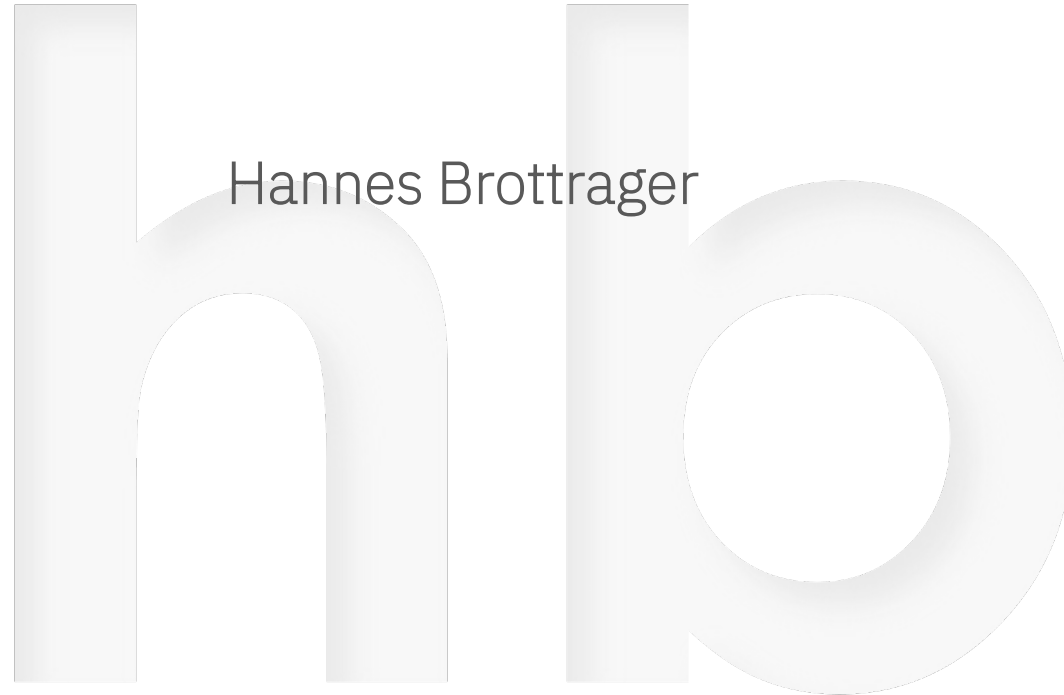


# CardGameApp

Gruppe International

Hannes Brottrager



# Themen

## JAVA Language

- Scanner / Environment
- Logging
- Exceptions
- File IO

## Framework

- Javadoc
- Apache Maven
- .PO-Files

## Fazit | Ausblick



# Java - Scanner

```
import java.util.Scanner;
```

- Input von der Console

```
... public static Scanner input = new Scanner(System.in);

... String[] validLang = {"German", "English", "X"};
... String actLang = standardLanguageEN; // Defaultvalue
... boolean selectedLang = false;

... while (!selectedLang) {
...     System.out.println("\n-----");
...     System.out.println("Choose a language: (German|English [X=Exit])");
...     System.out.println("-----\n");

...     String inputLang = input.nextLine();

...     for (String s : validLang)
...     {
...         /*
...         * Uses {@code equals()} and not {@code ==}
...         */
...         if (s.equals(inputLang)) {
...             selectedLang = true;
...             /*
...             * {@value #actLang}
...             */
...             actLang = inputLang;
...             System.out.println("selected language: " + actLang);
...             log.info("Selected language (" + actLang + ")");
...             break;
...         }
...     }
... }
```

# Java - Scanner

```
import java.util.Scanner;
```

- Input von der Console

```
...String[] validGameType = {"1", "2", "3", "X"};
...String actGameType = "1";
...boolean selectedgameType = false;
...
...// 2. GameTyp dialog
...while (!selectedgameType) {
...    System.out.println("\n" + questionGameTypeTxt + "(1=BlackJack, 2=Uno, 3=Schnapsen, X=Test-Exception)\n");
...    String inputGameType = new Scanner(System.in).nextLine();
...    // int inputGameType = new Scanner(System.in).nextInt();

...    for (String t : validGameType)
...        if (t.equals(inputGameType)) {
...            selectedgameType = true;

...            try {
...                gameType = Integer.parseInt(inputGameType);
...            } catch (NumberFormatException e) {
...                System.err.println(x: "'%s' kann man nicht in Zahl konvertieren '%n'");
...            }
...            System.out.println("inputGameType=" + gameType);
...            break;
...        }
...    }
...}
```

# Java - Environment

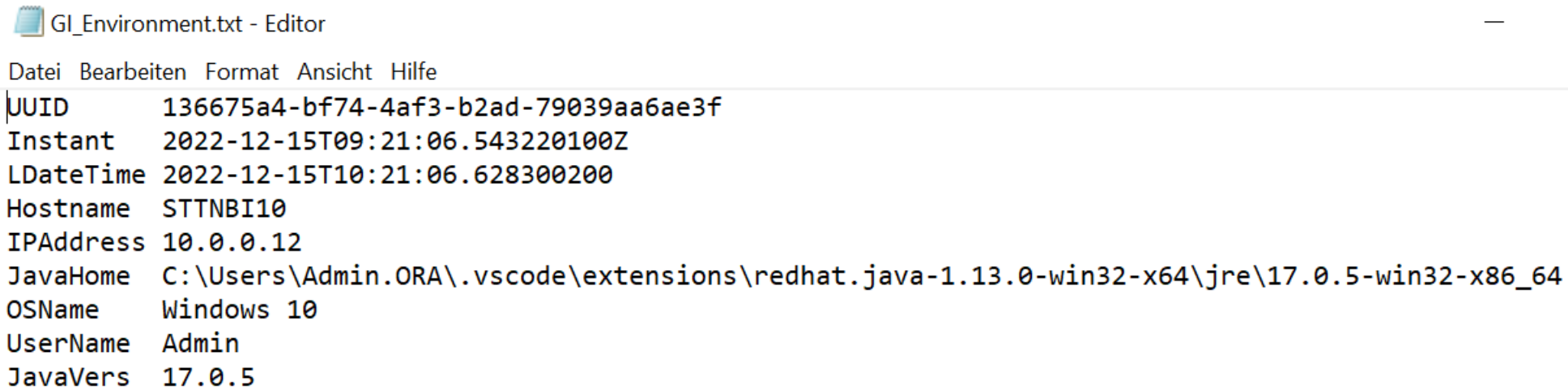
- Environment

```
import java.net.InetAddress;  
import java.time.Instant;  
import java.time.LocalDateTime;  
import static java.time.Instant.now;  
import java.util.UUID;
```

```
..// InetAddress -- Exception definieren!!!  
..try {  
    ...String Computer_Name = InetAddress.getLocalHost().getHostName();  
    ...String IP_Address = InetAddress.getLocalHost().getHostAddress();  
    ...System.out.println("Hostname: " + Computer_Name);  
    ...System.out.println("IPAdresse: " + IP_Address);  
..} catch (Exception e) {  
    ...System.out.println(x: "Error: UnknownHostException!");  
..}  
  
..Instant start = now();  
..LocalDateTime localDateTime = LocalDateTime.now();  
  
..String content = UUID.randomUUID() + "\n" + start + "\n" + localDateTime + "\n";  
  
..System.getenv().forEach((k, v) -> {  
    ...System.out.println(k + ":" + v);  
..});
```

# Java - Environment

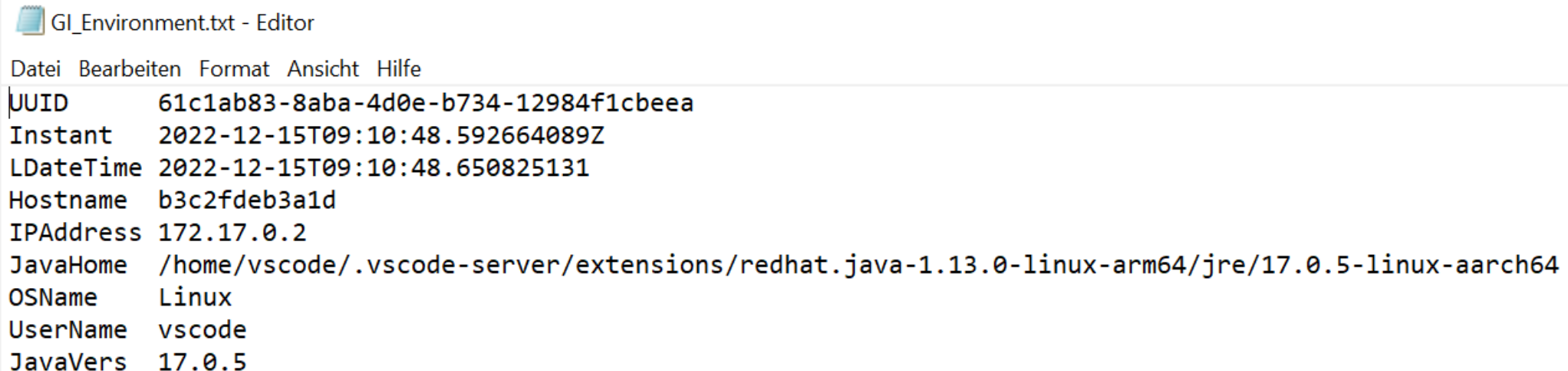
- Environment



Gl\_Environment.txt - Editor

Datei Bearbeiten Format Ansicht Hilfe

UUID 136675a4-bf74-4af3-b2ad-79039aa6ae3f  
Instant 2022-12-15T09:21:06.543220100Z  
LDateTime 2022-12-15T10:21:06.628300200  
Hostname STTNBI10  
IPAddress 10.0.0.12  
JavaHome C:\Users\Admin.ORA\.vscode\extensions\redhat.java-1.13.0-win32-x64\jre\17.0.5-win32-x86\_64  
OSName Windows 10  
UserName Admin  
JavaVers 17.0.5



Gl\_Environment.txt - Editor


Datei Bearbeiten Format Ansicht Hilfe

UUID 61c1ab83-8aba-4d0e-b734-12984f1cbeea  
Instant 2022-12-15T09:10:48.592664089Z  
LDateTime 2022-12-15T09:10:48.650825131  
Hostname b3c2fdeb3a1d  
IPAddress 172.17.0.2  
JavaHome /home/vscode/.vscode-server/extensions/redhat.java-1.13.0-linux-arm64/jre/17.0.5-linux-aarch64  
OSName Linux  
UserName vscode  
JavaVers 17.0.5

# JAVA - Logging

```
import java.util.logging.Logger;  
import java.util.logging.FileHandler;  
import java.util.logging.Level;
```

- Logging mit JUL

 GI\_Logging.xml

 GI\_Logging.xml

```
1  <?xml version="1.0" encoding="windows-1252" standalone="no"?>  
2  <!DOCTYPE log SYSTEM "logger.dtd">  
3  <log>  
4  <record>  
5    <date>2022-12-14T08:50:43.321095100Z</date>  
6    <millis>1671007843321</millis>  
7    <nanos>95100</nanos>  
8    <sequence>0</sequence>  
9    <logger>msc.ddb.international.App</logger>  
10   <level>INFO</level>  
11   <class>msc.ddb.international.App</class>  
12   <method>startLogging</method>  
13   <thread>1</thread>  
14   <message>Start Logging with Java Util Logging (GI_Logging.xml)</message>  
15 </record>
```

# JAVA - Logging

```
import java.util.logging.Logger;
import java.util.logging.FileHandler;
import java.util.logging.Level;
```

- Logging mit JUL

```
..private static final Logger log = Logger.getLogger(App.class.getName());
..public static FileHandler handlerLog;
..
..startLogging();

..try {
..    handlerLog = new FileHandler(FILENAME_LOGGING);
..    log.addHandler(handlerLog);
..} catch (IOException | SecurityException e) {
..    log.log(Level.WARNING, "Error while creating the logfile (" + FILENAME_LOGGING + ")", e);
..}
..log.info("Start Logging with Java Util Logging (" + FILENAME_LOGGING + ")");

..handlerLog.close(); .....
```



# JAVA - Logging

```
import java.util.logging.Logger;  
import java.util.logging.FileHandler;  
import java.util.logging.Level;
```

- Logging mit JUL

Level
SEVERE
WARNING
INFO
CONFIG
FINE
FINER
FINEST

```
log.info ("Write Environments: " + User_Name);
```

```
log.info ("selected language: " + actLang);
```

```
log.warning ("Warning - Not enough players in the game");
```

```
log.severe ("Error! - Error writing file (" + FILENAME_ENVIRONMENT + "), e);
```

```
log.info ( () -> String.format( "Runtime %s ms", start.until( now(), MILLIS))));
```

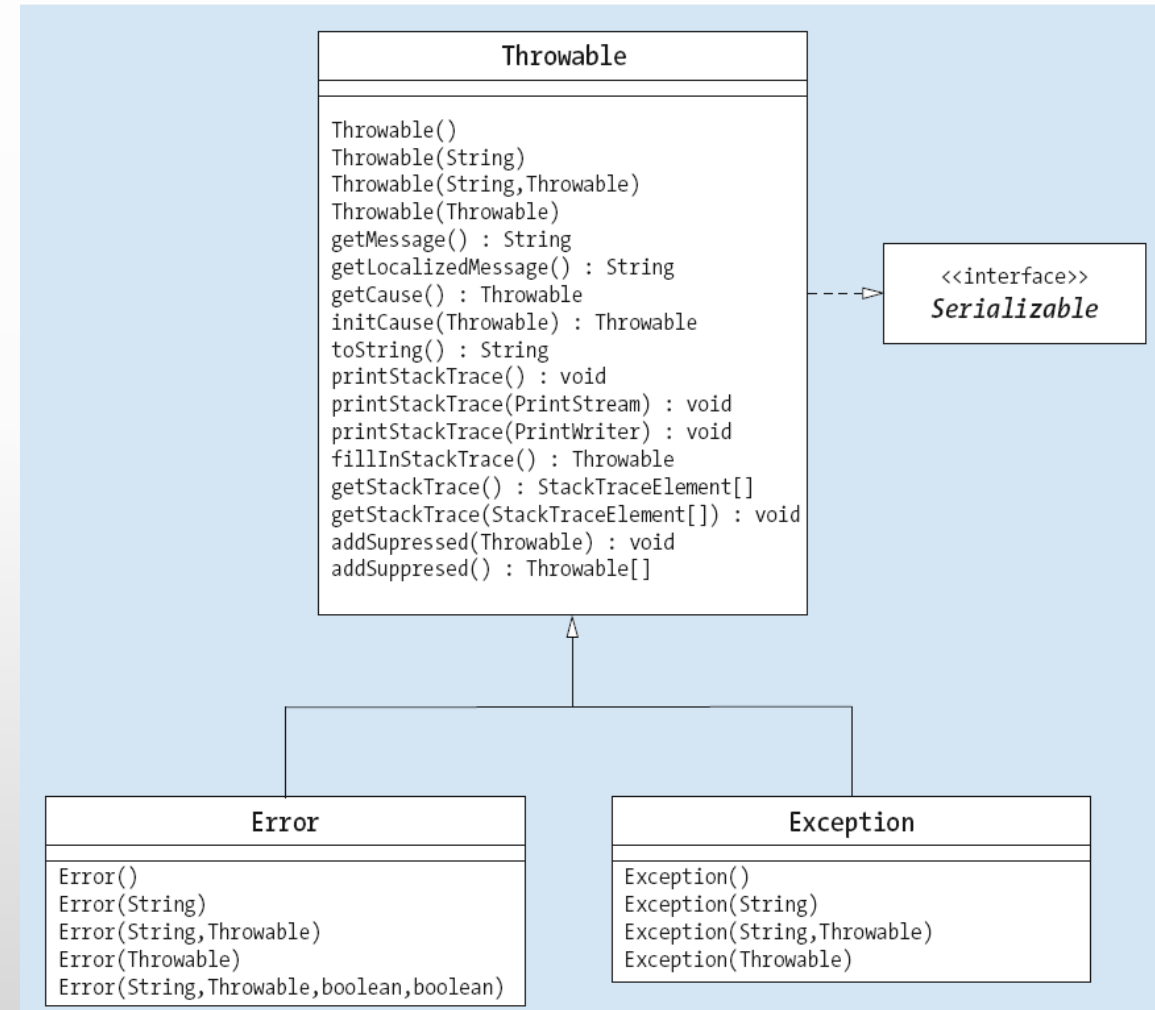
# Java - Exceptions

- **Zweck**

- Java bietet mit Exceptions eine Methode, um Ausnahmen (unerwartete) Fehler abzufangen, sodass Programme in fast jeder Situation weiterlaufen können!
- der Programmcode wird durch permanente Abfrage der Rückgabewerte nicht unterbrochen!

- **Umsetzung**

- **try**  
Code, der zu einer Ausnahme führen kann
- **catch**  
Code, zur Behandlung der Ausnahme
- **finally**  
Code, der immer ausgeführt wird

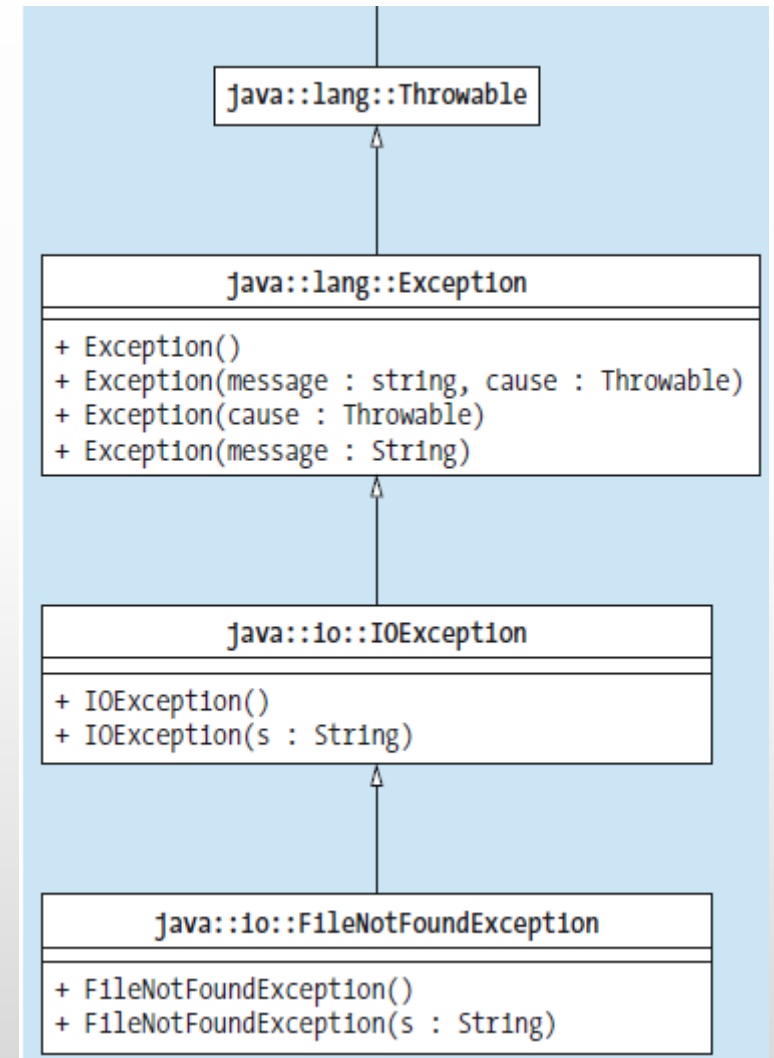


# Java - Exceptions

- Typen
  - behebbare
  - nicht behebbare
- Arten
  - geprüfte
  - ungeprüfte (RuntimeException)
- Stack Trace
  - + Methodenstapel
  - + Fehleranalyse
- Weiterleitung
- Reihenfolge

richtig

```
try {  
    ...  
}  
catch ( FileNotFoundException e ) {  
}  
catch ( IOException e ) {  
}
```



# Java - Exceptions

```
import java.io.IOException;
```

- Filehandling

```
...try{
...|...Files.writeString(Path.of(FILENAME_ENVIRONMENT), content, StandardOpenOption.CREATE_NEW);
...}|
...catch(IOException e){
...|...System.err.println("%s File cannot be saved '%n'");
...|...e.printStackTrace();
...|...log.severe("Error! - Error deleting file - Logging with 'JUL' (Java Util Logging)");
...|}
...finally{
...|...System.out.println("File write New (" + FILENAME_ENVIRONMENT + ") block finally");
...|...log.info("Info - File write new (" + FILENAME_ENVIRONMENT + ") block finally");
...|}
...}
```

# Java - Exceptions

```
import java.io.IOException;
```

- Filehandling

```
... Path path = Paths.get(FILENAME_LOGGING);

... try {
...     Files.deleteIfExists(path);
... } catch (IOException e) {
...     e.printStackTrace();
...     log.severe("Error! -- Error deleting file (" + FILENAME_LOGGING + ")");
... }

... try {
...     handlerLog = new FileHandler(FILENAME_LOGGING);
...     log.addHandler(handlerLog);
... } catch (IOException | SecurityException e) {
...     log.log(Level.WARNING, "Error while creating the logfile (" + FILENAME_LOGGING + ")", e);
... }

... log.info("Start Logging with Java Util Logging (" + FILENAME_LOGGING + ")");
```

java.util.logging.FileHandler.FileHandler(String arg0) throws IOException, SecurityException

# Java - Exceptions

```
import java.net.UnknownHostException;
```

- Weiterleitung Exception

```
..try {  
    InetAddress java.net.InetAddress.getLocalHost() throws UnknownHostException  
    ....String Computer_Name = InetAddress.getLocalHost().getHostName();  
    ....String IP_Address = InetAddress.getLocalHost().getHostAddress();  
    ....writeEnvironmentFile("Hostname" + Computer_Name + "\n");  
    ....writeEnvironmentFile("IPAddress" + IP_Address + "\n");  
..}  
..catch (UnknownHostException e) {  
    ....System.out.println("Error: UnknownHostException!");  
    ....log.severe("Error! -- Error unknown host (" + FILENAME_ENVIRONMENT + ")");  
..}
```

# Java - Exceptions

```
import msc.ddb.international.exceptions.NotEnoughPlayersException;  
import msc.ddb.international.exceptions.TooManyPlayersException;
```

- Benutzerdefinierte Exception

```
...try {  
    ...BlackJack game = new BlackJack();  
    ...game.addPlayer(new Player(name: "Harald"));  
    ...game.initializeGame();  
    ...game.startGame();  
} catch (TooManyPlayersException e) {  
    ...e.printStackTrace();  
    ...log.warning("Warning - To many players in the game!");  
} catch (NotEnoughPlayersException e) {
```

```
...@Override  
...public void initializeGame() throws NotEnoughPlayersException {  
    ...if (getDealer() != null && getPlayers().size() >= getMinimumPlayers()) {  
        ...setDeck(new Deck(multiplier: 6));  
        ...initialDeal();  
    }  
    ...else  
        ...throw new NotEnoughPlayersException("You need at least " + getMinimumPlayers() + " to play.");  
}
```

# Java - File I/O

- JAVA Files

- + Environment

- + Logging

```
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.nio.file.StandardOpenOption;
import java.io.IOException;
```

```
·private static final String FILENAME_ENVIRONMENT = "GI_Environment.txt";
·private static final String FILENAME_LOGGING = "GI_Logging.xml";
·private static final String FILENAME_GAME = "GI_Game.txt";
```

```
...
Path path = Paths.get(FILENAME_ENVIRONMENT);

try {
    boolean java.nio.file.Files.deleteIfExists(Path arg0)
    Files.deleteIfExists(path);
} catch (IOException e) {
    e.printStackTrace();
    log.severe("Error! - Error deleting file (" + FILENAME_ENVIRONMENT + ")");
}
```



# Java - File I/O

- JAVA Files

- + Environment
- + Logging

```
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.nio.file.StandardOpenOption;
import java.io.IOException;
```



```
· writeEnvironmentFile("Hostname" + Computer_Name + "\n");
· writeEnvironmentFile("IPAddress" + IP_Address + "\n");
```

```
· public static void writeEnvironmentFile(String contentParam) {
·     .....
·     try { Path java.nio.file.Files.writeString(Path arg0, CharSequence arg1, OpenOption... arg2)
·         Files.writeString(Path.of(FILENAME_ENVIRONMENT), contentParam, StandardOpenOption.APPEND);
·     }
·     catch (IOException e) {
·         System.err.println("'" + FILENAME_ENVIRONMENT + "' cannot be saved '%n'");
·         e.printStackTrace();
·         log.severe("Error! - Error writing File (" + FILENAME_ENVIRONMENT + ")");
·     }
· }
```

# Java - File I/O

- .JSON

- JavaScript Object Notation
- Data
- Schema

 card.json card-schema-v1.json

```
{ } card.json > ...
1 {
2   "$schema": ".card-schema-v1.json",
3   "cardGame": "UNO",
4   "nameGame": "UnoSpiel",
5   "cards": [
6     {
7       "id": 1,
8       "name": "ziffer1",
9       "color": "green",
10      "symbol": "number"
11    },
12    {
13      "id": 2,
14      "name": "ziffer2",
15      "color": "green",
16      "symbol": "number"
17    },
18    {
19      "id": 3,
20      "name": "ziffer3",
21      "color": "green",
22      "symbol": "number"
23    },
24    {
25      "id": 4,
26      "name": "ziffer4",
27      "color": "yellow",
28      "symbol": "number"
29    },
30    {
31      "id": 50,
32      "name": "Draw4",
33      "color": "",
34      "symbol": "action"
35    },
36    {
37      "id": 50,
38      "name": "Wild",
39      "color": "",
40      "symbol": "action"
41    }
42  ],
43   "self": "https://www.cardgameapp.at"
44 }
```

```
{ } card-schema-v1.json > ...
1 {
2   "$schema": "http://json-schema.org/draft-07/schema",
3   "type": "object",
4   "required": ["cardGame"],
5   "properties": {
6     "cardGame": {
7       "type": "string",
8       "description": "Name of the Card Game",
9       "pattern": "[A-Z]"
10    },
11    "nameGame": {
12      "type": "string"
13    },
14    "cards": {
15      "type": "object",
16      "required": ["id", "name"],
17      "properties": {
18        "id": {
19          "type": "integer",
20          "minimum": 1,
21          "maximum": 130,
22          "description": "eindeutige lfd. Ganzzahl je Spiel"
23        },
24        "name": {
25          "type": "string",
26          "pattern": "[a-z][0-9]"
27        },
28        "color": {
29          "type": "string",
30          "enum": ["black", "red", "green", "blue", "yellow", "orange", "gray", "pink", "white"]
31        },
32        "symbol": {
33          "type": "string",
34          "enum": ["number", "action"]
35        }
36      }
37    },
38    "self": {
39      "type": "string",
40      "format": "uri"
41    }
42  },
43   "additionalProperties": false
44 }
```

# Framework - Javadoc

- **Keywords**

- @author
- @version
- @param
- @return
- @depreceated
- @see
- @code
- @value

- ✓ Block Tags
- ✓ Inline Tags

- **javadoc**

- + Output html
- + Doclets
- + DocLint (Xdoclint)

```
javadoc -sourcepath src\main\java;src\test\java -private -d .\javadoc  
-subpackages . -windowtitle "CardGameApp" -author
```

Name	Änderungsdatum
msc	12.12.2022 17:10
allclasses-frame.html	12.12.2022 21:06
allclasses-noframe.html	12.12.2022 21:06
constant-values.html	12.12.2022 21:06
deprecated-list.html	12.12.2022 21:06
help-doc.html	12.12.2022 21:06
index.html	12.12.2022 21:06
index-all.html	12.12.2022 21:06
overview-frame.html	12.12.2022 21:06
overview-summary.html	12.12.2022 21:06
overview-tree.html	12.12.2022 21:06
package-list	12.12.2022 21:06
script.js	12.12.2022 21:06
stylesheet.css	12.12.2022 17:10

# Framework - Javadoc

```
msc.ddb.international.BlackJack
```

## Game Class

*Specialization of Class Game*

- **Author:**
  - Markus Hilbert

```
System.out.println(BlackJack.
```

+ VS-Code Unterstützung

CardGameApp | D2 Group Internatic X

file:///C:/Users/Admin.ORA/Documents/GitHub/cardgameappSSH/javadoc/index.html

Suchen

OVERVIEW PACKAGE **CLASS** TREE DEPRECATED INDEX HELP

PREV CLASS **NEXT CLASS** FRAMES NO FRAMES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

msc.ddb.international

### Class App

java.lang.Object  
msc.ddb.international.App

---

```
public class App  
extends java.lang.Object
```

### CardGameApp

written by the specialist Markus Hilbert and his 'international' boygroup (Felix, Eric, Hannes)

**Author:**  
Markus Hilbert

### Field Summary

Modifier and Type	Field and Description
private static java.lang.String	actLang

# Framework - Apache Maven

## Build-Management

- + Defacto-Standard (Apache)
- + Integriert in IDE's
- + Separate Verzeichnisstruktur
- + frei und quelloffen
- + transitierende Auflösung
- + Lebenszyklus

## Pom.xml

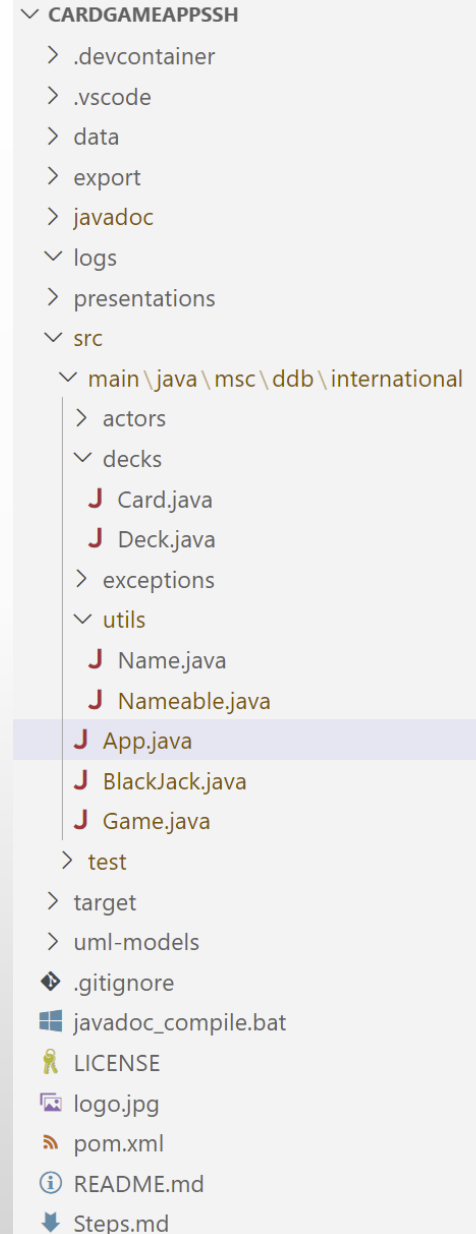
Projektbeschreibung

## Repository

\$HOME\.m2\repository

## Archetypes

<https://repo.maven.apache.org>



```
▼ CARDGAMEAPSSH
  > .devcontainer
  > .vscode
  > data
  > export
  > javadoc
  ▼ logs
  > presentations
  ▼ src
    ▼ main\java\msc\ddb\international
      > actors
      ▼ decks
        J Card.java
        J Deck.java
      > exceptions
      ▼ utils
        J Name.java
        J Nameable.java
        J App.java
        J BlackJack.java
        J Game.java
      > test
    > target
    > uml-models
    .gitignore
    javadoc_compile.bat
    LICENSE
    logo.jpg
    pom.xml
    README.md
    Steps.md
```

# Framework - .PO Files

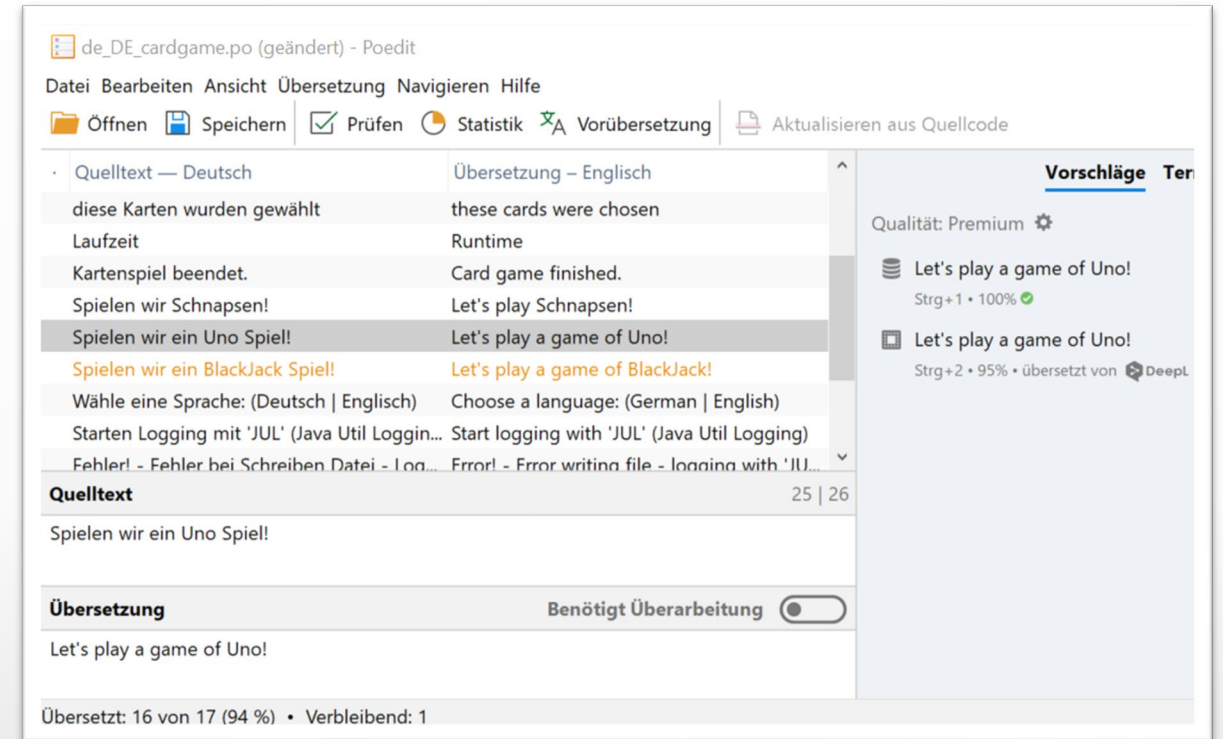
- .PO

- Portable Object Files
- Translation
- Native Language Support

+ GNU – gettext()-Method

+ .POT Templates

+ WordPress, CSS Vorlagen



```
msgid "Spielen wir ein Uno Spiel!"
msgstr "Let's play a game of Uno!"
```

```
ResourceBundle rb = ResourceBundle.getBundle("de_DE_cardgame.po");
System.out.println(GettextResource.gettext(rb, "Spielen wir ein Uno Spiel"));
```

# Fazit

- + Objektorientierter Ansatz
- + Zugriffsmodifizier
- + Wiederverwendbarer Code
- + #Packages und Libraries









# Danke!