

Übung 10

Tests, Dokumentation, Wiederholung

1. Wie kann man als Softwareentwickler sichergehen, dass ein Softwareprodukt nicht beim Akzeptanztest durchfällt?

Die folgenden Aufgaben sind etwas umfangreicher, teilweise Wiederholung bereits bekannter Themen, und sollen Ihnen bei der Klausurvorbereitung helfen. Die Bearbeitung ist (wie immer) freiwillig:

2. Zeichnen Sie eine UML-Grafik (suchen Sie sich die passende Diagrammform aus), die verdeutlichen soll, wie die Kernphasen eines Software-Projektes mit den begleitenden Prozessen zusammenhängen.
3. Dokumentieren Sie das „Caesar-Verschlüsselungsmodul“ aus der Übung 9 für Entwickler und Anwender getrennt, und
4. Implementieren Sie den in dieser Übung „geknackten“ Algorithmus neu, gemäß der Dokumentation (die damit gleichzeitig die Spezifikation darstellt), als Java-Klasse.
5. Überlegen Sie sich eine Lizenz für Ihr Produkt, und verweisen Sie in der Dokumentation an geeigneter Stelle darauf, wie es bei den meisten Softwareprodukten üblich und notwendig ist.
6. Wissensfragen:
 - (a) Mit welchen Tools lassen sich in Projekten, bei denen viele Entwickler am gleichen Quelltext arbeiten, die Fortschritte protokollieren und ggf. auch wieder rückgängig machen?
 - (b) Was genau soll die Systemdokumentation darstellen? Geben Sie ein Beispiel für eine Systemdokumentation einer Java-Klasse an.
 - (c) Warum sind Versionsnummern i.d.R. ungeeignet, um die „Marktreife“ bzw. Aktualität eines Software-Produktes feststellen zu können?
 - (d) Wie lässt sich der „Erfolg“ eines Softwareprojektes, nach Software-Engineering-Methoden, feststellen bzw. messen?
 - (e) Spielen Sie in Gedanken (ohne Implementation) die Entwicklung eines neuen Computerspiels mit Software-Engineering-Methoden von der Spielidee bis zur Markteinführung durch. Welche begleitenden Prozesse sind bei einem Computerspiel relevant, welche nicht?