



TECHNISCHE  
UNIVERSITÄT  
WIEN

Vienna University of Technology

---

Software Engineering & Projektmanagement, PR 6.0/4.0  
Sommersemester 2015

Einzelbeispiel - Wendy's Rennpferde  
Projektdokumentation

---

Enri Miho

0929003

e0929003@student.tuwien.ac.at

22.03.2015

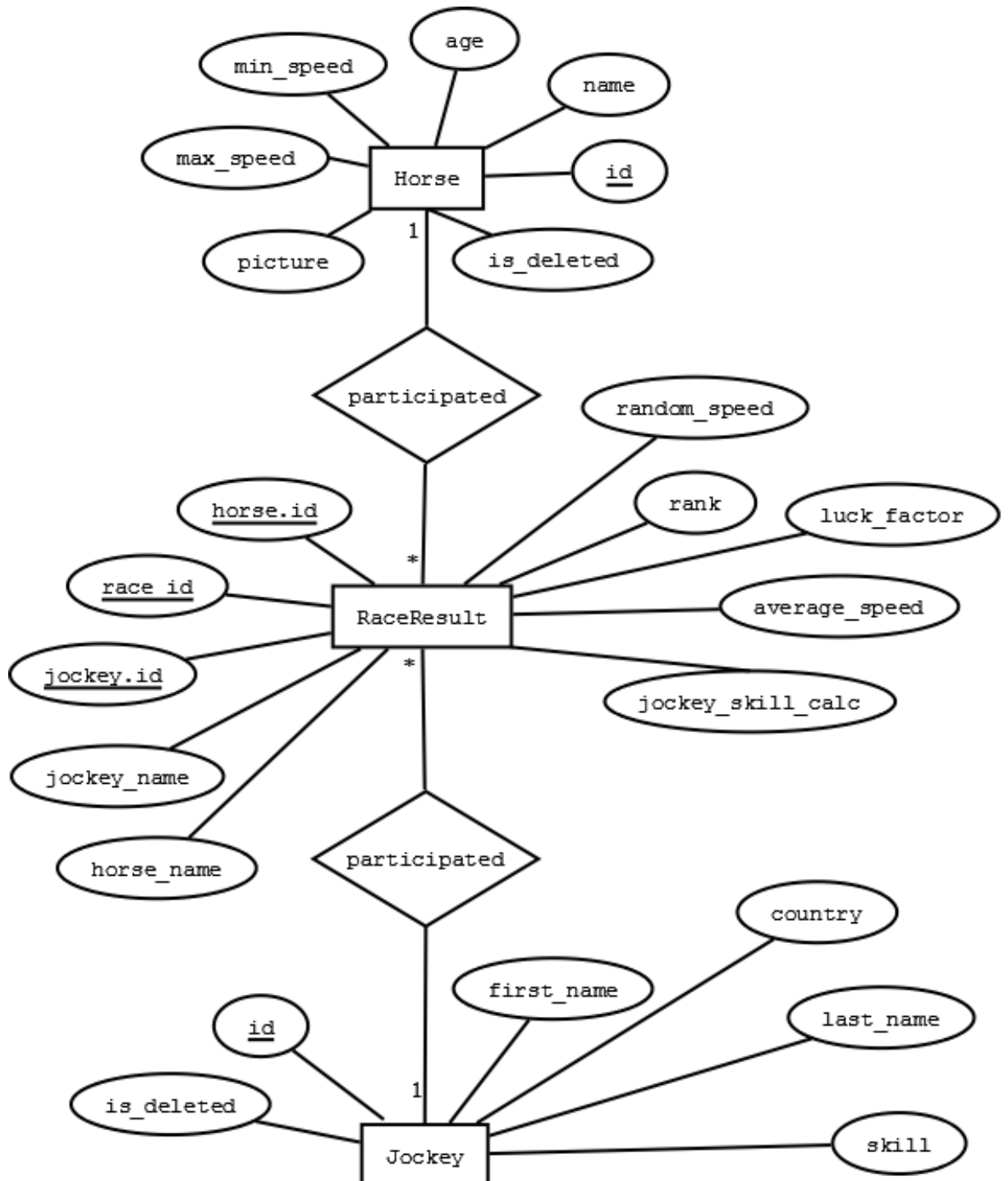
# 1 Stundenliste

Datum	Beschreibung	Aufwand
08.03.2015	Festlegen des Domänenmodells und Erstellung mittels DIA.	1.5
08.03.2015	Datenmodell festlegen und ER-Diagramm erstellen mittels DIA.	1
08.03.2015	database.sql erstellt. Abspeichern der Daten in die H2 Datenbank.	2
10.03.2015	Klassendiagramm erstellen mittels DIA. (Modellierung der DAO-Schicht).	2
10.03.2015	Einrichten von IntelliJ.	0.5
11.03.2015	Datenbankverbindung mit der Singleton-Klasse JDBCSingletonConnection erstellen.	1
11.03.2015	Erstellen der 3 Entities Horse, Jockey & RaceResult.	0.5
12.03.2015	DAO-Interfaces erstellen(mit Javadoc Kommentaren).	3
13.03.2015	JDBCHorseDAO implementiert. Junit Tests erstellt.	6
14.03.2015	JDBCJockeyDAO implementiert. Junit Tests erstellt.	4
15.03.2015	JDBCRaceResultDAO implementiert. Junit Tests erstellt.	5
18.03.2015	Überlegen und Erstellen des Anwendungsfalldiagramms.	3
19.03.2015	Erstellen der Anwendungsfallbeschreibungen.	3
20.03.2015	Überlegen und erweitern des Klassendiagramms um die Service-Schicht.	3
20.03.2015	Service-Interface: erstellen der Methoden(mit Javadoc Kommentaren).	2
22.03.2015	Implementierung der Methoden in SimpleService. JUnit Tests erstellt.	9
		46.5

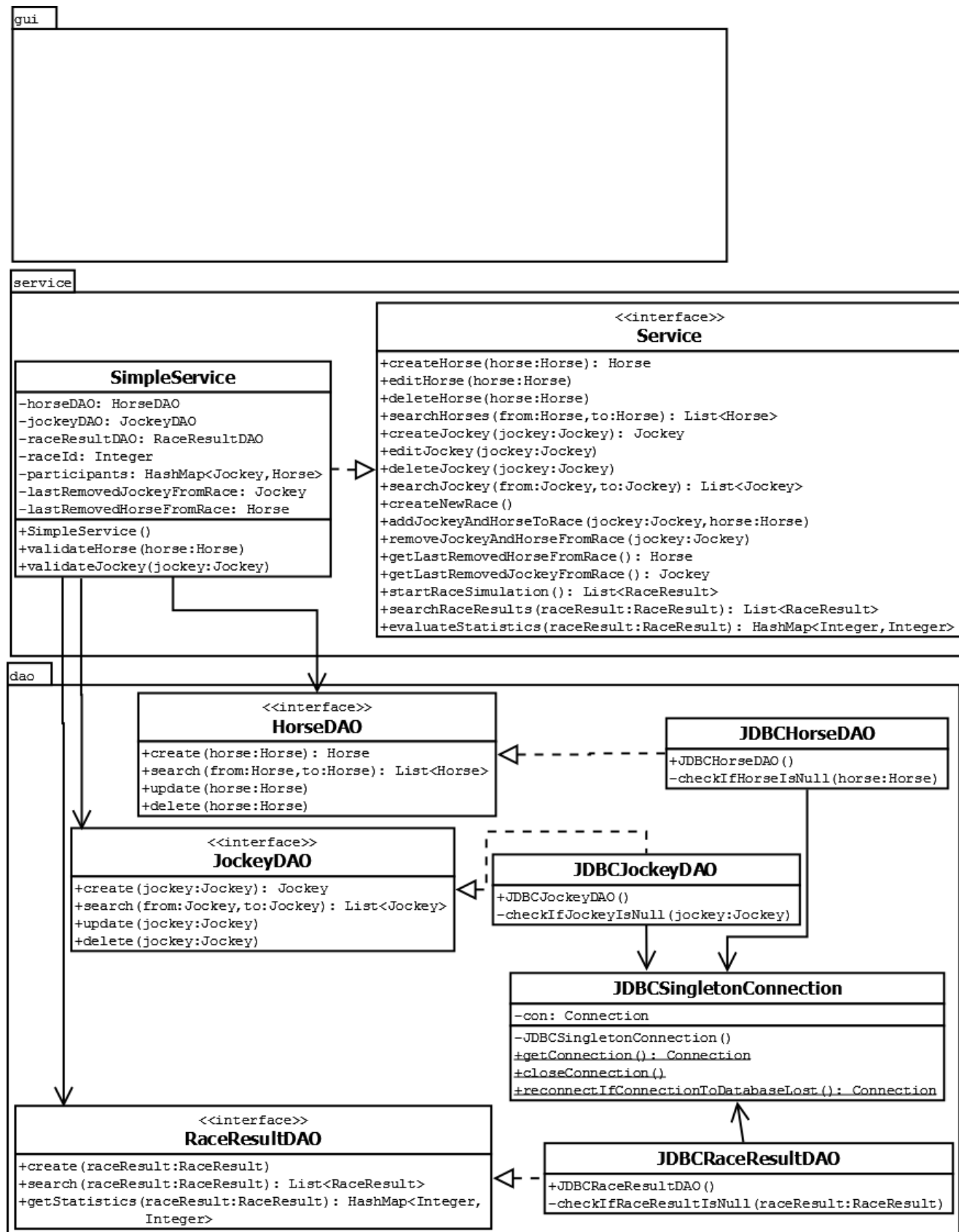
## 2 Domänenmodell



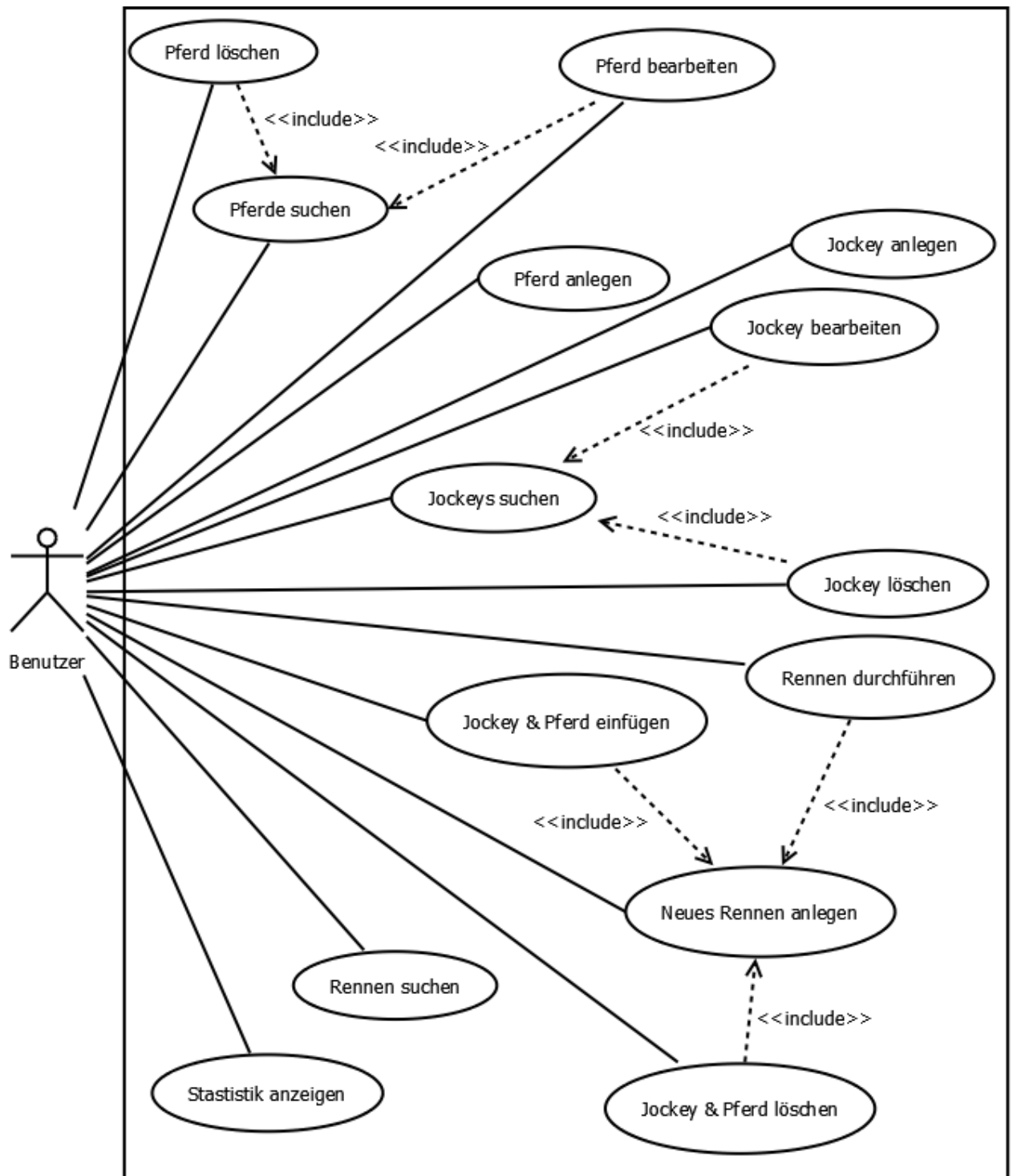
## 3 Entity-Relationship-Diagramm



## 4 Klassendiagramm



## 5 Anwendungsfalldiagramm



## 6 Anwendungsfallbeschreibungen

### 6.1 Pferd anlegen

#### 1. Identification summary

- Titel: Pferd anlegen
- Scope: Pferd
- Level: User Goal
- Akteure: User
- Kurzbeschreibung: Ein neues Pferd wird angelegt.

#### 2. Scenarios

- Vorbedingungen: Es besteht eine Datenbankverbindung.
- Hauptszenario: Der User klickt auf "New...", (neues Fenster erscheint) und gibt dann Name, Alter, minimale und maximale Geschwindigkeit und Bild ein und klickt anschließend auf "Ok" um ein neues Pferd abzuspeichern. Das System gibt eine Meldung zurück über diesen Vorgang (Dialogfenster wird angezeigt).
- Fehlerszenario:
  - Es besteht keine Verbindung zu der Datenbank. Es wird noch einmal versucht eine Verbindung aufzubauen, ist diese erfolglos, wird eine Fehlermeldung angezeigt.
  - Alter, minimale und maximale Geschwindigkeit sind negativ oder 0 oder zu groß oder leer, Name ist ungültig, der angegebene Pfad zum Bild existiert nicht. Entsprechende Fehlermeldungen werden angezeigt.
- Alternativszenario: Der User unterbricht den Vorgang indem er(im neuerscheintem Fenster) auf "Cancel" klickt oder das Fenster schließt.
- Nachbedingungen: Das neue Pferd kann jetzt im System gefunden werden.

#### 3. Non-functional constraints: Keine

### 6.2 Pferd bearbeiten

#### 1. Identification summary

- Titel: Pferd bearbeiten
- Scope: Pferd
- Level: User Goal
- Akteure: User
- Kurzbeschreibung: Ein existierendes Pferd wird bearbeitet.

#### 2. Scenarios

- Vorbedingungen:
  - Es besteht eine Datenbankverbindung.
  - Das Pferd das man bearbeiten möchte wurde im System gefunden(anderer Anwendungsfall).
- Hauptszenario: Der User klickt mit der rechten Maustaste auf das gefundene Pferd, klickt auf "Edit", (neues Fenster erscheint) und gibt dann Name, Alter, minimale und maximale Geschwindigkeit und Bild ein und klickt anschließend auf "Ok" um dieses zu verändern. Das System gibt eine Meldung zurück über diesen Vorgang (Dialogfenster wird angezeigt).
- Fehlerszenario:

- Es besteht keine Verbindung zu der Datenbank. Es wird noch einmal versucht eine Verbindung aufzubauen, ist diese erfolglos, wird eine Fehlermeldung angezeigt.
- Alter, minimale und maximale Geschwindigkeit sind negativ oder 0 oder zu groß oder leer, Name ist ungültig, der angegebene Pfad zum Bild existiert nicht. Entsprechende Fehlermeldungen werden angezeigt.
- Alternativszenario: Der User unterbricht den Vorgang indem er(im neuerscheintem Fenster) auf "Cancel" klickt oder das Fenster schließt.
- Nachbedingungen: Das veränderte Pferd kann jetzt im System gesucht und gefunden werden.

3. Non-functional constraints: Keine

## 6.3 Pferd löschen

### 1. Identification summary

- Titel: Pferd löschen
- Scope: Pferd
- Level: User Goal
- Akteure: User
- Kurzbeschreibung: Ein existierendes Pferd wird gelöscht.

### 2. Scenarios

- Vorbedingungen:
  - Es besteht eine Datenbankverbindung.
  - Das Pferd das man löschen möchte wurde im System gefunden(anderer Anwendungsfall).
- Hauptszenario: Der User klickt mit der rechten Maustaste auf das gefundene Pferd, klickt auf "Delete", und wird durch ein Dialogfenster gefragt ob er dieses Pferd wirklich löschen möchte.
- Fehlerszenario: Es besteht keine Verbindung zu der Datenbank. Es wird noch einmal versucht eine Verbindung aufzubauen, ist diese erfolglos, wird eine Fehlermeldung angezeigt.
- Alternativszenario: Der User unterbricht den Vorgang indem er(im Dialogfenster) auf "Cancel" klickt oder das Fenster schließt.
- Nachbedingungen: Dieses Pferd kann nicht mehr im System gefunden werden.

3. Non-functional constraints: Keine

## 6.4 Pferde Suchen

### 1. Identification summary

- Titel: Pferde Suchen
- Scope: Pferd
- Level: User Goal
- Akteure: User
- Kurzbeschreibung: Der User sucht nach einem bestimmten Pferd.

### 2. Scenarios

- Vorbedingungen: Es besteht eine Datenbankverbindung.

- Hauptszenario: Der User schränkt die Liste der Pferde durch Alter, minimale und maximale Geschwindigkeit ein. Er klickt anschließend auf "Search".
- Fehlerszenario:
  - Es besteht keine Verbindung zu der Datenbank. Es wird noch einmal versucht eine Verbindung aufzubauen, ist diese erfolglos, wird eine Fehlermeldung angezeigt.
  - Alter, minimale und maximale Geschwindigkeit sind negativ oder 0 oder zu groß. Entsprechende Fehlermeldungen werden angezeigt.
- Alternativszenario: Das System kann kein Pferd zu den angegebenen Kriterien finden, es wird eine Meldung durch ein Dialogfenster angezeigt.
- Nachbedingungen: Pferd kann bearbeitet oder gelöscht werden(anderer Anwendungsfall).

3. Non-functional constraints: Keine

## 6.5 Jockey anlegen

1. Identification summary

- Titel: Jockey anlegen
- Scope: Jockey
- Level: User Goal
- Akteure: User
- Kurzbeschreibung: Ein neuer Jockey wird angelegt.

2. Scenarios

- Vorbedingungen: Es besteht eine Datenbankverbindung.
- Hauptszenario: Der User klickt auf "New...", (neues Fenster erscheint) und gibt dann Vorname, Nachname, Land, und Können ein und klickt anschließend auf "Ok" um einen neuen Jockey abzuspeichern. Das System gibt eine Meldung zurück über diesen Vorgang (Dialogfenster wird angezeigt).
- Fehlerszenario:
  - Es besteht keine Verbindung zu der Datenbank. Es wird noch einmal versucht eine Verbindung aufzubauen, ist diese erfolglos, wird eine Fehlermeldung angezeigt.
  - Vorname, Nachname oder Land sind ungültig, Können ist leer. Entsprechende Fehlermeldungen werden angezeigt.
- Alternativszenario: Der User unterbricht den Vorgang indem er(im neuerscheintem Fenster) auf "Cancel" klickt oder das Fenster schließt.
- Nachbedingungen: Der neue Jockey kann jetzt im System gefunden werden.

3. Non-functional constraints: Keine

## 6.6 Jockey bearbeiten

1. Identification summary

- Titel: Jockey bearbeiten
- Scope: Jockey
- Level: User Goal
- Akteure: User
- Kurzbeschreibung: Ein existierender Jockey wird bearbeitet.

2. Scenarios



- Vorbedingungen:
  - Es besteht eine Datenbankverbindung.
  - Der Jockey den man bearbeiten möchte wurde im System gefunden(anderer Anwendungsfall).
- Hauptszenario: Der User klickt mit der rechten Maustaste auf den gefundenen Jockey, klickt auf "Edit", (neues Fenster erscheint) und gibt dann Vorname, Nachname, Land und Können ein und klickt anschließend auf "Ok" um diesen zu verändern. Das System gibt eine Meldung zurück über diesen Vorgang (Dialogfenster wird angezeigt).
- Fehlerszenario:
  - Es besteht keine Verbindung zu der Datenbank. Es wird noch einmal versucht eine Verbindung aufzubauen, ist diese erfolglos, wird eine Fehlermeldung angezeigt.
  - Vorname, Nachname oder Land sind ungültig, Können ist leer. Entsprechende Fehlermeldungen werden angezeigt.
- Alternativszenario: Der User unterbricht den Vorgang indem er(im neuerscheintem Fenster) auf "Cancel" klickt oder das Fenster schließt.
- Nachbedingungen: Der veränderte Jockey kann jetzt im System gesucht und gefunden werden.

3. Non-functional constraints: Keine

## 6.7 Jockey löschen

1. Identification summary

- Titel: Jockey löschen
- Scope: Jockey
- Level: User Goal
- Aktoren: User
- Kurzbeschreibung: Ein existierender Jockey wird gelöscht.

2. Scenarios

- Vorbedingungen:
  - Es besteht eine Datenbankverbindung.
  - Der Jockey den man löschen möchte wurde im System gefunden(anderer Anwendungsfall).
- Hauptszenario: Der User klickt mit der rechten Maustaste auf den gefundenen Jockey, klickt auf "Delete", und wird durch ein Dialogfenster gefragt ob er diesen Jockey wirklich löschen möchte.
- Fehlerszenario: Es besteht keine Verbindung zu der Datenbank. Es wird noch einmal versucht eine Verbindung aufzubauen, ist diese erfolglos, wird eine Fehlermeldung angezeigt.
- Alternativszenario: Der User unterbricht den Vorgang indem er(im Dialogfenster) auf "Cancel" klickt oder das Fenster schließt.
- Nachbedingungen: Dieser Jockey kann nicht mehr im System gefunden werden.

3. Non-functional constraints: Keine

## 6.8 Jockeys suchen

### 1. Identification summary

- Titel: Jockeys Suchen
- Scope: Jockey
- Level: User Goal
- Akteure: User
- Kurzbeschreibung: Der User sucht nach einem bestimmten Jockey.

### 2. Scenarios

- Vorbedingungen: Es besteht eine Datenbankverbindung.
- Hauptszenario: Der User schränkt die Liste der Jockeys durch das Können ein. Er klickt anschließend auf "Search".
- Fehlerszenario:
  - Es besteht keine Verbindung zu der Datenbank. Es wird noch einmal versucht eine Verbindung aufzubauen, ist diese erfolglos, wird eine Fehlermeldung angezeigt.
  - Das Können ist leer. Eine entsprechende Fehlermeldung wird angezeigt.
- Alternativszenario: Das System kann keinen Jockey zu den angegebenen Kriterien finden, es wird eine Meldung durch ein Dialogfenster angezeigt.
- Nachbedingungen: Jockey kann bearbeitet oder gelöscht werden(anderer Anwendungsfall).

### 3. Non-functional constraints: Keine

## 6.9 Neues Rennen anlegen

### 1. Identification summary

- Titel: Neues Rennen anlegen
- Scope: Rennen
- Level: User Goal
- Akteure: User
- Kurzbeschreibung: Es wird ein neues Rennen angelegt.

### 2. Scenarios

- Vorbedingungen:
  - Es besteht eine Datenbankverbindung.
  - Es gibt mindestens ein Pferd und einen Jockey im System.
- Hauptszenario: Der User klickt auf "New race..." und ein neues Fenster wird angezeigt. In diesem gibt es 3 Tabellen: eine die die Teilnehmer am Rennen anzeigt, eine die die Jockeys anzeigt und eine die die Pferde anzeigt.
- Fehlerszenario: Es besteht keine Verbindung zu der Datenbank. Es wird noch einmal versucht eine Verbindung aufzubauen, ist diese erfolglos, wird eine Fehlermeldung angezeigt.
- Alternativszenario: Der User unterbricht den Vorgang, löscht also das ganze Rennen, indem er auf "Cancel" klickt oder das Fenster schließt.
- Nachbedingungen: Der User hat die Möglichkeit neue Jockey-Pferd Kombinationen auszuwählen und ins Rennen einzufügen, sowie diese auch wieder zu entfernen, außerdem kann er nun ein Rennen durchführen(andere Anwendungsfälle).

### 3. Non-functional constraints: Keine

## 6.10 Jockey & Pferd einfügen

### 1. Identification summary

- Titel: Jockey & Pferd einfügen
- Scope: Rennen
- Level: User Goal
- Aktoren: User
- Kurzbeschreibung: Es wird eine neue Jockey-Pferd Kombination in ein existierendes Rennen eingefügt.

### 2. Scenarios

- Vorbedingungen: Man hat bereits ein Rennen erstellt, also "New race..." angeklickt (anderer Anwendungsfall).
- Hauptszenario: Der User wählt aus den beiden Tabellen Jockey und Pferd aus und fügt sie mit einem Klick auf "Add to race" in das Rennen ein.
- Fehlerszenario: Keines
- Alternativszenario: Keines
- Nachbedingungen: Die ausgewählten Jockey und Pferde sind nicht mehr in den beiden Tabellen zu finden.

### 3. Non-functional constraints: keine

## 6.11 Jockey & Pferd löschen

### 1. Identification summary

- Titel: Jockey & Pferd löschen
- Scope: Rennen
- Level: User Goal
- Aktoren: User
- Kurzbeschreibung: Es wird eine bestehende Jockey-Pferd Kombination aus dem Rennen entfernt.

### 2. Scenarios

- Vorbedingungen: Man hat bereits ein Rennen erstellt, also "New race..." angeklickt (anderer Anwendungsfall).
- Hauptszenario: Der User wählt eine bestehende Jockey-Pferd Kombination aus und klickt mit der rechten Maustaste und entfernt diese Kombination aus dem Rennen.
- Fehlerszenario: Keines
- Alternativszenario: Keines
- Nachbedingungen: Jockey und Pferd die aus der Tabelle entfernt wurden sind jetzt wieder in den Jockey- und Pferde-Tabellen zu finden.

### 3. Non-functional constraints: keine

## 6.12 Rennen durchführen

### 1. Identification summary

- Titel: Rennen durchführen
- Scope: Rennen
- Level: User Goal
- Aktoren: User
- Kurzbeschreibung: Das Rennen wird durchgeführt.

### 2. Scenarios

- Vorbedingungen:
  - Es besteht eine Datenbankverbindung.
  - Es gibt mindestens eine Jockey-Pferd Kombination die am Rennen teilnimmt).
- Hauptszenario: Der User klickt auf "Start race", wird gefragt er wirklich das Rennen durchführen möchte, klickt auf "Yes", und anschließend wird dann durch ein Dialogfenster über das Ergebnis informiert.
- Fehlerszenario: Es besteht keine Verbindung zu der Datenbank. Es wird noch einmal versucht eine Verbindung aufzubauen, ist diese erfolglos, wird eine Fehlermeldung angezeigt.
- Alternativszenario: Der User klickt auf "No" und das Rennen wird nicht durchgeführt.
- Nachbedingungen: Die Ergebnisse dieses Rennens sind jetzt im System vorhanden und können gesucht werden.

### 3. Non-functional constraints: keine

## 6.13 Rennen suchen

### 1. Identification summary

- Titel: Rennen suchen
- Scope: Rennen
- Level: User Goal
- Aktoren: User
- Kurzbeschreibung: Der User schränkt Rennergebnisse ein.

### 2. Scenarios

- Vorbedingungen: Es besteht eine Datenbankverbindung.
- Hauptszenario:
- Fehlerszenario:
  - Es besteht keine Verbindung zu der Datenbank. Es wird noch einmal versucht eine Verbindung aufzubauen, ist diese erfolglos, wird eine Fehlermeldung angezeigt.
  - RennenID oder JockeyID oder PferdID sind nicht gültig. Eine entsprechende Fehlermeldung wird angezeigt.
- Alternativszenario: Das System kann kein Rennen zu den angegebenen Kriterien finden, es wird eine Meldung durch ein Dialogfenster angezeigt.
- Nachbedingungen: keine

### 3. Non-functional constraints: keine

## 6.14 Statistik anzeigen

### 1. Identification summary

- Titel: Statistik anzeigen
- Scope: Rennen
- Level: User Goal
- Aktoren: User
- Kurzbeschreibung: Es werden Statistiken über Rennergebnisse evaluiert.

### 2. Scenarios

- Vorbedingungen: Es besteht eine Datenbankverbindung.
- Hauptszenario:
- Fehlerszenario:
  - Es besteht keine Verbindung zu der Datenbank. Es wird noch einmal versucht eine Verbindung aufzubauen, ist diese erfolglos, wird eine Fehlermeldung angezeigt.
  - JockeyID oder PferdID sind nicht gültig. Eine entsprechende Fehlermeldung wird angezeigt.
- Alternativszenario: Pferd, Jockey oder eine Kombination von diesen haben an keinem Rennen teilgenommen. Eine Meldung wird angezeigt.
- Nachbedingungen: keine

### 3. Non-functional constraints: keine