

Dokumentation Tic Tac Toe

von Sabrina Leuenberger WIBB2.61

Das TicTacToe Spiel kann von zwei Spielern übers Netz gespielt werden.

Zuerst wird der Server über die Klasse TicTacToeServer.java im gameServer gestartet und über die Konsole ein Port definiert. Danach können die beiden Clients gestartet werden (Klasse JavaFX_App_Driver.java im gameClient).

Im oberen Bereich des Fensters des Clients kann die Verbindung zum Server aufgebaut werden indem die IP-Adresse und die Port-Nummer des Servers angegeben werden und auf den Knopf „connect“ geklickt wird.

Im Weiteren wird im unteren Bereich des Fensters im linken Feld angegeben, was die nächste Aktion des entsprechenden Spielers ist. Seitlich sehen die Spieler welches Symbol (X bzw. O) sie spielen. Um die Spiel-Session zu beenden wird das Fenster geschlossen.

Der Server kann über eclipse gestoppt werden, wenn man ihn nicht mehr braucht. Er kann aber auch laufen gelassen und eine neue Spielsession der Clients gestartet werden.

Das Programm basiert auf dem Framework jat_v2 und dem MinimalApps von Prof. Dr. Brad Richards. Zudem wurden einzelne Codefragmente (z.B. für die Methode fromString() der Klasse Board) von Stackoverflow verwendet. Gewisse Funktionalitäten des Frameworks wurden weggelassen oder abgeändert (z.B. SplashScreen entspricht der Version 1 des Frameworks), da sie im Spiel nicht implementiert wurden.

Um über den Programmverlauf Informationen zu erhalten, wurde an verschiedensten Stellen der Logger, welcher im Framework schon angelegt ist, eingesetzt. Neben dem Logger werden teilweise Informationen auch über eine Alert Box ausgegeben.

Die Architektur des Spieles wurde weitgehend vom Framework übernommen. Zugefügt wurde noch eine Player Klasse mit zwei statischen Konstanten welche die beiden Spieler repräsentieren. Es wurde die Form der Klasse gewählt (und nicht des Enums), damit gegebenenfalls noch objektspezifische Attribute die keine Konstanten darstellen, wie zum Beispiel Anzahl der gewonnenen Spiele, eingefügt und das Spiel somit konsistent ergänzt werden könnte. Das Model wurde mit zwei Klassen ergänzt: die Board Klasse, welche wiederum von der GridContent-Klasse ergänzt wird. Für die Klasse View wurde die Klasse Tile geschaffen, welche von der Klasse Button erbt, damit ein Button um spielspezifische Attribute (Position des Buttons im Spielfeld) ergänzt werden konnte. Dabei wurde darauf geachtet, dass die beschriebenen Abhängigkeiten nicht durchbrochen wurden, indem darauf verzichtet wurde von einer weiteren Klasse auf diese zusätzlichen Klassen zu zugreifen.

Basierend auf dieser Grundstruktur und den Anforderungen des Spiels wurde im Weiteren auf eine Art und Weise vorgegangen, die man so ungefähr mit „error-driven-design“ beschreiben könnte. Es wurde Funktionalität um Funktionalität zugefügt und der Aufbau immer wieder so angepasst, dass die jeweils generierten Fehlermeldungen behoben werden konnten. Ein Beispiel dafür ist die main-Methode in Kombination mit der run-Methode des TicTacToe-Servers.