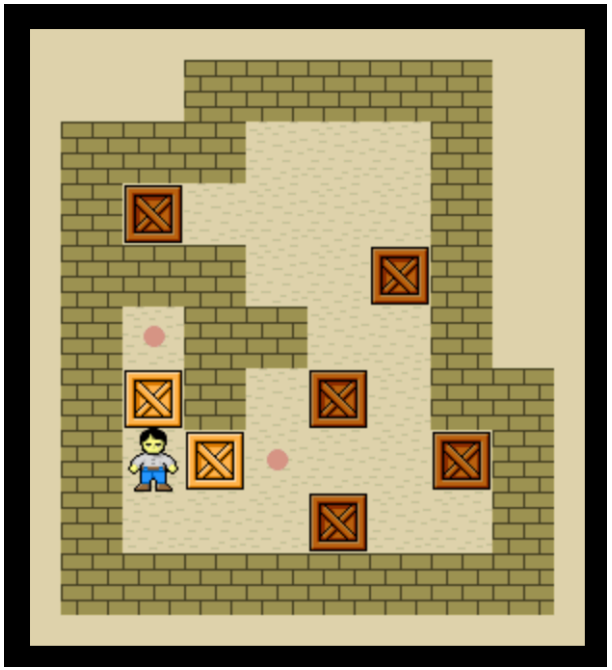
Projet Sokoban



Le sujet

Réaliser un programme de Sokoban en java. Ce programme devrait permettre de jouer, de calculer une solution (optionnel), de sélectionner un tableau de jeu et de charger des ensembles de tableaux au format xsb (XSokoban format).

1. Informations visualisées :
    - Le nom et l'auteur du tableau,
    - Le nombre de poussées et déplacements de Soko.
2. Liste des commandes permettant de jouer :
    - Commandes de déplacements de Soko (utiliser les flèches)
    - Commandes défaire et répéter un déplacement (undo/redo)
    - Commande rejouer : active une animation de tous les déplacements de Soko.
3. Commande de résolution du tableau en cours (optionnel).
4. Liste des commandes de sélection d'un tableau:
    - Commandes de sélection du tableau suivant et précédant,
    - Commande de chargement d'un ensemble de tableaux au format xsb.

Quoi et quand rendre

1. Quand ? le lundi 17 décembre 2018
2. Envoyer par email à votre chargé de TP, votre application sokoban.jar. Votre fichier jar doit être exécutable et doit contenir les sources et les ressources (fichier xsb et images) de votre application.
3. Donner à votre chargé de TP en version papier votre rapport.

Plan du rapport

1. Page de titre avec noms des participants et nom du chargé de TP
2. Travail réalisé : Lister les fonctionnalités réalisées, celles non réalisées et les bugs non corrigés.
3. Diagramme de conception : un ou plusieurs diagrammes de classes de votre application commentés.
4. Patron de conception : Lister les patrons de conception utilisés et commenter leur utilisation et intérêt pour votre application.

Difficultés du projet

Programmation Java

Technologique
JavaFX

Algorithmique

Génie logiciel
Comment démarrer et finir le projet

Conception
Patron de conception

Difficultés du projet

Programmation Java

Faible

Technologique
JavaFX

Nouveauté

Algorithmique

Faible

Génie logiciel
Comment démarrer et finir le projet

Toujours difficile

Conception
Patron de conception

Objet du cours

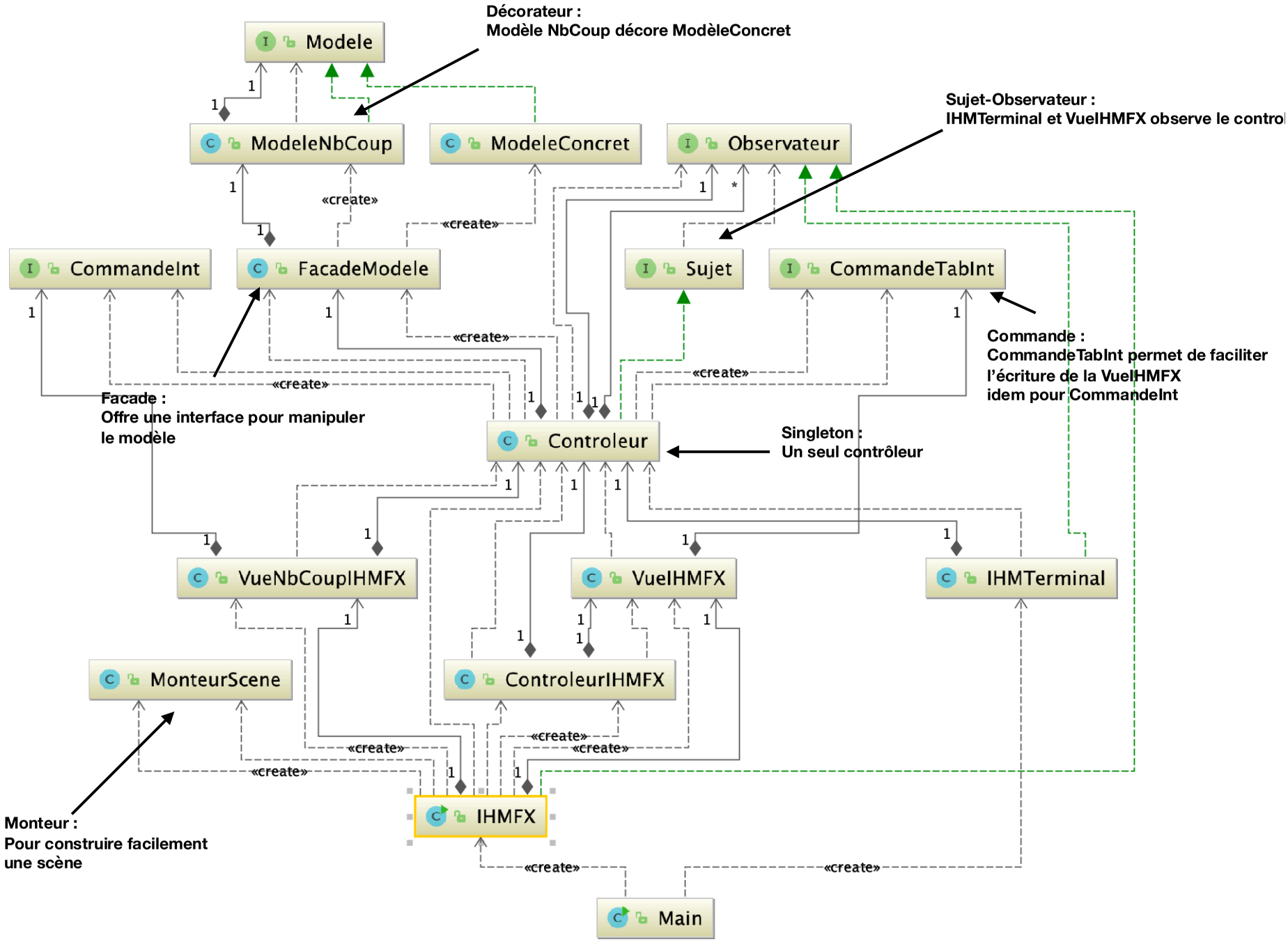
Technologie JavaFX

- Voir les exemples du cours
- Vous avez droit de reprendre le code du cours (exemple le monteur)
- Demander des exemples de codes
- Eventuellement lire un tuto :

http://tutorials.jenkov.com/javafx/index.html

Conception

- L’essentiel de la conception est basé sur Chameaux6.1
- **Attention** : pour le undo-redo, il faut être capable de savoir si un mouvement déplace une casse. Astuce !!! La méthode move peut renvoyer cette information par exemple sous la forme d’un entier : -1=pas de mouvement, 0 mouvement sans caisse et 1 avec caisse
- **Nouveauté : Gérer plusieurs états du jeu (voir exemple de code)**


Décorateur :
Modèle NbCoup décore ModèleConcret
Modele
ModeleNbCoup
ModeleConcret
Observateur
Sujet-Observateur :
IHMTerminal et VueIHMFX observe le contro
CommandeInt
FacadeModele
Sujet
CommandeTabInt
Commande :
CommandeTabInt permet de faciliter
l'écriture de la VueIHMFX
idem pour CommandeInt
Facade :
Offre une interface pour manipuler
le modèle
Controleur
Singleton :
Un seul contrôleur
VueNbCoupIHMFX
VueIHMFX
IHMTerminal
MonteurScene
ControleurIHMFX
Monteur :
Pour construire facilement
une scène
IHMFX
Main
«create»
1
*

Génie logiciel

Comment démarrer et finir le projet

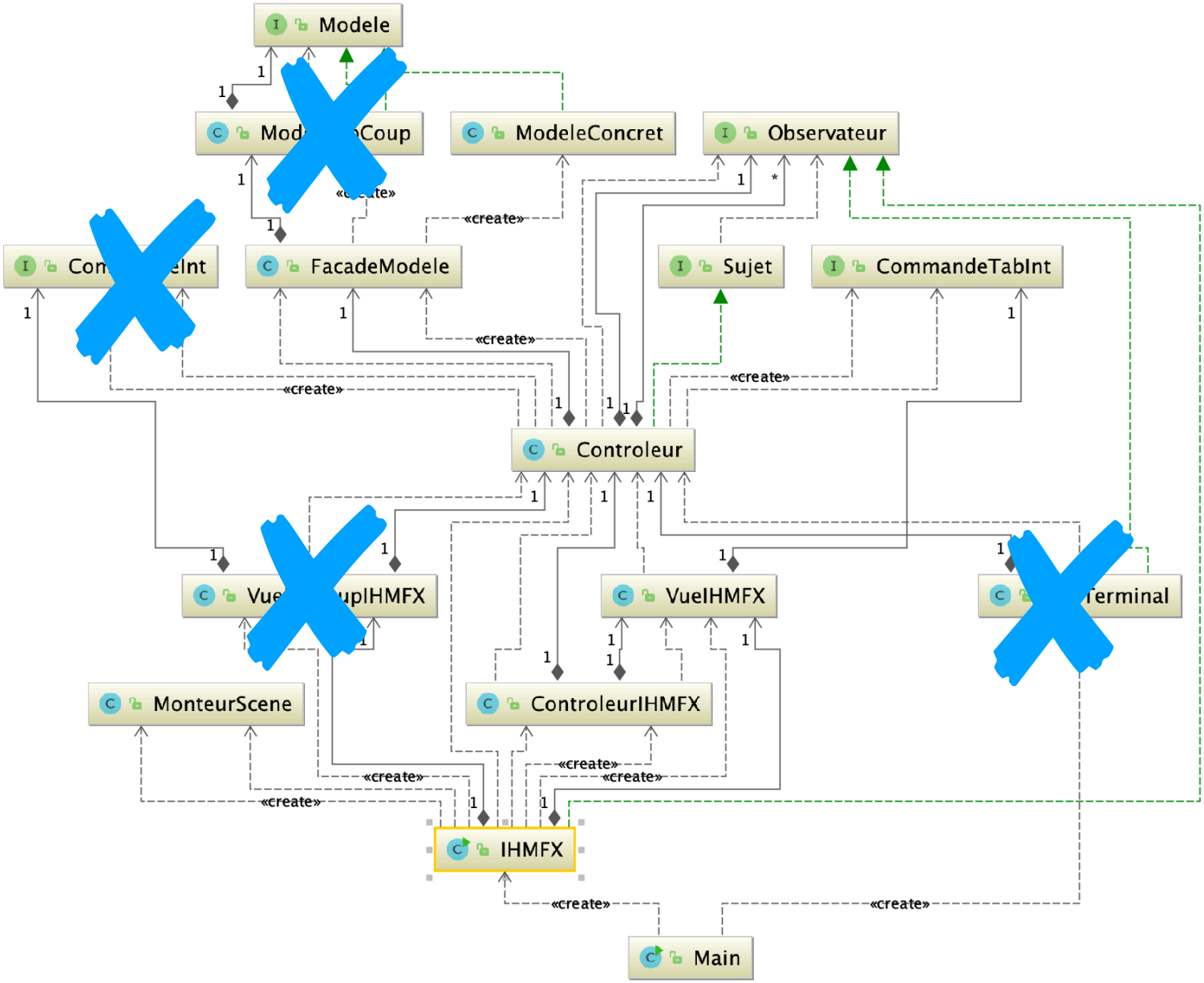
- **Proposition** : mettre en place une équipe chirurgical : désigner un architecte et les autres membres sont à son service !

- **Démarrage** : Reprendre chameaux6.1 et réaliser une application où Soko se déplace simplement sur une grille.

- **La poursuite :** L'architecte fait évoluer l'application en demandant la réalisation de code et en adaptant l'architecture aux besoins

- **Penser au rendu !!!**

Architecture de la première version



Questions !