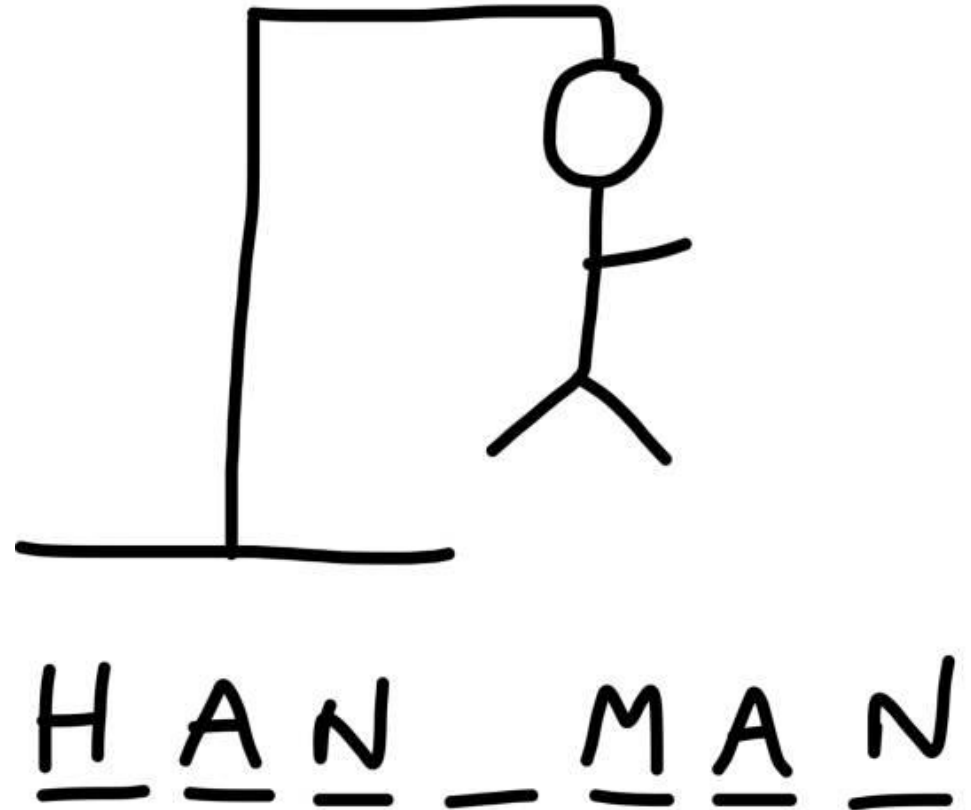


# PRÄSENTATION PROGRAMMIEREN GALGENMÄNNCHEN

Celine Gugel  
Lena Steinbrink  
Valentin Müller  
Jannik Oßwald  
Thi Tuong Vy Nguyen

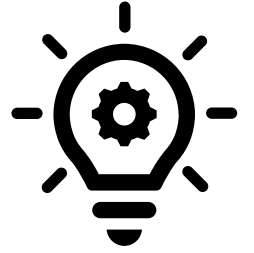
ON22



# INHALT

1. Ideenfindung und Spielerklärung
2. Umsetzungsplanung
3. Benötigte Funktionen
4. Spiel Aufbau

# 1. IDEENFINDUNG UND SPIELERKLÄRUNG



- Jeder hat sich selbst informiert, was möglich ist
- Wissen der Vorlesung umsetzen
- Bestimmter Begriff muss erraten werden
- Vorgeben sind nur Anzahl Buchstaben
- **Richtig** = Buchstaben erscheint
- **Falsch** = Abbildung Galgen bildet sich
- Ist Galgen vollständig – Spiel verloren
- **2 Spielvarianten – 1. Gegen Computer 2. Wort selbst angeben**

# 2. UMSETZUNGSPLANUNG

## 2.1 Auswahl Spielmodus

### **Spielmodus 1**

- Spiel gegen den Computer
- aus den vorgegebenen Wörtern soll ein Wort zufällig ausgegeben werden und das Spiel startet

### **Spielmodus 2**

- Spiel mit eigener Worteingabe

### **(Modus 3)**

- Spiel verlassen
- Leerzeichen einfügen, damit Spielauswahl nicht mehr zu sehen ist

# 2. UMSETZUNGSPLANUNG

## 2.2 Worteingabe und Buchstaben erraten

- Wort ist mindestens 2 Zeichen lang
- Wort und Buchstabe enthalten keine Leerzeichen oder Sonderzeichen
- Buchstabe kann nur einmal eingegeben werden
- Bei der Eingabe soll nur der erste Buchstabe genommen werden
- Buchstaben müssen an der entsprechenden Stelle vom Wort für die Ausgabe in der Konsole stehen

# 2. UMSETZUNGSPLANUNG

## 2.3 Ausgabe der Konsole

- "Das Spiel Hangman wurde gestartet!" wenn das Spiel gestartet wurde
- "Spielmodus 1: Spiele gegen den Computer"
- "Spielmodus 2: Gib ein eigenes Wort ein (Multiplayer)"
- „Es wurde Spielmodus 1/2 gewählt.“ um dem Spieler mitzuteilen, in welchem Spielmodus er sich befindet
- „Ein Buchstabe bitte:“ als Aufforderung für den Spieler ein Wort einzugeben
- „Hurra! Deine Eingabe war richtig!“, wenn Eingabe richtig ist
- „Falsche Eingabe“, wenn Eingabe falsch ist
- „Wenn du das Spiel beenden möchtest, kannst du exit eingeben.“ damit der Spieler weiß, wie er das Spiel beenden kann
- „Das Spiel wird beendet“, wenn der Spieler das Spiel beendet

### 3. BENÖTIGTE FUNKTIONEN

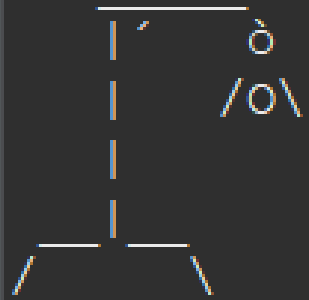
Funktionen	Variablen/Parameter
„validateInput“ überprüft, ob die Eingabe im definierten Wertebereich liegt	<i>char input</i> <i>String validLetters</i>
„prüfeEingabe“, überprüft, ob die Eingabe des zu erratenen Wortes mindestens 2 Buchstaben hat und alle gültigen Buchstaben beinhaltet	<i>boolean wert</i>
„verglChar“, vergleicht den eingegebenen Buchstaben mit den schon versuchten Buchstaben und dem gesuchten Wort	<i>char eingabe</i> <i>String gesWort</i>
„getstatus“ ergibt den Zwischenstand des Spiels, gibt am Ende aus, ob der Spieler das Spiel gewonnen oder verloren hat	<i>int fehlerAnzahl</i> <i>boolean gewonnen</i> <i>boolean verloren</i>
„zeichneSpiel“, um die Zeichnung dann auch ausgeben zu können	

### 3. BENÖTIGTE FUNKTIONEN

```
1 import java.util.ArrayList;
2
3 public class Spiel {
4
5     private String gesWort;
6     private int fehlerAnzahl = 0;
7     private char[] wortArr;
8     private ArrayList<Character> schonVersucht = new ArrayList<Character>();
9     private boolean gewonnen = false;
10    private boolean verloren = false;
11    private static final String validLetters = "abcdefghijklmnopqrstuvwxyzäöüß";
12
13    Spiel(String gesWort){
14
15
16    public void verglChar(char eingabe) { /*vergleicht den eingegebenen Buchstaben mit dem gesuchten Wort und den schon
17    versuchten Buchstaben */
18
19
20
21
22    private boolean schonVersucht(char test) { /*ueberprueft ob der Buschstabe zuvor schonmal eingegeben wurde*/
23
24
25
26    private static boolean contains(char in, String pruefWert) { /*ueberprueft ob das gesuchte Wort den Buschstabe
27    enthält*/
28
29
30
31
32    private static boolean validateInput(char input) { /*ueberprueft die ob die Eingabe im Deffinierten Wertebereich
33    liegt*/
34
35
36
37
38    public void zeichneSpiel() { /* zeichnet den aktuellen Spielstand*/
39
40
41    public boolean getStatus() {
42
43    public static boolean prüfeEingabe(String Wort) { //Überprüft ein Wort ob es mindestens 2 Buchstaben hat und alle
44    buchstaben akzeptiert sind. Diese Überprüfung muss extern durchgeführt werden.
45
46
47    }
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
```



## 4. SPIELAUFBAU



w e i \_ \_ a \_ \_ \_ e \_

Schonmal geratene Buschstaben: [f, u, q, x, p, ü, ä, ö, z]

Ein Buchstabe bitte:



# **VIELEN DANK!**

Celine Gugel, Lena Steinbrink, Valentin Müller,  
Jannik Oßwald , Thi Tuong Vy Nguyen