

TDDT

Benutzerhandbuch

Inhalt:

1. Übersicht über das Programm	Seite 3
2. Grundlegende Funktionen	Seite 4
2.1. Das Menü	
2.2. Laden einer vorhandenen Aufgabe	
2.3. Erstellen einer Aufgabe	
2.4. Bearbeiten einer Aufgabe	
3. Arbeiten mit dem Programm	Seite 9
4. Zusätzliche Funktionen	Seite 10
4.1. Babysteps	
4.2. Tracking	



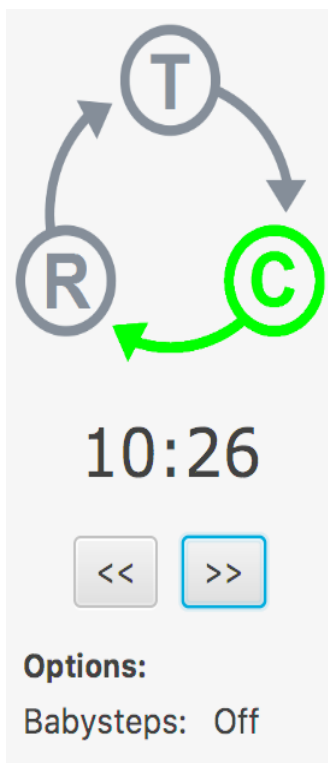
1. Übersicht über das Programm

Das Programm hat folgenden Aufbau:

The screenshot shows the 'Test Driven Development Trainer by GG Ez' application. It features a sidebar on the left with a circular diagram of modes (T, R, C) and a clock showing 10:19. The main area is divided into three sections: 'Java Klassen' (containing code for CalculatorMain and Calculator), 'Code' (containing the code for CalculatorTest), and 'Test' (containing the code for CalculatorTest). A 'Compiler Meldungen' section at the bottom displays an error message: 'CalculatorTest.java:7:error:cannot find symbol symbol: method assertEquals(int,int) location: class CalculatorTest assertEquals(1,2);'. The interface also includes a 'Description' section with instructions and a status bar at the bottom indicating a successful switch from 'Test schreiben' to 'Code schreiben'.

Annotations:

- Aufgabenstellung**: Points to the sidebar area containing the mode diagram and clock.
- Code**: Points to the 'CalculatorMain' and 'Calculator' class definitions.
- Test**: Points to the 'CalculatorTest' class definition.
- Compiler Meldungen**: Points to the error message in the bottom section.



Zeigt den aktuellen Programm Modus an

Aktuelle Uhrzeit bzw. verbleibende Zeit im Babysteps Modus

Nächster/Vorheriger Schritt

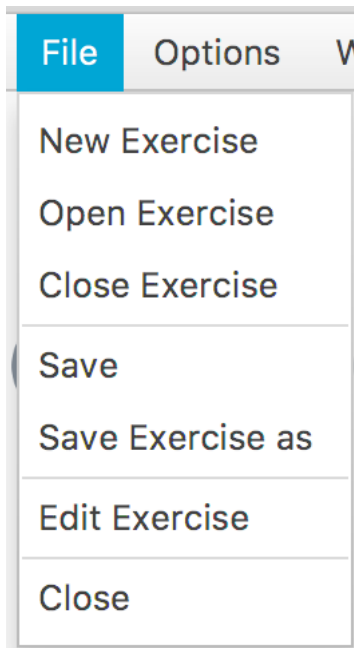
Status der Babysteps (s. Seite 10)



2. Grundlegende Funktionen

2.1. Das Menü

File



New Exercise

Erstellt eine neue Aufgabe

Open Exercise

Lädt eine vorhandene Aufgabe in das Programm.

Close Exercise

Schließt die aktuelle Aufgabe.

Save /
Save Exercise as

Speichert die aktuelle Aufgabe (nur verfügbar wenn eine Aufgabe geöffnet ist).

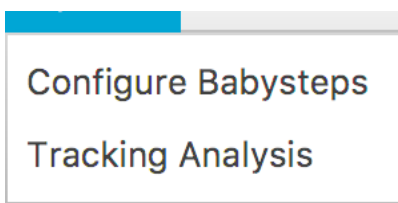
Edit Exercise

Ermöglicht das bearbeiten der aktuellen Aufgabe.

Close

Schließt das Programm.

Options



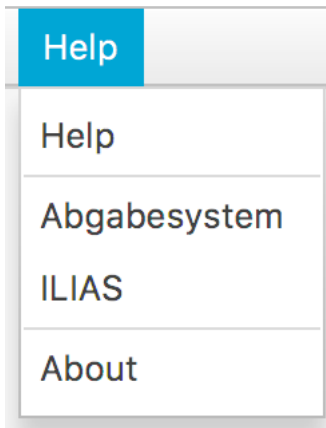
Configure Babysteps

Öffnet ein Menü zum konfigurieren der Babysteps Funktion (siehe S. 10).

Tracking Analysis

Tracking Funktion des Programms (siehe S. 10).

Help



Help

Öffnet das Benutzerhandbuch.

Abgabesystem

Öffnet das Abgabesystem für das Abgeben der bearbeiteten Aufgaben.

ILIAS

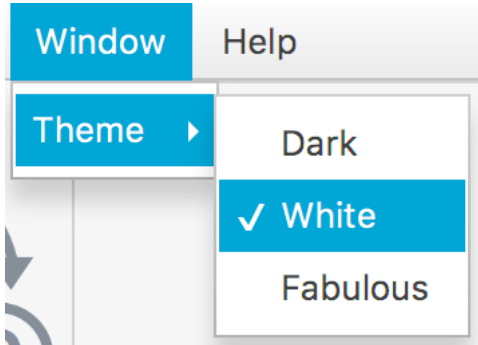
Zeigt das Ilias Forum.

About

Zeigt Informationen zum Programm.

Hinweis: Unter Linux können die Befehle Help, Abgabesystem und ILIAS nicht ausgeführt werden.

Window

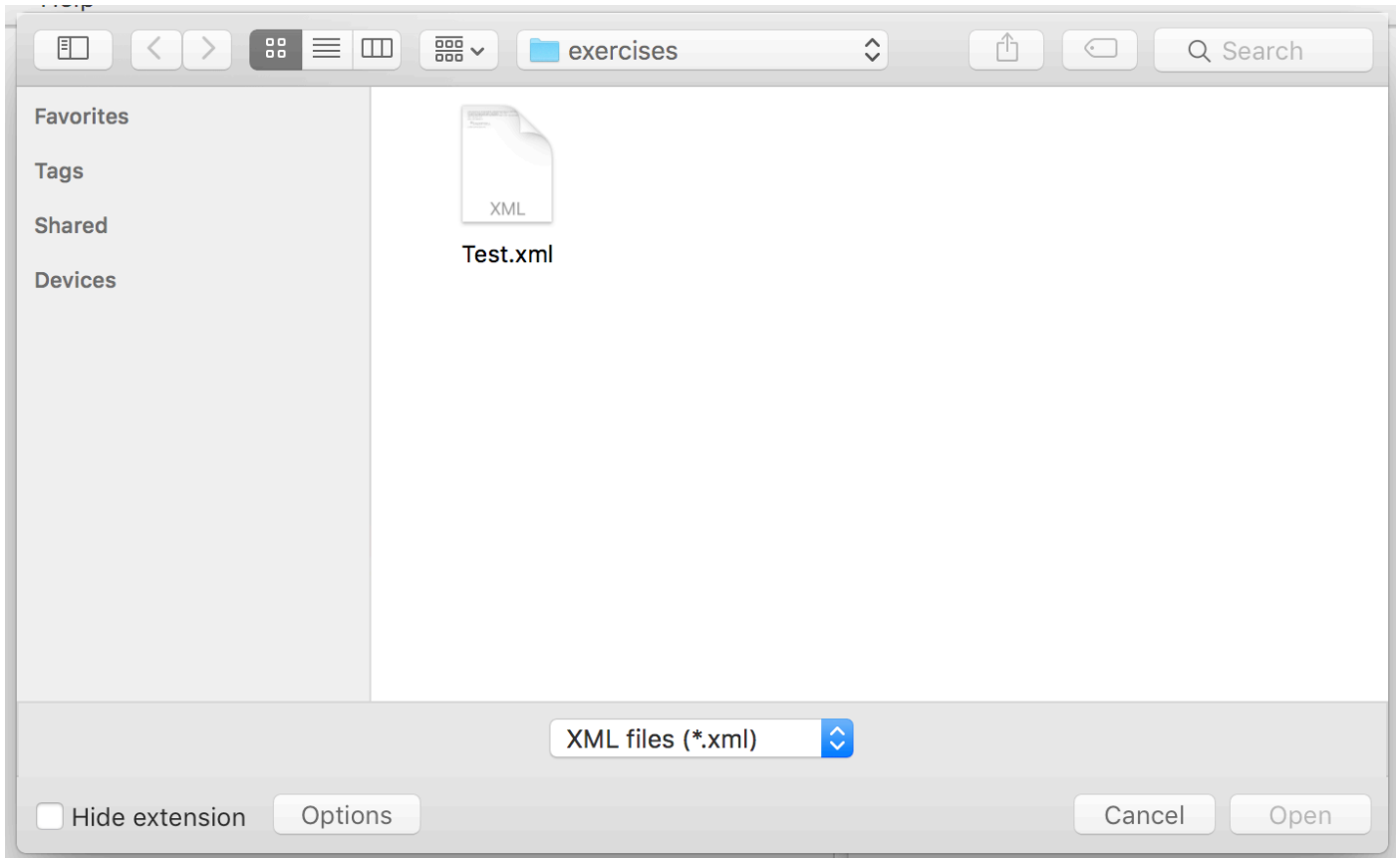


Hier kann das Desing des Programms verändert werden.

3.2 Laden einer vorhandenen Aufgabe

Wird „Open Exercise“ aus dem Menü ausgewählt, öffnet sich ein Fenster in welchem die zu öffnende Aufgabe ausgewählt werden muss.

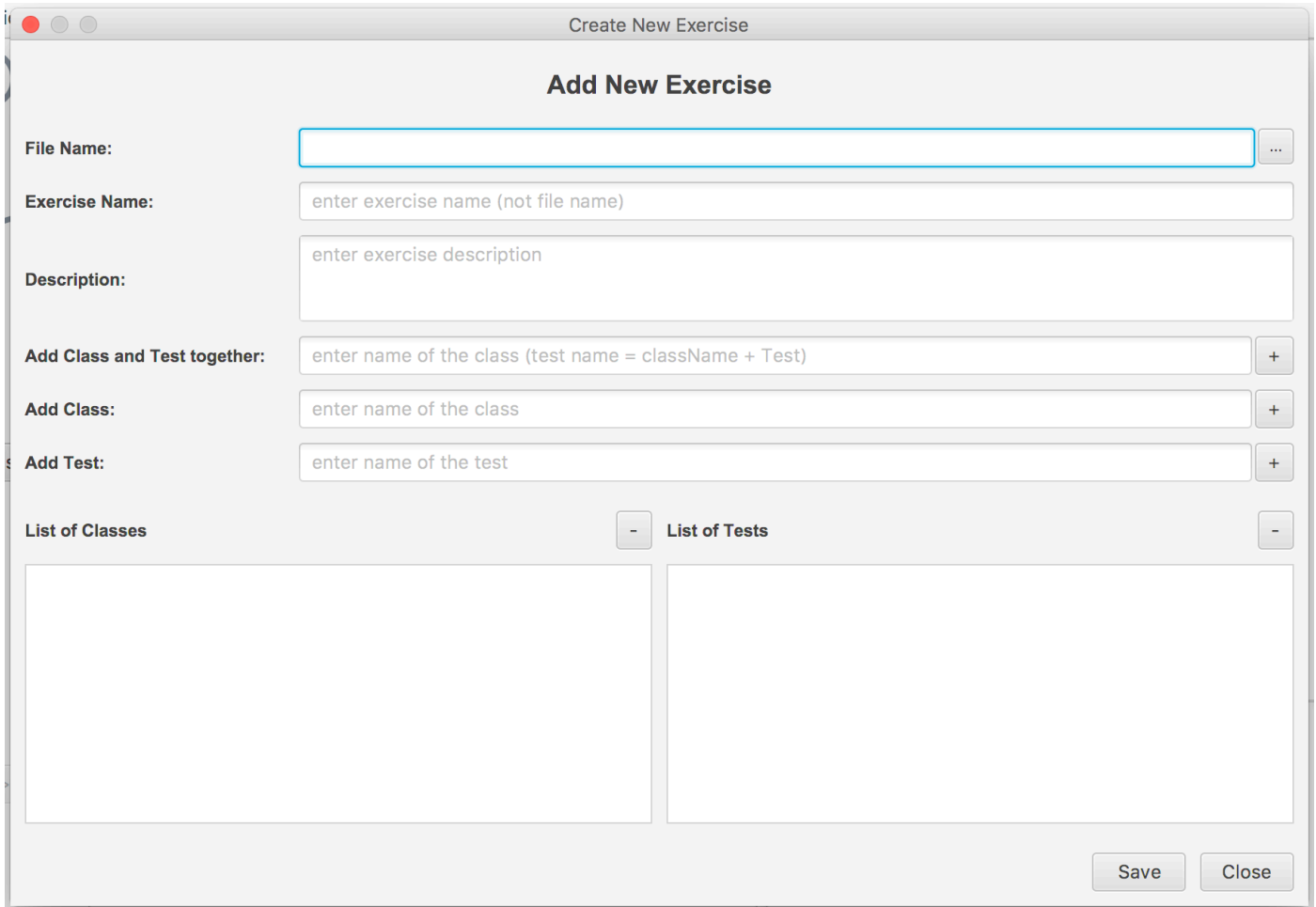
Die Aufgabe muss im xml-Format vorliegen.



Durch die Schaltfläche „Open“ wird die Aufgabe in das Programm geladen und kann dort nun bearbeitet werden.

3.3 Erstellen einer Aufgabe

Soll keine neue Aufgabe zum bearbeiten geöffnet werden, lässt sich im Programm auch eine neue Aufgabe erstellen. Dazu wird im Menü der „New Exercise“ Eintrag gewählt welcher das folgende Fenster anzeigt:



- | | |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| File Name | Hier wird der Dateiname der zu erstellenden XML-Datei eingegeben. |
| Exercise Name | Name der Aufgabe die erstellt wird. |
| Description | In dem Textfeld kann die Aufgabenstellung der neuen Aufgabe angegeben werden. |
| Add (...) | Ermöglicht das hinzufügen der Klassen und Test Dateien für die neue Aufgabe. Wahlweise können Klassen und Tests gemeinsam oder einzeln hinzugefügt werden. |

Die erstellten Klassen und Tests werden in den unteren beiden Feldern dargestellt und können durch Klicken des „ – „ wieder entfernt werden.

3.4 Bearbeiten einer Aufgabe

Im folgenden Menü lässt sich die aktuell geöffnete Aufgabe bearbeiten. Dazu wählt man unter „File“ den Eintrag „Edit Exercise“.

The screenshot shows a window titled "Create New Exercise" with a sub-header "Edit Exercise". The form includes the following elements:

- Exercise Name:** A text input field containing "newTest".
- Description:** A larger text input field containing "Test Description".
- Add Class and Test together:** A text input field with the placeholder "enter name of the class (test name = className + Test)" and a "+" button.
- Add Class:** A text input field with the placeholder "enter name of the class" and a "+" button.
- Add Test:** A text input field with the placeholder "enter name of the test" and a "+" button.
- List of Classes:** A list box with expandable items "Class" and "Main".
- List of Tests:** A list box with an expandable item "MainTest".
- Buttons:** "Save" and "Close" buttons at the bottom right.

Exercise Name Name der Aufgabe die erstellt wird.

Description In dem Textfeld kann die Aufgabenstellung der neuen Aufgabe angegeben werden.

Add (...) Ermöglicht das hinzufügen der Klassen und Test Dateien für die neue Aufgabe. Wahlweise können Klassen und Tests gemeinsam oder einzeln hinzugefügt werden.

Die erstellten Klassen und Tests werden in den unteren beiden Feldern dargestellt und können durch Klicken des „ – „ wieder entfernt werden.

3. Arbeiten mit dem Programm

1. Auswählen einer Aufgabe
2. Im Test Modus muss ein fehlschlagender Test geschrieben werden. Schlägt der Test nicht fehl wird nicht in den nächsten Modus gewechselt.
3. Im Code Modus muss der Benutzer den zuvor geschriebenen Test erfüllen. Wird der Test nicht erfüllt wird nicht in den nächsten Modus gewechselt.
4. Im Refactoring Modus lässt sich der geschriebene Code verbessern.
5. Zu Schritt 2 zurückkehren

Der Modus des Programms kann mit den Schaltflächen am linken Rand gewechselt werden. Es lässt sich jeweils nur das Textfeld des entsprechenden Modus bearbeiten.

Im Code Modus ist es zusätzlich möglich in den vorherigen Modus zu wechseln, falls im vorherigen Schritt ein Test geschrieben wurde der nie erfüllbar ist.



4. Zusätzliche Funktionen

4.1. Babysteps

Sind Babysteps eingeschaltet steht für die Modi Test und Code nur eine vorher eingestellte Zeit zur Verfügung. Werden nach Ablauf der Zeit die entsprechenden Anforderungen nicht erfüllt, wird der Code/Test gelöscht und es wird in den vorherigen Modus gewechselt. Es kann auch vor ablaufen der Zeit in den nächsten Schritt gewechselt werden.

Die verbleibende Zeit wird links angezeigt. Im Refactor Modus ist die Zeit nicht begrenzt.

4.2. Tracking

Die Tracking Analyse zeigt an, wie viel Zeit in den einzelnen Phasen benötigt wurde. Zusätzlich werden fehlgeschlagene Tests und Compiler Fehler gezählt.

