Software-Ontwerp

Iteratie 2

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

KU Leuven

April 18, 2013

Inhoud

Rolverdeling

Rolverdeling

Afgelopen iteratie:

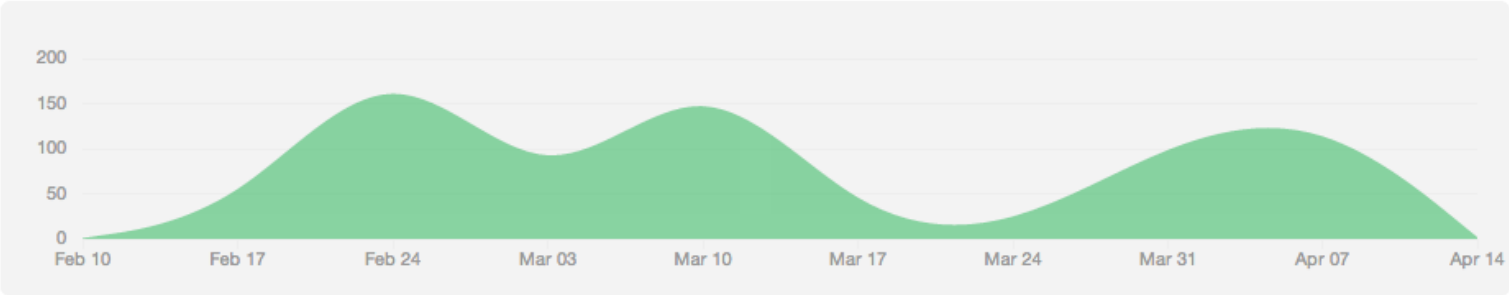
- Lead Designer: Jonas Devlieghere
- Lead Tester: Stefan Pante
- Domain Modeler: Vincent Reniers

Komende iteratie:

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

Werkverdeling

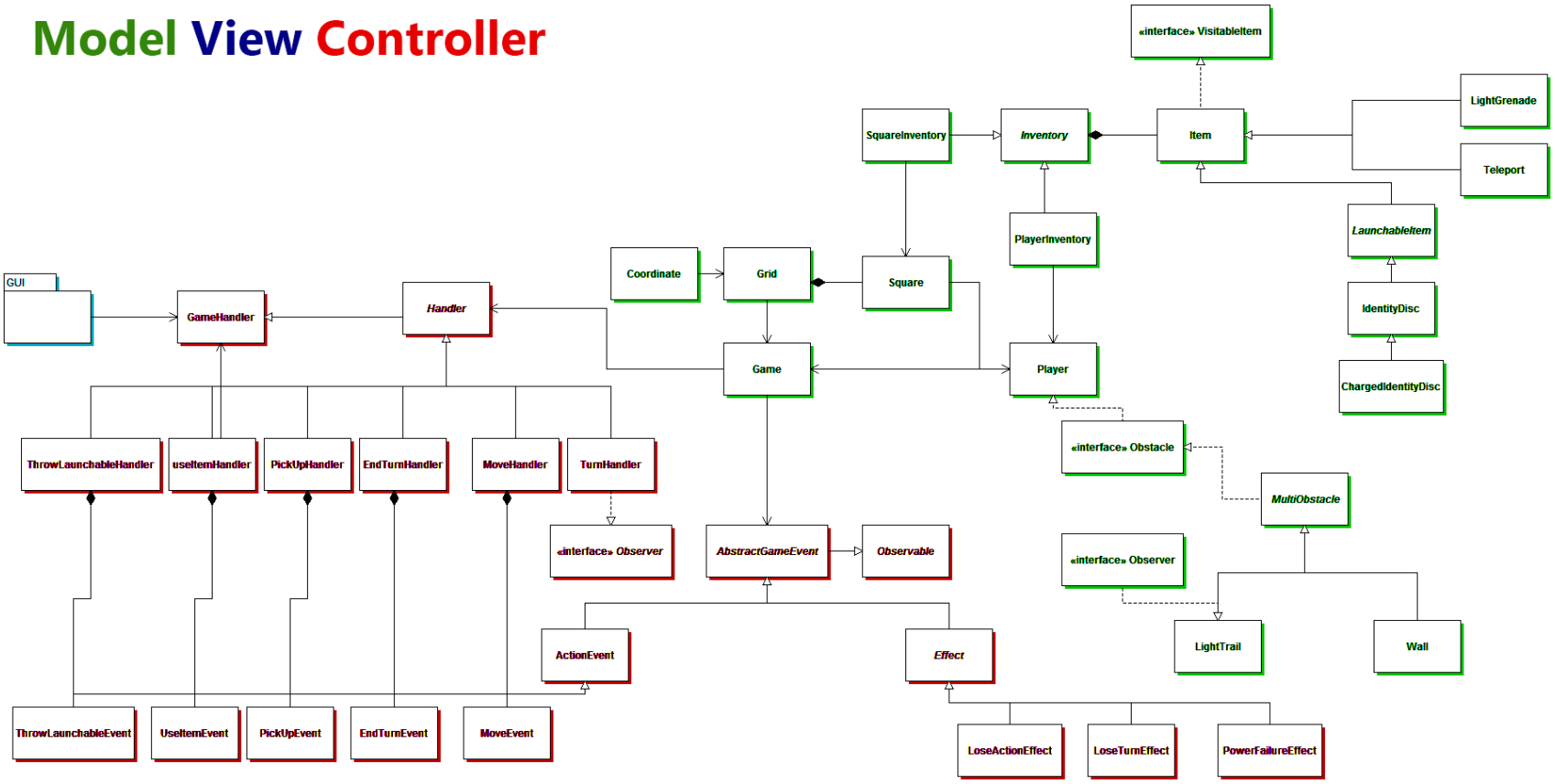
Iteratie 3: 18/03/2013 - 12/04/2013



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

Domain model

MVC

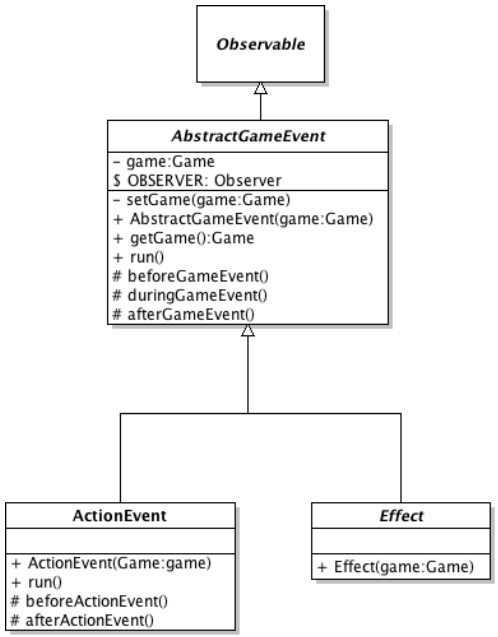

Model View Controller
«interface» VisitableItem
LightGrenade
SquareInventory
Inventory
Item
Teleport
PlayerInventory
LaunchableItem
GUI
Coordinate
Grid
Square
IdentityDisc
GameHandler
Handler
Game
Player
ChargedIdentityDisc
ThrowLaunchableHandler
useItemHandler
PickUpHandler
EndTurnHandler
MoveHandler
TurnHandler
«interface» Obstacle
MultiObstacle
«interface» Observer
AbstractGameEvent
Observable
«interface» Observer
ActionEvent
Effect
LightTrail
Wall
ThrowLaunchableEvent
UseItemEvent
PickUpEvent
EndTurnEvent
MoveEvent
LoseActionEffect
LoseTurnEffect
PowerFailureEffect

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

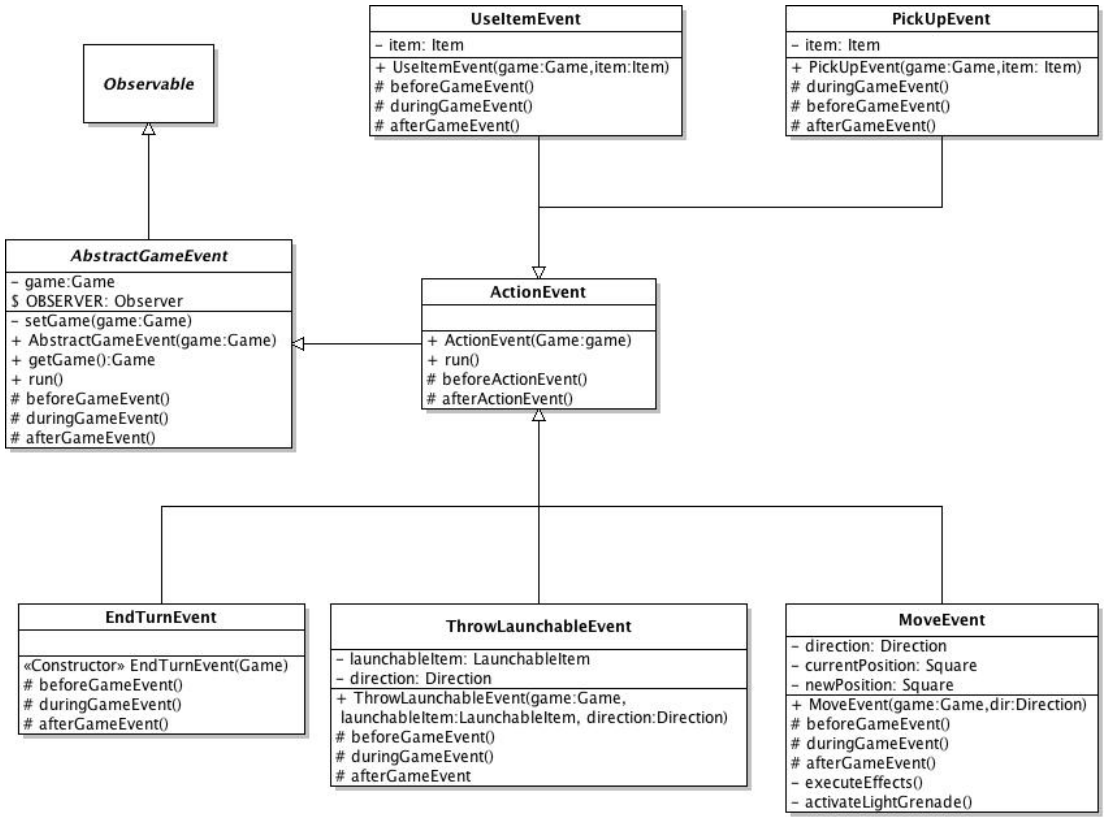
Twee soorten

- ActionEvent
- Effect

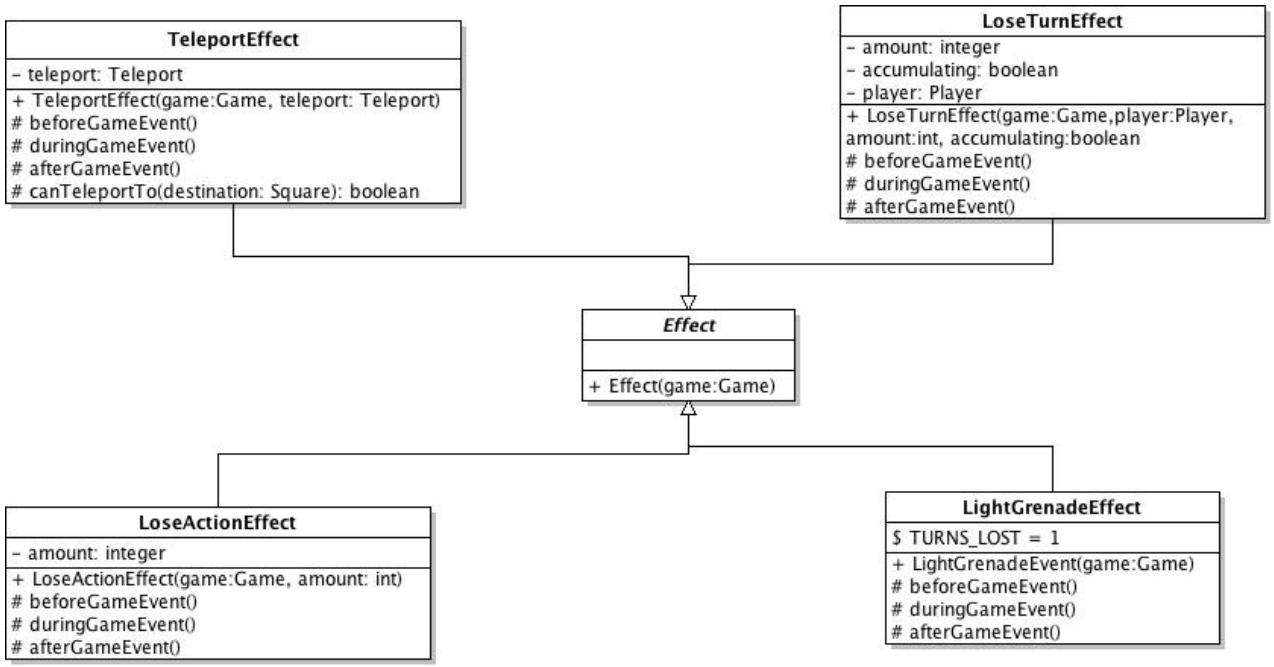
ActionEvent

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

ActionEvent



Effects


TeleportEffect
- teleport: Teleport
+ TeleportEffect(game:Game, teleport: Teleport)
# beforeGameEvent()
# duringGameEvent()
# afterGameEvent()
# canTeleportTo(destination: Square): boolean
LoseTurnEffect
- amount: integer
- accumulating: boolean
- player: Player
+ LoseTurnEffect(game:Game,player:Player, amount:int, accumulating:boolean
# beforeGameEvent()
# duringGameEvent()
# afterGameEvent()
Effect
+ Effect(game:Game)
LoseActionEffect
- amount: integer
+ LoseActionEffect(game:Game, amount: int)
# beforeGameEvent()
# duringGameEvent()
# afterGameEvent()
LightGrenadeEffect
$ TURNS_LOST = 1
+ LightGrenadeEvent(game:Game)
# beforeGameEvent()
# duringGameEvent()
# afterGameEvent()

TrajectoryMediator

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

Throw IdentityDisk

Besluit

Bedankt voor uw aandacht.