

## Reflexion Teilaufgabe 2:

Eine Änderung im Vergleich zum zuvor erstellten Klassendiagramm ist das Hinzufügen einer Klasse, welche das Model darstellt und welche nicht FullMap ist, somit wurde sozusagen eine Klasse „dazwischengeschaltet“, welche dann den Zugriff auf die Daten der FullMap auch ermöglicht, jedoch in einem kontrolliertem Umfeld. Zuerst wollte ich, wie in der Vorlesung hingewiesen, für die treasure finding bzw. castle finding Strategie eben ein Strategy-Pattern einsetzen, weil es hier noch am ehesten Sinn macht da es eine langfristige Änderung am Verhalten der KI bewirkt, jedoch hat mir dann eine Variable **target** welche dann wenn es soweit ist, zum *castle finding* umschaltet. Der Grund ist rein die Komplexität bzw. das Verständnis des Codes (zu viel Abstraktion). Zusätzlich habe ich noch eine Klasse als Converter für Netzwerkdaten erstellt da sonst die Netzwerkklassse einfach zu voll gewesen wäre und als eine zusätzliche Schicht zwischen Server und Client wirkt.

### Y-Statement 1:

- Im Kontext von der Model-Klasse FullMap habe ich mir auf Grund von Entkopplung für eine weitere Klasse entschieden und eine Lösung mit einer größeren, alleinstehenden Klasse vernachlässigt, um das Prinzip von single-responsability zu erreichen und auch eine gutes Maß an Entkopplung der Klasse FullMap zu erreichen, unter Inkaufnahme von ein wenig Mehraufwand, weil es hier eine Lösung mit wesentlich mehr Überblick bietet und die Qualität des Codes anhebt.

### Y-Statement 2:

- Im Kontext vom Wechsel zwischen treasure und castle finding habe ich mich angesichts von Komplexitätsgründen für eine Variable im Zusammenhang mit einer Abfrage in einer Methode entschieden und ein Strategy-Pattern vernachlässigt, um den Überblick zu behalten und nicht ein zu großes Maß an Abstraktion in den Code einzubringen, unter Inkaufnahme von Mangel an Erweiterbarkeit, weil hiermit die Lesbarkeit des Codes eher abnimmt und die Qualität im Allgemeinen eher dazu tendiert wenig zuzunehmen.

### Y-Statement 3:

- Im Kontext von der Netzwerkklassse habe ich mich angesichts von Unübersichtlichkeit und single-responsability für einen zusätzlichen Converter entschieden und eine größere Klasse, welche alle Umwandlungen der Serverobjekte selbst erledigt vernachlässigt, um den Client durch diese Schicht hindurch mit dem Server kommunizieren zu lassen, unter Inkaufnahme von Mehraufwand und einer neuen Klasse, weil somit die Qualität und Lesbarkeit des Codes angehoben werden.