

Gesamtengagement im Projekt

4-5 Punkte

Der Schüler/Die Schülerin bringt sich gewinnbringend in die Planung und Durchführung des Projekts ein. Er/ Sie erledigt verlässlich und zielbewusst die zugewiesenen Aufgaben (Tasks). Die verwendete Zeit und die erzielten Fortschritte stehen in einem guten bis sehr guten Verhältnis zueinander.

2-3 Punkte

Das Engagement im Projekt ist noch zielführend. Die Erledigung der Aufgaben wird nach Aufforderung größtenteils durchgeführt. Die verwendete Zeit und die erzielten Fortschritte stehen noch in einem akzeptablen Verhältnis zueinander.

0-1 Punkte

Das zielgerichtete Engagement ist kaum erkennbar. Der Schüler/die Schülerin ermöglichen nur schwer einen Projekterfolg. Es muss häufig ermahnt und zur Arbeit aufgefordert werden.

Arbeiten im Scrum-Prozess

Die Prozessschritte

- Backlog-Erstellung & Pflege
- Sprint-Planning
- Sprint-Durchführung (Daily & Task-Abarbeitung)
- Sprint-Review
- Sprint-Retrospektive

werden folgendermaßen ausgeführt:

4-5 Punkte

Der Schüler/die Schülerin bringt sich aktiv und zielführend in den SCRUM-Prozess mit ein. Die Mitarbeit in den einzelnen Prozessschritten ist engagiert und folgt den SCRUM-Regeln. Die Prozessschritte werden angemessen dokumentiert (z. B. Backlog-Items, SCRUM-Board, Versionierung, Protokoll-Retrospektive).

2-3 Punkte

Die Zusammenarbeit und eigene Arbeitsleistung im SCRUM-Prozess ist erkennbar aber nicht unbedingt zielführend. Prozessschritte werden inhaltlich und zeitpunktmäßig nur mäßig gut umgesetzt.

0-1 Punkte

Die nötige Zusammenarbeit im SCRUM-Team sowie die Einhaltung der Prozessschritte ist kaum bis gar nicht erkennbar. Das Arbeiten nach dieser Projektmethode wird nicht/schwer erkennbar umgesetzt.

Selbstkompetenz

Fachlicher eigenverantwortlicher Wissenszuwachs (Persönliche Lernkurve)

4-5 Punkte

Der Schüler / die Schülerin entwickeln ihr fachliches Wissen entlang der gestellten Aufgaben zielorientiert und eigenständig. Alle schulischen Lernziele werden erreicht und darüber hinaus werden weitere Technologien/Methoden angewendet. Der Schüler / die Schülerin stellt zielführende Fragen und arbeitet aktiv am eigenen Wissenszuwachs.

2-3

Die Herausforderungen werden angenommen, bearbeitet und führen zu einem Wissenszuwachs. Schulische Lernziele werden nur zum Teil erreicht. Der Schüler, die Schülerin kann ihren Wissenszuwachs durch ein konzentrierteres Arbeiten noch verbessern.

0-1

Der fachliche Wissenszuwachs leidet erkennbar unter der Arbeitshaltung/Motivation/Engagement. Die Aneignung von Kompetenzen leidet.

Fachkompetenz

0-5 Architektur & Programmierung

4-5

- Die eigene Software-Komponente läuft ohne erkennbare Fehler
- Die Software folgt durchgängig einem modularen Aufbau durch ein Architekturmuster (MVC, MVVM).
- Konzepte der Objektorientierung werden durchgängig sinnvoll umgesetzt.
- Die Pflege des Codes erfolgt mittels Versionierungs-Tools in sinnvoller Weise.
- Daten werden mittels Persistenzschicht gespeichert. Es werden durchgehend OR-Mapper-Frameworks angewendet und verstanden.
- CRUD-Szenarios sind sinnvoll umgesetzt.
- Die GUI ist intuitiv bedienbar.
- Es wird ein gemeinsamer StyleGuide eingehalten.
- Es werden zusätzliche Programmiertechniken angewendet. (Zusatzpunkte)
Z. B.
 - Exceptions
 - Frameworks (Entity, Hibernate, Material, usw.)
 - using-Statement
 - Lambdas, Linq
 - Unit-Tests

2-3

- Die eigene Software-Komponente läuft mit wenigen unkritischen Fehlern.
- Die Software folgt in wesentlichen Teilen einem modularen Aufbau.
- Konzepte der Objektorientierung werden umgesetzt, sind jedoch nicht immer durchgängig und sinnvoll.
- Die Pflege des Codes erfolgt mittels Versionierungs-Tools. Die Abgleichphasen (Commit, Push, Pull) im Team sind nicht immer effizient.
- Daten werden mittels Persistenzschicht gespeichert. OR-Mapper werden nicht bzw. teilweise verwendet.
- CRUD-Szenarios funktionieren. Sie sind jedoch nicht vollständig implementiert.
- Die GUI ist nicht vollständig sinnvoll aufgebaut und umständlich zu bedienen. Stilbrüche sind erkennbar.
- Ein gemeinsamer Styleguide wird nur teilweise eingehalten.

0-1

- Die eigene Software-Komponente läuft nicht oder mit überwiegend fehlerhafter bzw. geringer Funktionalität.
- Ein modularer Schichten-Aufbau ist nicht/kaum erkennbar.
- OOP wird nicht sinnvoll verwendet.
- Quellcode wird nicht geshared. Es bestehen unterschiedliche Stände. Eine Quellcode-Verwaltung nur wenig bis gar nicht genutzt.
- Styleguide ist kein Thema.