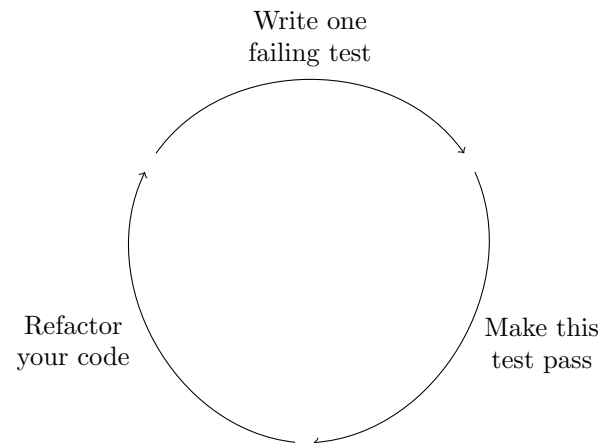


# TDD by Silvan Habenicht, Leander Nachtmann, Tobias Quest



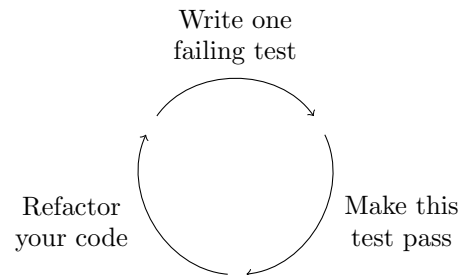
## Ziel der Anwendung

Eine Entwicklungsmethode, um gut wartbaren Code zu schreiben, ist die testgetriebene Entwicklung (*englisch: test driven development*, kurz TDD). Das Programm TDD sollen Ihnen diese Entwicklungsmethode näher bringen, indem es sie zur Einhaltung der zugrunde liegenden Regeln zwingt.

## Der TDD-Zyklus

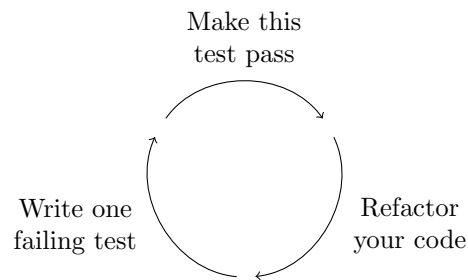
Der TDD-Zyklus besteht aus drei Phasen:

- **Test-Phase**



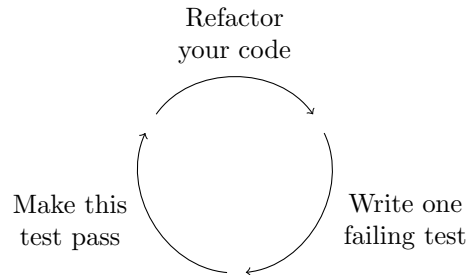
In der Test-Phase müssen sie einen Test schreiben, der fehlschlägt, weil die entsprechende Anforderung im Code noch nicht erfüllt ist.

- **Code-Phase**



In der Code-Phase müssen Sie möglichst einfachen Code zur Erfüllung des Tests ergänzen. Dabei ist explizit nicht das Ziel, erwartbare weitere Tests zu erfüllen.

- **Refactor-Phase**



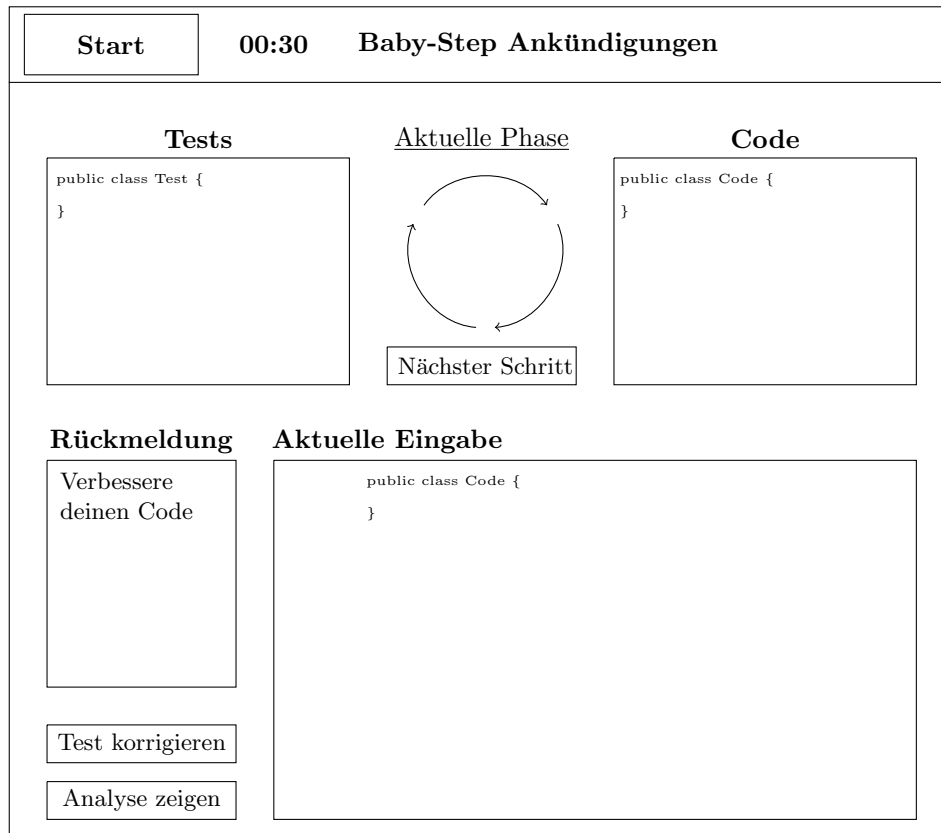
In der Refactor-Phase haben Sie die Möglichkeit ihren Code zu überarbeiten. Das kann zum Beispiel die Zusammenfassung mehrerer Anweisungen zu einer Schleife sein.

## Starten der Anwendung

TDD kann aus jeder Entwicklungsumgebung oder mittels java Main über die Konsole gestartet werden. Voraussetzung ist eine aktuelle java Umgebung (jdk-8 mit Alert und javafx Unterstützung).

Nach dem Starten der Anwendung werden Sie aufgefordert einen Katalog und dann aus diesem eine Aufgabe auszuwählen. Von dort gelangen Sie zur Programmieroberfläche.

## Erklärung der Oberfläche



Oben sehen Sie die Benutzeroberfläche. Mit dem Button **Start** aktivieren Sie den TDD-Zyklus. In den Feldern **Test** und **Code** können Sie die von Ihnen eingegebenen Tests beziehungsweise den von Ihnen eingegebenen Code sehen. Die aktuelle Phase steht in der rotierenden Grafik jeweils oben.

Mit dem Button **Nächster Schritt** gelangen Sie zur nächsten Entwicklungsphase. Über das Feld Rückmeldung erhalten Sie Informationen über Compilerfehler und fehlgeschlagene Tests.

Mit dem Button **Text korrigieren** haben Sie die Möglichkeit zur Test-Phase zurückzukehren, um Test zu korrigieren. Diese Korrekturmöglichkeit sollte aber nur dann verwendet werden, wenn ein Test falsch oder unerfüllbar ist.

Der Button Analyse zeigen liefert Ihnen eine Analyse des Programmierverlaufs. In das **Aktuelle Eingabe** Feld geben Sie Ihre Tests und Ihren Code ein und überarbeiten diesen.

## **Zusatzfunktionen**

### **BabySteps**

Ist im Auswahlkatalog die Funktion BabySteps aktiviert haben Sie für die Bearbeitung der Phasen Test und Code nur begrenzte Zeit. Diese wird Ihnen oben links angezeigt. In der Refactoring-Phase haben Sie weiterhin beliebig viel Zeit. Ziel dieser Funktion ist die möglichst kleinschrittige Entwicklung des Codes.

### **Tracking**

Wenn im Auswahlkatalog die Funktion Tracking aktiviert ist, wird eine Datei angelegt, die jeden Schritt Ihres Programmiervorgangs dokumentiert. Der Name der Textdatei lautet **Tracked.txt**. In dieser Datei können Sie jeden einzelnen Schritt inklusive Dauer und Fehlermeldungen nachvollziehen und Sie bekommen die Zeiten in den einzelnen Entwicklungsphasen angezeigt.

### **Lizenz**

Diese Software steht unter MIT-Lizenz und verwendet fremde Software. Genauereres über die Möglichkeiten der Verwendung und Modifikation dieser Software kann im mitgelieferten Licence-File eingesehen werden.

**Viel Spaß mit dem TDD wünschen Ihnen Silvan Habenicht,  
Leander Nachtmann und Tobias Quest**