

# Architektur

## Erklärung / Begründung der Architektur

Als grobes Designpattern haben wir das Model View Controll Design Pattern gewählt und grundsätzlich versuchen wir möglichst einen objektorientierten Ansatz umzusetzen. Dem entsprechend wurden Klassen in bis dato drei verschiedene Packages unterteilt:

- Controller: hier befinden sich Klassen die den Ablauf durch Inputs und Regeln bestimmen
  - Class Input: ließt den Input des Users und steuert den Ablauf des Spiels durch eine Ein-, Ausgabe-Routine
  - Class Check: beinhaltet die Regeln zu Schach- und Schachmatt-Situationen und regelt dadurch den Spielverlauf
- Model: hier finden sich Klassen, die nach einem objektorientierten Vorgehen die Objekte des Schachs und ihre Funktionen repräsentieren
  - Class Board: Repräsentiert das Schachbrett an sich und beinhaltet Methoden die den Spielverlauf auf dem Brett darstellen und steuern
  - Class Square: Stellt die einzelnen Felder auf dem Brett da, diese können verschiedene Werte aufweisen, bspw. ob eine Spielfigur auf dem Feld steht
  - Interface Piece: Bildet ein Interface, welches die einzelnen unterschiedlichen Figuren zusammenfasst und gemeinsame Funktion und Verhalten bündelt
  - Class King, Queen, Bishop, Knight, Rook, Pawn: Sind die konkreten einzelnen unterschiedliche Figuren
- View: hier werden in Zukunft die verschiedenen GUI Funktionalitäten implementiert