

Kontrolle M404, Theorie-Trainer

Teilauftrag 1

Beschreiben Sie textuell eine einfache Anwendung, die Daten in einem TUI anzeigt und bearbeiten lässt.

- Die Anwendung soll einen einfachen Sachverhalt mit mehreren Datensätzen (z.B. Adressliste, Notenliste, CD-Sammlung usw.) darstellen. (Erfüllt)

Ich habe eine einfache Idee genommen, die umzusetzen war. Die Idee hat einen besonderen Bezug auf die reale Welt. Die Prüfung wurde aufgrund Corona abgesagt. Daher hatte ich die Idee für diese App.

- Beschreiben Sie die Anwendung in wenigen Sätzen. Der Text soll auch für eine Nichtfachperson verständlich sein. Ein Merkmal gilt besonders den Daten, die durch die Anwendung repräsentiert werden. (übertroffen)

Den Text habe ich so einfach wie möglich formuliert. Daher sollte meine Idee für allen klar sein. Ich habe den Text meiner Familie gegeben, und gefragt, ob sie das Verstehen.

Teilauftrag 2

Entwerfen Sie das TUI zu Ihrer Anwendung.

- Entwerfen Sie die User-Dialoge für jede mögliche Anwendung der Applikation (Übertroffen)

Ich habe mir sehr viel Zeit genommen alles sauber, schön und übersichtlich zu gestalten. Ich habe auch auf die BZZ- Standards geachtet z.b Umlaute. Ich hatte ein klares Bild, wie das Programm aussehen sollte. Die Idee konnte gut in ein Word-Dokument umsetzen.

- Definieren Sie, wie das Programm auf falsche Eingaben reagiert. (Übertroffen)

Ich habe für die möglichen Fehler, die ich kenne, eine Meldung gedacht. Bei den einfacheren Fehlern habe ich eine Einheitliche Meldung genommen, damit sie nicht verwirren sollte. Daher sollte ich das mehr als genügend formuliert.

Teilauftrag 3

Erstellen Sie ein Zustandsdiagramm der Anwendung.

- Beschreiben Sie die Zustände der Anwendung aus Sicht des Benutzers. (erfüllt)

Ich konnte ein Zustandsdiagramm erstellen, aber ich hätte sie sicherlich noch besser darstellen können. Wenn ich Platz verwendet hätte, wäre das Diagramm übersichtlicher. Daher ist es meiner Meinung nach «okay»

- Geben Sie zu jedem Zustand an, was beim Eintritt (entry), während des Zustandes (do) und was beim Austritt aus dem Zustand (exit) zu tun ist. → Theorie zu Zustandsdiagramm (übertroffen)

Ich konnte überall alles einhalten. Aber bei einem Teil ist mit ein Copy/Paste Fehler passiert. Nach einer erstellten Kategorie konnte ich das Programm verlassen. Das gibt eigentlich keinen Sinn.

Teilauftrag 4

Erstellen Sie eine Liste der benötigten Klassen der Anwendungsdomäne, d.h. der Daten, die im TUI angezeigt und bearbeitet werden.

- Die Klassenbeschreibung muss vollständig sein, d.h. es sind alle Attribute und Methoden inklusive deren Sichtbarkeit festgehalten (übertroffen)

Ich konnte alle Beschreibungen ausführen und ich habe deren Sichtbarkeit festgehalten. Hier hatte ich keine Probleme und konnte diesen Teil schnell umsetzen.

- Falls sich Klassen gegenseitig kennen (z.B. Im Objekt Schulfach, gibt es ein Array von Noten) ist dies beschrieben. (übertroffen)

Bei den nötigen Klassen, die eine andere verwendet, habe ich im Beschreibungstext dies vermerkt. Das sollte mehr als genügend sein.

- Getter und Setter dürfen sie folgendermassen abkürzen: z.B. get/setNote() (übertroffen)

Ich habe alle Getter und Setter wie im Beispiel definiert, das sollte ebenfalls ausreichen.

Teilauftrag 5

Ergänzen Sie die Liste aus Teilauftrag 4 um alle Klassen, die Sie für die Applikation sonst noch brauchen.

- Die Klassenbeschreibung muss vollständig sein, d.h. es sind alle Attribute und Methoden inklusive deren Sichtbarkeit festgehalten. (übertroffen)

Ich konnte alle nötigen Klassen beschrieben und deren Sichtbarkeit notiert in einer Tabelle. Die Liste war für meiner Idee vollständig, zu diesem Zeitpunkt.

- Beschreiben Sie welche Klassen einander kennen und irgendwie benutzen. (übertroffen)

In einem Einleitungs-Text habe ich dieses Ziel vollständig umgesetzt.

Teilauftrag 6

Implementieren Sie Ihr Programm gemäss den Beschreibungen die Sie in Teilauftrag 1-5 erstellt haben.

- Erstellen Sie zuerst die Klassen der Anwendungsdomäne (Teilauftrag 4) und stellen Sie sicher, dass sich diese fehlerfrei übersetzen/compilieren lassen. (erfüllt)

Ich habe alle kompiliert, sodass mein Programm funktioniert. Die Funktion löschen habe ich rausgelassen, weil sie zu diesem Moment nicht funktionierte. Somit habe ich den Teil am Schluss auch ausgelassen, weil ansonsten das ganze Programm nicht mehr funktioniert hätte.

- Erstellen Sie das TUI, so dass in einem ersten Schritt alle Dialoge korrekt sind. Das Programm soll ausführbar sein, zeigt aber noch nicht unbedingt Daten an und weist nicht unbedingt Funktionalitäten auf. (übertroffen)

Ich konnte meine Vision 1:1 aufs TUI übertragen. Als ich das Programm ausführen konnte, sah es identisch zu dem Entwurf aus. Zudem habe ich den Code für einen Smiley herausgesucht, damit es besser aussieht.

- Ergänzen Sie nun die Funktionalität der Anwendung (Benutzeraktionen). (erfüllt)

Ich konnte alle Aktionen umsetzen, bis auf die Lösch-Funktion, für die ich nicht genügend Zeit hatte, aufgrund eines Fehlers. Die anderen Eingaben funktionieren solange man nicht die Fehlermeldungen einbezieht. Diese sind Folgefehler aus diesem einen Fehler.

Der Fehler lautet folgendes:

```
String test = " ";
```

```
test = einleser.readString();
if (test.equals("h")) {
    einleser.oneLine();
    welcome();
    home();

} else if (test.equals("z")) {
    einleser.oneLine();
    System.out.println("Das Programm wird beendet.\n" +
        "Achten Sie auf Ihr Fahrzeug, denn 90% der Unfälle sind Sie schuld\n" +
        "\uD83D\uDE0A\n");
} else {
    System.out.println("Das war eine falsche Eingabe. Bitte erneut eingeben:");
}
```

Dieser Teil war gedacht, um ins Hauptmenü zu kehren, oder das Programm zu verlassen. Weil man nur einen Buchstaben eingeben musste, versuchte ich die Logik mit char zu lösen. Aber immer wieder kam der Fehler «out.of.bounce» und das Ganze Programm funktionierte nicht mehr. Am Mittwoch-Morgen versuchte ich die Logik mit einem String, und es funktionierte dann. Danach habe ich mehr oder wenig mein ganzes Main neu gemacht, damit es übersichtlich sein sollte. Ich habe einige neue Methoden eingefügt, damit ich nicht zu viel Text in einer Methode hatte. Dadurch wurde welcome(), und home() neu entworfen. Nach der Abgabe fiel mir auf, dass die Logik, um die möglichen Fehlermeldungen anzuzeigen weg waren. Ich hatte die Logiken aus dem Main Teil gelöscht, was mich danach frustrierte. Das Programm funktionierte, aber die Fehlermeldungen, die ich definiert hatte, waren weg. Im letzten Teil 7 habe ich meine neuen Methoden noch in Kapitel 5 eingefügt.

- Fügen Sie in jeder Klasse sowohl Klassenköpfe wie auch Methodenköpfe ein (Dokumentation) **(übertrifft)**

Ich habe bei allen ausser einem alle Methoden eingefügt. Ich habe die Version, an denen ich etwas geändert habe immer mitgezählt. Bei einem fehlt der Kopf, ebenfalls aus einem Folgefehler aus den String-Problem.

- Kommentieren Sie im Code Bereiche, in denen Sie einen Kommentar als nützlich erachten. **(erfüllt)**
Ich habe keine Kommentare verwendet, weil ich diese nicht nötig hatte. Daher weiss ich nicht wie ich diesen Teil bewerten kann. Meiner Meinung nach habe ich dieses Ziel erfüllt.
- Halten Sie sich an die BZZ-Codingstandards. **(übertrifft)**

Ich habe mich an die Sprache gehalten (keine langen Ketten, Umlaute umschrieben). Somit hatte ich alle Punkte drangehalten bis auf eines, weil der nötige Teil gesamt gelöscht wurde, aufgrund des String Fehlers.

Teilauftrag 7

Schreiben Sie mind. 8 Testfälle für Ihr Programm. Führen Sie diese Tests aus und erstellen Sie ein Testprotokoll.

- Erstellen Sie zuerst die Testfälle. Überlegen Sie, was sinnvoll ist zu testen. (übertrouffen)

Ich habe zuerst die Fälle genommen, die auch funktionierten. Danach habe ich die unisicheren Fälle genommen, und zuletzt diejenigen, die 100 ein Fehler war. Danach habe ich sie nach der Programmierlogik grob sortiert. Ich habe 10 Fälle gedacht, was ausreichen sollte.

- Testen Sie anschliessend Ihr Programm anhand der Testfälle. Füllen Sie dazu ein Testprotokoll aus. (übertrouffen)

Ich konnte sehr gut ein Protokoll machen, nach den Fällen wusste ich bereits, was ich möglicherweise Falsch gemacht habe. Daher konnte ich diesen Teil schnell und effizient ausführen. Am Schluss gab es einen Nebenfehler (Darstellung könnte besser sein) und einige Hauptfehler, die ebenfalls aus dem String Fehler resultieren.