

---

# Softwareentwicklungsprojekt 2015

## – Aufgabenblatt 2 –

*spätester Abgabetermin: 15. Mai 2015 16 Uhr*

---

*Dieses Blatt richtet sich ausschließlich an die Teilnehmer des **Softwareentwicklungsprojekts**.*

### Ziel des Blattes

Auf dem letzten Aufgabenblatt wurde ein Pflichtenheft erstellt. Aufbauend auf diesen Ergebnissen wird nun die Systemanalyse durchgeführt. Die Schnittstellen (Methoden) werden in diesem Fall durch exploratives Prototyping mit Hilfe von Aktivitätsdiagrammen bestimmt. Diese Diagramme sollen dazu dienen, die nötigen Methoden ausfindig zu machen und um zu zeigen, dass diese prinzipiell in der Lage sind, die gestellten Aufgaben erfüllen zu können.

Dann wird eine Grobstruktur mittels eines Paketdiagramms festgelegt. Danach wird dieses durch ein Klassendiagramm verfeinert. Eine Aufteilung in Client und Server findet hier noch nicht statt. Dies wird erst Aufgabe für den Entwurf sein.

Beim Modellieren soll auf eine starke Bindung und lose Kopplung geachtet werden und jeder Klasse eine klar definierte Aufgabe gegeben werden (Single Responsibility Principle).

### Aufgaben

1. Überlegen Sie sich die Workflows, die beim Benutzen der Software auftreten und modellieren Sie diese entsprechend durch Aktivitätsdiagramme. Achten Sie darauf, dass jede Produktfunktion Ihres Pflichtenheftes in mindestens einem Workflow vorhanden ist.
2. Erstellen Sie, auf Basis Ihrer Aktivitätsdiagramme, ein Paketdiagramm, das die Grobstruktur des Systems veranschaulicht. Ignorieren Sie dabei jegliche Verteilungsaspekte (also die Unterscheidung zwischen Client und Server). Geben Sie zu jedem Paket jeweils dessen Zweck an.
3. Erstellen Sie ein statisches Analysemodell mittels eines oder mehrerer Klassendiagramme. Erläutern Sie das, was nicht offensichtlich aus dem Diagramm hervorgeht. Stellen Sie sicher, dass sich jeder der Workflows aus Aufgabe 1 darin widerspiegelt.
4. Wenn Sie sich zwischen mehreren alternativen Lösungswegen entscheiden, dokumentieren Sie jeweils Ihre Gründe.

## Hinweise

1. Wenn Ihr Modellierungstool Beschränkungen hinsichtlich der Gestaltung der Diagramme aufweist, sodass Sie von dem Abweichen müssen, was in der SE2 gelehrt wurde, schreiben Sie das bitte dazu.
2. Wenn Sie absichtlich von dem abweichen, was in der SE2 gelehrt wurde, ist das auch kein Problem, solange Sie das sinnvoll begründen.
3. Für die Grobstruktur kann es sinnvoll sein, Schichten zu definieren.
4. Benutzen Sie «use»-Beziehungen um zu zeigen, welche Pakete auf welche anderen Pakete zugreifen dürfen.
5. Machen Sie im Klassendiagramm deutlich, in welchen Paketen sich die einzelnen Klassen befinden. Das kann dadurch passieren, dass Sie die Klassen in das Paket „schachteln“.
6. Klassen die auf **-Manager** enden oder auf ähnliche Weise mehr einen technischen als einen domänenspezifischen Aspekt darstellen, gehören i. d. R. nicht in das Analysemodell. Technische Aspekte sind Sache des Entwurfs. Analyseklassen sollten vorrangig „etwas sein“ und nicht „etwas tun“.
7. Versuchen Sie, mit den Aktivitätsdiagrammen neben dem groben Spielablauf auch die Spielregeln zu verdeutlichen.
8. Bei Aktivitätsdiagrammen sind die Knoten Aktivitäten und die Kanten Kontrollflüsse.
9. Verlieren Sie sich beim Modellieren der dynamischen Aspekte nicht in Details. Fragen Sie sich immer, was sinnvoll ist. Diagramme, die Sie sich später nicht mehr ansehen, sind wertlos.

## Hilfreiche Links

- UML-Referenz: <http://uml-diagrams.org/>