



UNIVERSITÄT ULM

FREIWILLIGES EIDI PROJEKT

Placeholder

Tut Mi 16 2202

supervised by
Raphael STÖRK

29. November 2019

Zusammenfassung

In diesem Dokument sind alle wichtigen Prozessschritte, Entscheidungen und Ergebnisse dokumentiert, welche im Verlauf des Projektes 'Placeholder' entstanden sind. Das Projekt selber ist eine eigenständige, freiwillige Zusatzleistung der Tutoriumsteilnehmer, welches als Unterstützung der Studenten beim Erlernen wichtiger Grundlagen helfen soll. Das Projekt dient insbesondere NICHT als Pflichtteil der Übung sondern ist ein individuelles Zusatzangebot des Tutors.

[Ergebnisse]

Inhaltsverzeichnis

1	Einführung	3
2	Grundlagen	4
2.1	Vorgaben	4
2.2	Projekt-Parameter	5
2.2.1	Technische Vorgaben	5
2.2.2	Genre	5
2.2.3	Konzept	5
2.2.4	Story	5
3	Anforderungen	6
3.1	Funktionale Anforderungen	6
3.2	Qualitäts Anforderungen	6
4	Entwicklung	7
4.1	Woche 1 - Grundlagen	7
4.2	Woche 2 -	9
4.3	Woche 3 -	10
4.4	Woche 4 -	10
5	Implementierung	11
5.1	Framework	11
5.2	Architektur	11
5.3	Vorstellung der Software	11
6	Einordnung	12
6.1	Verwandte Arbeiten	12
6.2	Evaluation	12
6.3	Ausblick	12

1 Einführung

In Verbindung mit den aktuellen Tutorien in Einführung in die Informatik an der Universität Ulm möchten die Teilnehmer des Tutoriums von Raphael Störk, Mittwoch 16-18, neben den Pflichtübungen einen zusätzlichen Arbeitsaufwand leisten um eine breitere Grundlage und ein größeres Basiswissen für ihr weiteres Studium zu schaffen. Dafür wurde von dem Tutor ein gemeinsames Projekt vorgeschlagen, welches in den Tutorien als Zusatzübung erarbeitet werden soll. Ziel dieses Projektes ist nicht in erster Linie die Produktion einer vollständig entwickelten und getesteten Software sondern die Erlernung Grundlegender Programmierkenntnisse anhand praktischer Beispiele, welche im Verlauf des Projekts erarbeitet werden.

Der grundlegende Gedanke hinter diesem Projekt ist die Anwendung der im Tutorium kennen gelernten und vorgestellten Themen auf ein großes, kontinuierlich erweitertes Projekt um das Verständnis der Studenten in der Hinsicht zu fordern, dass klar wird wofür diese Themen im Bereich der Informatik und Programmierung wichtig sind und verwendet werden.

Klar muss auch sein, dass dieses Projekt eine rein freiwillige Übung für die Studenten ist. Die Teilnahme und Einbringung an dieser Übung wird nicht in die Bepunktung der Übungsvorleistung mit eingerechnet und ist auch niemals als eine Voraussetzung für das Bestehen einer Universitären Leistung gedacht. Auch ist dem Tutor keine weitere Tutoriumsgruppe bekannt, die solch ein Projekt durchzuführen gedenkt. Daher sollte eindeutig gesagt werden, dass dieses Projekt als reine Zusatzübung entwickelt wird. Entsprechend sollte die Richtigkeit und Vollständigkeit mit Bedacht genossen werden.

2 Grundlagen

2.1 Vorgaben

Folgende Vorgaben wurden zu Beginn des Semesters an das Tutorium angebracht:

Motivation Das Projekt wird nur verfolgt, wenn die Teilnehmer des Tutoriums dies auch möchten. Hierfür wurde eine entsprechende Umfrage erstellt bei der heraus kam, dass die Mehrzahl der Teilnehmer dieses Projekt angehen möchten.

Intention Das Projekt soll gemeinsam in der Zeit des Tutoriums, soweit es die pflichtigen Aufgaben zulassen, entwickelt werden. Die Intention hinter diesem Projekt ist die zusätzliche Bereitstellung von theoretischem und praktischem Wissen im Bereich der Grundlagen der Informatik und Programmierung.

Themenfreiheit Die Teilnehmer haben demokratische Stimmgewalt über die Themen und Funktionen des Projektes. Der Tutor behält sich jedoch ein allgemeines Veto-Recht vor, welches in Stichsituationen zum Einsatz kommen kann.

Erreichbarkeit Alle im Tutorium entwickelten Ressourcen werden den Teilnehmern online jederzeit zugänglich gemacht. Damit soll erreicht werden, dass die Teilnehmer jedweden Thematischen Vorwissen die Möglichkeit haben alle erarbeiteten Projektteile zu verstehen und zu diskutieren. Zudem dienen diese Ressourcen als Hilfe für weitere Aufgaben und als zusätzliche Vorbereitung auf die Klausur. Hierbei ist zu beachten, dass keine Musterlösungen oder Plagiate den Weg in dieses Projekt finden dürfen.

2.2 Projekt-Parameter

In diesem Abschnitt wird im Verlauf der Projektentwicklung aufgeführt, welche Parameter durch die Teilnehmer erdacht und gewählt wurden. Der bisherige Stand findet sich in folgender Auflistung:

2.2.1 Technische Vorgaben

Sprache min. Java 8 / Entwickelt mit Java 13

IDE Zu Beginn keine, eventuell wird im Verlauf des Semesters auf Eclipse/IntelliJ umgestiegen

Versionierung Git, mit Hilfe von GitHub und Gitkraken, für Teilnehmer nur optional

2.2.2 Genre

Noch offen

2.2.3 Konzept

Noch offen

2.2.4 Story

Folgende Storyelemente wurden bisher erarbeitet:

Zeit Industrialisierung, 18./19. Jahrhundert

Ort London, England

Lebensraum Industriegebiet/Armenviertel

Keywords Rebellengruppe, Kommunismus, Steampunk, Kriegstreibende Regierung, Drogenkriminalität

Main Character Jugendliche/Teenager, Arbeiterfamilie, Indisch/Irische Herkunft, Teil der Rebellengruppe

3 Anforderungen

3.1 Funktionale Anforderungen

3.2 Qualitäts Anforderungen

4 Entwicklung

Dieses Kapitel enthält eine kurze Übersicht über die in den einzelnen Übungswochen erarbeiteten Themen und den damit verbundenen Projektteilen.

4.1 Woche 1 - Grundlagen

KEYWORDS

Algorithmen, Zahlenbasen, Horner-Schema, Java Mini-Workflow

In Woche 1 (hier auch Woche 0 miteinbezogen, diese hat kein eigenes Kapitel verdient) wurde als Grundlage für die weiteren Wochen gezeigt, wie eine Kommandozeile zu verwenden ist und wie man mit dieser ein einfaches Java-Programm kompiliert (*übersetzt*) und ausführt. Wichtig dabei sind die Befehle **javac** zum compilieren und **java** zum Ausführen. Ein Programm in einer Datei namens *Hello.java* würde man also mit folgenden Befehlen ausführen:

```
.../> javac Hello.java
```

```
.../> java Hello
Hello World!
```

Außerdem wurde in dieser Woche über Zahlensysteme gesprochen. Es ist bei der Programmierung oft hilfreich und manchmal notwendig das Konzept binärer und anderer Zahlen zu kennen. Eine Zahl im allgemeinen ist eine Aneinanderreihung von Ziffern, die Stelle der Ziffer in dieser Reihung bestimmt deren Gewichtigkeit.

So ist im Dezimalsystem die Zahl 123_{10} zu verstehen als $1 \cdot 10^2 + 2 \cdot 10^1 + 3 \cdot 10^0$. Entsprechend wäre die Zahl 1010_2 im Binärsystem zu verstehen als $1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 8 + 2 = 10_{10}$. Die Umrechnung zwischen verschiedenen Systemen kann mittels des **Horner-Schemas** erreicht werden (vgl. Vorlesung).

Vor allem aber wurde in dieser Woche bereits die Entwicklung von Algorithmen besprochen. Dies geschah über die Schritte **Problemspezifikation**, **Problemabstraktion**, **Algorithmenentwurf**, **Verifikation** und **Aufwandsanalyse**. Mit diesen Schritten wurde für das Projekt versuchsweise ein Algorithmus erarbeitet, der einer Punkte-Berechnung am Ende eines Spiels darstellen könnte. Es entstand folgender Pseudocode:

Algorithm "Check Won":

```
input s, p, c;
c = c + round_down(p / 10);
while c > 1:
    if c % 2 == 0
        c = c / 2;
        s = s + 1;
    else
        c = 0
    end if
end loop

if s > 10
    print 'spieler hat gewonnen'
else
    print 'spieler hat verloren'
end if
```

Es wurde bestimmt, dass dieser Algorithmus einen Aufwand der Form $\log(n)$ aufweist.

Zuletzt wurde auch eine **Java-Klasse** als Start für unsere Anwendung aufgebaut. Diese enthielt nur eine simple main-Methode und eine Begrüßung des Nutzers:

"Application.java":

```
import java.util.Scanner;

public class Application {

    public static void main(String[] args) {

        // create a new Scanner that lets us read in input from the user
        Scanner scan = new Scanner(System.in);
        System.out.print("Welcome to <placeholder>, what is your name?");
        String name = scan.nextLine();
        System.out.print("Hello " + name + "!");
        scan.close();
    }
}
```

4.2 Woche 2 -

KEYWORDS

Pseudocode, Datentypen, Boolesche Ausdrücke, Menüführung

In der zweiten Übungswoche wurde das Konzept der Algorithmenentwicklung mit Pseudocode wiederholt. Zudem wurden Grundlegende Datentypen in Java kennen gelernt. Die primitiven Datentypen sind dabei: **byte, short, int, long, char, float, double und boolean**. Als nicht-primitiver Datentyp wurden **Strings** angesprochen. Es wurde erklärt wie diese Datentypen den zugehörigen Speicher belegen und in welcher Situation welcher Datentyp sinnvoll wäre.

Auch wurde bereits das Konzept Logischer Ausdrücke und die Verwendung von Wahrheitswerten besprochen. Dazu wurde die Funktion von Negation **!**, logischem und **&&**, logischem oder **||** und dem Vergleich **==** erklärt und angewandt.

Mit diesen nun zugänglichen Grundlagen wurde für das Projekt dann gemeinsam ein simples Anwendungsmenü aufgebaut. Es wurde ermittelt, dass ein Nutzer über die Eingabe verschiedener Zahlen entsprechende Punkte des Menüs erreichen kann. Über sinnvolle Abfragen im Programm soll die Eingabe des Nutzers mit den vorgegebenen Punkten abgeglichen werden. Dabei entstand folgendes Menü:

"Application.java" - Main Method

[...]

```
public static void main(String[] args)
    // create a new Scanner that lets us read in input from the user
    Scanner scan = new Scanner(System.in);
    System.out.print("Welcome to <placeholder>, what is your name?");
    String name = scan.nextLine();
    System.out.print("Hello " + name + "!");

    // Let user choose an option
    System.out.print("What do you want to do?");
    System.out.print("\t(1) Play Game");
    System.out.print("\t(2) Options");
    System.out.print("\t(3) Loading");
    System.out.print("\n::MENU::> ");
    int opt = scan.nextInt();
```



```
// The menu options
System.out.println("You have entered option: " + opt);

if(opt == 1) {
    System.out.print("This is game");
} else if(opt == 2) {
    System.out.print("This is options");
} else if(opt == 3) {
    System.out.print("This is loading");
} else {
    System.out.print("This is not a valid input");
}
scan.close();
}
[...]
```

4.3 Woche 3 -

4.4 Woche 4 -

5 Implementierung

5.1 Framework

5.2 Architektur

5.3 Vorstellung der Software

6 Einordnung

6.1 Verwandte Arbeiten

6.2 Evaluation

6.3 Ausblick