

## **Inhaltsverzeichnis**

1.Zielbestimmungen.....	2
1.1. Musskriterien.....	2
1.2. Wunschkriterien.....	2
1.3. Abgrenzungskriterien.....	2
2.Produkteinsatz.....	2
2.1. Anwendungsbereiche.....	2
2.2. Zielgruppen.....	2
2.3. Betriebsbedingungen.....	3
3.Produktübersicht.....	3
3.1. Software.....	4
3.2. Hardware.....	4
3.3. Produktschnittstellen.....	5
4.Produktfunktionen.....	5
5.Produktdaten.....	7
6.Produktleistungen.....	7
7.Qualitätsanforderungen.....	8
8.Benutzeroberfläche.....	9
9.Nichtfunktionale Anforderungen.....	9
10.Entwicklungsumgebung.....	9
10.1. Software.....	9
10.2. Hardware.....	9
10.3. Produktschnittstellen.....	9
11.Ergänzungen.....	10

## 1. Zielbestimmungen

Es soll ein PlugIn für Eclipse entwickelt werden, welches intuitivere Beschreibungsmöglichkeiten für GMFGRAPH-Modell, GMFTOOL-Modell und GMFMAP-Modell ermöglicht und somit die Erstellung von Editoren mittels GMF wesentlich vereinfacht.

### 1.1. *Musskriterien*

Der Nutzer kann sämtliche Einstellungen der \*.gmfgraph, \*.gmftool und \*.gmfmap, welche für die Generierung des Editors notwendig sind, in einer Zeichenfläche mit Hilfe von graphischer Darstellung vornehmen.

### 1.2. *Wunschkriterien*

Wünschenswert wäre eine Druckfunktion, die die Elemente der Zeichenfläche, oder die dynamische Erstellung eigener Figuren (Formen für den eigenen Editor).

### 1.3. *Abgrenzungskriterien*

Das PlugIn soll nur die Erstellung des graphischen Editors vereinfachen und nicht den Editor selbst erstellen. Für die endgültige Erstellung des Editors muss der User weiterhin noch fehlende Dateien erstellt und den Code generiert.

## 2. Produkteinsatz

Das PlugIn wird genutzt um mit Hilfe eines allgemeinen graphischen Editors, neue graphische Editoren zu erstellen, die spezielle, selbst definierte Sprachen modellieren.

### 2.1. *Anwendungsbereiche*

Anwendungsbereich ist die modellbasierte Entwicklung von graphischen Editoren.

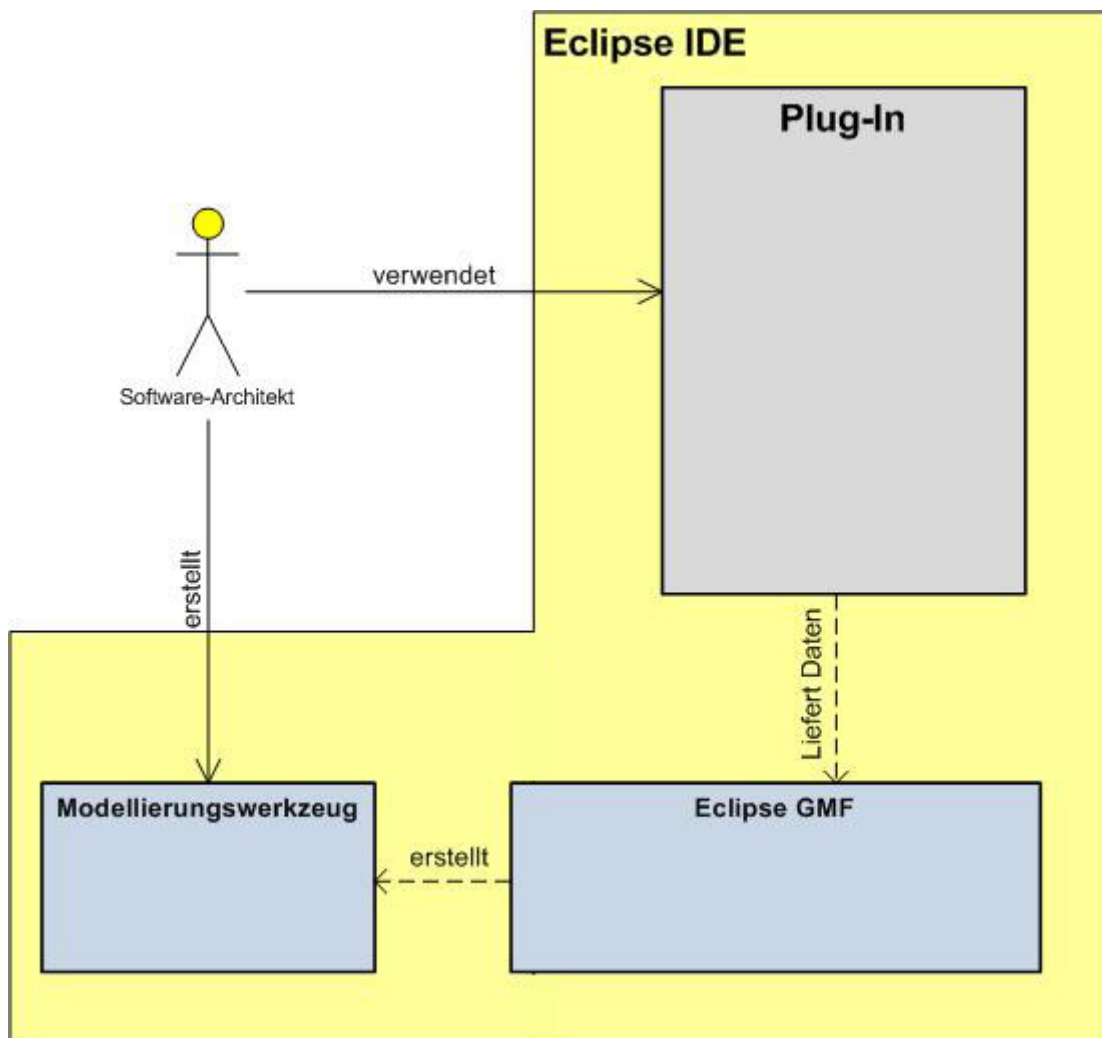
### 2.2. *Zielgruppen*

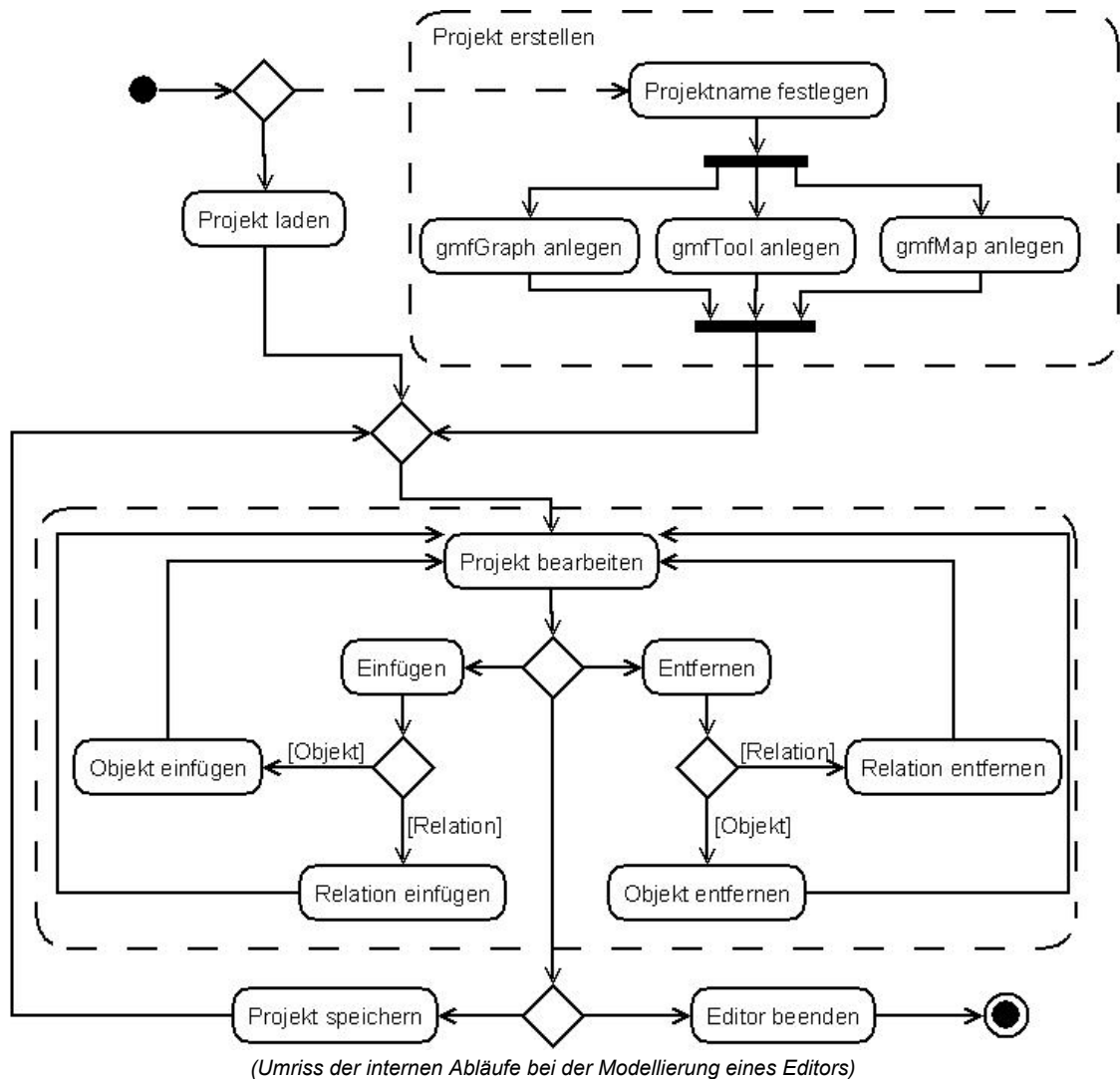
Die Zielgruppe sind hauptsächlich Entwickler von Modellierungssprachen, die bisher mit GMF gearbeitet haben und sich eine intuitivere Handhabung wünschen.

### 2.3. Betriebsbedingungen

Für die Verwendung des PlugIns wird Eclipse in der Version 3.2. mit den PlugIns EMF, GEF und GMF vorausgesetzt. Eclipse gibt es für die gängigen Betriebssysteme Windows, Linux oder Mac OS X.

## 3. Produktübersicht





### 3.1. Software

Da es sich bei dem Produkt um ein PlugIn und nicht um eine eigenständige Software handelt, ist es von der Existenz anderer Software abhängig. Für die Nutzung wird Eclipse in der Version 3.2 mit den PlugIns EMF, GEF und GMF vorausgesetzt. Eclipse gibt es für die gängigen Betriebssysteme Windows, Linux oder Mac OS X.

### 3.2. Hardware

Da es sich um eine Einzelplatzanwendung handelt, die auf Eclipse basiert, reicht ein Einzelplatzrechner, der die Anforderungen von Eclipse erfüllt.

- Java-Version: 5.0 oder höher empfohlen
- Hauptspeicher: 512 MByte, 1 GByte oder mehr empfohlen

Gruppe: HK-07-4

Gruppenleiter: Stanley Hillner

14.05.2007

- Freier Festplattenplatz: 300 MByte, 1 GByte oder mehr empfohlen
- Prozessorgeschwindigkeit: 800 MHz, 1.5 GHz oder schneller empfohlen

### **3.3. Produktschnittstellen**

- EMF Version 2.2.2
- GEF Version 3.2.2
- GMF Version 1.0.3
- Java Virtual Machine Version 1.5.0
- J2SE 5.0, J2EE 1.5.0

## **4. Produktfunktionen**

**/F10/ (/LF10/)**

### **Geschäftsprozess: Neues Projekt erstellen**

Akteur: Software-Architekt

Beschreibung: Der Anwender startet ein neues Projekt zur Erstellung eines graphischen Editors für Modellierungssprachen.

**/F20/ (/LF20/)**

### **Geschäftsprozess: Projekt speichern**

Akteur: Software-Architekt

Beschreibung: Speichern der Fortschritte bei der Erstellung eines Editors durch den Anwender zur späteren Fortführung des Projekts.

**/F30/ (/LF30/)**

### **Geschäftsprozess: Projekt laden**

Akteur: Software-Architekt

Beschreibung: Laden eines existierenden Projekts, welches zuvor gespeichert wurde (/LF20/).

Der Anwender kann auf bereits ausgeführte Arbeitsschritte aufbauen und sein Projekt weiter bearbeiten.

**/F40/ (/LF40/)**

### **Geschäftsprozess: Objekt einfügen**

Akteur: Software-Architekt

Beschreibung: Der Anwender fügt ein vorgefertigtes Objekt zu seinem Editor

## Softwaretechnik-Praktikum SS 2007

### Aufgabenblatt 5

Gruppe: HK-07-4

Gruppenleiter: Stanley Hillner

14.05.2007

hinzu, mit welchem er später arbeiten möchte.

**/F50/ (/LF50/)**

#### **Geschäftsprozess: Objekt löschen**

Akteur: Software-Architekt

Beschreibung: Ein bereits hinzugefügtes Objekt wird durch den Anwender von der Zeichenfläche gelöscht.

**/F60/ (/LF60/)**

#### **Geschäftsprozess: Objekteigenschaften ändern**

Akteur: Software-Architekt

Beschreibung: Der User kann verschiedene Eigenschaften Name, Form, Größe und Farbe der Objekte ändern.

**/F70/ (/LF70/)**

#### **Geschäftsprozess: Relationen zwischen Objekten hinzufügen**

Akteur: Software-Architekt

Beschreibung: Der User setzt zwei oder mehrere Objekte in Relation zueinander, indem er Linien oder Pfeile zwischen diesen malt.

**/F80/ (/LF80/)**

#### **Geschäftsprozess: Relationen zwischen Objekten löschen**

Akteur: Software-Architekt

Beschreibung: Unerwünschte (eventuell falsch gesetzte) Relationen zwischen Objekten können durch Löschen der Linien oder Pfeile zwischen den graphisch repräsentierten Objekten entfernt werden.

**/F90/ (/LF90/)**

**Geschäftsprozess: Verbindung der Objekte mit der \*.ecore-datei**

Akteur: Software-Architekt

Beschreibung: Ein Objekt, das auf der Zeichenfläche hinzugefügt wurde, kann per Rechtsklick auf dieses, mit Hilfe eines Pop-Up-Menüs, mit dem entsprechenden Element der \*.ecore-Datei verbunden werden. Diese Informationen werden in der \*.gmfmap gespeichert.

## **5. Produktdaten**

**/D10/ (/LD10/)**

maximal 50 graphische Elemente stehen dem Nutzer zur Erstellung seines Editors zur Verfügung.

**/D20/ (/LD20/)**

maximal 200 graphische Elemente können auf der Zeichenfläche verarbeitet werden.

## **6. Produktleistungen**

**/L10/ (/LL10/)**

Bei der Erstellung eines neuen Projekts sollte die Arbeitsoberfläche innerhalb von 2 Sekunden aufgebaut sein.

**/L20/ (/LL20/)**

Das Speichern eines Projekts sollte nicht länger als 5 Sekunden dauern.

**/L30/ (/LL30/)**

Beim Laden eines gespeicherten Projekts sollten bis zur Darstellung der geladenen Daten nicht mehr als 5 Sekunden vergehen.

**/L40/ (/LL40/)**

Das Einfügen und Löschen von Objekten und Relationen soll nicht mehr als 1 Sekunde dauern.

**/L50/ (/LL50/)**

Das Ändern von Objekteigenschaften und soll nicht länger als 1 Sekunde dauern.

# Softwaretechnik-Praktikum SS 2007

## Aufgabenblatt 5

Gruppe: HK-07-4

Gruppenleiter: Stanley Hillner

14.05.2007

### 7. Qualitätsanforderungen

<b>Produktqualität</b>	<b>sehr gut</b>	<b>gut</b>	<b>normal</b>	<b>nicht relevant</b>
<b>Funktionalität</b>				
Angemessenheit			x	
Richtigkeit		x		
Interoperabilität		x		
Ordnungsmäßigkeit			x	
Sicherheit			x	
<b>Zuverlässigkeit</b>				
Reife			x	
Fehlertoleranz				x
Wiederherstellbarkeit		x		
<b>Benutzbarkeit</b>				
Verständlichkeit	x			
Erlernbarkeit		x		
Bedienbarkeit	x			
<b>Effizienz</b>				
Zeitverhalten				x
Verbrauchsverhalten				x
<b>Änderbarkeit</b>				
Analysierbarkeit		x		
Modifizierbarkeit		x		
Stabilität			x	
Prüfbarkeit			x	
<b>Übertragbarkeit</b>				
Anpassbarkeit			x	
Illustrierbarkeit				x
Konformität			x	
Austauschbarkeit				x



## 8. Benutzeroberfläche

### **/B10/**

Die Benutzeroberfläche ist in Eclipse integriert und wird mittels Maus bedient.

### **/B20/**

Die graphischen Objekte und die Tools des Editors werden durch Auswahl des entsprechenden Werkzeugs der Werkzeugpalette und anschließendem Klicken und Ziehen auf der Zeichenflächen eingefügt.

### **/B30/**

Eigenschaften der Objekte auf der Zeichenfläche und somit auch der Elemente des zu erzeugenden Editors können mittels Rechtsklick auf das entsprechende Objekt im Kontextmenü geändert werden.

## 9. Nichtfunktionale Anforderungen

Es gibt keine speziellen, nichtfunktionalen Anforderungen an das Produkt.

## 10. Entwicklungsumgebung

### **10.1. Software**

- Eclipse IDE Version 3.2.2
- GMF build 1.0.3
- EMF build 2.2.2RC
- GEF build 3.2.2
- EMF OCL build M200611161546
- EMF Validation build M200611161604
- EMF Query build M200611161603
- EMF Transaction build M200611161636

### **10.2. Hardware**

Die Hardware unterscheidet sich nicht von den in Punkt 3.2 definierten Eigenschaften.

### **10.3. Produktschnittstellen**

- EMF Version 2.2.2
- GEF Version 3.2.2

Gruppe: HK-07-4

Gruppenleiter: Stanley Hillner

14.05.2007

- GMF Version 1.0.3
- Java Virtual Machine Version 1.5.0
- J2SE 5.0, J2EE 1.5.0

## **11. Ergänzungen**

Das Produkt ist nicht dazu bestimmt, fertige Editoren als Eclipse-Plugins zu erstellen. Es werden lediglich die drei Dateien \*.gmfgrah, \*.gmftool und \*.gmfmap generiert, welche dann für weitere Bearbeitung des Editors durch den Benutzer zur Verfügung stehen. Es wird also dem Benutzer nur die Tätigkeit der Erstellung dieser drei Dateien vereinfacht, also die grafische Definition des Editors, nicht aber die Generierung des erforderlichen Quellcodes.