

Sorbonne Université Build Order Optimizer

Groupe: **Smart Builders**

Membres du groupe :

- Amrouche Sara
- Boukhata Nora
- Fenek Ouarda
- Kadri Djaffar
- Maurice Nathan
- Oudali Saliha
- Ouerk Sara

Plan

I. Introduction

II. Use Cases

III. Diagramme Classe Métier

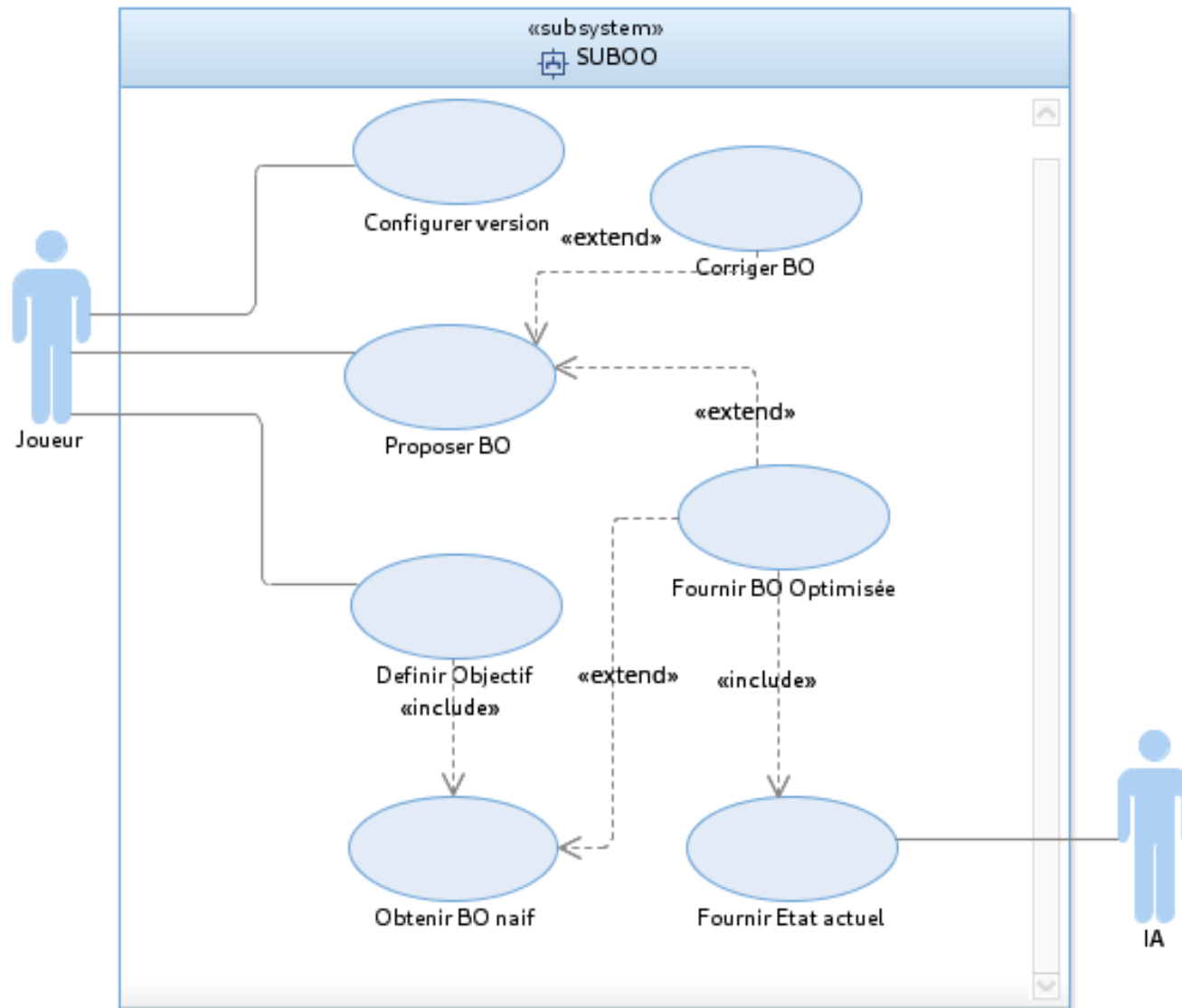
IV. Fiches détaillées

V. Diagrammes séquence

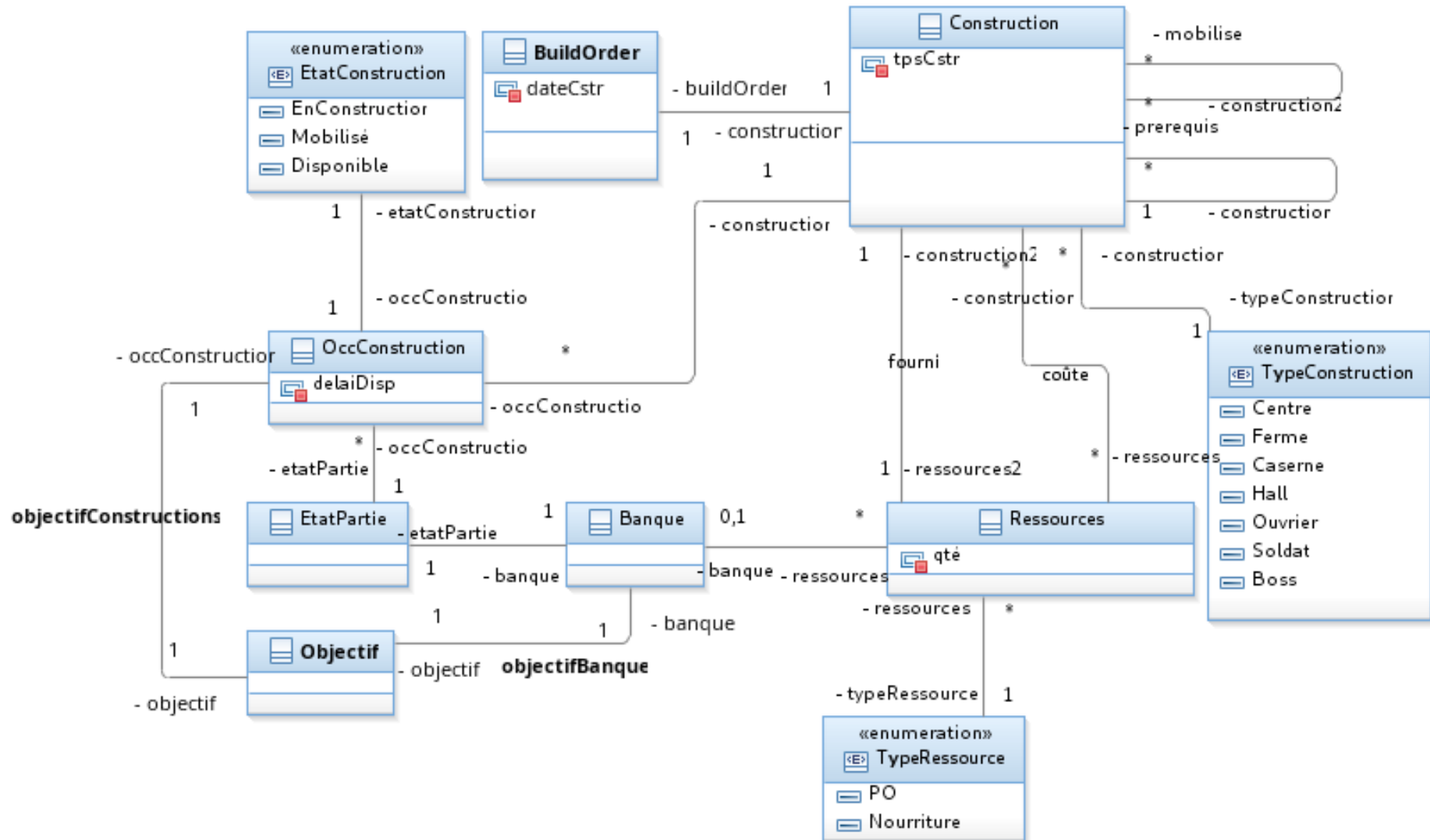
VI. Tests de validation

VII. Conclusion

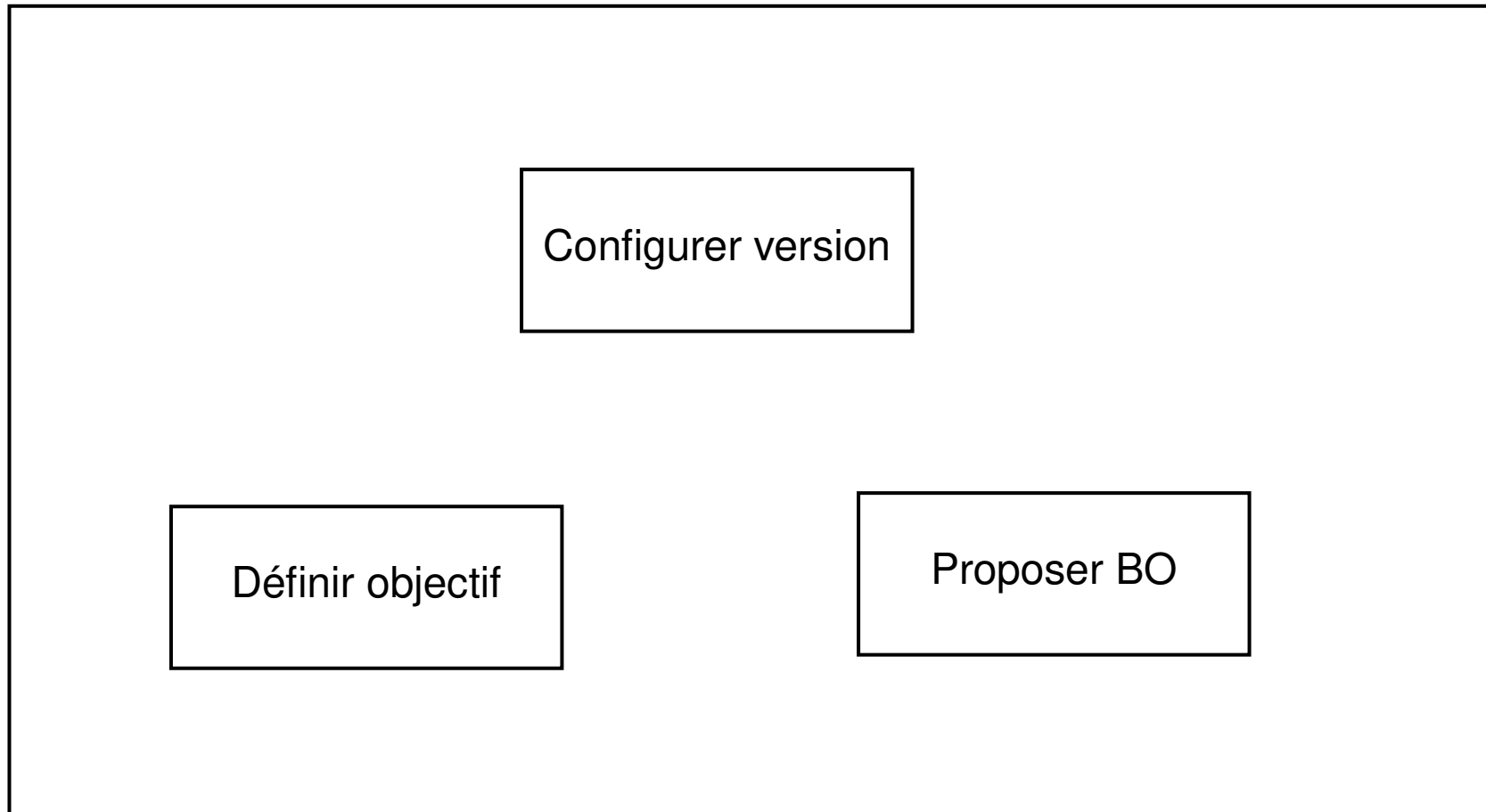
Use cases



Classes métier



Interface de démarrage



1. Configurer Version

```

1 UC01: « Configurer Version».
2 Date de création : 14/10/2018
3 Responsables: FENEK OUARDA, OUDALI SALIHA
4 Version : 1.1.
5 Description:
6     Il s'agit de définir la version du jeu (voire les règles de construction)
7     et de choisir l'approche du jeu (le joueur aura à choisir est ce qu'il fixera un objectif et le système lui fournira un BO lui
8     permettant de l'atteindre, ou bien, c'est lui qui devra proposer un BO selon un objectif fixé par le système)
9 Pré-conditions : aucune
10 Séquence nominale:
11     1- Le joueur clique sur configurer version
12     2- Le système demande de saisir l'unité concernée par la règle en cours
13     3- Le joueur saisie une unité et valide
14     4- Le système demande au joueur de saisir le coût en pièces d'or
15     5- Le joueur saisit le coût en pièces d'or et valide
16     6- Le système demande au joueur de saisir le coût en nourriture
17     7- Le joueur saisit le coût en nourriture et valide
18     8- Le système demande au joueur de saisir le temps de construction de cette unité
19     9- Le joueur saisit le temps de construction et valide
20     10- Le système propose au joueur de définir encore une règle de construction s'il en reste
21     11- Le système propose au joueur de définir un objectif à atteindre ou de proposer son propre BO
22     12- Le joueur choisit de proposer son propre BO
23     13- Le système fixe un objectif que le joueur devrait essayer d'atteindre
24     14- Appel au UC2: "Proposer BO"
25 Enchaînements alternatifs:
26     A1: Jouer avec les règles prédéfinies:
27     L'enchaînement démarre au le point 1 de la séquence nominale
28         1- Le système charge le fichier de configuration par défaut
29         2- La séquence nominale reprend au point 11
30
31     A2: Ajouter une règle de construction
32     L'enchaînement démarre après le point 10 de la séquence nominale
33         11- La séquence nominale reprend au point 3
34
35     A3: Définir un objectif pour le système
36     L'enchaînement démarre après le point 11
37         12- Le joueur choisit de définir un objectif.
38         13- Appel au UC3: "Définir Objectif"
39 Enchaînements d'exception:
40     E1: Abondan
41     L'enchaînement peut démarrer aux points 3, 5, 7 et 9
42         Le système charge le fichier de configuration par défaut.
43 Post-conditions:
44     Le système enregistre les règles de construction

```

Configurer version

Choisir l'unité :	<input type="text"/>
Coût en pièces d'or :	<input type="text"/>
Coût en nourriture :	<input type="text"/>
Temps de construction :	<input type="text"/>
Définir une autre règle	Terminer

2. Proposer BO

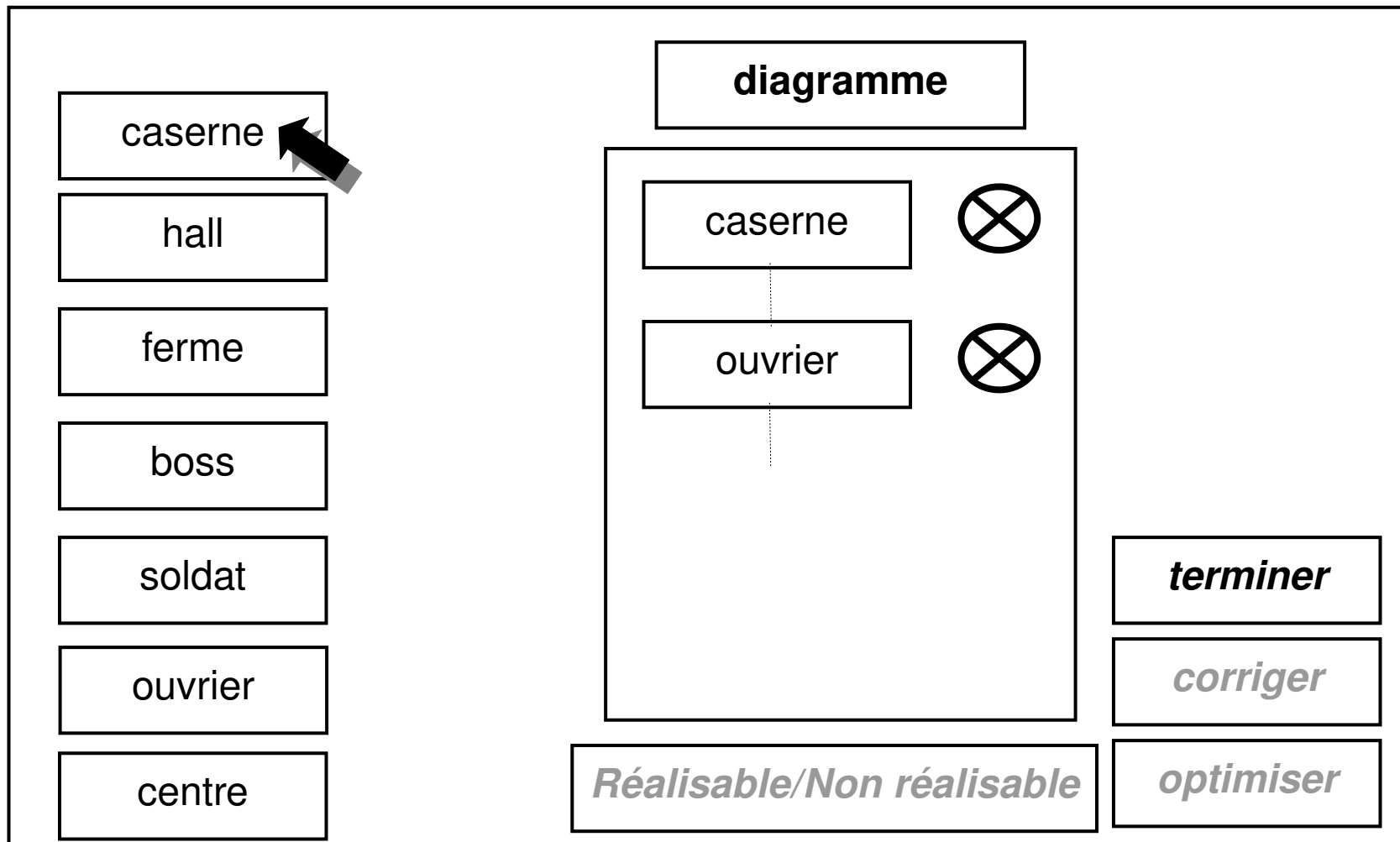
```

1 UC02 : « Proposer B0 ».
2 Date de création : 14/10/2018
3 Responsables: FENEK OUARDA, OUDALI SALIHA
4 Version : 1.1.
5 Description:
6         Le joueur proposera un B0 que selon lui permettra d'atteindre l'objectif qui lui a été défini par le système.
7 Pré-conditions :
8         Les règles du jeu sont définies
9 Séquence nominale:
10        1- Le joueur clique sur le bouton proposer B0
11        2- Le système affiche à l'utilisateur une interface lui permettant de choisir une unité à produire à l'étape courante et un
12           diagramme représentant le B0 en construction.
13        3- le joueur choisit une unité à produire.
14        4- le système ajoute cette construction au B0 et au diagramme en proposant de l'annuler.
15        5- le système propose au joueur de terminer son B0 ou de continuer sa construction
16        6- le joueur choisit de terminer
17        7- Le système simule l'exécution et vérifie si le B0 est réalisable
18        8- Le système affiche le temps d'exécution estimé à la réalisation de ce B0
19        9- Le système propose d'optimiser le B0 obtenu ou de terminer l'interaction
20        10- Le joueur choisit de terminer
21        11- Le système termine l'interaction
22 Enchaînements alternatifs:
23        A1: Annulation d'une construction
24        L'enchaînement démarre après le point 4 de la séquence nominale
25                5- le joueur clique sur annuler
26                6- Le système supprime la construction du B0 et du diagramme affiché
27                7- La séquence nominale reprend au point 5
28        A2: Poursuite de la construction
29        L'enchaînement démarre après le point 5 de la séquence nominale
30                6- La séquence nominale reprend au point 2.
31        A3: le B0 proposé est non réalisable:
32        L'enchaînement démarre après le point 7 de la séquence nominale
33                7- le système affiche que le B0 est non réalisable
34                8- le système propose de corriger le B0
35                9- le joueur clique sur corriger
36                10- faire appel à UC4 : Corriger B0
37                11- Le système affiche le B0 corrigé
38                12- la séquence nominale reprend au point 9
39        A3: Optimisation du B0
40        L'enchaînement démarre après le point 9
41                - le joueur choisit d'optimiser le B0
42                - Faire appel à UC6: Fournir B0 Optimisé
43                - le système affiche le B0 optimisé
44

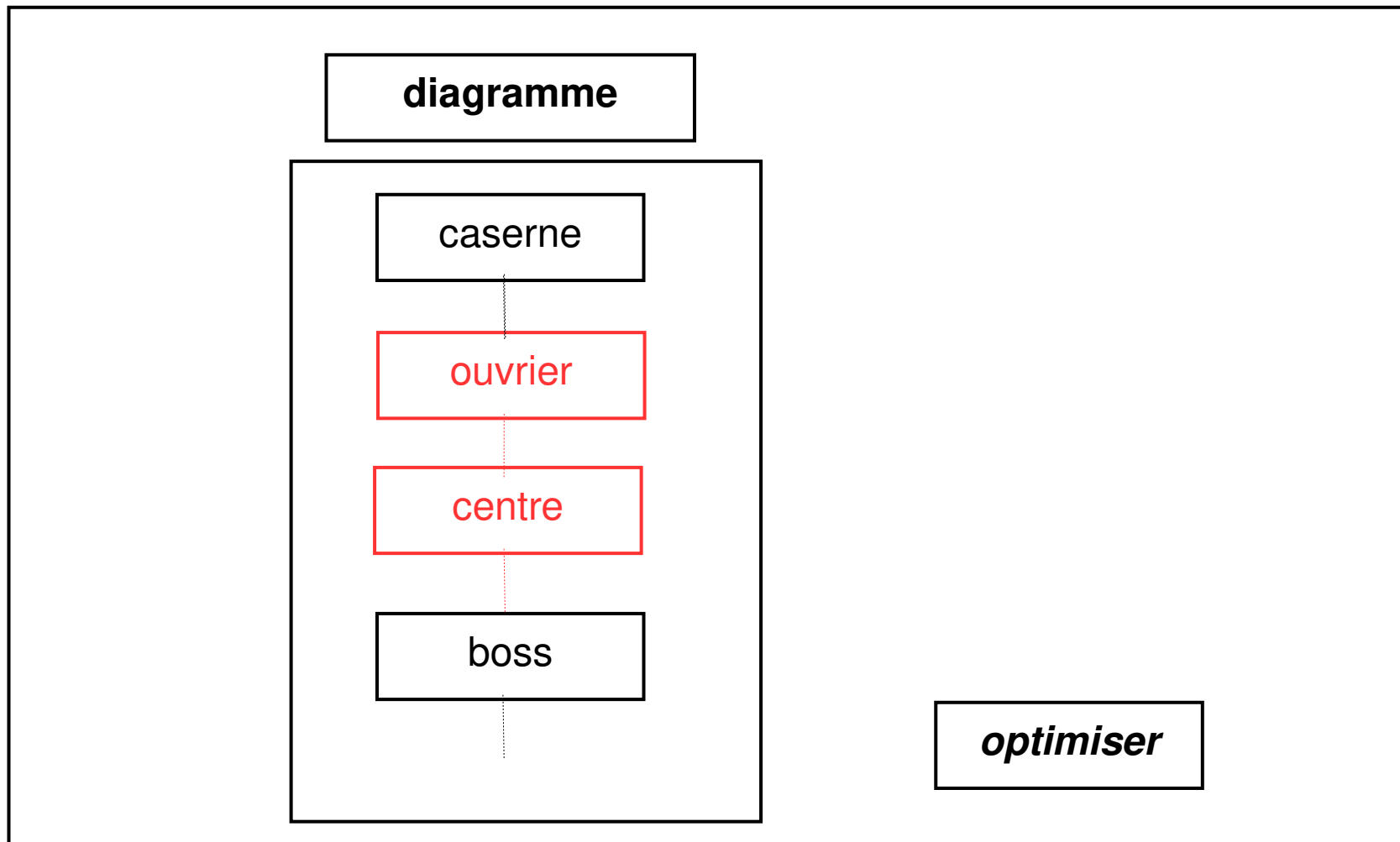
```

```
44
45 Enchaînements d'exception:
46     E1: Abandon de la construction du B0
47     L'enchaînement peut démarrer aux points 3, 6
48         Appel de l'exécution du UC1: « Configurer Version »
49 Post-conditions:
50     Le système enregistre le B0 proposé par le joueur.
51
52
```

Proposer BO

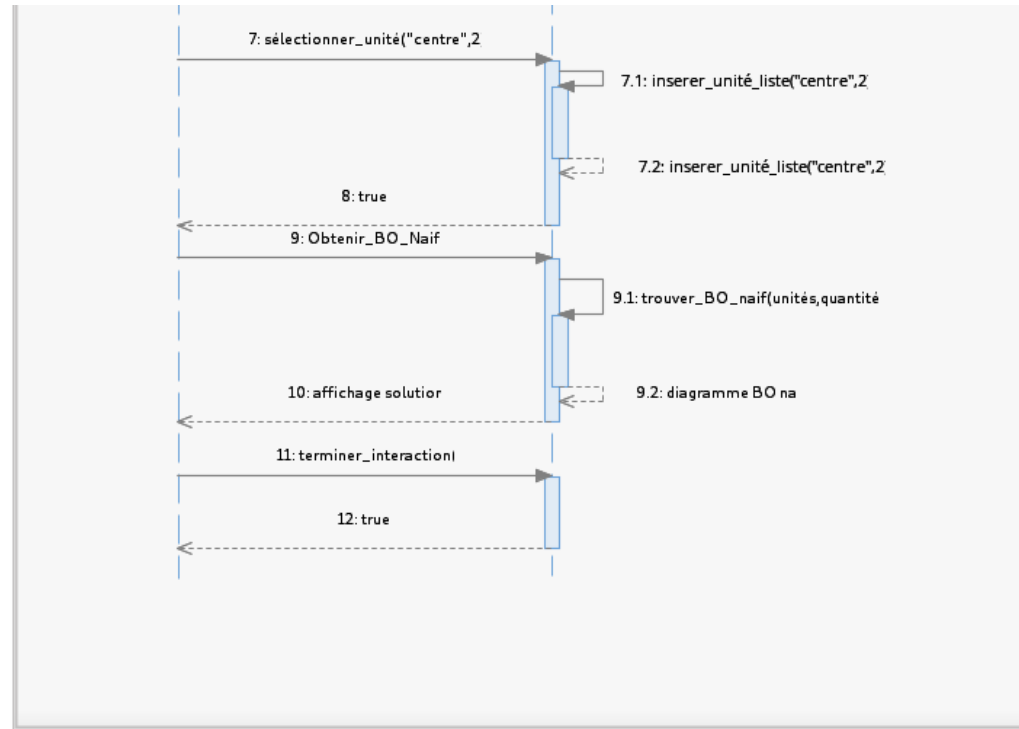
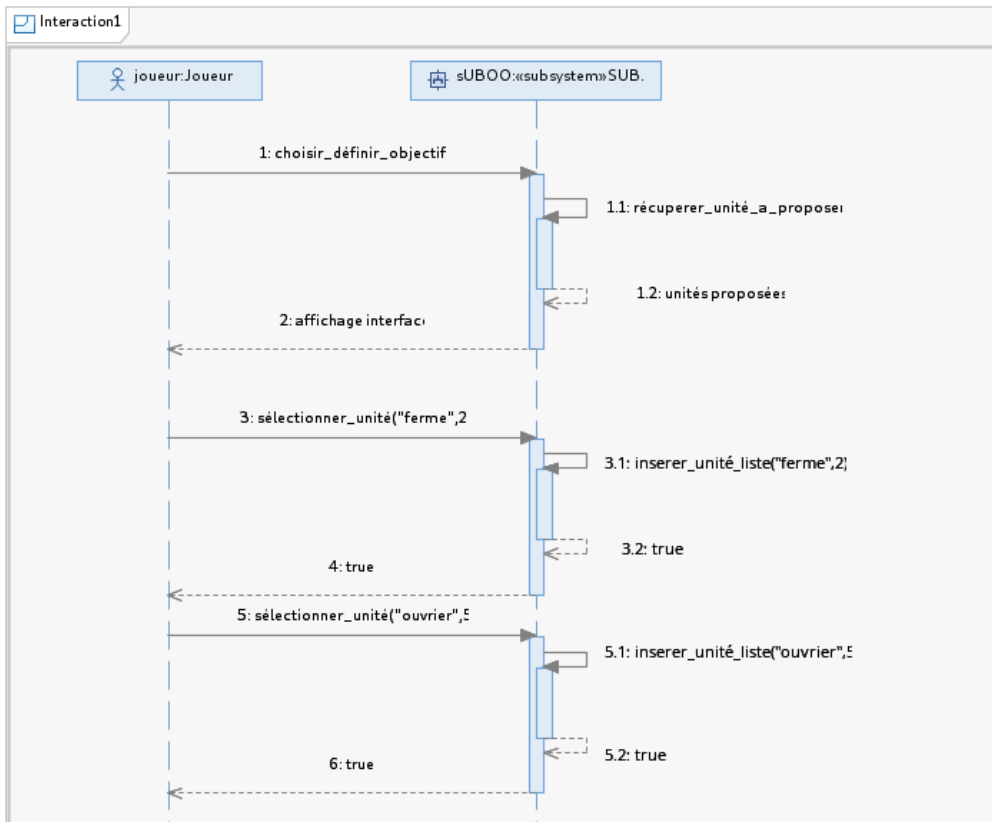


BO corrigé



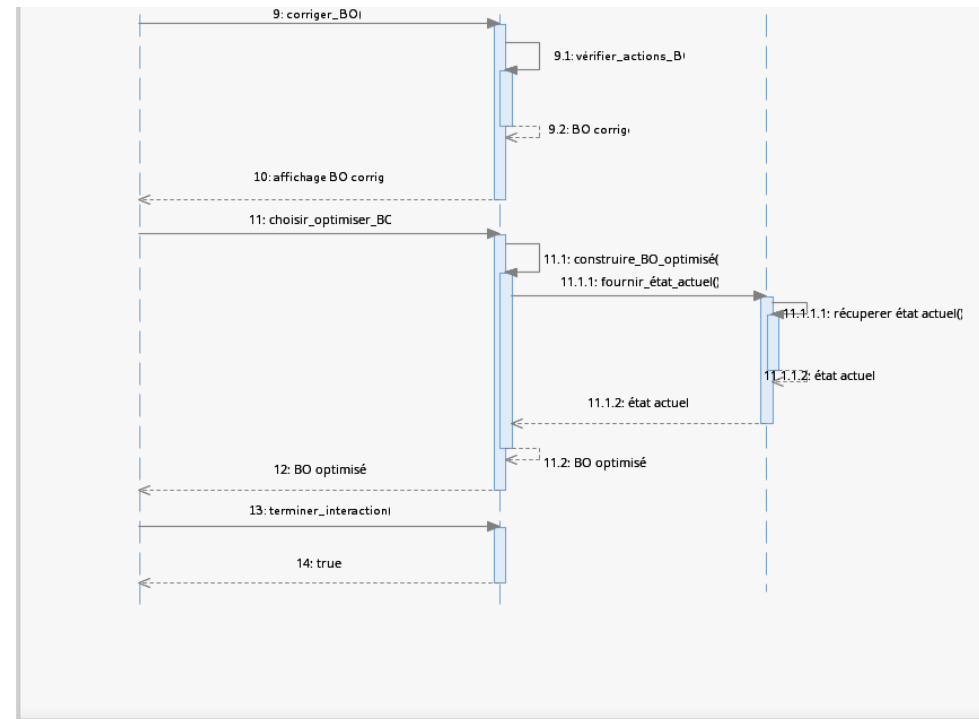
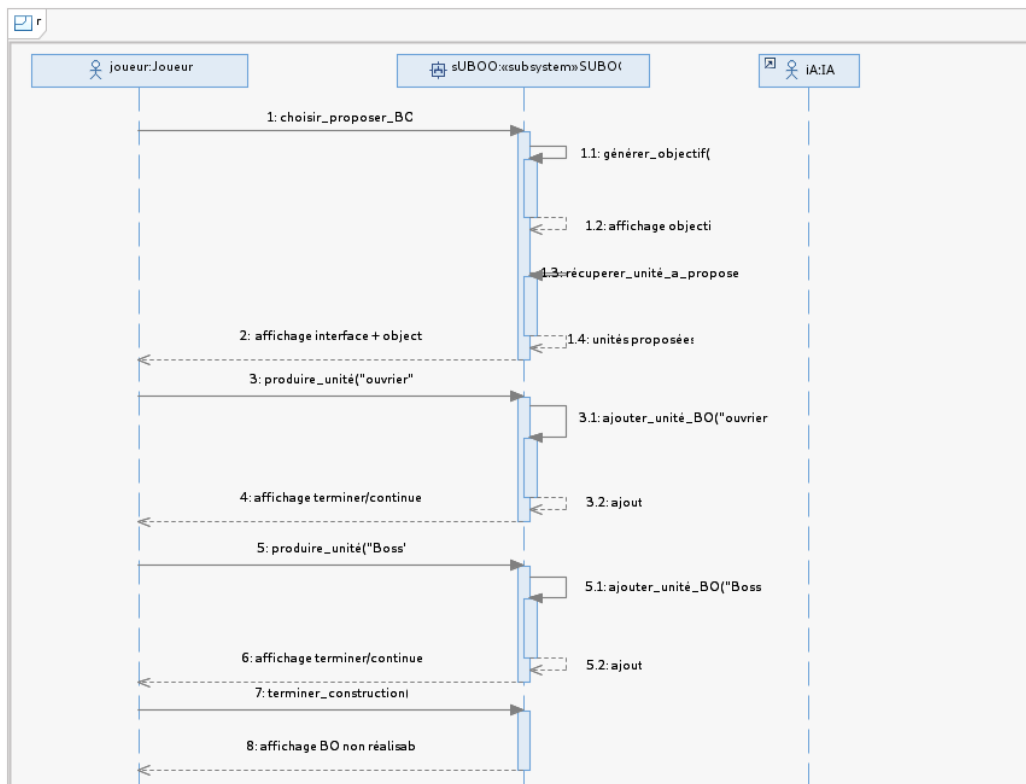
Diagrammes de séquence

Proposer BO naïf :



Diagrammes de séquence

Proposer BO:



TV01 : « Définir Objectif »

Contexte :

- le joueur démarre avec la version par défaut.
- le joueur saisie un objectif.

Entrée :

- 6 ouvriers.

Scenario :

le testeur appuie sur le bouton “définir objectif”.

Le testeur sélectionne 6 ouvriers.

Le testeur appuie sur valider qui lance la construction du BO.

R.A :

Séquence d'actions de construction:

-ouvrier

La durée =30s.

M.V :

visuel.

TV02 : « Le joueur propose un BO non réalisable »

Contexte :

le système a fixé un objectif.

le joueur introduit son BO.

Entrée :

**Banque initiale : 50 PO/ 5
ouvriers/centre**

Objectif : avoir 3 BOSS et une caserne.

BO : Ferme/ Boss/ hall/ Boss/ Boss

Scenario:

le testeur appuie sur le bouton “proposer BO ”.

Le joueur ajoute au BO une ferme.

Le joueur ajoute au BO un boss.

Le joueur ajoute au BO un hall.

Le joueur ajoute au BO un boss.

Le joueur ajoute au BO un boss.

Le joueur appuie sur « terminer ».

R.A:

Le système affiche que le BO est non réalisable et lui propose de le corriger.

TV03.

M.V :

Visuel, en comparant avec le résultat obtenu préalablement à la main.

TV03 : « corriger BO »

Contexte :

- le joueur a introduit un BO non réalisable.
- le joueur a appuyé sur corriger BO.

Entrée :

-BO non réalisable :

Ferme/ Boss/ Hall/ Boss/ Boss

-Objectif : avoir 3 BOSS et une caserne.

Scenario :

Le testeur appuie sur le bouton “corriger BO »

R.A :

Séquence d'actions du BO corrigé :

ferme/caserne/hall/ boss/ boss/ boss.

Durée d'exécution = 630s

Le système propose au joueur d'optimisé BO.

M.V : visuel

Conclusion ...

Merci Pour votre attention !

