

Westfälische Wilhelms-Universität Münster
Fachbereich Mathematik und Informatik
Programmierpraktikum WS 2000/2001
Dozent: Dr. Dietmar Lammers

Pflichtenheft

zum erweiterten UML-Tool

Dr. UTopp2001
Dr. UTopp2001
professional

Projektgruppe **SynergieSoft**

Baumann, Nadine	nbauman@uni-muenster.de
Debald, Tobias	debald@uni-muenster.de
Hartmann, Uta	hartmau@math.uni-muenster.de
Lohmeyer, Henning	lohmeyh@uni-muenster.de
Sahlmann, Arnd	sahlmann@uni-muenster.de
Starke, Alexander	alstar@uni-muenster.de
Wennmacher, Stefanie	wennmac@muenster.de

Inhaltsverzeichnis

1. ZIELBESTIMMUNG.....	1
1.1 <i>Musskriterien</i>	1
1.2 <i>Wunschkriterien</i>	1
1.3 <i>Abgrenzungskriterien</i>	1
2. PRODUKTEINSATZ.....	1
3. PRODUKTUMGEBUNG.....	2
4. PRODUKTINFORMATIONEN.....	3
<i>Diagrammart</i>	3
<i>Diagrammkomponente</i>	4
<i>Anwendungsfalldiagramm</i>	5
5. FACHKONZEPTKLASSEN UND PRODUKTDATEN	6
5.1 <i>GrafikManager</i>	6
5.2 <i>Verbindung</i>	7
5.3 <i>Akteur</i>	8
6. PRODUKTLEISTUNGEN	8
7. BENUTZEROBERFLÄCHE.....	9
7.1 <i>Dr. UTopp2001 professional</i>	9
7.2 <i>Akteur</i>	9
8. QUALITÄTSZIELE	10
9. TESTSZENARIEN/TESTFÄLLE	10
10. ENTWICKLUNGSUMGEBUNG.....	10

1. Zielbestimmung

Das U-Topp Hauptprogramm soll die von U-Topp erfassten UML-Einzelkomponenten, wie Klasse, Objekt, Use-Case und Verbindung, zu einem in der UML bekannten Diagramm zusammenfügen.

Ebenso wie in U-Topp, so soll auch im Hauptprogramm die graphische Darstellung interaktiv bearbeitet werden können, d.h. durch das Anklicken einer graphischen Komponente öffnet sich ein Dialogfenster, in dem man die detaillierten Angaben zur Komponente editieren kann. Zudem müssen die Einzelkomponenten **graphisch verbunden** werden können, z.B. sollte der User per Mausklick die Möglichkeit haben, eine Verbindung zwischen zwei Komponenten herzustellen oder Verbindungen hinzuzufügen.

1.1 Musskriterien

1. Darstellung von Klassen - und Use-Case-Diagrammen
2. Graphische Verbindung der Elemente (per Maus)
3. Erweiterbarkeit der graphischen Darstellung

1.2 Wunschkriterien

1. Automatische Verfolgung (d.h. das sich z.B. Verbindungslinien anpassen).

1.3 Abgrenzungskriterien

-

2. Produkteinsatz

2.1 Anwendungsbereiche

- Softwareentwicklung

2.2 Zielgruppen

- Softwareentwickler

2.3 Betriebsbedingungen

- Es ist davon auszugehen, dass die Benutzer über einige Erfahrung in Einsatz und Betrieb von Software verfügen, und ggf. auch über einige weitere Zusatzsoftware verfügen bzw. bereit sind, diese zu beschaffen/ zu installieren.

3. Produktumgebung

3.1 Software

- Eine Java-VM sollte bereitstehen.

3.2 Hardware

- Hinreichend leistungsfähig.

3.3 Orgware

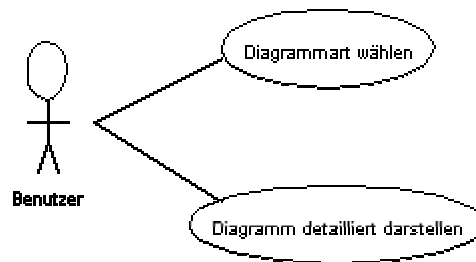
- -

3.4 Schnittstellen

- -

4. Produktinformationen

Diagrammart



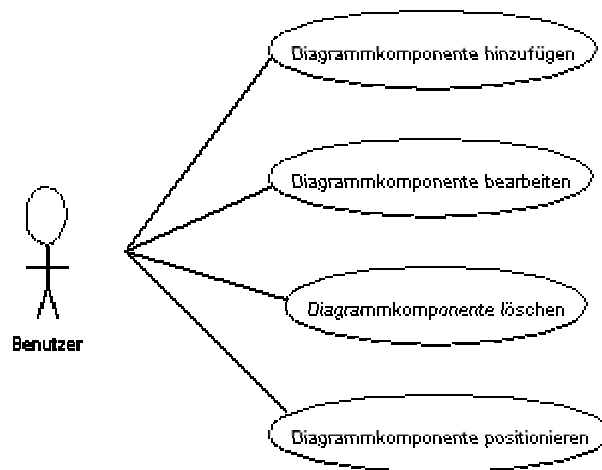
4.1 Wählen

Akteur	Benutzer
Vorbedingung	-
Nachbedingung	Komponenten-Buttons zum Anlegen der jeweiligen Diagrammart sind aktiviert.
Ablaufbeschreibung	Die gewünschte Diagrammart wird vom Benutzer ausgewählt und die entsprechende Symbolleiste wird geladen.
Ausnahmen, Fehlersituationen	-
Service	In der Voreinstellung ist als Diagrammart "Klassendiagramm" markiert.
Dialogbeispiel	Dr. UTopp2001 professional

4.2 Detailliert darstellen

Akteur	Benutzer
Vorbedingung	Diagramm vorhanden.
Nachbedingung	Diagramm ist entweder einfach oder ausführlich dargestellt.
Ablaufbeschreibung	Die gewünschte Darstellungsart wird vom Benutzer durch Auswählen eines Radio-Buttons ausgewählt.
Ausnahmen, Fehlersituationen	-
Service	-
Dialogbeispiel	Dr. UTopp2001 professional

Diagrammkomponente



4.3 Hinzufügen

Akteur	Benutzer
Vorbedingung	Es ist eine Diagrammkomponente vorhanden
Nachbedingung	Diagrammkomponente hinzugefügt.
Ablaufbeschreibung	Der Benutzer wählt eine mögliche UML-Komponente aus. Diese wird in einem gewählten Detaillierungsgrad in dem Ansichtsfenster oben links ausgegeben.
Ausnahmen, Fehlersituationen	Falls eine Komponente bereits hinzugefügt wurde, erscheint eine Fehlermeldung.
Dialogbeispiel	Dr. UTopp2001 professional

4.4 Bearbeiten

Akteur	Benutzer
Vorbedingung	Es ist eine Diagrammkomponente vorhanden.
Nachbedingung	Komponente und ihre graphische Darstellung sind bearbeitet.
Invarianten	Die Diagrammkomponente ist bereits hinzugefügt.
Ablaufbeschreibung	Fall 1: In dem Ausgabefenster kann der Benutzer durch Doppelklick die entsprechende Komponente aktivieren und in der bekannten KomponentenGUI bearbeiten Fall 2: Der Benutzer kann die zu bearbeitende Komponente in der Komponentenliste markieren und dort bearbeiten.
Ausnahmen, Fehlersituationen	-
Dialogbeispiel	Dr. UTopp2001 professional

4.5 Löschen

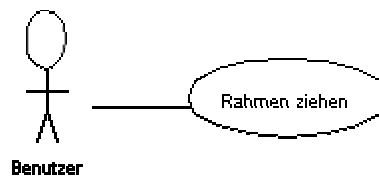
Akteur	Benutzer
Vorbedingung	Es ist eine Diagrammkomponente vorhanden.
Nachbedingung	Diagrammkomponente entfernt.

Ablaufbeschreibung	In dem Ausgabefenster kann der Benutzer durch einfaches Klicken die entsprechende Diagrammkomponente auswählen und diese mit dem Button "Entfernen" löschen.
Ausnahmen, Fehlersituationen	-
Service	Abfrage, ob die entsprechende Diagrammkomponente wirklich gelöscht werden soll.
Dialogbeispiel	Dr. UTopp2001 professional

4.6 Positionieren

Akteur	Benutzer
Vorbedingung	Es ist eine Diagrammkomponente vorhanden und bereits hinzugefügt.
Nachbedingung	Die Diagrammkomponente ist positioniert.
Ablaufbeschreibung	Die bereits eingefügte Komponente kann durch gedrückte Maustaste positioniert werden.
Service	Die zu der neu positionierten Komponente gehörigen Komponenten, z.B. Verbindungen, passen sich automatisch an.
Dialogbeispiel	Dr. UTopp2001 professional

Anwendungsfalldiagramm

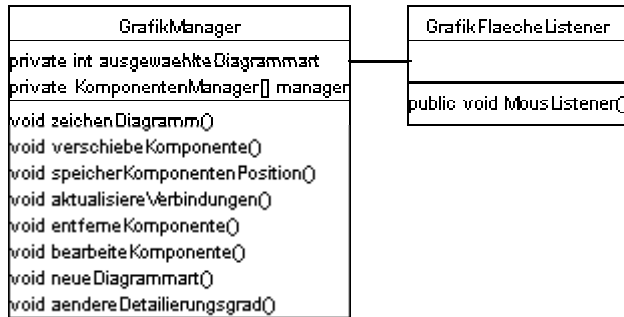


4.7 Rahmen ziehen

Akteur	Benutzer
Vorbedingung	Das Anwendungsfalldiagramm ist ausgewählt und dort ist bereits mind. eine Komponente hinzugefügt worden.
Nachbedingung	Alle Anwendungsfälle sind in mit einem Rahmen, der mit dem Diagrammnamen versehenen ist, eingebunden.
Ablaufbeschreibung	Der Benutzer aktiviert den Button "Rahmen setzen" und passt die Größe des Rahmens durch ziehen mit der Maustaste entsprechend an; außerdem wird eraufgefordert, dem Diagramm(rahmen) einen Namen zu geben.
Ausnahmen, Fehlersituationen	-
Dialogbeispiel	Dr. UTopp2001 professional

5. Fachkonzeptklassen und Produktdaten

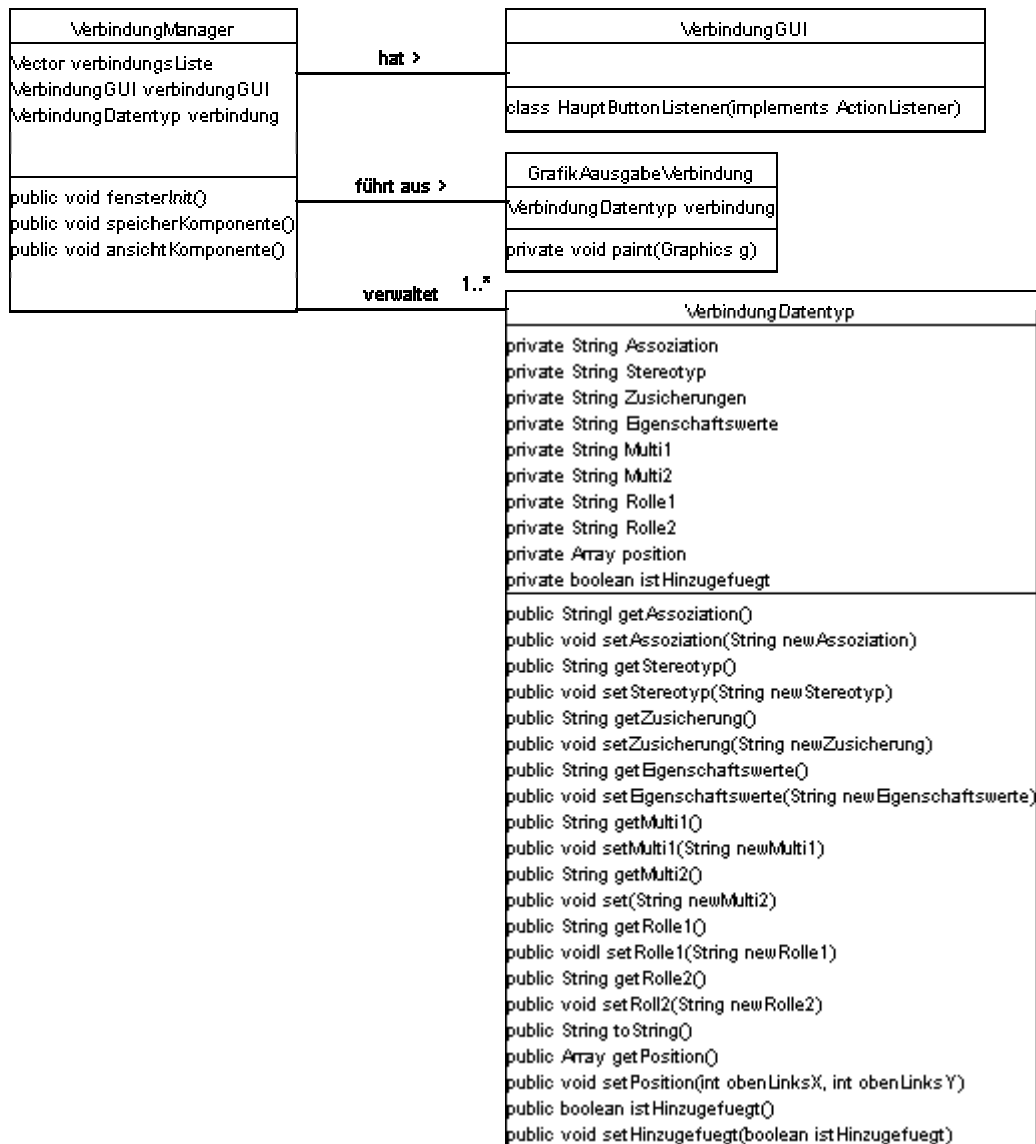
5.1 GrafikManager



Erklärungen zu den Methoden des GrafikManager

- **zeichneDiagramm()**
Die für die Diagrammart zulässigen Komponentenlisten werden überprüft, ob Instanzen zum Diagramm hinzugefügt wurden. Falls ja, wird ihre Position ermittelt und sie werden gezeichnet.
- **verschiebeKomponente()**
Es wird überprüft, welche Komponente per Mausklick ausgewählt wurde. Die Komponente wird verschoben, indem die neue Position dem Datentyp übergeben wird, d.h. die Methode "speichere Komponentenposition" wird aufgerufen. Die Verbindungen werden geupdatet und repainted.
- **speichereKomponentenPosition()**
Die Komponentenposition wird jeweils im dazugehörigen Datentyp gespeichert. Zudem wird die Variable "ist hinzugefügt" auf true gesetzt.
- **aktualisiereVerbindung()**
Die Verbindungsmanager holt sich die Position aller beteiligten Komponenten und richtet danach die neue Positionierung der Verbindungen aus. Anschließend folgt die Methode "speichere Komponentenposition" für die Verbindung.
- **entferneKomponente()**
Komponente wird entfernt und eventuell anhängende Verbindungen ebenfalls.
- **bearbeiteKomponente()**
Es wird überprüft welche Komponente per Doppelklick ausgewählt wurde. Der entsprechende KomponentenManager wird aufgerufen.
- **neueDiagrammart()**
beim Klassendiagramm: Es werden die Button "Akteur", "Use-Case" sowie der Button zur Rahmziehung deaktiviert. Verbindungen, die beim KClassendiagramm nicht benötigt werden, werden ebenfalls deaktiviert.
beim Use-Case-Diagramm: Es werden die Button "Klasse" und "Objekt" deaktiviert.
- **aendereDataillierungsgrad()**
Klasse, Objekt oder Verbindung wird entweder einfach oder ausführlich dargestellt.

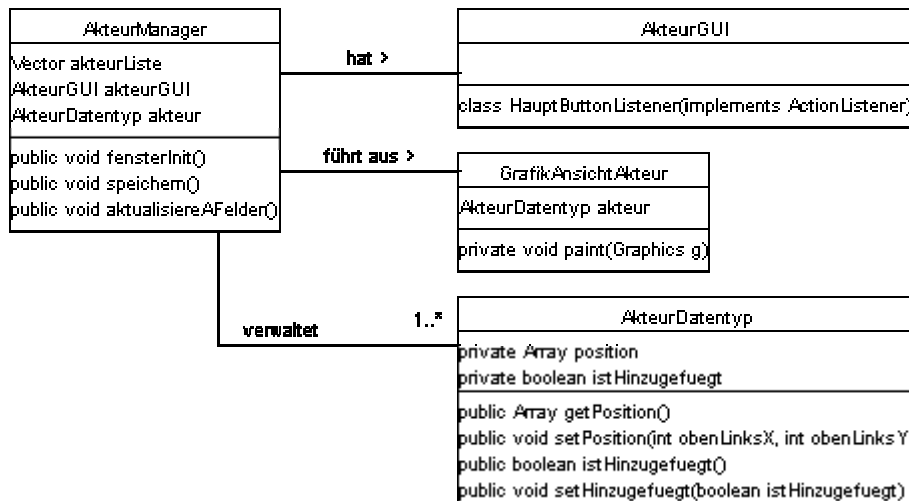
5.2 Verbindung



Erklärungen zu den anderen UML-Komponenten

Wie bei der UML-Komponente Verbindungen, werden auch bei den anderen Komponenten jeweils die Datentypen um Attribute und Operationen ergänzt, um die Position der eventuell gezeichneten grafischen Darstellung einer Komponente innerhalb der Diagrammzeichenfläche zu speichern.

5.3 Akteur

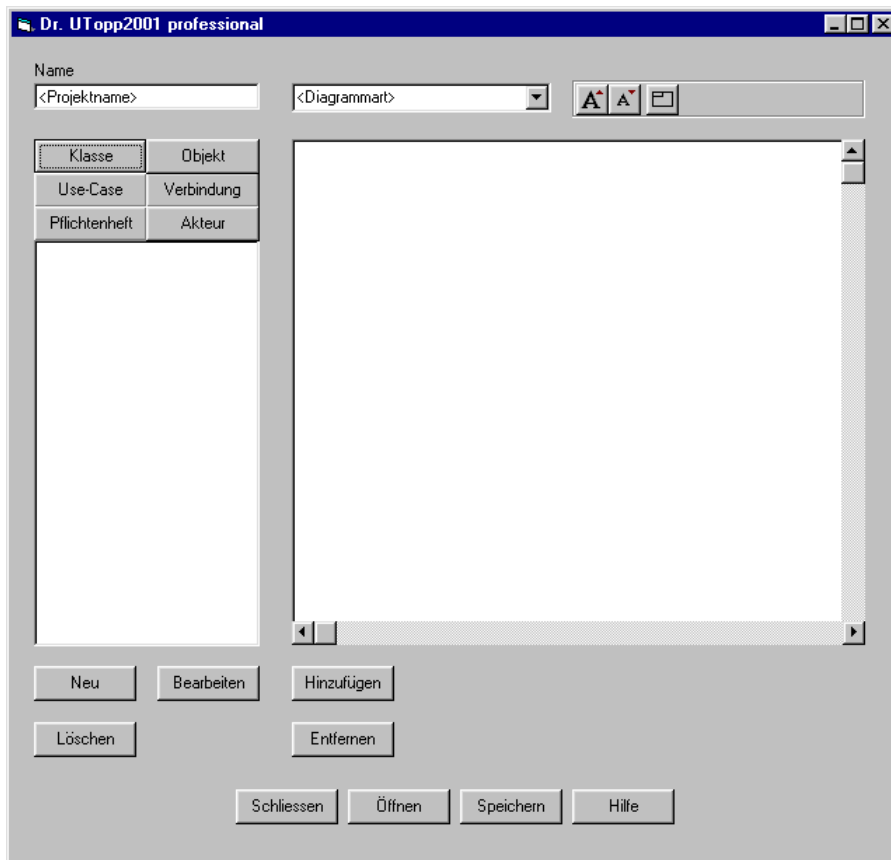


6. Produktleistungen

Die Produktleistungen sind den Produktinformationen unter Punkt 4, in diesem Pflichtenheft, wie im Pflichtenheft zum U-Topp Ausgangsprogramm, zu entnehmen.

7. Benutzeroberfläche

7.1 Dr. UTopp2001 professional



7.2 Akteur



8. Qualitätsziele

Ziel	++	+	+ -	-
Funktionalität		X		
Zuverlässigkeit			X	
Benutzbarkeit		X		
Effizienz			X	
Änderbarkeit	X			

9. Testszenarien/Testfälle

Da für dieses Projekt nur die Modellierungsphase vorgesehen war, können keine Tests am eigentlichen Programm durchgeführt werden.

10. Entwicklungsumgebung

1. Software
 - Java 2.0
 - JBuilder
 - Texteditoren
2. Hardware
 - PC
3. Orgware
 - -
4. Schnittstellen
 - -