

TITEL

Leitende Aufgabe Flight Management System
Abgabedokumentation Softwareengineering

ERSTELLE

Marc Landolt, Pascal Jenni, Romino Florio

DATUM

15. Juni 2008

VERSION

0.0.1

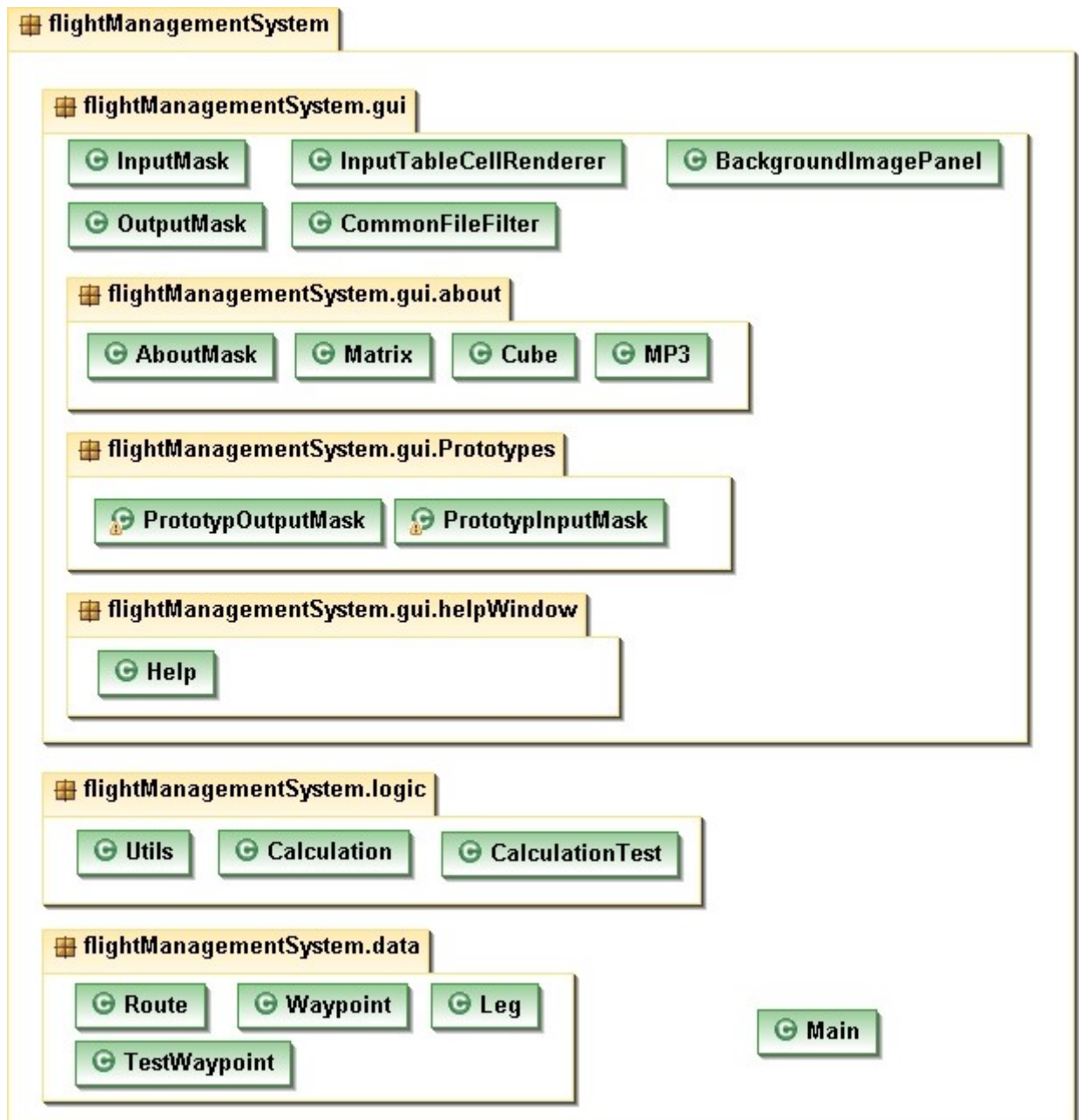
Leitende Aufgabe Flight Management System

Abgabedokumentation
Softwareengineering

INHALTSVERZEICHNIS

1	PAKETDIAGRAMM	3
2	KLASSENDIAGRAMM	4
3	ANWENDUNGSFALLDIAGRAMM	4
4	AKTIVITÄTSDIAGRAMM	7
4.1	WICHTIGER ABLAUF 1	7
4.2	WICHTIGER ABLAUF 2	7
5	SEQUENZDIAGRAMM	8
5.1	WICHTIGER ABLAUF 1	8
6	SOURCECODE	8
6.1	SOURCECODE KLASSEN	9
6.2	SOURCECODE AUF CD	9
7	PRINTSCREEN	10
8	BERICHT	11
8.1	TESTBERICHT	11
8.1.1	Funktion	11
8.1.2	Berechnung	11
8.1.3	Fehleingaben / Fehlerbehandlung	11
8.1.4	Hilfe	11
8.1.5	Navigation / Bedienerfreundlichkeit	11
8.2	ERFAHRUNGEN	11
8.3	ERKENNTNISSE	11
8.4	STOLPERSTEINE	11
	ANHANG	12

1 PAKETDIAGRAMM



2 KLASSENDIAGRAMM

2.1 ZUR DARSTELLUNG DES KLASSENDIAGRAMMS

Wir haben uns entschlossen (Haben wir Jungs?) das Klassendiagramm mit eUML2 zu erstellen, da dies einfach die beste Funktionalität hat. eUML2 ist ein Plugin von Eclipse (to eclipse = engl. in den Schatten stellen), welches es in einer Light Version gratis zur Verfügung steht. Nach vielen UML Tools, die wir getestet haben, fanden wir heraus, dass dies wohl das einzige frei verfügbare Tool ist, welches es erlaubt in Echtzeit am Diagramm und am Code Änderungen zu machen und das jeweils andere wird automatisch auch geändert. Auch ist dies wirklich ein Tool zum arbeiten, denn man kann auf eine beliebiges Attribut, eine beliebige Methode oder Klasse doppelklicken und das Codefenster im unteren Teil springt zur entsprechenden stelle im Code. So behält man einen guten Überblick.

Das ganze ohne lästige UML Marker im Code wie bei Netbeans. Auch bietet es im Zusammenhang mit Eclipse die Möglichkeit das UML Diagramm auf dem Subversion Server zu speichern, eine Funktion die bei Netbeans gänzlich fehlt, und auch der Versuch jeweils einfach ein neues UML vom Code mit den UML Markern zu generieren hat bei Netbeans versagt. Ein weiteres Feature ist, dass jede Methode welche eine kleine Unschönheit beinhaltet, z.B. eine Variable die instanziiert wird, aber nie gebraucht wird mit einem gelben Ausrufezeichen markiert wird. So lässt sich der Code auf einen wirklich sauberen Stand bringen.

Deshalb hier nun ein paar Erläuterungen zur Darstellung, im Prinzip ist es UML 2.0 einfach mit dem Zusatz, dass noch hilfreiche Symbole eingeblendet werden.

C: Class



I: Interface



Attribute (mit Loch):

privat: rotes Quadrat
 protected: gelbes Quadrat 45°
 gedreht
 public: grüner Kreis

Methoden:

Wie Attribute, aber gefüllt

static:

Mit hochgestelltem „S“

Unschönheiten im Code:

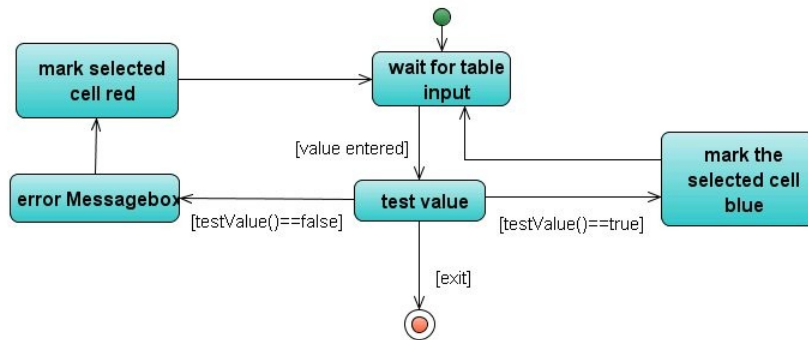
gelbes Ausrufezeichen

3 ANWENDUNGSFALLDIAGRAMM

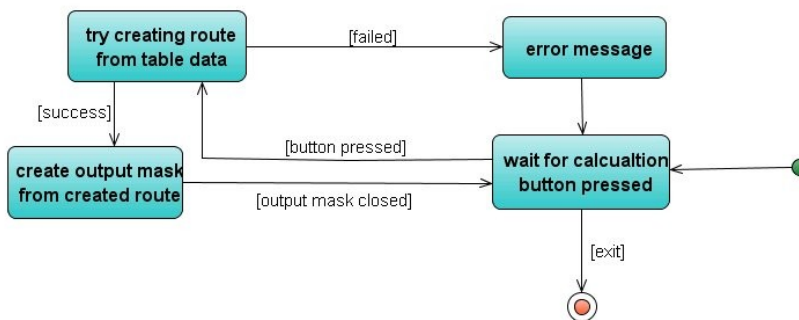
- Daten eingeben
- Route berechnen
- hilfe lesen
- Daten laden / speichern

4 AKTIVITÄTSDIAGRAMM

4.1 WICHTIGER ABLAUF 1

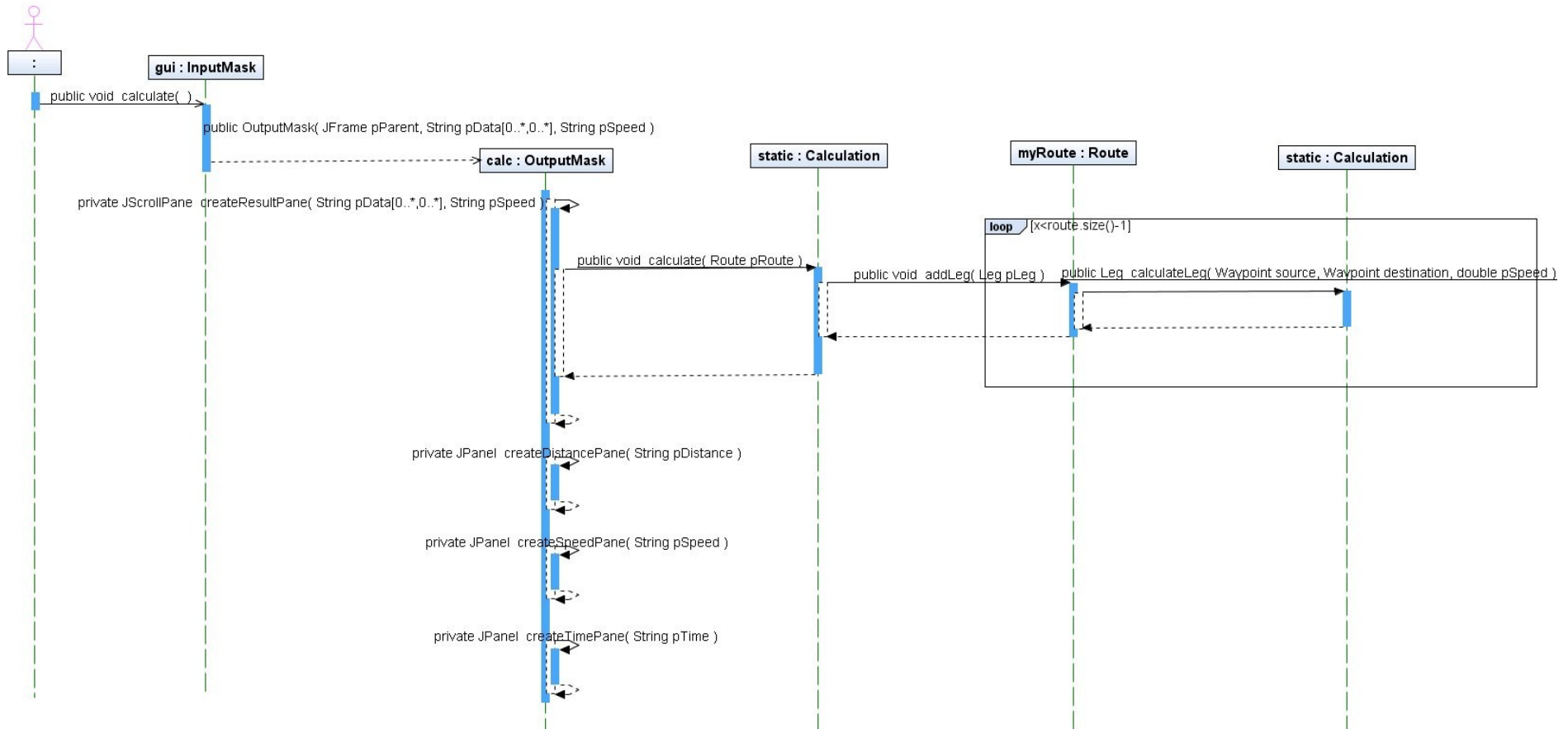


4.2 WICHTIGER ABLAUF 2



5 SEQUENZDIAGRAMM

5.1 WICHTIGER ABLAUF 1



6 SOURCECODE

Der Sourcecode teilt sich in die folgenden Klassen auf, welche in den Kapiteln ab 3.2 separat aufgeführt sind.

6.1 SOURCECODE KLASSEN

6.2 SOURCECODE AUF CD

7 PRINTSCREEN

The screenshot displays a Windows desktop environment with a blue map background. Several applications are open:

- Flight Management System:** A window with a menu bar (Datei, Optionen, Hilfe) and a speed input field set to 500 [Knoten]. It contains a table of waypoints:

Flughafen	Waypoint	Breitengrad	Breitengrad
Zürich-Kloten	LSZH	N47275	E008329
	VEBIT	N47161	E008004
	ROTOS	N47114	E007433
	ULMES	N46573	E004176
Genf	LSGG	N46143	E006066

 A 'Berechnen' button is located at the bottom right of this window.
- Flight Management System - Berechnung:** A window displaying calculated flight data:

Start Waypoint	Ziel Waypoint	Kurs	Distanz [NM]	Flugzeit [min]
LSZH	VEBIT	242.82	24.79	2.97
VEBIT	ROTOS	248.07	12.53	1.50
ROTOS	ULMES	265.51	140.79	16.90
ULMES	LSGG	240.80	86.36	10.36

 Below the table, there are input fields for speed (500 [Knoten]), distance (264.47 [NM]), and time (31.74 [min]).
- Flight Management System - Credits:** A window with a 3D wireframe cube and text:

==[Flightmanagementsystem]==
 Diese Applikation wurde konzipiert und entwickelt von:
 Romino Florio
 Pascal Jenni
 Marc Landolt
 Wir danken herzlich den Leuten von Javazoom für das...

The taskbar at the bottom shows the Start button, several open applications (Java EE - UML Class..., Flight Management S..., Abgabedokumentatio...), system tray icons (DE, 64%, network, volume), and the system clock (23:04).

8 BERICHT

Wir haben sehr gute Erfahrungen gemacht, insbesondere mit dem Subversion Server. Wir konnten jeweils gleichzeitig am Code arbeiten. Auch um zusammen Probleme zu lösen ist er sehr nützlich. Ausserdem sind so immer mindestens 4 relativ aktuelle Versionen im Umlauf, so dass man sich bezüglich Backup keine Sorgen machen muss.

Das Repository für Netbeans und Eclipse kann unter folgenden Adressen gefunden werden:

Server: `svn://filu.myftp.org/Gruppe1-LA-4`

Backup Server `svn://www.schizophrenie-forum.ch/Gruppe1-LA-4`

Ausserdem haben wir einen WEBSVN eingerichtet, auf dem man den Verlauf betrachten kann.

<http://filu.myftp.org/websvn/> dort kann man sehr schön nachvollziehen wer wann was hinzugefügt oder geändert hat.

Wir haben verschiedene Dinge ausprobiert, von denen wir danach teilweise wieder weg gekommen sind.

Hier bin ich grad noch dran...

8.1 TESTBERICHT

8.1.1 Funktion

8.1.2 Berechnung

8.1.3 Fehleingaben / Fehlerbehandlung

8.1.4 Hilfe

8.1.5 Navigation / Bedienerfreundlichkeit

8.2 ERFAHRUNGEN

8.3 ERKENNTNISSE

8.4 STOLPERSTEINE

ANHANG

SVN

TEST, PROTOKOLLE, ETC.

PROJEKTPLAN

ID	% Complete	Task Name	Duration	Start	F
25	0%	Berechnung	2 ewks	Mon 26.05.08	Mo
26	0%	Ausgabe	2 ewks	Mon 26.05.08	Mo
27	0%	Konsolen Interface	2 ewks	Mon 26.05.08	Mo
28	0%	Interface für GUI	1 ewk	Mon 26.05.08	Mo
29	0%	GUI	10 days	Mon 02.06.08	Mor
30	0%	GUI ab Interface	2 ewks	Mon 02.06.08	Mo