

# Übung 5: Domänenmodellierung



TECHNISCHE  
UNIVERSITÄT  
DARMSTADT

---

Software Engineering  
WS 2018/19 - Dr. Michael Eichberg

---

## Abgabe

---

Die Übung wird als *sbt*-Projekt (<http://www.scala-sbt.org>) bereitgestellt. *sbt* kann verwendet werden, um Java (und Scala) Projekte einfach auszuführen und zu testen. Installieren Sie *sbt* auf Ihrem Rechner und stellen Sie sicher, dass ein Java SDK (min. Java 8) installiert ist. Überprüfen Sie, dass der Java Compiler *javac* auf der Kommandozeile ausführbar ist, auch wenn Sie sich nicht im Verzeichnis mit der ausführbaren Datei befinden. Wenn nicht, passen Sie Ihren *PATH* entsprechend an, Hinweise wie dies bei Ihrem Betriebssystem möglich ist, finden Sie im Internet.

IntelliJ erlaubt das Importieren von *sbt*-Projekten direkt, dafür muss allerdings das Scala Plugin installiert sein. Um ein Eclipse-Projekt zu erzeugen, kann *sbt eclipse* im Projektverzeichnis (das Verzeichnis, das die Datei *build.sbt* enthält) ausgeführt werden, danach kann das Projekt mit Datei > Importieren > Vorhandene Projekte in Arbeitsbereich importiert werden. Wenn Sie eine *main*-Methode geschrieben haben, können Sie Ihr Programm mit *sbt run* ausführen, Tests können Sie unter *src/test/java/* anlegen und mit *sbt test* ausführen.

Das Ausführen des Kommandos *sbt* im Projektverzeichnis startet den interaktiven Modus von *sbt*. Im interaktiven Modus können Kommandos ausgeführt werden, ohne jedes Mal *sbt* eingeben zu müssen.

Um Ihre Lösung abzugeben, melden Sie sich zunächst unter

<https://submission.st.informatik.tu-darmstadt.de/course/se18>

an und erzeugen Sie ein *submission token*. Wenn Sie den interaktiven Modus von *sbt* verwenden, führen Sie dann folgendes Kommando aus:

```
submit <ihreTUID> <submissionToken>
```

wobei *<ihreTUID>* Ihre eindeutige Identifikationsnummer an der TU Darmstadt (**nicht Ihre Matrikelnummer!**) und *<submissionToken>* das zuvor generierte Token ist. Geben Sie die spitzen Klammern nicht mit an, der Befehl sollte etwa so aussehen: `submit ab01cdef 01234567`. Wenn Sie nicht den interaktiven Modus verwenden, muss das Kommando in Anführungszeichen gesetzt werden, also *sbt "submit <ihreTUID> <submissionToken>"*. Sie können (innerhalb der Abgabefrist) beliebig oft eine Lösung einreichen, allerdings wird nur die zuletzt eingereichte Lösung bewertet. Die letzte Lösung, die ein Gruppenmitglied eingereicht hat, wird zur Bewertung für die ganze Gruppe herangezogen. Koordinieren Sie sich daher in Ihrer Gruppe, wer Ihre gemeinsame Lösung einreicht.

Stellen Sie sicher, dass Sie das zur Verfügung gestellte Template nutzen und die Namen und Signaturen der vorgegeben Klassen und Methoden nicht verändern sowie dass von Ihnen hinzugefügte Klassen und Methoden die geforderten Namen und Signaturen verwenden. Ändern Sie außerdem nichts an der vorgegebenen Datei *build.sbt*. Andernfalls wird das System Ihre Lösung nicht bewerten können. Beachten Sie, dass der Zugriff auf die Abgabepattform nur im **internen Netz der Universität** möglich ist. Für einen Zugriff von außerhalb benötigen Sie daher eine VPN-Verbindung.

---

## Einführung

In dieser Übung befassen Sie sich mit der Domänenmodellierung. Bearbeiten Sie die folgenden drei Aufgaben und erstellen Sie eine einzelne PDF-Datei mit Ihren Antworten. Legen Sie diese Datei in den Ordner *solution* in dem Template, das Ihnen zur Verfügung gestellt wurde, und verwenden Sie zur Abgabe wie oben angegeben *sbt submit*.

---

**Problem 1 Konzepte identifizieren****4P**

Unten finden Sie eine Beschreibung zu einem Geoinformationssystem (GIS). Identifizieren und nennen Sie alle relevanten Konzepte.

---

**Problem 2 Konzepte modellieren****8P**

Erstellen Sie auf Grundlage der in Problem 1 identifizierten Konzepte ein UML Diagramm. Berücksichtigen Sie notwendige Klassen und ihre Beziehungen einschließlich Multiplizitäten. Geben Sie auch die wesentlichen Attribute der Klassen an.

---

**Problem 3 Konzepte implementieren****4P**

Implementieren Sie die in Problem 1 identifizierten Konzepte als Klassen im Package `ex05` des zur Verfügung gestellten Templates. Annotieren Sie jede Klasse und jedes Feld mit der Annotation `@Concept`, mit der Sie den *exakten* Begriff, den die Beschreibung für das von dieser Klasse modellierte Konzept verwendet hat, angeben. Z.B.

```
@Concept("Auto")
public class Car {
    @Concept("Hubraum")
    int engineDisplacement;
}
```

für die Modellierung eines Autos mit einem gewissen Hubraum.

---

**Beschreibung Geoinformationssystem**

Ein Geoinformationssystem ist ein Computersystem, mit dem geografische Daten erfasst werden, um daraus Kartendarstellungen zu erzeugen, Routen zu berechnen oder andere Geoinformationen bspw. zur Flächennutzung abzuleiten. Es verwaltet Geoobjekte wie Punkte, Wege und Flächen. Ein Punkt ist durch zwei Koordinaten gegeben, den Längengrad und den Breitengrad. Ein Weg besteht aus einer geordneten Menge von Punkten. Kein Weg kann nur einen Punkt enthalten. Eine Fläche besteht aus einer nichtleeren Menge von Wegen. Punkte können Teil mehrerer Wege sein, Wege Teil mehrerer Flächen. Außerdem ist jedem Geoobjekt eine eindeutige Identifikationsnummer zugeordnet sowie eine beliebige Anzahl von Eigenschaften. Die Eigenschaften sind eine Zuordnung von eindeutigen Schlüsseln zu Werten, die beide Zeichenketten sind. Beispielsweise könnten die Eigenschaften eines Wegs {"Name" -> "Hauptstraße", "Höchstgeschwindigkeit" -> "50km/h"} sein. Das Geoinformationssystem wird über eine Website benutzt, auf der die enthaltenen Informationen in verschiedenen Kartendarstellungen angezeigt werden. Das Geoinformationssystem kann auch zur Routenplanung verwendet werden. Dafür wird eine Anfrage gestellt, die zwei Punkte, den Startpunkt und den Zielpunkt beinhaltet. Als Antwort erhält man eine Route, die aus Wegen besteht. Außerdem hat eine Route eine Länge und eine Fahrzeit, die von der Routenplanung geschätzt wird. Benutzer können Geoobjekte mit einem Editor verändern, wobei zu jedem Geoobjekt der Bearbeiter und der Bearbeitungszeitpunkt der letzten Bearbeitung (bzw. der erstmaligen Erstellung) erfasst werden müssen. Benutzer werden über einen Benutzernamen und eine IP identifiziert. Ein Benutzer kann auch einen Punkt als seinen Heimatort angeben. Wenn Benutzer wiederholt falsche Änderungen vorgenommen haben, werden sie dauerhaft gesperrt, was bedeutet, dass sie keine Änderungen mehr vornehmen können. Um Benutzer bei Fehlverhalten zu sperren, existieren Moderatoren. Bei einem Moderator handelt es sich um einen Benutzer mit entsprechenden Sonderrechten, der als Zuständigkeitsbereich eine ihm zugewiesene Fläche hat.