TDDT Manual

xxxMathxxx

Christopher Geyer    Philipp Spohr    Lukas Wagner    Fabian Weiß

July 13, 2016

# Contents

# 1 Introduction

The following pages contain a detailed description on how to use the TDD-Trainer. Note that there are mainly 2 sections, one for teachers/tutors and another one for students.

## 2 Getting Started

If you haven't compiled the software yet or need an executable build, refer to our readme. This manual is assuming that TDDT is running on your platform.

## 3 For students

The following section is for students. It describes how to use the program properly and explains how the different features are used.

### 3.1 Startup-Notification

The first thing you will encounter when starting our application is an educational info panel, displaying important information about TDD to you. Even if you already are an experienced developer there might be some new facts for you.

### 3.2 Startup-Menu

After you closed down the startup-notification you will get into the Startup-Menu. If this is your first usage of the TDDT you might want to create a new profile. If you already have a profile or you just created one, you can go ahead and begin your training.

### 3.3 New Profile

If you choose to create a new profile, you just got to enter your name and upload a picture. As your profile picture is only allowed to be of a certain size, you might use the provided tool to get your image to the required measurement.

### 3.4 Profile-Menu

After you have chosen your profile, you end up in the profile menu, where you can either go directly to the exercises, or you check out your statistics. If this is your first start of the program, you might as well just start an exercise, as there are no statistics to show.

### 3.5 Exercise-Menu

When you choose to start an exercise you end up in this menu, here you can pick one of the various exercises and see which medals you already earned on each one. The rest is self-explanatory. Pick an exercise and start!

### 3.6 Achievements/Medals

In order to keep the motivation high no matter how dull an exercise seems you have the opportunity to collect medals on each exercise. The software tracks the total time you need to clear the exercise. If you perform especially well you will earn a medal. Your medal progress can be tracked on your profile page. There are three regular medals: bronze, silver and gold. In addition there is

the author medal which will only be awarded to the fastest TD-developers out there. Earn your place amongst the programming elite!

## 3.7 The Exercise

This is the main part of the program. Your goal now is it, to write a program using the technique of test-driven-development. At first you edit the tests until you have exactly one failing test or the code does not compile. If this is the case, you might switch to the next step. Now you may only edit the code, not the tests anymore. You might also swap back to editing the tests, but this deletes all the code you have written up until this point. If you get all tests to work and the program compiles without an error, you may switch to the last stage, the refactoring. Now you can improve your code until you decide you are done and go back to writing more tests. You repeat this cycle until your program is finished. Also, as soon as you start your exercise, a timer starts ticking, so you got to hurry up if you want to get a highscore!

## 3.8 Final Tests

As soon as you think your program has everything it needs, you may click on finalize. Your code runs through the final tests now. If you manage to pass every single one, you are done with this exercise!

# 4 For teachers/tutors

The following section is for teachers and tutors only, as it describes how to write more exercises for the program, as well as how the implement them.

## 4.1 How to add your own exercises to the software

To add your own exercise to the program, you have to create a proper xml-file and simply add it to the exercise folder. It will then be added into the list of exercises automatically.

### 4.1.1 How to create a proper xml-file

First you've got to define your XML-file as exercise by setting surrounding tags like this: `<exercise></exercise>`. Everything you write will have to be inside this tags.

### 4.1.2 Name

You will have to create a name for your exercise. Define this by adding `<name></name>`. Don't forget to add a name inside the tag!

### 4.1.3 ID

You will also have to create an id for your exercise. Define this by adding `<id></id>`. Never change the id after the exercise was used. Also don't create two exercises with the same id.

### 4.1.4 Description

To add a description define it with the tags `<description></description>`. This description will be displayed when the user selects an assignment.

### 4.1.5 Classes

Now you will have to create classes in which the user can write his code. First create the classes-segment by adding the tags `<classes></classes>`.
Then you can define as many classes as you want by adding them by this scheme: `<class name="Name">Code</class>` Now you will have to replace Name with the desired classname (Without ".class" in the end) and Code with the desired starting code.

### 4.1.6 Tests

You can add tests by the same scheme as classes. Just replace `<classes></classes>` by `<tests></tests>` and for each test `<class name="Name">Code</class>` by `<test name="Name">Code</test>`.

### 4.1.7 Finishtests

You also have to define a finishtest. This test is used to check if an assignment has been fully completed. Simply add a new segment `<finishtests></finishtests>` like you did for the classes. Then you can add your test like you did with the normal ones: `<finishtest name="Name">Code</finishtest>`. Also don't forget to replace Name and Code like described above.

### 4.1.8 Config

For the last step you will have to define the configuration. Do this by adding a new tag `<config></config>`.
Now you can define the parameters for babysteps, timetracking and medals:

`<babysteps value="True" time="3.22" />` This will enable babysteps and set their timer to 3.22 minutes.

`<timetracking value="True" />` This will enable timetracking.

`<medals bronze="13.37" silver="6.65" gold="1.00" author="0.5"/>` This will define the time (in minutes) needed to get a medal awarded.

### 4.1.9 It's not working!

If you've got more problems with this, check a premade exercise file in the exercises directory.

## 4.2 Extensions

Two extensions are implemented into the program: Babysteps and tracking.

### 4.2.1 Babysteps

This extension limits the time a user has to complete one task in the first or second step of the program. If they don't make it in time, the whole code which was generated up until this point gets deleted and he has to start all over again. The Babysteps timer is disable for the refactor stage.

### 4.2.2 Tracking

The second extension is all about creating statistics for the user to track where he makes the most mistakes and which tasks cost him the most time. The tracked and refined data for a profile can be found in the statistics menu.