

# Übungen zur Vorlesung Softwaretechnologie

- Wintersemester 2010/11 -  
Dr. Günter Kniessel

## Übungsblatt 5 - Lösungshilfe

### Aufgabe 1. Domain Object Modell (12 Punkte)

Stellen Sie sich vor, Sie sollen für die Informatik-Fachschaft ein Programm zur Tutoriums-Wahl entwerfen, das sich Ihre Auftraggeber wie folgt vorstellen:

Das Programm besteht aus einem Eingabe-Fenster, nämlich der Veranstaltungs-Auswahl und zwei Ausgabe-Fenstern, der Stundenplan-Ansicht und einer Hilfeseite.

Zuerst wählen die Studierenden in der Veranstaltungs-Auswahl ihre Veranstaltungen aus. Veranstaltungen sind Vorlesungen oder Seminare. Die Veranstaltungen haben als Veranstaltungsdaten einen Titel, eine Uhrzeit und eine Raumangabe, Seminare haben zusätzlich eine Maximal-Teilnehmerzahl.

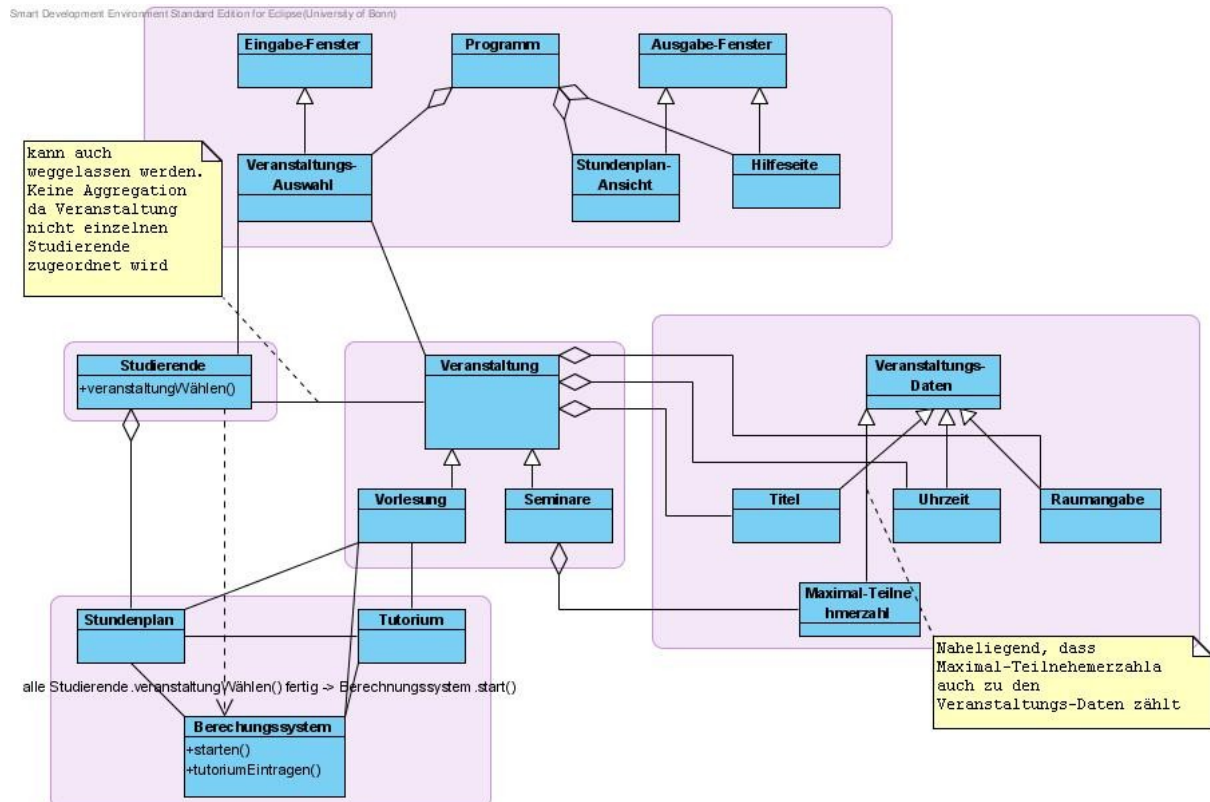
Nachdem alle Studierenden ihre Veranstaltungen ausgewählt haben, muss das Berechnungssystem starten, das in die Stundenpläne der Studierenden für jede Vorlesung ein zugehöriges Tutorium einträgt.

- a) Analysieren Sie obige Beschreibung mit Hilfe von Abbotts Technik und stellen sie die Ergebnisse tabellarisch dar. Hinweis: Geben Sie Ihre Lösung als PDF-Dokument ab.

Aussage	Wort	Wortart	Modellelement
Das Programm besteht aus einem Eingabe-Fenster, nämlich der Veranstaltungs-Auswahl und zwei Ausgabe-Fenstern, der Stundenplan-Ansicht und einer Hilfeseite.	Programm, Eingabe-Fenster, Veranstaltungs-Auswahl, Ausgabe-Fenster, Stundenplan-Ansicht, Hilfeseite	Nomen	Klassen
	... besteht aus ...	Verb (haben)	Aggregation
	Eingabe-Fenster [...] Veranstaltungsauswahl, [...] Ausgabe-Fenster [...] Stundenplan-Anzeige [...] Hilfeseite	(verstecktes ) Verb (sein)	Vererbung
Zuerst wählen die Studierenden in der Veranstaltungs-Auswahl ihre Veranstaltungen aus.	Studierenden, Veranstaltungen	Nomen	Klassen
	wählen	Verb	Methode
	Die Studierenden ... [wählen] in der Veranstaltungsauswahl ... aus, in der Veranstaltungs-Auswahl ... Veranstaltungen		
Veranstaltungen sind Vorlesungen oder Seminare.	Vorlesungen, Seminare	Nomen	Klassen
	Sind	Verb (sein)	Vererbung

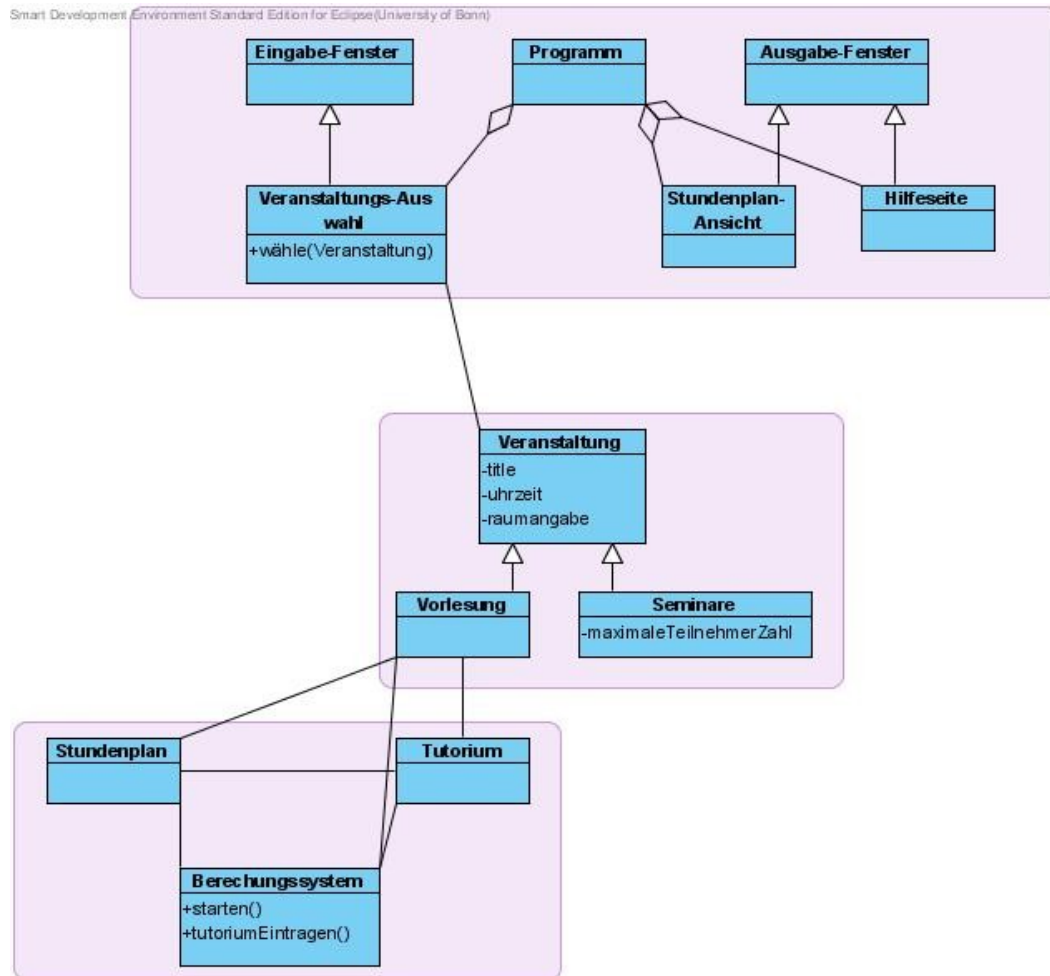
Die Veranstaltungen haben als Veranstaltungsdaten einen Titel, eine Uhrzeit und eine Raumangabe, Seminare haben zusätzlich eine Maximal-Teilnehmerzahl.	Veranstaltungsdaten, Titel, Uhrzeit, Raumangabe, Maximal-Teilnehmerzahl	Nomen	Klassen
	Haben ...	Verb (haben)	Aggregation
	... als ...	(verstecktes ) Verb (sein)	Vererbung
Nachdem alle Studierenden ihre Veranstaltungen ausgewählt haben, muss das Berechnungssystem starten, das in die Stundenpläne der Studierenden für jede Vorlesung ein zugehöriges Tutorium einträgt	Berechnungssystem, Stundenpläne, Tutorium	Nomen	Klassen
	... muss das Berechnungssystem ...	Einschränkung	Constraint
	... starten ..., ... einträgt	Verben	Methoden
	Studierende [tragen] ... Veranstaltungen [ein], [das Berechnungssystem trägt] in die Stundenpläne ... ein, ... Vorlesung zugehöriges Tutorium ... das Berechnungssystem [trägt] ... Tutorien ein		

b) Setzen Sie die Ergebnisse der textuellen Analyse aus Aufgabe a) zunächst schematisch (1-zu-1) in ein Domain Object Model (UML-Klassendiagramm mit Attributen und Assoziationen zwischen den Klassen) um.



c) Überdenken Sie das Domain Object Model und fassen die Ergebnisse aus Aufgabe b) zusammen.

- Klassen zu Attributen konvertieren, wenn Klasse kein eigenes Verhalten besitzt  
Klasse als einzige Information ihre Klassenzugehörigkeit liefert
- Studierender sind eigentlich nicht Teil des Programms sondern Akteure



## Aufgabe 2. Analysemodell (3 Punkte)

Im Rahmen der Anforderungserhebung für das Drucken einer E-Mail mit angehängter Stundenplan-Datei wurden folgende Objekte identifiziert:

- Stundenplan-Datei
- Druckdialog
- Drucke-Modul
- E-Mail-Programm
- Programm-Fenster
- E-Mail

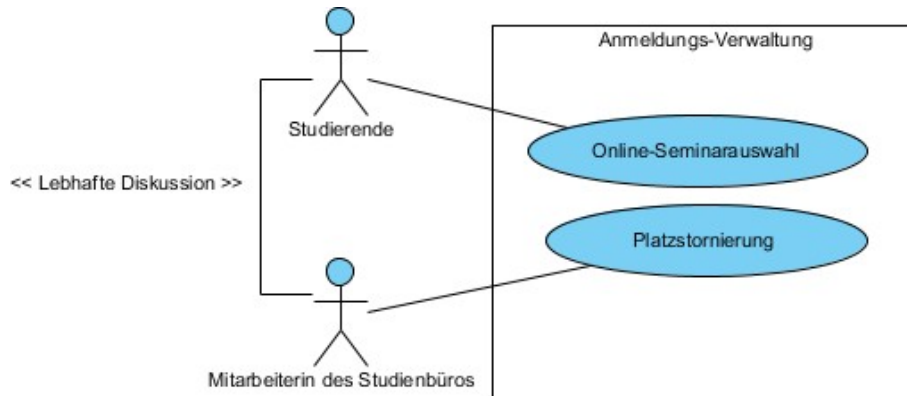
Klassifizieren Sie die Elemente als Boundary-, Controller- und Entity – Objekte.

- ◆ **Boundary**
  - ⇒ Druckdialog
  - ⇒ Programm-Fenster

- ◆ Controller
  - ⇒ E-Mail Programm
  - ⇒ Drucke-Modul
- ◆ Entity
  - ⇒ Stundenplan-Datei
  - ⇒ E-Mail

### Aufgabe 3. Analysemodell (10 Punkte)

Sie sind mit dem Design eines Systems zur Seminaerauswahl im Studienbüro beauftragt. Als Vorgabe erhalten Sie die zwei unten eingezeichneten Anwendungsfälle:



Die verbale und nonverbale Kommunikation zwischen Studierender und Mitarbeiterin ist hier nur durch die Beziehung <<Lebhaftes Diskussion>> angedeutet. Sie wird nicht weiter modelliert, denn sie läuft nicht über das System.

Zu den Anwendungsfällen gibt es die folgenden Beschreibungen:

<p><b>Name des Anwendungsfalls:</b> Online-Seminarplatzwahl</p> <p><b>Akteure:</b> Studierende</p> <p><b>Anfangsbedingung:</b> Dieser Anwendungsfall beginnt, wenn die Studierende die URL der Online-Seminarwahl aufruft.</p> <p><b>Ereignisfluss:</b></p> <ol style="list-style-type: none"><li>1. Studierende sucht Seminar im System.</li><li>2. Studierende wählt ein Seminar aus.</li><li>3. System zeigt Verfügbarkeit von Plätzen für das ausgewählte Seminar an und öffnet eine Eingabemaske für die Anmelde-Daten.</li><li>4. Studierende gibt Studiengang, Semesterzahl und Matrikelnummer an.</li><li>5. System reserviert Platz.</li><li>6. System druckt Bestätigung aus</li></ol> <p><b>Endbedingung:</b> Studierende muss ihre Reservierungs-Bestätigung mitnehmen um sie vorzeigen zu können.</p> <p><b>Sonderfälle:</b> keine.</p> <p><b>Spezielle Anforderungen:</b> Studierender steht Drucker zur Verfügung.</p>
---

**Name des Anwendungsfalls:** Platzstornierung

**Akteure:** Studierende, Mitarbeiterin des Studienbüros

**Anfangsbedingung:**

Dieser Anwendungsfall beginnt, wenn die Studierende das Studienbüro betritt.  
Studierende besitzt bereits eine Bestätigung für ein Seminar.

**Ereignisfluss:**

1. Mitarbeiterin nimmt die Reservierungs-Bestätigung der Studierenden entgegen.
2. Mitarbeiterin sucht Seminar im Stornierungs-System.
3. Mitarbeiterin wählt den Platz der Studierenden zur Freigabe aus
4. Mitarbeiterin bestätigt Platz-Freigabe
5. System gibt reservierten Platz wieder frei.
6. Mitarbeiterin vernichtet die Reservierungs-Bestätigung.

**Endbedingung:**

Reservierungs-Bestätigung ist vernichtet.

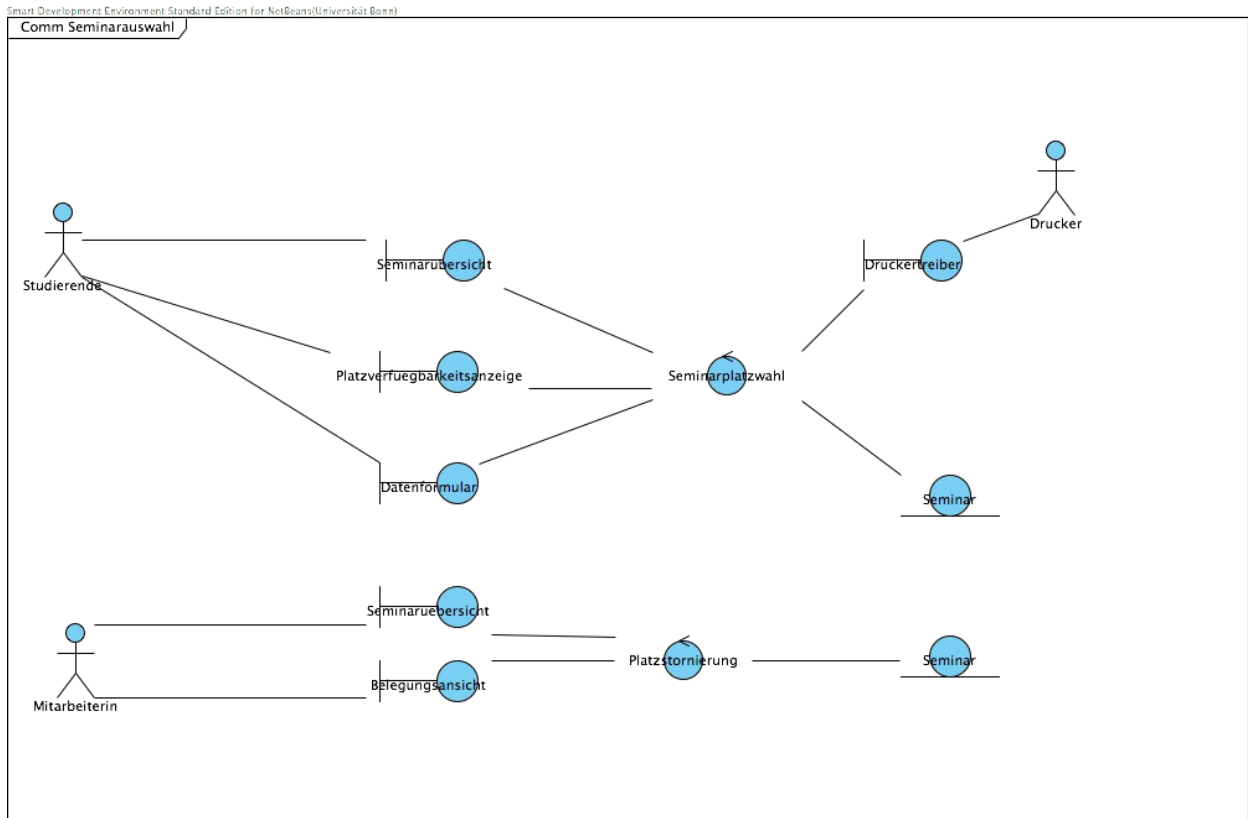
**Sonderfälle:** keine.

**Spezielle Anforderungen:** keine.

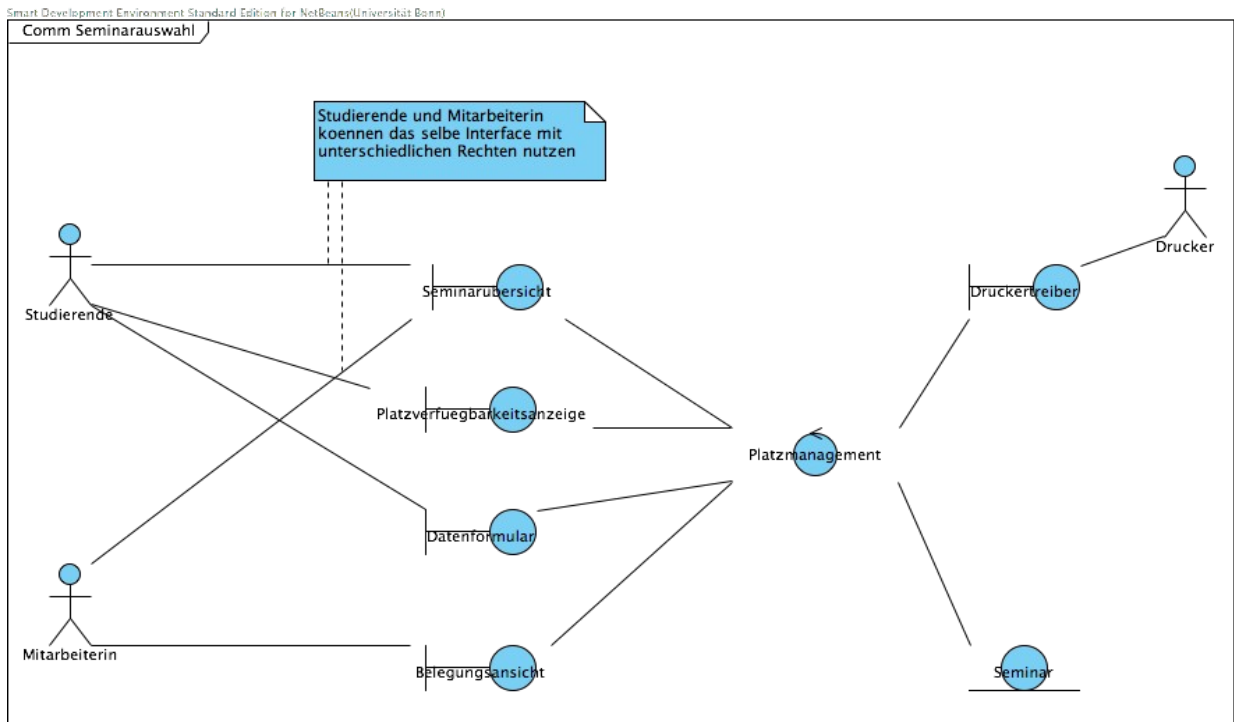
a) Identifizieren Sie für jeden der drei Anwendungsfälle jeweils Boundary-, Controller- und Entity-Objekte sowie die beteiligten Akteure. Es reicht, sie pro Anwendungsfall textuell aufzuzählen.

- **Online-Seminarplatzwahl**
  - Boundary
    - Seminarübersicht
    - Platzverfügbarkeits-Anzeige
    - Eingabemaske für Anmeldeinformationen
  - Controller
    - Seminarplatzwahl
  - Entity
    - Seminar
  - Akteure
    - Studierende
    - Drucker
- **Platzstornierung**
  - Boundary
    - Seminarübersicht
    - Belegungsansicht
  - Controller
    - Platzstornierung
  - Entity
    - Seminar
  - Akteure
    - Studierende
    - Mitarbeiterin

b) Setzen Sie die Ergebnisse aus Aufgabenteil a) in einem Kommunikationsdiagramm um. Nachrichten sind in diesem Schritt nicht erforderlich.



c) Erstellen Sie ein neues Diagramm in dem Sie die Analyseobjekte aus Teilaufgabe b), die ähnliches tun bzw. konzeptuell identisch sind, zusammenfassen.

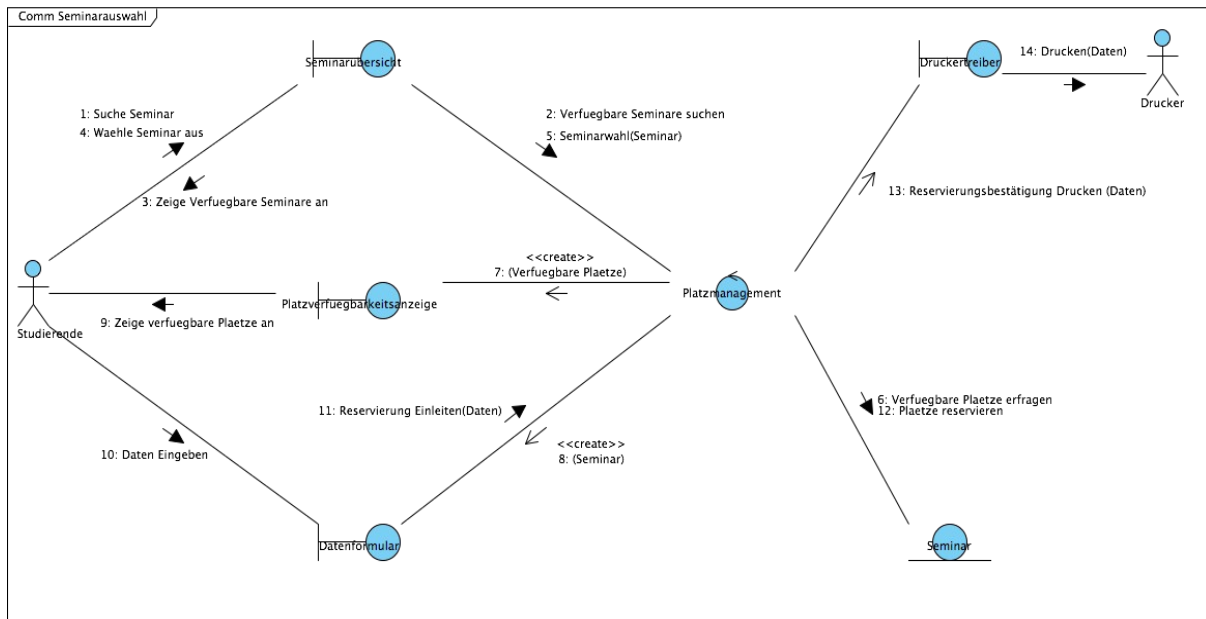


**Hinweis:** Wenn Ihnen die Vorgehensweise nicht klar ist, sehen Sie sich die Folien zur Objektmodellierung im Analyseworkflow aus der Vorlesung an.

**Aufgabe 4. Kommunikationsdiagramme (5 Punkte)**

Extrahieren Sie für den Anwendungsfall **Online-Seminarauswahl** die Nachrichten des Ereignisflusses und notieren Sie sie in einem zugehörigen Kommunikationsdiagramm. Nutzen Sie als Grundlage das Ergebnis der Aufgabe 3c.

Smart Developers Environment Standard Edition for NetBeans (Universität Bonn)



Anmerkungen:

- Alle Interaktionen zwischen einem Akteur und einem Boundary finden außerhalb des Systems statt und sind insofern informelle Beschreibungen von Tätigkeiten die der Akteur mit dem Boundary ausführt (z.B. „Suche Seminar“), bzw. das Boundary für den Akteur ausführt (z.B. „Zeige verfügbare Seminare an“). Es sind keine konkreten Nachrichten die an ein Objekt im System geschickt werden.
- Die Aktion „1: Suche Seminar“ ist zu interpretieren als „Der Studierende wählt die ‚Suche Seminar‘- Aktion auf der Benutzeroberfläche aus“. Wie das geschieht wird hier nicht modelliert (es kann z.B. durch klicken eines Links, Buttons, Menueeintrags, etc. oder Berühren eines bestimmten Elements auf einem Touchscreen realisiert sein). Ebenso wird der Rest der Benutzeroberfläche, die die Auswahl der „Suche Seminar“-Aktion zur Verfügung stellt nicht modelliert.
- Wenn man das Boundary in dem das „Suche Seminar“ angeklickt werden kann explizit modellieren würde, sollte man die „Suche Seminar“ Interaktion daran richten und dann als erste Unteraktion eine <<create>>-Nachricht einfügen, die klar macht, dass das als Reaktion das Boundary „Seminarübersicht“ erzeugt (und angezeigt) wird.
- Bezogen auf die Erzeugungsnachrichten fehlt im obigen Diagramm eine, die klar macht, dass der Controller „Platz-Management“ von dem Boundary „Seminarübersicht“ erzeugt wird (bevor es die „verfügbare Seminare Suchen“ Nachricht an den Controller schickt).