

# Übung 2

## Planung und Analyse (1)

1. Erstellen Sie einen Fragenkatalog für ein Sondierungsgespräch mit einem Kunden. Dieser möchte, das haben Sie im Vorfeld erfahren, eine Software zunächst nur für die Lohnabrechnung als Teil einer eher langfristig zu erstellenden Finanzbuchhaltung in seinem kleinen Unternehmen (20 Mitarbeiter) von Ihnen entwickeln lassen, wobei sein Budget sehr begrenzt ist. Es möchte dennoch eine für sein Unternehmen maßgeschneiderte Lösung einsetzen, die langfristig durch Sie gewartet und (bis zur genannten Komplett-Finanzbuchhaltung) erweitert werden soll. Welche Fragen würden Sie stellen, um ihm ein „passendes“ Angebot unterbreiten zu können?

*Die Fragen, die für ein Angebot relevant sind, ergeben sich teilweise aus dem **Lastenheft**:*

- (a) *Ausgangssituation und Zielsetzung: Was ist der IST-Zustand (d.h. wie wird die Lohnabrechnung/Finanzbuchhaltung im Moment abgewickelt, soll dies geändert werden oder soll die Software genau die aktuellen Prozesse abbilden)?*
- (b) *Produkteinsatz: Wie soll die Software zum Einsatz kommen? Wer soll mit ihr arbeiten, bzw. Zielpublikum?*
- (c) *Produktübersicht: Was genau soll die Software tun?*
- (d) *Funktionale Anforderungen: Technik, Details der zum Tragen kommenden Geschäftsprozesse...*
- (e) *Nichtfunktionale Anforderungen untergliedert in:*
  - i. *Funktionalität: Welche Funktionalitäten/Prozesse soll die Software beherrschen?*
  - ii. *Benutzbarkeit: Skill-Level des Anwenderkreises, wie „einfach“ muss die Benutzerschnittstelle sein?*
  - iii. *Zuverlässigkeit: Welcher Ausfallsicherheitsgrad wird angestrebt, in welcher Form ist Schutz und Sicherung der Daten notwendig?*
  - iv. *Effizienz: In diesem Beispiel spielt eher die Ergonomie eine Rolle als besonders performante Datenbank-Algorithmen oder schnelle Rechenoperationen, dennoch ist z.B. die Frage nach der benötigten Rechner-Mindestausstattung ein wichtiges Kriterium.*
  - v. *Änderbarkeit: Wann und in welchem Umfang ist geplant, die Software zu erweitern, und wer wird dies ggf. als weiterer Dienstleister tun (Schnittstellendokumentation)?*
  - vi. *Übertragbarkeit: Inwieweit lassen sich die abgedeckten Funktionalitäten anderweitig nutzen? Wer darf das (Teil der Lizenzbestimmungen)?*
- (f) *Risikoakzeptanz: Wie unverzichtbar ist die Software in welchem Bereich, ist sie als unternehmenskritisch einzustufen, wer ist für „Notfälle“ erreichbar, und gibt es Alternativen?*

- (g) *Skizze des Lebenszyklus und der Systemarchitektur: Wie wird die Software (ggf. schrittweise) im Betrieb etabliert, wie und wann erweitert, wie ist sie entsprechend schnittstellenmäßig zu definieren und in der Bauweise zu entwerfen?*
  - (h) *Lieferumfang: Was sind die Aufgaben des Auftragnehmers, und welche Aufgaben übernimmt er explizit NICHT? Wie lauten die Nutzungsbedingungen (Lizenz), unter der die Software überlassen wird?*
  - (i) *Abnahmekriterien: Unter welchen belegbaren Voraussetzungen gilt die Software als „fertig“, und wie und wann wird dies vom Kunden bestätigt?*
2. Überlegen Sie sich, welchen Einfluss die Beantwortung der Fragen auf den von Ihnen kalkulierten Arbeitsaufwand, und damit die Kosten der Entwicklung bis zur Fertigstellung, haben wird. Notieren Sie sich die entsprechenden Kriterien sortiert nach „Fixpreis“, „Laufenden Kosten“ und „Faktoren“ (d.h. Multiplikatoren, die sich gravierend auf die Preisgestaltung auswirken können).
- (a) *Einfache Faktoren mit Fixpreis (addieren sich auf):*
    - i. *Einzelne Funktionen, die implementiert werden müssen,*
    - ii. *Dokumentation,*
    - iii. *Lauffähigkeit auf bestimmten Plattformen gewährleisten,*
    - iv. *Anpassung auf bestimmte Arbeitssituationen.*
  - (b) *Laufende (regelmäßige, sporadische) Kosten, die im Laufe des Software-Lebenszyklus anfallen können:*
    - i. *Betriebskosten (Strom, Material wie Datenträger für Backups, Hardware-Erweiterungen, Arbeitszeit für Installation und Wartung),*
    - ii. *Erweiterungen, Updates,*
    - iii. *Backup, Sicherungen,*
    - iv. *Skalierung auf weitere Rechnersysteme,*
    - v. *Portierungen, Datenübernahme,*
    - vi. *Schulungen, Weiterbildung,*
    - vii. *Garantiefälle/Bugfixes.*
  - (c) *Multiplikatoren (vervielfachen die meisten Arbeitspreise):*
    - i. *Lizenz, beispielsweise:*
      - A. *Open Source würde erlauben, dass man sich bei der Softwareentwicklung aus einem großen Pool an vorhandenen und unter Open Source Lizenz stehenden Komponenten bedienen kann, ohne eine Nutzungsgebühr für das geistige Eigentum der Vorentwickler entrichten zu müssen. Allerdings darf dann auch keine „Nutzungsgebühr pro Anwender“ bei der Auslieferung der Software erhoben werden, und der vollständige Quelltext muss, ohne Beschränkung der weiteren Nutzung, an den Kunden mit ausgeliefert werden.*
      - B. *Eine proprietäre Lizenz erlaubt die Erhebung einer „Nutzungsgebühr pro Kopie“, bedingt aber, dass man die vollständigen Rechte und Lizenzen an der entwickelten Software, und allen Komponenten davon, besitzen muss,*

*und keinerlei Open Source Komponenten mit einbauen darf (was die Entwicklungskosten in die Höhe schraubt). Der Kunde hat bei proprietären Lizenzen i.d.R. keine Rechte an der Software, außer der kostenpflichtigen Nutzung, was für ihn wiederum ein Kriterium sein kann, sich einen anderen Anbietern auszusuchen.*

*Die Frage sollte also gestellt werden, ob und in wie weit, der Kunde Einfluss auf die weitere Entwicklung und (Eigen-)Vermarktung der Software haben möchte, und ob ihm ein einfaches Nutzungsrecht mit oder ohne Exklusivität ausreicht.*

- ii. Anzahl der Arbeitsplätze, die unmittelbar durch den Einsatz der Software betroffen sind. Dies beeinflusst das Risiko von Garantieansprüchen in allen Bereichen der Software-Funktionalität.*
- iii. Vorhandensein einer eigenen Wartungs- oder Entwicklerinstanz beim Kunden, d.h. soll die Software auch durch den Kunden wartbar sein, oder soll er bei allen Produkthaftungsfragen generell beim Softwarehersteller anfragen (müssen)?*
- iv. Welche Wartungs/Erweiterungsintervalle sind zu erwarten? Kommen häufig neue Ideen ins Spiel, die kurzfristig umgesetzt werden sollen (aka „Extreme Programming“)?*

3. Bilden Sie Ihren Fragebogen aus der ersten Aufgabe als Java-Klasse ab. Verwenden Sie Aggregation und/oder Vererbung, wo es Ihnen sinnvoll erscheint. Halten Sie den Programmier-Aufwand für den Fragebogen gering, d.h. konzentrieren Sie sich auf die Berechnungsvorschriften und nicht auf als gegeben angenommene Routinen zum Einlesen von Daten. Es muss kein lauffähiges Programm entstehen (je nach Ihren Java-Vorkenntnissen dürfen Sie aber natürlich eins schreiben). *Wir belassen es bei dieser Aufgabe bei einem einfachen Beispiel. Natürlich sollten die Ergebnisse einer Umfrage auch in irgendeiner Form gespeichert werden. Dies sollte die **Fragebogen**-Klasse in der Endfassung auch beherrschen, neben der Abfrage aller wichtigen Punkte und Beschreibungen für das Lastenheft.*

```
public class Situation { // IST, SOLL
    private String _istzustand;
    private String _sollzustand;
    private int _arbeitstage_geschaetzt;

    public String getIstZustand() { return _istzustand; }

    public void input() {
        _istzustand = prompt("Beschreibung aktueller Zustand: ");
    }

    ...
}

public class Fragebogen {
    Situation situation;
    private final int tagessatz_programmieren = 750;
```

```

...

private int gesamtpreis; // Normierter Gesamtpreis

// "Erhöhungsfaktor" aufgrund bestimmter Kriterien
private float faktor;

Fragebogen() { // Konstruktor
    // Startwert, f. Angebotserstellung, Sondierungsgespräch ...
    gesamtpreis = 500;

    // Frage nach IST und SOLL-Zustand stellen...
    situation = new Situation();

...

    // Abschätzung des Programmieraufwandes aufgrund der Lizenz
    if(askLizenzWunsch_is_OpenSource() == true) faktor = 1.0;
    else                                     faktor = 1.5;

...

    System.out.println("Aktuelle Kostenabschätzung des Projektes: " +
                       gesamtpreis * faktor);
    System.out.println("Details: ")
    printProjectDetails();
}
}

```

4. Erstellen Sie aufgrund der (hypothetischen) ausgewerteten Eingabedaten Ihres Fragebogens, die Sie frei wählen können, ein **Pflichtenheft** sowie ein einfach gehaltenes Angebot (Stichpunkte, nicht ausformuliert) für den Kunden.<sup>1</sup> Personentage können Sie mit 750-950 € ansetzen, je nach Skill-Level.

Siehe [uebung02-angebot-beispiel.pdf](#).

5. ROBOCODE: Lassen Sie die „sample“ Roboter gegeneinander antreten und schauen Sie sich die „Teams“, gebildet aus **Teamrobots** an. Versuchen Sie, nachzuvollziehen, wie die Teamrobots miteinander kommunizieren und sich gegenseitig erkennen können. *Der Schlüssel zur „Teamfähigkeit“ ist hier die Klasse **Teamrobot** aus der Robocode-API, in der Methoden wie **isTeammate()** und Messaging-Funktionen enthalten sind.*

Falls Ihnen Teile dieser Übung zu umfangreich bzw. zu zeitintensiv erscheinen, belassen Sie es bitte bei „Ansätzen“. Der Sinn der Übung ist nicht die Fleißarbeit, sondern das Verständ-

---

<sup>1</sup>Für diese Aufgabe können Sie mit 1-2 Seiten Text rechnen, inklusive Preiskalkulation, normalerweise sind solche Angebote natürlich wesentlich detail- und umfangreicher. Sie erhalten in der nächsten Übungsstunde ein Beispiel als Musterlösung.

nis beim Vorgehen bei einer Angebotserstellung bzw. einer einigermaßen realistischen Aufwandsabschätzung (im Gegensatz zu dem gigantischen Aufwand, den Sie bei Übung 1 hätten erkennen sollen).