

Software-Qualitätssicherung

Übung 1 – Abgabe

Bitte lesen Sie die Angabe sorgfältig durch und achten Sie dabei auch auf den Disclaimer.

Generell gilt, dass die Anzahl der zur Verfügung gestellten Zeilen nur aus Formatierungsgründen besteht. Es können auch Zeilen hinzugefügt oder leer gelassen werden.

Ihre Daten – Bitte füllen Sie alle Felder aus:

Nachname, Vorname:	Zahirovic, Belmin
Matrikelnummer:	01428177
Studienkennzahl:	033 534
E-Mail Adresse:	e1428177@student.tuwien.ac.at

Inhalt

- 1 Anforderungsanalyse und Review1
 - 1.1 Review der Anforderungen2
 - 1.2 Design Review der Anwendungsfälle4
- 2 Risk Based Testing6
 - 2.1.1 Quality/Feature Matrix6
 - 2.1.2 Risk-Profile8
 - 2.2 Planen des Reviewprozesses9

1 Anforderungsanalyse und Review

1.1 Review der Anforderungen

Nr.	Typ	Rückverfolgbarkeit	Überprüfbar
1	Functional	Die Anforderung ist nicht genau in der Projektbeschreibung spezifiziert.	Ja, durch einen Test: Wenn sich ein Mitarbeiter einloggt, dann hat er/sie den Überblick (z.B. seine/ihre Rolle- Verschiedene Mitarbeiter haben verschiedene Rolle)
2	Functional	Jede Mitarbeiter können Kunden in das System einfügen, das ist eine wesentliche Eigenschaft/Funktion dieses Systems. Im meisten Fällen machen das aber Kassierer.	Ja, es wird ständig (Datenbank) getestet und überprüft (sowohl vor als auch dem Einfügen der Informationen).
3	Functional	Diese Verantwortlichkeit hat der Systemadministrator. Weitere, genauere Beschreibung ist nicht angegeben.	Ja, es wird ständig (Datenbank-von Administratoren) getestet und überprüft (sowohl vor als auch dem Einfügen der Informationen).
4	Functional	Die Anforderung ist nicht genau in der Projektbeschreibung spezifiziert. Es ist aber klar, dass z.B. eine Kunde mehrere Filme ausleihen kann. (Aus Abschnitt 2.4)	Ja, durch die Überprüfung der Datenbank (während Ausleihen/ zurückgeben der Filme).
5	Functional	Die Anforderung ist nicht genau in der Projektbeschreibung spezifiziert: Es ist nicht bekannt ob die Kundendaten auf dem Beleg sein müssen oder nicht.	Ja, durch testen während der Implementation.
6	Supportability	Die Anforderung ist nicht in der Projektbeschreibung spezifiziert.	Testen ist manuell möglich.
7	Functional	Die Anforderung ist genau in der Projektbeschreibung spezifiziert.	Ja, durch testen. Es muss klar sein, welche Kunden Stammkunden und welche „normale Kunden“ sind

8	Functional	Die Anforderung ist genau in der Projektbeschreibung spezifiziert.	Ja, durch manuelle oder automatische Prüfungen.
9	Functional	Die Anforderung ist spezifiziert, außer für die SciFi filmen, wo der Faktor x1.15 beträgt.	Es sind keine Tests notwendig.
10	Functional	Die Anforderung ist spezifiziert, es gibt aber einen Widerspruch (für >20 Videopoints erhalten Kunden 5% und nicht 6% Rabatt)	Ja, durch testen, aber in Beschreibung steht 6% statt 5% Rabatt (Für >20 Videopoints)
11	Functional	Die Anforderung ist genau in der Projektbeschreibung spezifiziert.	Ja, durch testen während der Implementation.
12	Functional	Die Anforderung ist nicht in der Projektbeschreibung spezifiziert.	Ja, durch testen während der Implementation.
13	Usability	Die Anforderung ist genau in der Projektbeschreibung spezifiziert.	Keine Tests sind notwendig.
14	Usability	Die Anforderung ist in der Projektbeschreibung spezifiziert.	Keine Tests sind notwendig.
15	Usability	Die Anforderung ist nicht in der Projektbeschreibung spezifiziert.	Ja, nachdem die Kunden die Werbung „akzeptiert“ haben.
16	Usability	Die Anforderung ist in der Projektbeschreibung spezifiziert.	Ja, jedoch nicht automatisierbar.
17	Usability	Die Anforderung ist in der Projektbeschreibung spezifiziert.	Ja, jedoch nicht automatisierbar.
18	Reliability	Die Anforderung ist nicht in der Projektbeschreibung spezifiziert.	Ja, automatisch.
19	Reliability	Die Anforderung ist nicht in der Projektbeschreibung spezifiziert.	Ja, es gibt eine Reihe von Tests für die bessere Zuverlässigkeit des Systems.
20	Functional	Die Anforderung ist in der Projektbeschreibung spezifiziert.	Keine Tests sind notwendig.

21	Supportability	Die Anforderung hat Entwürfs- / Implementierungsvorschläge.	Ja,manuell, Testergebnisse können subjektiv sein.
22	Performance	Die Anforderung ist nicht genau in der Projektbeschreibung spezifiziert.	Ja, durch automatische testen.

1.2 Design Review der Anwendungsfälle

Nr.	Review
1	Anwendungsfall hat zugehörige Anforderung: 1,2. Hier ist ein login notwendig. Anwendungsfall ist unvollständig, da nicht alle Mitarbeiter die Rechte haben, die Kunden zu „bearbeiten“, „ändern“, „erstellen“ usw. Im Diagramm ist daher Akteur name falsch.
2	Anwendungsfall hat zugehörige Anforderung: 3. Anwendungsfall ist auch unvollständig. Es ist uns nicht bekannt (laut Anforderung), wer die Filme fügt hinzu, ändern und so weiter, und laut Anwendungsfallbeschreibung macht dass immer Administrator. Außerdem nirgenwo in der Anforderungen steht dass der Film wird gekauft, wenn es für den Filmverleih interresant ist.Dasselbe gilt auch für den Preis.
3	Anwendungsfall hat zugehörige Anforderung: 4,5,6. Hier sind aber keine Videopoints und Rabatte erwähnt.
4	Anwendungsfall hat zugehörige Anforderung:4,5 (auch 1 für login).
5	Anwendungsfall hat keine zugehörige Anforderung. Filmpreisanpassung ist nur oberflächlich in Szenario erwähnt und es gibt keine Anforderung die wir mit dieser Filmpreisanpassung verknüpfen können.
6	Anwendungsfall hat zugehörige Anforderung:7. Laut Diagramm kann Controller Film- und Kundenstatistik auswerten, er soll aber auch Stammkunden anlegen können.

2 Risk Based Testing

2.1.1 Quality/Feature Matrix

Tragen Sie Ihre Risikoeinschätzungen in die Matrix ein, verwenden Sie dazu den jeweiligen Anfangsbuchstaben (C, I, L) sowie eine farbliche Kodierung.

C: Critical, I: Important, L: Less important

Anforderung , Feature / Non-functional Requirement	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Usability	I	C	C	I	I	I	I	I	I	I	L	L	I	C	L	L	I	C	C	L	C	C
Reliability	I	I	C	I	C	C	L	I	I	I	L	C	I	I	I	I	I	C	C	L	I	I
Performance/Efficiency Capacity/Responsetime	L	L	L	I	I	L	L	I	L	I	L	L	L	L	L	L	I	C	C	I	L	C
Supportability	L	I	L	L	L	L	L	L	L	L	I	L	C	L	L	L	L	I	C	L	C	I
Security/Privacy/Safety/Integrity	L	C	L	L	L	L	L	L	L	L	L	I	L	L	I	L	L	C	L	L	L	L
Availability	L	L	L	L	L	L	I	L	L	L	L	L	L	L	L	L	I	I	C	L	L	L
Extensibility	L	L	I	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	I	L

Maintainability	L	I	I	L	I	L	L	L	L	L	L	L	L	L	I	L	L	L	I	L	C	L
-----------------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

2.1.2 Risk-Profile

Welche 5 Anforderungen sollen intensiv getestet werden, das heißt welche 5 Anforderungen haben ein großes Risiko-Profil laut Tabelle in 2.1?

1. Funktionstüchtigkeit
2. Filtern
3. Backup
4. Erweiterbarkeit
5. Buchungsbeleg

Nr.	Anforderung	Positiver Test	Negativer Test
1	Funktionstüchtigkeit	Es gibt keine Fehlermeldungen/Errors während Programmausführung. Das Programm läuft ohne Probleme.	Es gibt Fehlermeldungen, und das Programm gibt diese aus. (Irgendwelche Errors die nicht zu erwarten waren)
2	Filtern	Performance, max. Dauer ist 0.5s (Also Antwortzeit <0,5s)	Dauer>0.5s (Antwortzeit)
3	Backup	Alle Daten sind gesichert und auf die „sichere, gewünschte“ Stelle gespeichert. (Backup wird jeden Tag durchgeführt)	Die Daten fehlen (Sind nicht Verfügbar).
4	Erweiterbarkeit	Die Möglichkeit das System zu erweitern.	Die Erweiterung ist aus irgendwelchem Grund nicht möglich.
5	Buchungsbeleg	Buchungsbeleg ist korrekt (Die Daten sind korrekt aufsummiert)	Die Rechnungssumme ist nicht wie erwartet.

2.2 Planen des Reviewprozesses

Nr.	Reviewart	Schwerpunkt	Rolle	Lesetechnik	Zeitpunkt	Beginn der Vorbereitung	Nacharbeit	Begründung
X	Codereview	Smells	M, L, G, S, A	Checklisten	Nach 8 Wochen	2 Wochen vor Review	Ausbessern der Smells	Nach 8 Wochen sollen schon einige Pakete umgesetzt sein. Ein Code Review soll dem Team „schlechten“ Code aufzuzeigen. Für die Weiterentwicklung soll daraus gelernt werden.
1	Software Requirements Review	Anforderungen	M,L,G,S,A	Checklist	Nach eine Woche	3 Wochen vor Review	Check ob die Anforderungen erfüllt sind.	Das ist das erste Review, dient um leichtere, frühere Fehler herauszufinden. Bildet eine gute Basis.
2	Preliminary Design Review	Design	M,S,A,G	Ad-Hoc	Nach 2 Wochen	1 Woche vor Review	Die Verbesserungen in Design.	Vereinbarung über die bestimmte Designkonzepte.
3	Critical Design Review	Design	M,S,A,G	Ad-Hoc	Nach 2 Wochen	2 Wochen vor Review	Überprüfung der Design-Anforderungen.	Das letzte Review, Vereinbarung über finales Design.

4	In- Process Review	Quality und Performance	S,A,G	Ad-Hoc	Nach 3 Wochen	1 Woche vor Review	Verbesserungen während der Implementation bzw. Entwicklung.	Es ist effizienter, die Fehler früher, während der Implementation herauszufinden, als später (teurer)
5	Management Review	Development	M,A,G	Checklist	Nach 4 Wochen	2 Wochen vor Review	Enwicklungskorrekturen	Es soll klarer sein, an welcher Stelle Verbesserungen möglich sind.
6	Inspection	Defects	M,L,G,S,A	Checklist	Nach 4 Wochen	2 Wochen vor Review	Fehler die oft auftreten herausfinden.	Suchen nach Defekte, es werden nach Designfehler gesucht.(=Inspektion)
7	Code Walkthrough	General problems	M,A,G	Checklist	Nach 5 Wochen	1 Woche vor Review	Probleme bzw. Fehler identifizieren und beheben.	Vom Reviewer vorgetragen, Distribution der Informationen, Projekt durchläuft durch die üblichen Anwendungsfälle.
8	Technical Review	Why and where is Product usable	M,L,G,S,A	Ad-Hoc	Nach 6 Wochen	2 Woche vor Review	Fehler ausbessern.	Testen im Team um Fehler zu identifizieren.Die Verwendung zeigt die Fehler.

