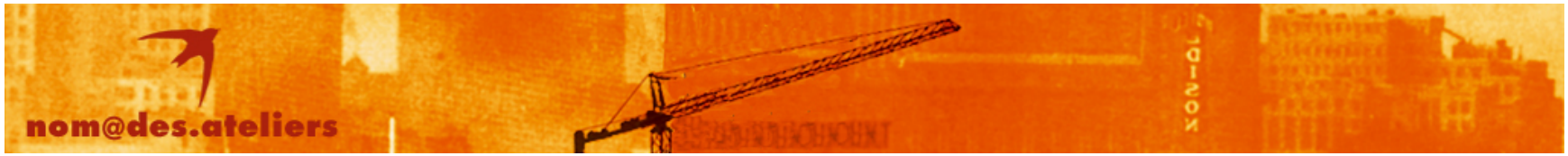
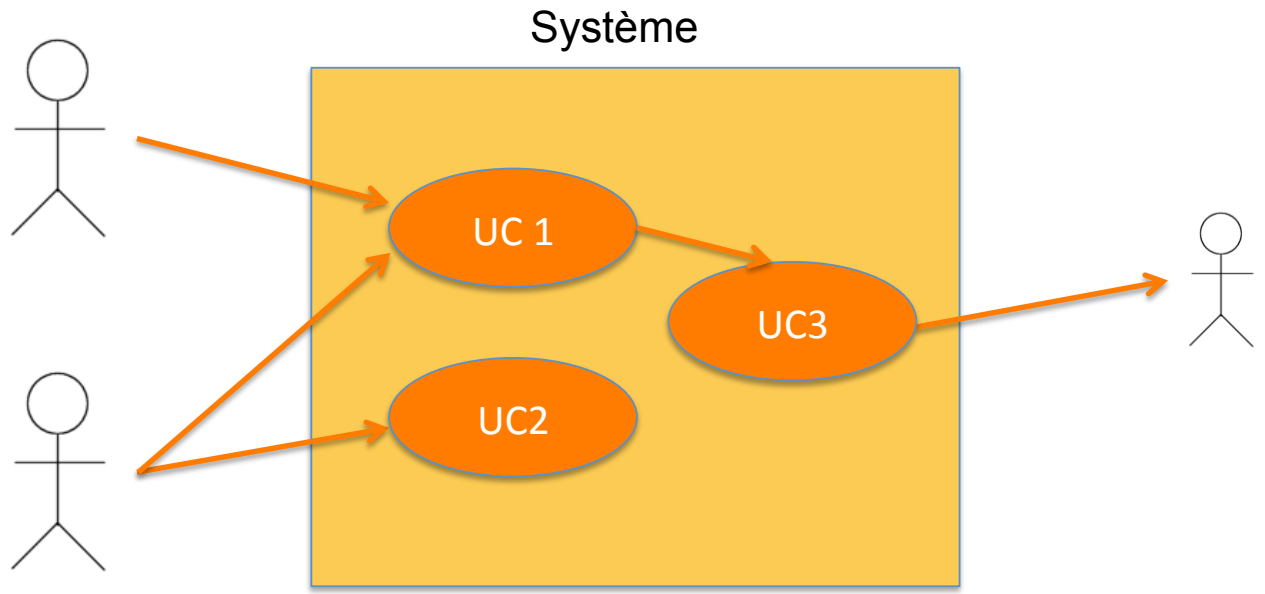


Introduction à la programmation

Rolf Hauri

Use case

- Une fois que le système est décomposé en UC:



Use case

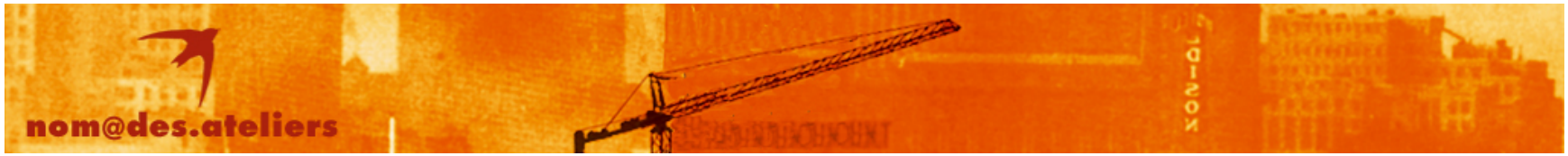
- Que les use cases sont explicités (et validés…)

Use case de l'ascenseur : Se déplacer

1. L'utilisateur appelle l'ascenseur
2. L'ascenseur se déplace à l'étage de l'utilisateur
3. L'ascenseur ouvre les portes
4. L'utilisateur entre dans l'ascenseur
5. L'utilisateur choisit l'étage
6. L'ascenseur ferme les portes
7. L'ascenseur valide le poids des occupants
8. L'ascenseur se déplace à l'étage choisi
9. L'ascenseur ouvre les portes
10. L'utilisateur sort dans l'ascenseur
11. L'ascenseur ferme les portes

7.a Le poids est trop élevé
7.a.1 L'ascenseur affiche un message d'erreur
7.a.2 L'ascenseur ouvre les portes
7.a.3 Un (des) utilisateurs sort(ent)
7.a.4 Le use case reprend en 5

*.a Un utilisateur appuie sur l'alarme
*.a.1 L'ascenseur s'arrête au prochain étage
*.a.2 L'ascenseur ouvre ses portes
*.a.3 Le use case se termine

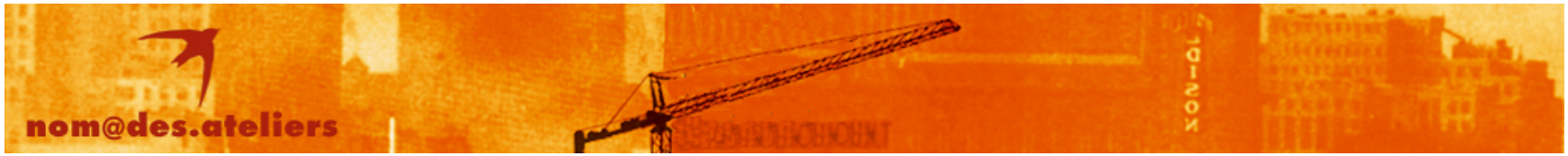


Use case

- Comment continuer?

Use case

- Comment continuer?
- Spécifications technique
- Conception du use case
  - Modèles (Modèle objet, diagramme de séquence, de classe, d'état, …)
- Réalisation du code du UC
- Test/implémentation…



Spécifications techniques

- Prototypes (navigation, vues, de faisabilité, …)
- Champs, labels, ordre, validations, …
- Type (champs texte, liste déroulante, …)
- …

Conception du UC

- Objets participants (*acteurs* internes au système)
- Interaction entre les objets
- Séquence des interactions

Modèle objet

- Quels sont les objets qui sont nécessaires, qui participent aux interactions?
- Participations possibles
  - L'objet contient des données
  - L'objet permet de saisir ou d'afficher des données
  - L'objet réalise des actions, ou les coordonne
- Attention: un acteur UC ne figure pas dans un diagramme de séquence "interne"!

Famille d'objet

- Entité – Contient des données

- Interface

- Contrôleur ou Manager

Modèle objet

Exemple: "L'ascenseur"

Diagramme de séquence

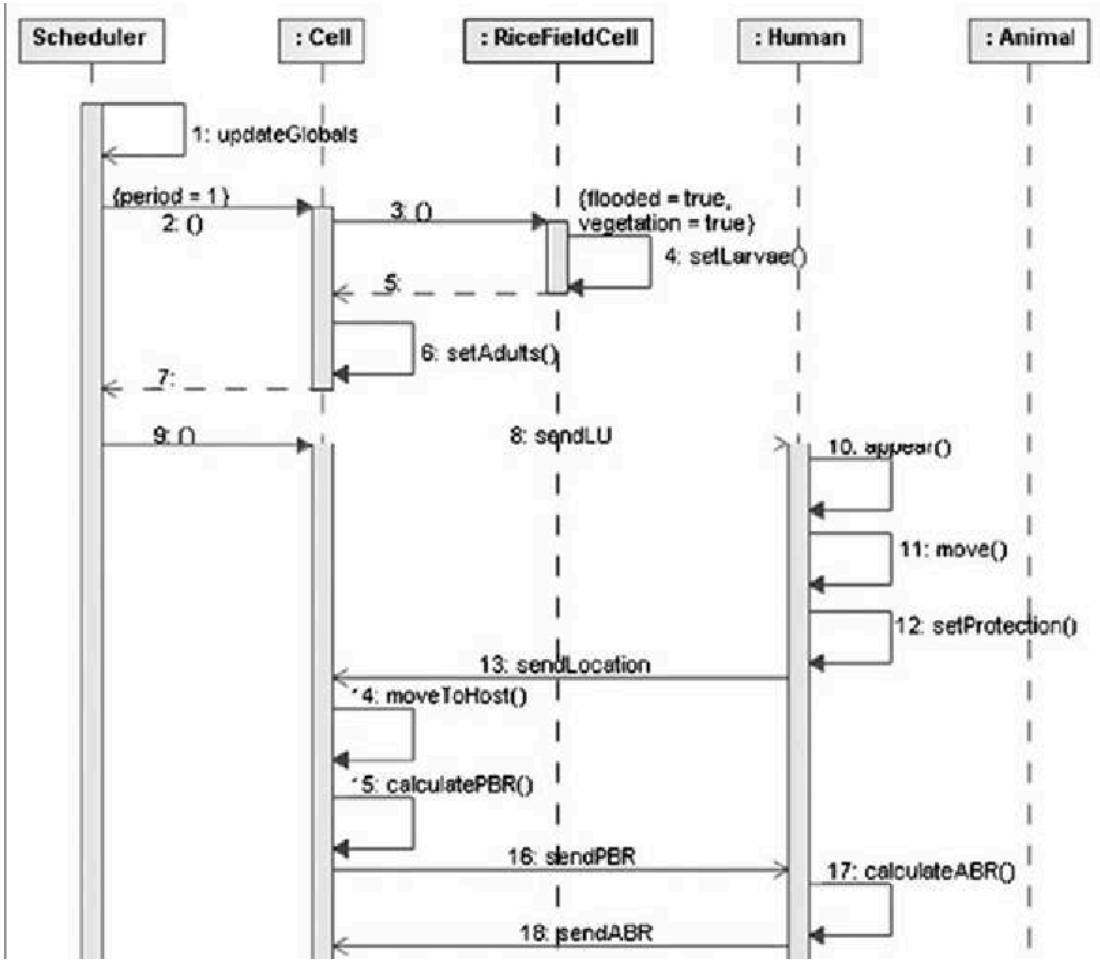
- Permet de spécifier la séquence, l'ordre dans lequel les différentes actions relatives aux objets doit être fait.
- Tous les objets identifiés y participent

Truc et astuces

- Le code réalisé "hors objet" sera ajouté sous forme d'un contrôleur 

Diagramme de séquence

- Exemple…



Décomposition du diagramme

- La décomposition s'effectue aussi sur les séquences d'action
- Les actions d'une séquence devraient être de même niveau
- Des sous diagrammes de séquence permettent de décomposer la solution

Best practice

- Les flux alternatifs peuvent être représentés dans les diagrammes de classe, attention ils les rendent moins lisibles!
- Option: créer des diagrammes alternatifs

Diagramme de séquence

Exemple: "L'ascenseur"

Code

- Chaque action deviendra une instruction:

Une ligne horizontale du diagramme de séquence

=

Une ligne de code dans le programme

Outil

- Les modèles vus proviennent de l'UML (Unified Modeling Language) – Présentation à venir

- Des éditeurs UML sont disponibles
  - argoUML par exemple : http://argouml.tigris.org/

Ne pas confondre

- Les organigrammes
- La décomposition
- Les use cases
- Le modèle objet
- Les diagrammes de séquence

Ne pas confondre

- Les organigrammes
  - Représentation visuel d'un processus, quel qu'il soit, à n'importe quelle étape
- La décomposition
  - Processus de découpage d'un problème, d'une interaction, …
- Les use cases
  - Décrit les interactions entre les acteurs et le système
- Le modèle objet
  - Décrit les objets participant au UC
- Les diagrammes de séquence
  - Décrit les interactions entre les objets internes participant au UC