

Fortgeschrittenen Praktikum

Bericht

Thema:

**Entwurf und Implementierung eines
Eclipse Plug-ins zur Spezifikation von
Geschäftsprozessprofilen
(Kalenderoberfläche)**

BetreuerIn: **Prof. Dr. Barbara Paech**
Robert Heinrich

Praktikantin: **Annika Boldt**

Gliederung:

- I. Einleitung
- II. Aufgabenstellung
- III. Technische Hilfsmittel
- IV. Anforderungen
 - i. Funktionale Anforderungen
 - ii. Nichtfunktionale Anforderungen
 - iii. Klassendiagramm
- V. Design
 - i. Klassendiagramm
 - ii. Komponentendiagramm
 - iii. Prozesskalender-View
 - iv. Tages Wizard
 - v. Tagesprofile Wizard
 - vi. Aktorkalender-View
 - vii. Tagesprofile Wizard
 - viii. Zeiteinheit Wizard
- VI. Implementierung
 - i. Kalender
 - ii. Tage und Tagesprofile
 - iii. Wizards
 - iv. Prozesskalender & Aktorkalender
 - v. Usagemodel und Organizationenvironmentmodel
- VII. Testspezifizierung
 - i. Kalender
 - ii. Tages Wizard
 - iii. Tagesprofile Wizard
 - iv. Usagemodel
 - v. Organizationenvironmentmodel
 - vi. conf.txt Datei
 - vii. Nicht-funktionale Anforderungen
- VIII. Erfahrungen
 - i. Persönliche Entwicklung
 - ii. Nutzen für Studium
- IX. Schluss
- X. Referenzen

I. Einleitung

Da es in großen oder auch in kleinen Betrieben möglich ist, dass im Laufe eines Tages mehrere Geschäftsprozesse gestartet werden, allerdings nur zu bestimmten Uhrzeiten, gegebenenfalls den Arbeitszeiten dieses Betriebs, ist es sinnvoll für einen festen Zeitraum Perioden zu definieren, in denen Prozesse gestartet werden können. Dieser Zeitraum könnte beispielsweise ein Jahr umfassen.

Der Benutzer, der die Bearbeitung der Geschäftsprozesse verwaltet, könnte von einem Programm gebrauch machen, das auf einer Benutzeroberfläche einen Kalender visualisiert auf dem verschiedenen Profile für unterschiedliche Tage angelegt werden können, die keine, bis unendlich viele Zeitperioden enthalten, in denen Prozesse gestartet werden können.

Zum Beispiel, wenn nachts niemand arbeitet, so gibt es zu der Zeit keine Periode, in der Prozesse starten könnten, oder wenn der Betrieb eine feste Zeit für die Mittagspause hat.

Es könnte beispielsweise zwischen zwei festen Tagesprofilen unterschieden werden, Working Days und Free Days, Tage an denen gearbeitet wird und Tage an denen grundsätzlich niemand im Betrieb arbeitet.

Die Perioden könnten in verschiedenen Zeiteinheiten abgespeichert werden, die der Benutzer ebenfalls frei wählen kann (unter Millisekunden, Sekunden, Minuten und Stunden).

II. Aufgabenstellung

In diesem Fortgeschrittenen Praktikum sollte ein Eclipse Plug-in zur Spezifikation von Geschäftsprozessprofilen entwickelt werden.

Diese Geschäftsprozessprofile sollen verschiedene Intervalle beinhalten, welche die Zeiträume festlegen, in denen die Bearbeitung eines Geschäftsprozesses starten könnte. Jedes Intervall besteht also aus einem Start- und einem Endzeitpunkt und aus einer statistischen Verteilungsfunktion, die angibt, wie oft im Durchschnitt Prozesse innerhalb dieses Intervalls gestartet werden.

Die Intervalle, innerhalb eines Profils, sollen sich nicht überschneiden.

Es sollte eine Oberfläche implementiert werden, auf der die Tagesprofile bearbeitet werden können. Die Bearbeitung schließt ein, neue Tagesprofile zu erstellen, bzw. bereits bestehende zu löschen, die Umbenennung des Tagesprofils, Das Hinzufügen, Ändern und Löschen von Intervallen.

Außerdem soll eine Kalenderoberfläche implementiert werden, welche die Möglichkeit bietet für ein Kalenderjahr allen Tagen unterschiedliche Tagesprofile zuzuordnen.

Zusätzlich soll es einen zweiten Kalender geben, einen sogenannten Aktorkalender, damit Arbeitsperioden von Akteuren im Prozess spezifiziert werden können, allerdings gibt es hierbei keine mittlere Verteilung der Startzeitpunkte der Perioden.

Für das Abspeichern der Intervalle sollten die Start- und Endzeitpunkte in abstrakte Zeiteinheiten umgerechnet werden.

III. Technische Hilfsmittel

Zur Realisierung wurden verschiedene technische Hilfsmittel verwendet. Als Programmieroberfläche wurde das Programm eclipse gewählt, welches im Internet als Open Source Programm heruntergeladen werden kann.

Eclipse ist ein sehr umfangreiches Programm, in welchem mit Java programmiert werden kann. Es bietet alle Features, die für dieses Praktikum notwendig waren. Außerdem ermöglicht es das Einbinden externer Tools. (siehe Referenz 01)

Desweiteren wurde die Programmbibliothek SWT (Standard-Widget-Toolkit) verwendet. Wie der Name bereits verrät, ist diese Programmbibliothek Standard für das erstellen von graphischen Oberflächen. (siehe Referenz 02)

Weitere Erläuterungen zu den Vorteilen von SWT befinden sich im Design Kapitel.

Innerhalb des Programms wurden mehrere Wizards erstellt. Auf diese Weise bleibt der Kalender auf der Oberfläche immer erhalten und es können dennoch Änderungen vorgenommen werden, wie zum Beispiel das Ändern einer Tagesperiode. (siehe Referenz 03)

IV. Anforderungen

i. Funktionale Anforderungen

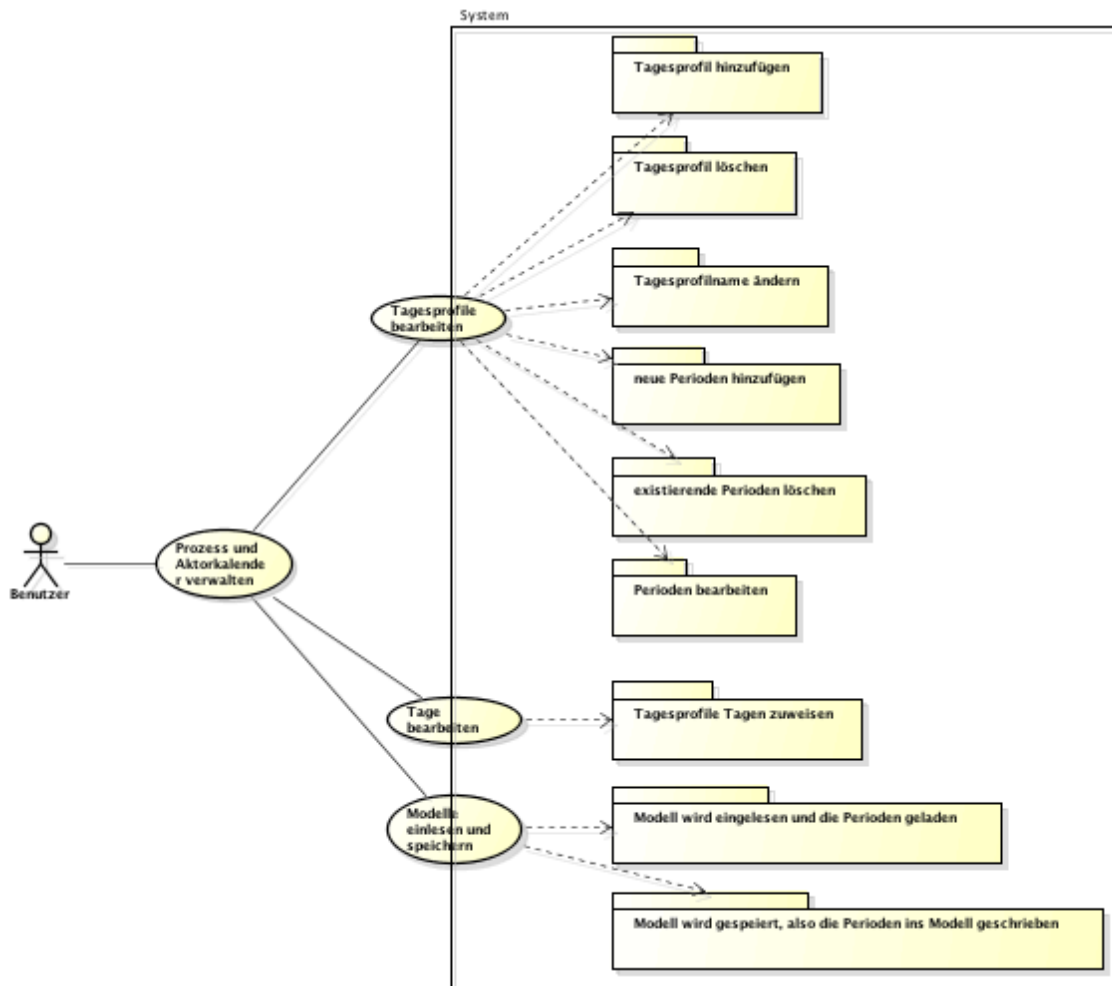
Ein Benutzer, der dieses Programm verwendet, soll die Möglichkeit haben den Prozesskalender bzw. Aktorkalender zu verwalten. Das bedeutet er bearbeitet Tagesprofile, Tage und ist in der Lage Modelle zu laden und wieder abzuspeichern.

Zum Bearbeiten der Tagesprofile gehört, dass ein Tagesprofil hinzugefügt und wieder gelöscht werden kann. Es soll außerdem auch geändert werden können. Das bedeutet, es kann der Name des Tagesprofils geändert werden. Neue Perioden können hinzugefügt werden. Perioden können außerdem gelöscht oder geändert werden.

Den 365 Tagen in den beiden Kalendern können Tagesprofile zugewiesen werden.

Modelle können in dieses Programm eingelesen werden, das bedeutet die Perioden, die in diesem Modell stehen werden übernommen und ebenso kann ein Modell gespeichert werden, in dem die geänderten Perioden in das Modell überschrieben werden.

Auch diese Funktion gilt für den Prozesskalender und den Aktorkalender. Um die Modelle zu speichern, hat der Benutzer die Wahl zwischen vier Zeiteinheiten (Millisekunden, Sekunden, Minuten und Stunden).



ii. Nichtfunktionale Anforderungen

1. Alle Wizards sollen in weniger als 2 sec geöffnet werden. Das gleiche gilt für alle Wizard Seiten.
2. Wenn ein Wizard geschlossen wird, geschieht dies ebenfalls in weniger als 2 sec.
3. Ein Modell soll in weniger als 2 sec geladen sein, und die zugehörigen Szenarien bzw. Akteure sollen in der gleichen Zeit, in der ComboBox bereitgestellt werden.
4. Das abspeichern sollte allerhöchstens in 2 sec geschehen. Nach diesen 2 sec, sollte das Programm wieder voll funktionstüchtig sein.

V. Design

Für die Umsetzung wurde das Design „SWT“ gewählt. Ausgeschrieben heißt SWT Standard Widget Toolkit.

Dieses Widget bietet viele Vorteile zum Beispiel, ist es weniger Ressourcen hungrig als Java Swing, die Ansprechgeschwindigkeit ist auch deutlich schneller, da direkt auf die Ereignisverarbeitung des jeweiligen OS aufgesetzt wird und es verhält sich robuster und toleranter gegenüber der verwendeten Hardware und der verschiedenen Grafikeinstellungen, als AWT um nur die drei, für dieses Projekt wichtigsten Vorteile zu nennen. (siehe Referenz 04)

SWT bietet verschiedene Layoutfunktionen. In diesem Projekt wurde sich für das GridLayout entschieden, da es ein mächtiges Layout ist und allen Anforderungen, die *hier* benötigt werden, gerecht wird. Es kann frei gewählt werden, wie viele Spalten das

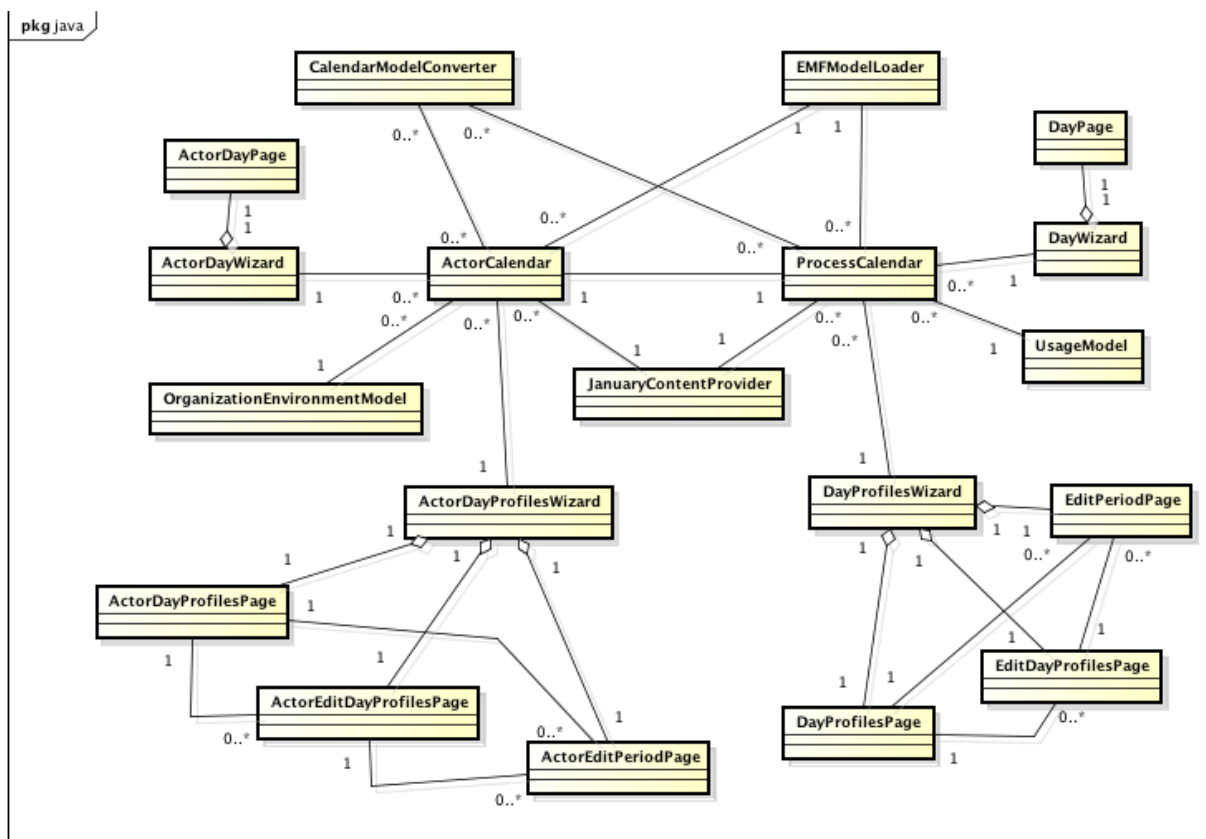
Layout umfasst und die Elemente haben automatisch identische Größen. Das ist *hier* sehr sinnvoll, da zum Beispiel an einigen Stellen mehrere Buttons auftauchen und diese dann direkt gleich groß sind.

Ein GridBagLayout wurde nicht benötigt, da in den Fällen, wenn ein Element, wie zum Beispiel der Kalender, größer sein sollte, als die Buttons darunter oder darüber, mehrere Composites verwendet wurden anstatt die Elementgrößen manuell anzupassen.

Auch das BorderLayout wurde nicht benötigt, da die Elemente auf der View eher links orientiert sein sollten. (siehe Referenz 05)

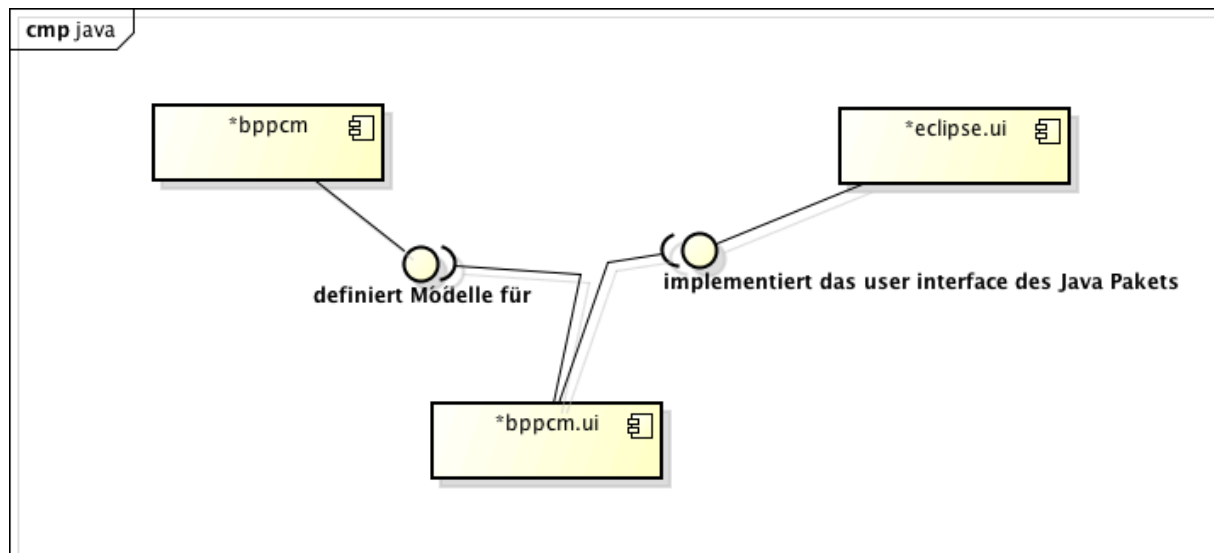
Einmalig wurde das RowLayout verwendet, um mehrere Elemente in eine Reihe zu setzen. Im Folgenden wird die Visualisierung des Programms dargestellt. Die Reihenfolge der Unterkapitel entspricht einer logischen Reihenfolge, die beim Verwenden des Programms denkbar wäre.

i. Klassendiagramm



Dies ist ein komprimiertes Klassendiagramm. Ein ausführliches Klassendiagramm mit allen Klassen, wichtigen Attributen und Methoden, und Zusammenhängen befindet sich im Anhang.

ii. Komponentendiagramm



iii. Prozesskalender-View

Beim Ausführen des Programms öffnet sich zunächst die Prozesskalender-View. Diese sieht folgendermaßen aus:

Usage File

Select a scenario

◀ January ▶

Mo	Di	Mi	Do	Fr	Sa	So
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

In der obersten Zeile kann das „Usage File“ von einem beliebigen Pfad auf dem Computer eingelesen werden. Darunter kann ein Szenario aus einer comboBox ausgewählt werden.

Hierunter befindet sich der Kalender. Oberhalb der Wochentage kann der Monat über die beiden Buttons links und rechts vom Monatsnamen ausgewählt werden. Die Oberfläche darunter entspricht den zugehörigen Wochentagen zu dem ausgewählten Monat.

Unten befinden sich drei Buttons, mit dem ersten Button „Day profiles“ öffnet sich ein Wizard (siehe v.). Mit dem Button „Persist“ kann der aktuelle Stand gespeichert werden. Mit dem untersten Button „Actor Calendar“ wechselt man in die Aktorkalender-View (siehe vi.).

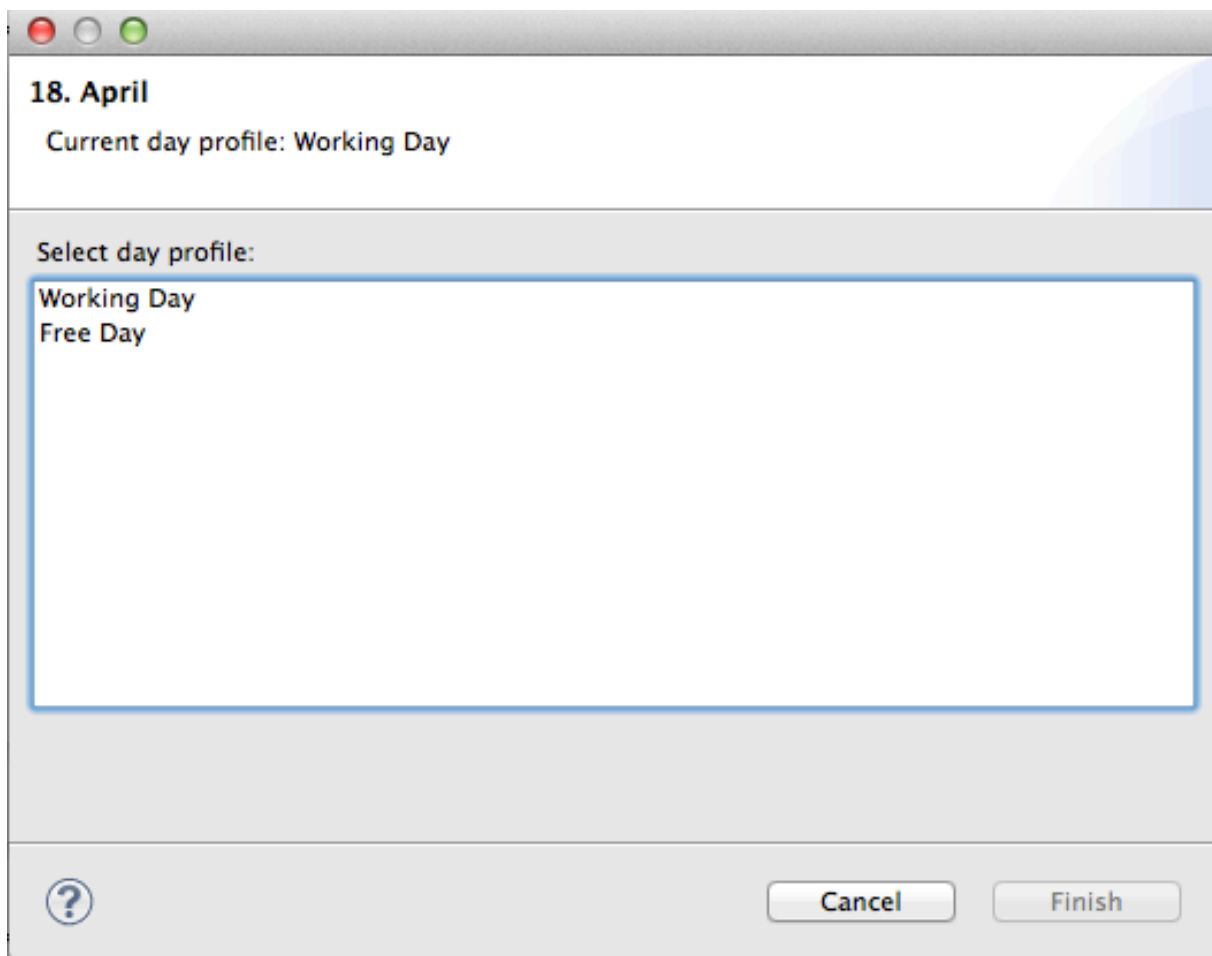
Der „Persist“ Button wird erst aktiviert, nach dem ein Scenario ausgewählt wurde.

Diese optische Anordnung von oben nach unten, wurde gewählt, da sie dem logischen Arbeitsfluss entspricht, so dass zunächst ein UsageModel und ein Szenario ausgewählt wird und dieses im Anschluss auf der Kalenderoberfläche bearbeitet werden kann. Tagesprofile können zugeordnet werden und diese können darunter über den Button „Day profiles“ bearbeitet werden.

Wären in diesem Programm keine Wizards verwendet worden, dann wäre diese erste Oberfläche sehr unübersichtlich geworden, wenn es die einzige wäre, wo Perioden und Tagesprofile geändert werden würden. Durch die visuelle Trennung bleibt das Programm optisch übersichtlich.

iv. Tages Wizard

Bei Doppelklick auf einen Tag öffnet sich ein Wizard.



Hier wird oben im Kopf zunächst das Datum des Tages angegeben und das zugeordnete Tagesprofil. Darunter besteht dann die Möglichkeit dieses Tagesprofil zu ändern, in dem die übrigen vorhandenen Tagesprofile aufgelistet werden.

v. Tagesprofil Wizard

Der Tagesprofil Wizard öffnet sich nach Klick auf den „Day profiles“ Button.

Day Profiles Page

These are the available day profiles. Select Day Profile to edit and click Next.

Day profiles:

- Working Day
- Free Day

Insert Name here to add:

Hier kann der Benutzer eine Liste von den aktuell vorhandenen Tagesprofilen sehen und in der Texteingabe-Zeile darunter einen Namen für ein neues Tagesprofil eingeben und mit Klick auf den sich darunter befindenden „Add“ Button dieses hinzufügen. Neben diesem Button befindet sich der „Delete“ Button um ein zuvor ausgewähltes Tagesprofil zu löschen.

Nach dem ein Tagesprofil ausgewählt wurde kann man mit „Next“ auf die nächste Seite des Wizards gelangen.

Edit a DayProfile, press the 'Add'-Button to add a new Period

Day Profile Name:

A new Period

Start Time Point:

End Time Point:

Inter Arrival Time:

Existing Periods:

- 08:00:00 - 12:00:00 : default
- 13:00:00 - 18:00:00 : default

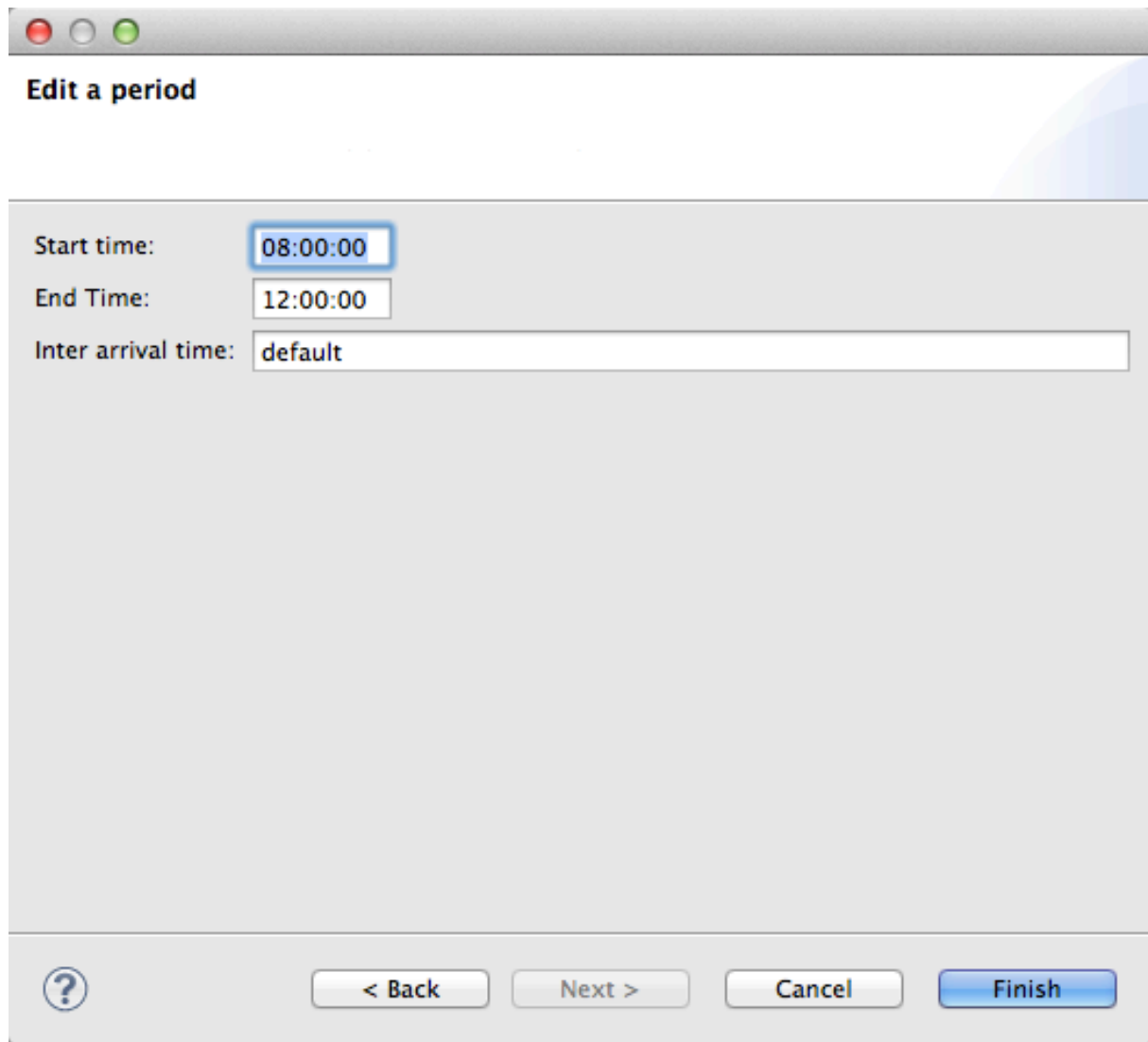
Hier kann in der obersten Zeile der Name des zuvor ausgewählten Tagesprofils geändert werden. Darunter gibt es Texteingabefelder um eine neue Periode zu dem Tagesprofil hinzuzufügen. Das hinzufügen funktioniert wie zuvor über den „Add“ Button der sich darunter befindet.

Die beiden Buttons befinden sich direkt unter den Feldern für die neue Periode damit diese über den Button „Add“ hinzugefügt werden können.

Unten auf dem Wizard befindet sich eine Liste der derzeitigen Perioden dieses Profils. Wenn eine ausgewählt ist kann man durch Klicken des „Delete“ Buttons, der sich neben dem „Add“ Button befindet, diese löschen.

Nach auswählen einer Periode kann man durch Klicken des „Next“ Buttons auf die dritte und letzte Seite des Wizards gelangen.

Um die Implementierung zu erleichtern wurden das Erstellen einer neuen Periode und das Bearbeiten einer bereits vorhandenen Periode sowohl optisch, als auch bei der Implementierung, durch die Verteilung auf zwei Klassen, von einander getrennt.



Edit a period

Start time: 08:00:00

End Time: 12:00:00

Inter arrival time: default

? < Back Next > Cancel Finish

Auf dieser Wizard Seite ist es möglich die zuvor ausgewählte Periode zu ändern. In den Textfeldern der drei Attribute Startzeit, Endzeit und mittlere Verteilung steht die aktuelle Belegung dieser Variablen. Es hätten auch alle Felder leer sein können, doch auf diese Art muss der Benutzer sich nicht merken, welche Periode ausgewählt wurde, falls er nur den Start- bzw Endzeitpunkt ändern möchte.

vi. Aktorkalender-View

Wenn der Benutzer auf den „Actor Calendar“ Button klickt öffnet sich die folgende View.

Organization Environment File

Select an actor

◀ January ▶

Mo	Di	Mi	Do	Fr	Sa	So
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Diese sieht der Prozesskalender-View sehr ähnlich. Die Unterschiede sind, dass statt des Usage-Files hier ein Organization Environment File ausgewählt wird und darunter ein Actor aus der comboBox ausgewählt werden muss. Der unterste Button „Process Calendar“ wechselt wieder in die Prozesskalender-View. Außerdem fehlt hier der „Persist“ Button.

Der Übersicht halber wurden alle Elemente identisch angeordnet, damit der Wechsel zwischen den zwei Sichten fließend geschieht.

Bei Doppelklick auf einen Tag im Kalender öffnet sich ein identischer Wizard wie beim Prozesskalender.

vii. Tagesprofile Wizard

Bei einem Klick auf den Button „Day profiles“ öffnet sich der Tagesprofile Wizard. Die erste Seite von diesem Wizard ist identisch zu der ersten Seite des Tagesprofil Wizards beim Prozesskalender.

Die zweite und dritte Seite unterscheiden sich jedoch in einer Einzelheit. Und zwar wird beim Aktorkalender keine mittlere Verteilung bei den Perioden eingegeben. Die Attribute der Perioden sind lediglich Start- und Endzeitpunkt. Eine mittlere Verteilung gibt es hierbei nicht.

Auch hier wurde bewusst, das Design der Wizards beim Prozesskalender und Aktorkalender identisch gewählt um die Handhabung zu erleichtern.

Mac OS window with title bar (red, yellow, green buttons).

Edit a DayProfile, press the 'Add'-Button to add a new Period

Day Profile Name:

Period

Start Time Point:

End Time Point:

Existing Periods:

08:00:00 - 12:00:00

13:00:00 - 18:00:00

Navigation bar:

Mac OS window with title bar (red, yellow, green buttons).

Edit a period

Here you can edit a period

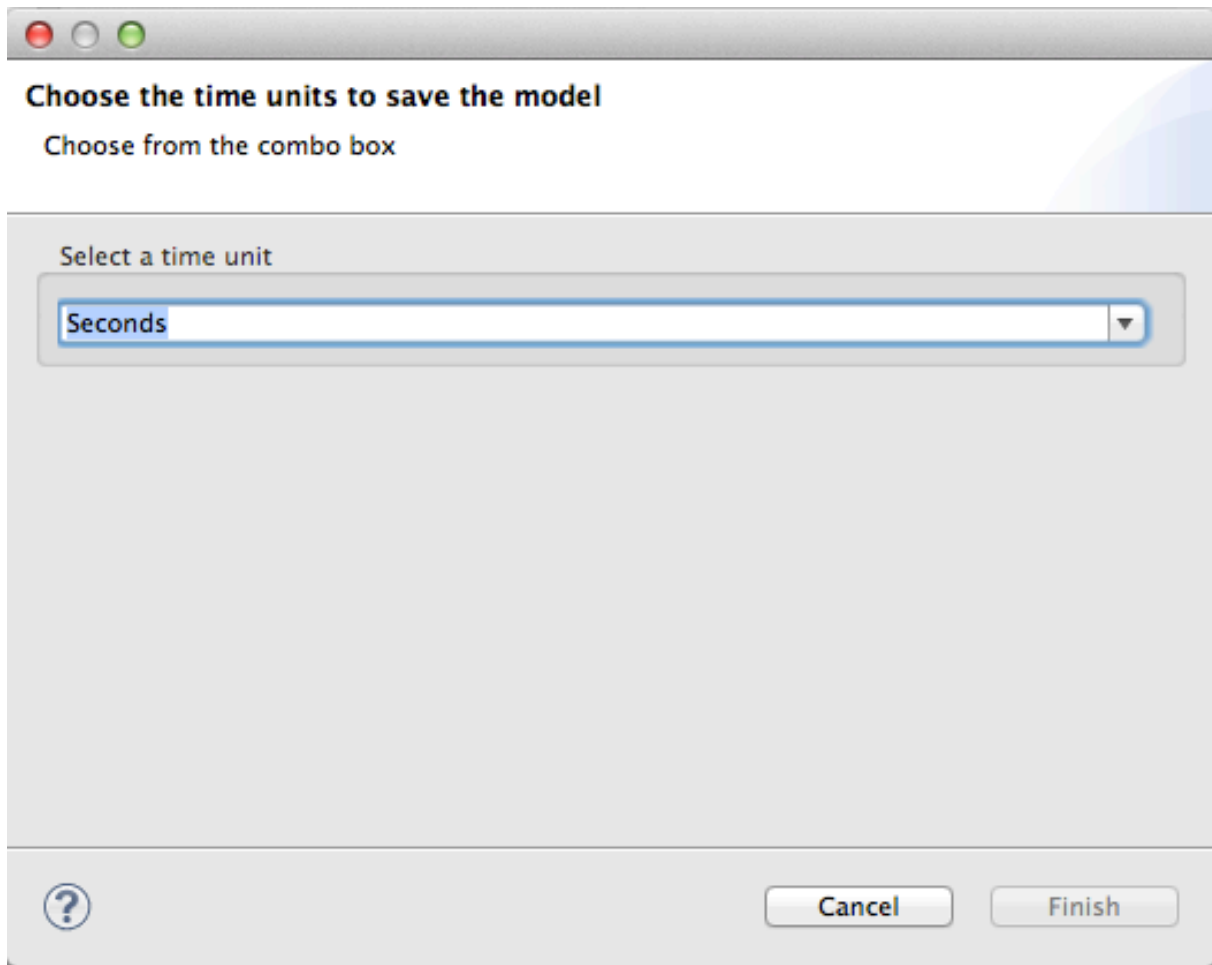
Start time:

End Time:

Navigation bar:

viii. Zeiteinheit Wizard

Wenn auf den Button „Persist“ geklickt wurde öffnet sich ebenfalls ein Wizard.



In diesem Wizard kann in einer Combo Box die Zeiteinheit ausgewählt werden, in der die Modelle gespeichert werden sollen.

Hier hätte auch eine Liste gewählt werden können, die direkt alle verfügbaren Zeiteinheiten anzeigt, doch da immer nur die aktuelle relevant ist wurde diese Art von Liste gewählt.

VI. Implementierung

Im Folgenden wird das Vorgehen bezüglich der Implementierung erläutert. Die Reihenfolge entspricht der Reihenfolge, die beim Programmieren gewählt wurde.

i. Kalender

Zu Beginn wurde zunächst der Kalender visualisiert, der auf beiden Views auftaucht. Hierfür wurden zwei Composite-Instanzen verwendet, die eine ist die obere Zeile, die den Monat enthält (*hier*: Januar) und die andere ist der Kalender, der den ausgewählten Monat darstellt (*hier*: 1. Januar = Montag).

Die erste Instanz beinhaltet zwei Buttons und ein Label. Mit den Buttons kann der jeweilige Monat ausgewählt werden und das Label gibt den aktuell dargestellten Monat an.

Die zweite Instanz beinhaltet eine Tabelleninstanz vom Typ Table. Sie hat sieben Spalten die von einem ContenProvider, je nach zugehörigem Monat, befüllt werden.

ii. Tage und Tagesprofile

Im nächsten Schritt wurde die Klasse DayProfile erstellt um wie in den Anforderungen beschrieben die Möglichkeit zu schaffen verschiedene Tagesprofile zu erstellen. Diese Klasse hat zwei Attribute, einen Namen und eine Liste von Perioden. Diese Perioden beinhalten Start- und Endzeitpunkte (beim Prozesskalender auch eine mittlere Verteilung) wann Prozesse an einem Tag dieses Tagesprofils starten könnten. Parallel dazu ist die Klasse Day entstanden. Diese Klasse enthält zwei Attribute. Eine ID, über die der jeweilige Tag von 364 übrigen Tagen im Jahr unterschieden werden kann und ein Tagesprofil (vgl. Klassendiagramm).

iii. Wizards

Um die Tage und Tagesprofile bearbeiten zu können, wurden Wizards verwendet, um auf der Benutzeroberfläche verschiedene Änderungen vornehmen zu können.

Beim Tages Wizard (vgl. Abschnitt III. ii.) erscheint eine dynamische Liste, die alle derzeitigen Tagesprofile enthält. Aus dieser Liste kann nun ein neues Tagesprofil für diesen Tag gewählt werden und mit klick auf „Finish“ wird diese Auswahl bestätigt und temporär gespeichert. Das bedeutet nur zur Laufzeit und wird nicht ins eventuell geladene Modell geschrieben.

Beim Tagesprofil Wizard (vgl. Abschnitt III. iii.) erscheint auf der ersten Seite auf die gleiche Weise eine identische Liste der derzeitigen Tagesprofile. Wird ein neues Tagesprofil benannt und mit „Add“ hinzugefügt, so wird es sofort an die dynamische Liste angehängt und auf dem Display dargestellt. Auch beim Löschen eines Tagesprofils wird die Änderung sofort wirksam und sichtbar.

Auf der zweiten Wizard Seite hat der Benutzer die Möglichkeit den Namen des Tagesprofils zu ändern. Die Änderung wird übernommen sobald etwas eingegeben wurde.

Außerdem kann eine neue Periode erstellt werden, dafür muss ein Startzeitpunkt, ein Endzeitpunkt und eine mittlere Verteilung angegeben werden. Sowohl Start- als auch Endzeitpunkt sind vom Typ „SimpleDateFormat“. Die Formatierung („hh:mm:ss“) muss eingehalten werden. Die mittlere Verteilung ist ein String, hier wird jede beliebige Eingabe akzeptiert.

Diese neue Periode wird dann bei den bereits existierenden Perioden unten an die Liste angehängt. Auch dies geschieht sofort nach Betätigung des „Add“ Buttons. Bei fehlerhafter Eingabe gibt es eine Fehlermeldung

Diese Liste hat als Listeneinträge Listen, bei denen die drei Attribute Startzeitpunkt, Endzeitpunkt und mittlere Verteilung aneinandergehängt werden. Ebenfalls dynamisch, wobei diese Listen in der Liste niemals mehr als drei Elemente haben.

Wenn eine Periode aus der Liste ausgewählt wurde besteht die Möglichkeit diese mit dem „Delete“ Button zu löschen oder, durch klicken auf den „Next“ Button, sie auf der nächsten Wizard Seite zu bearbeiten.

Auf der letzte Wizrad Seite stehen die bisherigen Belegungen der Variablen in den Texteingabefeldern. Bei einer Änderung wird diese sofort wirksam. Bei einer Eingabe die sich mit bereits bestehenden Perioden überschneidet gibt es sofort eine Fehlermeldung. Ebenso bei einer Verletzung des Formats. Zum Beispiel bei Eingabe von Buchstaben für die Zeitpunkte.

iv. Aktorkalender

Danach wurde ein Aktorkalender implementiert. Dieser ist dem Prozesskalender sehr ähnlich, allerdings gibt es bei den Tagesprofilen des Aktorkalenders keine mittlere Verteilung. Alle übrigen Implementierungen funktionierten bis hierher identisch, da die übrigen Funktionalitäten gleich bleiben.

v. Usagemodel und Organizationenvironmentmodel

Zum Ende der Implementierung gab es noch die Anforderung, dass die Perioden, die über die Tagesprofile den einzelnen Tagen zugeordnet sind, nach Beendigung des Programms gespeichert bleiben sollten.

Im Prozesskalender wurde es daher ermöglicht ein Usagemodel zu laden und anschließend ein Szenarium innerhalb dieses Modells auszuwählen. Durch diese Auswahl werden die im Modell stehenden Perioden sofort bezüglich der Zeiteinheit umgerechnet und die unterschiedlichen Perioden werden dynamisch dem Tagesprofil „Working Day“ zugeordnet. Um die im Anschluss geänderten Perioden abzuspeichern wird der „Persist“ Button geklickt und die Perioden werden in das Model überschrieben und damit gespeichert.

Analog dazu gibt es im Aktorkalender ein Organizationenvironmentmodel bei dem ein Akteur aus einer Combo Box ausgewählt werden kann. Auch hier werden die Perioden sofort geladen. Durch klicken des „Persist“ Buttons werden die Perioden in das ausgewählte Modell überschrieben.

Der Wizard der beim Klicken des Buttons geöffnet wird enthält eine Combo Box in der noch die Zeiteinheit ausgewählt werden kann in welcher die Modelle gespeichert werden sollen. Als Initialisierungswert sollen immer Sekunden ausgewählt sein. Der Benutzer kann allerdings noch zwischen Millisekunden, Minuten und Stunden wählen.

Bei klicken des „Finish“ Buttons werden dann alle Perioden in die geforderte Zeiteinheit umgerechnet und gespeichert.

Da die Zeiteinheit allgemeingültig für das ganze Projekt ist, werden immer alle Szenarien und Akteure im Hintergrund geladen um diese Perioden falls die Zeiteinheit geändert wurde gegebenenfalls anzupassen.

Die aktuelle Zeiteinheit des Projekts ist in einer Textdatei gespeichert.

VII. Testspezifizierung

Nach dem das Programm erfolgreich fertig implementiert wurde muss es nun noch ausführlich getestet werden.

Im Folgenden werden alle die Testfälle spezifiziert auf die das Programm getestet wurde.

i. Kalender

1. Testfall: Ändern des Monats

Beim Klicken der beiden Buttons links und rechts von dem Label der den Monat angibt ändert sich der Monatsname immer in der richtigen Reihenfolge und wird richtig und vollständig angezeigt. Auch der Kalender darunter ändert sich immer richtig mit.

Testschritt	Eingabe	Ausgabe
1	Button links vom Monatsnamen wird geklickt	
	Monatsname ist gesetzt	Monatsname entspricht dem Monat unmittelbar vor dem vorherigen Monat
	Kalender entsprechend zu Monatsnamen	Kalender entsprechend zu Monatsnamen
2	Button rechts vom Monatsnamen wird geklickt	
	Monatsname ist gesetzt	Monatsname entspricht dem Monat unmittelbar nach dem vorherigen Monat
	Kalender entsprechend zu Monatsnamen	Kalender entsprechend zu Monatsnamen

Ergebnis:

Der Test war erfolgreich. Für jeden beliebigen ursprünglichen Monat ändern sich der Monatsname und die Kalendertabelle, in beide Richtungen, immer korrekt. Die Kalendertabelle wird vollständig angezeigt.

ii. Tages Wizard

2. Testfall: Richtiges Datum im Tages Wizard

Wenn man auf unterschiedliche Tage in unterschiedlichen Monaten doppelt klickt, dann öffnet sich der Tages Wizard und darin wird oben das richtige Datum angegeben.

Testschritt	Eingabe	Ausgabe
1	Auf der View befindet sich ein gültiger Kalender	Wizard wird geöffnet
		Datum entspricht dem ausgewählten Datum

Ergebnis:

Der Test war erfolgreich. Für alle Tage im Kalender öffnet sich bei Doppelklick der Tages Wizard mit dem entsprechenden Datum.

3. Testfall: Änderung des Tagesprofils eines Tages bleibt erhalten

Wenn das Tagesprofil des Tages geändert wurde, dann bleibt die Änderung erhalten, nachdem der Wizard geschlossen und erneut geöffnet wurde.

Testschritt	Eingabe	Ausgabe
1	Ein Tagesprofil wird ausgewählt	Die Auswahl wird angezeigt
2	Der Finish Button wird geklickt	Der Tages Wizard wird geschlossen
3	Der selbe Tages Wizard wird erneut geöffnet	Das Tagesprofil entspricht dem zuvor Ausgewählten.

Ergebnis:

Der Test war erfolgreich. Für beide Kalender, Prozess- und Aktorkalender gilt, dass bei der Änderung des Tagesprofils die Änderung erhalten bleibt nach Schließen des Wizards mit dem Finish Button.

iii. Tagesprofile Wizard

4. Testfall: Hinzufügen von Tagesprofilen

Wenn auf den „Day profiles“ Button geklickt wurde öffnet sich die erste Wizard Seite. Wenn dann ein neuer Name für ein neues Tagesprofil eingegeben und anschließend auf „Add“ gedrückt wurde, wird das neue Tagesprofil sofort übernommen und unten in die dynamische Liste angehängt.

Testschritt	Eingabe	Ausgabe
1	Der Day profiles Button wird geklickt	Die erste Wizard Seite wird geöffnet
2	Die Liste aller existierender Tagesprofile wird angezeigt	
	Es wird ein neuer Name für ein neues Tagesprofil eingegeben	
3	Der Add Button wird geklickt	Das neue Tagesprofil ist das unterste Element der Liste der existierenden Tagesprofile

Ergebnis:

Der Test war erfolgreich. Nach Anlegen eines neuen Tagesprofils wird dieses in der Liste angezeigt.

5. Testfall: Löschen von Tagesprofilen

Mit „Delete“ wird das zuvor ausgewählte Tagesprofil gelöscht.

Testschritt	Eingabe	Ausgabe
1	Ein Tagesprofil wird ausgewählt	Ein Tagesprofil ist ausgewählt/markiert
2	Der Delete Button wird geklickt	Das Tagesprofil ist gelöscht (erscheint nicht mehr in der Liste)

Ergebnis:

Der Test war erfolgreich. Das ausgewählte Tagesprofil steht nach dem Löschen nicht mehr in der Liste. Die übrigen Listenelemente rücken auf.

6. Testfall: Zweite Wizard Seite öffnen

Der Tester befindet sich auf der ersten Wizard Seite. Wenn ein Tagesprofil ausgewählt wurde und danach der „Next“ Button geklickt wird, dann öffnet sich die nächste Wizard Seite und oben in dem Texteingabefeld für den neuen Namen steht der Name des zuvor ausgewählten Tagesprofils.

Testschritt	Eingabe	Ausgabe
1	Ein Tagesprofil wird ausgewählt aus der Liste aller existierenden Tagesprofile	Das Tagesprofil ist ausgewählt/markiert
2	Der Next Button wird geklickt	Die zweite Wizard Seite wird geöffnet
		Im Textfeld für den Namen des Tagesprofils steht der aktuelle Name des Tagesprofils

Ergebnis:

Der Test war erfolgreich. Die zweite Wizard Seite ist geöffnet und der Name des zuvor ausgewählten Tagesprofils steht im Texteingabefeld.

7. Testfall: Anlegen einer neuen Periode

Wenn eine neue Periode in der korrekten Formatierung angelegt wird, dann wird diese sofort unten in die Liste der existierenden Perioden angehängen.

Testschritte	Eingabe	Ausgabe
1	Textfeld des Startzeitpunktes existiert	Eingabe des Startzeitpunktes
	Textfeld des Endzeitpunktes existiert	Eingabe des Endzeitpunktes
	Textfeld der mittleren Verteilung existiert	Eingabe der mittleren Verteilung
2	Der Add Button wird geklickt	Die neue Periode wird an die Liste der existierenden Perioden angehängt

Ergebnis:

Der Test war erfolgreich. Nach dem Anlegen einer neuen Periode wird diese an die Liste der existierenden Perioden unten angehängt.

8. Testfall: Löschen einer Periode

Wenn eine Periode ausgewählt wurde, wird diese durch anschließendes Klicken auf den Button „Delete“ gelöscht.

Testschritt	Eingabe	Ausgabe
1	Periode wird ausgewählt	
	Delete Button wird geklickt	Periode steht nicht mehr in der Liste der Perioden.

Ergebnis:

Der Test war erfolgreich. Die ausgewählte Periode steht nicht mehr in der Liste der Perioden.

9. Testfall: Format Fehlermeldung

Wenn statt Zahlen, Buchstaben in die Start- bzw. Endzeit eingegeben werden oder Zahlen in einem anderen Format als "hh:mm:ss", dann gibt es einen regex-Fehler (Format Fehler) und die Fehlermeldung lautet: „invalid period definition“.

Testschritt	Eingabe	Ausgabe
1	Startzeitpunkt Textfeld existiert	Der Startzeitpunkt wurde in fehlerhaftem Format eingegeben
	Endzeitpunkt Textfeld existiert	Der Endzeitpunkt wurde in fehlerhaftem Format eingegeben
	Mittlere Verteilung Textfeld existiert	Die mittlere Verteilung wurde eingegeben
2	Der Add Button wurde geklickt	Die Fehlermeldung „invalid period definition“ erscheint
		Die Periode wurde nicht hinzugefügt

Ergebnis:

Der Test war erfolgreich. Die Ausgegebene Fehlermeldung in der Kopfzeile des Wizards lautet "invalid period definition". Die Periode wurde nicht hinzugefügt.

10. Testfall: Ungültige Periodendauer Fehlermeldung

Falls bei einer Periode der Endzeitpunkt vor dem Startzeitpunkt liegt, bzw. gleichzeitig, dann darf diese Periode auch nicht hinzugefügt werden. Als Fehlermeldung erhält man: „end time is earlier than start time“.

Testschritt	Eingabe	Ausgabe
1	Startzeitpunkt Textfeld existiert	Der Startzeitpunkt wurde im richtigen Format eingegeben
	Endzeitpunkt Textfeld existiert	Der Endzeitpunkt wurde vor oder gleich dem Startzeitpunkt definiert
	Mittlere Verteilung Textfeld existiert	Die mittlere Verteilung wurde eingegeben
2	Der Add Button wird geklickt	Die Fehlermeldung „end time is earlier than start time“ erscheint
		Die Periode wurde nicht hinzugefügt

Ergebnis:

Der Test war erfolgreich. Die Fehlermeldung erscheint in der Kopfzeile, die Periode wurde nicht hinzugefügt.

11. Testfall: Periode überlappt sich mit einer anderen

Die dritte Möglichkeit eine falsche Periode einzugeben ist, wenn sie sich mit einer bereits existierenden Periode überlappt. In diesem Fall gibt es die Fehlermeldung: „this period overlaps with an already existing period“.

Testschritt	Eingabe	Ausgabe
1	Startzeitpunkt Textfeld existiert	Der Startzeitpunkt wurde eingegeben
	Endzeitpunkt Textfeld existiert	Der Endzeitpunkt wurde eingegeben
	Mittlere Verteilung Textfeld existiert	Mittlere Verteilung wurde eingegeben
		Eingegebene Periode überlappt sich mit einer bereits existierenden Periode
2	Add Button wird geklickt	Periodenliste ist unverändert
		Fehlermeldung „this period overlaps with an already existing period“ erscheint

Ergebnis:

Der Test war erfolgreich. Die Periode wird nicht hinzugefügt.

Die Fehlermeldung lautet: "this period overlaps with an already existing period".

12. Testfall: Dritte Wizard Seite öffnen

Wenn eine Periode aus der Liste ausgewählt wird und man anschließend auf den „Next“ Button drückt, dann öffnet sich die dritte Wizard Seite und in den Texteingabefeldern stehen die Start- und Endzeitpunkte sowie die mittlere Verteilung (beim Prozesskalender) der ausgewählten Periode.

Testschritt	Eingabe	Ausgabe
1	Eine Periode wird ausgewählt	Eine Periode ist ausgewählt/markiert
2	Next Button wird geklickt	Dritte Wizard Seite wird geöffnet
		Startzeitpunkt Textfeld hat die Belegung wie die ausgewählte Periode
		Endzeitpunkt Textfeld hat die Belegung wie die ausgewählte Periode
		Mittlere Verteilung Textfeld hat die Belegung wie die ausgewählte Periode

Ergebnis:

Der Test war erfolgreich. Die dritte Wizard Seite wurde geöffnet und in den Textfeldern stehen die richtigen Belegungen.

Auf der dritten Wizard Seite gelten die gleichen Bedingungen für eine veränderte Periode, wie für die neue oben. D.h. es gibt bei gleichen Fehlern die gleichen Fehlermeldungen.

Also können Testfall 8-10 analog für diese Seite wiederholt werden. Auch hier waren alle drei Tests erfolgreich.

13. Testfall: Schließen des Wizards

Wenn der „Finish“ Button gedrückt wird, schließt sich der gesamte Wizard und alle Änderungen werden gespeichert.

Testschritt	Eingabe	Ausgabe
1	Eine Änderung wird vorgenommen	
	Finish Button ist aktiviert und wird geklickt	Der Tagesprofil Wizard schließt sich

Ergebnis:

Der Test war erfolgreich. Der Wizard ist geschlossen, der Tester befindet sich auf der Prozesskalender View.

Alle bisherigen Testfälle, also Testfall 1-12, können analog wiederholt werden für den Aktorkalender. Zu beachten ist, dass die mittlere Verteilung weggelassen wurde. Ansonsten funktionieren alle Testfälle analog. Auch hier waren alle Tests erfolgreich und führten zu den gleichen Ergebnissen.

iv. Usagemodel

14. Testfall: Usagemodel einlesen

Zunächst wird ein Usagemodel eingelesen aus einem beliebigen Dateipfad, bzw. aus dem Workspace. Sobald es ausgewählt wurde, sollte die ComboBox aktiviert sein und es ist Möglich ein Szenarium auszuwählen.

Testschritt	Eingabe	Ausgabe
1	Usagemodel wird von beliebigem Dateipfad eingelesen	Usagemodel ist eingelesen
		Der Dateipfad steht im Textfeld
		ComboBox ist aktiviert
		Szenarium kann ausgewählt werden

Ergebnis:

Das Usagemodel wurde eingelesen. Die ComboBox wurde aktiviert und es ist möglich ein Szenarium auszuwählen.

15. Testfall: Szenarium auswählen

Ein Szenarium wird ausgewählt. Dann wird das wird das Szenarium dieses Modells eingelesen und die Perioden des Tagesprofils „WorkDay“ entsprechen nun den Perioden der eingelesenen Tage des Modells.

Testschritt	Eingabe	Ausgabe
1	Szenarium aus ComboBox wird ausgewählt	Szenarium ist ausgewählt/markiert
2	DayProfiles Button wird geklickt	Die erste Wizard Seite wird geöffnet
3	Tagesprofil WorkDay wird ausgewählt	Tagesprofil WorkDay ist ausgewählt/markiert
	Next Button wird geklickt	Die zweite Wizard Seite wird geöffnet
		Periodenliste umfasst alle Perioden aus dem eingelesenen Modell

Ergebnis:

Der Test war erfolgreich. Das Szenarium wurde ausgewählt. Die Perioden entsprechen denen im Modell.

16. Testfall: Änderungen nur im Prozesskalender, diese speichern

Der Kalender kann nun bearbeitet werden. Nach dem der Kalender bearbeitet wurde, allerdings der Aktorkalender nicht geöffnet wurde, oder zumindest kein Modell im Aktorkalender geladen wurde wird der „Persist“ Button gedrückt. Dadurch wird der Zeiteinheiten Wizard geöffnet, es kann eine Zeiteinheit ausgewählt werden und diese Auswahl wird mit „Finish“ bestätigt. Die aktuellen Tagesperioden werden gespeichert, d.h. im Modell werden alle Perioden zurückgesetzt und mit den neuen Perioden der einzelnen Tage überschrieben. Falls die Zeiteinheit sich von der alten unterscheidet werden die Modelle im Organizationenvironmentmodel ebenfalls überarbeitet.

Testschritt	Eingabe	Ausgabe
1	Mind. ein Tag wird geändert	
	Persist Button wird geklickt	Zeiteinheiten Wizard wird geöffnet
2	Eine Zeiteinheit wird aus der ComboBox ausgewählt	Eine Zeiteinheit ist ausgewählt/markiert
	Finish Button wird geklickt	Zeiteinheiten Wizard wird geschlossen
		Im Usagemodel sind die Änderungen übernommen
3	Ein Scenario wird eingelesen.	
	Ein Aktor wird eingelesen	Im Tagesprofil Working Day entsprechen die Perioden denen aus den Modellen, sie sind im korrekten Format

Ergebnis:

Der Test war erfolgreich. In den Modellen stehen die aktuellen Perioden in der richtigen Zeiteinheit.

17. Testfall: Änderungen im Prozesskalender und im Aktorkalender speichern

Nach dem beide Kalender bearbeitet wurden, wird der „Persist“ Button gedrückt. Dadurch wird der Zeiteinheiten Wizard geöffnet, es kann eine Zeiteinheit ausgewählt werden und diese Auswahl wird mit „Finish“ bestätigt. Die aktuellen Tagesperioden werden gespeichert, d.h. im ausgewählt Szenario und im ausgewählten Aktor werden alle Perioden zurückgesetzt und mit den neuen Perioden der einzelnen Tage überschrieben. Falls die Zeiteinheit sich von der alten unterscheidet werden die Perioden in den nicht geänderten Szenarien und Akteuren ebenfalls überarbeitet.

Testschritt	Eingabe	Ausgabe
1	Mind. ein Tag wird jeweils geändert	

	Persist Button wird geklickt	Zeiteinheiten Wizard wird geöffnet
2	Eine Zeiteinheit wird aus der ComboBox ausgewählt	Eine Zeiteinheit ist ausgewählt/markiert
	Finish Button wird geklickt	Zeiteinheiten Wizard wird geschlossen
		Im Usagemodel sind die Änderungen übernommen
		Im Organizationenvironmentmodel sind die Änderungen übernommen
3	Ein Scenario wird eingelesen.	
	Ein Aktor wird eingelesen	Im Tagesprofil Working Day entsprechen die Perioden denen aus den Modellen, sie sind im korrekten Format

Ergebnis:

Der Test war erfolgreich. In den Modellen stehen die aktuellen Perioden in der richtigen Zeiteinheit.

v. Organizationenvironmentmodel

Wenn in den Testfällen 13-15 jeweils das Usagemodel durch das Organizationenvironmentmodel ersetzt wird und das Szenarium durch den Aktor, dann lassen sich alle Testfälle für das Organizationenvironmentmodel auf der Aktorkalender View analog genauso wiederholen.

Alle Tests waren erfolgreich.

18. Testfall: Änderungen nur im Aktorkalender, diese speichern

Nach dem der Kalender bearbeitet wurde, allerdings im Prozesskalender kein Modell geladen wurde wird der „Persist“ Button gedrückt. Dadurch wird der Zeiteinheiten Wizard geöffnet, es kann eine Zeiteinheit ausgewählt werden und diese Auswahl wird mit „Finish“ bestätigt. Die aktuellen Tagesperioden werden gespeichert, d.h. im Modell werden alle Perioden zurückgesetzt und mit den neuen Perioden der einzelnen Tage überschrieben. Falls die Zeiteinheit sich von der alten unterscheidet werden die Modelle im Usagemodel ebenfalls überarbeitet.

Testschritt	Eingabe	Ausgabe
1	Mind. ein Tag wird geändert	
	Persist Button wird geklickt	Zeiteinheiten Wizard wird geöffnet
2	Eine Zeiteinheit wird aus	Eine Zeiteinheit ist

	der ComboBox ausgewählt	ausgewählt/markiert
	Finish Button wird geklickt	Zeiteinheiten Wizard wird geschlossen
		Im Organizationenvironmentmodel sind die Änderungen übernommen
3	Ein Scenario wird eingelesen	
	Ein Actor wird eingelesen	Im Tagesprofil Working Day entsprechen die Perioden denen aus den Modellen, sie sind im korrekten Format

Ergebnis:

Der Test war erfolgreich. In den Modellen stehen die aktuellen Perioden in der richtigen Zeiteinheit.

vi. conf.txt Datei

19. Testfall: Ablegen der conf.txt Datei im Projektordner

Nach dem das Programm mindestens einmal ausgeführt wurde und ein Projekt eingelesen wurde, muss danach in diesem Projekt auf der Ebene des UsageModels und des OrganizationEnvironmentModels eine Textdatei erstellt worden sein, die die aktuelle Zeiteinheit des Projekts abspeichert.

Testschritt	Eingabe	Ausgabe
1	Programm wird ausgeführt	
	Projekt wird eingelesen	Projekt ist eingelesen
2	Project Explorer wird geöffnet	Die Datei conf.txt befindet sich im Projektordner

Ergebnis:

Der Test war erfolgreich. Die Datei „conf.txt“ befindet sich im Projektordner.

20. Testfall: Datei conf.txt enthält die richtige Zeiteinheit

Nach dem ein Modell (zum Beispiel UsageModel) eingelesen, daraufhin abgespeichert und dadurch automatisch die Zeiteinheit gespeichert wurde, stehen in dem Modell alle Perioden in der richtigen Zeiteinheit.

Testschritt	Eingabe	Ausgabe
1	Usagemodel wird geladen	
	Persist Button wird geklickt	
	Zeiteinheit wird	

	ausgewählt	
2	Project Explorer wird geöffnet	
	Default.usagemodel wird geöffnet	Darin sind alle Perioden in der richtigen Zeiteinheit

Ergebnis:

Der Test war erfolgreich. Alle Perioden sind in der richtigen Zeiteinheit.

vii. Nicht-funktionale Anforderungen

21. Testfall: (1) Wizards öffnen

Alle Wizards sollen in weniger als 2 sec geöffnet werden.

Zum Beispiel der Tagesprofile Wizard. (Wiederhole den Test für alle Wizards)

Testschritt	Eingabe	Ausgabe
1	Day profiles Button wird geklickt	Tagesprofile Wizard ist nach weniger als 2 Sekunden geöffnet.

Ergebnis:

Der Test war erfolgreich. Alle Wizards haben sich in weniger als 2 Sekunden geöffnet.

22. Testfall: (2) Wizard schließen

Wenn ein Wizard geschlossen wird, geschieht dies ebenfalls in weniger als 2 sec.

Zum Beispiel der Tagesprofile Wizard. (Wiederhole den Test für alle Wizards)

Testschritt	Eingabe	Ausgabe
1	Cancel Button wird geklickt	Wizard ist nach weniger als 2 Sekunden geschlossen.

Ergebnis:

Der Test war erfolgreich. Alle Wizards haben sich nach weniger als 2 Sekunden geschlossen.

23. Testfall: (3) Modell laden

Ein Modell soll in weniger als 2 sec geladen sein, und die zugehörigen Szenarien bzw. Akteure sollen in der gleichen Zeit, in der ComboBox bereitgestellt werden.

Zum Beispiel ein UsageModel. (Wiederhole den Test auch für OrganizationEnvironmentModel)

Testschritt	Eingabe	Ausgabe
1	Usagemodel wird geladen	Usagemodel ist geladen

		ComboBox ist aktiviert und enthält Szenarien
--	--	--

Ergebnis:

Der Test war erfolgreich. Die Modelle sind in weniger als 2 Sekunden geladen.

24. Testfall: (4) Projekt speichern

Das abspeichern sollte allerhöchstens in 2 sec geschehen. Nach diesen 2 sec, sollte das Programm wieder voll funktionstüchtig sein.

Testschritt	Eingabe	Ausgabe
1	Projekt wird geladen	
	Persist Button wird geklickt	Projekt wird gespeichert und Programm ist wieder voll funktionsfähig

Ergebnis:

Der Test war erfolgreich. Das Programm ist wieder voll funktionsfähig in weniger als 2 Sekunden.

VIII. Erfahrungen

i. Persönliche Entwicklung

Mein abschließender Eindruck zu diesem Praktikum ist, dass es mir als Programmierer sehr viel weitergeholfen hat. Ich habe nicht nur gelernt, ein größeres Projekt selbstständig vorzubereiten und zu implementieren sondern auch mit verschiedenen Foren im Internet zu arbeiten und das Internet nach den Befehlen zu durchsuchen, die für mich relevant sind.

Außerdem habe ich gelernt mit der Debuggingfunktion von Eclipse umzugehen, die mir sehr oft sehr viel weiter geholfen hat.

Ich konnte mein Vorwissen aus fast allen bisherigen Informatikvorlesungen einbringen und vertiefen.

Ich weiß auch für die Zukunft, dass ich nun weniger abgeschreckt bin, wenn ich mit einem neuen Projekt konfrontiert werde, zu dem mir spontan keine mögliche Implementierung einfallen will, da ich nun gelernt habe, mich an so ein Projekt langsam heranzutasten.

Zwar habe ich die meisten Dinge, die ich hier angewendet habe, irgendwo schon mal gelernt (die meisten in der Vorlesung Einführung in die Software Engineering) doch da ich, wie die meisten Studenten, in Vorlesungszeiten von einem Übungszettel zum nächsten lebe, hatte ich erst jetzt die Gelegenheit mich mit diesen Werkzeugen intensiv zu beschäftigen und vertraut zu machen.

ii. Nutzen für Studium

Ich sehe in diesem Praktikum einen großen Nutzen für mein restliches Studium, da es eine sehr gute Vorbereitung auf die mir bevorstehende Bachelorarbeit war, da auch wenn ich dort kein Programm schreiben werde, ich trotzdem gezwungen bin mich mit einem Projekt über einen längeren Zeitraum zu befassen und mich intensiv damit auseinander zu setzen.

Außerdem war dieses Praktikum sehr viel „praktischer“ als die Vorlesungen die ich in den letzten Semestern belegt habe. Hier sehe ich eine engere Verbindung zum späteren Berufsleben als vergleichsweise bei trockenen Vorlesungen in denen ich Definitionen für Klausuren auswendig lerne.

IX. Schluss

Es wurde ein Programm entwickelt, in dem es beim Einsatz in Betrieben möglich ist, innerhalb eines Kalenders verschiedene Tagesprofile zu definieren mit zugehörigen Perioden, in denen Prozesse gestartet werden können.

Der Kalender umfasst genau ein Jahr.

Es ist möglich die Zeiteinheit auszuwählen, in der die Perioden für die jeweiligen Tage gespeichert werden und diese Zeiteinheit wird in einer Datei gespeichert und bleibt bis zur nächsten Änderung erhalten.

Ein Ansatz, wie dieses Programm weiterentwickelt werden könnte, wäre, dass ein zusätzliches Feature entwickelt werden könnte, mit dem bereits auf der Kalenderoberfläche sichtbar wäre, welches Tagesprofil der jeweilige Tag hat und aus Platzgründen könnte eventuell neben dem Kalender das Tagesprofil stehen mit den zugehörigen Perioden.

Auch könnte das Erstellen und Bearbeiten der Perioden visuell ansprechender gestaltet werden. Zum Beispiel über nebeneinander liegende Zahlenräder, in denen die Periode neu definiert wird. Auf diese Art würde auch direkt vermieden werden, dass bei der Eingabe fälschlicherweise Buchstaben oder Satzzeichen eingegeben würden.

Abschließend kann ich sagen, dass dieses Praktikum eine gute Vorbereitung für die noch folgenden Semester und das später Leben ist und ich auch froh bin, dieses Thema gewählt zu haben, da ich es sehr interessant fand. Besonders hat mir die Arbeit mit den Wizards gefallen, da ich zuvor noch nie eine Oberfläche implementiert habe, bei der man als Nutzer und Programmierer aktiv so viele Möglichkeiten hatte zu sehen, was da tatsächlich implementiert wurde.

X. Referenzen

Web-Adressen:

01. <http://www.eclipse.org>
02. <http://www.eclipse.org/swt/>
03. <http://www.vogella.com/articles/EclipseWizards/article.html>
04. <http://www.hs-weingarten.de/~keller/Downloads/grabo/java/JavaSWT.pdf>
05. <http://docs.oracle.com/javase/tutorial/uiswing/layout/visual.html>

Anhang: Klassendiagramm, Komponentendiagramm, Nutzungsdiagramm