

Praktikum Design Patterns

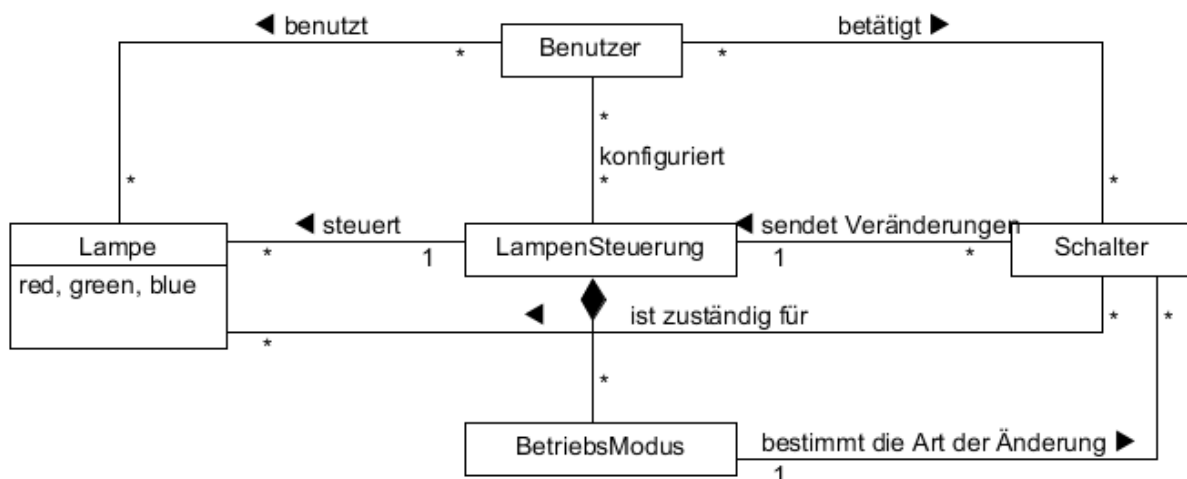
1 Problembeschreibung

Eine Lampensteuerung ist mit mehreren „intelligenten“ Lampen verbunden und kann so ihre Farbe und Leuchtstärke einstellen. Pro Farbkomponente kann ein Wert von 0% bis 100% eingestellt werden. Ebenfalls an der Lampensteuerung angeschlossen sind Schalter, die sowohl gedreht wie auch gedrückt werden können. Diese Schalter kennen ihre eigene Position oder ihren Schaltzustand nicht und melden der Lampensteuerung nur, dass sie entweder gedrückt oder gedreht wurden.

Im Rahmen dieses Praktikums verbessern Sie eine bestehende Lösung für die Ansteuerung der Lampen. Dazu gehören dann auch 2 Betriebsmodi, die steuern, wie die Tasten die Lampen ein- und ausschalten sowie ihre Leuchtstärke verändern.

Zentral bei diesem Praktikum ist der Einsatz geeigneter Entwurfsmuster. Es sollen neben den GRASP auch die GoF Entwurfsmuster eingesetzt werden.

Domänenmodell:



Jedem Schalter sind fix Lampen zugeordnet, die von ihm gesteuert werden. Pro Schalter kann einer der folgenden Betriebsmodi eingestellt werden:

1. **Arbeitsplatz:** Auf Druck des Schalters wird zwischen 0% und 100% umgeschaltet. Sollte der Wert dazwischen liegen, wird 0% gesetzt. Eine Drehung des Schalters erhöht resp. reduziert die Helligkeit um 10%.
2. **Wohnung:** Auf Druck des Schalters wird zwischen 0%, 45%, 70%, 85% und 100% umgeschaltet. Sollte der Wert dazwischen liegen, wird die nächst höhere Stufe gewählt, und wenn bereits 100% erreicht ist, wird auf 0% gesetzt. Eine Drehung des Schalters erhöht resp. reduziert die Helligkeit um 5%.

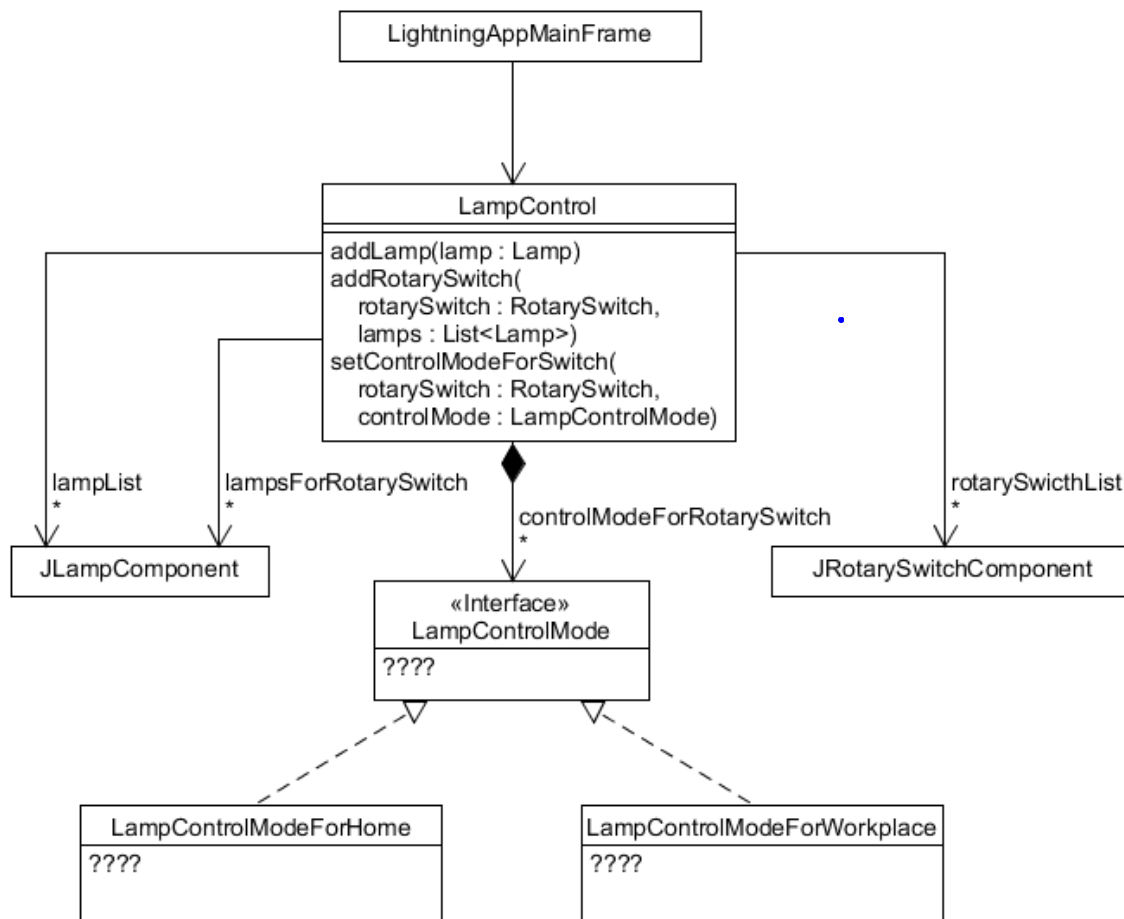
Diese 2 Betriebsmodi können an der Steuerung pro Schalter beliebig gewählt werden.

Sie erhalten eine lauffähige Lösung, die aber noch nicht korrekt funktioniert und grössere Design-Schwächen enthält. Die Lampen und die Schalter werden mit Swing-Komponenten simuliert, was in Ihrer Lösung auch so bleiben soll. Die Funktionalität der Steuerung beschränkt sich aber darauf, die Lampen ein- respektive auszuschalten.

2 Aufgaben

2.1 Entwurf (2 P)

Das nachfolgende Design-Klassendiagramm visualisiert den Code, den Sie als Basis erhalten haben .



Ihre Aufgabe besteht darin, die folgenden Ergänzungen und Verbesserungen umzusetzen:

1. Der Schalter meldet seine Änderungen direkt der Lampensteuerung. Obwohl der Schalter kein richtiges UI Element sondern nur die Simulation einer HW Komponente ist, sollte diese direkte Abhängigkeit vom Schalter zur Lampensteuerung vermieden werden.
2. Definieren Sie für die Lampe und den Schalter Abstraktionen, für die später die Software Klassen der richtigen Lampen und Schalter verwendet werden können.
3. Die 2 Betriebsmodi müssen vollständig realisiert werden.
4. Fügen Sie noch einen beliebigen 3. Betriebsmodus hinzu. Es geht hier nur darum zu zeigen, dass dies mit Ihrer Lösung einfach geht.

Entwerfen Sie Lösungen für die obigen 4 Punkte. Zeichnen Sie geeignete Klassen- und Interaktionsdiagramme, um Ihre jeweiligen Design Ideen zu erklären. Beziehen Sie dabei geeignete Entwurfsmuster in Ihre Lösung ein und zeigen Sie auf, wo Sie welche Entwurfsmuster eingesetzt haben.

2.2 Implementation (1 P)

Setzen Sie nun Ihre Design Ideen um und verbessern und erweitern Sie den bestehenden Code. Dabei soll die Steuerung auch mit automatischen Unit-Tests ausreichend getestet sein.

3 Abgabe

Bitte erstellen Sie ein einziges PDF-Dokument, das alle Diagramme mit allfälligen Erklärungen und den gesamten Code inklusive Tests enthält. Das ist alles, was Sie abgeben. Den Quelltext müssen/sollen Sie nicht separat abgeben.

Seien Sie bereit, den Code dem Dozenten zu demonstrieren und zu erklären.

Lampe abschalten

