

Software-Ontwerp

Iteratie 2

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

KU Leuven

April 18, 2013

Inhoud

1 Inleiding

- Rolverdeling
- Werkverdeling

2 Het ontwerp

- MVC
- Handlers
- Events
- Grid

- Obstacles
- States and Penalty

3 System State Diagrams

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

4 Slot

Rolverdeling

1 Inleiding

- Rolverdeling
- Werkverdeling

2 Het ontwerp

- MVC
- Handlers
- Events
- Grid

3 System State Diagrams

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

4 Slot

Rolverdeling

Afgelopen iteratie:

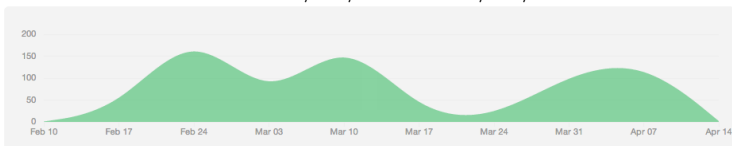
- Lead Designer: Jonas Devlieghere
- Lead Tester: Stefan Pante
- Domain Modeler: Vincent Reniers

Komende iteratie:

- Lead Designer: Vincent Reniers
- Lead Tester: Jonas Devlieghere
- Domain Modeler: Dieter Castel

Werkverdeling

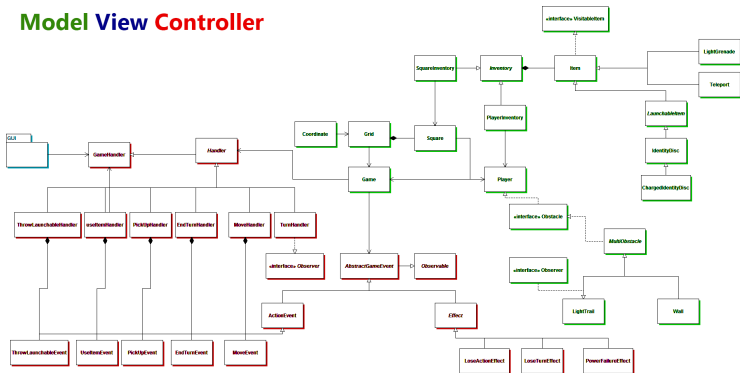
Iteratie 3: 18/03/2013 - 12/04/2013



- Gedurende 8 dagen 6/7 uur per dag
- Gemiddeld 50 uur per persoon

MVC

Model View Controller

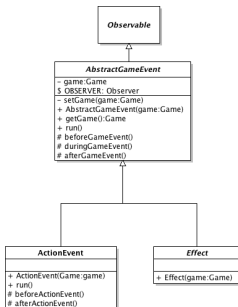


Handlers

- Handlers hadden *te veel* verantwoordelijkheid
- Uitbreidbaarheid kwam in het gedrang
- Juiste flow werd nergens afgedwongen

Events

Gebeurtenis in het spel met vaste volgorde van uitvoering



Flow

- **Voor:** Checks
- **Tijdens:** Eigenlijke afhandeling
- **Na:** Check, gevolgen

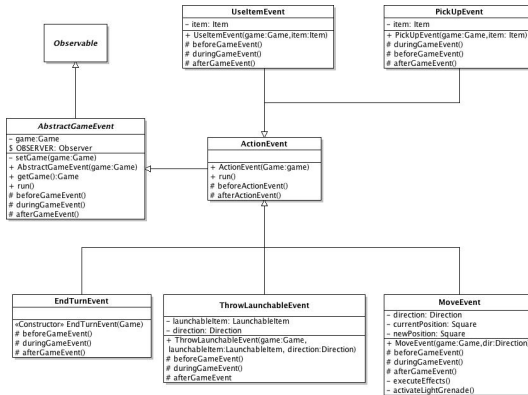
Twee soorten

- ActionEvent
- Effect

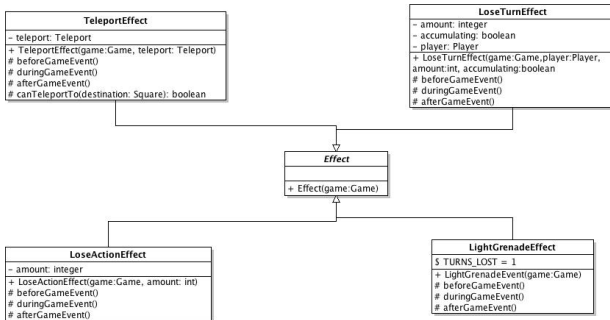
ActionEvent

- Gemeenschappelijke checks voor en na de uitvoer
- Observerbaar door de TurnHandler

ActionEvent



Effects



Grid Builder

Grid Constraint

- **Percentage:** De limiet op het aantal squares in verhouding met de totale hoeveelheid squares in het grid.
- **Excluded:** Een lijst van squares die niet gekozen mogen worden.
-

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

Start New Game

Start New Game

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.