# HTWG Konstanz

Eisenstadt 1.4

Anwendung der Linearen Optimierung

Alexander Akkus 288186

Guangyu Lu 287843

27.01.2016

1 Inhaltsverzeichnis ........................................................................................................... 2

2 Einleitung ....................................................................................................................... 3

3 Aufgabenstellung............................................................................................................ 4

4 Tätigkeiten ...................................................................................................................... 6

5 Fazit………………………………………………………………………………………………………………………………………..9

# 2. Einleitung

Im Rahmen der Vorlesung „Anwendung der linearen Optimierung“ bei Prof. Dr. Michael Grütz sind von den Studierenden in Zweier-Gruppen vorhandene Methoden der Methodenbank zu überarbeiten.

Hierbei wird jeder Zweier-Gruppe eine Methode zugeteilt, bei der die Gruppe vorhandene

Fehler beheben, sowie zusätzliche Funktionen implementieren soll.

# 3 Aufgabenstellung

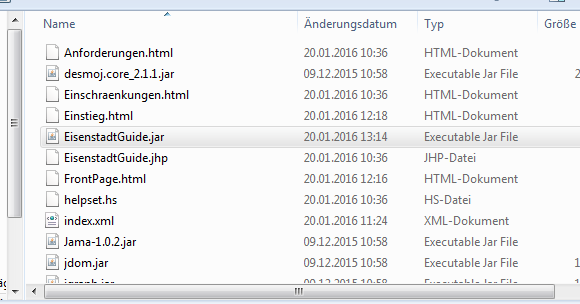
Im Sommersemester 15 sollte die Weiterentwicklung der Methode „Eisenstadt 1.4“

erneut aufgegriffen werden. In Rücksprache mit Serkan Önnisan und Prof. Dr. Michael Grütz wurden folgende funktionale und nicht funktionale Anforderungen im Pflichtenheft definiert, die implementiert werden sollten:

* ***Der Benutzer muss im Programm durch einen Button die Hilfedatei aufrufen können***
  + ***Es wurde eine Hilfedatei eingebunden (EisenstadtGuide.jar) liegt im lib – Ordner des Projekts.***
  + ***In der Klasse View wurde ein Hilfebutton erzeugt (Zeile 360-364)***
  + ***In der Klasse MenuListener wurde eine Methode eingebunden, die aufgerufen wird, wenn der Hilfe Button geklickt wird. Sie startet die Hilfe (Zeile 147 – 173)***
* ***Der Benutzer muss Tooltipps bei Mouseover angezeigt bekommen***
  + ***Wurde in allen Klassen im Package View bei Buttons hinzugefügt mit dem Befehl .setToolTip(„Text der angezeigt wird“).***
* ***Der Benutzer muss bei auftretenden Fehlern eine Fehlerausgabe bekommen***
  + ***Lässt sich in den Einstellungen auswählen, wird dann in eine Datei geschrieben (war schon programmiert)***
* ***Der Benutzer muss beim Klick auf Report anzeigen sofort den Report angezeigt bekommen (Wartezeit von max. 5 Sekunden)***
  + ***Der Java Viewer ist nicht so performant, weswegen die Reports jetzt im Browser geöffnet werden (Klasse ReportFrame Zeile 54-56, der alte Code wurde einfach gelöscht)***
* ***Der Benutzer soll bei Eingabe zu vieler Buslinien per PopUp darauf aufmerksam gemacht werden***
  + ***Bei mehr als 20 Linien kommt ein PopUp (Klasse SimuButtonListener Zeile 56-57 und 60-61)***
* ***Der Benutzer soll nicht zu viele Objekte erzeugen können (Bushaltestellen, Linien, Strecken)***
  + ***Wird auch hier per Pop-Up darauf aufmerksam gemacht (bei mehr als 15 Zielen) (Klasse SimuPanel Zeile 447-448 und 451-452)***
* ***Der Benutzer soll durch Klick die Simulation starten können (Es fehlen einige Klassen im Sourcecode, weswegen „Unresolved compilation problems“ auftreten)***

# 4 Tätigkeiten

**Hilfedatei wurde eingebunden und in der JAVA Applikation mit einem Hilfe Button ausführbar gemacht.**



Die Hilfe-Datei wird beim „Eisenstadt 1.4“ als externe JAR eingebunden und in der

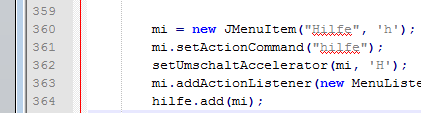
Windows-Hilfe Oberfläche angezeigt. Zum Bearbeiten der Hilfedatei muss zuerst die JAR

entpackt werden (z.B. mit WinRAR). Sämtliche HTML-Dateien können dann ohne Probleme

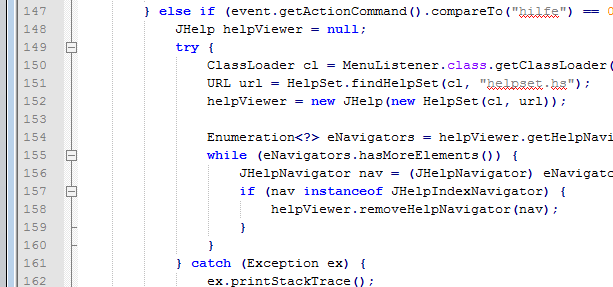
mit einem Text-Editor bearbeitet werden. Die Hilfe-Datei ist im Programmverzeichnis unter

„EisenstadtGuide.jar“ zu finden.

**Einbindung des Button im Code der Klasse View.**



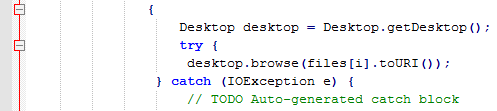
**Try and Catch Implementierung bzw Erweiterung der Methode „Action Performed Event” zum Aufruf der Hilfedatei“.**



**Mouseover für Benutzerfreundlichkeit. Wurde in allen Klassen im Package View bei Buttons hinzugefügt mit dem Befehl .setToolTip(„Text der angezeigt wird“).**

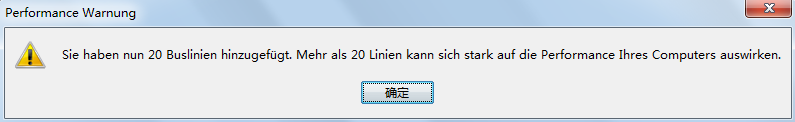


**ReportFrame Klasse für die Ausgabe eines Reports wird nun im Browser angezeigt, änderung des Codes in der Klasse „ReportFrame“.Der alte Code wurde einfach gelöscht Zeile 54-56.**



**Einfügen eines SimButtonListener um die Maximale Anzahl der Buslinen zu beschränken.**

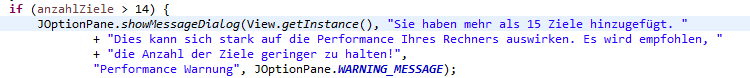


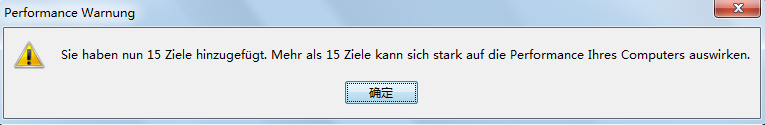


**Der Benutzer soll nicht zu viele Objekte erzeugen können (Bushaltestellen, Linien, Strecken.**

Wird auch hier per Pop-Up darauf aufmerksam.







# 5 Fazit

Es war nicht einfach sich zum Ersten Mal in ein solch Komplexes Programm einzuarbeiten. Es hat sehr lange gedauert den Zusammenhang der Klassen zu verstehen. Erst nach einer langen Einarbeitungszeit konnten die Ersten Änderungen vollzogen werden. Die Zeit die wir zu zweit für dieses Projekt zur Verfügung hatten war zu eng bemessen. In den voran gehenden Jahren waren für weniger Aufwand mehr Personen zur Projekt Bearbeitung beteiligt.