**Package dienstleistungenverwaltung**

**Beschreibung des Packages**

Das Package *dienstleistungenverwaltung* umfasst alle Aspekte zur Erstellung und Bearbeitung, sowie zum Löschen von einzelnen Dienstleistungen. Es enthält sowohl Klassen zur Präsentation aller angelegten Dienstleistungen, sowie Frames zum Hinzufügen bzw. Bearbeiten von Dienstleistungen. Des Weiteren sind Klassen der internen Logik vorhanden.

Da die Daten der Dienstleistungen zum Zeitpunkt der Erstellung dieses Dokuments noch nicht in einer Datenbank gespeichert wurden, existiert noch keine Klasse, welche diese Verknüpfung regelt. Die Klassen zur Verbindung und somit zur Speicherung der Daten in der Datenbank werden im weiteren Entwicklungsverlauf in naher Zukunft hinzugefügt.

**Diagramme**

**TO DO**

**Schnittstellen**

Das Package *dienstleistungenverwaltung* besitzt Schnittstellen zum Package *kassenfunktion*, in welcher die vorhandenen Dienstleistungen zum Verkauf präsentiert werden. Eine weitere Schnittstelle existiert zum Package *account*, welches die Startseite der Software beinhaltet. Auf der Startseite kann der Benutzer dann die Verwaltung der Dienstleistungen aufrufen.

**Aufteilung der Klassen in das Drei-Schichten-Modell:**

GUI-Schicht: DLVerwaltungFrame, DLHinzufügenFrame, DLBearbeitenFrame

Logik-Schicht: Dienstleistung, DLTableModel

Persisitenz-Schicht: (noch keine Klasse in dieser Schicht)

**Operationen**

Dienstleistung:

* Konstruktor: public Dienstleistung(String, double, int)

DLVerwaltungFrame:

* Konstruktor: public DLVerwaltungFrame(List<Dienstleistung>=
* private void addDL()
* private void bearbeiteDL()
* private void deleteDL()

DLTableModel:

* Konstruktor: package DLTableModel(List<Dienstleistung>)
* public String getColumnName(int)
* public int getColumnCount()
* public int getRowCount()
* public Object getValueAt(int,int)
* public Class<?> getColumnClass(int)
* package Dienstleistung getDienstleistung(int)
* package List<Dienstleistung> getDienstleistungen()

DLHinzufuegenFrame:

* Konstruktor: DLHinzufuegenFrame(JFrame, List<Dienstleistung)
* innere Klasse: MyOkHandler
* public void actionPerformed(ActionEvent)
* innere Klasse: MyAbbHandler
* public void actionPerformed(ActionEvent)

DLBearbeitenFrame:

* Konstruktor: DLBearbeitenFrame(JFrame, Dienstleistung)
* innere Klasse: MyOkHandler
* public void actionPerformed(ActionEvent)
* innere Klasse: MyAbbHandler
* public void actionPerformed(ActionEvent)