

QSE-ASE-08

Tenez

Testfälle

Alexander Bohn

Manuel Djalili

Dominik Gruber

Manuel Mayrhofer

Philipp Naderer

13. November 2009

Inhaltsverzeichnis

1	Test-Testfall	5
1.0.1	Testfall:	5
2	Testfälle	7
2.1	Testfälle zu UC-01-01 (Verein anlegen)	7
2.1.1	Testfall: E-mail Adresse	7
2.1.2	Testfall: Telefonnummer	8
2.1.3	Testfall: Name	8
2.1.4	Testfall: Adresse	9
2.1.5	Testfall: URL	9
2.1.6	Testfall: Währung	11
2.1.7	Testfall: Zeitzone	11
2.2	Testfälle zu UC-01-02 (Vereine anzeigen)	13
2.2.1	Testfall: Name	13
2.3	Testfälle zu UC-01-03 (Verein löschen)	13
2.3.1	Testfall: Name	13
2.4	Testfälle zu UC-01-04 (Plätze löschen)	14
2.4.1	Testfall: ID-Plätze	14
2.4.2	Testfall: Name	15
2.5	Testfälle zu UC-01-05 (Platz anlegen)	15
2.5.1	Testfall: Name	15
2.5.2	Testfall: Beschreibung	17
2.6	Testfälle zu UC-01-06 (Vereinsdaten bearbeiten)	17
2.6.1	Testfall: E-mail Adresse	17
2.6.2	Testfall: Telefonnummer	19
2.6.3	Testfall: Name	19
2.6.4	Testfall: Adresse	20
2.6.5	Testfall: URL	21
2.7	Testfälle zu UC-01-07 (Usergruppe anlegen)	22
2.7.1	Testfall: Name	22
2.7.2	Testfall: Beschreibung	23
2.7.3	Testfall: Rabatt	23
2.8	Testfälle zu UC-01-08 (Plätze anzeigen)	24
2.8.1	Testfall: ID	24
2.9	Testfälle zu UC-01-09 (Zeiteinheiten festlegen)	25
2.9.1	Testfall: Zeit	25
2.9.2	Testfall: Intervall	26
2.9.3	Testfall: Periode	29
2.9.4	Testfall: Tag	29

2.9.5	Testfall: Preis	29
2.10	Testfälle zu UC-01-10 (Zeiteinheit ändern)	31
2.10.1	Testfall: Zeit	31
2.10.2	Testfall: Intervall	31
2.10.3	Testfall: Periode	34
2.10.4	Testfall: Tag	36
2.10.5	Testfall: Preis	36
2.11	Testfälle zu UC-01-11 (Zeiteinheit löschen)	37
2.11.1	Testfall: ID-Zeiteinheit	37
2.12	Testfälle zu UC-01-12 (Platz reservieren)	37
2.12.1	Testfall: Datum	37
2.12.2	Testfall: start	39
2.12.3	Testfall: ende	40
2.12.4	Testfall: Wert	42
2.12.5	Testfall: Rabatt	42
2.12.6	Testfall: Preis	44
2.13	Testfälle zu UC-01-13 (Standort anzeigen)	46
2.13.1	Testfall: Name	46
2.13.2	Testfall: ID	46
2.14	Testfälle zu UC-01-14 (Standort löschen)	47
2.14.1	Testfall: Name	47
2.14.2	Testfall: ID	49
2.15	Testfälle zu UC-01-15 (Standort anlegen)	50
2.15.1	Testfall: Name	50
2.15.2	Testfall: Beschreibung	50
2.16	Testfälle zu UC-01-16 (Rechtegruppe zuordnen)	52
2.16.1	Testfall: ID	52
2.17	Testfälle zu UC-02-01 (Kunden registrieren)	53
2.17.1	Testfall: Name	53
2.17.2	Testfall: Verein	54
2.17.3	Testfall: Adresse	54
2.17.4	Testfall: Telefonnummer	55
2.17.5	Testfall: Email	56
2.17.6	Testfall: Guthaben	57
2.18	Testfälle zu UC-02-02 (Kunden anzeigen)	58
2.18.1	Testfall: Name	58
2.18.2	Testfall: ID	59
2.19	Testfälle zu UC-02-03 (Gruppe zuordnen)	59
2.19.1	Testfall: ID	59
2.19.2	Testfall: Name	59
2.20	Testfälle zu UC-02-04 (Kunden löschen)	60
2.20.1	Testfall: Name	60
2.21	Testfälle zu UC-02-05 (Guthaben verändern)	60
2.21.1	Testfall: Wert	60
2.22	Testfälle zu UC-02-06 (Guthaben aufladen)	61
2.22.1	Testfall: Wert	61
2.23	Testfälle zu UC-02-07 (Kundendaten exportieren)	62
2.23.1	Testfall: Name	62
2.24	Testfälle zu UC-02-08 (Newsletter verschicken)	63

2.24.1	Testfall: Name	63
2.25	Testfälle zu UC-02-09 (registrieren)	64
2.25.1	Testfall: Name	64
2.25.2	Testfall: Verein	64
2.25.3	Testfall: Adresse	65
2.25.4	Testfall: Telefonnummer	65
2.25.5	Testfall: Email	67
2.26	Testfälle zu UC-02-10 (anmelden)	67
2.26.1	Testfall: Name	67
2.26.2	Testfall: Passwort	68
2.27	Testfälle zu UC-02-11 (Guthaben abfragen)	70
2.27.1	Testfall:	70
2.28	Testfälle zu UC-02-12 (Buchungshistory anzeigen)	70
2.28.1	Testfall:	70
2.29	Testfälle zu UC-02-13 (Platz reservieren)	70
2.29.1	Testfall: Datum	70
2.29.2	Testfall: Wert	70
2.29.3	Testfall: Rabatt	70
2.29.4	Testfall: Preis	70
2.30	Testfälle zu UC-02-14(Daten bearbeiten)	70
2.30.1	Testfall: Name	70
2.30.2	Testfall: Verein	70
2.30.3	Testfall: Adresse	70
2.30.4	Testfall: Telefonnummer	70
2.30.5	Testfall: Email	70
2.31	Testfälle zu UC-02-15(Admin kontaktieren)	70
2.31.1	Testfall: Text	70
2.32	Testfälle zu UC-03-01(Auswertung wählen)	70
2.32.1	Testfall:	70
2.33	Testfälle zu UC-03-02(Reports on demand generieren)	70
2.33.1	Testfall:	70
2.34	Testfälle zu UC-03-03(Top Tageszeiten)	70
2.34.1	Testfall:	70
2.35	Testfälle zu UC-03-04(Auswärtige Kunden vs. Mitglieder)	70
2.35.1	Testfall:	70
2.36	Testfälle zu UC-03-05(Buchungsbelegung)	70
2.36.1	Testfall:	70
2.37	Testfälle zu UC-03-06(Meistgebuchte Wochen)	70
2.37.1	Testfall:	70
2.38	Testfälle zu UC-03-07(Kundenliste)	70
2.38.1	Testfall:	70
2.39	Testfälle zu UC-03-08(Zugriffsauswertung)	70
2.39.1	Testfall:	70
2.40	Testfälle zu UC-03-09(Filterung aktivieren)	70
2.40.1	Testfall:	70
2.41	Testfälle zu UC-03-10(periodische Reports generieren)	70
2.41.1	Testfall:	70
2.42	Testfälle zu UC-04-01(Reservierung abrufen)	70
2.42.1	Testfall: Datum	70

2.42.2	Testfall: Platzname	70
2.42.3	Testfall: Kundenname	70
2.43	Testfälle zu UC-04-02(Reservierung stornieren)	70
2.43.1	Testfall: Datum	70
2.43.2	Testfall: Platzname	70
2.43.3	Testfall: Kundenname	70
2.44	Testfälle zu UC-04-03(Reservierung anlegen)	70
2.44.1	Testfall: Datum	70
2.44.2	Testfall: Platzname	70
2.44.3	Testfall: Kundenname	70
2.44.4	Testfall: Wert	70
2.44.5	Testfall: Rabatt	70
2.44.6	Testfall: Preis	70
2.45	Testfälle zu UC-04-03(Abonnement anlegen)	70
2.45.1	Testfall: Datum	70
2.45.2	Testfall: Platzname	70
2.45.3	Testfall: Kundenname	70
2.45.4	Testfall: Wert	70
2.45.5	Testfall: Rabatt	70
2.45.6	Testfall: Preis	70
2.45.7	Testfall: Dauer	70
2.46	Version	71

Kapitel 1

Test-Testfall

1.0.1 Testfall:

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

2a) gültig: maximal 50 beliebige Zeichen aus M1

2b) ungültig: mehr als 50 Zeichen aus M1

2c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF			1a, 2a,	
2	FF			1b	Fehlermeldung erscheint
3	FF			2b	Fehlermeldung erscheint
4	FF			2c	Fehlermeldung erscheint

Kapitel 2

Testfälle

2.1 Testfälle zu UC-01-01 (Verein anlegen)

2.1.1 Testfall: E-mail Adresse

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ], [],[@],[/],[],[+]
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

- 2a) gültig: [@] kommt exakt einmal vor
- 2b) ungültig: [@] kommt öfter als 1 mal vor
- 2c) ungültig: [@] kommt gar nicht vor

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige E-Mail Adresse wird eingegeben: tenez-reservierung+24@gmx.at	1a, 2a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: ten«>()"@gmx.at	1b	Fehlermeldung erscheint
3	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: tenez@@gmx.at	2b	Fehlermeldung erscheint
4	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: tenez.gmx.at	2c	Fehlermeldung erscheint

2.1.2 Testfall: Telefonnummer

Äquivalenzklassen

1a) gültig: M1 = natürliche Zahlen, [+], [/], [Aa-zZ],[] 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige Telefonnummer wird eingegeben: +43 0676/123456 privat	1a, 2a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige Telefonnummer: 676«>"123456	1b	Fehlermeldung erscheint

2.1.3 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ],[]
1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes
1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültiger Name wird eingegeben: Tennis1210	1a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültiger Name: Tennis@1210"/.Wien	1b	Fehlermeldung erscheint
3	FF	Alle anderen Vereinsdaten korrekt eingegeben	kein Name wird eingegeben	1c	Fehlermeldung erscheint
5	FF	Alle anderen Vereinsdaten korrekt eingegeben aber Vereinsname schon in DB	gültiger Name	1a	Fehlermeldung erscheint

2.1.4 Testfall: Adresse

Äquivalenzklassen

1a) gültig: Jedes Zeichen

1b) ungültig: Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige Adresse wird eingegeben: Astraße 150/50	1a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	Code	1b	Fehlermeldung erscheint

2.1.5 Testfall: URL

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ], [-], [/], [.]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

2a) gültig: http://einmal

2b) ungültig: http://keinmal

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige URL wird eingegeben: http://www.Tennis1210.at	1a, 2a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige URL: www.Tennis@1210.at	1b, 2b	Fehlermeldung erscheint

2.1.6 Testfall: Währung

Äquivalenzklassen

1a) gültig: M1= [Aa-zZ]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige Währung wird eingegeben: Euro	1a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige Währung: E@ur0	1b	Fehlermeldung erscheint

2.1.7 Testfall: Zeitzone

Äquivalenzklassen

1a) gültig: M1= Aa-zZ, [/]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige Zeitzone wird eingegeben: Europa/Austria	1a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige Zeitzone: Europa23	1b	Fehlermeldung erscheint

2.2 Testfälle zu UC-01-02 (Vereine anzeigen)

2.2.1 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ], []

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Verein muss in der DB sein	gültiger Name wird eingegeben: Tennis1210	1a	Verein wird angezeigt
2	FF	Verein muss in der DB sein	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	FF	Verein ist nicht in der DB	gültiger Name wird eingegeben	1a	Fehlermeldung erscheint
4	NF	Verein(e) in der DB	kein Name wird eingegeben	1a	Alle vereine werden angezeigt

2.3 Testfälle zu UC-01-04 (Plätze deaktivieren)

2.3.1 Testfall: ID-Plätze

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Zeiteinheit muss in der DB sein	gültige ID wird eingegeben: 0001	1a	Zeiteinheit wird deaktiviert
2	FF	Zeiteinheit muss in der DB sein	ungültige ID: 15@30	1b	Fehlermeldung erscheint
3	FF	Zeiteinheit muss in der DB sein	keine ID wird eingegeben	1c	Fehlermeldung erscheint
4	FF	Zeiteinheit ist nicht in der DB	gültige ID wird eingegeben	1a	Fehlermeldung erscheint

2.3.2 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Verein muss in der DB sein	gültiger Name wird eingegeben: Platz01	1a	Verein wird angezeigt
2	NF	Verein muss in der DB sein	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	NF	Verein muss in der DB sein	kein Name wird eingegeben	1c	Fehlermeldung erscheint
4	NF	Verein ist nicht in der DB	gültiger Name wird eingegeben	1a	Fehlermeldung erscheint

2.4 Testfälle zu UC-01-05 (Platz anlegen)

2.4.1 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Verein muss in der DB sein	gültiger Name wird eingegeben: Platz01	1a	Verein wird angezeigt
2	NF	Verein muss in der DB sein	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	NF	Verein muss in der DB sein	kein Name wird eingegeben	2c	Fehlermeldung erscheint
4	NF	Verein ist nicht in der DB	gültiger Name wird eingegeben	1a	Fehlermeldung erscheint

2.4.2 Testfall: Beschreibung

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Verein muss in der DB sein	gültige Beschreibung wird eingegeben: Platz01	1a	Verein wird angezeigt
2	FF	Verein muss in der DB sein	ungültige Beschreibung: Code	1b	Fehlermeldung erscheint
3	FF	Verein ist nicht in der DB	gültige Beschreibung wird eingegeben	1a	Fehlermeldung erscheint

2.5 Testfälle zu UC-01-06 (Vereinsdaten bearbeiten)

2.5.1 Testfall: E-mail Adresse

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ], [.,[@],[/],[-],[+]
1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

- 2a) gültig: [@] kommt exakt einmal vor
2b) ungültig: [@] kommt öfter als 1 mal vor
2c) ungültig: [@] kommt gar nicht vor

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige E-Mail Adresse wird eingegeben: tenez-reservierung+24@gmx.at	1a, 2a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: ten«>()"@gmX.at	1b	Fehlermeldung erscheint
3	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: tenez@@gmX.at	2b	Fehlermeldung erscheint
4	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: tenez.gmx.at	2c	Fehlermeldung erscheint

2.5.2 Testfall: Telefonnummer

Äquivalenzklassen

- 1a) gültig: M1 = natürliche Zahlen, [+], [/], [Aa-zZ],[] 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige Telefonnummer wird eingegeben: +43 0676/123456 privat	1a, 2a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige Telefonnummer: 676 «>"123456	1b	Fehlermeldung erscheint

2.5.3 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ],[]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültiger Name wird eingegeben: Tennis1210	1a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültiger Name: Tennis@1210"/.Wien	1b	Fehlermeldung erscheint
3	FF	Alle anderen Vereinsdaten korrekt eingegeben	kein Name wird eingegeben	1c	Fehlermeldung erscheint
5	FF	Alle anderen Vereinsdaten korrekt eingegeben aber Vereinsname schon in DB	gültiger Name	1a	Fehlermeldung erscheint

2.5.4 Testfall: Adresse

Äquivalenzklassen

- 1a) gültig: Jedes Zeichen
 1b) ungültig: Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige Adresse wird eingegeben: Astraße 150/50	1a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	Code	1b	Fehlermeldung erscheint

2.5.5 Testfall: URL

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ], [-], [/], [.]
 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

- 2a) gültig: http://einmal
 2b) ungültig: http://keinmal

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige URL wird eingegeben: http://www.Tennis1210.at	1a, 2a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige URL: www.Tennis@1210.at	1b, 2b	Fehlermeldung erscheint

2.6 Testfälle zu UC-01-07 (Usergruppe anlegen)

2.6.1 Testfall: Name

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Usergruppendaten korrekt eingegeben	gültiger Name wird eingegeben: Tennis1210	1a	Usergruppe wird in DB gespeichert
2	FF	Alle anderen Usergruppendaten korrekt eingegeben	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	FF	Alle anderen Usergruppendaten korrekt eingegeben	kein Name wird eingegeben	1c	Fehlermeldung erscheint
5	FF	Alle anderen Usergruppendaten korrekt eingegeben und Gruppe schon vorhanden	gültiger Name wird eingegeben	1a	Fehlermeldung erscheint

2.6.2 Testfall: Beschreibung

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ], [],[.],[!],[?]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Usergruppendaten korrekt eingegeben	gültige Beschreibung wird eingegeben: Tennis1210	1a	Usergruppe wird in DB gespeichert
2	FF	Alle anderen Usergruppendaten korrekt eingegeben	ungültige Beschreibung: Code	1b	Fehlermeldung erscheint

2.6.3 Testfall: Rabatt

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen bis 100, [., [,
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes, Zahlen >100
- 1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Usergruppendaten korrekt eingegeben	gültige Beschreibung wird eingegeben: 20	1a	Usergruppe wird in DB gespeichert
2	FF	Alle anderen Usergruppendaten korrekt eingegeben	ungültiger Rabatt: Zehn	1b	Fehlermeldung erscheint
3	FF	Alle anderen Usergruppendaten korrekt eingegeben	kein Rabatt	1c	Fehlermeldung erscheint
4	FF	Alle anderen Usergruppendaten korrekt eingegeben	ungültiger Rabatt: 500	1b	Fehlermeldung erscheint

2.7 Testfälle zu UC-01-08 (Plätze anzeigen)

2.7.1 Testfall: ID

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Verein und Platz muss in der DB sein	gültige ID wird eingegeben: 3	1a	Platz wird angezeigt
2	FF	Verein und Platz muss in der DB sein	ungültige ID: yxc	1a	Fehlermeldung erscheint
3	FF	Verein ist nicht in der DB	gültige ID wird eingegeben	1a	Fehlermeldung erscheint
4	FF	Platz ist nicht in der DB	gültige ID wird eingegeben	1a	Fehlermeldung erscheint
5	NF	Verein und Plätze in der DB	keine ID wird eingegeben	1a, 2a	Alle Plätze von diesem Verein werden angezeigt

2.8 Testfälle zu UC-01-09 (Zeiteinheiten festlegen)

2.8.1 Testfall: Zeit

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [:]
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes
- 1c) ungültig: keine Zeichen

- 2a) gültig: [:] kommt einmal oder keinmal vor
- 2b) ungültig: [:] kommt öfter als 1 mal vor

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz in der DB	gültig Zeiteinheit wird eingegeben: 08:00	1a, 2a	Zeiteinheit wird in DB gespeichert
2	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz in der DB	ungültige Zeiteinheit wird eingegeben: 08"/+:00	1b	Fehlermeldung erscheint
3	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz in der DB	ungültige Zeiteinheit wird eingegeben: 07::00	2b	Fehlermeldung erscheint
4	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz in der DB	keine Zeiteinheit wird eingegeben	1c	Fehlermeldung erscheint
5	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz nicht in der DB	gültige Zeiteinheit	1a,2a	Fehlermeldung erscheint

2.8.2 Testfall: Datum

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [.]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz ist in der DB	gültiges Datum wird eingegeben: 10.01.2010	1a	Zeiteinheit wird in DB gespeichert
2	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz ist in der DB	ungültiges Datum wird eingegeben: 1a/.0	1b	Fehlermeldung erscheint
3	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz ist in der DB	kein Datum wird eingegeben	1c	Fehlermeldung erscheint
4	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz ist nicht in der DB	gültiges Intervall	1a	Fehlermeldung erscheint

2.8.3 Testfall: Tag

Äquivalenzklassen

1a) gültig: M1 = [0],[1]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

2a) gültig: maximal 7 beliebige Zeichen aus M1

2b) ungültig: mehr als 7 Zeichen aus M1

2c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Zeiteinheiten korrekt eingegeben	Tag ist gültig: 1110011	1a, 2a	Tag wird in DB gespeichert
2	FF	Alle anderen Zeiteinheiten korrekt eingegeben	Tag ist ungültig: 12@0.1	1b	Fehlermeldung erscheint
3	FF	Alle anderen Zeiteinheiten korrekt eingegeben	Tag ist ungültig: [>7 Zeichen]	2b	Fehlermeldung erscheint
4	FF	Alle anderen Zeiteinheiten korrekt eingegeben	keine Zeichen werden eingegeben	2c	Fehlermeldung erscheint

2.8.4 Testfall: Preis

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [.]
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes
- 1c) ungültig: keine Zeichen

- 2a) gültig: [.] einmal
- 2b) ungültig: [.] öfters
- 2c) ungültig: [.] gar nicht

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz in der DB	Preis gültig: 15.50	1a, 2a,	Preis wird gespeichert
2	FF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz in der DB	Preis ungültig: 15a,30	1b	Fehlermeldung erscheint
3	FF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz in der DB	Preis ungültig: 15.5.0	2b	Fehlermeldung erscheint
4	FF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz in der DB	kein Wert eingegeben	1c	Fehlermeldung erscheint
5	FF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz nicht in der DB	Preis gültig	1a, 2a,	Fehlermeldung erscheint

2.9 Testfälle zu UC-01-10 (Zeiteinheit ändern)

2.9.1 Testfall: Zeit

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [:]
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes
- 1c) ungültig: keine Zeichen

- 2a) gültig: [:] kommt einmal oder keinmal vor
- 2b) ungültig: [:] kommt öfter als 1 mal vor

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz in der DB	gültig Zeiteinheit wird eingegeben: 08:00	1a, 2a	Zeiteinheit wird in DB gespeichert
2	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz in der DB	ungültige Zeiteinheit wird eingegeben: 08"/+:00	1b	Fehlermeldung erscheint
3	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz in der DB	ungültige Zeiteinheit wird eingegeben: 07::00	2b	Fehlermeldung erscheint
4	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz in der DB	keine Zeiteinheit wird eingegeben	1c	Fehlermeldung erscheint
5	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz nicht in der DB	gültige Zeiteinheit	1a,2a	Fehlermeldung erscheint

2.9.2 Testfall: Datum

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [.]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz ist in der DB	gültiges Datum wird eingegeben: 10.01.2010	1a	Zeiteinheit wird in DB gespeichert
2	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz ist in der DB	ungültiges Datum wird eingegeben: 1a/.0	1b	Fehlermeldung erscheint
3	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz ist in der DB	kein Datum wird eingegeben	1c	Fehlermeldung erscheint
4	FF	Alle anderen Zeiteinheitsdaten korrekt eingegeben und Platz ist nicht in der DB	gültiges Intervall	1a	Fehlermeldung erscheint

2.9.3 Testfall: Tag

Äquivalenzklassen

1a) gültig: M1= [0],[1]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

2a) gültig: maximal 7 beliebige Zeichen aus M1

2b) ungültig: mehr als 7 Zeichen aus M1

2c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Zeiteinheiten korrekt eingegeben	Tag ist gültig: 1110011	1a, 2a	Tag wird in DB gespeichert
2	FF	Alle anderen Zeiteinheiten korrekt eingegeben	Tag ist ungültig: 12@0.1	1b	Fehlermeldung erscheint
3	FF	Alle anderen Zeiteinheiten korrekt eingegeben	Tag ist ungültig: [>7 Zeichen]	2b	Fehlermeldung erscheint
4	FF	Alle anderen Zeiteinheiten korrekt eingegeben	keine Zeichen werden eingegeben	2c	Fehlermeldung erscheint

2.9.4 Testfall: Preis

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [.]
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes
- 1c) ungültig: keine Zeichen

- 2a) gültig: [.] einmal
- 2b) ungültig: [.] öfters
- 2c) ungültig: [.] gar nicht

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz in der DB	Preis gültig: 15.50	1a, 2a,	Preis wird gespeichert
2	FF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz in der DB	Preis ungültig: 15a,30	1b	Fehlermeldung erscheint
3	FF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz in der DB	Preis ungültig: 15.5.0	2b	Fehlermeldung erscheint
4	FF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz in der DB	kein Wert eingegeben	1c	Fehlermeldung erscheint
5	FF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz nicht in der DB	Preis gültig	1a, 2a,	Fehlermeldung erscheint

2.10 Testfälle zu UC-01-11 (Zeiteinheit löschen)

2.10.1 Testfall: ID-Zeiteinheit

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Zeiteinheit muss in der DB sein	gültige ID wird eingegeben: 1	1a	Zeiteinheit wird gelöscht
2	FF	Zeiteinheit muss in der DB sein	ungültige ID: 15@3a0	1b	Fehlermeldung erscheint
3	FF	Zeiteinheit muss in der DB sein	keine ID wird eingegeben	2c	Fehlermeldung erscheint
5	FF	Zeiteinheit ist nicht in der DB	gültige ID wird eingegeben	1a	Fehlermeldung erscheint

2.11 Testfälle zu UC-01-12 (Platz reservieren)

2.11.1 Testfall: Datum

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [:], [.]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Reservierungsdaten korrekt eingegeben und Platz in der DB	gültig Zeiteinheit wird eingegeben: 21.10.2009	1a	Platz wird reserviert und in DB gespeichert
2	FF	Alle anderen Reservierungsdaten korrekt eingegeben und Platz in der DB	ungültige Zeiteinheit wird eingegeben: 2321.210.2009	1b	Fehlermeldung erscheint
3	FF	Alle anderen Reservierungsdaten korrekt eingegeben und Platz in der DB	keine Zeiteinheit wird eingegeben	1c	Fehlermeldung erscheint
4	FF	Alle anderen Reservierungsdaten korrekt eingegeben und Platz nicht in der DB	gültige Zeiteinheit	1a	Fehlermeldung erscheint

2.11.2 Testfall: Wert

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [.]
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes
- 1c) ungültig: kein Wert

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Reservierungsdaten korrekt eingegeben und Platz in der DB	Wert gültig: 15.50	1a,	Wert wird gespeichert
2	FF	Alle anderen Reservierungsdaten korrekt eingegeben und Platz in der DB	Wert ungültig: 15a,30	1b	Fehlermeldung erscheint
3	FF	Alle anderen Reservierungsdaten korrekt eingegeben und Platz in der DB	kein Wert eingegeben	1c	Fehlermeldung erscheint
4	FF	Alle anderen Reservierungsdaten korrekt eingegeben und Platz nicht in der DB	Wert gültig	1a	Fehlermeldung erscheint

2.11.3 Testfall: Rabatt

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [,], <100
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes, >100
- 1c) ungültig: keine Zeichen

- 2a) gültig: [,] kommt genau einmal oder keinmal vor

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Reservierungsdaten korrekt eingegeben und Platz in der DB	gültige Beschreibung wird eingegeben: 12,0	1a, 2a	Reservierung wird in DB gespeichert
2	FF	Alle anderen Reservierungsdaten korrekt eingegeben und Platz in der DB	ungültiger Rabatt: 250@.20	1b	Fehlermeldung erscheint
3	FF	Alle anderen Reservierungsdaten korrekt eingegeben und Platz in der DB	kein Rabatt	1c	Fehlermeldung erscheint
4	FF	Alle anderen Reservierungsdaten korrekt eingegeben und Platz nicht in der DB	Rabatt gültig	1a, 2a	Fehlermeldung erscheint

2.11.4 Testfall: Preis

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [.]
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes
- 1c) ungültig: keine Zeichen

- 2a) gültig: [.] einmal
- 2b) ungültig: [.] öfters
- 2c) ungültig: [.] gar nicht

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz in der DB	Preis gültig: 15.50	1a, 2a,	Preis wird gespeichert
2	FF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz in der DB	Preis ungültig: 15a,30	1b	Fehlermeldung erscheint
3	FF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz in der DB	Preis ungültig: 15.5.0	2b	Fehlermeldung erscheint
4	FF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz in der DB	kein Wert eingegeben	1c	Fehlermeldung erscheint
5	FF	Alle anderen Zeiteinheiten korrekt eingegeben und Platz nicht in der DB	Preis gültig	1a, 2a,	Fehlermeldung erscheint

2.12 Testfälle zu UC-01-13 (Standort anzeigen)

2.12.1 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Standort muss in der DB sein	gültiger Name wird eingegeben: Standort01	1a, 2a	Standort wird angezeigt
2	NF	Standort muss in der DB sein	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	NF	Standort ist nicht in der DB	gültiger Name wird eingegeben	1a, 2a	Fehlermeldung erscheint

2.12.2 Testfall: ID

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Verein und Standort muss in der DB sein	gültige ID wird eingegeben: 3	1a	Standort wird angezeigt
2	FF	Verein und Standort muss in der DB sein	ungültige ID: 3a	1b	Fehlermeldung erscheint
4	FF	Verein ist nicht in der DB	gültige ID wird eingegeben	1a	Fehlermeldung erscheint
5	FF	Standort ist nicht in der DB	gültige ID wird eingegeben	1a	Fehlermeldung erscheint
6	NF	Verein und Standort in der DB	keine ID wird eingegeben	1a	Alle Standorte von diesem Verein werden angezeigt

2.13 Testfälle zu UC-01-14 (Standort deaktivieren)

2.13.1 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Standort muss in der DB sein	gültiger Name wird eingegeben: Standort01	1a	Standort wird deaktiviert
2	NF	Standort muss in der DB sein	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	NF	Standort muss in der DB sein	kein Name wird eingegeben	1c	Fehlermeldung erscheint
4	NF	Standort ist nicht in der DB	gültiger Name wird eingegeben	1a	Fehlermeldung erscheint

2.13.2 Testfall: ID

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Verein und Standort muss in der DB sein	gültige ID wird eingegeben: 3	1a	Standort wird deaktiviert
2	FF	Verein und Standort muss in der DB sein	ungültige ID: 3a	1b	Fehlermeldung erscheint
3	FF	Verein ist nicht in der DB	gültige ID wird eingegeben	1a	Fehlermeldung erscheint
4	FF	Standort ist nicht in der DB	gültige ID wird eingegeben	1a	Fehlermeldung erscheint
5	FF	Verein und Standort in der DB	keine ID wird eingegeben	1c	Fehlermeldung erscheint

2.14 Testfälle zu UC-01-15 (Standort anlegen)

2.14.1 Testfall: Name

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ]
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes
- 1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Eingaben sind gültig/korrekt	gültiger Name wird eingegeben: Standort01	1a, 2a	Standort wird gespeichert
2	NF	Alle anderen Eingaben sind gültig/korrekt	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	FF	Alle anderen Eingaben sind gültig/korrekt	kein Name wird eingegeben	2c	Fehlermeldung erscheint

2.14.2 Testfall: Beschreibung

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ], [],[.],[!],[?]
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Standortdaten korrekt eingegeben	gültige Beschreibung wird eingegeben: Tennis1210	1a, 2a	Standort wird in DB gespeichert
2	FF	Alle anderen Standortdaten korrekt eingegeben	ungültige Beschreibung: Code	1b	Fehlermeldung erscheint

2.15 Testfälle zu UC-01-16 (Rechtegruppe zuordnen)

2.15.1 Testfall: ID

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Rechtegruppe muss in DB sein	gültige ID wird eingegeben: 3	1a	User wird Rechtegruppe zugeordnet
2	FF	Rechtegruppe muss in DB sein	ungültige ID: 3a	1b	Fehlermeldung erscheint
3	FF	Rechtegruppe ist nicht in der DB	gültige ID wird eingegeben	1a, 2a	Fehlermeldung erscheint

2.16 Testfälle zu UC-02-01 (Kunden anlegen)

2.16.1 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ],[]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben	gültiger Name wird eingegeben: manuel007	1a	Verein wird in DB gespeichert
2	FF	Alle anderen Daten korrekt eingegeben	ungültiger Name: code	1b	Fehlermeldung erscheint
3	FF	Alle anderen Daten korrekt eingegeben	kein Name wird eingegeben	1c	Fehlermeldung erscheint
5	FF	Alle anderen Daten korrekt eingegeben aber Name schon in DB	gültiger Name	1a	Fehlermeldung erscheint

2.16.2 Testfall: Adresse

Äquivalenzklassen

1a) gültig: Jedes Zeichen

1b) ungültig: Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben	gültige Adresse wird eingegeben: Astraße 150/50	1a	Kunde wird in DB gespeichert
2	FF	Alle anderen Daten korrekt eingegeben	Code	1b	Fehlermeldung erscheint

2.16.3 Testfall: Telefonnummer

Äquivalenzklassen

1a) gültig: M1 = natürliche Zahlen, [+], [/], [Aa-zZ],[] 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben	gültige Telefonnummer wird eingegeben: +43 0676/123456 privat	1a	Kunde wird in DB gespeichert
2	FF	Alle anderen Daten korrekt eingegeben	ungültige Telefonnummer: 676 «>"123456	1b	Fehlermeldung erscheint

2.16.4 Testfall: E-mail

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ], [.,][@],[/],[.],[+]
1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

- 2a) gültig: [@] kommt exakt einmal vor
2b) ungültig: [@] kommt öfter als 1 mal vor
2c) ungültig: [@] kommt gar nicht vor

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige E-Mail Adresse wird eingegeben: tenez-reservierung+24@gmx.at	1a, 2a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: ten«>()"@gmx.at	1b	Fehlermeldung erscheint
3	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: tenez@@gmx.at	2b	Fehlermeldung erscheint
4	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: tenez.gmx.at	2c	Fehlermeldung erscheint

2.16.5 Testfall: Guthaben

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [·],[,]
1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

- 2a) gültig: [·] oder [,] einmal
2b) ungültig: [·] oder [,] öfters
2c) gültig: [·] oder [,] gar nicht

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben	Guthaben gültig: 1550	1a, 2a, 3a	Preis wird gespeichert
2	FF	Alle anderen Daten korrekt eingegeben	Guthaben ungültig: 15a,30	1b	Fehlermeldung erscheint
3	FF	Alle anderen Daten korrekt eingegeben	Guthaben ungültig: 15.5.0	2b	Fehlermeldung erscheint

2.17 Testfälle zu UC-02-02 (Kunden anzeigen)

2.17.1 Testfall: Name

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ], []
1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Kunde muss in der DB sein	gültiger Name wird eingegeben: Manuel m007	1a, 2a	Kunde wird angezeigt
2	FF	Kunde muss in der DB sein	ungültiger Name: Code	1b	Fehlermeldung erscheint
4	FF	Kunde ist nicht in der DB	gültiger Name wird eingegeben	1a	Fehlermeldung erscheint
5	NF	Kunde in der DB	kein Name wird eingegeben	1a	Alle Kunden werden angezeigt

2.18 Testfälle zu UC-02-03 (Gruppe zuordnen)

2.18.1 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ], []

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Gruppe muss in der DB sein	gültiger Name wird eingegeben: Gruppe 01	1a	Verein wird angezeigt
2	NF	Gruppe muss in der DB sein	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	NF	Gruppe muss in der DB sein	kein Name wird eingegeben	1c	Fehlermeldung erscheint
4	NF	Gruppe ist nicht in der DB	gültiger Name wird eingegeben	1a	Fehlermeldung erscheint

2.19 Testfälle zu UC-02-04 (Kunden inaktiv setzen)

2.19.1 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ], [-], []

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Kunde muss in der DB sein	gültiger Name wird eingegeben: Kunde01	1a	Kunde wird inaktiv gesetzt
2	NF	Kunde muss in der DB sein	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	NF	Kunde muss in der DB sein	kein Name wird eingegeben	1c	Fehlermeldung erscheint
4	NF	Kunde ist nicht in der DB	gültiger Name wird eingegeben	1a	Fehlermeldung erscheint

2.20 Testfälle zu UC-02-05 (Guthaben aufladen)

2.20.1 Testfall: Wert

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [.]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

2a) gültig: [.] einmal

2b) ungültig: [.] öfters

2c) gültig: [.] gar nicht

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Kunde muss in der DB sein	Wert gültig: 1550	1a, 2a	Guthaben wird verändert
2	FF	Kunde muss in der DB sein	Wert ungültig: 15a,30	1b	Fehlermeldung erscheint
3	FF	Kunde muss in der DB sein	Wert ungültig: 15.5.0	2b	Fehlermeldung erscheint
4	FF	Kunde muss in der DB sein	kein Wert eingegeben	1c	Fehlermeldung erscheint
5	FF	Kunde ist nicht in der DB	Wert gültig	1a	Fehlermeldung erscheint

2.21 Testfälle zu UC-02-06 (Kundendaten exportieren)

2.21.1 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ], []

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Kunde muss in der DB sein	gültiger Name wird eingegeben: Tenezkunde 007	1a	Kundendaten werden exportiert
2	FF	Kunde muss in der DB sein	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	FF	Kunde muss in der DB sein	kein Name wird eingegeben	1c	Fehlermeldung erscheint
4	FF	Kunde ist nicht in der DB	gültiger Name wird eingegeben	1a	Fehlermeldung erscheint

2.22 Testfälle zu UC-02-07 (Newsletter verschicken)

2.22.1 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ], []

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Kunde muss in der DB sein	gültiger Name wird eingegeben: Kunde 01	1a	Newsletter wird verschickt
2	FF	Kunde muss in der DB sein	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	FF	Kunde muss in der DB sein	kein Name wird eingegeben	1c	Fehlermeldung erscheint
4	FF	Kunde ist nicht in der DB	gültiger Name wird eingegeben	1a	Fehlermeldung erscheint

2.23 Testfälle zu UC-02-08 (Kunden registrieren)

2.23.1 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ],[]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben	gültiger Name wird eingegeben: manuel007	1a	Verein wird in DB gespeichert
2	FF	Alle anderen Daten korrekt eingegeben	ungültiger Name: code	1b	Fehlermeldung erscheint
3	FF	Alle anderen Daten korrekt eingegeben	kein Name wird eingegeben	1c	Fehlermeldung erscheint
5	FF	Alle anderen Daten korrekt eingegeben aber Name schon in DB	gültiger Name	1a	Fehlermeldung erscheint

2.23.2 Testfall: Adresse

Äquivalenzklassen

1a) gültig: Jedes Zeichen

1b) ungültig: Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben	gültige Adresse wird eingegeben: Astraße 150/50	1a	Kunde wird in DB gespeichert
2	FF	Alle anderen Daten korrekt eingegeben	Code	1b	Fehlermeldung erscheint

2.23.3 Testfall: Telefonnummer

Äquivalenzklassen

1a) gültig: M1 = natürliche Zahlen, [+], [/], [Aa-zZ],[] 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben	gültige Telefonnummer wird eingegeben: +43 0676/123456 privat	1a	Kunde wird in DB gespeichert
2	FF	Alle anderen Daten korrekt eingegeben	ungültige Telefonnummer: 676 «>"123456	1b	Fehlermeldung erscheint

2.23.4 Testfall: E-mail

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ], [],[@],[/],[],[+]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

- 2a) gültig: [@] kommt exakt einmal vor
 2b) ungültig: [@] kommt öfter als 1 mal vor
 2c) ungültig: [@] kommt gar nicht vor

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige E-Mail Adresse wird eingegeben: tenez-reservierung+24@gmx.at	1a, 2a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: ten«>()"@gmX.at	1b	Fehlermeldung erscheint
3	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: tenez@@gmX.at	2b	Fehlermeldung erscheint
4	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: tenez.gmx.at	2c	Fehlermeldung erscheint

2.24 Testfälle zu UC-02-09 (Kunden anmelden)

2.24.1 Testfall: Name

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ],[]
 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes
 1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben und Kunde in DB	gültiger Name wird eingegeben: manuel007	1a	Kunde meldet sich erfolgreich an
2	FF	Alle anderen Daten korrekt eingegeben und Kunde in DB	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	FF	Alle anderen Daten korrekt eingegeben und Kunde in DB	kein Name wird eingegeben	1c	Fehlermeldung erscheint
4	FF	Alle anderen Daten korrekt eingegeben aber Kunde nicht in DB	gültiger Name	1a	Fehlermeldung erscheint

2.24.2 Testfall: Passwort

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

2a) gültig: genau 5 beliebige Zeichen aus M1

2b) ungültig: mehr/weniger als 5 Zeichen aus M1

2c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben und Kunde in DB	gültiges Passwort wird eingegeben: A1B2C	1a, 2a	Kunde wird erfolgreich angemeldet
2	FF	Alle anderen Daten korrekt eingegeben und Kunde in DB	ungültiges Passwort: A"1<>"D4E5F6G7H	1b	Fehlermeldung erscheint
3	FF	Alle anderen Daten korrekt eingegeben und Kunde in DB	ungültiges Passwort wird eingegeben: [>15 Zeichen]	2b	Fehlermeldung erscheint
4	FF	Alle anderen Daten korrekt eingegeben und Kunde in DB	ungültiges Passwort wird eingegeben: [<15 Zeichen]	2b	Fehlermeldung erscheint
5	FF	Alle anderen Daten korrekt eingegeben und Kunde in DB	kein Passwort wird eingegeben	2c	Fehlermeldung erscheint
6	FF	Alle anderen Daten korrekt eingegeben aber Kunde nicht in DB	gültiges Passwort	1a, 2a	Fehlermeldung erscheint

2.25 Testfälle zu UC-02-10 (Guthaben abfragen)

2.25.1 Testfall: Name

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ],[]
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes
- 1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben und Kunde in DB	gültiger Name wird eingegeben: manuel007	1a	Guthaben wird gezeigt
2	FF	Alle anderen Daten korrekt eingegeben und Kunde in DB	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	FF	Alle anderen Daten korrekt eingegeben und Kunde in DB	kein Name wird eingegeben	1c	Fehlermeldung erscheint
4	FF	Alle anderen Daten korrekt eingegeben aber Kunde nicht in DB	gültiger Name	1a	Fehlermeldung erscheint

2.26 Testfälle zu UC-02-11(Buchungshistory anzeigen)

2.26.1 Testfall: Name

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ],[]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben und Kunde in DB	gültiger Name wird eingegeben: manuel007	1a	Buchungshistory wird angezeigt
2	FF	Alle anderen Daten korrekt eingegeben und Kunde in DB	ungültiger Name: Code	1b	Fehlermeldung erscheint
3	FF	Alle anderen Daten korrekt eingegeben und Kunde in DB	kein Name wird eingegeben	1c	Fehlermeldung erscheint
4	FF	Alle anderen Daten korrekt eingegeben aber Kunde nicht in DB	gültiger Name	1a	Fehlermeldung erscheint

2.27 Testfälle zu UC-02-12 (Platz reservieren)

2.27.1 Testfall: Guthaben

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [·],[·]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

2a) gültig: [·] oder [·] einmal

2b) ungültig: [·] oder [·] öfters

2c) gültig: [·] oder [·] gar nicht

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben	Guthaben gültig: 1550	1a, 2a, 3a	Guthaben wird abgebucht
2	FF	Alle anderen Daten korrekt eingegeben	Guthaben ungültig: 15a,30	1b	Fehlermeldung erscheint
3	FF	Alle anderen Daten korrekt eingegeben	Guthaben ungültig: 15.5.0	2b	Fehlermeldung erscheint
Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Freie Slots in der Stundenliste wurden markiert und Kunde hat genug Guthaben	Kunde bestätigt die Reservierung		Reservierung wird erfolgreich durchgeführt
2	FF	Freie Slots in der Stundenliste wurden markiert und Kunde hat zu wenig Guthaben	Kunde bestätigt die Reservierung		Fehlermeldung erscheint

2.28 Testfälle zu UC-02-13(Daten bearbeiten)

2.28.1 Testfall: Name

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ],[]
- 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes
- 1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben	gültiger Name wird eingegeben: manuel007	1a	Verein wird in DB gespeichert
2	FF	Alle anderen Daten korrekt eingegeben	ungültiger Name: code	1b	Fehlermeldung erscheint
3	FF	Alle anderen Daten korrekt eingegeben	kein Name wird eingegeben	1c	Fehlermeldung erscheint
5	FF	Alle anderen Daten korrekt eingegeben aber Name schon in DB	gültiger Name	1a	Fehlermeldung erscheint

2.28.2 Testfall: Adresse

Äquivalenzklassen

- 1a) gültig: Jedes Zeichen
- 1b) ungültig: Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben	gültige Adresse wird eingegeben: Astraße 150/50	1a	Kunde wird in DB gespeichert
2	FF	Alle anderen Daten korrekt eingegeben	Code	1b	Fehlermeldung erscheint

2.28.3 Testfall: Telefonnummer

Äquivalenzklassen

1a) gültig: M1 = natürliche Zahlen, [+], [/], [Aa-zZ],[] 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben	gültige Telefonnummer wird eingegeben: +43 0676/123456 privat	1a	Kunde wird in DB gespeichert
2	FF	Alle anderen Daten korrekt eingegeben	ungültige Telefonnummer: 676«>"123456	1b	Fehlermeldung erscheint

2.28.4 Testfall: E-mail

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ], [],[@],[/],[],[+]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

2a) gültig: [@] kommt exakt einmal vor

2b) ungültig: [@] kommt öfter als 1 mal vor

2c) ungültig: [@] kommt gar nicht vor

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Vereinsdaten korrekt eingegeben	gültige E-Mail Adresse wird eingegeben: tenez-reservierung+24@gmx.at	1a, 2a	Verein wird in DB gespeichert
2	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: ten«>()"@gmx.at	1b	Fehlermeldung erscheint
3	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: tenez@@gmx.at	2b	Fehlermeldung erscheint
4	FF	Alle anderen Vereinsdaten korrekt eingegeben	ungültige E-Mail Adresse wird eingegeben: tenez.gmx.at	2c	Fehlermeldung erscheint

2.29 Testfälle zu UC-02-15(Admin kontaktieren)

2.29.1 Testfall: Text

2.30 Testfälle zu UC-02-16(Kundendaten importieren)

2.30.1 Testfall: Kundendaten Anzahl

Äquivalenzklassen

1a) gültig: 14 [;]

1b) ungültig: < 14 [;] <

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten in der CSV-Datei sind korrekt	Jede Kundenzeile beinhaltet 14 [;]	1a	Daten werden erfolgreich importiert
2	FF	Alle anderen Daten in der CSV-Datei sind korrekt	Eine Kundenzeile beinhaltet weniger als 14 [;]	1b	Fehlermeldung erscheint
3	FF	Alle anderen Daten in der CSV-Datei sind korrekt	Eine Kundenzeile beinhaltet mehr als 14 [;]	1b	Fehlermeldung erscheint

2.30.2 Testfall: E-Mail

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ], [],[@],[/],[],[+]
1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes
- 2a) gültig: [@] kommt exakt einmal vor
2b) ungültig: [@] kommt öfter als 1 mal vor
2c) ungültig: [@] kommt gar nicht vor

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten korrekt eingegeben	nur gültige E-Mail Adressen in der CSV-Datei: tenez-reservierung+24@gmx.at	1a, 2a, 3a	Daten werden erfolgreich importiert
2	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	eine ungültige E-Mail Adresse in der CSV-Datei: ten«>()"@gm.x.at	1b	Fehlermeldung erscheint
3	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	eine ungültige E-Mail Adresse in der CSV-Datei: tenez@@gm.x.at	2b	Fehlermeldung erscheint
4	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	eine ungültige E-Mail Adresse in der CSV-Datei: tenez.gm.x.at	2c	Fehlermeldung erscheint

2.30.3 Testfall: Url

Äquivalenzklassen

- 1a) gültig: M1= natürliche Zahlen, [Aa-zZ], [-], [/], [.]
1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

- 3a) gültig: "http://"muss am Anfang stehen
3b) ungültig: "http://steht nicht am Anfang

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	nur gültige URL Adressen in der CSV-Datei: http://Tennis1210.at	1a, 2a, 3a	Daten werden erfolgreich importiert
2	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	eine ungültige URL in der CSV-Datei: www.Tennis@1210«>.at	1b, 3b	Fehlermeldung erscheint

2.30.4 Testfall: Telefonnummer

Äquivalenzklassen

- 1a) gültig: M1 = natürliche Zahlen, [+], [/], [Aa-zZ],[] 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	nur gültige Telefonnummer in der CSV-Datei: +43 0676/123456 privat	1a, 2a	Daten werden erfolgreich importiert
2	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	eine ungültige Telefonnummer in der CSV-Datei: 676«>"123456	1b	Fehlermeldung erscheint

2.30.5 Testfall: Faxnummer

Äquivalenzklassen

1a) gültig: M1 = natürliche Zahlen, [+], [/], [Aa-zZ],[] 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	nur gültige Faxnummer in der CSV-Datei: +43 0676/123456 privat	1a, 2a	Daten werden erfolgreich importiert
2	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	eine ungültige Faxnummer in der CSV-Datei: 676«>"123456	1b	Fehlermeldung erscheint

2.30.6 Testfall: Stadt

Äquivalenzklassen

1a) gültig: M1 = natürliche Zahlen 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

2a) gültig: genau 4 beliebige Zeichen aus M1

2b) ungültig: mehr als 4 Zeichen aus M1

2c) ungültig: weniger als 4 Zeichen aus M1

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	nur gültige ZIP-Codes in der CSV-Datei: 1010	1a, 2a	Daten werden erfolgreich importiert
2	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	ein ungültiger ZIP-Code in der CSV-Datei: rred	1b	Fehlermeldung erscheint
3	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	ein ungültiger ZIP-Code in der CSV-Datei: 12345	2b	Fehlermeldung erscheint
4	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	ein ungültiger ZIP-Code in der CSV-Datei: 123	2c	Fehlermeldung erscheint
5	FF	Land zu einer Stadt fehlt	nur gültige ZIP-Codes in der CSV-Datei: 1010	1a, 2a	Fehlermeldung erscheint
6	FF	falsches Land	nur gültige ZIP-Codes in der CSV-Datei: 1010	1a, 2a	Fehlermeldung erscheint

2.30.7 Testfall: Land

Äquivalenzklassen

1a) gültig: M1 = Alle Landcodes die in der Datenbank vorhanden sind 1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

2a) gültig: genau 4 beliebige Zeichen aus M1

2b) ungültig: mehr als 1 Code aus M1

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	nur gültige Länder in der CSV-Datei: AUT	1a, 2a	Daten werden erfolgreich importiert
2	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	ein ungültiges Land in der CSV-Datei: rred	1b	Fehlermeldung erscheint
3	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	ein ungültiges Land in der CSV-Datei: AUTDEU	2b	Fehlermeldung erscheint
4	FF	Stadt zu einem Land fehlt	nur gültige Länder in der CSV-Datei: AUT	1a, 2a	Fehlermeldung erscheint
5	FF	ungültige Stadt für ein Land	nur gültige Länder in der CSV-Datei: AUT	1a, 2a	Fehlermeldung erscheint

2.30.8 Testfall: Benutzername

Äquivalenzklassen

1a) gültig: M1= natürliche Zahlen, [Aa-zZ],[]

1b) ungültig: Zeichen nicht aus M1, Ebene eines ganzen Codes

1c) ungültig: keine Zeichen

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	nur gültige Namen in der CSV-Datei: manuel007	1a	Daten werden erfolgreich importiert
2	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	ein ungültiger Name in der CSV-Datei: manuel 0«»7	1b	Fehlermeldung erscheint
3	FF	Struktur und alle anderen Daten in der CSV-Datei sind korrekt	kein Name in der CSV-Datei	1c	Fehlermeldung erscheint
5	FF	Benutzername existiert bereits	gültiger Name	1a	Fehlermeldung erscheint

2.30.9 Testfall: Kopfzeile

Äquivalenzklassen

1a) gültig: M1= 'username','title','firstname','lastname','company','iscompany','street','street2','country','city','phone','fax','email','url','isprepaid'

1b) ungültig: Zeichen nicht aus M1 und nicht in dieser Reihenfolge, Ebene eines ganzen Codes

1c) ungültig: weniger oder mehr als 15 Zeichen aus M1

Lfd Nr.	Typ	Vorbedingungen	Beschreibung Testfall	Äquivalenzklassen	Erwartetes Ergebnis
1	NF	Alle anderen Daten in der CSV-Datei sind korrekt	gültige Kopfzeile	1a	Daten werden erfolgreich importiert
2	FF	Alle anderen Daten in der CSV-Datei sind korrekt	ungültige Kopfzeile in der CSV-Datei	1b,1c	Fehlermeldung erscheint

2.31 Version

Version 1.2 Manuel Djalili - 20.01.2010 - verbessert, erweitert
Version 1.1 Alexander Bohn - 20.01.2010 - verbessert, erweitert
Version 1.0 Alexander Bohn - 08.01.2010 - erweitert
Version 0.9 Alexander Bohn - 04.01.2010 - erweitert
Version 0.8 Manuel Djalili - 08.12.2009 - erweitert
Version 0.7 Manuel Djalili - 23.11.2009 - erweitert
Version 0.6 Manuel Djalili - 21.11.2009 - ausgebessert
Version 0.5 Manuel Djalili - 20.11.2009 - erweitert
Version 0.4 Alexander Bohn - 17.11.2009 - Gesamtes Doc querformat
Version 0.3 Manuel Djalili - 14.11.2009 - ausgebessert und erweitert
Version 0.2 Manuel Djalili - 13.11.2009 - erweitert
Version 0.1 Alexander Bohn - 13.11.2009 - Anfang