

## Abweichung von der Spezifikation:

### ➤ Turniermodus

FXML steht, Klasse wurde vorbereitet aber für die Implementierung der Logik hatten wir zu wenig Zeit zur Verfügung.

### ➤ Einfache KI als Gegner

FXML steht, Klasse wurde vorbereitet aber für die Implementierung der Logik hatten wir zu wenig Zeit zur Verfügung.

## Gedanken zum Testen:

Wir würden das Projekt nach folgenden Testfällen testen:

| Test case  | Expected Value |
|--|----------------|
| Werden alle Mühlen erkannt?  | Ja             |
| Kann man einen Stein entfernen wenn man eine Mühle gemacht hat?  | Ja             |
| Kann man vom Gegner einen Stein der sich in einer geschlossenen Mühle befindet entfernen?                        | Nein           |
| Kann man einen Stein vom Gegner der sich in einer Mühle befindet entfernen, wenn nur noch Mühlen vorhanden sind? | Ja             |
| Kann man springen, wenn man nur noch 3 Steine hat?   | Ja             |
| Kann man die Steine immer nur um eins verschieben?   | Ja             |
| Wird erkannt wer gewonnen hat?   | Ja             |
| Funktionieren alle Buttons und Menü Punkte?  | Ja             |
| Funktioniert das Aufgeben?   | Ja             |

## Abweichungen vom Entwurfs- zum Final-Klassendiagramm:

Eine zusätzliche Klasse Mill, sie wird gebraucht um Mühlen abzuspeichern.

Die Klasse Game hat einige zusätzliche Methoden, da sie nun noch mehr Logik enthält als beim Entwurf gedacht.

Wir haben für jeden Screen einen eigenen Controller. Beim Entwurf dachten wir, wir könnten alles mit einem einzelnen Controller schaffen.

Da die Methode Game alle Pieces verwaltet:

- Ist die Methode isOldPointNext nicht mehr in der Klasse Point sondern in der Klasse Game.
- Gibt es die Methode selectPiece nicht mehr in der Klasse Piece und View, jedoch ein Attribut vom Typ Piece mit Setter in der Klasse Game.
- Ist die Methode movePiece von der Klasse View in die Klasse Game verschoben worden.