

## UE : 3I012

### TME Solitaire Mars 2019

**Durée : 45 Minutes**

**Ce TP doit être réalisé seul et dans un délai de 45 minutes.**

Lorsque vous aurez terminé ce TP:

1. générez la documentation Web de votre modèle
  - a. saisie d'un <nom> de la documentation
  - b. choix de l'emplacement, de génération :
    - i. par défaut dans <Projet Courant>/mda/WebModelPublisher/
    - ii. la page principale est <nom>.html, ses données sont dans le répertoire <nom>
2. créez une archive de la documentation (qui contiendra la page principale et le répertoire de données)
3. envoyez cette archive à l'adresse suivante : [Tewfik.Ziadi@lip6.fr](mailto:Tewfik.Ziadi@lip6.fr)

#### **Démarrage**

- 1) Créez un répertoire de travail nommé <VotreNom>\_<VotreNuméroEtu> dans lequel vous allez :
  - Créer un nouveau projet Modelio nommé <VotreNom>\_<VotreNuméroEtu>\_TPSolitaire

#### **Reverse-Engineering**

- 1) Faites le reverse Engineering du code source que vous avez téléchargé (les classes se trouvent dans le répertoire sources).
- 2) Construisez un diagramme total (tous les packages, toutes les classes, toutes les opérations, toutes les propriétés, toutes les associations, tous les liens d'héritage doivent apparaître).
- 3) Construisez un diagramme par classe (la classe, ses opérations, ses attributs, et les autres classes avec lesquelles elle est liée doivent apparaître) pour deux classes de votre choix.

#### **Retro-conception et Design Pattern**

- 4) Nous souhaitons mettre les classes *Mission* et *Robot*, *Capteur* et *Moteur* dans un package robots et le reste de classes dans un package comportement. Réalisez ce découpage en faisant attention aux règles vues en cours.
- 5) Dans un nouveau package IHM que vous créez, construisez une classe *Accueil*. Ajoutez une association navigable de la classe *Accueil* vers la classe *Mission* du package robots. Ajoutez une opération *afficheMessage(bi message : string)* à la classe *Accueil*.
- 6) On veut afficher un message de notification (en appelant l'opération *afficheMessageQ* de la classe *Accueil*) à chaque fois qu'il y a l'exécution d'une de la mission (cf. l'opération *executerMissionQ* dans la classe *Mission*). Appliquez le design pattern Observer pour réaliser cette fonctionnalité.
- 7) Créez un diagramme de séquence montrant l'enregistrement de l'observer et la notification.