

Was dieses Dokument NICHT ist:

Diese Fragensammlung ist weder eine Veröffentlichung der Übungsleitung noch des Lehrstuhls für Software Engineering.

Dieses Dokument ist ein inoffizielles Übungsblatt für Studierende der Gruppen 10 und 20. Obwohl ich mich um Korrektheit bemühe, kann ich keine absolute Fehlerfreiheit garantieren. Wem ein Fehler auffällt bitte kurzen Hinweis per Mail an m.schwarz@tum.de.

Die Fragen sind keine alten Klausurfragen. Außerdem umfassen sie weder alles was in der Klausur drankommen kann, noch kommt notwendigerweise alles was hier vorkommt auch in der Klausur vor. Auf jeden Fall sehr wichtig für die Klausur sind (vllt. mit mancher Ausnahme) alle Themen, die auf einem der Übungsblätter bearbeitet wurden.

Diese Fragensammlung dient hauptsächlich dazu, nach dem Lernen die Themen aus der Vorlesung nochmal zu wiederholen und einen Überblick darüber zu bekommen, was schon sitzt und was noch nicht so ganz. Die Angaben in den eckigen Klammern beziehen sich auf die Folie vom Prof. Pretschner, auf der die Antwort zu finden ist.

Außerdem geht es hier nur um die Theoriefragen, die ja nur einen Teil der Klausur ausmachen.

Was dieses Dokument ist:

hoffentlich trotzdem hilfreich ;)

1 Einführung

1. Nennen Sie 5 Merkmale, nach denen Software klassifiziert werden kann ! [Einführung 18-19; Tü 1]
2. Definieren Sie den Begriff des Software Engineering und grenzen Sie ihn vom Begriff der Softwaretechnik ab ! [Einführung 21/40; Tü 1]
3. Zeichnen und erklären Sie das Teufelsviereck des Projektmanagements ! [Einführung 22]
4. Nennen und erklären Sie Brook's Law ! [Einführung 23]
5. Inwiefern liegen Software-Projekte in einem Spannungsfeld ! Nennen Sie 5 mögliche Einflüsse ! [Einführung 24; Tü 1]
6. Erklären Sie den Unterschied zwischen Standard- und Individualsoftware! [Einführung 41]
7. Nennen Sie weitere Aktivitäten die neben der Implementierung zum Software Engineering gehören ! [Einführung 45]

2 Requirements Engineering

8. Nennen Sie 3 typische Probleme beim Requirements Engineering ! [ReqEng 6]
9. Nennen und erklären Sie das Gesetz von Boehm ! [ReqEng 9]
10. Nennen Sie die 4 Aufgabenbereiche des Requirements Engineering ! [ReqEng 11]
11. Definieren Sie die Begriffe funktionale Anforderungen und nichtfunktionale Anforderungen ! Grenzen Sie diese von Pseudoanforderungen ab ! Nennen Sie je 1 Beispiel ! [ReqEng 19]
12. Nenne Sie jeweils eine konkrete nichtfunktionale Anforderung, die in die Kategorien Wartbarkeit, Verfügbarkeit, Performanz, Verwendbarkeit und Robustheit fällt ! [ReqEng 20-21]
13. Inwiefern können hierbei Zielkonflikte auftreten? Wie lassen sich solche Zielkonflikte lösen ? [ReqEng 35-37]
14. Was unterscheidet Lastenheft und Pflichtenheft ? [ReqEng 40]
15. Welche Gliederungspunkte hat ein Lastenheft beispielhaft [ReqEng 42-43]
16. Welche ein Pflichtenheft ? [ReqEng 44-47]
17. Nennen Sie Qualitätseigenschaften für eine Anforderung ! [ReqEng 50]
18. Nennen Sie Qualitätseigenschaften eines Spezifikationsdokuments ! [ReqEng 51]
19. Kann man unter Umständen auf das Requirements Engineering verzichten ? Wenn ja, wann ? Welche Hilfsmittel nutzt man alternativ ? [ReqEng 54]

20. Was unterscheidet Use Case, Beispielszenario, abstraktes Szenario, Typszenario und konkretes Szenario voneinander ? [ReqEng 57-58]
21. UML: Use-Case-Diagramme / Klassendiagramme
22. Was ist Abbot's Technique ? [ReqEng 110]
23. Nennen Sie drei mögliche Sichtweisen auf ein Klassendiagramm [ReqEng 114]

3 Architekturf Entwurf

24. Definieren Sie den Begriff der Softwarearchitektur ! [Architekturf Entwurf 8]
25. Was ist Kopplung, was Kohäsion ? Was ist hier erstrebenswert ? [Architekturf Entwurf 22]
26. Was sind Komponenten / Module, was Schnittstellen, was Interfaces ? [Architekturf Entwurf 28-30]
27. Erklären Sie das Prinzip von Information Hiding ! [Architekturf Entwurf 32]
28. Was ist der Unterschied zwischen einer Komponentenschnittstelle und einer API ? [Architekturf Entwurf 34]
29. Erklären Sie Kovarianz und Kontravarianz ! Wo tritt welche Form der Varianz in Java auf ? [Architekturf Entwurf 41]
30. Syntaktische Kompatibilität: Eher nicht als Theoriefrage, dringend Übung aus der Tü nochmal anschauen...
31. evtl. auch Liskovsches Substitutionsprinzip (besprochen i.d. Übung)
32. UML
33. Erklären Sie die Rolle eines Softwarearchitekten ! [Architekturf Entwurf 60-65]
34. Nennen Sie 4 wichtige Architekturmuster ! [Architekturf Entwurf 69]
35. Was ist der Unterschied zwischen strengen und offenen Schichtenarchitekturen ? [Architekturf Entwurf 73]
36. Aus welchen Schichten besteht üblicherweise eine 3-Schichten-Architektur ? [Architekturf Entwurf 74]
37. Welche 3 Entwurfsphasen unterscheidet man ? Welche Werkzeuge werden jeweils genutzt ? [Architekturf Entwurf 80]

4 Objektorientierter Entwurf

38. Zustandsdiagramme
39. SD
40. Kommunikationsdiagramm
41. Was sind CRC-Cards ? Wozu werden sie genutzt ? Wie sind sie aufgebaut ? [OOEntwurf 32-34]
42. Definieren Sie den Begriff des Entwurfsmusters ! [OOEntwurf 38]
43. Inwiefern ist die Verwendung von Entwurfsmustern sinnvoll ? [OOEntwurf 39]
44. Welche Kategorien von Mustern gibt es ? was unterscheidet sie ? [OOEntwurf 41]
45. Wie kann man Entwurfsmuster klassifizieren ? [OOEntwurf42]
46. Nennen sie die 4 grundlegenden Elemente eines Entwurfsmusters ! [OOEntwurf 43]
47. Erklären Sie die 3 Entwurfsmuster Singleton, Strategie, Kompositum (+ Iterator) ! Wofür werden sie jeweils angewendet ? [OOEntwurf 47-58, für Iterator Übung]

5 Implementierung

48. Nennen und erklären sie die verschiedenen Transformationsarten zwischen Modell und Quellcode ! [Implementierung 4]
49. Abbildungen von Assoziation (vor allem interessant falls es Aufgabe zum Reverse Engineering gibt, als Theorie eher nicht) [Implementierung 14-18]
50. Erklären Sie was mit "Don't live with broken windows" gemeint ist ! [Implementierung 21]
51. Was ist mit "DRY" gemeint ? [Implementierung 25]
52. Nennen und erklären Sie 4 Formen der Dopplung [Implementierung 26]
53. Was ist "You ain't gonna need it" ? [Implementierung 31]
54. Was sind Grundregeln für das Refaktorisieren ? [Implementierung 46]
55. Erklären sie das Prinzip der 2 Hüte ! [Implementierung 48]
56. Nennen sie eine Faustregel wann Refaktorisieren sinnvoll ist ! [Implementierung 52]
57. Was sind Vor- was Nachteile von Indirektionen ? [Implementierung 53]
58. Was sind "Bad Smells" in der Softwareentwicklung ? [Implementierung 56]
59. Nennen Sie einige Refaktorisierungsmöglichkeiten ! [Implementierung 58-65]

6 Testen

60. Grenzen Sie die Begriffe Failure / Fault und Error voneinander ab ! Legen sie mögliche Wirkungszusammenhänge dar ! [Testen 4]
61. Grenzen Sie Debugging vom Testen ab ! [Testen 8]
62. Was sind Unit-, was Integrations-, was System-, was Abnahme und was Regressionstest ? [Testen 9]
63. Nach welchen Kriterien sollen Testfälle ausgewählt werden ? [Testen 21]
64. Was unterscheidet White-Box-Tests von Glass-Box-Tests und was von Black-Box-Tests ? [Testen 22]
65. Erklären Sie die Begriffe der Äquivalenzklassenbildung und der Grenzwertanalyse ! [Testen 25]
66. Was ist Anweisungs-, was Zweig- und was Pfadüberdeckung ? Was meint man mit C_2 - und C_3 -Tests ? [Testen 30-32]
67. Überdeckungen sind auch als Praxisaufgabe gut denkbar !
68. All-p-use-Kriterium bei datenflussorientierter Überdeckung
69. Wie sinnvoll ist der Einsatz von Überdeckungskriterien a-priori, wie sinnvoll a-posteriori ? [Testen 35]
70. Ist partitionsbasiertes oder randomisiertes Testen besser ? [Testen 47]
71. Welche Arten vom Fehler bemerkt man unter Umständen beim Integrationstest ? [Testen 58]
72. Nennen und beschreiben Sie vier mögliche Integrationsstrategien ! [Testen 59-60]
73. Stellen sie Dummies, Stubs, Testtreiber und Stümpfe gegenüber ! [Testen 66]
74. Es gibt bei den Integrationstest verschiedene Vorgehensweisen. Welche sind das und welche Vor-/Nachteile haben sie jeweils ? [Testen 68-73]
75. Definieren Sie System- und Akzeptanztest ! Was unterscheidet die Beiden ? [Testen 75,83]
76. Was meint man mit vertraglicher Akzeptanz, was mit Benutzerakzeptanz ? [Testen 84-85]
77. Was ist TDD ? Was zeichnet es aus ? [Testen 92]
78. Nennen Sie 3 mögliche Vorteile von TDD ! [Testen 95]
79. Nennen Sie 2 mögliche Nachteile von TDD ! [Testen 97]
80. Was sind Regressionstests ? Wieso sind diese wichtig ? [Testen 101]
81. Nennen Sie die größte konzeptbedingte Einschränkung des Testens im Allgemeinen ! [Testen 102]

7 Qualitätssicherung

82. Nennen und erklären Sie 5 verschiedenen Sichten auf Software-Qualität ! [QS 6-7]
83. Was ist Software-Qualität ? Geben Sie zwei verschiedene Definitionen von Software-Qualität an ! [QS 9]
84. Was ist FTA was FME[C]A ? Bei welchem Qualitätskriterium sind diese Methoden von Bedeutung ? [QS 14 + TÛ]
85. In welche Teilattribute wird Qualität in der ISO 9126-Norm zerlegt ? [QS 17-19]
86. Grenzen Sie Validierung und Verifikation voneinander ab ! [QS 26]
87. Nennen Sie 4 Methoden der Verifikation ! [QS 27]
88. Welche unterschiedlichen Lesetechniken gibt es ? [QS 32]
89. Was sind Fehlermuster und wann werden diese eingesetzt ? [QS 35-36]
90. Wieso wird die Technik der Verifikation durch Beweis in der Praxis kaum eingesetzt ? [QS 42]
91. Nennen Sie je eine Laufzeit-, eine Design- und eine Codemetrik ! [QS 44]
92. McCabe, DIT, NOC [!TÛ!]
93. Wofür steht CBO ? Erklären Sie diese Metrik kurz ! [QS 48]
94. GQM [TÛ, wenn dann nicht allzu detailliert]

8 Wartung und Evolution

95. Welche Schritte durchläuft der Wartungsprozess ? [Wartung 7 + TÛ]
96. Nennen Sie 4 Aspekte des Reengineering ! [Wartung 20 + TÛ]
97. Wie kann man Wartungsfälle systematisieren ? [Wartung 21]
98. Nennen Sie einige von Lehmans Gesetzen der Softwareevolution ! [Wartung 23]
99. Nennen Sie Faktoren, die Einfluss auf die Wartung haben ! [Wartung 28]
100. Inwiefern spielt die Theorie der kognitiven Dissonanz, die aus der Psychologie kommt, bei der Softwarewartung eine Rolle ? [Wartung 37]
101. Was sind Wartbarkeitseigenschaften ? Inwiefern sind diese problematisch ? [Wartung 38+TÛ]
102. Nennen Sie 3 Beispiele für reale Wartbarkeitsprobleme ! [Wartung 40]

9 Konfigurationsmanagement

103. Was ist eine Konfiguration ? Was eine Version ? [KonMgmt 4+6]
104. Was ist ein Release ? Zwischen welchen Formen eines Release kann unterschieden werden ? [KonMgmt 11]
105. Nennen und grenzen Sie die beiden Begriffe von Kompatibilität voneinander ab ! [KonMgmt 16]
106. Was ist Aufwärts-, was Abwärtskompatibilität ? [KonMgmt 16]

10 Vorgehensmodelle

107. Was sind die grundlegenden Aktivitäten im Software Engineering ? [Vorgehen 2]
108. Was ist ein Vorgehensmodell ? Welche Fragen beantwortet es ? [Vorgehen 4 + TÜ]
109. Was meint man mit Produkt. was mit Aktivitäts- und Prozess- und was mit Rollenmodell ? [Vorgehen 6]
110. Beschreiben Sie das Wasserfallmodell und seine Nachteile ! [Vorgehen 10]
111. Beschreiben Sie kurz das V-Modell ! [Vorgehen 11]
112. Was ist Prototyping ? Welche Arten kann man unterscheiden ? [Vorgehen 12]
113. Was sind Spiralmodelle ? Was haben sie zum Ziel ? [Vorgehen 13]
114. Was zeichnet agile Vorgehensweisen aus ? [Vorgehen 14]
115. Nennen Sie bedeutende Prinzipien beim XP ! [Vorgehen 15]
116. Nennen Sie ein Beispiel für ein agiles Verfahren und erklären Sie kurz dessen zentrale Bestandteile ! [Vorgehen 16-18]
117. V-Modell XT siehe TÜ

11 Projektmanagement

118. Nennen Sie Rahmenbedingungen, die sich auf Projekt auswirken können ! [PM 4]
119. Welche Ziele verfolgt die Projektorganisation ? [PM 6]
120. Projektlebenszyklus siehe TÜ