

Objektorientierte Programmierung in Java

– Teilleistung 3 –

Dokumentation Kantinenplaner

1. Bedienungsanleitung

Der Kantinenplaner kann in automatisierter Form und über eine grafische Oberfläche genutzt werden.

Automatisierte Planung

Konfigurieren Sie zunächst die Software in der "config/planer.cfg". Die Planung wird durch das Starten des Skripts "Start_automatisierte_Planung.bat" durchgeführt und die Ausgabedateien in konfigurierten Ordner erstellt (Default ist „planungsergebnisse“) erzeugt.

```
# Konfigurationsdatei zum Kantinenplaner
# Autor Kim Michael Jansen
# Datum 09.01.2013
# Version 1.4
# Liste der Dateipfade für Preislisten der Händler (relativ
oder absolut)
FILE.PREISLISTE=testdaten/beispieldateien/preisliste_
1.csv,testdaten/beispieldateien/preisliste_
2.csv,testdaten/beispieldateien/preisliste_
3.csv,testdaten/beispieldateien/preisliste_
4.csv,testdaten/beispieldateien/preisliste_
5.csv,testdaten/beispieldateien/preisliste_
6.csv,testdaten/beispieldateien/preisliste_7.csv
# Dateipfad der Rezeptdatei alias "bekannte Gerichte des
Kochs"
FILE.REZEPE=testdaten/beispieldateien/rezepte.csv
# Dateipfad der Hitliste alias "beliebte Gerichtedatei"
FILE.HITLISTE=testdaten/beispieldateien/hitliste.csv
# Liste der Kantinen im Format [(Name Kantine 1).(Anzahl
Mitarbeiter)][,(Name Kantine n).(Anzahl Mitarbeiter)]
# Die Kantinen werden durch Komma (,) getrennt
# Die Parameter eine Kantine werden durch Punkt (.) getrennt
CLASS.KANTINE=Mühlheim.500,Essen.300
# Planungszeitraum, Datumsformat = tt.MM.jjjj
# t = Tag, M = Monat, j = Jahr
PERIODE.STARTDATUM=07.01.2013
PERIODE.ENDEDATUM=27.01.2013
# Die Toleranz in Euro. Bei der Planung werden grundsätzlich
die günstigsten Gerichte bevorzugt.
# Durch Angabe einer Toleranz ist es möglich z.B.
Hitlistengerichte zu bevorzugen,
# wenn die Preisdifferenz die Toleranz nicht überschreitet.
PERIODE.TOLERANZ=0,10
# Verzeichnis für Ausgabedateien (relativ oder absolut)
DIR.OUTPUT=planungsergebnis
# Wahl der Ausgabeformate
# Gültige Werte sind "HTML" und "TXT"
FORMAT.OUTPUT=HTML
```

Abbildung 1: Beispiel einer Konfiguration

1. Konfiguration der Software

- a. Konfigurieren Sie das Programm über die mitgelieferte Konfigurationsdatei "config/planer.cfg". Eine ausführliche Beschreibung der Parameter und den Eingabemöglichkeiten entnehmen Sie bitte direkt den Kommentaren, gekennzeichnet durch "#", der Konfigurationsdatei.

Folgende Eingabedateien müssen angegeben werden:

1. Dateipfade der Preislisten
2. Dateipfad der Rezeptdatei
3. Dateipfad der Hitlistendatei

- b. Folgende Parameter können geändert werden:

- i. Angaben zu Kantinen und Mitarbeitern
- ii. Startdatum des Planungszeitraums
- iii. Endedatum des Planungszeitraums
- iv. Toleranz um beliebtere Gerichte zu bevorzugen
- v. Ausgabeverzeichnis
- vi. Ausgabeformat

2. Ausführen des mitgelieferten Skripts "Start_automatisierte_Planung.bat".

Graphische Oberfläche

Konfigurieren Sie zunächst die Software in der "config/planer.cfg" (s.o.). Die Anwendung wird durch das Starten des Skripts "Start_Benutzeroberflaeche.bat" ausgeführt. In der Oberfläche können Sie Einstellungen ergänzen oder verändern. Die Planung und das Erstellen der Ausgaben werden in der Oberfläche manuell gestartet.

1. Startwerte in Konfigurationsdatei wie in Nr. "1.a.i." beschrieben anpassen. Diese Einstellungen sind optional. Die Werte können in der Oberfläche geändert und/oder ergänzt werden.
2. Ausführen des mitgelieferten Skripts "Start_Benutzeroberflaeche.bat".

2. Funktionsbeschreibung

Zur Bedienung des Programms wurde eine grafische Oberfläche implementiert.

Genauere Funktionsbeschreibung des Programms:

Die Kantinenplanung kann über eine grafische Oberfläche konfiguriert und durchgeführt werden. Es gibt zwei unterschiedliche Fenster die im Folgenden beschrieben werden:

(1) OberflächeMain

Kantinenplaner

Kantine: Mühlheim
 Ausgabezeitraum: Von: 07.01.2013 Bis: 27.01.2013
 Ausgabeordner: planungsergebnis
 Mitarbeiteranzahl: 500
 Toleranz: 0.1
 Ausgabeformat: ☒ HTML ☐ Text

Properties aktualisier...
 Eingabedaten erfassen
 Planung durchführen

Speiseplan Einkaufsliste Gesamtkosten

Speiseplan Kantine Mühlheim

Kalenderwoche 2 (7. bis 13.1.2013)

Montag, den 7.1.2013	Dienstag, den 8.1.2013	Mittwoch, den 9.1.2013	Donnerstag, den 10.1.2013	Freitag, den 11.1.2013	Samstag, den 12.1.2013	Sonntag, den 13.1.2013
Spaghetti Carbonara	Hausgemachtes Schmalzbrot	Spaghetti Bolognese	Koenigsberger Klopse	Penne Arrabiata mit Speck	Kein Angebot	Kein Angebot
Fischburger	Ringli Aglio e Olio	Penne Arrabiata Vegetarisch	Schwaebische Maultaschen	Penne mit Pesto Rosso	Kein Angebot	Kein Angebot
Spaghetti Aglio e Olio	Penne Aglio e Olio	Omelette mit Kartoffeln	Penne mit Spinatsauce	Belgische Pommes Rot-Weiss	Kein Angebot	Kein Angebot

Speichern der Dateien
 Beenden

Abbildung 2: Oberfläche des Kantinenplaners

Auswahlfenster Kantine: In dem Auswahlfeld Kantine kann zwischen die Kantine Mühlheim und Essen ausgewählt werden.

Textfeld Mitarbeiteranzahl: Je nach Kantine befindet sich in diesem Feld ein vorgegebener Wert für die Anzahl der Mitarbeiter am ausgewählten Standort. Dieser kann überschrieben werden.

Textfeld Ausgabeordner: Speicherort der Ausgabedateien

Radiobutton Ausgabeformat: Hier kann zwischen einer HTML- und Text-Ausgabe ausgewählt werden.

Textfeld Ausgabezeitraum: Der Planungszeitraum kann über ein Anfangs- und Enddatum (Von:,Bis:) manuell ausgewählt werden.

Button Properties aktualisieren: Aktualisiert die Properties-Eigenschaften

Button Eingabedaten erfassen: Öffnet das Eingabedateienfenster (siehe Beschreibung unten)

Button Planung durchführen: Führt die Planung im Rahmen der vorgegebenen Einstellung durch und zeigt das Ergebnis im Ausgabeformat

Ausgabefenster Speiseplan: Zeigt den generierten Speiseplan als HTML-Version im Tab Speiseplan

Ausgabefenster Einkaufsliste: Zeigt die generierte Einkaufsliste als HTML-Version im Tab Einkaufsliste

Ausgabefenster Gesamtkosten: Zeigt die generierten Kosten als HTML-Version im Tab Gesamtkosten

Button Speichern der Dateien: Speichert die Ausgabedateien im Ausgabeordner

Button Beenden: Beendet die Planung und schließt die Oberfläche

(2) Eingabedateienfenster

Dieses Dialogfenster bietet dem Anwender die Möglichkeit die verwendeten Eingabedateien auszuwählen.

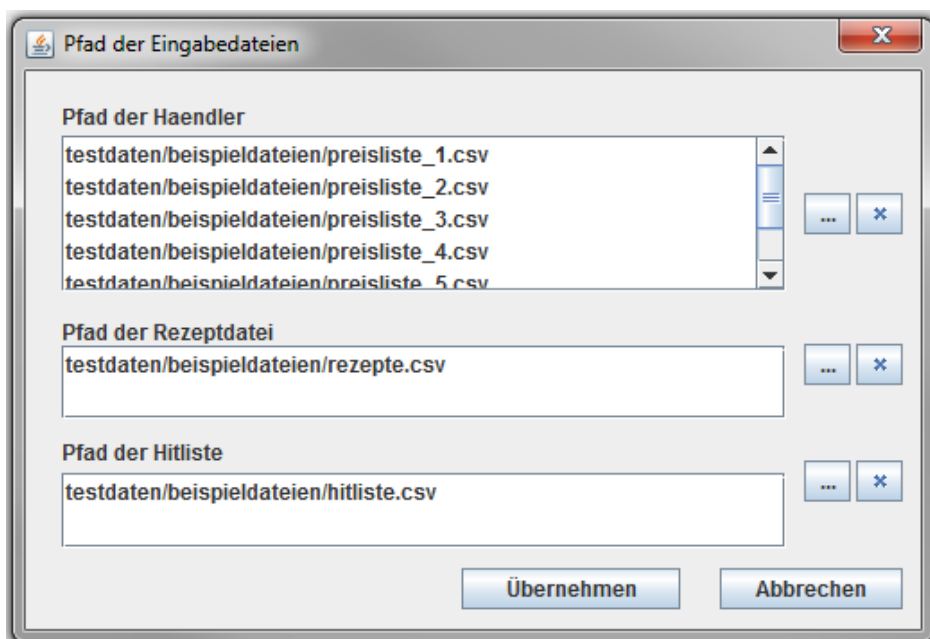


Abbildung 3: Eingabefenster für Pfad der Händler, der Rezeptdatei und der Hitliste

Textfeld Pfad der Haendler: Hier werden alle für die Planung verwendeten Preislisten der Haendler angezeigt.

Textfeld Pfad der Rezeptdatei: Hier werden alle für die Planung verwendeten Rezeptlisten angezeigt.

Textfeld Pfad der Hitliste: Hier werden alle für die Planung verwendeten Hitlisten der Rezepte angezeigt.

Button “...” und “x”: Hinter jedem Textfeld befinden sich jeweils 2 Button zur Verwaltung der Dateien. Nach Klick des Öffnen-Buttons (“...”) öffnet sich ein Fenster zur Auswahl einer Datei aus dem Dateiensystem. Um eine Preisliste zu löschen, muss diese zunächst markiert und mit Klick auf dem Löschen-Button (“x”) entfernt werden.

Button Übernehmen: Übernahme der Einstellungen. Die Properties-Datei wird aktualisiert.

Button Abbrechen: Bricht den Vorgang ab und schließt das Fenster.

3. Anforderungen an die Software

Minimalanforderungen:

Die Mindestanforderung an den Kantinenplaner, die wir im Rahmen dieses Kurses entwickelt haben, ist das Generieren eines gültigen Speiseplans und das Erzeugen der entsprechenden Dateien (Speiseplan, Kostenübersicht und Einkaufsliste).

Zusatzanforderungen:

Darüber hinaus haben wir noch weitere Zusatzfunktionen entwickelt und implementiert:

- leistungsfähige Algorithmen zum Preisvergleich , Hitlisten
- leistungsfähiger Algorithmus zur Planung
- Konfiguration per Konfigurationsdatei. Diese ermöglicht es dem Benutzer auf einer abstrakten Ebene das Programm zu konfigurieren, insbesondere ohne Parameter an einen Programmaufruf in der Kommandozeile anfügen zu müssen. Die Parameter sind jeweils in der Konfigurationsdatei noch einmal erklärt und im Auslieferungszustand mit Beispielen gefüllt. Hiermit wird das Konzept verfolgt, dass jeder das Programm starten können muss, der einen Windowsrechner bedienen kann.

Achtung! Die ausgelieferten Beispiele müssen nicht lauffähig sein. Vor der ersten Nutzung muss der Benutzer diese Werte an sein System anpassen. Dies sind insbesondere die Dateipfade für Preislisten, die Hitliste und die Rezepte.

- Startskripte ermöglichen das Starten des Programms sowohl im automatisierten als auch im Fenstermodus ohne das der Benutzer Kenntnisse über Java oder MS-DOS benötigt (der normale „Fensterutzer“ eben). Dadurch stellen wir sicher, dass jeder das Programm starten kann, der einen Windowsrechner bedienen kann.
- Automatisierter Buildvorgang inklusive Testklassen (build.xml):
 - Ermöglicht das automatisierte Kompilieren, Testen und Bauen des Produktes. Dadurch wird eine konstante Qualität gewährleistet, da keine händischen Eingriffe mehr notwendig sind. Zudem wird gewährleistet, dass ein einmal funktionsfähiges Produkt reproduzierbar ausgeliefert werden kann.
 - Das Buildskript baut außerdem das als Teilleistung 3 abgegebene ZIP-Archive. Dadurch wird sichergestellt, dass keine Dokumente bei der Abgabe vergessen werden kann.
- Toleranz bei der Planung des Speiseplans. Über die Konfigurationsdatei oder die Oberfläche kann eine Toleranz in Euro angegeben werden. Grundsätzlich werden günstigere Gerichte vor teureren Gerichten geplant. Das bedeutet, dass beliebte Gerichte nur eingeplant werden, wenn sie wenigstens gleich teuer sind als ein anderes Gericht. Die Toleranz gibt an, wie viel Euro ein beliebtes Gericht teurer sein darf als das nächst günstigere Gericht, um in den Speiseplan einzufließen. Dadurch kann der Koch nach eigenem Ermessen sein Budget optimieren, um mit geringem Mehraufwand einen wesentlich beliebteren Speiseplan zu erstellen. Dadurch werden auch mehr Mitarbeiter in der Kantine erwartet, wodurch der Gewinn trotz geringer Mehrausgaben steigen sollte.

Beispiel:

Toleranz = 0,10 ; Gericht A kostet im Einkauf 0,22€ je Portion, Gericht B kostet 0,25€ je Portion. Gericht B ist laut Hitliste jedoch beliebter, als Gericht A.

Der Speiseplan wird bevorzugt Gericht B enthalten, da die Kostendifferenz ($0,25 - 0,22 = 0,03$) geringer ist als die Toleranz von 10 Cent.

- Erstellung einer benutzerfreundlichen Oberfläche:
 - Es ist eine übersichtliche Oberfläche erstellt worden, die automatisch Eingaben aus der Konfigurationsdatei einliest und dem Anwender eine einfache, übersichtliche Änderung dieser ermöglicht.
 - Die Planung ist einfach durchzuführen und kann vor Speicherung am Bildschirm angeschaut werden. Der Anwender kann selber entscheiden, ob er mit dem Ergebnis zufrieden ist oder nicht. Er kann die Eingaben erneut ändern und erst dann Speichern, wenn er mit dem Ergebnis zufrieden ist.
 - Anhand eines „Tabbed Pane“ kann der Nutzer zwischen der Sicht auf die Einkaufsliste, den Speiseplan und die Kostenübersicht wechseln.

- Die Eingabedateien können mit einem FileChooser ausgewählt werden. Er zeigt nur im Programmverzeichnis liegenden Ordner und csv-Dateien an. Nicht verarbeitbare Dateitypen werden ausgeblendet.
- mit ToolTips werden dem Anwender Funktionen (z.B. Toleranz) erläutert.
- Überprüfen der Eingabewerte / Abfangen von Falscheingaben:
 - **Datumsprüfung**, beim Verlassen der Datumsfelder wird das Datum auf Plausibilität geprüft. Liegt kein gültiges Datum vor, wird dies durch Tooltipp und roter Hintergrundfarbe des Textfelds kenntlich gemacht.
 - Vor Speichern der Properties wird die **Datumsdifferenz** der beiden angegebenen Daten überprüft; liegt der Beginn auch vor dem Ende, wird dies auch hier mit ToolTip und roter Hintergrundfarbe kenntlich gemacht.
 - Die Mitarbeiteranzahl und die Toleranz werden ebenfalls auf Validität geprüft. Es werden nur Zahlen bzw. bei Toleranz auch Kommazahlen (Darstellung mit Punkt) akzeptiert. Hinweis durch ToolTip und roten Hintergrund.
 - Das Programm ist nur ausführbar wenn eine die Mindestanzahl an Eingabedateien erfüllt ist. (min. eine Händlerdatei und genau eine Hitliste und eine Rezeptdatei).
 - es ist immer nur **genau eine Hitliste und Rezeptdatei** auswählbar. Bei Zuwiderhandlung wird auch dies mit ToolTip und rot hinterlegtem Textfeld kenntlich gemacht.
- Ausgabe in HTML-Format
- flexibler Speiseplan: Start- und Endedatum kann separat gewählt werden. Dadurch ist eine Planung für beliebig viele Wochen/ Tage möglich.
- Erweiterungsfreundlicher Aufbau des Systems durch Programmierung gegen Interfaces und abstrakte Klassen. Dadurch können Änderungen später leicht eingefügt werden, z.B. wenn das Projekt schnell fertig wird, und der Planungsalgorithmus erweitert werden soll. Es ermöglicht dem Team außerdem schnell eine lauffähige Version zu erstellen, die bereits integrativ getestet werden kann. So erhält das Team schnell Feedback und kann das Programm stufenweise erweitern/ fertigstellen.
- Logging ermöglicht dem Benutzer einen fertigen Bericht an die Entwickler zu senden, sodass diese alle Informationen erhalten, um Fehlersituationen zu rekonstruieren und Fehler zu berichtigen. Durch die Nutzung des Loggingframeworks „Log4j“ können Loglevel konfiguriert werden, sodass der Detailgrad steuerbar wird und gezielt Probleme betrachtet werden können.
- Testklassen erhöhen die Qualität der Software. Sie sind automatisiert ausführbar und ermöglichen somit schnell Aussagen über die Funktionsfähigkeit des Programms zu machen.