

Software-Ontwerp

Iteratie 2

Reniers V. - Devlieghere J. - Castel D. - Pante S.

KU Leuven

March 19, 2013

Inhoud

1 Inleiding

- Rolverdeling
- Werkverdeling

2 Het ontwerp

- MVC
- Grid
- Obstacles
- States and Penalty

3 System State Diagrams

- Start New Game

- Move

- Move

- Pick Up Item

- Pick Up Item

- Use Item

- Use Item

- End Turn

- End Turn

4 Slot

Inleiding

Thema's die aan bod komen:

- High-Level bespreking van het ontwerp.
- Onderdelen in detail bekeken.
- GRASP en design patterns.
- Uitbreidbaarheid van het ontwerp.
- Test cases.

Rolverdeling

- 1 Inleiding
 - Rolverdeling
 - Werkverdeling
- 2 Het ontwerp
 - MVC
 - Grid
 - Obstacles
 - States and Penalty
- 3 System State Diagrams

- Start New Game
- Move
- Move
- Pick Up Item
- Pick Up Item
- Use Item
- Use Item
- End Turn
- End Turn

- 4 Slot

Rolverdeling

Iteratie 2:

- Lead Designer: Dieter Castel
- Lead Tester: Vincent Reniers

Iteratie 3:

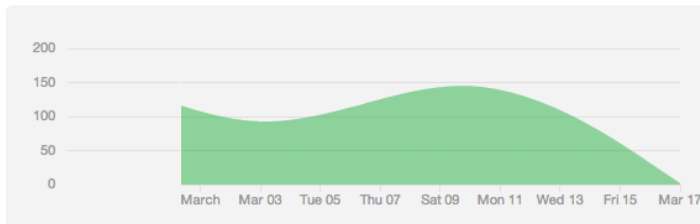
- Lead Designer: Jonas Devlieghere
- Lead Tester: Stefan Pante
- Domain Modeler: Vincent Reniers

Werkverdeling

Iteratie 2: 1 maart - 15 maart

February 10th 2013 - March 17th 2013

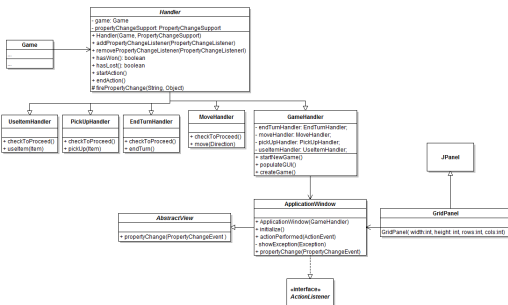
Commits to master, excluding merge commits

Contribution Type: **Commits** ▼

MVC



Handlers and view



Handlers and view

MVA (Model-View-Adapter)

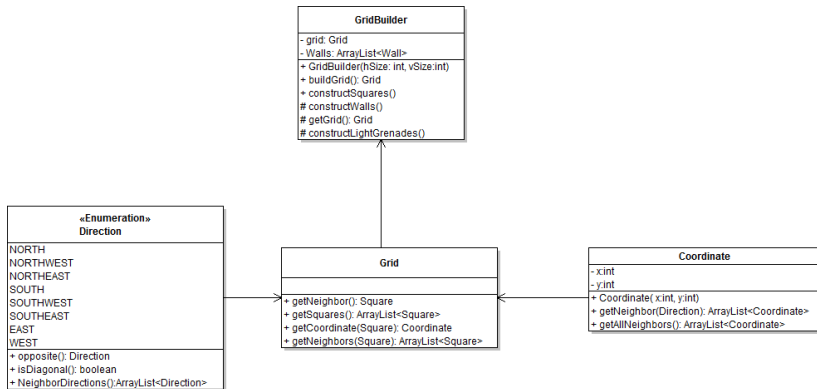
- Model en View communiceren niet rechtstreeks
- Handlers zijn **mediating controllers**
- *ApplicationWindow* implementeert *PropertyChangeListener*

Handlers

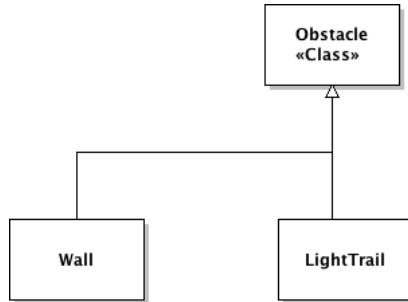
- Handler voor elke Use-Case
- Geen GUI controller meer

Grid

Grid



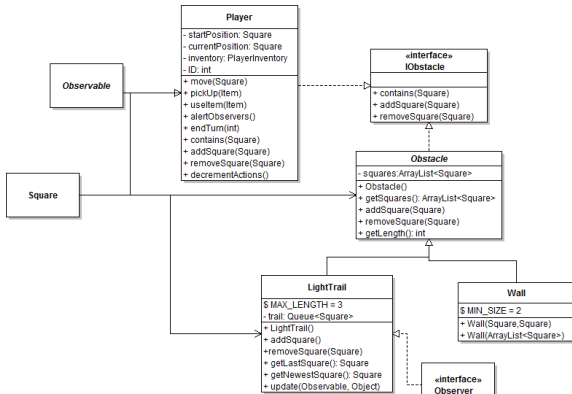
Obstacle interface



Obstacle

- Interface *IObstacle*
 - Abstracte klasse *Obstacle* implementeert *IObstacle*
 - *LightTrail* implementeert *Obstacle*
 - *Wall* implementeert *Obstacle*
 - *Player* implementeert *IObstacle*
 - *Square* kan *Obstacle* bevatten
- LightTrail implementeert de *Observer* interface.

Obstacle

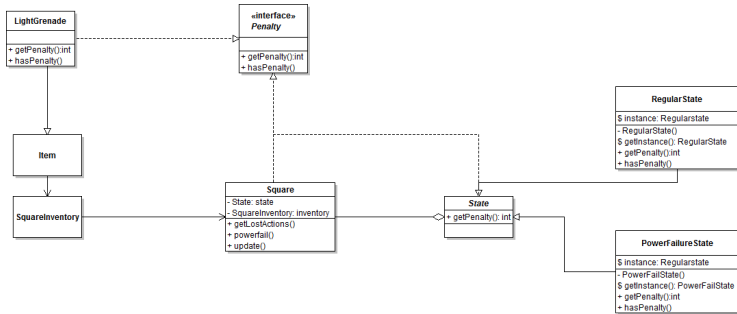


States and Penalty

- State Pattern
 - Square heeft meerdere toestanden: *RegularState*, *PowerFailureState*
 - Square zorgt voor overgang van staat
- Chain of Responsibility (Command)
 - State bepaalt eigen penalty
 - LightGrenade bepaalt eigen penalty
 - Square is eigenaar van concept penalty

States and Penalty

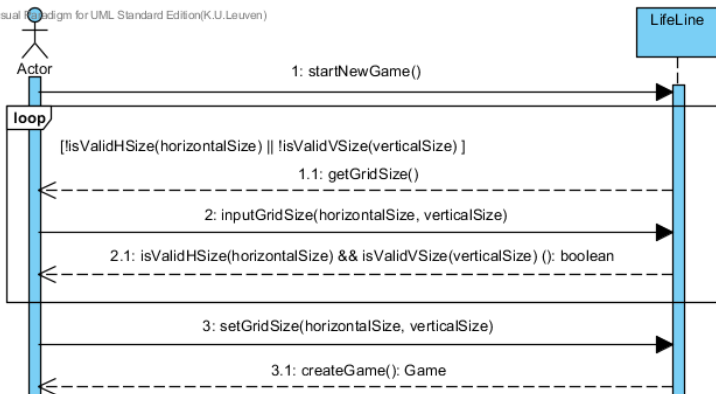
States and Penalty



Start New Game

Start New Game

Visual Paradigm for UML Standard Edition(K.U.Leuven)



Move

Move deel 1

Move

Move deel 2

Pick Up Item

Pick Up Item deel 1

Pick Up Item

Pick Up Item deel 2

Use Item deel 1

Use Item deel 2

End Turn

End Turn deel 1

End Turn

End Turn deel 2

Besluit

Bedankt voor uw aandacht.