Lösungskonzept zur Parametrisierung

# Parameter

Nachfolgend sind die Parameter aufgelistet, welche laut Aufgabenstellung bei der Generierung des Tunnels zu setzen sind:

* Strecke
  + Länge des Tunnels
  + Gerade Abschnitte/Kurven
  + Bodenbeschaffenheit
* Tunnelbeschaffenheit
  + Drehrichtung
  + Drehgeschwindigkeit
  + Tunnelquerschnitt
  + Tunnelwand-Muster

# Varianten zur Parameter-Eingabe

In den folgenden Abschnitten werden verschiedene Varianten zur Parameter-Eingabe angeschaut und im Anschluss verglichen. Die Parameter-Eingabe muss nicht durch den Endbenutzer erfolgen, sondern durch Vertreter des iHome-Labs. Aus diesem Grund muss die Eingabe nicht zwingend über ein im Programm integriertes GUI erfolgen. Das Verändern der Parameter soll aber das Öffnen von Unity/einer IDE nicht voraussetzen.

## Variante 1: Eingabemaske beim Start der Tunnel-Simulation

Die Parameter können direkt vom Benutzer, der die VR-Brille trägt, vor Start der Simulation gesetzt werden. Unity bietet einfache Steuerelemente an, die dies ermöglichen.

## Variante 2: Parameter in einem XML-File

Die Parameter werden strukturiert in einem XML-File eingegeben. Anschliessend wird die VR-Simulation gestartet und die Werte werden aus der Datei übernommen.

## Variante 3: Ein simpler Editor zum Tunnelbau

Ein simples Tool zum zeichnen des Tunnels. Die Benutzeroberfläche des Tools könnte den Tunnel aus der Vogelperspektive darstellen und das Setzen von neuen Tunnelabschnitten ermöglichen. Das Tool wird eine Datei erzeugen (xml, txt oder anderes Format), die dann beim Programmstart eingelesen und zur Generierung des Tunnels verwendet wird. Im Vergleich zu den ersten beiden Varianten wird bei dieser Lösung kein zufälliger Tunnel generiert.

### Vergleich der drei Varianten

Zur Auswahl der besten Variante werden alle Konzepte verglichen. Die Kriterien an eine Parametereingabe sind die folgenden:

* Einfachheit der Eingabe: Die Parametereingabe soll möglichst kein technisches Know-How erfordern. (1= Gelernter Informatiker, 3=Technischer Assistent, 5= Microsoft-Office Anwender)
* Komfort: Wie angenehm ist es für den Benutzer, die Parameter einzugeben. (1=Mehrere Zusatzprogramme notwendig, 3=Ein Zusatzprogramm notwendig, 5=Alles in der Applikation)
* Entwicklungsaufwand: Wie aufwändig ist die Entwicklung? (1= sehr aufwändig und grosser Zeitaufwand, 3=tragbarer Aufwand ohne zusätzliche Kenntnisse/Libraries, 5=wenig Aufwand und kann gut in C# umgesetzt werden.)
* Genauigkeit: Wie gezielt und präzise können die Parameter gesetzt werden? (1=grobe Definition ohne Massangabe, 3=Masse in bestimmten Intervallen, 5=freie Massangaben)
* Erweiterbarkeit: Wie einfach können neue Parameter hinzugefügt werden? (1=Entwickler für die Anpassung benötigt, 3=Technischer Assistent, 1=Leihe)

Die drei Varianten wurden an den genannten Kriterien gemessen und verglichen. Jedes Kriterium wurde zusätzlich gewichtet.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Gewichtung** | **Variante 1 -GUI** | **Variante 2 -XML** | **Variante 3 -Editor** |
| **Einfachheit** | 3 | 5 | 3 | 5 |
| **Komfort** | 2 | 3 | 3 | 4 |
| **Entwicklungsaufwand** | 3 | 2 | 5 | 1 |
| **Genauigkeit** | 1 | 3 | 5 | 3 |
| **Erweiterbarkeit** | 1 | 1 | 5 | 1 |
| **Gesamt** | 10 | 31 | 40 | 30 |

Aus dem Vergleich geht hervor, dass sich die Parameter-Eingabe über eine XML-Datei am besten eignet. Deshalb wird dieser Ansatz weiter verfolgt.

# Auswahl des Dateiformats

Was die Parameter-Eingabe angeht stehen noch einige Fragen offen, die mit dem Auftraggeber zu klären sind. Ein erstes Konzept der XML-Struktur ist jedoch bereits vorhanden und hier zu sehen:

