

# Nutzerhandbuch - TDDT

## Warum dieses Tool?

Im Rahmen Ihrer Vorlesung sollen Sie lernen Softwareentwicklung nach der Methode des TDD (test driven development) anzuwenden. Dieses Tool unterstützt Sie bei der Anwendung der Regeln. Dieses Dokument gibt Ihnen einen Überblick über die Verwendung dafür.

## TDD

Das Prinzip des TDD (test driven development) schreibt vor, dass als erstes ein Test geschrieben und anschließend die Methode zur Lösung des Test implementiert wird. Danach folgen ein weiterer Test & eine weitere Methode. Und so weiter.

## Wie funktioniert das Tool?

1. Wählen Sie „Exercise“
2. „Select Exercise“
  - Wählen Sie eine Übungsaufgabe zur Bearbeitung aus.
  - Klicken Sie im neuen Fenster auf die gewünschte Übung.
  - Dann „select“.
  - Optional können Sie hier die Babysteps aktivieren (siehe unten Babysteps)
3. Sie beginnen in Phase **RED**
  - Schreiben Sie ihre erste(n) Test(s).
  - Klicken Sie auf „next Step“
  - Das Tool prüft ob du Fehler gemacht wurden. Sie kommen nur weiter, wenn es genau einen Fehler im Code gibt.
  - Das Tool zeigt Ihnen den Fehlercode dazu an.
4. Sie gelangen in Phase **GREEN**
  - Schreiben Sie die Methode(n) zur Lösung des/r Tests.
  - Klicken Sie auf „next Step“
  - Analog zum Verhalten in Phase **RED** prüft das Tool ob Sie Fehler gemacht haben und zeigt Ihnen den Fehlercode an.
  - Sie gelangen nur in die nächste Phase, wenn alle Tests bestehen.
5. (alternativ) Klicken Sie auf „back to **RED**“
  - Sie können nun weitere Tests schreiben und analog zu 2. & 3. Fortfahren
6. (alternativ) Klicken Sie auf „next Step“
  - Sie befinden sich nun in der **REFACTOR** Phase
  - Sie können sowohl die Tests als auch den Code editieren.
  - Beachten Sie, dass alle Tests nach dem editieren weiterhin funktionieren müssen, um diese Phase erfolgreich abzuschließen.
7. Sie befinden sich jetzt wieder in Phase **RED**
  - Vorgehen analog zu 2. und folgenden

## Phasen

Es gibt 3 Phasen.

### RED

Die Startphase. Sie sollen hier einen oder mehrere Test(s) schreiben.

### GREEN

Die zweite Phase. Sie haben bereits einen oder mehrere Test(s) verfasst und stehen nun vor der Aufgabe den Code zur Lösung zu editieren.

### REFACTOR

Die dritte Phase. Sie haben es geschafft sowohl mindestens einen Test zu schreiben und diese(n) fehlerfrei zu lösen. Zur Belohnung haben Sie hier die Möglichkeit gleichzeitig Test & Code zu editieren. Sinn dieser Phase ist die Vereinfachung des Codes & der zugehörigen Tests. Beachten Sie, dass nach der Bearbeitung immer noch alle Tests funktionieren müssen.

## Übersicht Funktionen

Sprache:

-> „Settings“ -> „Language“

Ändert deine Spracheinstellung. Zur Auswahl stehen Deutsch & Englisch.

Nutzerhandbuch:

-> „Help“ -> „Handbook“

Zeigt dir dieses Dokument an.

Tutorialmode:

-> „Help“ -> „Tutorialmode“

Aktiviert den Tutorialmode. Sie erhalten nun Handlungshinweise, zum Erlernen des Umgangs mit dem Tool.

Select Exercise:

-> „Exercise“ -> „Tutorialmode“

Ihr Startpunkt für jede neue Übung/Aufgabe. Hier wählen Sie, aus dem Popup Fenster, die nächste Übung aus und klicken auf „select“.

Description:

-> „Exercise“ -> „Description“

Zeigt Ihnen die hinterlegte Beschreibung zur Aufgabe an.

Back:

-> „Exercise“ -> „Back“

Sie gelangen durch zurücksetzen des letzten Fortschrittes eine Phase zurück.

ACHTUNG. Möglicherweise gehen Änderungen verloren.

Erweiterungen:

Tracking

- -> „Exercise“ -> „Tracking“
- Die Tracking Funktion zeichnet auf, was Sie wann geändert haben. In übersichtlicher Form wird Ihnen hier eine Analyse präsentiert.

Babysteps

- -> „Exercise“ -> „Babysteps“
- Kann eingeschaltet werden. (optional)
- Bei Aktivierung gibt diese Erweiterung Ihnen ein Zeitlimit für das Lösen der Aufgabe vor. Sollten Sie nicht innerhalb der eingestellten Zeit in die nächste Phase wechseln wird der Code/Test zurückgesetzt. Sie können von vorne beginnen.
- Sinn ist es, dem Nutzer die Entwicklung in kleineren Schritten nahezulegen/anzutrainieren.