Agile Softwareentwicklung mit SCRUM Refactoring with M. Fowler

Version 0.2

20.06.2020

# 

# Dokumentenversionen

|  |  |  |  |
| --- | --- | --- | --- |
| Versionsnr. | Datum | Autor | Änderungsgrund / Bemerkungen |
| 0.1 | 02.06.2020 | Dr. Marc Schanne | Ersterstellung |
| 0.2 | 20.06.2020 | Gennaro Izzo | Review |
|  |  |  |  |
|  |  |  |  |

Inhalt

[Dokumentenversionen 1](#_Toc43487186)

[Epic 1: Projektplanung mit Meilensteinen und Definition von Artefakten 3](#_Toc43487187)

[User-Story 1: 3](#_Toc43487188)

[Epic 2: Refactorings entsprechend der Vorgaben von Martin Fowler 5](#_Toc43487189)

[User-Story 2: 5](#_Toc43487190)

[User-Story 3: 5](#_Toc43487191)

[User-Story 4: 5](#_Toc43487192)

[User-Story 5: 5](#_Toc43487193)

[User-Story 6: 5](#_Toc43487194)

# Epic 1: Projektplanung mit Meilensteinen und Definition von Artefakten

Epic: Projektumsetzung mithilfe agiler Methoden und zugehöriger Artefakte

* Backlog muss vorliegen
  + User-Stories sollten in Sprints verteilt werden (Sprint Planning)
  + User-Stories sollten in entsprechende Tasks eingeteilt werden
  + Tasks sollten nachweislich umgesetzt werden (siehe pushes)
* Durchführung eines Backlog-Groomings pro Sprint
  + Hierzu gehört auch die Bestimmung der Story-Points (ggf. Story Point Poker?)
* Durchführung eines Sprint plannings vor Beginn eines jeden Sprints
* Sprint-Review sollte durchgeführt werden
* Github o.ä. um Code-Iterationen (pushes) zu erkennen (sollten zu User-Stories, bzw. den daraus abgeleiteten Tasks passen)

Die hierfür erforderliche Arbeit eines Product-Owners wurde schon teilweise abgenommen, dies beinhaltet:

* Erstellen relevanter User-Stories aus dem entsprechenden Haupt-Epic
* Priorisieren der User-Stories nach diversen Faktoren (Bearbeitungs-, bzw. Relevanz-Priorisierung wird durch die Nummerierung, bzw. Reihenfolge der Refactoring-User-Stories vorgegeben)

Folgende Punkte sind für die vorhandenen User-Stories noch zu erledigen (Definition of Ready):

* Story-Point-vergabe je User-Story
* Verständnis im Team
* DoD geklärt?

Aufbau einer User-Story:

Als [Nutzer / Rolle] möchte ich [was?], dass / weil / um [warum? / Nutzen / Wert].

### User-Story 1:

Als „Software Quality Engineer“ möchte ich, dass sich das Entwicklungsteam im zu bearbeitenden Coding auskennt, um eine hohe Entwicklungsqualität und -geschwindigkeit zu gewährleisten.

* Beispielstask: Zusammensetzen und Code (IST-Stand) gemeinsam analysieren.
* Definition of Ready:
  + Definition of Done geklärt (Projektübergreifend)
  + Story-Points vergeben (Grooming)
  + jedem Teammitglied ist klar, dass es darum geht, den Code zu verstehen
  + In Task(s) unterteilt -> z.B. Einrichten der Entwicklungsumgebung + erstes öffnen des Projekts (jeder im Team) + gemeinsame Analyse / selbstständige Analyse etc.

# Epic 2: Refactorings entsprechend der Vorgaben von Martin Fowler

Die in den Folien von Martin-Fowler beschriebenen Arbeitsschritte werden im Folgenden in Form von User-Stories abgebildet (siehe Anforderung 1).

### User-Story 2:

Als „Software Quality Engineer“ möchte ich, dass die switch-Anweisung zur Preisberechnung in eine eigene Methode ausgelagert wird, sodass hierdurch die Übersichtlichkeit und Wartbarkeit des Codes gesteigert wird.

Tasks:

* Amount-Calculation in eigene Methode auslagern (entsprechende Benennung)
* Aufruf der neuen Methode in altem Code
* Interne Variablen der extrahierten Methode umbenennen

### User-Story 3:

Als „Software Quality Engineer“ möchte ich, dass sich die Methode zur Preisberechnung in der hierzu geeigneten Klasse befindet, sodass die Kohäsion des Codes erhöht wird.

Tasks:

* Methode in Rental-Klasse verschieben
* Methodenaufruf anpassen

### User-Story 4:

Als „Software Quality Engineer“ möchte ich, dass temporäre Variablen vermieden und die Methode zur Preisberechnung soweit möglich als Query aufgerufen wird, sodass die Code-Menge reduziert wird.

### User-Story 5:

Als „Software Quality Engineer“ möchte ich, dass die Berechnung der frequentRenterPoints in die relevante Klasse ausgelagert (Kombination aus Extract und Move Method), um ???

### User-Story 6:

Als „Software Quality Engineer“ möchte ich, dass die temp-Variablen „totalAmount“ und „frequentRenterPoints“ aus dem Code entfernt werden, um ???

**Ab hier sollten die Studenten die User-Stories selbstständig beschreiben.**