

# Lernatelier: Projektdokumentation

---

Olivier Kistler

Datum	Version	Änderung	Autor
24.08.21	0.0.1		Kistler
31.08.21	0.3.3	Zufallszahl wird generiert, Benutzer kann so viel raten wie er will, Hinweis wird ausgegeben, Rateversuche werden ausgegeben, Fehlereingaben werden umgangen	Kistler
07.09.21	0.6.6	Farbige Schrift, Easter Egg, Name wird erfragt, Spiel kann wiederholt werden	Kistler
14.09	1.0.0	Hintergrund-musik	Kistler

## 1. Informieren

### 1.1 Ihr Projekt

---

Ich programmiere ein Programm, welches eine zufällige Zahl generiert und den User die Zahl erraten lässt.

### 1.2 Quellen

---

#### Was weiss ich nicht?

- Wie kann man eine zufällige Zahl generieren
- Wie kann man die Anzahl Rateversuche speichern
- Wie bringt man es hin, dass das Programm auf Fehlereingaben reagiert
- Wie vergleicht man die geratene Zahl mit der Zufallszahl und gibt darauf eine passende Antwort

#### Meine Quellen:

Stackoverflow

<https://stackoverflow.com/questions/4620565/int-tryparse-syntactic-sugar>

Microsoft Docs

Youtube Video: <https://www.youtube.com/watch?v=KRjCxMSEwts>

<https://www.youtube.com/watch?v=wAYN2BABnG0&t=303s>

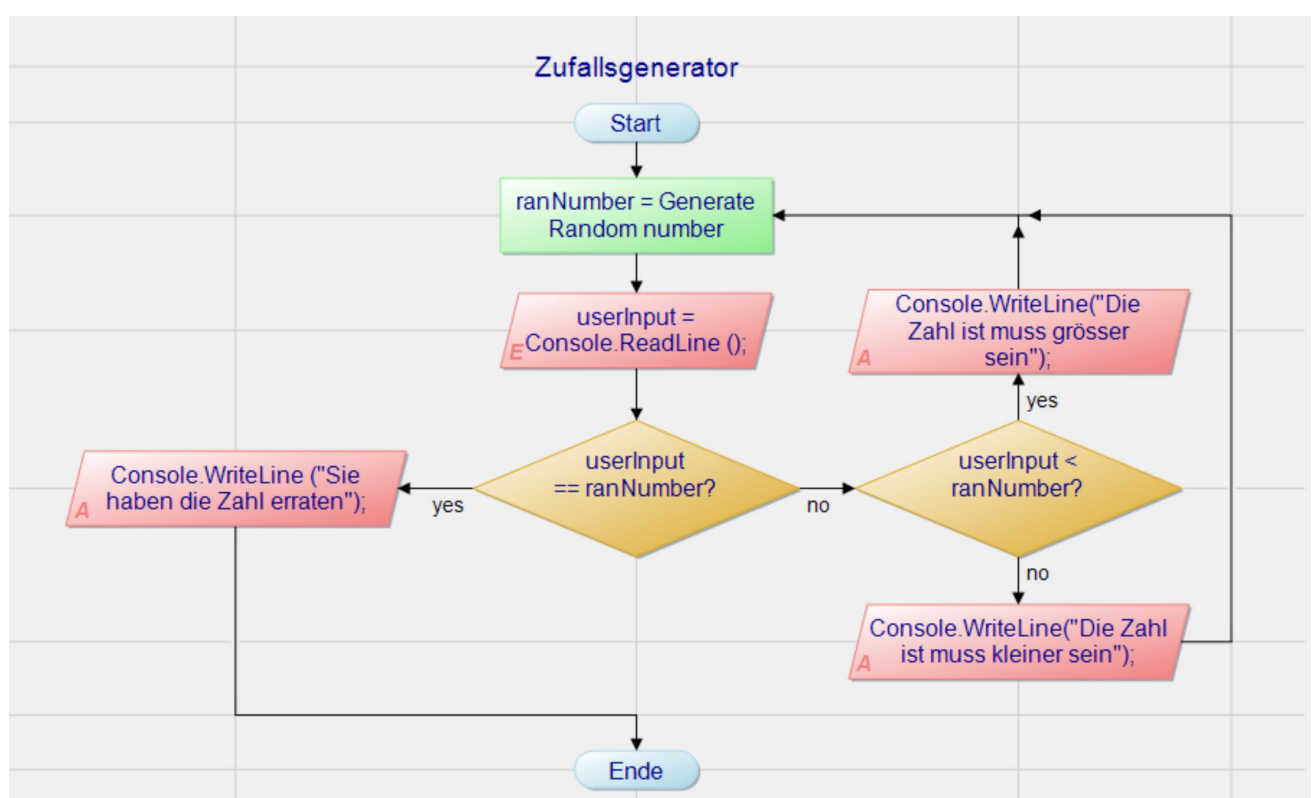
Programming Reddit

GitHub

### 1.3 Anforderungen

Nummer	Muss / Kann?	Funktional? Qualität? Rand?	Beschreibung
1	Muss	Funktional	Der Computer speichert eine Zufallszahl zwischen und mit 1 bis 100 als Geheimzahl.
2	Muss	Funktional	Der Benutzer kann Zahlen raten.
3	Muss	Funktional	Für jede der geratenen Zahlen gibt der Computer einen Hinweis aus:
4	Muss	Qualität	Wenn die Geheimzahl erraten wurde, soll die Anzahl der Rateversuche ausgegeben werden.
5	Muss	Funktional	Das Programm soll mit Fehleingaben umgehen oder sie vermeiden können.
6	Kann	Qualität	Das System schreibt mit verschiedenen Farben
7	Kann	Qualität	User kann Spiel neu starten
8	Kann	Qualität	Musik wird gespielt
9	Kann	Qualität	Kleine Easter Eggs sind eingebaut

### 1.4 Diagramme



## Testfälle

---

Nummer	Voraussetzung	Eingabe	Erwartete Ausgabe
1.1	IDE ist gestartet	Programm erstellt eine Zufallszahl zwischen 1 und 100	Zufällige Zahl
2.2	IDE läuft, Programm gestartet	Der User errät die Zahl	Das ist die geheime Zahl
3.3	IDE läuft, Programm gestartet	Der User rät zu klein	Die gesuchte Zahl ist grösser
3.4	IDE läuft, Programm gestartet	Der User rät zu gross	Die gesuchte Zahl ist kleiner
4.6	IDE gestartet, richtige Zahl wird geraten	Der User errät die Zahl	Gratulation, Sie haben die geheime Zahl erraten
5.7	IDE gestartet, Worteingabe	User gibt «Hallo» ein	Bitte geben Sie eine Zahl ein!
4.8	IDE läuft, Programm gestartet	User hat die Zahl erraten	Sie haben x Versuche gebraucht
6.9	IDE läuft, Programm gestartet	Text	Farbige Schrift
7.10	IDE läuft, Programm gestartet	Ich will das Spiel wiederholen	Erraten sie eine Zahl zwischen 1-100
8.11	IDE läuft, Programm gestartet	keine	Kahoot Musik
9.12	IDE läuft, Programm gestartet	6969	Witziges Symbol-Art
9.13	IDE läuft, Programm gestartet	1976	Apple Logo

## 2. Planen

Nummer	Frist	Beschreibung	Zeit (geplant)
1.1	31.08	Das Programm generiert eine zufällige Zahl zwischen 1-100 und speichert diese	45 min
2.2	31.08	Der Benutzer kann so viel Raten wie er will	45 min
3.3	31.08	Der Computer vergleicht die geratene Zahl mit der Zufallszahl und erkennt, ob diese grösser oder kleiner ist	45 min
3.4	31.08	Der Computer gibt einen passenden Tipp aus	45min
4.5	31.08	Die richtige Zahl wird erkannt und eine Gratulationsnachricht erscheint	45 min
4.6	31.08	Die Anzahl Rateversuche wird ermittelt und ausgegeben	2x 45min
5.7	31.08	Das Programm erkennt Fehler und gibt eine passende Antwort darauf	45min
0.78	14.09	Letzte Fehler beheben und kleinere Extras programmieren	2x 45min
6.9	14.09	Farbige Schrift	45 min
7.10	14.09	Das Spiel kann wiederholt werden	45 min
8.11	14.09	Im Hintergrund läuft Musik	90 min
9.12	14.09	Beim eingeben von 1976 wird das Apple Logo gedruckt	30 min
9.13	14.09	Beim eingeben von 6969 wird ein lustiges Symbol-Art gedruckt	15 min

## 3. Entscheiden

Was mache ich, wenn ich fertig bin?

- Farbige Schrift
- Spielwiederholung
- Name erfragen
- Easter Egg
- Hintergrundmusik

#### 4. Realisieren

Nummer	Frist	Beschreibung	Zeit (geplant)	Zeit (effektiv)
1.1	31.08	Das Programm generiert eine zufällige Zahl zwischen 1-100 und speichert diese	45 min	20 min
2.2	31.08	Der Benutzer kann so viel Raten wie er will	45 min	20 min
3.3	31.08	Der Computer vergleicht die geratene Zahl mit der Zufallszahl und erkennt, ob diese grösser oder kleiner ist	45 min	20 min
3.4	07.09	Der Computer gibt einen passenden Tipp aus	45min	15 min
4.5	07.09	Die richtige Zahl wird erkannt und eine Gratulationsnachricht erscheint	45 min	15 min
4.6	07.09	Die Anzahl Rateversuche wird ermittelt und ausgegeben	2x 45min	30 min
5.7	14.09	Das Programm erkennt Fehler und gibt eine passende Antwort darauf	45min	45 min
0.8	14.09	Letzte Fehler beheben	45min	20 min
6.9	14.09	Farbige Schrift	45 min	15 min
7.10	14.09	Spiel kann wiederholt werden	45 min	15 min
8.11	14.09	Es läuft Hintergrundmusik	90 min	90 min
9.12	14.09	Beim eingeben von 1976 wird das Apple Logo gedruckt	30 min	20 min
9.13	14.09	Beim eingeben von 6969 wird ein lustiges Symbol-Art gedruckt	15 min	15 min

#### 5. Kontrollieren

##### 5.1 Testprotokoll

Nummer	Datum	Resultat	Durchgeführt
1.1	21.09	OK	Kistler
2.2	21.09	OK	Kistler
3.3	21.09	OK	Kistler
3.4	21.09	OK	Kistler
4.6	21.09	OK	Kistler
5.7	21.09	OK	Kistler
4.8	21.09	OK	Kistler
6.9	21.09	OK	Kistler
7.10	21.09	OK	Kistler
8.11	21.09	OK	Kistler
9.12	21.09	OK	Kistler
9.13	21.09	OK	Kistler

Meine Testfälle sind alle positiv ausgefallen und es gibt keine Probleme mit dem Programm

## 6. Auswerten

---

**Gut gemacht:** Ich wurde schnell mit den Mindestanforderungen des Programms fertig und hatte so mehr Zeit, um Extras einzubauen und noch mehr neue Sachen zu lernen.

**Nächstes Mal besser machen:** Ich sollte mehr Testfälle schreiben und die Anforderungen genauer beschreiben.