

## **Tic-Tac-Toe mit Python und GUI – Beschreibung und Umsetzung**

### **1. Einführung**

In diesem Projekt soll ein Tic-Tac-Toe-Spiel mit Python und einer grafischen Benutzeroberfläche (GUI) erstellt werden. Das Ziel ist es, ein einfaches, aber funktionales Programm zu entwickeln, das entweder mit oder ohne Spiellogik funktioniert. Die Umsetzung erfolgt hauptsächlich mit der Bibliothek Tkinter, aber es werden auch alternative Bibliotheken vorgestellt.

### **2. Wichtige Python-Bibliotheken für GUI-Entwicklung (ChatGPT)**

#### **2.1 Tkinter**

Tkinter ist die Standard-GUI-Bibliothek in Python und wird direkt mit Python installiert. Sie ist einfach zu verwenden und eignet sich sehr gut für kleinere Projekte wie Tic-Tac-Toe. Man kann mit wenigen Zeilen Code Fenster, Buttons und Labels erstellen. Tkinter ist besonders gut für Einsteiger geeignet, da es leicht verständlich und gut dokumentiert ist.

#### **2.2 CustomTkinter**

CustomTkinter ist eine modernere Version von Tkinter, die auf dem gleichen Prinzip basiert, aber ein schöneres Design bietet. Damit kann man ansprechende, dunkle oder moderne Oberflächen gestalten, ohne viel zusätzlichen Aufwand.

#### **2.3 PyQt5 / PySide6**

PyQt5 (bzw. PySide6) sind leistungsstarke Bibliotheken zur GUI-Entwicklung. Sie bieten sehr viele Möglichkeiten, sind aber deutlich komplexer als Tkinter. Für ein kleines Projekt wie Tic-Tac-Toe sind sie oft zu umfangreich.

#### **2.4 Pygame**

Pygame ist eigentlich für Spieleentwicklung gedacht und nicht speziell für klassische GUIs. Es kann aber auch für Tic-Tac-Toe verwendet werden, wenn man Animationen oder grafische Effekte einbauen möchte.

### **3. Schritte zur Umsetzung des Tic-Tac-Toe-Projekts**

1. Planung: Überlege dir, wie das Spielfeld aussehen soll (3x3 Raster, Buttons für jede Zelle).
2. GUI-Erstellung: Mit Tkinter ein Fenster erstellen und neun Buttons anordnen.
3. Spiellogik: Funktionen programmieren, die prüfen, ob X oder O gewonnen hat.
4. Interaktion: Beim Klick auf einen Button soll sich das Symbol (X oder O) ändern.
5. Erweiterung: Option für Neustart oder Spieler gegen Computer hinzufügen.

### **4. Schwierigkeitseinschätzung (ChatGPT)**

Leicht:

- Installation von Python und Tkinter

- Erstellung eines Fensters und Platzierung von Buttons
- Grundlegende Spiellogik mit if-Abfragen

Mittel:

- Erkennung von Gewinnbedingungen
- Wechsel zwischen X und O korrekt umsetzen

Schwer:

- Einbau eines Computergegners (KI)
- Schönes Layout mit ansprechendem Design (CustomTkinter)
- Speicherung von Spielständen oder Highscores

## **5. Fazit**

Das Projekt ist ideal, um den Umgang mit Python und GUI-Programmierung zu lernen. Mit Tkinter kann man schnell sichtbare Ergebnisse erzielen. Wer möchte, kann das Spiel später mit CustomTkinter oder Pygame erweitern, um es optisch und funktional zu verbessern.