

Protokoll

Datum: 28.06.2016, 14:35 – 16:50 Uhr

Autor: Marcel Käufler

Teilnehmer:

Antonia Pütz

Benedikt Schmeitz

Marcel Käufler

Simon Weber

Luisa Schuhmacher

Geplante Themen

- Fortschritte/Probleme der letzten Woche
- Besprechung Projektdesign
- Aufgabenverteilung

Probleme

Push führte zu Build-Fail bei Travis:

Grund: Travis hat mit jdk 1.7 gebaut, es war aber jdk 1.8 notwendig.

Lösung: jvm in .travis.yml angegeben

Bei manchen Teammitgliedern führt das Bauen eines Eclipse Projekts mit gradle eclipse dazu, dass die Testfälle des Executors nicht mehr fehlerfrei laufen.

Grund: Gradle erstellt auch die .classpath Datei, in welcher die zur Ausführung verwendete JRE gesetzt ist. Dabei wird die standard Java SE 1.8 gesetzt. Da für das Ausführen „der Bibliothek“ aber auch ein Java Compiler benötigt wird, muss in Eclipse anschließend wieder das jdk1.8 ausgewählt werden.

Lösung: - offen -

Vorstellung der Fortschritte

Marcel:

- Exercise Klasse zur Darstellung von Übungen mit zugehörigen Klassen und Tests.
- CatalogDatasource zum Laden des Exercise-Katalogs aus einer XML Datei.

Luisa:

- Executor Klasse zur Ausführung von Code und Testfällen einer Exercise
- ExecutionResult als Rückgabotyp eines Kompilier- und Ausführungs-Versuches

Antonia

- GUI Hauptfester mit Editor für Code und Testsfälle

Benedikt, Simon

- Entwurf CRC Karten für Programmlogik

Besprechung Projektdesign

Benötigte Komponenten:

- Logikklasse, welche entsprechend der aktuellen Phase Code und Tests ausführt und einen Status gibt ob in die nächste Phase fortgefahren werden kann
- Controller-Klasse für GUI Steuerung
- PhasenManager (verwaltet die aktuelle Phase und testet die aktuellen
 - Interface erstellt um Implementierung der Komponenten von unterschiedlichen Projektmitgliedern zu ermöglichen
- TrackingManager für die Tracking Erweiterung

Diskussion über Strukturierung der Komponenten und deren Hierarchie.

z.B. beim Auslösen eines Resets (zurückkehren zum Anfang des aktuellen Zyklus und Verwerfen aller Codeänderungen) durch GUI Button oder Babysteps Erweiterung nach abgelaufener Zeit:

Welche Komponente ruft welche Funktion auf? z.B. kein Aufruf von GUI Methoden aus tieferen Schichten. Verwendung von Listnern / EventBus

(<https://github.com/google/guava/wiki/EventBusExplained>)?

Aufgabenverteilung

Luisa: PhaseManager implementation

Marcel: GUI Katalogauswahl

Antonia: GUI und Controller

Benedikt: Erweiterung Babysteps

Simon: Erweiterung Tracking