

# Lernatelier: Projektdokumentation

---

Datum	Version	Änderung	Autor
1.9.2021	0.1	Start des Programms	Bajramovic Kenan
8.9.2021	0.2	Funktionale Änderungen	Bajramovic Kenan
15.9.2021	0.3	Funktionale und qualitative Änderungen	Bajramovic Kenan
22.9.2021	1.0	Finale Version	Bajramovic Kenan

## 1. Informieren

### 1.1 Ihr Projekt

---

Der Benutzer soll die per Zufall generierte Zahl erfolgreich erraten.

### 1.2 Quellen

---

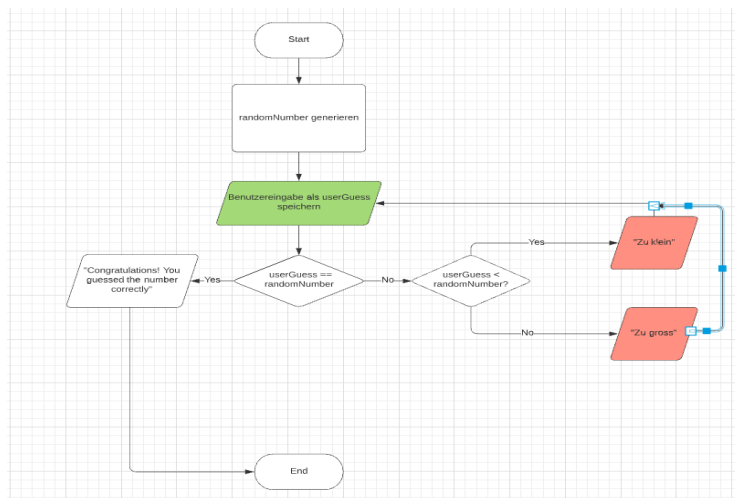
Microsoft Einführung zu C# und Stack Overflow.

### 1.3 Anforderungen

---

Nummer	Muss / Kann?	Funktional? Qualität? Rand?	Beschreibung
1	Kann	Funktion	Der Computer speichert eine Zufallszahl zwischen 1 bis 100 als Geheimzahl ab.
2	Kann	Qualität	Für jede der geratenen Zahlen, gibt der Computer einen Hinweis aus.
3	Kann	Qualität	Wenn die Geheimzahl erraten worden ist, soll die Anzahl der Rateversuche ausgegeben werden.
4	Muss	Funktion	Das Programm soll mit Fehleingaben umgehen oder sie vermeiden können.
5	Kann	Funktion	Der Benutzer kann Zahlen raten/eingeben.
6	Kann	Qualität	Ein Aufmunterungstext soll erscheinen, falls der Benutzer die Zahl falsch erraten tut.

## 1.4 Diagramme



## 1.5 Testfälle

Nummer	Voraussetzung	Eingabe	Erwartete Ausgabe
1-5	Visual Studio starten, Programm fertigstellen	Code, Zahl eingeben, enter	Beschreibung 1-5 soll einwandfrei funktionieren
1	Visual Studio starten, Code eingeben	Code damit der Computer eine Zufallszahl speichert	Computer generiert Zufallszahl.
2	Der Benutzer kann Zahlen eingeben.	Code für die Hinweise	Hinweise (Zahl zu klein/gross)
3	Hinweise sollen hilfreich sein.	Code für die Ausgabe der Rateversuche	Ausgabe der Anzahl Rateversuche.
4	Der Benutzer kann Zahlen eingeben.	Keine Eingabe ausser Code	Soll Fehleingaben umgehen.
5	Der Benutzer hat ein Keyboard, worauf er tippen kann.	Die Zahleingabe	Benutzer kann Zahlen eingeben.

## Das Programm:

```
1 using System;
2
3 namespace Random
4 {
5     class Program
6     {
7         static void Main(string[] args)
8         {
9             Random rd = new Random();
10
11             int rand_num = rd.Next(1, 100);
12             Console.WriteLine(rand_num);
13             Console.WriteLine("Random number got generated");
14             Console.WriteLine("Please a number between 1 - 100");
15             int userGuess = Convert.ToInt32(Console.ReadLine());
16             int count = 0;
17
18             if (userGuess > rand_num)
19             {
20                 Console.WriteLine("Your Number is bigger");
21             }
22             if (userGuess < rand_num)
23             {
24                 Console.WriteLine("Your Number is smaller");
25             }
26             while (userGuess != rand_num)
27             {
28                 Console.WriteLine("You get an another try. GOOD LUCK!");
29                 userGuess = Convert.ToInt32(Console.ReadLine());
30                 if (userGuess > rand_num) { Console.WriteLine("Your Number is bigger"); }
31                 if (userGuess < rand_num) { Console.WriteLine("Your Number is smaller"); }
32                 if (userGuess == rand_num) { Console.WriteLine("You got right!"); }
33             }
34         }
35     }
36 }
```

## 2. Planen

Nummer	Frist	Beschreibung	Zeit (geplant)
1	1.9	Microsoft C# Tutorials anschauen	1h
2	8.9	Ideensammlung bei Stack Overflow	30min
3	8.9	Wenn die Geheimzahl erraten worden ist, soll die Anzahl der Rateversuche ausgegeben werden.	20min
4	8.9	Das Programm soll Fehleingaben umgehen oder sie vermeiden können.	1h
5	8.9	Der Benutzer kann Zahlen raten/eingeben.	20min

## 3. Entscheiden

Falls ich noch genügend Zeit habe, nachdem das Projekt fertiggestellt ist, dann füge ich noch zusätzliche features ein.

## 4. Realisieren

Nummer	Frist	Beschreibung	Zeit (geplant)	Zeit (effektiv)
1	1.9	Microsoft C# Tutorials anschauen	1h	45min
2	1.9	Ideensammlung bei Stack Overflow	30min	35min
3	8.9	Wenn die Geheimzahl erraten worden ist, soll die Anzahl der Rateversuche ausgegeben werden.	20min	30min
4	8.9	Das Programm soll mit Fehleingaben umgehen oder sie vermeiden können.	1h	1h
5	8.9	Der Benutzer kann Zahlen raten/eingeben	20min	10min

## 5. Kontrollieren

### 5.1 Testprotokoll

Nummer	Datum	Resultat	Durchgeführt
1	22.9.2021	Ok	Kenan Bajramovic
2	22.9.2021	Ok	Kenan Bajramovic
3	22.9.2021	Ok	Kenan Bajramovic
4	22.9.2021	Not Ok	Kenan Bajramovic
5	22.9.2021	Ok	Kenan Bajramovic

Das Programm läuft einwandfrei, bedarf aber erhebliche Optimierung im Bereich "Fehleingaben umgehen".

## 6. Auswerten

Das Programm ist gut gelungen, die Aufgabe an sich hat mir grosse Freude bereitet und ich habe auch vieles dazu gelernt. Nächstes Mal möchte ich aber mehr while-Schleifen einbauen.