Szoftver labor 3.

Házi feladat

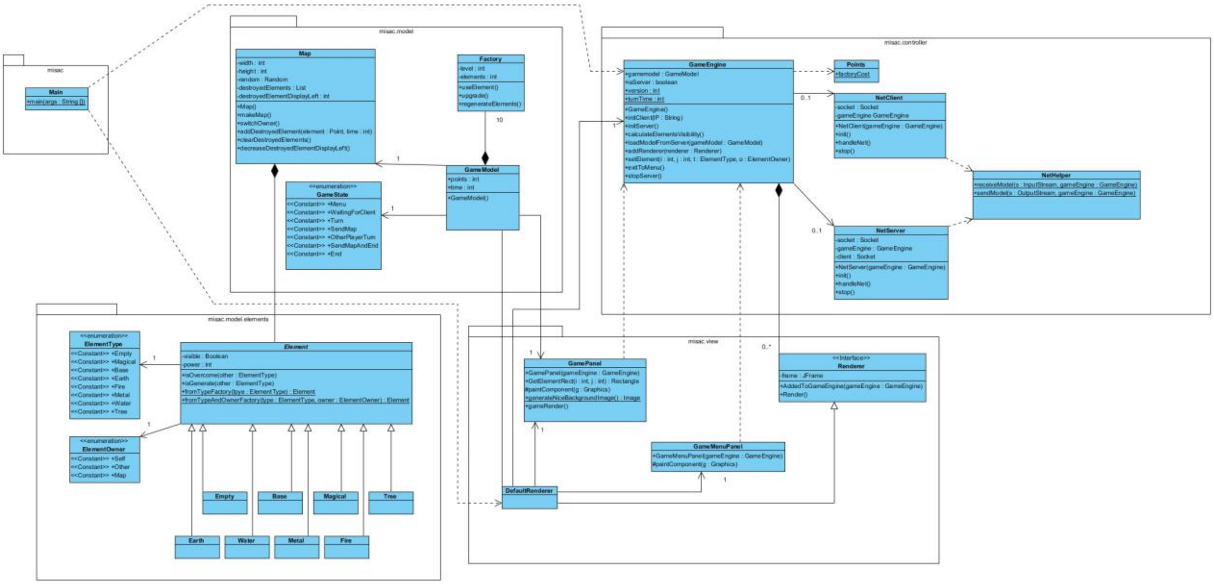
Misac

Szabó Levente

2013.11.02.

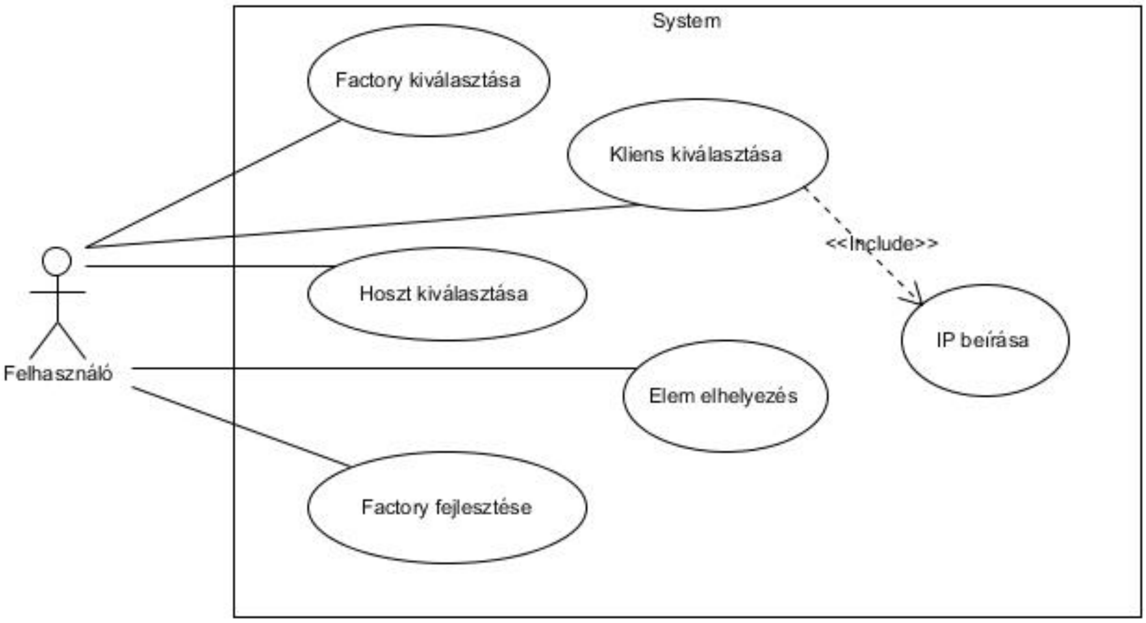
DOKUMENTÁCIÓ

CLASS DIAGRAM



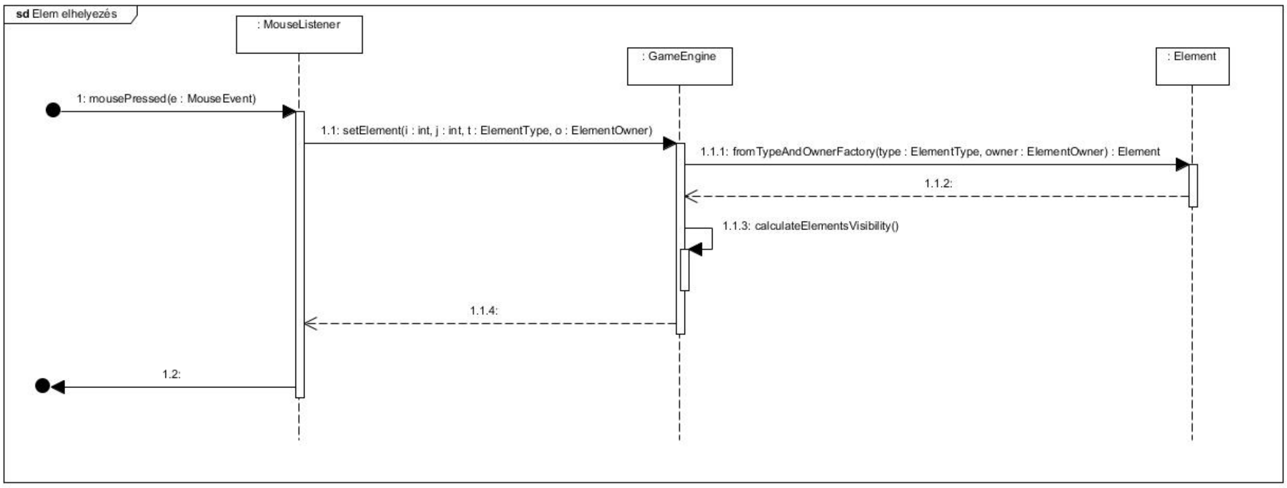
[Teljes méret (link)]

USE CASE DIAGRAM

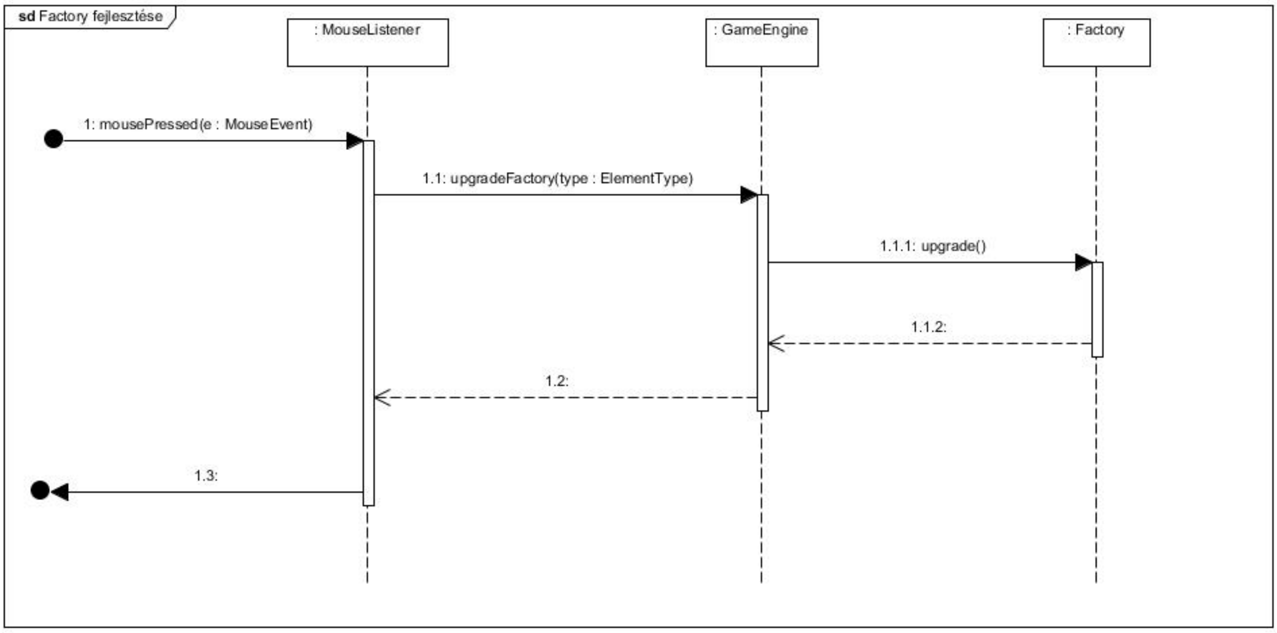


SZEKVENCIADIAGRAMMOK

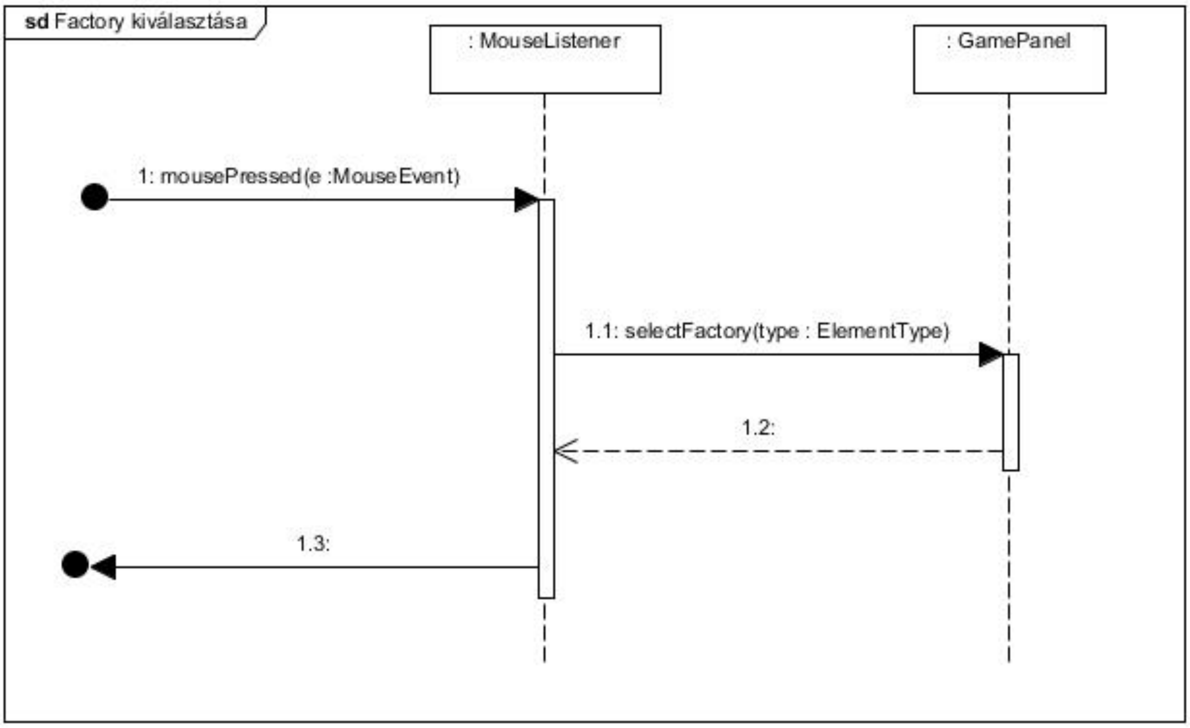
ELEM ELHELYZÉS


sd Elem elhelyezés
: MouseListener
: GameEngine
: Element
1: mousePressed(e : MouseEvent)
1.1: setElement(i : int, j : int, t : ElementType, o : ElementOwner)
1.1.1: fromTypeAndOwnerFactory(type : ElementType, owner : ElementOwner) : Element
1.1.2:
1.1.3: calculateElementsVisibility()
1.1.4:
1.2:

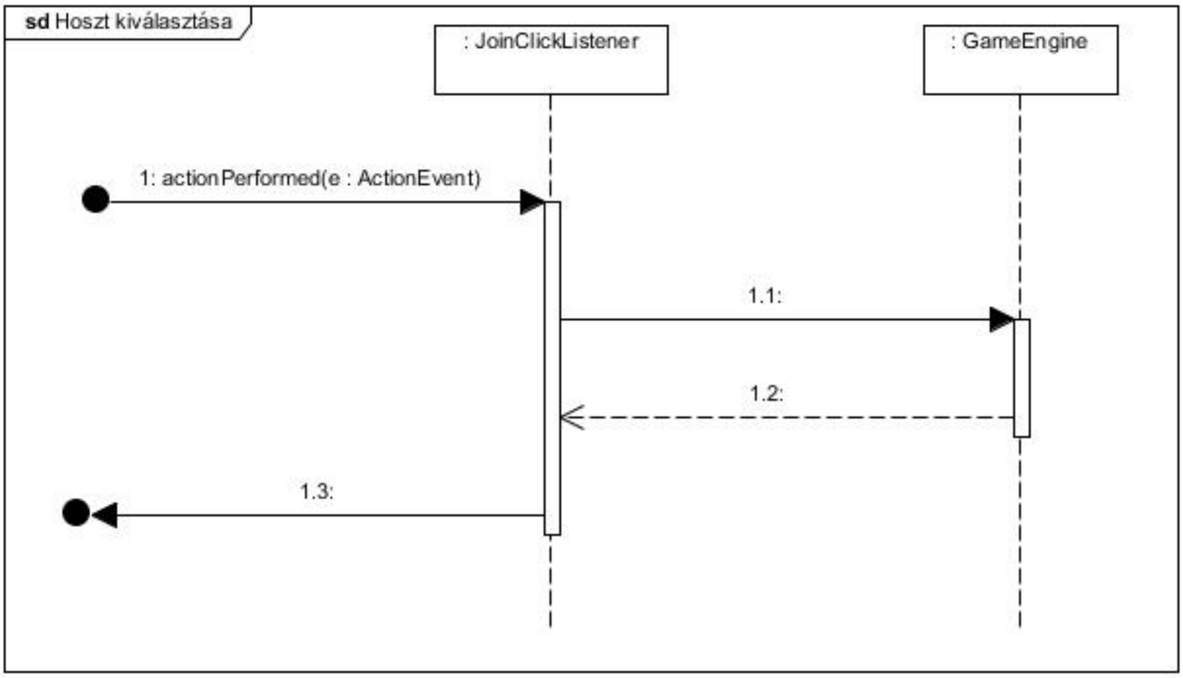
FACTORY FEJLESZTÉSE



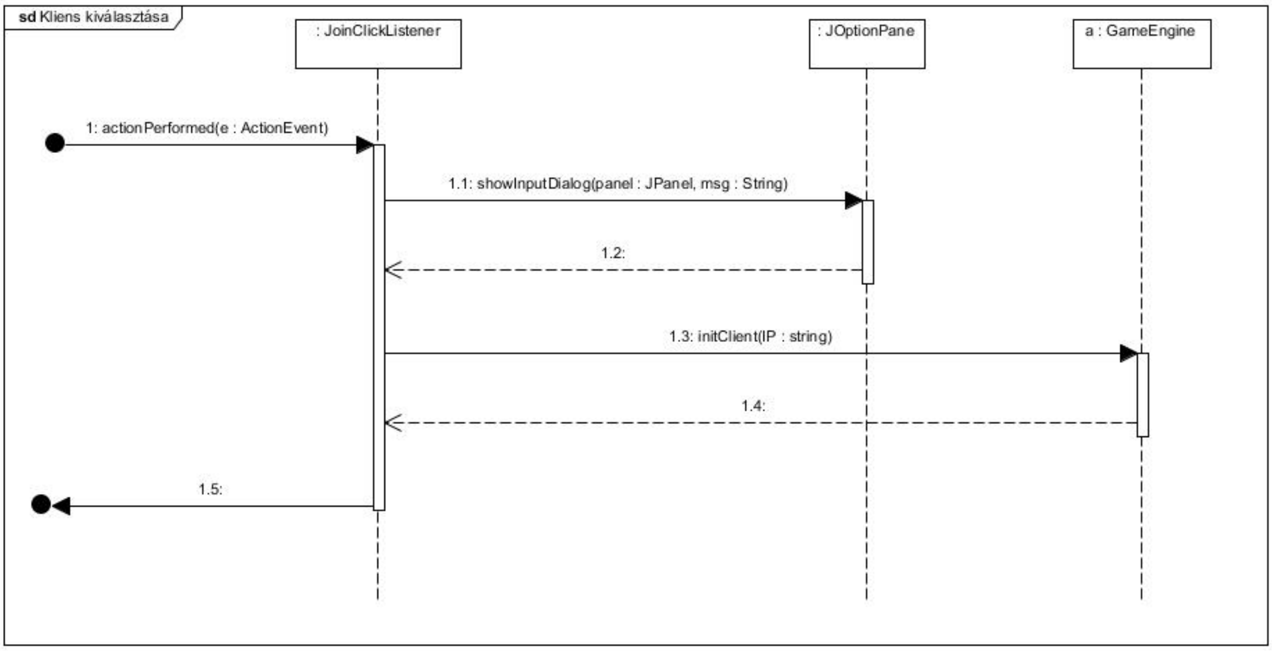
FACTORY KIVÁLASZTÁSA


sd Factory kiválasztása
: MouseListener
: GamePanel
1: mousePressed(e :MouseEvent)
1.1: selectFactory(type : ElementType)
1.2:
1.3:

HOSZT KIVÁLASZTÁSA



KLIENS KIVÁLASZTÁSA


sd Kliens kiválasztása
: JoinClickListener
: JOptionPane
a : GameEngine
1: actionPerformed(e : ActionEvent)
1.1: showInputDialog(panel : JPanel, msg : String)
1.2:
1.3: initClient(IP : string)
1.4:
1.5:

GENERÁLT JAVADOC

Table Of Content

Package misac

Class Summary

Main

Main class to launch the GameEngine and add Renderers

jn

---

misac

Class Main

```
java.lang.Object
    |
    +--misac.Main
```

< Constructors > < Methods >

public class **Main**
extends java.lang.Object

Main class to launch the GameEngine and add Renderers

Author:

Szabó Levente

Constructors

Main

```
public  Main()
```

Methods

main

```
public static void main(java.lang.String[] args)
```

misac

Class jn

```
java.lang.Object
    |
    +--misac.jn
```

< Constructors > < Methods >

public class **jn**
extends java.lang.Object

Constructors

jn

```
public  jn()
```

Create the application.

Methods

main

```
public static void main(java.lang.String[] args)
```

Launch the application.

Package misac.controller

Class Summary

GameEngine

Responsable for the main game logic

GameEngineTest

NetClient

Client

NetHelper

NetServer

Server

Points

Point for various action in the game

---

misac.controller

Class GameEngine

```
java.lang.Object
    |
    +--misac.controller.GameEngine
```

< Fields > < Constructors > < Methods >

public class **GameEngine**
extends java.lang.Object

Responsable for the main game logic

Author:

Szabó Levente

Fields

Version

```
public static int Version
```

gameModel

```
public GameModel gameModel
```

isServer

```
public boolean isServer
```

turnTime

```
public static int turnTime
```

Constructors

GameEngine

```
public  GameEngine()
```

Methods

addRenderer

```
public void addRenderer(Renderer renderer)
```

Adds a renderer Note: Observer pattern

Parameters:

renderer -

calculateElementsVisibility

```
public void calculateElementsVisibility()
```

Calculate each element's visibility in the map.

exitToMenu

```
public void exitToMenu()
```

Exits to the menu

initClient

```
public void initClient(java.lang.String IP)
                throws java.net.UnknownHostException,
                               java.io.IOException
```

Initializes the client.

Parameters:

IP - The server IP to connect to

Throws:

java.net.UnknownHostException - UnknownHostException is thrown when the host is not present.
java.io.IOException - IOException is thrown when networking error occures.

---

initServer

```
public void initServer()
                throws java.io.IOException
```

Initializes the server.

Throws:

java.io.IOException - IOException is thrown when networking error occures.

---

loadModelFromServer

```
public void loadModelFromServer(GameModel other)
```

Loads the incoming model

Parameters:

other - The incoming model

---

setElement

```
public void setElement(int i,
                       int j,
                       ElementType type,
                       ElementOwner owner)
```

Parameters:

i - X coordinate
j - Y coordinate
type - Element type to set
owner - Element owner to set

---

stopServer

```
public void stopServer()
```

Stops the server (when cancelling)

upgradeFactory

```
public void upgradeFactory(ElementType type)
```

Updgrades a factory

Parameters:

type - The type of factory to upgrade

misac.controller

Class GameEngineTest

```
java.lang.Object
    |
    +--misac.controller.GameEngineTest
```

< Fields > < Constructors > < Methods >

public class **GameEngineTest**
extends java.lang.Object

Fields

rendercalled

```
public static boolean rendercalled
```

Constructors

GameEngineTest

```
public  GameEngineTest()
```

Methods

testRender

```
public void testRender()
```

testSetElement

```
public void testSetElement()
```

testUpgradeFactory

```
public void testUpgradeFactory()
```

misac.controller

Class NetClient

```
java.lang.Object
    |
    +--misac.controller.NetClient
```

< Constructors > < Methods >

public class **NetClient**
extends java.lang.Object

Client

Author:
Szabó Levente

Constructors

NetClient

```
public  NetClient(GameEngine gameEngine)
```

Methods

handleNet

```
public void handleNet()
```

init

```
public void init(java.lang.String IP)
```

stop

```
public void stop()
```

misac.controller

Class NetHelper

```
java.lang.Object
    |
    +--misac.controller.NetHelper
```

< Constructors > < Methods >

public class **NetHelper**
extends java.lang.Object

Constructors

NetHelper

```
public  NetHelper()
```

Methods

receiveModel

```
public static void receiveModel(java.io.InputStream s,
                                GameEngine gameEngine)
```

Deserialize GameEngine from inputstream

Parameters:

s - the InputStream
gameEngine - the GameEngine

sendModel

```
public static void sendModel(java.io.OutputStream os,
                             GameEngine gameEngine)
```

Serialize GameEngine from inputstream

Parameters:

othe - OutputStream
gameEngine -

misac.controller

Class NetServer

```
java.lang.Object
    |
    +--misac.controller.NetServer
```

< Constructors > < Methods >

public class **NetServer**
extends java.lang.Object

Server

Author:

Szabó Levente

Constructors

NetServer

```
public  NetServer(GameEngine gameEngine)
```

Methods

handleNet

```
public void handleNet()
```

init

```
public void init()
```

stop

```
public void stop()
```

misac.controller

Class Points

```
java.lang.Object
    |
    +--misac.controller.Points
```

< Fields > < Constructors >

public class **Points**
extends java.lang.Object

Point for various action in the game

Author:
Szabó Levente

Fields

factoryCost

```
public static int factoryCost
```

forDestroyingMapElement

```
public static int forDestroyingMapElement
```

forDestroyingOtherElement

```
public static int forDestroyingOtherElement
```

forDestroyingSelfElement

```
public static int forDestroyingSelfElement
```

forOtherElementPowerDown

```
public static int forOtherElementPowerDown
```

forSelfElementPowerUp

```
public static int forSelfElementPowerUp
```

Constructors

Points

```
public  Points()
```

Package misac.model

Class Summary

Factory

Factory that can have elements and updgrades

GameModel

Game model for storing actual game state

GameState

Game states

Map

Map contains elements in a grid

misac.model

Class Factory

```
java.lang.Object
    |
    +--misac.model.Factory
```

All Implemented Interfaces:

java.io.Serializable

< Constructors > < Methods >

public class **Factory**
extends java.lang.Object
implements java.io.Serializable

Factory that can have elements and updgrades

Author:

Szabó Levente

Constructors

Factory

```
public  Factory()
```

Methods

getElements

```
public int getElements()
```

---

getLevel

```
public int getLevel()
```

---

regenerateElements

```
public void regenerateElements()
```

---

upgrade

```
public void upgrade()
```

---

useElement

```
public void useElement()
```

---

misac.model

Class GameModel

```
java.lang.Object
    |
    +--misac.model.GameModel
```

All Implemented Interfaces:
java.io.Serializable

< Fields > < Constructors >

---

public class **GameModel**
extends java.lang.Object
implements java.io.Serializable

Game model for storing actual game state

Author:

Szabó Levente

Fields

factoriesOther

`public java.util.HashMap **factoriesOther**`

public java.util.HashMap **factoriesOther**

---

factoriesSelf

public java.util.HashMap **factoriesSelf**

---

gameState

public GameState **gameState**

---

map

public Map **map**

---

pointsOther

public int **pointsOther**

---

pointsSelf

public int **pointsSelf**

---

time

public int **time**

Constructors

GameModel

public **GameModel**()

---

misac.model

Class GameState

```
java.lang.Object
    |
    +--java.lang.Enum
        |
        +--misac.model.GameState
```

All Implemented Interfaces:

java.io.Serializable, java.lang.Comparable

< Fields > < Methods >

public final class **GameState**
extends java.lang.Enum

Game states

Author:

Szabó Levente

Fields

End

`public static final GameState End`

Menu

`public static final GameState Menu`

OtherPlayerTurn

`public static final GameState OtherPlayerTurn`

SendMap

`public static final GameState SendMap`

SendMapAndEnd

`public static final GameState SendMapAndEnd`

Turn

```
public static final GameState Turn
```

WaitingForClient

```
public static final GameState WaitingForClient
```

Methods

valueOf

```
public static GameState valueOf(java.lang.String name)
```

values

```
public static misac.model.GameState[] values()
```

misac.model

Class Map

```
java.lang.Object
     |
     +--misac.model.Map
```

All Implemented Interfaces:
java.io.Serializable

< Fields > < Constructors > < Methods >

public class **Map**
extends java.lang.Object
implements java.io.Serializable

Map contains elements in a grid

Author:
Szabó Levente

Fields

elements

```
public misac.model.Elements.Element[][] elements
```

Constructors

Map

```
public  Map()
```

Methods

addDestroyedElement

```
public void addDestroyedElement(java.awt.Point element)
```

Add a new position to the destroyed elements for the default time

Parameters:

element - The element

---

addDestroyedElement

```
public void addDestroyedElement(java.awt.Point element,
                                int time)
```

Add a new position to the destroyed elements for a specified time

Parameters:

element - The element
time - The time in ms

---

clearDestroyedElements

```
public void clearDestroyedElements()
```

Removes all destroyed elements

---

decreaseDestroyedElementDisplayLeft

```
public void decreaseDestroyedElementDisplayLeft()
```

Decreases destroyed elements left time

getDestroyedElements

```
public java.util.ArrayList getDestroyedElements()
```

Gets the destroyed element positions

Returns:

The destroyes elements

getHeight

```
public int getHeight()
```

getWidth

```
public int getWidth()
```

makeMap

```
public void makeMap()
```

Make new map

switchOwner

```
public void switchOwner()
```

Changes element owners to the oposite Self -> Other Other -> Self

Package misac.model.Elements

Class Summary

Base

Earth

Element

Base of all elements

ElementOwner

Element owner

ElementType

Element types

Empty

Fire

Magical

Metal

Tree

Water

---

misac.model.Elements

Class Base

```
java.lang.Object
     |
     +--Element
          |
          +--misac.model.Elements.Base
```

All Implemented Interfaces:

java.io.Serializable

< Constructors > < Methods >

public class **Base**
extends Element

Constructors

Base

```
public  Base()
```

Methods

getDefaultPower

```
public byte getDefaultPower()
```

Overrides:

getDefaultPower in class Element

---

misac.model.Elements

Class Earth

```
java.lang.Object
     |
     +--Element
          |
          +--misac.model.Elements.Earth
```

All Implemented Interfaces:

java.io.Serializable

< Constructors > < Methods >

---

public class **Earth**
extends Element

Constructors

Earth

```
public  Earth()
```

Methods

isGenerate

```
public boolean isGenerate(ElementType other)
```

Overrides:

isGenerate in class Element

---

isOvercome

```
public boolean isOvercome(ElementType other)
```

Overrides:

isOvercome in class Element

---

misac.model.Elements

Class Element

```
java.lang.Object
    |
    +--misac.model.Elements.Element
```

All Implemented Interfaces:

java.io.Serializable

Direct Known Subclasses:

Base, Earth, Empty, Fire, Magical, Metal, Tree, Water

< Constructors > < Methods >

---

public abstract class **Element**
extends java.lang.Object
implements java.io.Serializable

Base of all elements

Author:

Szabó Levente

Constructors

Element

```
public  Element()
```

Methods

fromTypeAndOwnerFactory

```
public static Element fromTypeAndOwnerFactory(ElementType type,
                                              ElementOwner owner)
```

fromTypeFactory

```
public static Element fromTypeFactory(ElementType type)
```

getDefaultPower

```
public byte getDefaultPower()
```

getElementType

```
public ElementType getElementType()
```

getOwner

```
public ElementOwner getOwner()
```

getPower

```
public byte getPower()
```

isGenerate

```
public boolean isGenerate(ElementType other)
```

isOvercome

```
public boolean isOvercome(ElementType other)
```

isVisible

```
public boolean isVisible()
```

setOwner

```
public void setOwner(ElementOwner owner)
```

setPower

```
public void setPower(byte power)
```

setVisibiliy

```
public void setVisibiliy(boolean visible)
```

misac.model.Elements

Class ElementOwner

```
java.lang.Object
  |
  +--java.lang.Enum
        |
        +--misac.model.Elements.ElementOwner
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

< Fields > < Methods >

public final class **ElementOwner**
extends java.lang.Enum

Element owner

Author:

Szabó Levente

Fields

Map

```
public static final ElementOwner Map
```

Other

```
public static final ElementOwner Other
```

Self

```
public static final ElementOwner Self
```

Methods

valueOf

```
public static ElementOwner valueOf(java.lang.String name)
```

---

values

```
public static misac.model.Elements.ElementOwner[] values()
```

---

misac.model.Elements

Class ElementType

```
java.lang.Object
   |
   +--java.lang.Enum
         |
         +--misac.model.Elements.ElementType
```

All Implemented Interfaces:

java.io.Serializable, java.lang.Comparable

< Fields > < Methods >

---

public final class **ElementType**
extends java.lang.Enum

Element types

Author:

Szabó Levente

Fields

Base

```
public static final ElementType Base
```

Earth

```
public static final ElementType Earth
```

Empty

```
public static final ElementType Empty
```

Fire

```
public static final ElementType Fire
```

Magical

```
public static final ElementType Magical
```

Metal

```
public static final ElementType Metal
```

Water

```
public static final ElementType Water
```

Wood

```
public static final ElementType Wood
```

Methods

valueOf

```
public static ElementType valueOf(java.lang.String name)
```

values

```
public static misac.model.Elements.ElementType[] values()
```

misac.model.Elements

Class Empty

```
java.lang.Object
     |
     +--Element
          |
          +--misac.model.Elements.Empty
```

All Implemented Interfaces:
java.io.Serializable

< Constructors >

public class **Empty**
extends Element

Constructors

Empty

```
public  Empty()
```

misac.model.Elements

Class Fire

```
java.lang.Object
  |
  +--Element
      |
      +--misac.model.Elements.Fire
```

All Implemented Interfaces:

java.io.Serializable

< Constructors > < Methods >

public class **Fire**
extends Element

Constructors

Fire

```
public  Fire()
```

Methods

isGenerate

```
public boolean isGenerate(ElementType other)
```

Overrides:

isGenerate in class Element

isOvercome

```
public boolean isOvercome(ElementType other)
```

Overrides:

isOvercome in class Element

misac.model.Elements

Class Magical

```
java.lang.Object
  |
  +--Element
      |
      +--misac.model.Elements.Magical
```

All Implemented Interfaces:
java.io.Serializable

< Constructors >

public class **Magical**
extends Element

Constructors

Magical

```
public  Magical()
```

misac.model.Elements

Class Metal

```
java.lang.Object
  |
  +--Element
      |
      +--misac.model.Elements.Metal
```

All Implemented Interfaces:
java.io.Serializable

< Constructors > < Methods >

public class **Metal**
extends Element

Constructors

Metal

```
public  Metal()
```

Methods

isGenerate

```
public boolean isGenerate(ElementType other)
```

Overrides:

isGenerate in class Element

---

isOvercome

```
public boolean isOvercome(ElementType other)
```

Overrides:

isOvercome in class Element

---

misac.model.Elements

Class Tree

```
java.lang.Object
    |
    +--Element
        |
        +--misac.model.Elements.Tree
```

All Implemented Interfaces:

java.io.Serializable

< Constructors > < Methods >

---

public class **Tree**
extends Element

Constructors

Tree

```
public  Tree()
```

Methods

isGenerate

```
public boolean isGenerate(ElementType other)
```

Overrides:

isGenerate in class Element

---

isOvercome

```
public boolean isOvercome(ElementType other)
```

Overrides:

isOvercome in class Element

---

misac.model.Elements

Class Water

```
java.lang.Object
   |
   +--Element
        |
        +--misac.model.Elements.Water
```

All Implemented Interfaces:

java.io.Serializable

< Constructors > < Methods >

---

public class **Water**
extends Element

Constructors

Water

```
public  Water()
```

Methods

isGenerate

```
public boolean isGenerate(ElementType other)
```

Overrides:

isGenerate in class Element

---

isOvercome

```
public boolean isOvercome(ElementType other)
```

Overrides:

isOvercome in class Element

Package misac.view

Interface Summary

Renderer

Render interface for GameEngine

Class Summary

DefaultRenderer

Default renderer for Misac Displays game in a window Manages menu and gameplay screen

GameMenuPanel

Displays main menu

GamePanel

Displays the game Manages interact with the GameEngine

GamePanelTest

---

misac.view

Class DefaultRenderer

```
java.lang.Object
    |
    +--misac.view.DefaultRenderer
```

All Implemented Interfaces:

Renderer

< Constructors > < Methods >

public class **DefaultRenderer**
extends java.lang.Object
implements Renderer

Default renderer for Misac Displays game in a window Manages menu and gameplay screen

Author:

Szabó Levente

Constructors

DefaultRenderer

```
public  DefaultRenderer()
```

Methods

AddedToGameEngine

```
public void AddedToGameEngine(GameEngine ge)
```

Added to a GameEngine Gets gameModel

Render

```
public void Render()
```

Render GameModel

---

misac.view

Class GameMenuPanel

```
java.lang.Object
    |
    +--java.awt.Component
        |
        +--java.awt.Container
            |
            +--javax.swing.JComponent
                |
                +--javax.swing.JPanel
                    |
                    +--misac.view.GameMenuPanel
```

All Implemented Interfaces:

java.awt.MenuContainer, java.awt.image.ImageObserver, java.io.Serializable, javax.accessibility.Accessible, javax.swing.TransferHandler.HasGetTransferHandler

---

< Constructors > < Methods >

---

public class **GameMenuPanel**
extends javax.swing.JPanel

Displays main menu

Author:

Szabó Levente

Constructors

GameMenuPanel

```
public  GameMenuPanel(GameEngine gameEngine)
```

Methods

getPreferredSize

```
public java.awt.Dimension getPreferredSize()
```

Overrides:

getPreferredSize in class javax.swing.JComponent

---

misac.view

Class GamePanel

```
java.lang.Object
  |
  +--java.awt.Component
        |
        +--java.awt.Container
              |
              +--javax.swing.JComponent
                    |
                    +--javax.swing.JPanel
                          |
                          +--misac.view.GamePanel
```

All Implemented Interfaces:

java.awt.MenuContainer, java.awt.image.ImageObserver, java.io.Serializable, javax.accessibility.Accessible, javax.swing.TransferHandler.HasGetTransferHandler

---

< Constructors > < Methods >

---

public class **GamePanel**
extends javax.swing.JPanel

Displays the game Manages interact with the GameEngine

Author:

Szabó Levente

Constructors

GamePanel

```
public  GamePanel(GameEngine gameEngine)
```

Initializes java Components and instance variables

Parameters:

gameEngine -

Methods

gameRender

```
public void gameRender()
```

Updates explositions (destroyed elements) Called when game requests render

generateNiceBackgroundImage

```
public static java.awt.Image generateNiceBackgroundImage()
```

Generates a nice background image

Returns:

Returns a 500x500 Image

getPreferredSize

```
public java.awt.Dimension getPreferredSize()
```

Overrides:

getPreferredSize in class javax.swing.JComponent

misac.view

Class GamePanelTest

```
java.lang.Object
   |
   +--misac.view.GamePanelTest
```

< Constructors > < Methods >

public class **GamePanelTest**
extends java.lang.Object

Constructors

GamePanelTest

`public  GamePanelTest()`

Methods

testGamePanel

`public void testGamePanel()`

---

testGenerateNiceBackgroundImage

`public void testGenerateNiceBackgroundImage()`

---

testGetPreferredSize

`public void testGetPreferredSize()`

---

misac.view

Interface Renderer

< Methods >

---

public interface **Renderer**

Render interface for GameEngine

Author:

Szabó Levente

Methods

AddedToGameEngine

```
public void AddedToGameEngine(GameEngine ge)
```

Called, when added to a GameEngine

Parameters:

ge - The GameEngine

Render

```
public void Render()
```

Renders the GameModel

INDEX