Benutzerhandbuch

1. Baue das Programm mit gradlew build und starte das Programm mit gradlew run

2. Das Hauptmenü öffnet sich, nun kann man wählen welche Aufgabe man bearbeiten möchte

- Entweder aus dem bestehenden Aufgabenkatalog der sich in der Liste befindet
- Oder man kann sich unten dem Button *Aufgabe hinzufügen* externe .txt Dateien laden

3. Es gibt 2 zusätzliche Features Babysteps und ATDD

- In Babysteps kann die Zeit innerhalb der Phasen begrenzt werden. Wenn es angeklickt wird öffnet sich ein Slider indem man das Zeitlimit auswählen kann (hier: zwischen 2 und 5 Minuten)
- ATDD ergänzt das Programm um einen zusätzlich auszuführenden Akzeptanztest

4. Ohne ein Feature gewählt zu haben:

- Nachdem man auf *Start* geklickt hat öffnet sich ein neues Fenster
- Auf der linken Seite findet man die zuvor ausgewählte Aufgabe
- Mit dem Button *Zurück zum Aufgabenkatalog* kann man jederzeit zurück zum Menü
- Unter der Aufgabe sieht man die Kompilierfehler
- In der Mitte kann man den Code schreiben / bearbeiten
- Es gibt zwei Tabs in dem ersten wird der Testcode geschrieben im zweiten der Sourcecode
- Mit dem Button *Kompilieren* kann man seinen Code kompilieren und wenn bestanden in die nächste Phase gelangen
- Wenn man sowohl den Sourcecode als auch den Testcode bestanden hat kann man mit dem Button Refactor seinen Code verbessern oder wenn nicht gewünscht mit nochmaligen Klick wieder in den Zyklus gelangen
- Mit dem Button *Zurück zum Testcode* kann man von der Sourcecode Phase (GREEN) wieder in die Testphase (RED) zurück gelangen
- In welcher Phase man ist, wird durch die oben dargestellten Pfeile verdeutlicht
- Und was zu tun ist steht im Status

Phasen von TDD

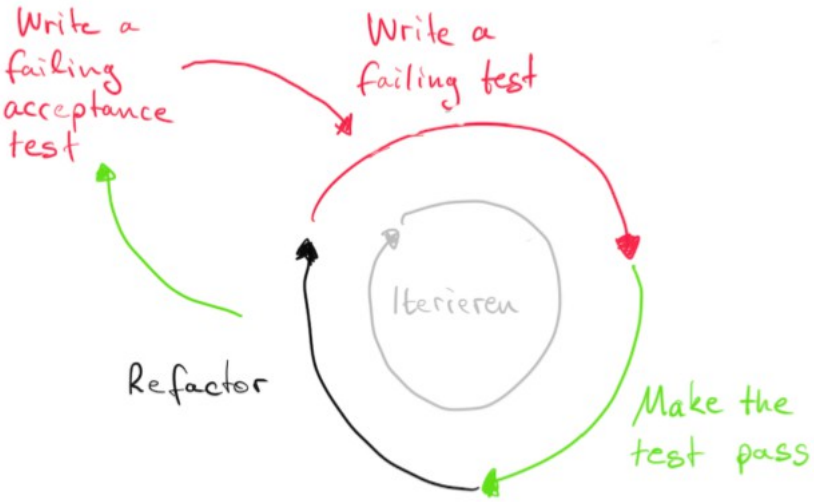
5. TDD mit Babysteps

- Oben rechts wird die zuvor ausgewählte Zeit angezeigt, die direkt abläuft
- Die Zeit, die der Nutzer in den Phasen RED (TestCode) und GREEN (SourceCode) hat ist limitiert, in der Phase REFACTOR nicht. Ist die Zeit abgelaufen, wird der neue Test/Code gelöscht und es wird in die vorangegangene Phase zurück gewechselt
- Der restliche Ablauf bleibt bestehen

6. TDD mit Akzeptanztest

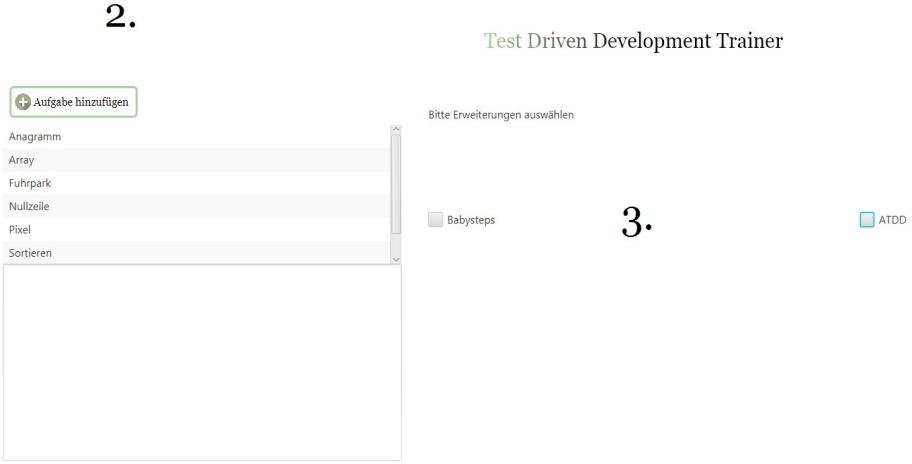
- Ein Akzeptanztest ist ein übergeordneter Test, der grün wird, wenn ein Feature vollständig implementiert wurde, ansonsten bleibt er rot
- Der Test wird vor dem normalen TDD Zyklus geschrieben
- Dann startet der normale TDD Zyklus. Das Feature wird mit Unit-Tests schrittweise entwickelt, bis alle Tests (also auch der Akzeptanztest) bestehen, danach kann ein neuer Akzeptanztest geschrieben werden

7. Man kann sowohl ADDT als auch Babysteps gleichzeitig ausführen



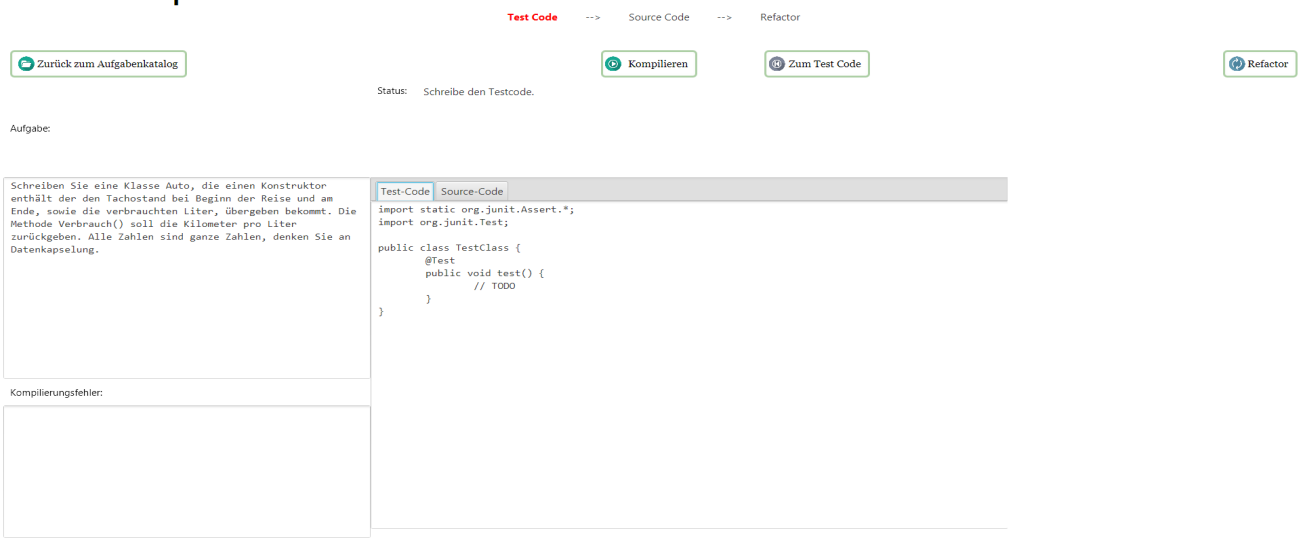
Phasen mit Akzeptanztest

2.



Sceenshot zum Punkt 2 und 3

4.



Screenshot zum Punkt 4

Bildquelle: Programmierpraktikum Projekt 7