Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ

УНИВЕРСИТЕТ ИНФОРМАТИКИ И

РАДИОЭЛЕКТРОНИКИ

Кафедра ЭВМ

Лабораторная работа №2

Тема: «Программирование алгоритмов с использованием механизмов объектно-

ориентированного программирования».

Проверил: Выполнил:

Кухарчук И.В. Студент гр. №450503

Федорцов К.С.

Минск 2016

Введение

Необходимо разработать программу для игры в Сёги. Необходимо предусмотреть возможность сохранения, разработать интерфейс, разработать разделение на сервер и клиент.

Компоновка классов

Для любой игры, где роль одного из игроков выполняет компьютер, необходимо создать класс, который будет отвечать за хранение текущего состояния игры, а также будет содержать алгоритмы для игры против другого игрока. В случае сложных игр, где в игровом пространстве (2-х или 3-х мерном) содержатся объекты, которые могут сильно различаться по свойствам, имеет смысл создать класс для этих объектов и хранить их массив/vector/очередь в этом классе (например, в игре «морской бой» в классе EnemyField хранить массив 10 объектов класса Ship). В случае более простых игр, текущее состояние игрового поля можно хранить в целочисленном массиве. Роль такого класса в программе «крестики-нолики» играет GameLogic. Он содержит в себе основной алгоритм, который в зависимости от текущих условий вызывает другие функции класса для принятия решения о следующем ходе.

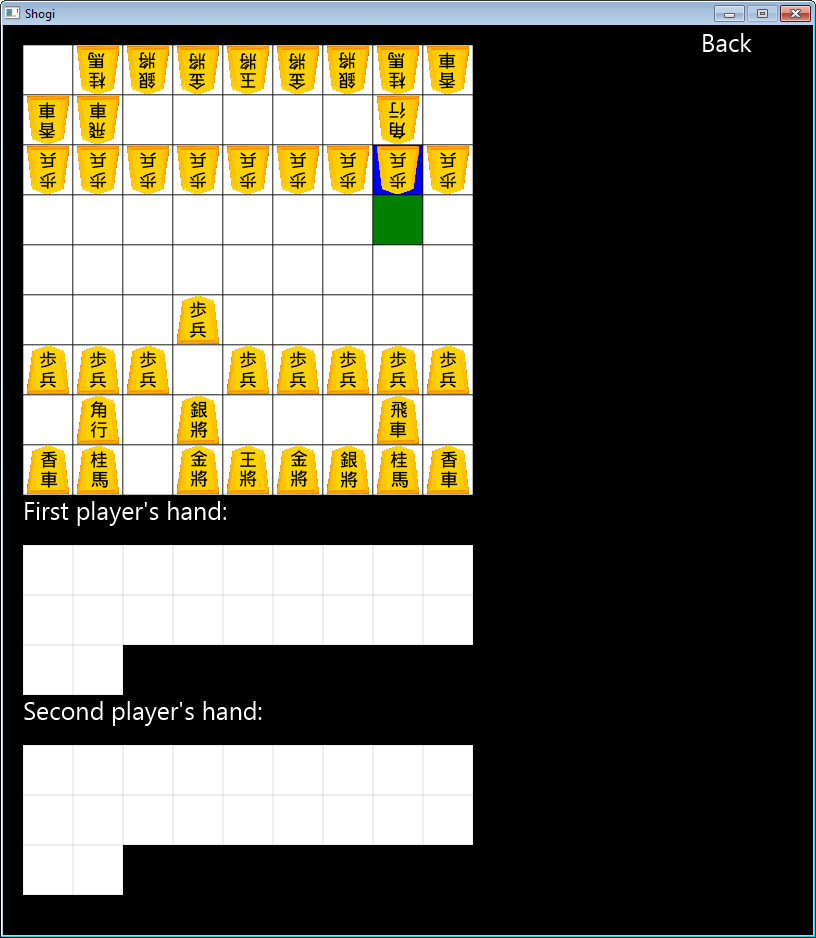
Графический интерфейс программы состоит из массива кнопок, которые представляют из себя игровое поле. Для обеспечения правильного функционирования необходимо создать класс для контроля этого массива: установка рисунка крестика или нолика при совершении хода, блокировка кнопок в конце игры, вызов метода класса GameLogic с передачей и возвратом координат, сброс кнопок в исходное состояние.

Чтобы не перегружать класс main, автор работы решил вынести бòльшую часть кода, в котором создаются и настраиваются элементы GUI, в отдельный класс UserInterface. В main создается его экземпляр и вызывается метод init.

Чтобы не объявлять в всех классах один и тот же набор констант, создан интерфейс, который их содержит.

Для реализации возможности сохранения и воспроизведения игры, создан класс GameReplay, который содержит список состояний. Во время игры каждое состояние сохраняется в список, а при сохранении в файл происходит чтение из списка. При воспроизведении файл считывается в список, а затем происходит чтение из списка каждого состояния и его отображение.





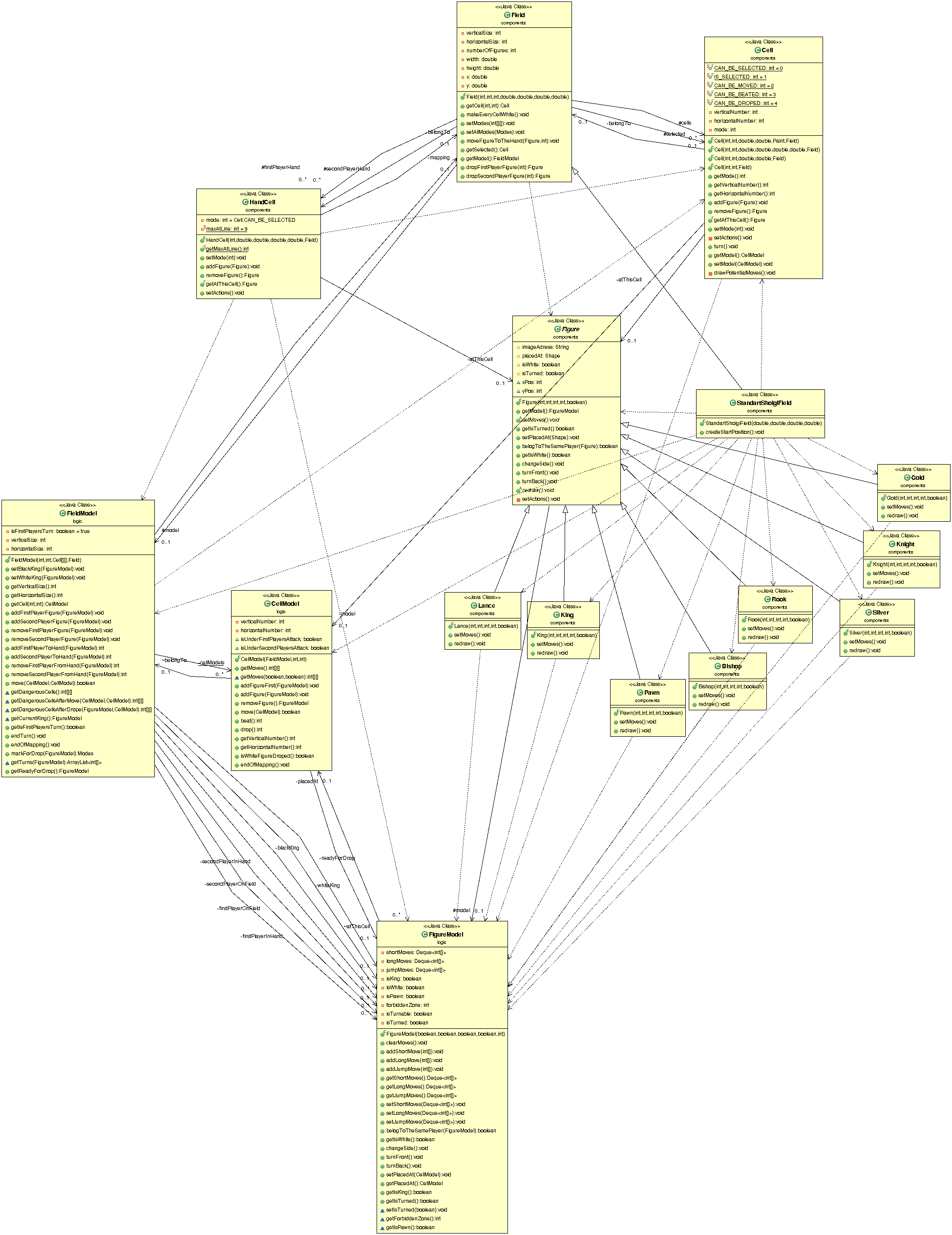
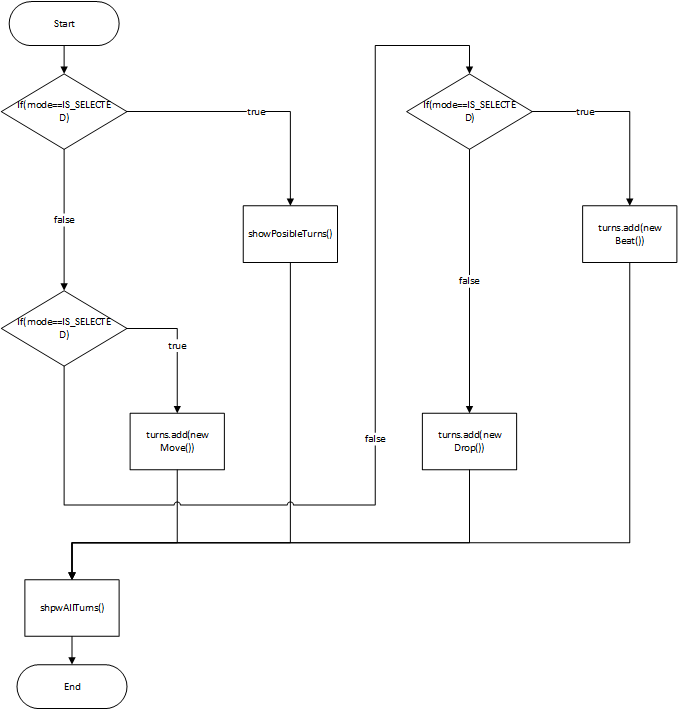


Диаграмма классов

Блок схема основного алгоритма приложения

Автоматически генерируемая документация

application

## Class GameWindow

* java.lang.Object
  + javafx.scene.Node
    - javafx.scene.Parent
      * javafx.scene.layout.Region
        + javafx.scene.layout.Pane

javafx.scene.layout.BorderPane

application.GameWindow

* **All Implemented Interfaces:**

javafx.css.Styleable, javafx.event.EventTarget

public class **GameWindow**

extends javafx.scene.layout.BorderPane

|  |
| --- |
| **Constructors** |
| **Constructor and Description** |
| [**GameWindow**](file:///E:Laba1docapplicationGameWindow.html#GameWindow--)() |

### *Method Summary*

### *Constructor Detail*

#### GameWindow

public GameWindow()

application

## Class Main

* java.lang.Object
  + javafx.application.Application
    - application.Main

public class **Main**

extends javafx.application.Application

### *Constructor Summary*

|  |
| --- |
| **Constructors** |
| **Constructor and Description** |
| [**Main**](file:///E:Laba1docapplicationMain.html#Main--)() |

### *Method Summary*

|  |  |
| --- | --- |
| **All Methods**[**Static Methods**](javascript:show(1);)[**Instance Methods**](javascript:show(2);)[**Concrete Methods**](javascript:show(8);) | |
| **Modifier and Type** | **Method and Description** |
| static void | [**main**](file:///E:Laba1docapplicationMain.html#main-java.lang.String:A-)(java.lang.String[] args) |
| void | [**start**](file:///E:Laba1docapplicationMain.html#start-javafx.stage.Stage-)(javafx.stage.Stage primaryStage) |

### *Constructor Detail*

#### Main

public Main()

### *Method Detail*

#### start

public void start(javafx.stage.Stage primaryStage)

**Specified by:**

start in class javafx.application.Application

#### main

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

application

## Class MenuWindow

* java.lang.Object
  + javafx.scene.Node
    - javafx.scene.Parent
      * javafx.scene.layout.Region
        + javafx.scene.layout.Pane

javafx.scene.layout.BorderPane

application.MenuWindow

* **All Implemented Interfaces:**

javafx.css.Styleable, javafx.event.EventTarget

public class **MenuWindow**

extends javafx.scene.layout.BorderPane

*Summary*

|  |
| --- |
| **Constructors** |
| **Constructor and Description** |
| [**MenuWindow**](file:///E:Laba1docapplicationMenuWindow.html#MenuWindow--)() |

### *Constructor Detail*

#### MenuWindow

public MenuWindow()

application

## Class NewGameSettingsWindow

* java.lang.Object
  + javafx.scene.Node
    - javafx.scene.Parent
      * javafx.scene.layout.Region
        + javafx.scene.layout.Pane

javafx.scene.layout.BorderPane

application.NewGameSettingsWindow

* **All Implemented Interfaces:**

javafx.css.Styleable, javafx.event.EventTarget

public class **NewGameSettingsWindow**

extends javafx.scene.layout.BorderPane

|  |
| --- |
| **Constructors** |
| **Constructor and Description** |
| [**NewGameSettingsWindow**](file:///E:Laba1docapplicationNewGameSettingsWindow.html#NewGameSettingsWindow--)() |

### *Constructor Detail*

#### NewGameSettingsWindow

public NewGameSettingsWindow()

application

## Class RulesText

* java.lang.Object
  + javafx.scene.Node
    - javafx.scene.shape.Shape
      * javafx.scene.text.Text
        + application.RulesText
* **All Implemented Interfaces:**

javafx.css.Styleable, javafx.event.EventTarget

public class **RulesText**

extends javafx.scene.text.Text

### *Constructor Summary*

|  |
| --- |
| **Constructors** |
| **Constructor and Description** |
| [**RulesText**](file:///E:Laba1docapplicationRulesText.html#RulesText-java.lang.String-)(java.lang.String text) |

### *Constructor Detail*

#### RulesText

public RulesText(java.lang.String text)

application

## Class RulesWindow

* java.lang.Object
  + javafx.scene.Node
    - javafx.scene.Parent
      * javafx.scene.layout.Region
        + javafx.scene.layout.Pane

javafx.scene.layout.BorderPane

application.RulesWindow

* **All Implemented Interfaces:**

javafx.css.Styleable, javafx.event.EventTarget

public class **RulesWindow**

extends javafx.scene.layout.BorderPane

### *Constructor Summary*

|  |
| --- |
| **Constructors** |
| **Constructor and Description** |
| [**RulesWindow**](file:///E:Laba1docapplicationRulesWindow.html#RulesWindow--)() |

### *Constructor Detail*

#### RulesWindow

public RulesWindow()

application

## Class Storage

* java.lang.Object
  + application.Storage

public abstract class **Storage**

extends java.lang.Object

### *Field Summary*

|  |  |
| --- | --- |
| **Fields** | |
| **Modifier and Type** | **Field and Description** |
| static int[] | [**BACKWARD**](file:///E:Laba1docapplicationStorage.html#BACKWARD) |
| static int[] | [**BACKWARD\_LEFT**](file:///E:Laba1docapplicationStorage.html#BACKWARD_LEFT) |
| static int[] | [**BACKWARD\_RIGHT**](file:///E:Laba1docapplicationStorage.html#BACKWARD_RIGHT) |
| static double | [**BUTTON\_HEIGHT**](file:///E:Laba1docapplicationStorage.html#BUTTON_HEIGHT) |
| static double | [**BUTTON\_WIDTH**](file:///E:Laba1docapplicationStorage.html#BUTTON_WIDTH) |
| static double | [**FIELD\_SIZE**](file:///E:Laba1docapplicationStorage.html#FIELD_SIZE) |
| static java.lang.String[] | [**FIGURE\_NAMES**](file:///E:Laba1docapplicationStorage.html#FIGURE_NAMES) |
| static int | [**FIGURES\_IN\_COLUMN**](file:///E:Laba1docapplicationStorage.html#FIGURES_IN_COLUMN) |
| static java.lang.String | [**FIRST\_PAGE\_TEXT**](file:///E:Laba1docapplicationStorage.html#FIRST_PAGE_TEXT) |
| static int[] | [**FORWARD**](file:///E:Laba1docapplicationStorage.html#FORWARD) |
| static int[] | [**FORWARD\_LEFT**](file:///E:Laba1docapplicationStorage.html#FORWARD_LEFT) |
| static int[] | [**FORWARD\_RIGHT**](file:///E:Laba1docapplicationStorage.html#FORWARD_RIGHT) |
| static boolean | [**isFirstPlayerAComputer**](file:///E:Laba1docapplicationStorage.html#isFirstPlayerAComputer) |
| static boolean | [**isSecondPlayerAComputer**](file:///E:Laba1docapplicationStorage.html#isSecondPlayerAComputer) |
| static int[] | [**JUMP\_BACKWARD\_LEFT**](file:///E:Laba1docapplicationStorage.html#JUMP_BACKWARD_LEFT) |
| static int[] | [**JUMP\_BACKWARD\_RIGHT**](file:///E:Laba1docapplicationStorage.html#JUMP_BACKWARD_RIGHT) |
| static int[] | [**JUMP\_FORWARD\_LEFT**](file:///E:Laba1docapplicationStorage.html#JUMP_FORWARD_LEFT) |
| static int[] | [**JUMP\_FORWARD\_RIGHT**](file:///E:Laba1docapplicationStorage.html#JUMP_FORWARD_RIGHT) |
| static int[] | [**LEFT**](file:///E:Laba1docapplicationStorage.html#LEFT) |
| static int[] | [**RIGHT**](file:///E:Laba1docapplicationStorage.html#RIGHT) |
| static java.lang.String[] | [**TURNED\_FIGURE\_NAMES**](file:///E:Laba1docapplicationStorage.html#TURNED_FIGURE_NAMES) |

### *Constructor Summary*

|  |
| --- |
| **Constructors** |
| **Constructor and Description** |
| [**Storage**](file:///E:Laba1docapplicationStorage.html#Storage--)() |

### *Method Summary*

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### *Field Detail*

#### BUTTON\_HEIGHT

public static final double BUTTON\_HEIGHT

**See Also:**

[Constant Field Values](file:///E:Laba1docconstant-values.html#application.Storage.BUTTON_HEIGHT)

#### BUTTON\_WIDTH

public static final double BUTTON\_WIDTH

**See Also:**

[Constant Field Values](file:///E:Laba1docconstant-values.html#application.Storage.BUTTON_WIDTH)

#### FIELD\_SIZE

public static final double FIELD\_SIZE

**See Also:**

[Constant Field Values](file:///E:Laba1docconstant-values.html#application.Storage.FIELD_SIZE)

#### FIGURES\_IN\_COLUMN

public static final int FIGURES\_IN\_COLUMN

**See Also:**

[Constant Field Values](file:///E:Laba1docconstant-values.html#application.Storage.FIGURES_IN_COLUMN)

#### FIRST\_PAGE\_TEXT

public static final java.lang.String FIRST\_PAGE\_TEXT

**See Also:**

[Constant Field Values](file:///E:Laba1docconstant-values.html#application.Storage.FIRST_PAGE_TEXT)

#### FIGURE\_NAMES

public static final java.lang.String[] FIGURE\_NAMES

#### TURNED\_FIGURE\_NAMES

public static final java.lang.String[] TURNED\_FIGURE\_NAMES

#### BACKWARD\_LEFT

public static final int[] BACKWARD\_LEFT

#### BACKWARD

public static final int[] BACKWARD

#### BACKWARD\_RIGHT

public static final int[] BACKWARD\_RIGHT

#### LEFT

public static final int[] LEFT

#### RIGHT

public static final int[] RIGHT

#### FORWARD\_LEFT

public static final int[] FORWARD\_LEFT

#### FORWARD

public static final int[] FORWARD

#### FORWARD\_RIGHT

public static final int[] FORWARD\_RIGHT

#### JUMP\_FORWARD\_RIGHT

public static final int[] JUMP\_FORWARD\_RIGHT

#### JUMP\_FORWARD\_LEFT

public static final int[] JUMP\_FORWARD\_LEFT

#### JUMP\_BACKWARD\_RIGHT

public static final int[] JUMP\_BACKWARD\_RIGHT

#### JUMP\_BACKWARD\_LEFT

public static final int[] JUMP\_BACKWARD\_LEFT

#### isFirstPlayerAComputer

public static boolean isFirstPlayerAComputer

#### isSecondPlayerAComputer

public static boolean isSecondPlayerAComputer

### *Constructor Detail*

#### Storage

public Storage()

components

components

## Class Field

* java.lang.Object
  + javafx.scene.Node
    - javafx.scene.Parent
      * components.Field
* **All Implemented Interfaces:**

javafx.css.Styleable, javafx.event.EventTarget

**Direct Known Subclasses:**

[StandartShoigiField](file:///E:\Laba1\doc\components\StandartShoigiField.html)

public class **Field**

extends javafx.scene.Parent

### *Constructor Summary*

|  |
| --- |
| **Constructors** |
| **Constructor and Description** |
| [**Field**](file:///E:Laba1doccomponentsField.html#Field-int-int-int-double-double-double-double-)(int verticalSize, int horizontalSize, int numberOfFigures, double x, double y, double width, double height) |

### *Method Summary*

|  |  |
| --- | --- |
| **All Methods**[**Instance Methods**](javascript:show(2);)[**Concrete Methods**](javascript:show(8);) | |
| **Modifier and Type** | **Method and Description** |
| [**Figure**](file:///E:\Laba1\doc\components\Figure.html) | [**dropFirstPlayerFigure**](file:///E:Laba1doccomponentsField.html#dropFirstPlayerFigure-int-)(int index) |
| [**Figure**](file:///E:\Laba1\doc\components\Figure.html) | [**dropSecondPlayerFigure**](file:///E:Laba1doccomponentsField.html#dropSecondPlayerFigure-int-)(int index) |
| [**Cell**](file:///E:\Laba1\doc\components\Cell.html) | [**getCell**](file:///E:Laba1doccomponentsField.html#getCell-int-int-)(int horizontalNumber, int verticalNumber) |
| [**FieldModel**](file:///E:\Laba1\doc\logic\FieldModel.html) | [**getModel**](file:///E:Laba1doccomponentsField.html#getModel--)() |
| [**Cell**](file:///E:\Laba1\doc\components\Cell.html) | [**getSelected**](file:///E:Laba1doccomponentsField.html#getSelected--)() |
| void | [**makeEveryCellWhite**](file:///E:Laba1doccomponentsField.html#makeEveryCellWhite--)() |
| void | [**moveFigureToTheHand**](file:///E:Laba1doccomponentsField.html#moveFigureToTheHand-components.Figure-int-)([**Figure**](file:///E:\Laba1\doc\components\Figure.html) figure, int number) |
| void | [**setAllModes**](file:///E:Laba1doccomponentsField.html#setAllModes-logic.FieldModel.Modes-)([**FieldModel.Modes**](file:///E:\\Laba1\\doc\\logic\\FieldModel.Modes.html) modes) |
| void | [**setModes**](file:///E:Laba1doccomponentsField.html#setModes-int:A:A-)(int[][] modes) |

### *Constructor Detail*

#### Field

* + - public Field(int verticalSize,
    - int horizontalSize,
    - int numberOfFigures,
    - double x,
    - double y,
    - double width,

double height)

### *Method Detail*

#### getCell

* + - public [Cell](file:///E:\Laba1\doc\components\Cell.html) getCell(int horizontalNumber,

int verticalNumber)

#### makeEveryCellWhite

public void makeEveryCellWhite()

#### setModes

public void setModes(int[][] modes)

#### setAllModes

public void setAllModes([FieldModel.Modes](file:///E:\\Laba1\\doc\\logic\\FieldModel.Modes.html) modes)

#### moveFigureToTheHand

* + - public void moveFigureToTheHand([Figure](file:///E:\Laba1\doc\components\Figure.html) figure,

int number)

#### getSelected

public [Cell](file:///E:\Laba1\doc\components\Cell.html) getSelected()

#### getModel

public [FieldModel](file:///E:\\Laba1\\doc\\logic\\FieldModel.html) getModel()

#### dropFirstPlayerFigure

public [Figure](file:///E:\Laba1\doc\components\Figure.html) dropFirstPlayerFigure(int index)

#### dropSecondPlayerFigure

public [Figure](file:///E:\Laba1\doc\components\Figure.html) dropSecondPlayerFigure(int index)

components

## Class Figure

* java.lang.Object
  + javafx.scene.Node
    - javafx.scene.image.ImageView
      * components.Figure
* **All Implemented Interfaces:**

javafx.css.Styleable, javafx.event.EventTarget

**Direct Known Subclasses:**

[Bishop](file:///E:\Laba1\doc\components\Bishop.html), [Gold](file:///E:\Laba1\doc\components\Gold.html), [King](file:///E:\Laba1\doc\components\King.html), [Knight](file:///E:\Laba1\doc\components\Knight.html), [Lance](file:///E:\Laba1\doc\components\Lance.html), [Pawn](file:///E:\Laba1\doc\components\Pawn.html), [Rook](file:///E:\Laba1\doc\components\Rook.html), [Silver](file:///E:\Laba1\doc\components\Silver.html)

public abstract class **Figure**

extends javafx.scene.image.ImageView

### *Constructor Summary*

|  |
| --- |
| **Constructors** |
| **Constructor and Description** |
| [**Figure**](file:///E:Laba1doccomponentsFigure.html#Figure-int-int-int-int-boolean-)(int x, int y, int height, int width, boolean isWhite) |

### *Method Summary*

|  |  |
| --- | --- |
| **All Methods**[**Instance Methods**](javascript:show(2);)[**Abstract Methods**](javascript:show(4);)[**Concrete Methods**](javascript:show(8);) | |
| **Modifier and Type** | **Method and Description** |
| boolean | [**belogToTheSamePlayer**](file:///E:Laba1doccomponentsFigure.html#belogToTheSamePlayer-components.Figure-)([**Figure**](file:///E:\\Laba1\\doc\\components\\Figure.html) compared) |
| void | [**changeSide**](file:///E:Laba1doccomponentsFigure.html#changeSide--)() |
| boolean | [**getIsTurned**](file:///E:Laba1doccomponentsFigure.html#getIsTurned--)() |
| boolean | [**getIsWhite**](file:///E:Laba1doccomponentsFigure.html#getIsWhite--)() |
| [**FigureModel**](file:///E:\Laba1\doc\logic\FigureModel.html) | [**getModel**](file:///E:Laba1doccomponentsFigure.html#getModel--)() |
| abstract void | [**redraw**](file:///E:Laba1doccomponentsFigure.html#redraw--)() |
| abstract void | [**setMoves**](file:///E:Laba1doccomponentsFigure.html#setMoves--)() |
| void | [**setPlacedAt**](file:///E:Laba1doccomponentsFigure.html#setPlacedAt-javafx.scene.shape.Shape-)(javafx.scene.shape.Shape placedAt) |
| void | [**turnBack**](file:///E:Laba1doccomponentsFigure.html#turnBack--)() |
| void | [**turnFront**](file:///E:Laba1doccomponentsFigure.html#turnFront--)() |

### *Constructor Detail*

#### Figure

* + - public Figure(int x,
    - int y,
    - int height,
    - int width,

boolean isWhite)

### *Method Detail*

#### getModel

public [FigureModel](file:///E:\\Laba1\\doc\\logic\\FigureModel.html) getModel()

#### setMoves

public abstract void setMoves()

#### getIsTurned

public boolean getIsTurned()

#### setPlacedAt

public void setPlacedAt(javafx.scene.shape.Shape placedAt)

#### belogToTheSamePlayer

public boolean belogToTheSamePlayer([Figure](file:///E:\Laba1\doc\components\Figure.html) compared)

#### getIsWhite

public boolean getIsWhite()

#### changeSide

public void changeSide()

#### turnFront

public void turnFront()

#### turnBack

public void turnBack()

#### redraw

public abstract void redraw()

## Class HandCell

* java.lang.Object
  + javafx.scene.Node
    - javafx.scene.shape.Shape
      * javafx.scene.shape.Rectangle
        + components.HandCell
* **All Implemented Interfaces:**

javafx.css.Styleable, javafx.event.EventTarget

public class **HandCell**

extends javafx.scene.shape.Rectangle

### *Constructor Summary*

|  |
| --- |
| **Constructors** |
| **Constructor and Description** |
| [**HandCell**](file:///E:Laba1doccomponentsHandCell.html#HandCell-int-double-double-double-double-components.Field-)(int number, double x, double y, double width, double height, [**Field**](file:///E:\Laba1\doc\components\Field.html) belongTo) |

### *Method Summary*

|  |  |
| --- | --- |
| **All Methods**[**Static Methods**](javascript:show(1);)[**Instance Methods**](javascript:show(2);)[**Concrete Methods**](javascript:show(8);) | |
| **Modifier and Type** | **Method and Description** |
| void | [**addFigure**](file:///E:Laba1doccomponentsHandCell.html#addFigure-components.Figure-)([**Figure**](file:///E:\\Laba1\\doc\\components\\Figure.html) figure) |
| [**Figure**](file:///E:\Laba1\doc\components\Figure.html) | [**getAtThisCell**](file:///E:Laba1doccomponentsHandCell.html#getAtThisCell--)() |
| static int | [**getMaxAtLine**](file:///E:Laba1doccomponentsHandCell.html#getMaxAtLine--)() |
| [**Figure**](file:///E:\Laba1\doc\components\Figure.html) | [**removeFigure**](file:///E:Laba1doccomponentsHandCell.html#removeFigure--)() |
| void | [**setActions**](file:///E:Laba1doccomponentsHandCell.html#setActions--)() |
| void | [**setMode**](file:///E:Laba1doccomponentsHandCell.html#setMode-int-)(int mode) |

### *Constructor Detail*

#### HandCell

* + - public HandCell(int number,
    - double x,
    - double y,
    - double width,
    - double height,

[Field](file:///E:\Laba1\doc\components\Field.html) belongTo)

### *Method Detail*

#### getMaxAtLine

public static int getMaxAtLine()

#### setMode

public void setMode(int mode)

#### addFigure

public void addFigure([Figure](file:///E:\Laba1\doc\components\Figure.html) figure)

#### removeFigure

public [Figure](file:///E:\Laba1\doc\components\Figure.html) removeFigure()

#### getAtThisCell

public final [Figure](file:///E:\Laba1\doc\components\Figure.html) getAtThisCell()

#### setActions

public void setActions()

Для ходов движения и взятия фигуры применяется следующая нотация:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Figure.toString() | from.toString() | turn.toString() | to.tostring() | + (если был совершён переворот) |

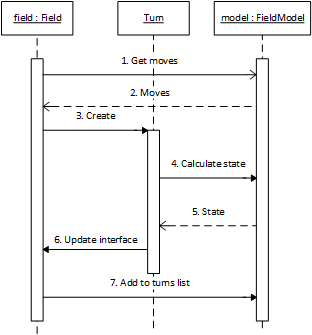
Figure.toString() – однобуквенный код фигуры (P – пешка, B – ладья, R – слон, L – колесница, K – король, N – конь, и.т.д.)

from.toString() – ячейка, из которой был совершён ход (например, a1). Отсутствует при сбросе.

turn – код действия («-» -- движение, «x» -- взятие, «\*» -- сброс)

to.tostring() – ячейка, в которую был совершён ход.

Диаграмма последовательностей:



Описание взаимодействия клиент-сервер:

1. После первого нажатия поле (клиент) отправляет модели поля (серверу) запрос на получение возможных ходов.
2. Модель вычисляет возможные ходы и возвращает их полю.
3. Поле создаёт объект класса, дочернего Turn, передавая ему параметры хода.
4. Объект Turn отправляет модели запрос на расчёт состояния после хода.
5. Модель считает состояние и возвращает её объекту Turn.
6. Turn отправляет полю запрос на обновление интерфейса.
7. Поле добавляет Turn в список ходов.

Время выполнения с учётом открытия файлов (мс).

|  |  |  |  |
| --- | --- | --- | --- |
|  | 1000 | 10,000 | 100,000 |
| Java | 592 | 5813 | 60120 |
| Scala | 580 | 5910 | 58245 |

Результаты поучились приблизительно равными, поскольку основной промежуток времени занимает открытие и чтение файлов, которое выполняется одинаково, как на Java.

Время чистой сортировки (мс).

|  |  |  |  |
| --- | --- | --- | --- |
|  | 1000 | 10,000 | 100,000 |
| Java | 2 | 21 | 194 |
| Scala | 2 | 14 | 135 |

Результаты поучились приблизительно равными, поскольку основной промежуток времени занимает открытие и чтение файлов, которое выполняется одинаково, как на Java.

