

Dokumentation TicTacToe Refactoring

Einleitung

Die Solution wurde in drei Projekte unterteilt, ein Konsolenprojekt und zwei Klassenbibliotheken. Das Konsolenprojekt ist nur für die Anzeige des Spiels und die Verarbeitung von Nutzereingaben durch die Konsole verantwortlich. Eine der Klassenbibliotheken enthält konkrete Implementationen für das Spiel TicTacToe. Bei der Anderen handelt es sich um ein allgemeines Game Framework. Die Struktur wurde so gewählt, dass soviel Quellcode wie möglich wieder verwendet werden kann.

Game Framework

Das Game Framework enthält Klassen, die für weitere Spieleprojekte wiederverwendet werden können. Es beinhaltet Utility-Klassen aber auch Basisklassen für die Strukturierung eines Programms. Es beinhaltet zur Zeit eine generische Event Engine die Basis für eine ereignisgetriebene Architektur- und eine generische Implementation des Singleton-Musters ist. Diese Klassenbibliothek könnte noch um weitere generische Implementationen von Architektur- und Entwurfsmustern und um Utility-Klassen erweitert werden.

Grid

Bei der Grid-Klasse handelt es sich um eine generische Klasse zur Repräsentation eines Grids aus Tiles. Den Tiles kann ein typisierter Wert zugewiesen werden. So können Tiles zum Beispiel Kosten für die Passierung enthalten oder einen besetzten/freien Zustand aufweisen. Solche Grids werden häufiger in der Spieleentwicklung gebraucht. Hier wird das Grid genutzt um das Spielfeld zu verwalten. Die Tiles haben Referenzen auf ihre benachbarten Tiles.

TicTacToeLib

Diese Klassenbibliothek implementiert die Logik für das Spiel TicTacToe. Sie enthält keine Abhängigkeiten zur Konsole. Sie enthält das gesamte Grundgerüst für das Spiel. Es muss lediglich die Anzeige und gegebenenfalls das Modell für das Spielbrett konkret implementiert werden.

TicTacToeConsole

Dieses Projekt implementiert die Anzeige des Spielbretts in der Konsole sowie die Verarbeitung von Nutzereingaben und benachrichtigt den Nutzer im Falle ungültiger Eingaben oder im Falle des Spielgewinns.

Design-Fehler in der alten Solution

Die Klasse UserInput in der Klassenbibliothek TicTacToe.Lib steht in Abhängigkeit zu der Konsole und sollte deshalb im Konsolenprojekt beinhaltet sein. Die GameBoard-Klasse verstößt gegen das Single-Responsibility-Prinzip, denn sie ist zuständig für die Spielerverwaltung, die Verwaltung des Spielfelds sowie die Prüfung von Siegbedingungen. Das GameBoard eignet sich des Weiteren auch nur für diese Art der Anzeige.

Wenn eine grafische Oberfläche implementiert werden soll, muss das gesamte GameBoard umgebaut werden, oder es muss eine sehr unübersichtliche Anzeige des Spielbretts auf Basis dessen binärer Repräsentation implementiert werden. Die binäre Repräsentation des GameBoards ist im Vergleich zum objektorientierten Ansatz sehr viel unverständlicher und sorgt dafür, dass der Code nur mithilfe vieler Kommentare verstanden werden kann. Der Spielfluss ist über GameStates in dem Konsolenprojekt implementiert und somit davon abhängig. Der Spielfluss müsste also neu implementiert werden, wenn eine andere Anzeige implementiert werden soll. Wenn ein anderer UserInput implementiert werden soll, zum Beispiel ein Auswählen eines Spielfelds durch Mausklick, ist dies mit dem aktuellen Aufbau nicht sauber realisierbar; auch hier würde sich ein objektorientierter Ansatz anbieten.