

26.02.2011

Figuren

Gabriel Pawlowsky,
Josef Sochovsky
4BHITS



Inhalt

Aufgabenstellung:	1
Designüberlegung.....	2
Arbeitsaufteilung + Aufwandsschaetzung:.....	4
Endzeitaufteilung	4
Testbericht.....	4
Alle Figuren + verschiedene Farben gezeichnet.....	4
Änderungen im Verlauf der neuen Aufgabe	4

Aufgabenstellung:

Erstellt gemeinsam eine Klasse `Figur`. Diese `Figur` soll folgende Attribute und Methoden besitzen:

- `x`-Koordinate: ganzzahliger Wert
- `y`-Koordinate: ganzzahliger Wert
- `farbe`: `Color`-Objekt
- passende Konstruktoren
- setter und getter-Methoden
- `draw`: Methode mit einem `Graphics`-Objekt als Parameter, die in der Klasse `Figur` nichts tut.
- `toString`

Diese Klasse soll als Basis für ein Programm dienen, bei dem zu einer graphischen Oberfläche verschiedene Figuren hinzugefügt werden können. Es soll möglich sein, die gewünschte Figur auszuwählen, ihre Werte festzulegen und sie dann zeichnen zu lassen. Zur Verwaltung der Figuren soll eine Liste dienen, die beliebig viele Figuren speichern kann.

Die Aufteilung im Team soll so erfolgen, dass ein Team-Mitglied sich um die Oberfläche und um die Figurenverwaltung kümmert während das zweite Team-Mitglied sich um die Erstellung von Sub-Klassen von `Figur` kümmert, die in der Oberfläche dargestellt werden können. Es sollen mind. 4 verschiedene `Figur`-Klassen existieren.

Gruppenarbeit: 2 Personen (Protokoll!)

Gesamtpunkte: 8

+++++

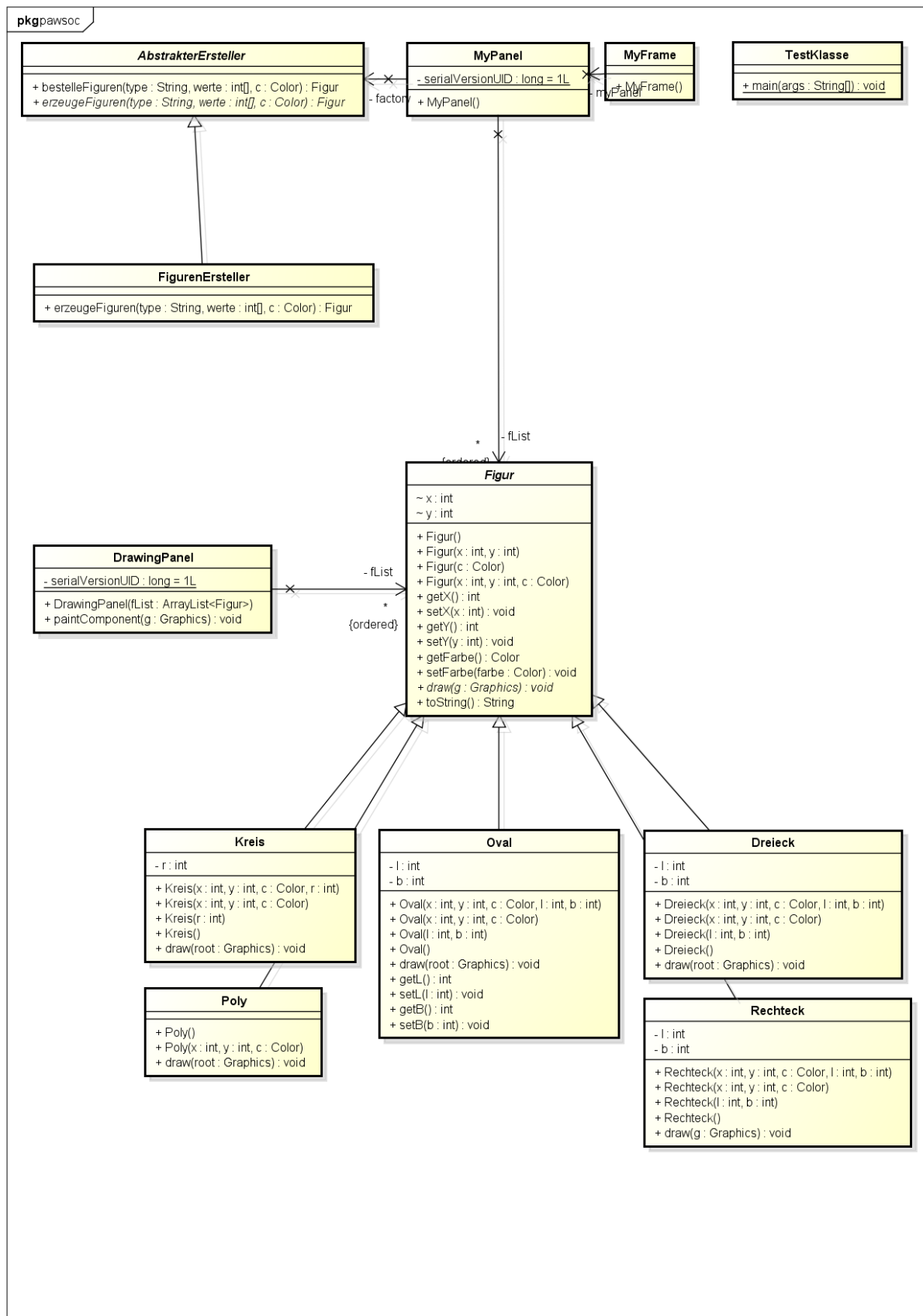
Wandelt eure Gruppenabgabe zu "JavaAufgab02 - Figuren" so um, dass das Factory-Method Pattern angewendet wird. Dokumentiert eure Änderungen auch entsprechend. Es muss im

Protokoll (besonders im UML-Klassendiagramm) nachvollziehbar sein was und warum geändert wurde.

Designüberlegung

Die einzige Designüberlegen, deren Anmerkung sinnvoll ist, ist dass wir ein eigenes DrawingPanel angelegt haben, in dem gezeichnet wird. Dieses wird in ein anderes Panel gemeinsam mit den Components für die Einstellungen eingebunden. Außerdem haben wir ganz ohne Applet und nur mit JPanels gearbeitet.

Fuer die Factory Method haben wir 2 Klassen angelegt und bearbeitet wurde die Klasse MyPanel.java der Rest wurde belassen wie er bei der letzten Abgabe war.



Arbeitsaufteilung + Aufwandsschaetzung:

Es ist eine genaue Arbeitsaufteilung in der Angabe beschrieben

Pawlowsky: Oberfläche 6 h

Sochovsky: Figurenverwaltung + Factory Method 7 h

Endzeitaufteilung

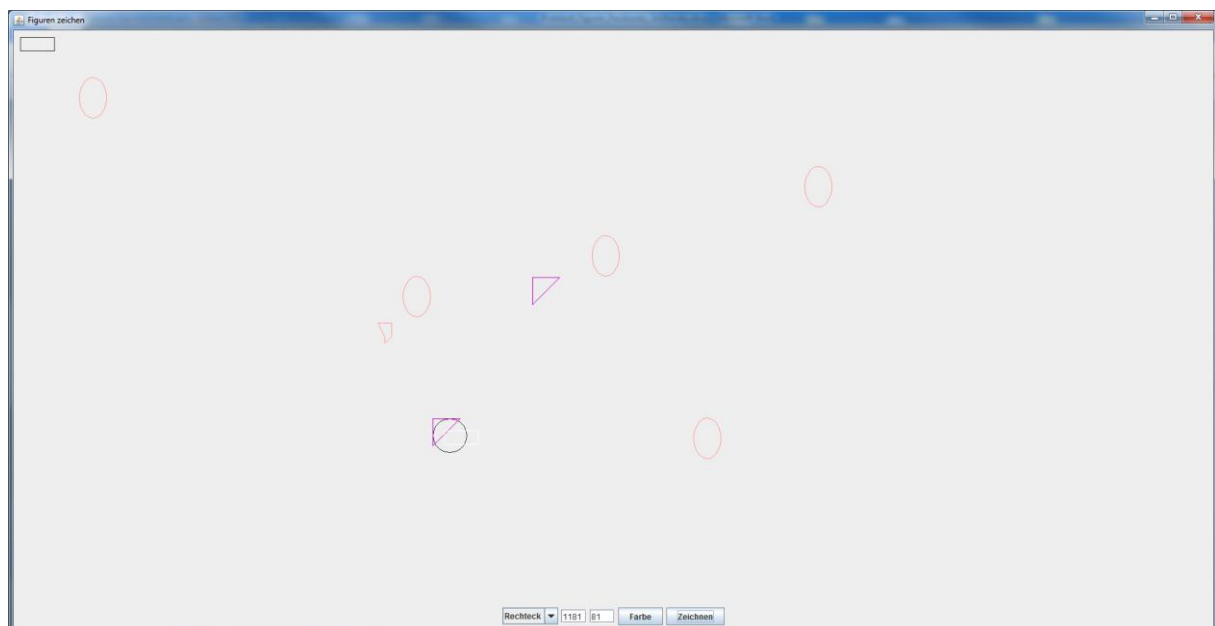
Pawlowsky: Oberfläche 6 h

Sochovsky: Figurenverwaltung + Factory Method 5 h

Testbericht

Getestet wurde die alte Abgabe nicht mehr nachdem diese 2 Punkte sowieso verloren sind, jedoch wurde bei der implementation der Factory Method ein Testing durchgefuehrt.

Alle Figuren + verschiedene Farben gezeichnet



Änderungen im Verlauf der neuen Aufgabe

Es wurden 2 Klassen angelegt:

- Die abstrakte Klasse: AbstrakterErzeuger
- Die Klasse die AbstrakterErzeuger extended: FigurenErzeuger

In AbstrakterErzeuger wurde die Methode bestelleFiguren implementiert, weil nicht mehrere Operationen die immer stattfinden werden ruft diese Methode nur erzeugeFiguren auf, diese Methode muss in der vererbten Klasse implementiert werden und enthält die ifelse Anweisungen die dann die jeweilige Figur erzeugen.

Außerdem wurde in der Klasse MyPanel die Art der Objekterzeugung verändert und es wird im Konstruktor am Anfang ein Objekt von FigurenErzeuger erstellt.