

Familienname, Vorname: _____

Firmenadresse: _____

Telefon: _____

Fax: _____

E-Mail-Adresse: _____

Rechnungsanschrift: _____

Schulungsunternehmen: _____

Referent: _____

Foundation Level – Probeprüfung
SET E (v1.3) – GTB-Edition

CTFL-Lehrplan Version v4.0

ISTQB® Certified Tester Foundation Level

Urheberrecht

Copyright © 2023 International Software Testing Qualifications Board (im Folgenden ISTQB® genannt). Alle Rechte vorbehalten.

Übersetzung und Anpassung des englischsprachigen Sample Exams des International Software Testing Qualifications Board (ISTQB®), Originaltitel: Certified Tester, Foundation Level Sample Exams v3.1 und Sample Exam Paper v4.0.

Änderungsübersicht

| Version | Datum | Bemerkungen |
|---------|------------|--|
| | | Hinweis: Die vorliegende Probeprüfung wurde überwiegend abgeleitet und weiterentwickelt aus früheren Probeprüfungsfragen zum ISTQB®-Lehrplan CTFL 2018 (V3.1) und ergänzt mit neu erstellten Fragen. |
| 0.1 | 10.10.2023 | Internal BETA 01 DRAFT version |
| 0.2 | 11.11.2023 | Einarbeitung der Befunde der Reviewer zur deutschsprachigen Fassung. |
| 0.3 | 29.11.2023 | Frage 26 ersetzt, weil ähnlich in SET A der Probeprüfung bereits enthalten; V0.2 gemäß gemeldeten Befunden aus 2. Review verbessert. |
| 0.4 | 29.02.2024 | Befunde aus dem Review bis zum 31.01.2024 eingearbeitet. |
| 0.5 | 12.03.2024 | Internes Review und Befunde eingearbeitet. |
| 0.6 | 27.06.2024 | Englische Fassung ergänzt. |
| 0.7 | 27.07.2024 | Doppelte Fragen durch neue Fragen ersetzt. |
| 1.0 | 28.07.2024 | Finale GTB-Edition |
| 1.1 | 30.10.2024 | Korrektur der Fragen 15, 22 und 33 |
| 1.2 | 18.11.2024 | Korrektur der Frage 1, Distraktor a); Frage 22, Distraktor b); Frage 14, Distraktor a); Frage 29, Ergänzung im Fragenstamm; Frage 36, Distraktor a) |
| 1.3 | 23.03.2025 | Fragen 14, 21, 23, 32 optimiert, CTFL V4.0.2 (errata) im Wording Anpassungen vorgenommen. Korrekturen durch Lektorin; |

Einführung

Dies ist eine Probeprüfung. Sie hilft den Kandidaten bei ihrer Vorbereitung auf die Zertifizierungsprüfung. Enthalten sind Fragen, deren Format der regulären ISTQB®/GTB-Prüfung Certified Tester Foundation Level ähnelt. Es ist strengstens verboten, diese Prüfungsfragen in einer echten Prüfung zu verwenden.

- 1) Jede Einzelperson und jeder Schulungsanbieter kann diese Probeprüfung in einer Schulung verwenden, wenn ISTQB® als Quelle und Copyright-Inhaber der Probeprüfung anerkannt bzw. genannt wird.
- 2) Jede Einzelperson oder Gruppe von Personen kann diese Probeprüfung als Grundlage für Artikel, Bücher oder andere abgeleitete Schriftstücke verwenden, wenn ISTQB® als Quelle und Copyright-Inhaber der Probeprüfung bestätigt wird.
- 3) Jedes vom ISTQB® anerkannte nationale Board kann diese Probeprüfung übersetzen und öffentlich zugänglich machen, wenn ISTQB® als Quelle und Copyright-Inhaber der Probeprüfung bestätigt wird.
- 4) Zu fast jeder Frage wird genau eine zutreffende Lösung erwartet. Bei den Ausnahmen wird explizit auf die Möglichkeit mehrerer Antworten hingewiesen.

Allgemeine Angaben zur Probeprüfung:

Anzahl der Fragen: 40

Dauer der Prüfung: 60 Minuten

Gesamtpunktzahl: 40 (ein Punkt pro Frage)

Punktzahl zum Bestehen der Prüfung: 26 (oder mehr)

Prozentsatz zum Bestehen der Prüfung: 65 % (oder mehr)

Feedback zur vorliegenden Probeprüfung als Ganzes (40 Fragen) oder zu einzelnen Fragen wurden in den deutschsprachigen BETA-Versionen des SET E im Zeitraum September 2023 – Januar 2024 abgegeben von:

Jörn Münzel (ehem. GTB), Horst Pohlmann (GTB), Stephan Weissleder (GTB), Marc-Florian Wendland (GTB), Stephanie Ulrich (GTB), Matthias Hamburg (GTB), Helmut Pichler (ATB), Paul Müller (Software Quality Lab), Andre Baumann (imbus AG), Sabine Gschwandtner (imbus AG), Arne Becher (imbus AG), Christian Odenthal, Joachim Schulz (sepp.med) und Mario Winter (GTB).

| | | | |
|---------|----------|----|------------|
| Frage 1 | FL-1.1.1 | K1 | Punkte 1.0 |
|---------|----------|----|------------|

Welche der folgenden Aussagen beschreibt ein typisches Testziel?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Der Test beginnt kurz vor der Freigabe des Testobjekts, um Fehlerzustände zu finden, die eine Abnahme verhindern. | <input type="checkbox"/> |
| b) | Es erfolgt eine Validierung, dass das Testobjekt so funktioniert, wie es von den verschiedenen Stakeholdern erwartet wird. | <input checked="" type="checkbox"/> |
| c) | Es lässt sich nachweisen, dass alle Fehlerzustände identifiziert wurden. | <input type="checkbox"/> |
| d) | Es lässt sich nachweisen, dass die verbleibenden Fehlerzustände keine negativen Auswirkungen haben werden. | <input type="checkbox"/> |

FL-1.1.1 (K1) Der Lernende kann typische Testziele identifizieren.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Diese Aussage beschreibt ein Testziel, das sich auf die Identifizierung von Fehlerzuständen kurz vor der Freigabe konzentriert. Laut CTFL-Lehrplan V4.0 ist ein Ziel des Testens, Fehlerzustände zu finden und zu beheben, bevor das System in Produktion geht. Allerdings ist es nicht typisch, dass Tests erst kurz vor der Freigabe beginnen; Tests sollten während des gesamten Entwicklungsprozesses durchgeführt werden, um frühzeitig Fehler zu finden und zu beheben (siehe CTFL-Lehrplan V4.0, Abschnitt 1.1.1 und Abschnitt 1.3).
- b) **KORREKT** – Dies ist eines der typischen Ziele des Testens (siehe CTFL-Lehrplan V4.0, Abschnitt 1.1.1, 9. Aufzählungspunkt). Dieses Testziel zielt darauf ab, sicherzustellen, dass das Testobjekt die Anforderungen und Erwartungen der verschiedenen Interessengruppen (Stakeholder) erfüllt. Es geht darum, die Funktionalität, Gebrauchstauglichkeit (Interaktionsfähigkeit, Usability) und andere Aspekte des Testobjekts zu überprüfen, um sicherzustellen, dass es den beabsichtigten Zweck erfüllt.
- c) FALSCH – Steht im Widerspruch zum 2. Grundsatz des Testens “Vollständiges Testen ist unmöglich” (siehe CTFL-Lehrplan V4.0, Abschnitt 1.3) und der daraus folgenden Ableitung, dass man nicht beweisen kann, alle Fehlerzustände gefunden zu haben. Somit ist dies kein typisches Testziel (siehe CTFL-Lehrplan V4.0, Abschnitt 1.1.1).
- d) FALSCH – Um beurteilen zu können, ob ein Fehlerzustand eine Fehlerwirkung verursacht oder nicht, muss der Fehlerzustand bekannt sein. Der Nachweis, dass die verbleibenden Fehlerzustände keine Fehlerwirkungen verursachen, bedeutet implizit, dass alle Fehlerzustände gefunden wurden. Dies widerspricht (analog zu Option c)) dem 2. Grundsatz des Testens (siehe CTFL-Lehrplan V4.0, Abschnitt 1.3). Somit ist dies kein typisches Testziel (siehe CTFL-Lehrplan V4.0, Abschnitt 1.1.1).

| | | | |
|---------|----------|----|------------|
| Frage 2 | FL-1.1.2 | K2 | Punkte 1.0 |
|---------|----------|----|------------|

Welche der folgenden Aussagen beschreibt den Unterschied zwischen Testen und Debugging am BESTEN?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Testen identifiziert die Ursache von Fehlerzuständen. Debugging analysiert die Fehlerzustände und schlägt Präventionsmaßnahmen vor. | <input type="checkbox"/> |
| b) | Dynamisches Testen zeigt Fehlerwirkungen auf, die durch Fehlerzustände verursacht werden. Debugging analysiert und behebt den zugehörigen Fehlerzustand. | <input checked="" type="checkbox"/> |
| c) | Testen beseitigt Fehlerwirkungen; während Debugging Fehlerzustände beseitigt, die Fehlerwirkungen verursachen. | <input type="checkbox"/> |
| d) | Dynamisches Testen verhindert die Ursache von Fehlerwirkungen. Debugging beseitigt die Fehlerwirkungen. | <input type="checkbox"/> |

FL-1.1.2 (K2) Der Lernende kann Testen von Debugging unterscheiden.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Beim Testen werden nicht die Ursachen von Fehlerzuständen identifiziert, sondern Fehlerwirkungen (dynamischer Test) oder Fehlerzustände (statischer Test). Beim Debugging werden die Ursachen von Fehlerwirkungen gefunden (Fehlerzustände) (siehe CTFL-Lehrplan V4.0, Abschnitt 1.1.2, 1. und 2. Absatz). Präventionsmaßnahmen sind weder Bestandteil des Testens noch des Debuggings.
- b) KORREKT – Dynamisches Testen zeigt Fehlerwirkungen auf, die durch Fehlerzustände verursacht werden. Durch Debugging können die Ursachen der Fehlerwirkungen analysiert und die Fehlerzustände beseitigt werden (siehe CTFL-Lehrplan V4.0, Abschnitt 1.1.2).**
- c) FALSCH – Der erste Teil des Satzes ist falsch, da Fehlerzustände und daraus resultierende Fehlerwirkungen nicht durch Testen, sondern nur durch Debugging beseitigt werden (siehe CTFL-Lehrplan V4.0, Abschnitt 1.1.2).
- d) FALSCH – Durch dynamisches Testen können die Ursachen von Fehlerwirkungen (d. h. Fehlerzustände) nicht verhindert werden, sondern nur das Vorhandensein von Fehlerzuständen, die Fehlerwirkungen hervorrufen, nachgewiesen werden (siehe CTFL-Lehrplan V4.0, Abschnitt 1.1.2 sowie Abschnitt 1.3, 1. Grundsatz).

| | | | |
|---------|----------|----|------------|
| Frage 3 | FL-1.3.1 | K2 | Punkte 1.0 |
|---------|----------|----|------------|

Ein Product Owner sagt, dass Ihre Rolle als Tester in einem agilen Team darin besteht, alle Fehlerzustände vor dem Ende jeder Iteration aufzudecken.

Welche der folgenden Optionen ist ein Grundsatz, das als Antwort auf diese (falsche) Aussage verwendet werden könnte?

Wählen Sie **EINE** Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Häufung von Fehlerzuständen. | <input type="checkbox"/> |
| b) | Testen zeigt die Anwesenheit von Fehlerzuständen. | <input checked="" type="checkbox"/> |
| c) | Trugschluss: "Keine Fehler" bedeutet ein brauchbares System. | <input type="checkbox"/> |
| d) | Analyse der Grundursache. | <input type="checkbox"/> |

FL-1.3.1 (K2) Der Lernende kann die sieben Grundsätze des Testens erklären.

Begründung: (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Die Fehlerhäufung hat damit zu tun, wo Fehlerzustände (Defekte) am wahrscheinlichsten zu finden sind, nicht, ob alle gefunden werden können (siehe CTFL-Lehrplan V4.0, Abschnitt 1.3, 4. Grundsatz).
- b) KORREKT – Testen kann das Vorhandensein von Fehlern zeigen, aber nicht deren Abwesenheit beweisen, was es unmöglich macht, zu wissen, ob Sie alle Bugs erwischt haben. Außerdem macht es die Unmöglichkeit des vollständigen Testens für Sie unmöglich, alle Fehlerzustände aufzudecken (siehe CTFL-Lehrplan V4.0, Abschnitt 1.3, 1. Grundsatz).**
- c) FALSCH – Dieser Grundsatz besagt, dass Sie viele Fehler finden und beseitigen können, aber trotzdem ein erfolgloses Softwareprodukt veröffentlichen können, was nicht das bietet, was der Product Owner von Ihnen verlangt (siehe CTFL-Lehrplan V4.0, Abschnitt 1.3, 7. Grundsatz).
- d) FALSCH – Die Grundursachenanalyse ist kein Grundsatz des Testens (siehe CTFL-Lehrplan V4.0, Abschnitt 1.3).

| | | | |
|---------|----------|----|------------|
| Frage 4 | FL-1.4.1 | K2 | Punkte 1.0 |
|---------|----------|----|------------|

Welche der folgenden Optionen ist ein Beispiel für eine Aufgabe, die im Rahmen der Testrealisierung des Testprozesses durchgeführt werden kann?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | Analysieren eines Fehlerzustands. | <input type="checkbox"/> |
| b) | Entwerfen von Testdaten. | <input checked="" type="checkbox"/> |
| c) | Einem Testobjekt eine Version zuordnen. | <input type="checkbox"/> |
| d) | Schreiben einer User-Story. | <input type="checkbox"/> |

FL-1.4.1 (K2) Der Lernende kann die verschiedenen Testaktivitäten und die damit verbundenen Aufgaben erklären.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0, Abschnitt 1.4.1):

a) FALSCH – Das Analysieren eines Fehlerzustands ist Teil des Debuggings, nicht des Testens (siehe CTFL-Lehrplan V4.0, Abschnitt 1.1.2, 2. Absatz).

b) KORREKT – Das Erstellen von Testdaten ist eine Aufgabe der Testrealisierung (siehe CTFL-Lehrplan V4.0, Abschnitt 1.4.1, 6. Absatz).

c) FALSCH – Obwohl ein Tester die Version eines Testobjekts für die Ergebnisberichterstattung identifizieren muss, ist die Zuweisung der Version eines Testobjekts Teil des Konfigurationsmanagements.

d) FALSCH – Das Schreiben einer User-Story ist keine Testaktivität und sollte vom Product Owner gemeinsam mit Fachbereichsvertretern und Testern durchgeführt werden (siehe CTFL-Lehrplan V4.0, Abschnitt 3.1, 2. Absatz).

| | | | |
|---------|----------|----|------------|
| Frage 5 | FL-1.4.2 | K2 | Punkte 1.0 |
|---------|----------|----|------------|

Welche der folgenden Aussagen ist ein Beispiel für einen technischen Faktor, der den Testprozess beeinflusst?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Die Software ist eine Webanwendung, die auf verschiedenen Browsern funktionieren muss. | <input checked="" type="checkbox"/> |
| b) | Die Software ist für einen Finanzdienstleister bestimmt, der strenge Sicherheitsanforderungen hat. | <input type="checkbox"/> |
| c) | Die Software wird mit einer agilen Methode entwickelt, die kurze Iterationen und häufige Releases erfordert. | <input type="checkbox"/> |
| d) | Die Software wird von einem Team getestet, das über unterschiedliche Kompetenzen und Erfahrungen verfügt. | <input type="checkbox"/> |

FL-1.4.2 (K2) Der Lernende kann die Auswirkungen des Kontexts auf den Testprozess erklären.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0, Abschnitt 1.4.2):

- a) **KORREKT** – Dies ist ein Beispiel für einen technischen Faktor, der den Testprozess beeinflusst. Zu den technischen Faktoren, die den Testprozess beeinflussen, zählen die Art der Software, die Produktarchitektur und die verwendete Technologie (siehe CTFL-Lehrplan V4.0, Abschnitt 1.4.2, 4. Aufzählungspunkt).
- b) **FALSCH** – Dies ist kein Beispiel für einen technischen Faktor, sondern für einen Unternehmensbereich, der den Testprozess beeinflusst, da die Kritikalität des Testobjekts, die identifizierten Risiken, die Marktbedürfnisse und die spezifischen gesetzlichen Vorschriften die Teststrategie, die Testverfahren, die Testdokumentation und die Berichterstattung bestimmen können (siehe CTFL-Lehrplan V4.0, Abschnitt 1.4.2, 3. Aufzählungspunkt).
- c) **FALSCH** – Dies ist kein Beispiel für einen technischen Faktor, sondern für den Faktor Softwareentwicklungslebenszyklus, der den Testprozess beeinflusst, da die technologischen Praktiken und die Entwicklungsmethoden die Teststrategie, die Testverfahren, die Testautomatisierung und die Testdokumentation bestimmen können (siehe CTFL-Lehrplan V4.0, Abschnitt 1.4.2, 7. Aufzählungspunkt).
- d) **FALSCH** – Dies ist kein Beispiel für einen technischen Faktor, sondern für den Faktor Teammitglieder, der den Testprozess beeinflusst, da die Kompetenz, das Wissen, der Erfahrungsstand, die Verfügbarkeit und der Schulungsbedarf die Teststrategie, die Testverfahren, die Testautomatisierung und die Testdokumentation bestimmen können (siehe CTFL-Lehrplan V4.0, Abschnitt 1.4.2, 2. Aufzählungspunkt).

| | | | | |
|---------|----------|----|--------|-----|
| Frage 6 | FL-1.4.5 | K2 | Punkte | 1.0 |
|---------|----------|----|--------|-----|

Welche der folgenden Aussagen beschreibt am BESTEN die Unterschiede zwischen der Rolle des Testmanagements und der Rolle des Testens?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Die Rolle des Testmanagements konzentriert sich hauptsächlich auf die Aktivitäten der Testanalyse, des Testentwurfs, der Testrealisierung und der Testdurchführung, während die Rolle des Testens die Gesamtverantwortung für den Testprozess, das Testteam und die Leitung der Testaktivitäten übernimmt. | <input type="checkbox"/> |
| b) | Die Rolle des Testmanagements und die Rolle des Testens sind identisch und können von derselben Person gleichzeitig übernommen werden. | <input type="checkbox"/> |
| c) | Die Rolle des Testmanagements übernimmt die Gesamtverantwortung für den Testprozess, das Testteam und die Leitung der Testaktivitäten, während die Rolle des Testens sich hauptsächlich auf die Aktivitäten der Testanalyse, des Testentwurfs, der Testrealisierung und der Testdurchführung konzentriert. | <input checked="" type="checkbox"/> |
| d) | Die Rolle des Testmanagements und die Rolle des Testens haben keine spezifischen Aufgaben und können je nach Kontext variieren. | <input type="checkbox"/> |

FL-1.4.5 (K2) Der Lernende kann die verschiedenen Rollen beim Testen vergleichen.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0, Abschnitt 1.4.5):

- a) FALSCH – Hier sind die Rollen des Testmanagements und des Testens vertauscht.
- b) FALSCH – Die Rollen des Testmanagements und des Testens sind nicht identisch, weil sie unterschiedliche Aufgaben und Verantwortlichkeiten haben.
- c) **KORREKT – Diese Aussage gibt in Anlehnung an den Lehrplanteil die Unterschiede zwischen den Rollen des Testmanagements und des Testens wieder (siehe CTFL-Lehrplan V4.0, Abschnitt 1.4.5, 2. und 3. Absatz).**
- d) FALSCH – Obwohl die Art und Weise, wie die Rollen ausgeübt werden, je nach Kontext variieren kann, haben sowohl das Testmanagement als auch das Testen spezifische Aufgaben und Verantwortlichkeiten.

| | | | |
|---------|----------|----|------------|
| Frage 7 | FL-1.5.3 | K2 | Punkte 1.0 |
|---------|----------|----|------------|

Während der Besprechung einer User-Story in einem agilen Projekt stellen Sie als Tester einen Widerspruch in der Interpretation eines Akzeptanzkriteriums einer User-Story zwischen dem Product Owner und dem Entwicklungsteam fest. Sie bringen diesen Widerspruch zur Sprache.

Welche der folgenden Optionen beschreibt einen Vorteil der Unabhängigkeit des Testens, der in dieser Situation deutlich wird?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | Tester können verschiedene Arten von Fehlerwirkungen und -zuständen erkennen. | <input type="checkbox"/> |
| b) | Tester können die Hauptverantwortung für die Qualität übernehmen. | <input type="checkbox"/> |
| c) | Die Entwickler können darauf vertrauen, dass die Tester für die gewünschte Qualität der Arbeitsergebnisse sorgen. | <input type="checkbox"/> |
| d) | Tester können Annahmen der Stakeholder hinterfragen. | <input checked="" type="checkbox"/> |

FL-1.5.3 (K2) Der Lernende kann die Vor- und Nachteile des unabhängigen Testens unterscheiden.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Das Erkennen verschiedener Arten von Fehlerwirkungen und -zuständen ist ein Vorteil der Unabhängigkeit des Testens (siehe CTFL-Lehrplan V4.0, Abschnitt 1.5.3, 3. Absatz), wird aber in diesem Szenario nicht genutzt. Hier geht es um den Vorteil des Hinterfragens von Annahmen und Aussagen, insbesondere aus der Sicht des Testens.
- b) FALSCH – Die Übernahme der Verantwortung für die Qualität durch unabhängige Tester ist kein Vorteil, sondern eher ein Nachteil, da die anderen Stakeholder ihre Verantwortung für die Qualität verlieren können (siehe CTFL-Lehrplan V4.0, Abschnitt 1.5.3, 4. Absatz).
- c) FALSCH – Dies beschreibt keinen Vorteil der Unabhängigkeit des Testens, sondern einen Nachteil (siehe CTFL-Lehrplan V4.0, Abschnitt 1.5.3, 4. Absatz, vorletzter Satz: „Die Entwickler verlieren möglicherweise das Gefühl der Verantwortung für die Qualität“).
- d) **KORREKT** – Die Annahmen der Stakeholder zu überprüfen und zu hinterfragen, ist ein Vorteil der Unabhängigkeit des Testens (siehe CTFL-Lehrplan V4.0, Abschnitt 1.5.3, 3. Absatz). Dieser Vorteil kommt in diesem Szenario zum Tragen, da der Tester die Widersprüche in den Annahmen aufzeigen kann und so eine frühzeitige Klärung erreicht sowie ggf. zukünftige Fehlhandlungen der Entwicklung vermeidet.

| | | | |
|---------|----------|----|------------|
| Frage 8 | FL-1.5.2 | K1 | Punkte 1.0 |
|---------|----------|----|------------|

Welche der folgenden Optionen beschreibt am BESTEN die Verantwortlichkeiten, die sich aus dem agilen Whole-Team-Ansatz ergeben?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Tester sind für die Entwicklung von Unittests verantwortlich und übergeben diese an die Entwickler zur Testdurchführung. | <input type="checkbox"/> |
| b) | Fachbereichsvertreter haben die Aufgabe, die Werkzeuge auszuwählen, die das Entwicklungsteam zu verwenden hat. | <input type="checkbox"/> |
| c) | Von den Testern wird erwartet, dass sie gemeinsam mit den Fachbereichsvertretern und dem Entwicklungsteam die Testfälle erstellen. | <input checked="" type="checkbox"/> |
| d) | Von den Entwicklern wird erwartet, dass sie nicht-funktionale Anforderungen (Performance, Usability, Sicherheit etc.) testen. | <input type="checkbox"/> |

FL-1.5.2 (K1) Der Lernende kann die Vorteile des Whole-Team-Ansatzes wiedergeben.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Abhängig von den Fähigkeiten eines Teammitglieds kann jedes Mitglied diese Aufgabe übernehmen. Die Aufgabe ist nicht an eine bestimmte Rolle gebunden (siehe CTFL-Lehrplan V4.0, Abschnitt 1.5.2, 2. Absatz). Daher ist diese Aussage nicht auf den Whole-Team-Ansatz anwendbar.
- b) FALSCH – Der Whole-Team-Ansatz hat keinen hierarchischen Ansatz, sondern löst Aufgaben auf der Grundlage von Kompetenzen und Fähigkeiten, die jede(r) Einzelne einbringt (siehe CTFL-Lehrplan V4.0, Abschnitt 1.5.2, 2. Absatz). Das Team kann daher auch über Werkzeuge entscheiden, nicht einzelne Rollen.
- c) **KORREKT – Tester unterstützen die Fachbereichsvertreter bei der Definition geeigneter Abnahmetestfälle (siehe CTFL-Lehrplan V4.0, Abschnitt 1.5.2, 3. Absatz). Der agile Whole-Team-Ansatz betont die Zusammenarbeit und Verantwortung des gesamten Teams, einschließlich Entwicklern, Testern und Fachbereichsvertretern.**
- d) FALSCH – Abhängig von den Fähigkeiten der Teammitglieder können diese Aufgaben von jedem Teammitglied übernommen werden. Die Aufgaben sind nicht an bestimmte Rollen gebunden (siehe CTFL-Lehrplan V4.0, Abschnitt 1.5.2, 2. Absatz). Daher ist diese Aussage nicht auf den Whole-Team-Ansatz anwendbar.

| | | | |
|---------|----------|----|------------|
| Frage 9 | FL-2.1.2 | K1 | Punkte 1.0 |
|---------|----------|----|------------|

Welche der folgenden Aussagen beschreibt ein bewährtes Verfahren für das Testen in allen Softwareentwicklungslebenszyklus-(SDLC-) Modellen?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Testaktivitäten für eine Teststufe beginnen bereits während der entsprechenden Entwicklungsphase. | <input checked="" type="checkbox"/> |
| b) | Eine Teststufe im Softwareentwicklungslebenszyklus-Modell beginnt, wenn die vorhergehende Teststufe abgeschlossen ist. | <input type="checkbox"/> |
| c) | Das Testen wird als separate Phase betrachtet. Sie beginnt, wenn die Entwicklung abgeschlossen ist. | <input type="checkbox"/> |
| d) | Das Testen wird der Entwicklung als Inkrement hinzugefügt. | <input type="checkbox"/> |

FL-2.1.2 (K1) Der Lernende kann gute Praktiken für das Testen, die für alle Softwareentwicklungslebenszyklen gelten, wiedergeben.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

a) **KORREKT – Das Testen sollte während der Entwicklung beginnen, z. B. Testanalyse und Testentwurf, um Fehlerzustände frühzeitig zu erkennen und zu beheben (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1.2, 3. Aufzählungspunkt).**

b) FALSCH – Dies beschreibt eine Vorgehensweise, die nur für ein sequenzielles Modell gilt und selbst dort nicht unbedingt eine gute Praktik ist (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1).

c) FALSCH – Dies beschreibt die Vorgehensweise im sequenziellen Wasserfallmodell und ist daher keine gute Praktik bzw. kein bewährtes Verfahren für alle Modelle (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1).

d) FALSCH – Das Testen selbst ist kein Inkrement/zusätzlicher Schritt in der Entwicklung, aber *während* der Entwicklung gibt es Inkremente, die getestet werden müssen (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1.1).

| | | | |
|----------|----------|----|------------|
| Frage 10 | FL-2.1.3 | K1 | Punkte 1.0 |
|----------|----------|----|------------|

Welcher der folgenden beschriebenen Entwicklungsansätze definiert Testen NICHT als Treiber der Softwareentwicklung?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Zuerst werden Tests erstellt. Dann wird der Code geschrieben. | <input type="checkbox"/> |
| b) | Die Testfälle steuern die Codierung. | <input type="checkbox"/> |
| c) | Das gewünschte Verhalten einer Anwendung wird durch Testfälle definiert. | <input type="checkbox"/> |
| d) | Tests werden aus Akzeptanzkriterien (Abnahmekriterien) abgeleitet und teilweise automatisiert. | <input checked="" type="checkbox"/> |

FL-2.1.3 (K1) Der Lernende kann die Beispiele für Test-First-Ansätze in der Entwicklung wiedergeben.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Dies ist ein korrekter Ansatz für eine testgetriebene Entwicklung (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1.3, 1. Absatz).
- b) FALSCH – Dies ist ein korrekter Ansatz für eine testgetriebene Entwicklung, hier das Beispiel TDD (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1.3, 2. Absatz, 1. Aufzählungspunkt).
- c) FALSCH – Dies ist ein korrekter Ansatz für eine testgetriebene Entwicklung, hier das Beispiel BDD (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1.3, 4. Absatz, 1. Aufzählungspunkt).
- d) **KORREKT – Dieser Ansatz definiert keine testgetriebene Entwicklung, da in dieser Aussage die Testfälle zwar nach den Akzeptanzkriterien (Abnahmekriterien) entworfen und automatisiert werden, aber nicht unbedingt vor der Entwicklung (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1.3, 1. Absatz, 2. Satz). Die Tests wirken sich nicht unbedingt explizit steuernd auf die Entwicklung aus, somit ist dies kein effektiver Ansatz.**

| | | | |
|----------|----------|----|------------|
| Frage 11 | FL-2.1.5 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Welche der folgenden Aussagen beschreibt im Softwareentwicklungslebenszyklus (SDLC) am BESTEN Shift-Left im Testen?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | Testaktivitäten beginnen so früh wie möglich im SDLC und werden parallel zu den Entwicklungsaktivitäten durchgeführt. | <input checked="" type="checkbox"/> |
| b) | Testaktivitäten beginnen so spät wie möglich im SDLC und werden nach den Entwicklungsaktivitäten durchgeführt. | <input type="checkbox"/> |
| c) | Testaktivitäten beginnen in der Mitte des SDLC und werden parallel zu den Entwicklungsaktivitäten durchgeführt. | <input type="checkbox"/> |
| d) | Testaktivitäten werden über mehrere Phasen des SDLC verteilt und in jeder Phase entsprechend dem Reifegrad des Produkts durchgeführt. | <input type="checkbox"/> |

FL-2.1.5 (K2) Der Lernende kann Shift-Left erklären.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) **KORREKT** – Shift-Left setzt das Prinzip des frühen Testens um, bei dem die Testaktivitäten so früh wie möglich im Softwareentwicklungslebenszyklus beginnen und parallel zu den Entwicklungsaktivitäten durchgeführt werden (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1.5, 1. Absatz).
- b) **FALSCH** – Da Shift-Left das Prinzip des frühen Testens umsetzt, beschreibt diese Option eher das Gegenteil von Shift-Left und ist daher keine korrekte Aussage (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1.5).
- c) **FALSCH** – Shift-Left setzt das Prinzip des frühen Testens um, bei dem die Testaktivitäten so früh wie möglich im Softwareentwicklungslebenszyklus beginnen. Dies legt keinen Zeitpunkt fest, so dass die 'Mitte des Softwareentwicklungslebenszyklus' keine korrekte Aussage für Shift-Left ist (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1.5).
- d) **FALSCH** – Da Shift-Left nicht vorschreibt, dass die Testaktivitäten in mehrere Phasen des Softwareentwicklungslebenszyklus aufgeteilt und in jeder Phase entsprechend dem Reifegrad des Produkts durchgeführt werden, ist diese Aussage in Bezug auf Shift-Left nicht korrekt (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1.5).

| | | | |
|----------|----------|----|------------|
| Frage 12 | FL-2.1.1 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Sie sind als Tester in einem Projekt eingesetzt, das nach dem iterativ-inkrementellen Entwicklungsmodell vorgeht. Welche der folgenden Aussagen sollten Sie in dieser Situation unbedingt berücksichtigen, um die Testaktivitäten optimal zu integrieren?

Wählen Sie genau EINE korrekte Option aus! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | Sie planen den Test als einmalige Aktivität, wenn alle Inkremente des Produkts umgesetzt wurden. | <input type="checkbox"/> |
| b) | Statische Tests sollten ausschließlich auf Komponententestebene stattfinden, um möglichst viele codenahe Fehlerzustände frühzeitig zu finden. | <input type="checkbox"/> |
| c) | Da schnelle Rückmeldung über die Qualität eines Inkrements wichtig ist, sollten die Tester die Regressionstests idealerweise manuell ausführen. | <input type="checkbox"/> |
| d) | Durch die Lieferung neuer Inkremente über verschiedene Iterationen hinweg ist es wichtig, umfangreiche Regressionstests aufzubauen. | <input checked="" type="checkbox"/> |

FL-2.1.1 (K2) Der Lernende kann die Auswirkungen des gewählten Softwareentwicklungslebenszyklus auf das Testen erklären.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0, Abschnitt 2.1.1):

- a) FALSCH – Diese Vorgehensweise beschreibt eine (ebenfalls schlechte) Auslegung des Wasserfallmodells, nicht eine iterativ-inkrementelle Vorgehensweise.
- b) FALSCH – Statische Tests sollten grundsätzlich in jeder Iteration auf allen Teststufen vorgesehen werden, nicht nur im Komponententest (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1.1, 3. Absatz: „Dies impliziert, dass in jeder Iteration sowohl statische als auch dynamische Tests auf allen Teststufen durchgeführt werden können.“).
- c) FALSCH – Es ist zwar korrekt, dass schnelle Rückmeldung für jedes Inkrement nötig ist, allerdings spricht das eher dafür, dass die Regressionstests automatisiert werden und nicht manuell durchgeführt werden.
- d) **KORREKT** – In jeder Iteration wird ein Stück neue Funktionalität geliefert, d. h., dass die Menge der bestehenden und bereits abgenommenen Funktionalität ebenfalls von Iteration zu Iteration wächst. Dies macht umfangreiche Regressionstests notwendig, um das Regressionsrisiko abzumildern (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1.1, 3. Absatz, letzter Satz: „Die häufige Lieferung von Inkrementen erfordert eine schnelle Rückmeldung und umfangreiche Regressionstests.“).

| | | | |
|----------|----------|----|------------|
| Frage 13 | FL-2.2.1 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Ein Testfall hat folgende Eigenschaften:

- **Er basiert auf Schnittstellenspezifikationen.**
- **Der Schwerpunkt liegt auf dem Finden von Fehlerwirkungen in der Interaktion zwischen Komponenten.**
- **Es werden sowohl funktionale als auch strukturbasierte Tests angewendet.**

In welcher der folgenden Teststufen wird dieser Testfall am WAHRSCHEINLICHSTEN ausgeführt?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|-----------------------------|-------------------------------------|
| a) | Komponentenintegrationstest | <input checked="" type="checkbox"/> |
| b) | Abnahmetest | <input type="checkbox"/> |
| c) | Systemtest | <input type="checkbox"/> |
| d) | Komponententest | <input type="checkbox"/> |

FL-2.2.1 (K2) Der Lernende kann die verschiedenen Teststufen unterscheiden.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) **KORREKT** – Der Komponentenintegrationstest konzentriert sich auf das Testen der Schnittstellen und des Zusammenspiels, d. h. der Interaktion zwischen Komponenten (siehe CTFL-Lehrplan V4.0, Abschnitt 2.2.1, 2. Aufzählungspunkt). Basis ist die Schnittstellenspezifikation, wobei neben dem funktionalen Test auch strukturbasierte Tests (White-Box-Tests) sinnvoll sind.
- b) **FALSCH** – Der Abnahmetest konzentriert sich auf die Validierung des Systems und seiner Einsatzfähigkeit und somit auf die Funktion und Leistungsfähigkeit (siehe CTFL-Lehrplan V4.0, Abschnitt 2.2.1, 5. Aufzählungspunkt). Basis sind nicht die Schnittstellenspezifikationen, die Testziele sind nicht auf die interne Kommunikation ausgerichtet.
- c) **FALSCH** – Der Systemtest konzentriert sich auf das Gesamtverhalten und die Leistungsfähigkeit durch funktionale und nicht-funktionale Tests (siehe CTFL-Lehrplan V4.0, Abschnitt 2.2.1, 3. Aufzählungspunkt). Basis sind nicht die Schnittstellenspezifikationen und die Testziele sind nicht auf die interne Kommunikation ausgerichtet.
- d) **FALSCH** – Der Komponententest konzentriert sich auf das Testen isolierter Komponenten (siehe CTFL-Lehrplan V4.0, Abschnitt 2.2.1, 1. Aufzählungspunkt). Basis ist nicht allein die Schnittstellenspezifikation, die Testarten sind zwar spezifikationsorientiert und strukturbasiert, aber die Testziele sind nicht auf die Kommunikation über Schnittstellen hin ausgerichtet.

| | | | |
|----------|----------|----|------------|
| Frage 14 | FL-2.3.1 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Welche der folgenden Optionen ist KEIN Auslöser für Wartung und Wartungstest?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Außerbetriebnahme | <input type="checkbox"/> |
| b) | Korrigierende Änderungen oder Hotfixes | <input type="checkbox"/> |
| c) | Upgrades oder Migrationen der Betriebsumgebung | <input type="checkbox"/> |
| d) | Implementierung neuer Funktionen | <input checked="" type="checkbox"/> |

FL-2.3.1 (K2) Der Lernende kann den Wartungstest und dessen Auslöser zusammenfassen.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Außerbetriebnahme ist ein Auslöser für Wartung und Wartungstest (siehe CTFL-Lehrplan V4.0, Abschnitt 2.3, 3. Absatz, 3. Aufzählungspunkt).
- b) FALSCH – Korrigierende Änderungen oder Hotfixes sind ebenfalls Auslöser für Wartung und Wartungstest, da sie Fehlerbehebungen oder dringende Änderungen am System beinhalten (siehe CTFL-Lehrplan V4.0, Abschnitt 2.3, 3. Absatz, 1. Aufzählungspunkt).
- c) FALSCH – Upgrades oder Migrationen der Betriebsumgebung können auch Wartung und Wartungstests auslösen, da sie Tests der neuen Umgebung sowie der geänderten Software erfordern können (siehe CTFL-Lehrplan V4.0, Abschnitt 2.3.1, 3. Absatz, 2. Aufzählungspunkt).
- d) **KORREKT – Die Implementierung neuer Funktionen ist kein Auslöser für Wartung und Wartungstest, da sie in der Regel in Entwicklungsaktivitäten und nicht in Wartungsaktivitäten durchgeführt wird (siehe CTFL-Lehrplan V4.0, Abschnitt 2.1).**

| | | | |
|----------|----------|----|------------|
| Frage 15 | FL-3.1.2 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Welche der folgenden Aussagen zum statischen Test ist am EHESTEN zutreffend?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Der statische Test ist eine kostengünstige Möglichkeit, Fehlerzustände zu erkennen. | <input checked="" type="checkbox"/> |
| b) | Der statische Test macht den dynamischen Test theoretisch überflüssig. | <input type="checkbox"/> |
| c) | Der statische Test ermöglicht, Laufzeitprobleme frühzeitig im Lebenszyklus zu erkennen. | <input type="checkbox"/> |
| d) | Bei der Prüfung sicherheitskritischer Systeme hat der statische Test einen geringen Stellenwert, da der dynamische Test den Fehlerzustand besser findet. | <input type="checkbox"/> |

FL-3.1.2 (K2) Der Lernende kann den Wert statischer Tests erklären.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) **KORREKT** – Laut CTFL-Lehrplan V4.0, Abschnitt 3.1.2, 1. Satz kann der statische Test, Fehlerzustände in den frühesten Phasen des SDLC aufdecken und erfüllt damit den Grundsatz des frühen Testens. Früh entdeckte Fehlerzustände sind oft viel kostengünstiger zu beheben als Fehlerzustände, die später im Lebenszyklus erkannt werden.
- b) **FALSCH** – Dynamische Tests haben ihre Berechtigung, da sie andere Fehlerarten finden als statische Tests (siehe CTFL-Lehrplan V4.0, Abschnitt 3.1.2).
- c) **FALSCH** – Dies geschieht beim dynamischen Testen (siehe Glossar).
- d) **FALSCH** – Statischer Test ist wichtig für sicherheitskritische Computersysteme (siehe CTFL-Lehrplan V4.0, Abschnitt 3.1.2).

| | | | |
|----------|----------|----|------------|
| Frage 16 | FL-3.2.1 | K1 | Punkte 1.0 |
|----------|----------|----|------------|

Welche der folgenden Aussagen beschreibt KEINEN Vorteil von frühem und häufigem Stakeholder-Feedback?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Häufiges Feedback von Stakeholdern hilft, dass Änderungen der Anforderungen früher verstanden und umgesetzt werden. | <input type="checkbox"/> |
| b) | Häufiges Feedback von Stakeholdern hilft dem Entwicklungsteam, besser zu verstehen, was sie entwickeln. | <input type="checkbox"/> |
| c) | Häufiges Feedback von Stakeholdern hilft dem Entwicklungsteam, sich auf die Features zu konzentrieren, die den größten Nutzen bringen. | <input type="checkbox"/> |
| d) | Häufiges Feedback von Stakeholdern kann zu Missverständnissen über Anforderungen führen. | <input checked="" type="checkbox"/> |

FL-3.2.1 (K1) Der Lernende kann Vorteile eines frühzeitigen und häufigen Stakeholder-Feedbacks erkennen.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Dies ist ein Vorteil von frühem und häufigem Stakeholder-Feedback (siehe CTFL-Lehrplan V4.0, Abschnitt 3.2.1, 2. Absatz: „(...) Änderungen an den Anforderungen verstanden und früher umgesetzt werden.“).
- b) FALSCH – Dies ist ein Vorteil von frühem und häufigem Stakeholder-Feedback (siehe CTFL-Lehrplan V4.0, Abschnitt 3.2.1, 2. Absatz: „Dies hilft dem Entwicklungsteam dabei, besser zu verstehen, was es entwickelt.“).
- c) FALSCH – Dies ist ein Vorteil von frühem und häufigem Stakeholder-Feedback (siehe CTFL-Lehrplan V4.0, Abschnitt 3.2.1, 2. Absatz: „Es ermöglicht ihm, sich auf die Features zu konzentrieren, die für die Stakeholder den größten Nutzen bringen“).
- d) **KORREKT** – Die Aussage ist kein Vorteil nach Lehrplan, da sie den beschriebenen Vorteil negiert (siehe CTFL-Lehrplan V4.0, Abschnitt 3.2.1, 2. Absatz: „Häufiges Feedback der Stakeholder während des SDLC kann Missverständnissen über Anforderungen vorbeugen und sicherstellen, dass Änderungen an den Anforderungen verstanden und früher umgesetzt werden.“).

| | | | |
|----------|----------|----|------------|
| Frage 17 | FL-3.2.4 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Welche der unten aufgeführten Reviewarten ist am BESTEN geeignet, wenn das Review in Übereinstimmung mit dem vollständigen allgemeinen Reviewprozess und mit dem Ziel, möglichst viele Anomalien zu finden, durchgeführt werden soll?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--------------------|-------------------------------------|
| a) | Informelles Review | <input type="checkbox"/> |
| b) | Technisches Review | <input type="checkbox"/> |
| c) | Inspektion | <input checked="" type="checkbox"/> |
| d) | Walkthrough | <input type="checkbox"/> |

FL-3.2.4 (K2) Der Lernende kann verschiedene Arten von Reviews vergleichen und gegenüberstellen.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Ein informelles Review verwendet keinen definierten Prozess (siehe CTFL-Lehrplan V4.0, Abschnitt 3.2.4, 3. Absatz, 1. Aufzählungspunkt).
- b) FALSCH – Das Hauptziel eines technischen Reviews ist die Konsensfindung und Entscheidungsfindung in Bezug auf ein technisches Problem. Ein definierter Reviewprozess und das Auffinden von Anomalien sind möglich, aber nicht ausschlaggebend für die Wahl dieses Reviewtyps (siehe CTFL-Lehrplan V4.0, Abschnitt 3.2.4, 3. Absatz, 3. Aufzählungspunkt).
- c) KORREKT – Die Inspektion nutzt den vollständigen allgemeinen Reviewprozess mit dem Ziel, u. a. so viele Anomalien bzw. Abweichungen wie möglich aufzudecken (siehe CTFL-Lehrplan V4.0, Abschnitt 3.2.4, 3. Absatz, 4. Aufzählungspunkt).**
- d) FALSCH – Ein Walkthrough erfordert nicht grundsätzlich einen definierten Prozess, z. B. kann das individuelle Review entfallen. Darüber hinaus verfolgt ein Walkthrough eine Vielzahl von Zielen, zu denen auch die Aufdeckung von Anomalien gehören kann. Das Finden einer maximalen Anzahl von Anomalien steht jedoch nicht im Vordergrund (siehe CTFL-Lehrplan V4.0, Abschnitt 3.2.4, 3. Absatz, 2. Aufzählungspunkt).

| | | | |
|----------|----------|----|------------|
| Frage 18 | FL-3.2.5 | K1 | Punkte 1.0 |
|----------|----------|----|------------|

Während einer Phase intensiver Projektüberstunden wird eine umfangreiche Systemarchitekturspezifikation an verschiedene Projektteilnehmer versandt, zusammen mit weiteren Informationen und der Ankündigung eines technischen Reviews in drei Tagen. Das technische Review war ursprünglich nicht eingeplant. Es werden keine weiteren Anpassungen an den zugewiesenen Aufgaben der Projektteilnehmer vorgenommen, die während des technischen Reviews als Gutachter fungieren sollen.

Welcher der folgenden Erfolgsfaktoren für Reviews fehlt allein aufgrund dieser Information?

Wählen Sie **EINE** Option! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | Geeignete Art des Reviews. | <input type="checkbox"/> |
| b) | Ausreichend Zeit zur Vorbereitung. | <input checked="" type="checkbox"/> |
| c) | Die Festlegung klarer Ziele und messbarer Endkriterien. | <input type="checkbox"/> |
| d) | Gut geleitete Reviewsitzung. | <input type="checkbox"/> |

FL-3.2.5 (K1) Der Lernende kann die Faktoren, die zu einem erfolgreichen Review beitragen, wiedergeben.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Technische Reviews sind für technische Dokumente wie z. B. eine Systemarchitektur, geeignet, d. h., es wurde eine geeignete Reviewart gewählt (siehe CTFL-Lehrplan V4.0, Abschnitt 3.2.5, 2. Aufzählungspunkt).
- b) **KORREKT – Ausreichende Vorbereitungszeit ist ein wichtiger Erfolgsfaktor für Reviews, aber die Projektteilnehmer machen bereits Überstunden und haben keine zusätzlichen Freiräume für ein „individuelles Review“, da die zugewiesenen Aufgaben nicht reduziert werden. Kurz: Es wird keine ausreichende Vorbereitungszeit eingeplant (siehe CTFL-Lehrplan V4.0, Abschnitt 3.2.5, 5. Aufzählungspunkt). Ausreichende Vorbereitungszeit ist jedoch wichtig, um sicherzustellen, dass die Gutachter die Systemarchitekturspezifikation gründlich überprüfen und qualitativ hochwertiges Feedback geben können. Ohne ausreichende Vorbereitungszeit kann die Effektivität des Reviews beeinträchtigt werden.**
- c) FALSCH – Die Vorgabe klarer Ziele und messbarer Endkriterien ist ein richtiger Erfolgsfaktor, dessen Fehlen aber aus den Informationen im Fragenstamm nicht abgeleitet werden kann. Diese könnten in den "weiteren Informationen" gegeben werden (siehe CTFL-Lehrplan V4.0, Abschnitt 3.2.5, 1. Aufzählungspunkt).
- d) FALSCH – Eine gut geleitete Reviewsitzung kann ein Erfolgsfaktor sein, aber es gibt keinen Grund zu der Annahme, dass die Reviewsitzung aufgrund der bereitgestellten Informationen nicht gut geführt wird (siehe CTFL-Lehrplan V4.0, Abschnitt 3.2.5, 9. Aufzählungspunkt).

| | | | |
|----------|----------|----|------------|
| Frage 19 | FL-4.1.1 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Im Vorfeld einer Sitzung zur Iterationsplanung analysieren Sie eine User-Story und deren Akzeptanzkriterien (Abnahmekriterien). Daraus leiten Sie entsprechende Testfälle ab, um das Prinzip des frühen Testens anzuwenden.

Welches Testverfahren bzw. welchen Testansatz wenden Sie an?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|------------------------------|-------------------------------------|
| a) | White-Box-Testen | <input type="checkbox"/> |
| b) | Black-Box-Testen | <input checked="" type="checkbox"/> |
| c) | Erfahrungsbasierter Test | <input type="checkbox"/> |
| d) | Intuitive Testfallermittlung | <input type="checkbox"/> |

FL-4.1.1 (K2) Der Lernende kann Black-Box-Testverfahren, White-Box-Testverfahren und erfahrungsbasierte Testverfahren unterscheiden.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – White-Box-Testverfahren (auch als strukturbasierte Verfahren bekannt) basieren auf einer Analyse der internen Struktur und Verarbeitung des Testobjekts. Da die Testfälle vom Entwurf der Software abhängig sind, können sie erst nach dem Entwurf oder der Implementierung des Testobjekts erstellt werden (siehe CTFL-Lehrplan V4.0, Abschnitt 4.1, 4. Absatz).
- b) KORREKT – Black-Box-Testverfahren (auch spezifikationsbasierte Verfahren genannt) basieren auf einer Analyse des spezifizierten Verhaltens des Testobjekts ohne Kenntnis der internen Struktur. Das spezifizierte Verhalten ist in der User-Story und den Akzeptanzkriterien festgelegt und kann entsprechend analysiert werden (siehe CTFL-Lehrplan V4.0, Abschnitt 4.1, 3. Absatz).**
- c) FALSCH – Erfahrungsbasierte Testverfahren nutzen das Wissen und die Erfahrung der Tester, um Testfälle zu entwerfen (siehe CTFL-Lehrplan V4.0, Abschnitt 4.1, 5. Absatz). Sie nutzen Spezifikationen nur als Rahmen und nicht als Basis für den Testentwurf (siehe CTFL-Lehrplan V4.0, Abschnitt 4.4).
- d) FALSCH – Die intuitive Testfallermittlung ist ein erfahrungsbasiertes Testverfahren, das auf dem Wissen der Tester beruht (siehe auch Begründung zu Option c)).

| | | | |
|----------|----------|----|------------|
| Frage 20 | FL-4.2.1 | K3 | Punkte 1.0 |
|----------|----------|----|------------|

Ein Gerät zur Messung des täglichen Strahlungseinfalls für Pflanzen ermittelt einen Einstrahlungswert für Sonnenschein. Dieser ergibt sich aus der Kombination der Anzahl der Stunden, in denen eine Pflanze der Sonne ausgesetzt ist (unter 3 Stunden, 3 bis 6 Stunden, über 6 Stunden), und der durchschnittlichen Intensität des Sonnenscheins (sehr niedrig, niedrig, mittel, hoch). Die folgenden Testfälle existieren bereits:

| | Dauer (Stunden) | Intensität | Einstrahlung |
|----|-----------------|--------------|--------------|
| T1 | 1,5 | sehr niedrig | 10 |
| T2 | 7,0 | mittel | 60 |
| T3 | 0,5 | sehr niedrig | 10 |

Wie viele Testfälle müssen mindestens noch erzeugt werden, um eine vollständige Überdeckung **ALLER GÜLTIGEN** Eingabe-Äquivalenzklassen zu gewährleisten?

Wählen Sie **EINE** Option! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | 1 | <input type="checkbox"/> |
| b) | 2 | <input checked="" type="checkbox"/> |
| c) | 3 | <input type="checkbox"/> |
| d) | 4 | <input type="checkbox"/> |

FL-4.2.1 (K3) Der Lernende kann die Äquivalenzklassenbildung zur Ableitung von Testfällen anwenden.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

Die folgenden Äquivalenzklassen können identifiziert werden:

- Dauer:
 1. Unter 3 Stunden
 2. 3 – 6 Stunden
 3. Über 6 Stunden

- Intensität:
 4. sehr niedrig
 5. niedrig
 6. mittel
 7. hoch

Die existierenden Testfälle decken die folgenden gültigen Eingabe-Äquivalenzklassen ab:

| | | |
|----|---------|------------------|
| T1 | 1,5 (1) | sehr niedrig (4) |
| T2 | 7,0 (3) | mittel (6) |
| T3 | 0,5 (1) | sehr niedrig (4) |

Folglich sind die fehlenden gültigen Eingabe-Äquivalenzklassen (2), (5) und (7).

Sie können mit zwei zusätzlichen Testfällen abgedeckt werden, da (2) sowohl mit (5) als auch mit (7) kombiniert werden kann.

Folglich ist Antwort b) **KORREKT**.

| | | | |
|----------|----------|----|------------|
| Frage 21 | FL-4.2.2 | K3 | Punkte 1.0 |
|----------|----------|----|------------|

Eine Smart-Home-App misst die durchschnittliche Temperatur im Haus während der vergangenen Woche und gibt den Bewohnern basierend auf diesem Wert Informationen zur Umweltfreundlichkeit ihres Verhaltens. Das Feedback für die verschiedenen Durchschnittstemperaturbereiche (gerundet auf die nächsten ganzen °C) soll lauten:

| | |
|---------------|----------------------|
| Bis zu 10°C | Eiskalt! |
| 11°C bis 15°C | Ganz schön schattig! |
| 16°C bis 19°C | Cool! |
| 20°C bis 22°C | Zu warm! |
| Über 22°C | Sauna! |

Welches der folgenden Testsets liefert die HÖCHSTE Überdeckung von Grenzwerten, wenn die 2-Wert-Grenzwertanalyse verwendet wird?

Wählen Sie genau EINE korrekte Option aus! (1 aus 4)

| | | | | | | |
|----|-------|-------|-------|-------|------------|-------------------------------------|
| a) | 0°C, | 11°C, | 20°C, | 22°C, | 23°C | <input type="checkbox"/> |
| b) | 9°C, | 15°C, | 19°C, | 23°C, | 100°C | <input type="checkbox"/> |
| c) | 10°C, | 16°C, | 19°C, | 22°C, | 23°C | <input checked="" type="checkbox"/> |
| d) | 14°C, | 15°C, | 18°C, | 19°C, | 21°C, 22°C | <input type="checkbox"/> |

FL-4.2.2 (K3) Der Lernende kann die Grenzwertanalyse zur Ableitung von Testfällen anwenden.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0, Abschnitt 4.2.2):

Bei den angegebenen Eingabe-Äquivalenzklassen ergibt das Zwei-Punkt-Grenzwertverfahren folgende 8 Überdeckungselemente:

10°C, 11°C, 15°C, 16°C, 19°C, 20°C, 22°C, 23°C.

Die Überdeckung durch die Optionen ist folglich jeweils:

- a) FALSCH – 4 von 8 (11, 20, 22 und 23).
- b) FALSCH – 3 von 8 (15, 19 und 23).
- c) KORREKT – 5 von 8 (10, 16, 19, 22 und 23).**
- d) FALSCH – 3 von 8 (15, 19 und 22)

| | | | |
|----------|----------|----|------------|
| Frage 22 | FL-4.2.3 | K3 | Punkte 1.0 |
|----------|----------|----|------------|

Ein System zur Berechnung der Strafe für Geschwindigkeitsübertretungen im Straßenverkehr wird mit folgender Entscheidungstabelle spezifiziert:

| Regeln | | R1 | R2 | R3 | R4 |
|-------------|----------------------|----|----|----|----|
| Bedingungen | Geschwindigkeit > 50 | J | J | N | N |
| | Schulzone | J | N | J | N |
| Aktionen | 250 € Geldstrafe | - | X | - | - |
| | Führerscheinentzug | X | - | - | - |

Ihnen liegen bereits die folgenden Testfälle und deren Eingaben vor:

- TF1: Geschwindigkeit = 65, Schulzone = Ja
- TF2: Geschwindigkeit = 45, Schulzone = Ja
- TF3: Geschwindigkeit = 50, Schulzone = Nein
- TF4: Geschwindigkeit = 49, Schulzone = Nein

Welche der Regeln der Entscheidungstabelle ist (noch) NICHT durch einen Testfall überdeckt?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|---------|-------------------------------------|
| a) | Regel 4 | <input type="checkbox"/> |
| b) | Regel 1 | <input type="checkbox"/> |
| c) | Regel 2 | <input checked="" type="checkbox"/> |
| d) | Regel 3 | <input type="checkbox"/> |

FL-4.2.3 (K3) Der Lernende kann den Entscheidungstabellentest zur Ableitung von Testfällen anwenden

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

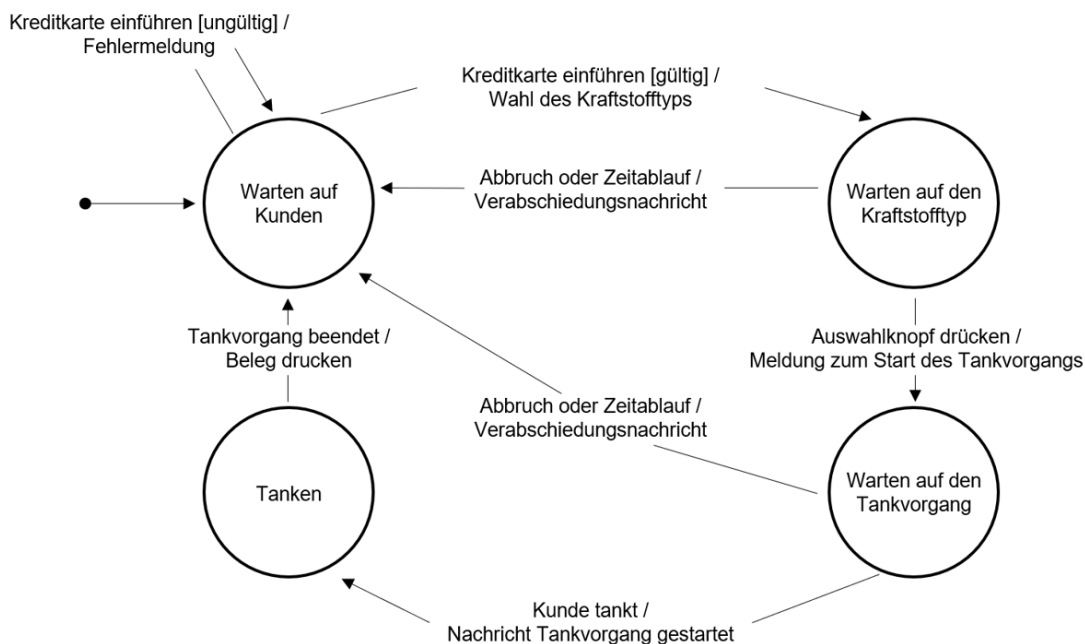
Regel 2 ist nicht überdeckt, da es keinen Testfall für die Eingaben Geschwindigkeit > 50 = Ja und Schulzone = Nein gibt. Ein möglicher Testfall wäre:

TF5: Geschwindigkeit = 51, Schulzone = Nein

- a) FALSCH – Regel 4 ist durch Testfälle 3 und 4 überdeckt.
- b) FALSCH – Regel 1 ist durch Testfall 1 überdeckt.
- c) KORREKT – Regel 2 ist nicht überdeckt.**
- d) FALSCH – Regel 3 ist durch Testfall 2 überdeckt.

| | | | |
|----------|----------|----|------------|
| Frage 23 | FL-4.2.4 | K3 | Punkte 1.0 |
|----------|----------|----|------------|

Betrachten Sie das folgende Zustandsdiagramm für eine Zapfsäule, die nur mit einer Kreditkarte bedient wird:



Nehmen Sie an, dass Sie eine minimale Anzahl von Tests entwickeln möchten, um jeden Übergang im Zustandsdiagramm abzudecken. Nehmen Sie weiter an, dass jeder Test im Anfangszustand beginnen muss, also beim Warten auf den Kunden, und jeder Test endet, wenn ein Übergang im Anfangszustand ankommt.

Wie viele Tests benötigen Sie?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|-----------------------|-------------------------------------|
| a) | 4 | <input checked="" type="checkbox"/> |
| b) | 7 | <input type="checkbox"/> |
| c) | 1 | <input type="checkbox"/> |
| d) | Unendlich viele Tests | <input type="checkbox"/> |

FL-4.2.4 (K3) Der Lernende kann den Zustandsübergangstest zur Ableitung von Testfällen anwenden.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

Jeder Übergang muss mindestens einmal durchlaufen werden. Dabei kann der erste Test einen erfolgreichen Kauf, der zweite Test den Abbruch oder Zeitablauf vom Warten auf das Tanken, der dritte Test den Abbruch oder Zeitablauf vom Warten auf den Kraftstofftyp und der letzte Test das Einführen einer ungültigen Kreditkarte abdecken. Die Reihenfolge ist unerheblich, aber bei weniger als vier Tests wird einer der Übergänge vom Eingang zum Warten auf den Kunden nicht abgedeckt oder die Regeln darüber, wo ein Test beginnt oder endet, werden verletzt. Mehr als vier Tests beinhalten Tests, die bereits abgedeckte Übergänge erneut durchlaufen.

Daher ist Option a) KORREKT.

| | | | |
|----------|----------|----|------------|
| Frage 24 | FL-4.3.1 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Welche der folgenden Beschreibungen der Anweisungsüberdeckung trifft zu?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | Die Anweisungsüberdeckung ist ein Maß für die Anzahl der Quellcodezeilen (ohne Kommentare), die während des Tests ausgeführt wurden. | <input type="checkbox"/> |
| b) | Die Anweisungsüberdeckung ist ein Maß für den prozentualen Anteil der Anweisungen im Quellcode, die im Test ausgeführt wurden. | <input checked="" type="checkbox"/> |
| c) | Die Anweisungsüberdeckung ist ein Maß für den prozentualen Anteil der Quellcodezeilen (ohne Kommentare), die im Test ausgeführt wurden. | <input type="checkbox"/> |
| d) | Die Anweisungsüberdeckung ist ein Maß für die Anzahl der Anweisungen im Quellcode, die während des Tests ausgeführt wurden. | <input type="checkbox"/> |

FL-4.3.1 (K2) Der Lernende kann den Anweisungstest erklären.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

Glossareintrag 'Anweisungsüberdeckung': Die Überdeckung von ausführbaren Anweisungen (Stand 09.2024).

Siehe CTFL-Lehrplan V4.0, Abschnitt 4.3.1, 1. Absatz: „Die Überdeckung wird als Anzahl der durch die Testfälle ausgeführten Anweisungen, geteilt durch die Gesamtanzahl der ausführbaren Anweisungen im Code, gemessen und in Prozent ausgedrückt.“

- a) FALSCH – Die Anweisungsüberdeckung bezieht sich auf die durch Tests überdeckten Anweisungen. Da in einer Quellcodezeile mehrere Anweisungen stehen können oder eine Anweisung sich über mehrere Zeilen erstrecken kann, ist die Anzahl der Quellcodezeilen kein Maß für die Anweisungsüberdeckung (siehe oben).
- b) KORREKT – Der prozentuale Anteil der Anweisungen ist der Anteil im Test ausgeführter Anweisungen bezogen auf alle Anweisungen, also Anzahl durch den Test ausgeführter Anweisungen / Anzahl aller Anweisungen als Prozentwert (siehe oben).**
- c) FALSCH – Die Anweisungsüberdeckung bezieht sich nicht auf Quellcodezeilen, sondern auf Anweisungen, die durch Tests ausgeführt werden (siehe auch Begründung zu a)).
- d) FALSCH – Die Anweisungsüberdeckung bezieht sich nicht auf die absolute Anzahl der von der Testsuite ausgeführten Anweisungen, sondern auf deren Anteil an allen Anweisungen (siehe oben).

| | | | |
|----------|----------|----|------------|
| Frage 25 | FL-4.3.3 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Welche der folgenden Aussagen stellt einen Mehrwert von White-Box-Tests dar?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | White-Box-Tests können Messgrößen für die Überdeckung von z. B. Anweisungen liefern. | <input checked="" type="checkbox"/> |
| b) | White-Box-Tests können überprüfen, ob der Code die Akzeptanzkriterien (Abnahmekriterien) erfüllt. | <input type="checkbox"/> |
| c) | White-Box-Tests können die Kompatibilität mit anderen Systemen testen. | <input type="checkbox"/> |
| d) | White-Box-Tests können alle Fehlerzustände im Code aufdecken. | <input type="checkbox"/> |

FL-4.3.3 (K2) Der Lernende kann den Wert des White-Box-Tests erklären.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) **KORREKT** – White-Box-Tests gehören zur Kategorie der struktur- und codebasierten Testverfahren und können daher Messgrößen für codebasierte Metriken, wie z. B. Anweisungsüberdeckung, liefern (siehe CTFL-Lehrplan V4.0, Abschnitt 4.3.3, 3. Absatz und Abschnitt 4.3.1).
- b) **FALSCH** – White-Box-Tests gehören zur Kategorie der strukturbasierten Testverfahren und haben die Schwäche, dass sie bei Nichterfüllung von Anforderungen die daraus resultierenden Fehlerzustände möglicherweise nicht erkennen können (siehe CTFL-Lehrplan V4.0, Abschnitt 4.3.3, 1. Absatz). Akzeptanzkriterien (Abnahmekriterien) sind eine Art von Anforderungen. Dies ist die Stärke von Black-Box-Tests, die zu den spezifikationsbasierten Testverfahren gehören.
- c) **FALSCH** – White-Box-Tests gehören zur Kategorie der strukturbasierten Testverfahren und konzentrieren sich auf die interne Struktur des Testobjekts (siehe CTFL-Lehrplan V4.0, Abschnitt 4.1, 4. Absatz). Die Kompatibilität, insbesondere die Interoperabilität, mit anderen Komponenten oder Systemen wird in der Regel im Rahmen eines Integrationstests geprüft.
- d) **FALSCH** – White-Box-Tests können nicht alle Fehlerzustände im Code aufdecken. Beispielsweise werden sowohl datenabhängige Fehlerzustände mitunter nicht erkannt (siehe CTFL-Lehrplan V4.0, Abschnitt 4.3.1, letzter Absatz: „So werden beispielsweise Fehlerzustände, die datenabhängig sind, nicht erkannt (z. B. eine Division durch null, die nur fehlschlägt, wenn der Nenner auf null gesetzt wird)“) als auch Fehlerzustände, die auf einem bestimmten, unter Umständen nicht ausgeführten Ausführungspfad liegen (siehe CTFL-Lehrplan V4.0, Abschnitt 4.3.2, 3. Absatz: „Das Ausführen eines Zweigs mit einem Testfall wird jedoch nicht in allen Fällen Fehlerzustände aufdecken. So werden beispielsweise Fehlerzustände, die die Ausführung eines bestimmten Pfades in einem Code erfordern, nicht erkannt.“).

| | | | |
|----------|----------|----|------------|
| Frage 26 | FL-4.4.3 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Sie testen eine mobile App, mit der Kunden auf ihre Bankkonten zugreifen und diese verwalten können. Sie führen eine Testsuite aus, die die Bewertung jedes Bildschirms und jedes Feldes auf jedem Bildschirm anhand einer Sammlung von Heuristiken für Benutzerschnittstellen umfasst. Sie wurde aus einem populären Buch zu diesem Thema abgeleitet und soll die Attraktivität, Gebrauchstauglichkeit (Interaktionsfähigkeit) und Zugänglichkeit für solche Apps maximieren.

Welche der folgenden Testverfahren kategorisiert das von Ihnen verwendete Testverfahren am BESTEN?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|------------------------------|-------------------------------------|
| a) | Entscheidungstabellentest | <input type="checkbox"/> |
| b) | Explorativer Test | <input type="checkbox"/> |
| c) | Checklistenbasiertes Testen | <input checked="" type="checkbox"/> |
| d) | Intuitive Testfallermittlung | <input type="checkbox"/> |

FL-4.4.3 (K2) Der Lernende kann den checklistenbasierten Test erklären.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Das Buch bietet einen allgemeinen Leitfaden und ist kein formales Anforderungsdokument, keine Spezifikation oder eine Sammlung von Anwendungsfällen, User-Stories oder Geschäftsprozessen.
- b) FALSCH – Während Sie die Sammlung als eine Reihe von Test-Chartas betrachten könnten, ähnelt sie eher der Reihe von Testbedingungen.
- c) **KORREKT – Die Sammlung der Best Practices für die Benutzerschnittstellen ist die Liste der Testbedingungen. Checklistenbasiertes Testen verwendet vordefinierte Listen und Kriterien, um die Qualität und Konformität einer Anwendung zu überprüfen. Im o. a. Fall wird eine Sammlung von Heuristiken verwendet, um die Attraktivität, Gebrauchstauglichkeit (Interaktionsfähigkeit) und Zugänglichkeit der mobilen App zu bewerten.**
- d) FALSCH – Die Tests konzentrieren sich nicht auf Fehler, die auftreten könnten, sondern auf die Erkenntnis, was für den Benutzer wichtig ist, im Sinne der Leichtigkeit der Nutzung (Gebrauchstauglichkeit, Interaktionsfähigkeit).

| | | | |
|----------|----------|----|------------|
| Frage 27 | FL-4.4.2 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Für welche der folgenden Situationen ist der Einsatz von explorativen Tests am EHESTEN sinnvoll?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Wenn unter Zeitdruck die Durchführung bereits spezifizierter Tests beschleunigt werden muss. | <input type="checkbox"/> |
| b) | Wenn das System inkrementell entwickelt wird und keine Test-Charta vorhanden ist. | <input type="checkbox"/> |
| c) | Wenn Tester mit ausreichenden Kenntnissen über ähnliche Anwendungen und Technologien zur Verfügung stehen. | <input checked="" type="checkbox"/> |
| d) | Wenn eine umfangreiche Spezifikation des Systems zur Verfügung steht, die für die Testanalyse und den Testentwurf verwendet werden kann. | <input type="checkbox"/> |

FL-4.4.2 (K2) Der Lernende kann den explorativen Test erklären.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Der explorative Test kann nützlich sein, wenn Zeitdruck besteht, ist aber nicht geeignet, die Durchführung bereits spezifizierter Tests zu beschleunigen. Beim explorativen Testen werden die Tests gleichzeitig entworfen, durchgeführt und ausgewertet, aber es werden keine bereits spezifizierten Tests durchgeführt (siehe CTFL-Lehrplan V4.0, Abschnitt 4.4.2, 1. Absatz).
- b) FALSCH – Ein explorativer Test ist unabhängig vom SDLC, egal, ob er inkrementell oder sequenziell entwickelt wird. Der explorative Test sollte eine Test-Charta verwenden (siehe CTFL-Lehrplan V4.0, Abschnitt 4.4.2).
- c) KORREKT – Der explorative Test ist nützlich, wenn Zeitdruck besteht und/oder die Spezifikationen unzureichend sind. Er ist besonders effektiv, wenn die Tester erfahren sind und über Fachwissen verfügen, und kann andere Testverfahren unterstützen (siehe CTFL-Lehrplan V4.0, Abschnitt 4.4.2, 3. Absatz).**
- d) FALSCH – In dieser Situation böten sich vermutlich spezifikationsorientierte Testverfahren an, um Testfälle aus der umfangreichen Spezifikation abzuleiten. Exploratives Testen eignet sich insbesondere dann, wenn keine ausreichende Spezifikation zur Verfügung steht (siehe CTFL-Lehrplan V4.0, Abschnitt 4.4.2, 3. Absatz, erster Satz: „Explorative Tests sind sinnvoll, wenn es nur wenige oder unzureichende Spezifikationen gibt ...“).

| | | | |
|----------|----------|----|------------|
| Frage 28 | FL-4.5.2 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Ein agiles Entwicklungsteam hat folgende User-Story formuliert: „Als Anwender möchte ich, dass der Alarmton der elektronischen Eieruhr in der Lautstärke variabel einstellbar ist, so dass ich ihn immer hören kann.“

Welches der folgenden Abnahmekriterien (Akzeptanzkriterien) ist aus der Sicht des Testens am BESTEN geeignet, um eindeutige Abnahmetests zu entwerfen?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Die Lautstärke ist für jede Person einfach einzustellen, d. h., die Einstelltasten müssen eine nutzbare Größe haben. | <input type="checkbox"/> |
| b) | Der Tester kann den Alarmton auch auf der tiefsten Stufe gut hören. | <input type="checkbox"/> |
| c) | Die Lautstärke kann in einem Bereich von 40 bis 80 Dezibel eingestellt werden. | <input checked="" type="checkbox"/> |
| d) | Die Lautstärkeeinstellung funktioniert in den meistverkauften Modellen dieser Eieruhr korrekt. | <input type="checkbox"/> |

FL-4.5.2 (K2) Der Lernende kann die verschiedenen Möglichkeiten zum Schreiben von Akzeptanzkriterien einordnen.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Abnahmekriterien werden als Testbedingungen betrachtet und sollen daher eine Prüfung auf Korrektheit oder Angemessenheit unterstützen, ohne Realisierungsvorgaben zu machen (siehe CTFL-Lehrplan V4.0, Abschnitte 4.5.2 und 2.2.2). Dies ist hier nicht der Fall, da erstens nicht vorgeschrieben ist, wie die Lautstärkeregelung zu bedienen ist, z. B. als Drehknopf oder über Tasten, und zweitens eine 'nutzbare Größe' keine messbaren Werte vorgibt, z. B. als Verweis auf eine Norm.
- b) FALSCH – Akzeptanzkriterien (Abnahmekriterien) werden als Testbedingungen betrachtet und sollen daher eine Prüfung auf Korrektheit oder Angemessenheit unterstützen (siehe CTFL-Lehrplan V4.0, Abschnitte 4.5.2 und 2.2.2). Dies ist hier nicht der Fall, da erstens das Gehör des Testers individuell und damit als Korrektheitskriterium nicht verlässlich ist und zweitens eine Größe von 'gut hörbar' keinen messbaren Wert liefert.
- c) **KORREKT – Akzeptanzkriterien (Abnahmekriterien) werden als Testbedingungen betrachtet und sollen daher eine Prüfung auf Korrektheit oder Angemessenheit unterstützen (siehe CTFL-Lehrplan V4.0, Abschnitte 4.5.2 und 2.2.2). Dies ist hier der Fall, da der Einstellbereich klar definiert ist und die zugewiesene Lautstärke messbar ist (und den allgemeinen Grenzen von 'leise' bis 'laut' entspricht).**
- d) FALSCH – Akzeptanzkriterien (Abnahmekriterien) werden als Testbedingungen betrachtet und sollen daher eine Prüfung auf Korrektheit oder Angemessenheit unterstützen (siehe CTFL-Lehrplan V4.0, Abschnitt 4.5.2 und 2.2.2). Dies ist hier nicht der Fall, da erstens bei einer Neuentwicklung nicht klar ist, welche Eieruhren die meistverkauften sein werden, und zweitens die Anforderung 'korrekt' nicht spezifiziert, was damit gemeint ist (nicht testbar, nicht messbar).

| | | | |
|-----------------|-----------------|-----------|-------------------|
| Frage 29 | FL-4.5.3 | K3 | Punkte 1.0 |
|-----------------|-----------------|-----------|-------------------|

Bitte betrachten Sie die folgende User-Story:

"Als Systemadministrator möchte ich die Performanz des Servers überwachen können, um über einen Performanztest sicherzustellen, dass das System effizient läuft."

Welcher Testfall eignet sich am BESTEN für eine abnahmetestgetriebene Entwicklung der User-Story?

Wählen Sie eine Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | <p>1. Login als Systemadministrator; wähle den Server aus; überprüfe die Serverleistung. GEGEBEN: Ich bin als Systemadministrator angemeldet UND GEGEBEN: Ich habe den Server ausgewählt, WENN ich „Serverleistung überprüfen“ auswähle, DANN wird mir eine Übersicht über die Performanz des Servers angezeigt.</p> | <input type="checkbox"/> |
| b) | <p>2. Login als Benutzer; führe eine Aufgabe aus; überprüfe die Serverleistung. GEGEBEN: Ich bin als Benutzer angemeldet UND GEGEBEN: Ich habe eine Aufgabe ausgeführt, WENN ich „Serverleistung überprüfen“ auswähle, DANN wird mir eine Übersicht über die Performanz des Servers angezeigt.</p> | <input type="checkbox"/> |
| c) | <p>3. Login als Systemadministrator; wähle den Server aus; führe einen Performanztest durch. GEGEBEN: Ich bin als Systemadministrator angemeldet UND GEGEBEN: Ich habe den Server ausgewählt, WENN ich „Performanztest durchführen“ auswähle, DANN wird ein Performanztest durchgeführt und ich erhalte eine Übersicht über die Ergebnisse.</p> | <input checked="" type="checkbox"/> |
| d) | <p>4. Login als Systemadministrator; führe einen Performanztest durch; überprüfe die Serverleistung. GEGEBEN: Ich bin als Systemadministrator angemeldet UND GEGEBEN: Ich habe einen Performanztest durchgeführt, WENN ich „Serverleistung überprüfen“ auswähle, DANN wird mir eine Übersicht über die Performanz des Servers angezeigt.</p> | <input type="checkbox"/> |

FL-4.5.3 (K3) Der Lernende kann die abnahmetestgetriebene Entwicklung (ATDD) zur Ableitung von Testfällen anwenden.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

Begründung der Testfälle:

- a) FALSCH – weil NICHT GEEIGNET – Obwohl diese Option die Rolle des Systemadministrators und die Aktion der Überprüfung der Serverleistung beinhaltet, fehlt die spezifische Aktion des Durchführens eines Performanztests.
- b) FALSCH – weil nicht GEEIGNET – Diese Option beinhaltet nicht die Rolle des Systemadministrators, was ein Schlüsselaspekt der User-Story ist.
- c) **KORREKT – weil GEEIGNET – Diese Option beinhaltet sowohl die Rolle des Systemadministrators als auch die spezifischen Aktionen des Auswählens des Servers und des Durchführens eines Performanztests, was der User-Story entspricht.**
- d) FALSCH – weil NICHT GEEIGNET – Obwohl diese Option die Rolle des Systemadministrators und die Aktion der Überprüfung der Serverleistung beinhaltet, fehlt das Ausführen des Performanztests.

| | | | |
|----------|----------|----|------------|
| Frage 30 | FL-5.1.2 | K1 | Punkte 1.0 |
|----------|----------|----|------------|

Welche der folgenden Aktivitäten führen Tester während der Releaseplanung im agilen Projekt durch?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | Tester identifizieren und verfeinern funktionale und nicht-funktionale Aspekte des Testobjekts. | <input type="checkbox"/> |
| b) | Tester unterstützen die Ableitung von Aufgaben (Tasks) aus den User-Stories. | <input type="checkbox"/> |
| c) | Tester nehmen an der detaillierten Risikoanalyse der User-Stories teil. | <input type="checkbox"/> |
| d) | Tester unterstützen bei der Erstellung von User-Stories, bei deren Testbarkeit und bei den Akzeptanzkriterien (Abnahmekriterien). | <input checked="" type="checkbox"/> |

FL-5.1.2 (K1) Der Lernende kann den möglichen Mehrwert, den ein Tester für die Iterations- und Releaseplanung schafft, erkennen.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Dies ist eine Aufgabe in der Iterationsplanung: „... und identifizieren und verfeinern die funktionalen und nicht-funktionalen Aspekte des Testobjekts“ (siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.2, letzter Absatz).
- b) FALSCH – Die Releaseplanung beinhaltet (noch) keine Aufgabenplanung, diese ist Teil der Iterationsplanung: „... zerlegen die User-Stories in Aufgaben (insbesondere Testaufgaben), schätzen den Testaufwand für alle Testaufgaben ...“ (siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.2, letzter Absatz).
- c) FALSCH – Die Releaseplanung beinhaltet nicht die Beteiligung an der detaillierten Risikoanalyse der User-Stories, dies ist ein Teil der Iterationsplanung (siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.2, letzter Absatz).
- d) **KORREKT** – Dies ist eine Aufgabe der Tester in der Releaseplanung: „..., beteiligen sich an der Erstellung von testbaren User-Stories und Akzeptanzkriterien (...), ...“ (siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.2, 2. Absatz). Tester arbeiten eng mit dem Produktteam zusammen, um sicherzustellen, dass die User-Stories klar definiert sind, testbare Akzeptanzkriterien (Abnahmekriterien) haben und die Anforderungen der Kunden erfüllen.

| | | | |
|----------|----------|----|------------|
| Frage 31 | FL-5.1.3 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Gegeben seien die folgenden Beispiele für Eingangs- und Endekriterien für einen Systemtest:

1. Das geplante Testbudget von 400 Aufwandsstunden für den Systemtest ist aufgebraucht.
2. Mehr als 95 % der geplanten Testfälle sind vollständig durchgeführt.
3. Die Testumgebung für den Performanztest ist entworfen, eingerichtet und verifiziert.
4. Es sind keine Prio-1-Fehlerzustände und maximal 4 Prio-2-Fehlerzustände offen.
5. Die Designspezifikation ist durch ein technisches Review geprüft und freigegeben.
6. Der Unittest für die Komponenten 'Steuersatz' und 'Gesamtpreis' ist abgeschlossen und freigegeben.

Welche der folgenden Kombinationen ordnet die Beispiele am **BESTEN** als Eingangs- und Endekriterien ein?

Wählen Sie **EINE** Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Eingangskriterien: 5, 6; Endekriterien: 1, 2, 3, 4 | <input type="checkbox"/> |
| b) | Eingangskriterien: 2, 3, 4; Endekriterien: 1, 5, 6 | <input type="checkbox"/> |
| c) | Eingangskriterien: 1, 3; Endekriterien: 2, 4, 5, 6 | <input type="checkbox"/> |
| d) | Eingangskriterien: 3, 5, 6; Endekriterien: 1, 2, 4 | <input checked="" type="checkbox"/> |

FL-5.1.3 (K2) Der Lernende kann Eingangskriterien und Endekriterien vergleichen und gegenüberstellen.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

Die korrekte Zuordnung der Beispiele zu Eingangs- und Endekriterien ist:

- Eingangskriterien:
 - (3) Die Testumgebung für den Performanztest ist entworfen, eingerichtet und verifiziert – ein Beispiel dafür, dass eine Testumgebung bereit sein muss, bevor mit dem Testen begonnen werden kann (Verfügbarkeit von Ressourcen, siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.3, 2. Absatz).
 - (5) Die Designspezifikation für das System wurde einem Review unterzogen, ggf. nachgebessert und freigegeben – ein Beispiel dafür, dass die Testmittel zur Verfügung stehen müssen, bevor mit dem Testen begonnen werden kann (Verfügbarkeit von Testbasis / testbaren Anforderungen, siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.3, 2. Absatz).
 - (6) Die Komponenten für die Berechnung des Steuersatzes und des Gesamtpreises haben die Unittests bestanden und sind freigegeben – ein Beispiel für die Notwendigkeit, dass ein Testobjekt eine Anfangsqualität erfüllen muss, bevor mit dem Testen begonnen werden kann (Qualität des Testobjekts, z. B. Freigabe aus einer vorherigen Teststufe, siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.3, 2. Absatz).
- Endekriterien:
 - (1) Das Ausschöpfen eines geplanten Testbudgets kann ein Beispiel für eine Endekriterium sein, wenn die Stakeholder dies wünschen und das Risiko akzeptieren (siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.3, 4. Absatz).
 - (2) Das Erreichen von einer Grenze an durchgeführten Testfällen im Rahmen der geplanten Tests kann ein Testabschlusskriterium und somit ein Endekriterium für eine Teststufe sein (siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.3, 3. Absatz).
 - (4) Die Einhaltung einer bestimmten Anzahl bekannter, priorisierter und nicht behobener Fehlerzustände (offene Fehler) kann als Maß für die Vollständigkeit und somit als Endekriterium angesehen werden (siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.3, 3. Absatz). Dies sollte jedoch nie das alleinige Endekriterium sein, sondern im Zusammenhang mit der Testfallüberdeckung stehen (siehe Endekriterium (2) für durchgeführte Testfälle).

Somit ist Option d) KORREKT.

| | | | |
|----------|----------|----|------------|
| Frage 32 | FL-5.1.4 | K3 | Punkte 1.0 |
|----------|----------|----|------------|

Sie möchten den Testaufwand für ein neues Projekt mit Hilfe einer Drei-Punkt-Schätzung abschätzen. Sie haben die folgenden Schätzungen von den Experten erhalten: Die optimistischste Schätzung (a) ist 300 Personentage, die wahrscheinlichste Schätzung (m) ist 400 Personentage und die pessimistischste Schätzung (b) ist 500 Personentage.

Wie hoch schätzen Sie den Testaufwand auf Basis der Drei-Punkt-Schätzung für dieses Projekt ein?

Wählen Sie **EINE** Option! (1 aus 4)

| | | |
|----|-----------------------|-------------------------------------|
| a) | 350 Personentage | <input type="checkbox"/> |
| b) | 420 Personentage | <input type="checkbox"/> |
| c) | 400 ± 33 Personentage | <input checked="" type="checkbox"/> |
| d) | 450 Personentage | <input type="checkbox"/> |

FL-5.1.4 (K3) Der Lernende kann Schätzverfahren zur Berechnung des erforderlichen Testaufwands anwenden.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Diese Schätzung wäre das Ergebnis, wenn ausschließlich die optimistische (300 PT) und die wahrscheinlichste (400 PT) Schätzung berücksichtigt würden (z. B. als einfacher Durchschnitt: $(300 + 400) / 2 = 350$ PT). Die pessimistische Schätzung (500 PT) wurde hierbei nicht beachtet. Damit fehlen kritische Risikofaktoren, und das Ergebnis wäre wahrscheinlich zu optimistisch; zum Beispiel einfach den Durchschnitt: $(300 + 400) / 2 = 350$ PT.
- b) FALSCH – Die Zahl 420 Personentage entsteht durch eine gewichtete Drei-Punkt-Schätzung nach der PERT-Formel: (E) wird berechnet als $E = (a + 4 \cdot m + b) / 6 = (300 + 4 \cdot 400 + 500) / 6 = 400$. Die korrekte Berechnung ergibt jedoch eindeutig genau 400 Personentage, nicht 420. Option b enthält also einen Rechenfehler und ist daher falsch. Hier zeigt sich jedoch, dass die korrekte Berechnung exakt 400 Personentage ergibt, nicht 420. Daher wäre Option b rechnerisch fehlerhaft und somit falsch.
- c) KORREKT – Option c basiert korrekt auf der Drei-Punkt-Schätzung (PERT-Methode), welche optimistische ($a = 300$ PT), wahrscheinlichste ($m = 400$ PT) und pessimistische Schätzung ($b = 500$ PT) vollständig einbezieht: Erwarteter Wert (PERT-Formel): (E) wird berechnet als $E = (a + 4 \cdot m + b) / 6 = (300 + 4 \cdot 400 + 500) / 6 = 400$. Standardabweichung: $SD = (b - a) / 6 = (500 - 300) / 6 = 33$.
- d) FALSCH – Diese Option berücksichtigt lediglich die wahrscheinlichste (400 PT) und die pessimistischste Schätzung (500 PT) (z. B. einfacher Durchschnitt: $(400 + 500) / 2 = 450$ PT). Die optimistische Schätzung (300 PT) wird hier ignoriert, was dazu führt, dass die Schätzung zu hoch (zu pessimistisch) ist. Kritische Chancen oder Potenziale für geringeren Aufwand werden nicht einbezogen.

| | | | |
|----------|----------|----|------------|
| Frage 33 | FL-5.1.5 | K3 | Punkte 1.0 |
|----------|----------|----|------------|

Sie wurden gebeten, eine optimale, risikobasierte Ausführungsreihenfolge nachfolgender Testfälle festzulegen, die bereits priorisiert und auf etwaige Abhängigkeiten hin untersucht wurden:

| Testfall-ID | Priorität | Abhängig von |
|-------------|-----------|--------------|
| T7 | 2 | - |
| T8 | 1 | T7 |
| T9 | 3 | T8 |
| T10 | 3 | T8 |
| T11 | 1 | T9 |
| T12 | 2 | T10 |

Priorität 1 ist dringlicher als Priorität 2 usw.

Welche der folgenden Testabläufe berücksichtigt die oben genannten Abhängigkeiten und Prioritäten?

Wählen Sie **EINE** Option! (1 aus 4)

| | | |
|----|-------------------------------------|-------------------------------------|
| a) | T7 -> T8 -> T10 -> T11 -> T9 -> T12 | <input type="checkbox"/> |
| b) | T7 -> T8 -> T9 -> T10 -> T11 -> T12 | <input type="checkbox"/> |
| c) | T7 -> T8 -> T10 -> T9 -> T11 -> T12 | <input type="checkbox"/> |
| d) | T7 -> T8 -> T9 -> T11 -> T10 -> T12 | <input checked="" type="checkbox"/> |

FL-5.1.5 (K3) Der Lernende kann die Priorisierung von Testfällen anwenden.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Diese Reihenfolge berücksichtigt nicht die Abhängigkeit von T11, der von T9 abhängig ist und daher nach T9 ausgeführt werden sollte.
- b) FALSCH – Diese Reihenfolge berücksichtigt nicht die Priorität von T11, der vor T10 ausgeführt werden sollte (siehe auch Begründung zu Option d)).
- c) FALSCH – Diese Reihenfolge berücksichtigt nicht die Abhängigkeit und Priorität von T9 und T10, die von T8 abhängen. T8 aktiviert T9 und T10. Hier muss zum ersten Mal priorisiert werden: T9 und T10 haben beide Prio 3, aber T9 schaltet mit T11 einen Prio-1-Testfall frei, während T10 mit T12 nur einen Prio-2-Testfall freigibt. Somit wäre T9 eindeutig besser für die optimale Reihenfolge, aber hier ist T10 vor T9 aufgeführt, was nicht die optimale Reihenfolge darstellt (siehe auch Begründung zu Option d)).
- d) KORREKT – Diese Reihenfolge berücksichtigt sowohl die Prioritäten als auch die Abhängigkeiten korrekt.

Im Detail:

- Zu Beginn der Testausführung ist T7 alternativlos, da alle anderen Testfälle zunächst unerfüllte Abhängigkeiten haben.
- T7 schaltet nur T8 frei, der deshalb als zweiter Testfall folgen muss.
- T8 aktiviert T9 und T10. Hier muss zum ersten Mal priorisiert werden: T9 und T10 haben beide Prio 3, aber T9 schaltet mit T11 einen Prio-1-Testfall frei, während T10 mit T12 nur einen Prio-2-Testfall freigibt. Somit ist T9 eindeutig besser.
- Jetzt sind T10 und T11 frei. T11 gewinnt eindeutig nach Priorität.
- Danach ist T10 die einzige Option.
- T12 wird zum Schlusslicht.

| | | | |
|----------|----------|----|------------|
| Frage 34 | FL-5.1.7 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Welche der folgenden Aussagen veranschaulicht den Nutzen der Testquadranten?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Der Tester kann sich bei der Auswahl der Testarten auf den entsprechenden Quadranten beziehen, so dass alle beteiligten Stakeholder den Zweck der Tests besser verstehen. | <input checked="" type="checkbox"/> |
| b) | Der Tester kann die Testarten, die durch Testquadranten beschrieben werden, als Überdeckungsmetrik verwenden; je mehr Tests von jedem Quadranten ausgeführt werden, desto höher ist die Überdeckung. | <input type="checkbox"/> |
| c) | Das Team sollte für jeden Quadranten die ungefähr gleiche Anzahl von Testfällen einplanen, um sicherzustellen, dass alle Teststufen und Testarten gleichwertig berücksichtigt werden. | <input type="checkbox"/> |
| d) | Der Tester kann die Testquadranten zur Risikoanalyse einsetzen, wobei niedrigere Stufen der Quadranten ein geringeres Risiko für den Kunden repräsentieren. | <input type="checkbox"/> |

FL-5.1.7 (K2) Der Lernende kann die Testquadranten und ihre Beziehungen zu Teststufen und Testarten zusammenfassen.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) **KORREKT** – Die Testquadranten bieten eine Möglichkeit, die Testarten zu differenzieren und allen Stakeholdern, einschließlich Entwicklern, Testern und Fachbereichsvertretern, zu beschreiben und das Verständnis zu erhöhen (siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.7, 1. Absatz).
- b) FALSCH – Das Modell der Testquadranten unterstützt das Testmanagement dabei, die Teststufen und Testarten zu visualisieren, um sicherzustellen, dass alle geeigneten Testarten und Teststufen in den SDLC einbezogen werden, und um zu verstehen, dass einige Testarten für bestimmte Teststufen relevanter sind als für andere (siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.7, 1. Absatz). Das Modell bietet somit keine Art von Metrik an.
- c) FALSCH – Das Modell der Testquadranten unterstützt das Testmanagement dabei, die Teststufen und Testarten zu visualisieren, um sicherzustellen, dass alle geeigneten Testarten und Teststufen in den SDLC einbezogen werden, und um zu verstehen, dass einige Testarten für bestimmte Teststufen relevanter sind als für andere (siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.7, 1. Absatz). Welche Teststufen für das zu prüfende Testobjekt relevant sind und welche Testarten und damit welche Anzahl von Testfällen sich daraus ableiten, ist durch das Modell nicht vorgegeben. Gerade die Anzahl der Testfälle eines jeden Quadranten hängt vom jeweiligen System ab und wird eigentlich nie für alle Quadranten gleich sein, z. B. wird der Quadrant Q1 mit Komponenten- und Komponentenintegrationstest in der Regel deutlich mehr Testfälle umfassen als der Quadrant Q3 mit Abnahmetests und Gebrauchstauglichkeitstests.
- d) FALSCH – Testquadranten haben keinen Bezug zu Risikostufen, denn das Modell der Testquadranten unterstützt das Testmanagement nur dabei, die Teststufen und Testarten zu visualisieren, um sicherzustellen, dass alle geeigneten Testarten und Teststufen in den SDLC einbezogen werden, und um zu verstehen, dass einige Testarten für bestimmte Teststufen relevanter sind als für andere (siehe CTFL-Lehrplan V4.0, Abschnitt 5.1.7).

| | | | |
|----------|----------|----|------------|
| Frage 35 | FL-5.2.4 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Welche der folgenden Aussagen über die Steuerung von Produktrisiken im Rahmen der Risikominderung trifft NICHT zu?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | Die Komplexität des Datenbankmoduls wurde als hoch eingestuft, daher wurden zusätzliche Testfälle für das Modul erstellt. | <input type="checkbox"/> |
| b) | Die Anforderungen an das Benutzerinterface sind unklar, daher wird ein Experte für Benutzererfahrung in das Projekt einbezogen. | <input type="checkbox"/> |
| c) | Die Performanz des Systems ist entscheidend für den Erfolg des Projekts, daher wird auf Code-Reviews verzichtet, um Zeit zu sparen. | <input checked="" type="checkbox"/> |
| d) | Das System muss eine hohe Verfügbarkeit aufweisen, daher werden zusätzliche Lasttests durchgeführt. | <input type="checkbox"/> |

FL-5.2.4 (K2) Der Lernende kann mögliche Maßnahmen, die als Reaktion auf analysierte Produktrisiken ergriffen werden können, erklären.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Dies ist eine geeignete Maßnahme zur Risikominderung, da zusätzliche Testfälle helfen können, die Qualität des Moduls zu gewährleisten und mögliche Fehler frühzeitig zu erkennen.
- b) FALSCH – Dies ist eine geeignete Maßnahme zur Risikominderung, da ein Experte für Benutzererfahrung dazu beitragen kann, die Anforderungen zu klären und eine benutzerfreundliche Gestaltung des Interface zu gewährleisten.
- c) **KORREKT – Dies trifft nicht zu, da Code-Reviews eine wichtige Maßnahme zur Risikominderung sind und nicht übergangen werden sollten, um Zeit zu sparen. Sie helfen dabei, mögliche Fehler und Probleme in der Codebasis frühzeitig zu erkennen und zu beheben, was letztendlich zur Verbesserung der Systemperformanz beitragen kann.**
- d) FALSCH – Dies ist eine geeignete Maßnahme zur Risikominderung, da Lasttests dazu beitragen können, die Leistung und Verfügbarkeit des Systems unter hoher Last zu überprüfen und sicherzustellen.

| | | | |
|----------|----------|----|------------|
| Frage 36 | FL-5.3.3 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

In einem regulatorischen Projekt, das bereits in Verzug ist, haben die relevanten Stakeholder darum gebeten, täglich über den Teststatus informiert zu werden.

Welche ist die **EFFEKTIVERE** Möglichkeit, den Teststatus zu kommunizieren, wenn die Stakeholder sich aufgrund räumlicher und zeitlicher Gegebenheiten nicht direkt abstimmen können?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | Es sollten formelle Kommunikationsmittel (z.B. formale Berichte, E-Mails) verwendet werden, um sicherzustellen, dass wichtige Informationen die Empfänger erreichen. | <input checked="" type="checkbox"/> |
| b) | Die Kommunikation sollte über eine Chatgruppe erfolgen, damit alle Teammitglieder möglichst zeitnah über den Teststatus informiert werden können. | <input type="checkbox"/> |
| c) | Die relevanten Stakeholder sollten mündlich über den Teststatus informiert werden, um die wichtigsten Informationen direkt zu übermitteln. | <input type="checkbox"/> |
| d) | Der Teststatus sollte in täglichen Abstimmungsrunden per Videokonferenz kommuniziert werden, an denen Stakeholder aus allen beteiligten Zeitzonen teilnehmen sollten. | <input type="checkbox"/> |

FL-5.3.3 (K2) Der Lernende kann Beispiele geben, wie man den Teststatus kommunizieren kann.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) **KORREKT** – Für zeitlich und räumlich verteilte Teams ist eine **formellere** Art der **Kommunikation** vorzuziehen (siehe CTFL-Lehrplan V4.0, Abschnitt 5.3.3, 2. Absatz 2. Satz: „Eine formellere Kommunikation kann sich für verteilte Teams anbieten, in denen eine direkte Kommunikation von Angesicht zu Angesicht aufgrund von geografischen Entfernungen oder Zeitunterschieden nicht immer möglich ist.“). **Zusammengefasst bedeutet formellere Kommunikation in diesem Kontext, dass die Informationen strukturiert, dokumentiert und nachvollziehbar übermittelt werden, was in verteilten Teams besonders wichtig ist.**
- b) **FALSCH** – Chatgruppen sind eine sehr gute Möglichkeit für einen schnellen Austausch, aber nicht unbedingt für zeitlich verteilte Gruppen, da es schwierig sein kann, relevante Informationen im Chatverlauf zu finden. Außerdem ist das Argument falsch. In zeitlich verteilten Teams ist es nicht relevant, dass Informationen zeitnah an den Empfängerkreis übermittelt werden.
- c) **FALSCH** – Die mündliche Kommunikation ist bei zeitlich verteilten Teams nicht immer möglich und daher nicht die effizienteste Art der Kommunikation. In der Frage wurde klar angegeben, dass sich die Teammitglieder aufgrund der zeitlichen Verteilung nicht direkt abstimmen können.
- d) **FALSCH** – Da die Stakeholder bereits täglich über den Teststatus informiert werden möchten, ist es keine Option, sie durch ein dafür ungeeignetes Werkzeug wie eine Videokonferenz zu unterstützen.

| | | | |
|----------|----------|----|------------|
| Frage 37 | FL-5.4.1 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Welche der folgenden Aussagen beschreibt, wie das Konfigurationsmanagement die Testaktivitäten unterstützen kann?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | Eine Testerin hinterlegt den Fortschritt, den sie bei der Testdurchführung am aktuellen Tag gemacht hat, in einem Testmanagementwerkzeug. | <input type="checkbox"/> |
| b) | Ein Tester speichert Testdaten für die datengetriebene Testausführung in einer Datenbank und stellt sicher, dass die Daten zur Testausführungszeit aus der Datenbank gelesen werden können. | <input type="checkbox"/> |
| c) | Ein Tester verwendet ein Tabellenkalkulationsprogramm, um die Geschäftsregeln eines zu testenden Systems in Form von Entscheidungstabellen zu formalisieren. | <input type="checkbox"/> |
| d) | Ein Tester oder eine Testerin stellt automatisiert die relevanten Testmittel für eine ältere Version eines Produkts wieder her, um den Wartungstest für diese ältere Version durchführen zu können. | <input checked="" type="checkbox"/> |

FL-5.4.1 (K2) Der Lernende kann eine mögliche Unterstützung des Testens durch das Konfigurationsmanagement zusammenfassen.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Die Verwendung von Testmanagementwerkzeugen als Informations- und Fortschrittsrepositorien ist Teil der Aktivität Testüberwachung und Teststeuerung und nicht Teil des Konfigurationsmanagements.
- b) FALSCH – Diese Aktivitäten haben in erster Linie nichts mit dem Konfigurationsmanagement zu tun, sondern sind Teil der Testrealisierung.
- c) FALSCH – Die Formalisierung von logischen Geschäftsregeln in Form von Entscheidungstabellen ist Teil der Testanalyse, bei der die Testbedingungen formal dokumentiert werden, und hat primär nichts mit dem Konfigurationsmanagement von Testmitteln zu tun.
- d) **KORREKT – Ein wesentlicher Teil des Konfigurationsmanagements besteht darin, bei Bedarf zu einer früheren Baseline eines Konfigurationselements zurückzukehren (siehe CTFL-Lehrplan V4.0, Abschnitt 5.4, 3. Absatz). Dies entspricht genau der in Antwort d) beschriebenen Situation.**

| | | | |
|----------|----------|----|------------|
| Frage 38 | FL-5.5.1 | K3 | Punkte 1.0 |
|----------|----------|----|------------|

Sie testen eine neue Version der Software für eine Kaffeemaschine. Mit dieser Software kann die Maschine verschiedene Kaffeevarianten zubereiten, die in vier Kategorien eingeteilt sind: Kaffeemenge, Zucker, Milch und Kaffeegeschmack.

Die Kategorien sind wie folgt:

- Kaffeemenge (klein, mittel, groß)
- Zucker (keiner, 1 Einheit, 2 Einheiten, 3 Einheiten, 4 Einheiten)
- Milch (ja oder nein)
- Kaffeegeschmack (kein Sirup, Karamell, Haselnuss, Vanille)

Sie schreiben einen Fehlerbericht mit den folgenden Informationen:

Titel: Niedrige Kaffeetemperatur.

Kurze Zusammenfassung: Bei der Auswahl von Kaffee mit Milch, ist die Temperatur des Getränks zu niedrig (unter 40 °C).

Erwartetes Ergebnis: Die Temperatur des Kaffees sollte dem Standard entsprechen (ca. 75 °C).

Grad der Auswirkung: Mittel

Priorität: Normal

Welche relevanten Informationen haben Sie im obigen Fehlerbericht vergessen?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|--|-------------------------------------|
| a) | Tatsächliches Testergebnis. | <input type="checkbox"/> |
| b) | Identifikation der getesteten Softwareversion. | <input checked="" type="checkbox"/> |
| c) | Ideen zur Verbesserung des Testfalls. | <input type="checkbox"/> |
| d) | Qualität des Arbeitsergebnisses, das getestet wurde. | <input type="checkbox"/> |

FL-5.5.1 (K3) Der Lernende kann einen Fehlerbericht erstellen.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Das erwartete Ergebnis und das tatsächliche Ergebnis sind normalerweise wertvolle Informationen im Fehlerbericht (siehe CTFL-Lehrplan V4.0, Abschnitt 5.5, 3. Absatz und 7. Aufzählungspunkt). Das erwartete Testergebnis wird angegeben (ca. 75 °C) und das tatsächliche Ergebnis wird ebenfalls angegeben („Temperatur des Getränks zu niedrig (weniger als 40 °C“).
- b) KORREKT – Die Identifikation des Testobjekts und der Testumgebung sind in der Regel wertvolle Informationen im Fehlerbericht (siehe CTFL-Lehrplan V4.0, Abschnitt 5.5, 3. Absatz und 4. Aufzählungspunkt), daher ist die Angabe der eingesetzten Softwareversion ein wichtiger und wertvoller Teil des Fehlerberichts.
- c) FALSCH – Diese Information kann für den Tester nützlich sein, ist aber normalerweise nicht Teil eines Fehlerberichts (siehe CTFL-Lehrplan V4.0, Abschnitt 5.5, 3. Absatz).
- d) FALSCH – Die Qualität des getesteten Arbeitsergebnisses ist eine wichtige Information für das Projekt, aber nicht für einen Fehlerbericht. Vielmehr ergibt sich die geschätzte Qualität des Arbeitsergebnisses unter anderem aus der Anzahl der für das Arbeitsergebnis erstellten Fehlerberichte.

| | | | |
|----------|----------|----|------------|
| Frage 39 | FL-6.1.1 | K2 | Punkte 1.0 |
|----------|----------|----|------------|

Gegeben seien folgende Testaktivitäten:

- 1. Performanzmessung und IT-Sicherheitsprüfungen**
- 2. Testautomatisierung**
- 3. Management der Testaktivitäten**
- 4. Testentwurf und Testrealisierung**

und Testwerkzeuge:

- A. Werkzeuge zur Testdurchführung**
- B. Testwerkzeuge für nicht-funktionale Tests**
- C. Werkzeuge zur Vorbereitung von Testfällen und Testdaten**
- D. Fehlermanagementwerkzeuge**

Welche Zuordnung der Werkzeuge zu den Aktivitäten trifft am BESTEN zu?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|----------------------------|-------------------------------------|
| a) | 1 – D, 2 – C, 3 – B, 4 – A | <input type="checkbox"/> |
| b) | 1 – B, 2 – A, 3 – C, 4 – D | <input type="checkbox"/> |
| c) | 1 – B, 2 – A, 3 – D, 4 – C | <input checked="" type="checkbox"/> |
| d) | 1 – A, 2 – B, 3 – D, 4 – C | <input type="checkbox"/> |

FL-6.1.1 (K2) Der Lernende kann eine mögliche Unterstützung des Testens durch verschiedene Arten von Testwerkzeugen erklären.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

Die korrekte Zuordnung von Testaktivitäten zu Arten von Testwerkzeugen ist wie folgt (siehe CTFL-Lehrplan V4.0, Abschnitt 6.1):

- 1. Performanzmessung und IT-Sicherheitsprüfungen sind nicht-funktionale Tests, daher werden sie durch Testwerkzeuge für nicht-funktionale Tests (B) unterstützt (siehe CTFL-Lehrplan V4.0, Abschnitt 6.1, 5. Aufzählungspunkt).
- 2. Testautomatisierung gehört zur automatisierten Testdurchführung, daher wird sie durch Werkzeuge zur Testdurchführung (A) unterstützt (siehe CTFL-Lehrplan V4.0, Abschnitt 6.1, 4. Aufzählungspunkt).
- 3. Das Management der Testaktivitäten beinhaltet die Verwaltung von Fehlerzuständen, daher wird es durch ein Fehlermanagementwerkzeug (D) unterstützt (siehe CTFL-Lehrplan V4.0, Abschnitt 6.1, 1. Aufzählungspunkt).
- 4. Zum Testentwurf und zur Testrealisierung gehört die Vorbereitung von Testdaten, daher werden sie durch Werkzeuge zur Vorbereitung von Testfällen und Testdaten (C) unterstützt (siehe CTFL-Lehrplan V4.0, Abschnitt 6.1, 3. Aufzählungspunkt).

Somit ist Option c) [1 – B, 2 – A, 3 – D, 4 – C] KORREKT.

| | | | |
|----------|----------|----|------------|
| Frage 40 | FL-6.2.1 | K1 | Punkte 1.0 |
|----------|----------|----|------------|

Welche der folgenden Aussagen beschreibt am BESTEN den potenziellen Nutzen des Einsatzes von Werkzeugen für die automatisierte Testdurchführung?

Wählen Sie EINE Option! (1 aus 4)

| | | |
|----|---|-------------------------------------|
| a) | Die Implementierung von Regressionstests ist einfacher, da diese direkt mit einem Testskript implementiert werden können. | <input type="checkbox"/> |
| b) | Es findet eine effizientere Bewertung des Testobjekts durch das Automatisierungswerkzeug statt. | <input type="checkbox"/> |
| c) | Ein Testwerkzeug wird verwendet, wenn manuelles Testen besser geeignet ist. | <input type="checkbox"/> |
| d) | Regressionstests können schneller durchgeführt werden und ermöglichen somit eine schnellere Rückmeldung an das Team. | <input checked="" type="checkbox"/> |

FL-6.2.1 (K1) Der Lernende kann Nutzen und Risiken von Testautomatisierung wiedergeben.

Begründung (siehe ISTQB®-Lehrplan CTFL V4.0):

- a) FALSCH – Testausführungswerkzeuge, die die Testautomatisierung unterstützen, erleichtern nicht die Erstellung und Implementierung von Regressionstests, sondern deren Durchführung (siehe CTFL-Lehrplan V4.0, Abschnitt 6.2, 1. Aufzählungspunkt).
- b) FALSCH – Durch eine automatisierte Testdurchführung wird eine objektive Bewertung des Testobjekts ermöglicht. Objektiv ist aber nicht dasselbe wie eine effizientere Bewertung (siehe CTFL-Lehrplan V4.0, Abschnitt 6.2, 2. Absatz, 3. Aufzählungspunkt).
- c) FALSCH – Dies beschreibt keinen Nutzen der Testautomatisierung, sondern ein Risiko (siehe CTFL-Lehrplan V4.0, Abschnitt 6.2, 3. Absatz, 3. Aufzählungspunkt).
- d) **KORREKT** – Zu den **potenziellen Vorteilen der Testautomatisierung** und des **Einsatzes von Testausführungswerkzeugen** gehört die **Zeitersparnis durch die Reduzierung sich wiederholender manueller Arbeiten (z. B. Ausführung von Regressionstests)** (siehe CTFL-Lehrplan V4.0, Abschnitt 6.2, 2. Absatz, 1. und 5. Aufzählungspunkt).

Platz für Ihre Notizen:

(Sie werden bei der Korrektur weder gelesen noch bewertet)

Platz für Ihre Notizen:

(Sie werden bei der Korrektur weder gelesen noch bewertet)