

Mitfahr-Agentur Professional Textuelle Spezifikation aller Klassen

AKD
Musterstraße 123
10123 Berlin

Erstellt von:

Claudia Adamietz
Robert Kalweit
Dennis Dechau

Ansprechpartner:

Claudia Adamietz
Robert Kalweit
Dennis Dechau

Erreichbar unter:

Tel. (030) 1234567
kalle-ntz@web.de

Stand:

09.01.2006

Textuelle Spezifikation aller Klassen:

Farblich hervorgehobene Klassen sind Containerklassen und enthalten daher ausschließlich Objektattribute, keine Klassenattribute. Die jeweiligen „public void delete()“ und „public void search()“ Methoden dienen dazu, Objekte aus dem Bestand zu löschen bzw. bestimmte Objekte im Bestand zu suchen.

MFApro

public boolean validateLogin()	Überprüft ob das eingegebene Passwort korrekt ist und liefert einen boolean-Wert zurück.
public void performLogin()	Innerhalb dieser Methode wird „validateLogin()“ ausgeführt. Für „true“ wird der Login vollzogen, liefert „validateLogin()“ „false“, wird der Login verweigert.
public void erstelleAccount()	Erstellt mit den Werten der entsprechenden Eingabefelder einen neuen Account. Überprüft dabei die Typ-Korrektheit der Daten (z.B. Eingabe von Buchstaben im Eingabefeld „Postleitzahl“).

Reservierung_Bestand

Reservierung

Attribute:	AngebotsID GesuchID	ID des Angebots für das Plätze reserviert wurden ID des Gesuchs für das Plätze gefunden und reserviert wurden
	Plätze Reservierungsdatum	Anzahl der Plätze, die reserviert wurden Datum der Reservierung

Buchung_Bestand

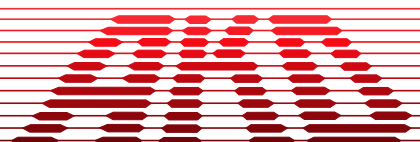
Buchung

Attribute:	AngebotsID GesuchID	ID des Angebots für das Plätze reserviert wurden ID des Gesuchs für das Plätze gefunden und reserviert wurden
	Plätze Reservierungsdatum Buchungsgebühr Bezahlt	Anzahl der Plätze, die reserviert wurden Datum der Reservierung Liefert true/false, ob die Gebühr bezahlt wurde

Fahrt_Bestand

Fahrt

Attribute:	Start Ziel Datum Uhrzeit Plaetze	Abfahrtsort der Fahrt Fahrziel Datum, an dem die Fahrt getätigt werden soll Ungefähre Uhrzeit der Abfahrt Anzahl der freien oder gesuchten Plätze
------------	--	---



Angebot_Bestand

Angebot

public Gesuch[] sucheMitfahrer()

public void speichern()

public void reservierePlaetze(AngebotsID:String, GesuchID:String, Plaetze:int)

public void loescheAngebot()

public void buchen()

Attribute:	AngebotsID	ID des Angebots, für das Plätze reserviert wurden
	KundenID	ID des Kunden, der das Angebot erstellt hat
	Erstellungsdatum	Datum, an dem die Fahrt getätigt werden soll
	Serie	Ungefähre Uhrzeit der Abfahrt

Gesuch_Bestand

Gesuch

public Angebot[] sucheAngebote()

public void speichern()

public void loescheGesuch()

Attribute:	GesuchID	ID des Gesuchs für das Plätze gefunden und reserviert wurden
	KundenID	ID des Kunden, der das Gesuch aufgegeben hat

Account_Bestand

Account

private void logout()

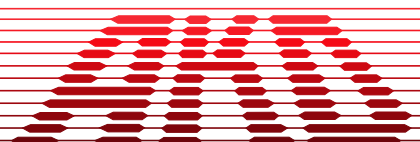
Meldet den jeweiligen Nutzer von seinem Account ab.

Attribute:	KundenID	ID des Kunden, dem der Account zugeordnet ist
	Loginname	Loginname des Kunden, der auch in öffentlich sichtbaren Angeboten genannt wird, zur Wahrung der Anonymität frei wählbar
	Passwort	Passwort des Accounts

Kunde_Bestand

Kunde

Attribute:	KundenID	ID des Kunden, fortlaufende Nummer
	Vorname	
	Name	
	Geburtsdatum	
	Strasse_HNR	Straße und Hausnummer des Kunden
	PLZ	Postleitzahl des Wohnorts des Kunden
	Ort	Wohnort des Kunden



Schwarze_Liste_Bestand

Schwarze_Liste

private void eintragen(KundenID)	Trägt den Kunden mit der angegebenen ID in die Schwarze Liste ein.
private void entfernen(KundenID)	Entfernt den Kunden mit der angegebenen ID von der Schwarzen Liste.
Attribute:	
KundenID	ID des eingetragenen Kunden
Datum	Datum, an dem der Kunde in die Schwarze Liste eingetragen wurde.
Grund	Grund des Eintrags
Bemerkungen	Weitere Bemerkungen

Anbieter

public void erstelleAngebot()	Erstellt mit den Werten der entsprechenden Eingabefelder einen neues Angebot. Überprüft dabei die Typ-Korrektheit der Daten (z.B. Eingabe von Buchstaben im Eingabefeld „Datum“).
-------------------------------	---

Suchender

public void erstelleGesuch()	Erstellt mit den Werten der entsprechenden Eingabefelder einen neues Gesuch. Überprüft dabei die Typ-Korrektheit der Daten (z.B. Eingabe von Buchstaben im Eingabefeld „Datum“).
------------------------------	--

Statistiken

public void erstelleFahrtstatistik()	Erstellt eine Statistik über die aufgegebenen Angebote und Gesuche in variablen Zeiträumen und mit diversen Parametern.
public void erstelleKundenStatistik()	Erstellt eine Statistik über die Entwicklung des Kundenbestands im Zusammenhang mit den getätigten Fahrten etc.
public void erstelleVermittlungsStatistik()	Erstellt eine Statistik über den Zusammenhang der Anzahl der aufgegebenen Fahrten mit den tatsächlich vermittelten Fahrten und die dadurch erhaltenen Buchungsgebühren.

Das erweiterte Referenzglossar, sowie eine Online-Version dieser Spezifikation finden Sie unter
<http://www.tfh-berlin.de/~s17094/se>

