



## Aufgabe 9.1 – Software-Produktlinie

In welche beiden Arbeitsbereiche unterteilt sich die Entwicklung einer Software-Produktlinie und was wird in diesen beiden Arbeitsbereichen getan?

Man trennt die Software-Produktlinie in zwei Bereichen welche abhängig von einander entwickelt werden.

Die erste Phase wird als *Domänen Engineering* bezeichnet. Man erfasst alle relevanten Konzepte der Domäne, aus der man die Realisierung einer wieder verwendbaren Plattform produziert (man muss die Gemeinsamkeiten als auch die Unterschiede der einzelnen Produkte einbauen). Durch die Wiederverwendung der Komponenten spart man bei den Produktionskosten und trägt zu einer besseren Qualität der Komponenten bei.

Um die Endprodukte zugeschnitten auf die Bedürfnisse der einzelnen Kunden zu entwickeln, werden in dem zweiten Bereich, das *Applikation Engineering*, die wieder verwendbaren Komponenten der Plattform ausgewählt und miteinander integriert. Dabei wird mit dem Domänen Engineering insoweit zusammengearbeitet, als dass wieder verwendbare Komponenten bestimmt werden.

## Aufgabe 9.2 – Testen

Sie bekommen einen Taschenrechner geschenkt. Eine Funktion darin überprüft bei Eingabe einer natürlichen Zahl, ob die Zahl gerade oder ungerade ist. Welche aus der Vorlesung bekannte Testmethode können Sie einsetzen um die Testfälle zu spezifizieren? Wie gehen Sie dabei vor? Geben Sie eine Liste von repräsentativen Eingabemengen an, die voraussichtlich das vollständige Verhalten der Funktion für alle möglichen Eingaben prüft. Geben Sie außerdem einen zu prüfenden konkreten Wert für jede repräsentative Eingabemenge an.

Ohne ein zu überprüfendes Programm, muss man die Testfälle aus der Spezifikation ableiten.

Dabei wird man die Testmethode der Blackbox benutzen. Man kann das Verhalten der Funktion durch folgende Mengen testen:

- Nur gerade Zahlen  $\{2, 4, 6, \dots\}$
- Nur ungerade Zahlen  $\{1, 3, 5, \dots\}$
- Mischung von natürlichen Zahlen  $\{1, 2, 3, \dots\}$
- Wir überprüfen 0 als Spezialfall, bezüglich ob wir es in der Menge der natürlichen Zahlen einbeziehen werden

Die Funktion erwartet eine natürliche Zahl, dabei sollen wir vor der Überprüfung der Parität erstmal feststellen ob die Eingabe eine gültige Eingabe ist.

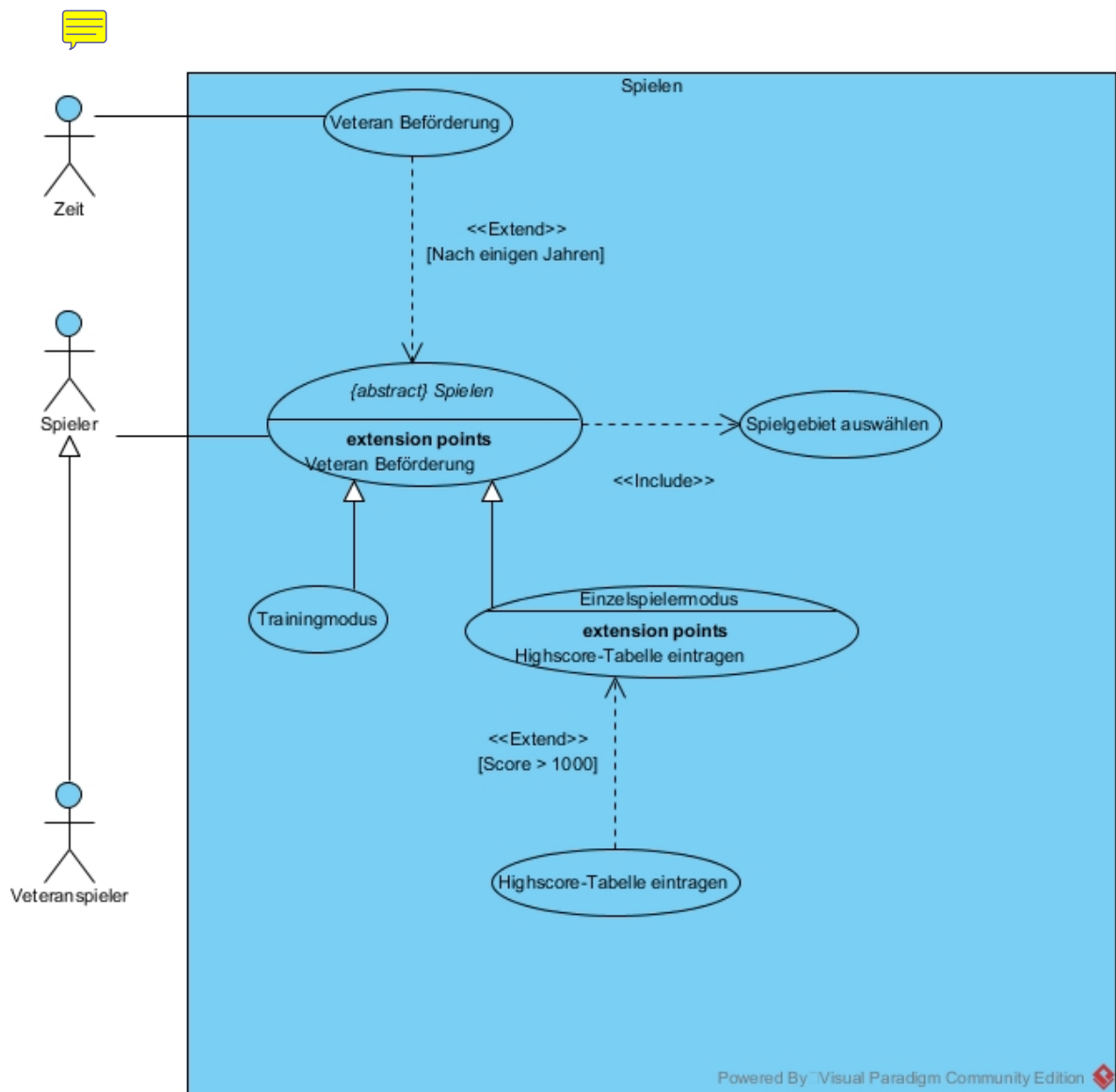
z.B. Zeichenstrings, keine negativen Zahlen, keine Zahlenwerte mit Nachkommastellen, u.a.

Die konkreten Werte die man zum Testen verwenden kann sind: **-101, -100, 0, 100, 101, -0.5, 0.6** und ein beliebiges String das man vielleicht mit den Tasten des Taschenrechners eingeben kann z.B. „(“ ;

## Aufgabe 9.3 – Use-Case-Diagramm

Sie befinden sich gegenwärtig in der Anforderungsanalyse des Computerspiels Mega Maria Sisters. Verschaffen Sie sich einen ersten strukturierten Überblick über den Funktionsumfang des Spiels, indem Sie ein geeignetes Use-Case-Diagramm auf Grundlage der folgenden textuellen Beschreibung erstellen.

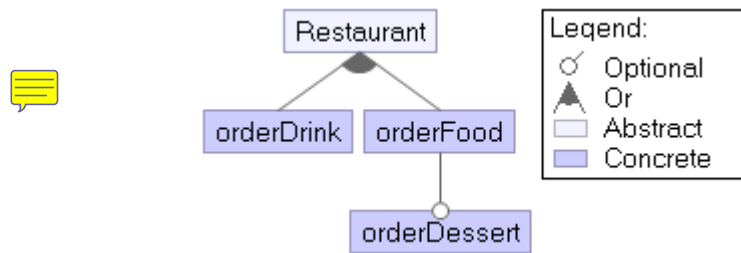
Die zentrale Funktion von Mega Maria Sisters ist das Spielen. Ein Spieler kann entweder im Einzelspielermodus oder im Trainingsmodus spielen. Vor dem Spielen muss ein Spielgebiet ausgewählt werden. Nach dem Spielen im Einzelspielermodus darf sich der Spieler in der Highscore-Tabelle eintragen, falls sein Score > 1000 ist. Nach einigen Jahren Spielerfahrung wird ein Spieler zum Veteranspieler.



## Aufgabe 9.4 – Feature- und Sequenzdiagramme

a) Modellieren Sie in einem Feature Diagramm die folgende Situation in einem Restaurant. Geben Sie noch dazu alle erlaubten Konfigurationen explizit an.

*Um sich in einem Restaurant setzen zu dürfen, muss man etwas zum Essen oder etwas zum Trinken(oder beides) bestellen. Beim Essen kann man sich optional noch einen Nachtisch holen.*



Mit den folgenden Konfigurationen: {Restaurant, orderDrink}, { Restaurant, orderFood}, {Restaurant, orderDrink, orderFood}, {Restaurant, orderDrink, orderFood, orderDessert};

b) Modellieren Sie in einem UML-Sequenzdiagramm folgende Interaktionen bei einem Restaurantbesuch:

*Der Kellner bringt dem Besucher die Karte und wartet eine bestimmte Zeit lang, während der Besucher auswählt. Anschließend fragt der Kellner den Besucher nach seiner Bestellung. Dieser antwortet, indem er eine Bestellung macht. Der Kellner leitet die Bestellung weiter an den Koch und serviert dem Besucher das Getränk seiner Wahl. Später teilt der Koch dem Kellner mit, dass die Bestellung des Besuchers fertig ist. Daraufhin serviert der Kellner dem Besucher seine Bestellung. Irgendwann später teilt der Besucher dem Kellner mit, dass er zahlen möchte. Der Kellner antwortet, indem er dem Kunden einen Kassenbon ringt. Der Kunde zahlt den Betrag und bekommt vom Kellner das Wechselgeld.*

