

Software-Projekt (Software Engineering und Java)

Zielsetzung

Es soll ein Programm erstellt werden, welche auf dem Desktop eines MAC/Windows/Linux-Computer laufen kann. Gesamtaufwand pro Student ca 10h. Am 12.1.2018 soll die Gruppe eine Demo der Arbeit über 5 Minuten durchgeführt werden.

Formulieren Sie eine Aufgabenstellung welche die Ausgangslage, die Zielsetzung und die Teilaufgaben formuliert.

Ausgabenstellung Software Engineering

Erstellen Sie auf GitHub im Markdown-Format ein Dokument mit folgender Strukturierung

- 1) Einleitung, Ziele, Randbedingungen
- 2) Build-Anleitung (checkout, mvn, java -jar xxxxx.jar)
- 3) Kurze Bedienungsanleitung
- 4) User Stories inkl. Akzeptanzkriterium, Aufwandschätzung in Story-Points und Priorisierung
- 5) Releaseplan mit den Ausbaustufen
- 6) Dokumentation Sprint 1
 - a. Taskliste für die Umsetzung der User Story (Schätzung in Stunden)
 - b. Anreicherung der User Stories für die Umsetzung
 - c. UML Package, Klassen- und Sequenzdiagramm
 - d. Dokumentation wichtiger Code Snippets
 - e. Herleitung der Testfälle aus den Akzeptanzkriterien der User Stories
- 7) Dokumentation Sprint 2 (inkl. Punkte a-e)
- 8) Optional: Dokumentation Sprint 3 (inkl. Punkte a-e)
- 9) Optional: Dokumentation Sprint 4 (inkl. Punkte a-e)

Ausgabenstellung Java

Die Umsetzung der Sprints muss mittels Java erfolgen. Die Klassen verfügen über entsprechende Unit-Tests. Die Publikation des Projekts erfolgt auf GitHub. Für die Build-Automatisierung muss Maven benutzt werden. Es soll eine JAR-Datei erstellt werden, welche mittels `java -jar xxxxx.jar` gestartet werden kann.

Die Umsetzung der Ausbaustufen erfolgt gemäss Ihrer Zeitplanung. Es müssen nicht alle Ausbaustufen umgesetzt werden (minimal eine Ausbaustufe). Pro Ausbaustufe soll auf GitHub ein entsprechender Release-Branch mit Tags vorhanden sein.

Bewertungskriterien Software Engineering

- a) Ist die Build-Anleitung verständlich?
- b) Sind die User Stories verständlich und vollständig formuliert?
- c) Existiert ein realistischer Releaseplan?
- d) Liefert die Planung genügend Information für eine Umsetzung?
- e) Wurde ein guter Entwurf erstellt (Verständlichkeit, Kopplung und Kohäsion)?
- f) Sind die Testfälle vollständig und verständlich?
- g) Wie viele Sprints wurden geplant?

Bewertungskriterien Java

- a) Wurden die Vorgaben der Planung umgesetzt?
- b) Wurden die Programmierkonventionen eingehalten?
- c) Sind UnitTests vorhanden?
- d) Wurde Maven eingesetzt?
- e) Sind die Release Branches auf GitHub vorhanden?
- f) Funktioniert das Programm gemäss den Vorgaben?
- g) Kann das Programm mittels `mvn clean package` gebaut und mittels `java -jar xxxxx.jar` gestartet werden.
- h) Wie viele Ausbaustufen wurden umgesetzt?

