



Übungsblatt 3: GUIs

25 Punkte möglich

Aufgabe 1: Aus dem Skript

[2 Punkte] Liefern Sie alle Quellcodes aus dem Skript, Kapitel 4 und 5 in einem Eclipse-Workspace als ZIP-Datei bei Prof. Dopatka ab. Der Eclipse-Workspace muss eine wohl strukturierte Package-Struktur aufweisen. Sind Quellcodes fehlerhaft oder unvollständig, so erhalten Sie keine Punkte.

Aufgabe 2: Live-Coding

Bei der Abnahme des Aufgabentyps „Live-Coding“ erstellen Sie an einem einzigen PC bei Prof. Dopatka Quellcode ohne weitere Hilfsmittel. Der Eclipse-Workspace muss geöffnet, aber leer sein. Weitere Dokumente, online oder in Papierform, sind untersagt. Prof. Dopatka prüft anhand dieses Tests, ob er Ihrem Kleinunternehmen¹ einen folgenden Großauftrag vergibt oder ob er Ihr Kleinunternehmen dazu nicht in der Lage sieht. Den geforderten Quellcode erstellen Sie zügig live während der Abgabe und erklären währenddessen Ihre Vorgehensweise. Prof. Dopatka kann alternative Vorgehensweisen zur Lösung des Problems fordern.

[8 Punkte] Programmieren Sie eine Swing-GUI mit dem Titel „Live-Coding 3“, die aus einem Array von 3 JButtons sowie aus einem JLabel besteht. Die Buttons tragen die Aufschrift 1 bis 3 und sollen auch 1 bis 3 in das JLabel ausgeben, wenn man auf sie klickt.

Ob die Swing-GUI selbst ein JFrame ist oder nur einen JFrame kennt, entscheidet Prof. Dopatka zu Beginn der Abnahme durch Würfeln:

- 1 oder 2 oder 3: Swing-GUI ist ein JFrame
- 4 oder 5 oder 6: Swing-GUI kennt einen JFrame

Den zu verwendenden Layout-Manager entscheidet Prof. Dopatka zu Beginn der Abnahme durch Würfeln:

- 1 oder 2: FlowLayout
- 3 oder 4: GridLayout
- 5 oder 6: BorderLayout

Auch die Art und Weise des Eventhandlings entscheidet Prof. Dopatka zu Beginn der Abnahme durch Würfeln:

- 1 oder 2: Klassen-zentriertes Eventhandling in der GUI-Klasse
- 3 oder 4: separater Eventhandler in eigener Klasse
- 5 oder 6: dezentrales Eventhandling über anonyme innere Klasse

¹ Ihr Kleinunternehmen ist Ihre Praktikumsgruppe. Sobald Sie etwas abgeben, ist Prof. Dopatka Ihr Großkunde. Treten Sie also entsprechend freundlich, team-fähig, organisiert und sehr gut vorbereitet gegenüber dem Großkunden auf, denn die Existenz Ihres Kleinunternehmens ist von diesem Großkunden abhängig. Sie sind Dienstleister des Großkunden!



Aufgabe 3: Das Spiel

Alle Abgaben erfolgen ausschließlich über GIT. Wir behalten uns vor, die GIT-Aktivitäten jedes einzelnen Benutzer eines Teams zu überwachen und bei entsprechenden Hinweisen auf mangelnde Aktivität entsprechend intensiv nachzufragen.

a) [10 Punkte] Stellen Sie die gesamte Funktionalität des Spiels aus Übungsblatt 1, Aufgabe 3 sowie aus Übungsblatt 2, Aufgabe 3c, 3d und 4a über eine Swing-GUI zur Verfügung. Die GUI soll stets den aktuellen Status des Spiels anzeigen, den Würfel, alle Spieler und welcher Spieler gerade an der Reihe ist.

Es soll auch dargestellt werden, wenn ein Spieler eine KI ist. Es sollen auch nur KIs gegeneinander spielen können. Nachdem eine KI einen Zug beendet hat, so ist stets ein Button zu drücken, bevor es weitergeht.

Besprechen Sie rechtzeitig mit dem Kunden, wie er sich eine solche GUI vorstellt. Wie soll das Spielbrett dargestellt werden?

Alle Ausgaben auf der System-Konsole sind untersagt. Fehler sind statt dessen über JDialog als Popup auszugeben. Erscheinen Fehlermeldungen auf der Konsole oder bricht Ihre GUI sogar nach einer fehlerhaften Eingabe ab, so müssen Sie mit Punktabzug rechnen.

b) [2 Punkte] Sorgen Sie dafür, dass der aktuelle Gesamtzustand des Spiels in einer textbasierten Datei zu jeder Zeit abgespeichert werden kann. Dazu soll ein Button gedrückt werden. Danach wählt man mit einem JFileChooser den Pfad und die Datei aus, die dann abzuspeichern ist.

c) [3 Punkte] Beenden Sie nach Abgabe von 3b ihr Programm und starten Sie es dann neu. Sorgen Sie dafür, dass der aktuelle Gesamtzustand des Spiels aus einer textbasierten Datei zu jeder Zeit wieder eingelesen werden kann. Das Spiel wird dann an derselben Stelle fortgesetzt.