

1 Ausgezeichneter Programmorganisationsplan

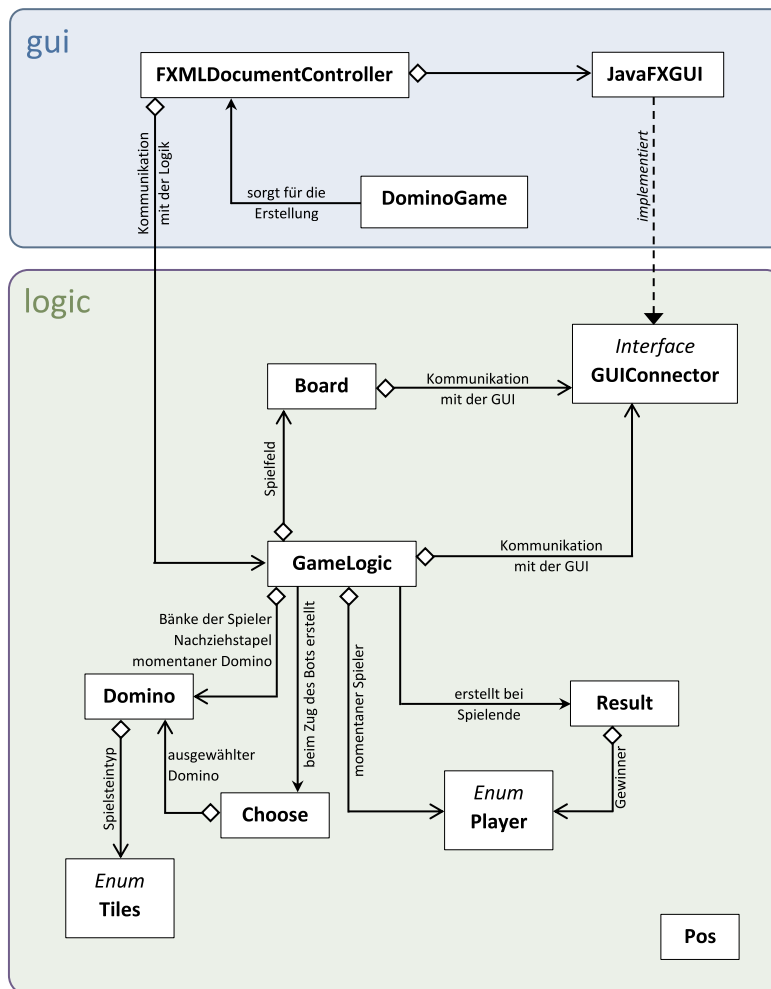


Abbildung 1: Händisch erstellter POP

1.1 Anmerkungen

Die farbliche Kennzeichnung ermöglicht hier eine einfache Zuordnung der Klassen zu den Paketen. Durch die Beschriftung der Pfeile lässt sich schnell nachvollziehen, in welchem Zusammenhang die Klassen stehen. Trotzdem ist hier eine Erläuterung angebracht, wie der Plan zu lesen ist, bzw. nach welchen Kriterien Pfeile eingezeichnet worden sind (die Klasse *JavaFX* arbeitet z.B. durchaus mit *Result*). In diesem Beispiel sollte zudem kurz auf die Sonderrolle der Klasse *Pos* eingegangen werden.

2 Ausreichender Programmorganisationsplan

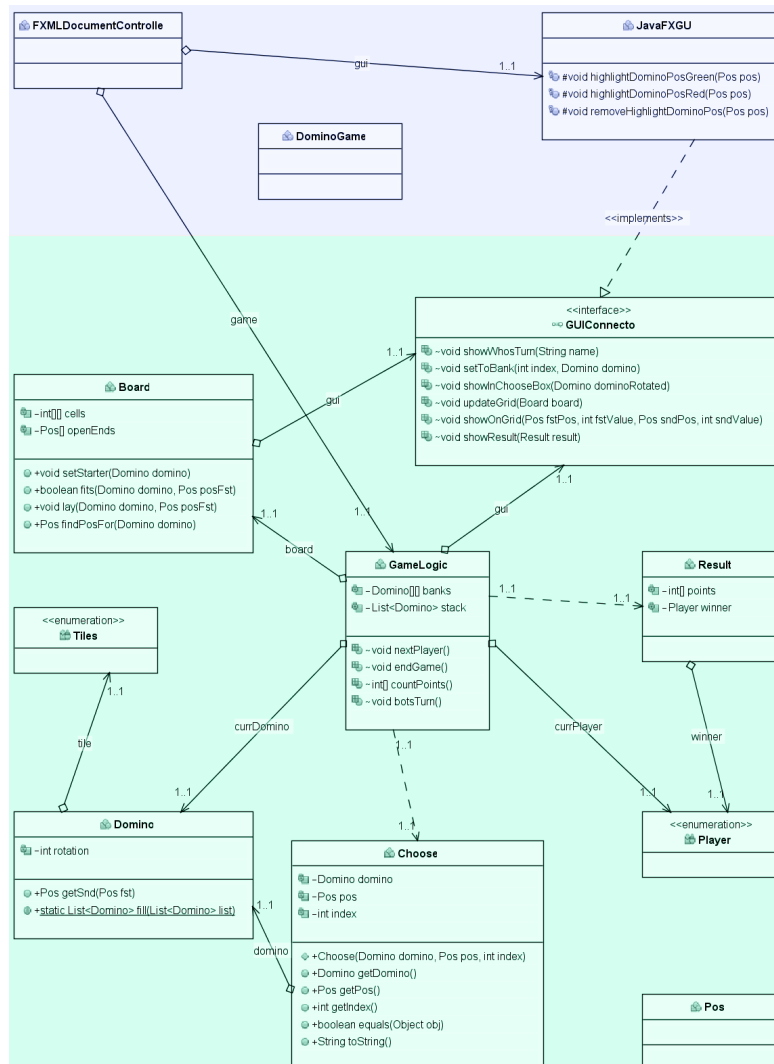


Abbildung 2: Automatisch erzeugter, nachbearbeiteter POP

2.1 Anmerkungen

Auch hier ist die Paketzugehörigkeit durch die Hintergrundfarbe zu erkennen. Allerdings sind die Zusammenhänge zwischen den Klassen durch die fehlende Beschriftung der Pfeile nicht direkt erkennbar. Die wichtigsten Attribute und Methoden der jeweiligen Klassen hätten auch weggelassen werden können: diese Information sollte bei der Beschreibung grundlegender Klassen zu finden sein. Insgesamt ist der Plan nicht optimal und gibt nicht die volle Punktzahl.

3 Inakzeptabler Programmorganisationsplan

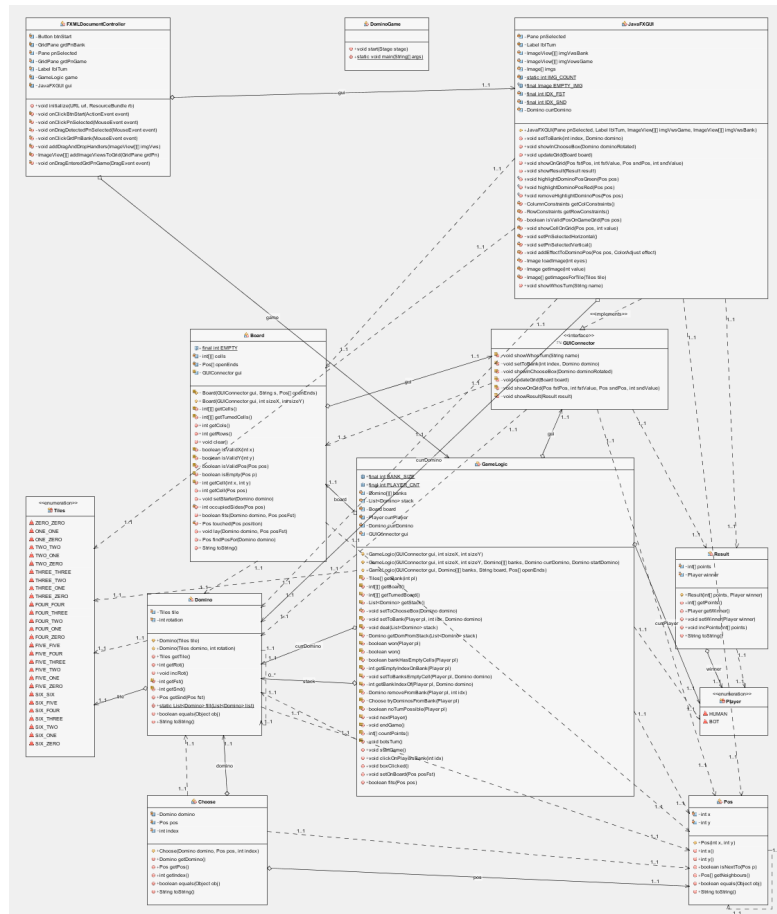


Abbildung 3: Automatisch erstellter POP (mit easyUML¹)

3.1 Anmerkungen

Totaler Quark. Wer einen solchen Plan abgibt, wird gekiehl².

Durch die Vielzahl von Pfeilen absolut nicht lesbar. Es gibt keine vernünftige Begründung, alle Methoden und Attribute einer jeden Klasse in den Programmorganisationsplan mit aufzunehmen. Der Programmorgansationsplan sollte einen Mehrwert bieten und nicht Kopfschmerzen bereiten.

¹siehe <http://plugins.netbeans.org/plugin/55435/easyuml>

²Kiel holen: „eine schwere, in der Seefahrt bis ins 19. Jahrhundert gebräuchliche Disziplinarstrafe, bei der der Bestrafte an einem Tau unter dem Rumpf des Schiffs durchgezogen wurde“ (laut <https://de.wikipedia.org/wiki/Kielholen>)