

## Test Driven Development Trainer (TDDT) by GG Ez :

Main.java:

- führt das Programm aus
- lädt MainView.fxml

MainController.java:

- koordiniert alle Aktionen in der Hauptoberfläche
- interagiert mit allen anderen Classe

Exercise.java:

- erzeugt mithilfe eines vorhandenen .xml files ein Objekt, das die zu bearbeitende Aufgabe ist

EditExerciseController.java:

- kontrolliert die Aktionen um ein vorhandenes Exercise Objekt zu bearbeiten

NewExerciseController.java:

- kontrolliert die Aktionen um ein neues Exercise Objekt, bzw. ein neues .xml file zu erzeugen

CloseWindow.java:

- dieses Fenster poppt up wenn der User das Programm beenden möchte und fragt nochmal nach

Modus.java:

- Objekt, das den aktuellen Modus (Test, Code, Refactor) beinhaltet und die Methoden um den Modus zu ändern

Tracking.java:

- In diesem Objekt wird festgehalten wie oft das Kompilieren/ Testen nicht erfolgreich durchgeführt wurde und gemessen wie viel Zeit der User benötigt um einen Zyklus bzw. einen Teil- Zyklus zu absolvieren

Analysis.java:

- Benutzt das im MainController.java erzeugte Objekt um die Daten graphisch ausgeben zu können

ProgramStyle.java:

- beinhaltet einfach die verschiedenen Fonts/ Styles des Programms

Compiler.java:

- die eingebundene Bibliothek von Jens um die .xml files kompilieren zu können

BabystepsTimer.java:

- hier wird ein separater Thread gestartet um die Zeit bei den Babysteps zu berechnen

BabystepsOptions.java:

→ hier kann ausgewählt werden wie lang ein Babystep dauern soll

AboutScreenController.java:

→ handelt den AboutScreen (also einfach das Bild)

HelpFiles.java:

→ kontrolliert die Aktionen um zum Beispiel das Benutzerhandbuch zu laden etc.

FileHandler.java:

→ durch diese Class kann man die Exercises, also die .xml files speichern und laden