

Programmierpraktikum 2016

iuvba

Bewertungsbogen

17.07.2016

Grundlegende Anforderungen

- | | |
|---------------------------------------------------------------------|-------------------------------------------------------------------|
| <input type="checkbox"/> Anwendung lässt sich kompilieren | <input type="checkbox"/> <code>Exceptions</code> werden behandelt |
| <input checked="" type="checkbox"/> Bericht vorhanden | <input checked="" type="checkbox"/> Nutzerhandbuch vorhanden |
| <input checked="" type="checkbox"/> Bericht verständlich formuliert | <input type="checkbox"/> Nutzerhandbuch verständlich formuliert |

Kommentar

Nachdem ein Katalog gewählt wurde, erscheint beim Klicken auf "Weiter" ein neues Fenster in dem ein Ordner ausgewählt werden soll. Im Nutzerhandbuch wird nicht beschrieben wofür der ausgewählte Ordner benutzt wird. Wird das Fenster zum Auswählen der Datei geschlossen, ohne eine Datei auszuwählen, wirft die Anwendung eine `NullPointerException`. In `gradle.build` wird die Option `sourceCompatibility = 1.5` gesetzt wodurch sich die Anwendung mittels `gradle build` nicht kompilieren lässt (da der Diamond Operator in Java 5 nicht unterstützt wird).

Projektbezogene Anforderungen

- | | |
|-------------------------------------------------------------------------------|------------------------------------------------|
| <input checked="" type="checkbox"/> Gradle-Skript fürs Bauen der Anwendung | <input type="checkbox"/> Travis CI eingebunden |
| <input type="checkbox"/> <code>README.md</code> mit Gradle Task und Anleitung | <input type="checkbox"/> Lizenz vorhanden |

Kommentar

Die Aufgabenstellung verlangt nach einer Readme-Datei, in der beschrieben wird welcher Task für das Bauen der Anwendung zuständig ist und wie die Anwendung ausgeführt wird. Die im Repository eingetragene Datei enthält diese Informationen nicht. Ebenso fehlt eine Lizenz und die Einbindung von Travis CI.

Anwendungsbezogene Anforderungen

- | | |
|---------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| <input checked="" type="checkbox"/> Anwendungsszenario wurde eingehalten | <input checked="" type="checkbox"/> Katalog kann mit Texteditor editiert werden |
| <input checked="" type="checkbox"/> Katalog von möglichen Übungen ist vorhanden | <input checked="" type="checkbox"/> Katalog ist nicht fest eingebaut |
| <input type="checkbox"/> Übungen beinhalten Vorlagen für Klassen und Tests | <input type="checkbox"/> Übungen beinhalten Konfiguration |
| <input checked="" type="checkbox"/> Prozess wird visuell dargestellt | <input type="checkbox"/> Mindestens zwei Erweiterungen implementiert |
| <input type="checkbox"/> <code>virtual-kata-lib</code> wird benutzt | <input type="checkbox"/> Ausreichende Menge an Unit-Tests |

Die Anwendung bietet zur Zeit nur das Einlesen von Übungen und den Wechsel zwischen Phasen (ohne Überprüfungen). Der implementierte Timer startet bei jedem Phasenwechsel einen neuen Thread und zählt die Uhr somit immer schneller runter. Beim Ablaufen der Zeit fängt der Timer an in den negativen Bereich zu zählen. Die Übungen innerhalb des Katalogs beinhalten keine Vorlagen für Klassen und zugehörige Tests. Ebenso fehlen Einträge für die Konfigurationen einzelner Übungen. Von den vorgegebenen Erweiterungen der Anwendung wurde keine implementiert. Unit-Tests, die die Programmlogik überprüfen, sind ebenfalls nicht vorhanden. Die zur Verfügung gestellte Bibliothek wird zwar im Projekt eingebunden, jedoch an keiner Stelle benutzt.

Feedback/Bewertung

Da die Anwendung im aktuellen Zustand nicht den Anforderungen der Aufgabenstellung genügt muss ich sie vorerst mit einem Mangelhaft bewerten. Der Aufbau und die Gestaltung der graphischen Oberfläche sind in dieser Form völlig ausreichend und gefallen mir persönlich auch. Leider scheint das Nutzerhandbuch auf eine ältere Version der Anwendung ausgerichtet zu sein und es ist somit teilweise nicht klar was gemeint ist.

Die Struktur des Codes ist etwas unordentlich und bietet eine sehr beschränkte Übersicht über die Funktionen der Anwendung. An dieser Stelle wäre es sinnvoll einzelne Methoden, die den entsprechenden Effekt erzielen, zu schreiben und nicht alles in eine Methode zu quetschen.

Für das Bestehen des Projektes müssen mindestens folgende Punkte bearbeitet werden:

- Es dürfen während der Ausführung der Anwendung keine unbehandelten Exceptions auftreten.
- Das Nutzerhandbuch muss an die Anwendung angepasst sein und einem unerfahrenen Benutzer die Bedienung der einzelnen Komponenten verständlich erklären.
- In der `README.md` muss beschrieben werden welcher Gradle-Task für das Bauen der Anwendung zuständig ist und wie diese gestartet wird.
- Es muss ein Katalog von möglichen (sinnvollen) Übungen erstellt werden. In diesem müssen die einzelnen Übungen Vorlagen für Klassen und zugehörigen Tests beinhalten.
- Es müssen mindestens zwei der vorgegebenen Erweiterungen implementiert werden.
- Wichtige Stellen in der Programmlogik müssen durch Tests abgedeckt werden.
- Die zur Verfügung gestellte Bibliothek muss für die zu implementierende Programmlogik benutzt werden.

Im Gesamtergebnis muss die Anwendung den in der Aufgabenstellung beschriebenen Ablauf einhalten und alle hierfür nötigen Schritte in Form von Code beinhalten.

Da bisher nicht viel Programmlogik geschrieben wurde könnt ihr die Aufgaben selbstständig untereinander aufteilen. Jedes beteiligte Teammitglied soll am Ende in etwa gleich viel **Code (Programmlogik)** zum Projekt beigetragen haben. CSS, JSON, FXML und andere Textdateien sind also nebensächlich.

Ob ihr individuell bestanden habt, erfahrt ihr im Aus.

Die letzte Abgabefrist ist der **25.07.2016 23:59 Uhr** (andernfalls schaffe ich die Korrektur evtl. nicht).