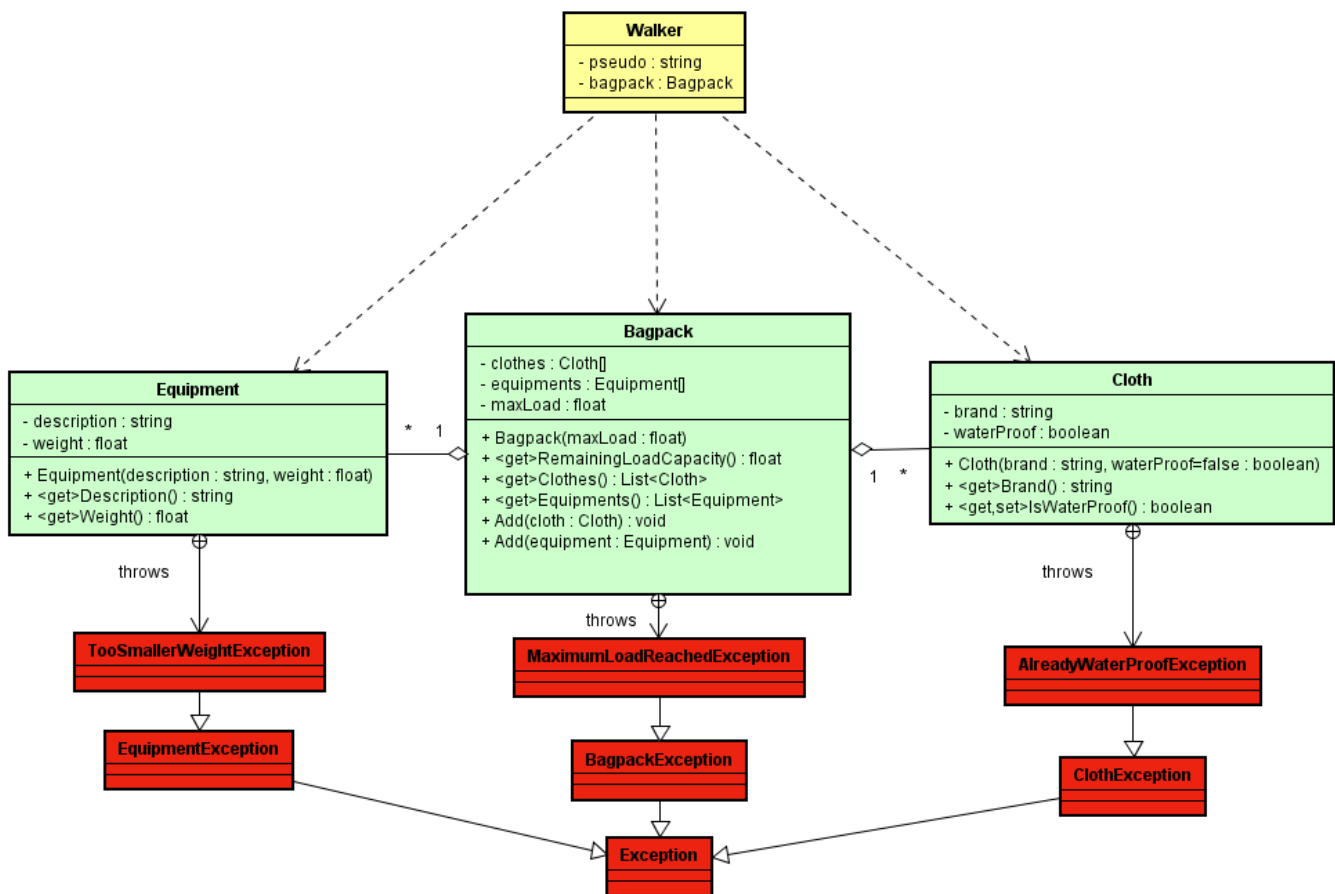


# 226a - Walking In The Wild

## Sujet à traiter

Vous êtes responsable de coder des librairies qui serviront une équipe développant un simulateur de marcheurs à pied.

## Architecture



## Explications sur l'architecture du code

- La classe **Walker.cs** est pour l'instant inutile. Ne vous en préoccupez pas.
- La classe **Backpack.cs** est conçue pour se comporter comme un sac à dos qui pourra intégrer aussi bien du matériel en tout genre (gourde, couverture, jumelles,...) que des vêtements.
- Les classes **Equipment** et **Cloth** quant à elles simulent des équipements et les habits à charger dans le sac à dos.

## Aides métier

1. On ne peut actuellement qu'ajouter des éléments au sac à dos. Retirer des éléments n'est pas attendu.
2. L'ajout d'un équipement dans le sac à dos, réduit progressivement sa capacité de charge.

3. L'ajout de vêtement est pour l'instant illimité.
4. Prenez garde aux différentes exceptions à traiter.
  - Un équipement doit avoir un poids positif.
  - Un sac à dos à une charge limite à ne pas dépasser.
  - Un vêtement peut être traité pour être résistant à la pluie. Inutile de le faire sur un vêtement qui l'est déjà.

## Prise en main du code

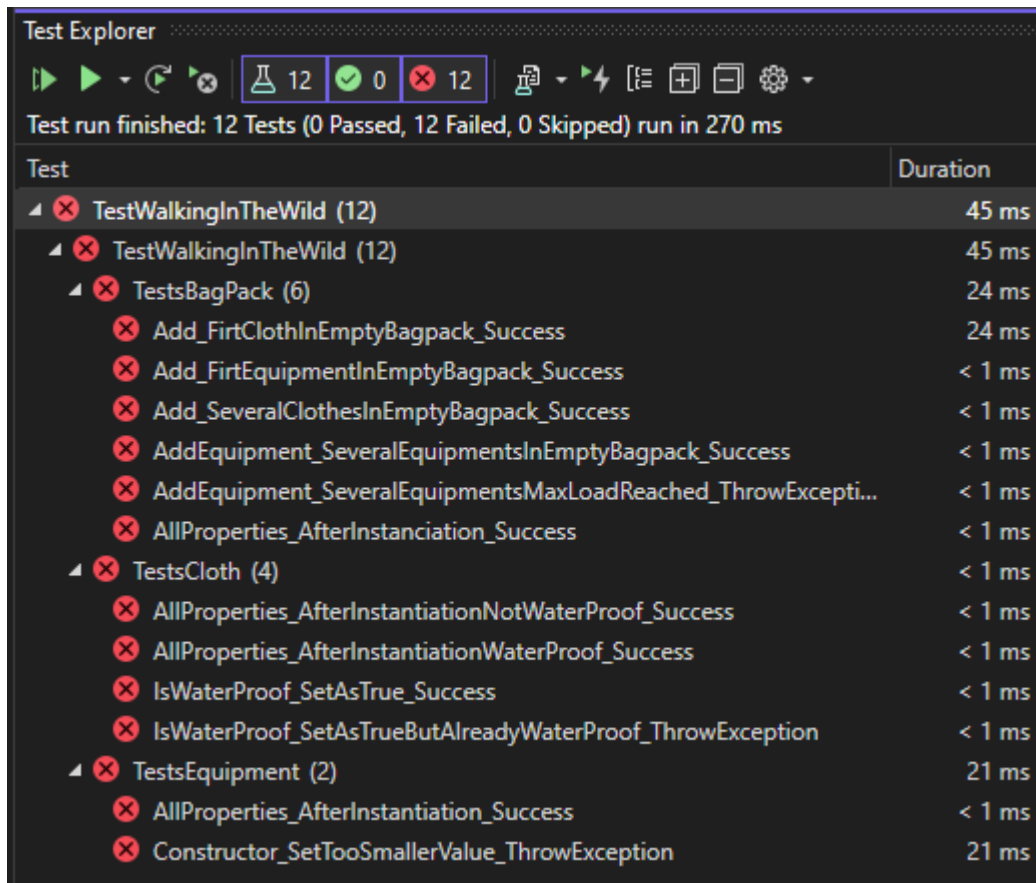
1. Récupérer le dépôt

```
git clone https://NGY_CPNV@bitbucket.org/cpnv_ngy/226a_walkinginthewild.git
```

2. Réaliser la première compilation. Il est important que le réseau soit disponible. Le gestionnaire de dépendences (Nguet) en a besoin pour fonctionner correctement.

```
2>----- Build started: Project: TestWalkingInTheWild, Configuration: Debug Any
CPU -----
2>[...[]]
2>Done building project "TestWalkingInTheWild.csproj".
===== Build: 2 succeeded, 0 failed, 0 up-to-date, 0 skipped =====
```

3. Lancer ensuite tous les tests du projet



Test	Duration
TestWalkingInTheWild (12)	45 ms
TestWalkingInTheWild (12)	45 ms
TestsBagPack (6)	24 ms
Add_FirtClothInEmptyBagpack_Success	24 ms
Add_FirtEquipmentInEmptyBagpack_Success	< 1 ms
Add_SeveralClothesInEmptyBagpack_Success	< 1 ms
AddEquipment_SeveralEquipmentsInEmptyBagpack_Success	< 1 ms
AddEquipment_SeveralEquipmentsMaxLoadReached_ThrowExcepti...	< 1 ms
AllProperties_AfterInstanciacion_Success	< 1 ms
TestsCloth (4)	< 1 ms
AllProperties_AfterInstantiationNotWaterProof_Success	< 1 ms
AllProperties_AfterInstantiationWaterProof_Success	< 1 ms
IsWaterProof_SetAsTrue_Success	< 1 ms
IsWaterProof_SetAsTrueButAlreadyWaterProof_ThrowException	< 1 ms
TestsEquipment (2)	21 ms
AllProperties_AfterInstantiation_Success	< 1 ms
Constructor_SetTooSmallerValue_ThrowException	21 ms

## Moyens à disposition

- un environnement de développement Visual Studio 2022 Entreprise
- tout le code produit durant le cours
- le réseau est exclu (excepté pour la première compilation) tout comme le travail collaboratif
- 45 minutes

## Livrables attendus

Une archive compressée contenant **uniquement** les trois classes métiers([nomDeFamille\_TDD].zip).

Modalités de livraison spécifique sont à définir avec votre animateur/trice.

## Evaluation

1pt par test d'acceptation réussi.

12pts au total.

Conditions d'acceptations:

- Respect des principes POO.
- Les classes de tests ne doivent avoir subi aucune modification.
- L'architecture des classes productives ne doit pas avoir été modifiée (signature des fonctions)

## Notation

$$\text{Note} = (\text{nbPointsObtenus} / \text{nbPointsMax}) / 5 + 1$$

## Auteur

Nicolas Glassey

06-NOV-2022