

Benutzerhandbuch

Inhalt

Vorab	2
Funktionalität	3
Settings:	4
Task List:	5
TDDT	6

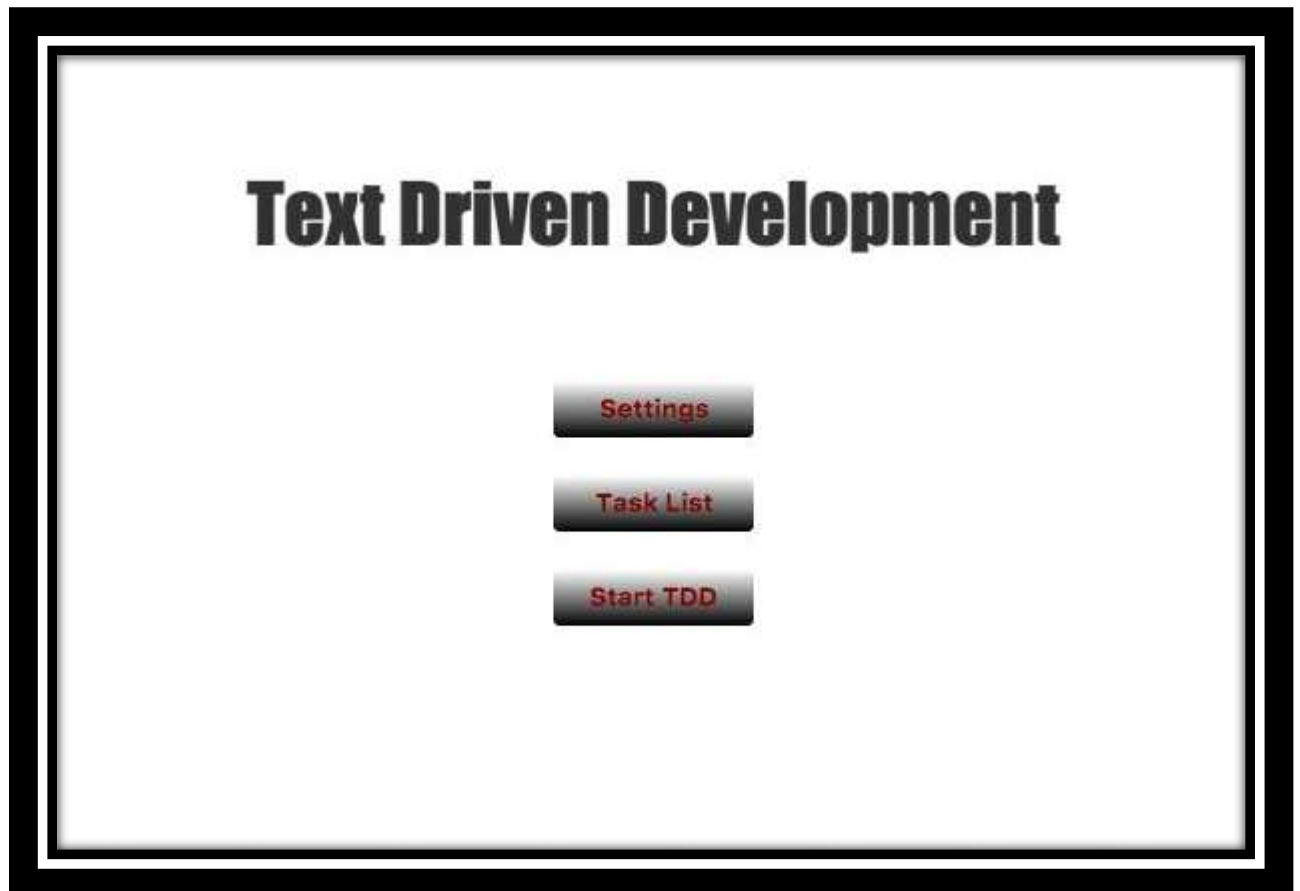
Vorab

Das TDDT Programm soll dazu benutzt werden um testgetriebenes Programmieren zu fördern. Wenn man sich damit noch nicht befasst hast hier ein weiterführender [Link](#).

Info I Studenten sollen anhand dieses Programm ihre Java Skills verfeinern können in einer kleinen überschaubaren Entwickler Umgebung. Denn „testdriven programming“ ist nicht nur sicher sondern garantiert, dass das Projekt immer funktioniert.

Funktionalität

Hauptmenü:



Settings:

Ruft das Settings Menü auf.

Task List:

Ruft das Aufgabenmenü auf.

Start TDD:

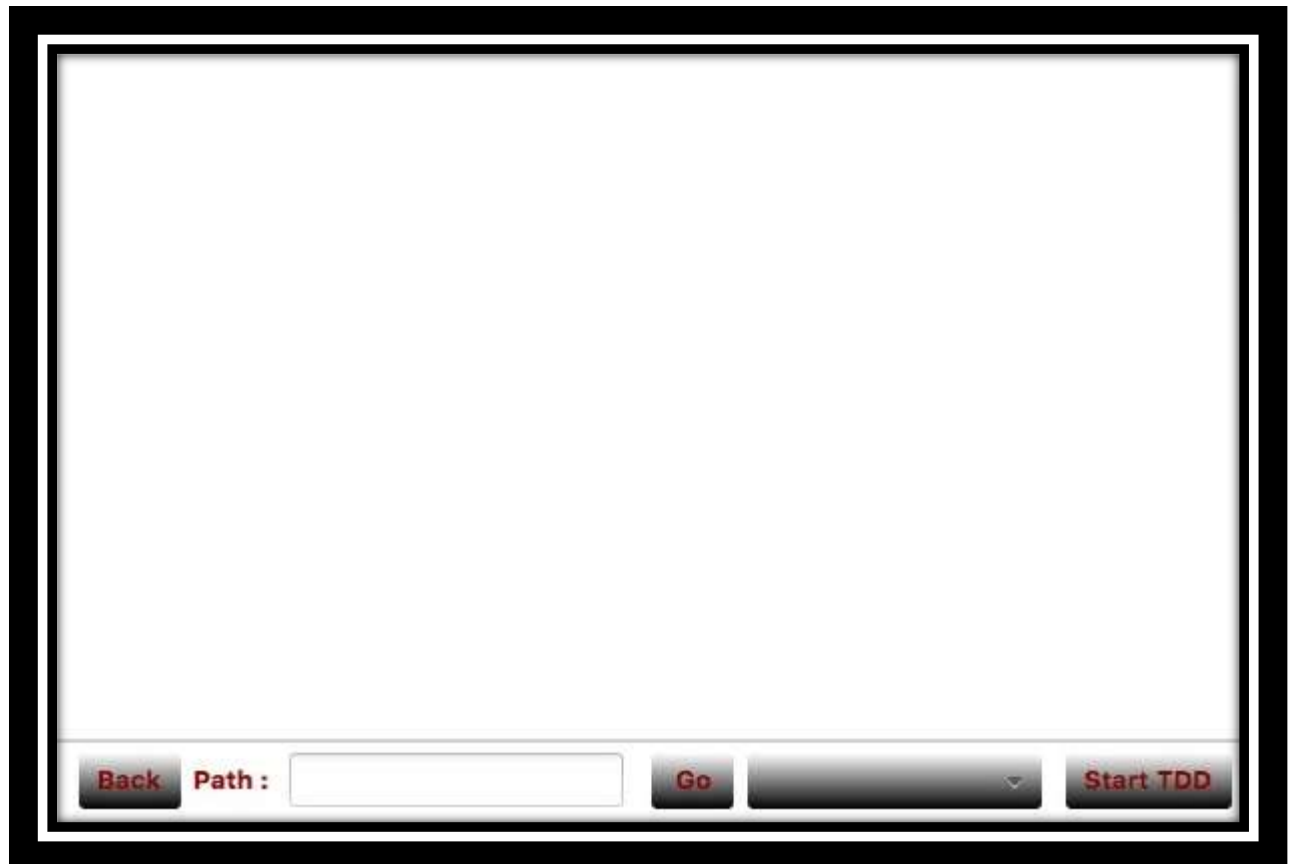
Startet den Testentwicklungsbereich.

Settings:



Mit dem Back Button kehrt man zurück in das Hauptmenü und der Start TDD Button öffnet den Testentwicklungsbereich. In diesem Settings Menü kann man sich für die beiden Plugins Tracking oder BabySteps entscheiden.

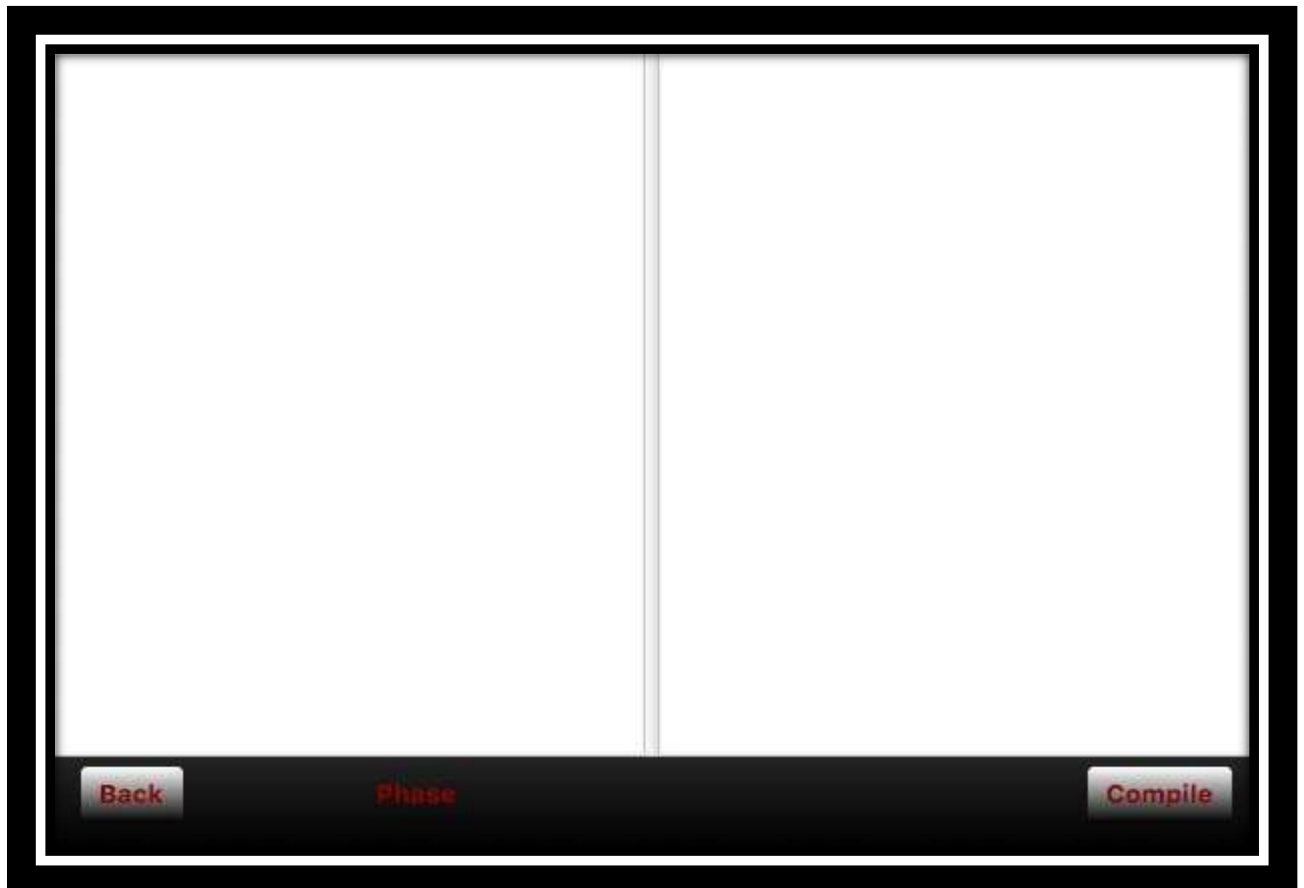
Task List:



The screenshot displays a software interface for task management. It features a large, empty rectangular area at the top, likely for displaying task details or a list. Below this area is a horizontal control bar. On the left of this bar is a 'Back' button. Next to it is a label 'Path :' followed by a text input field. To the right of the input field is a 'Go' button. Further right is a dropdown menu with a downward-pointing arrow. On the far right of the control bar is a 'Start TDD' button. The entire interface is enclosed in a thick black border.

Der Benutzer muss unter „Path:“ den Pfad zu den Aufgaben angeben, die der Nutzer benutzen möchte. Über das Dropdown Menü kann der User sich dann für eine Aufgabe entscheiden. Diese wird dann in der Mitte des Programms angezeigt. Mit Start TDD startet man dann die Entwicklungsumgebung.

TDDT:



Hier ist das Herz des Programms, auf der linken Seite kann man seinen Test schreiben, drückt man dann auf den Compile Button so wird überprüft, ob der Test einen (aber auch nur einen) Fehlschlag hat. Sobald diese Bedingung erfüllt ist, darf der Benutzer in die nächste Phase überwechseln und seinen Code in den rechten Teil des Bildschirms zu schreiben. Wenn man nun auf den Compile Button drückt wird der Test überprüft und der Code. Wenn beide fehlerfrei laufen, darf der Benutzer in die nächste Phase gehen und seinen Code und Test anpassen. Wenn danach beide weiterhin funktionieren, ist ein Kreislauf abgeschlossen und der User befindet sich wieder in der Testphase. Man kann also nun wieder einen neuen Test schreiben und danach den Code.

Unter Phase kann der User jederzeit überprüfen in welcher Phase er sich momentan befindet.