

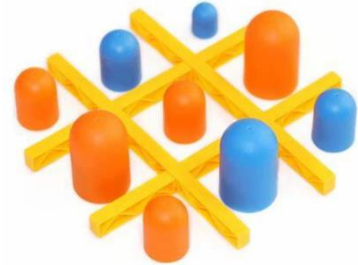
Big Eat Small- Dokumentation

Tic-Tac-Toe Variante

Beschreibung

Bei Big Eat Small handelt es sich um eine Tic-Tac-Toe Variante bei der Felder mehrfach besetzt/übertrumpft werden können.

(Für eine Beschreibung des Spiels und der Regeln siehe [\(Big Eat Small\) Tic-Tac-Toe Game – Deaflo](#))



Die In-Scope, Nice-To-Have und Out-of-Scope wurden angepasst, da vor allem einige Nice-to-Have's vom Zeitaufwand zu groß waren. Generell war die Logik des Spiels (Win-Condition, Draw-Condition etc.) umfangreicher und komplexer als zuerst angenommen.

In-Scope

- Offline-Multiplayer-Modus für zwei Personen
- Main-Menü zum Auswählen der Spielmodi oder einstellen der Settings
- Nach Ende eines Spiels wird der Gewinner auf einem „Winscreen“ geehrt
- Bei einem Unentschieden wird der Unentschieden-Screen aufgerufen
- Musik und Soundeffekte

Nice-to-have

- Ändern der Musik
- Verschiedene Hintergründe
- Verschiedene Skins/Muster für die Spielfiguren

Out-of-Scope

- Online-Modus
- Die Möglichkeit ein „Best-of-Three“ oder „Best-of-Five“ zu spielen.
- Single-Player-Modus gegen einen „Computer-Gegner“
- Schwierigkeitsgrade des „Computer-Gegners“
- Hintergründe, Musik und Skins durch ein Level-System freischaltbar

Quellcode

BigEatSmall läuft auf allen gängigen Betriebssystemen (Windows, Mac, Linux).

Das Programm besteht aus drei python-Dateien. Der „**big_eat_small_main.py**“-Datei, der „**button.py**“-Datei und der „**piece_rep.py**“-Datei. Die „**big_eat_small_main.py**“-Datei stellt dabei die ausführbare Datei dar, durch die das Spiel gestartet wird.

Nach Starten des Programms landet man zunächst in einem Menü. Dieses Menü und die dazugehörigen „Buttons“ habe ich mit kleinen Änderungen von folgender Quelle übernommen: [baraltech/Menu-System-PyGame](https://github.com/baraltech/Menu-System-PyGame)

Die „**button.py**“-Datei ist ebenfalls aus derselben Quelle stammend. Die Button-Klasse habe ich leicht angepasst und habe sie ebenfalls für die auswählbaren Spielfiguren verwendet. Generell ermöglicht die Button-Klasse die einfache Instanziierung von Button-Objekten, welche einen Hover-Effekt, Text, ein Hintergrundbild und weitere nützliche Felder besitzen.

Die „**piece_rep.py**“-Datei beinhaltet die **PieceRep-Klasse**. Diese ist eine kleine Hilfsklasse die ich gebaut habe, um das Übertrumpfen kleinerer Spielfiguren auf dem Tic-Tac-Toe-Feld zu ermöglichen. Die gesetzten Spielfiguren haben dabei sowohl das Wissen über den Spieler, der sie gesetzt hat, als auch über den Rang bzw. ihrer Größe. Durch Vergleichen des Ranges als auch kontrollieren anderer Kriterien wird so entschieden, ob eine Figur an eine Stelle gesetzt werden darf.

Die „**big_eat_small_main.py**“-Datei enthält die loop-Funktionen aller im Spiel enthaltenen screens. Dazu gehört das Main-Menü, der Versus-Mode (Das Spiel), die Settings sowie der win- und draw-screen. Der win-screen gratuliert dem Sieger, während der draw-screen ein Unentschieden verkündet. In den Settings ist es möglich die Musik sowie den Hintergrund zu ändern. Die meiste Komplexität findet sich jedoch im game-loop des Versus-Modes. Hier finden sich Funktionen die das „Zeichnen“ der Spielfiguren in das Tic-Tac-Toe-Feld realisieren oder nach dem Zug eines Spielers auf einen Sieg oder ein Unentschieden prüfen.

Für die Logik der „Win-Condition“, das Zeichnen der Figuren und des Feldes und anderen wesentlichen Bestandteilen des Tic-Tac-Toe-Spieles habe ich mich zunächst stark an der unten verlinkten Website und dem darin enthaltenen Quellcode orientiert. Im weiteren Verlauf des Programmierens meines Spieles musste ich aber jede eigene Methode grundsätzlich eigenständig aufbauen.

[Tic Tac Toe GUI In Python using PyGame - GeeksforGeeks](https://www.geeksforgeeks.org/tic-tac-toe-gui-in-python-using-pygame/)

Fazit

Grundsätzlich wurden alle In-Scope-Anforderungen sowie einige Nice-to-Have-Anforderungen erfüllt. Mit der früheren Projektbeschreibung habe ich viele Nice-To-Haves in ihrem Aufwand stark unterschätzt, wodurch ich diese etwas anpassen musste. Somit gibt es nun leider keinen Singleplayer Modus gegen Computergegner. Es kamen viele neue Aspekte hinzu, die bei einem klassischen Tic-Tac-Toe Spiel nicht existieren. Zum Beispiel das überschreiben von Spielfiguren, die Limitierung jeder Spielfigur auf drei Benutzungen sowie den unterschiedlichen Graphiken. Diese wurden alle implementiert und das Programm läuft stabil und flüssig. Generell bin ich sehr zufrieden mit dem Resultat meiner Arbeit.

Ausführen des Programms

Nachdem der Quellcode durch die ZIP-Datei oder über das Klonen der GitHub-Repository erlangt wurde, kann die Main-Datei: „big_eat_small_main.py“ in einer Python Umgebung geöffnet werden.

Bevor diese jedoch ausgeführt werden kann, muss zuerst pygame installiert werden.

Wenn Sie eine IDE wie PyCharm oder ähnliche verwenden können Sie das pygame-Package über eingebaute Untermenüs installieren.

Auf dem Klassischen Weg geht es über die Kommandozeile mit dem Befehl: `pip install pygame`

Mehr Details zur installation finden Sie auf folgender Seite: [GettingStarted - pygame wiki](#)

Nun kann „big_eat_small_main.py“ ausgeführt werden und das Spiel startet.

Viel Spaß!