**NOTES FOR TRANSLATOR**

**Please write the translation directly at the end of each section (see note after section 1.1)**

**Please keep the italics and underlining the same (they indicate the correct answer(s))**

**Please keep code the same and do not translate**

**Do not worry about other formatting. (Each question and its separate answers are copied and pasted individually by us into another program. This is the reason these questions in this document are not identically formatted since they didn’t come from the same standard document.)**

**Unit 1.1**

1. Grenzen Sie die Begriffe Spezifikation und Requirements Engineering voneinander ab.

*Ziel von Aktivitäten der Spezifikation ist die Erstellung einer technischen Dokumentation der nach außen relevanten Anforderungen, nach denen ein Softwaresystem produziert werden soll. Aufbauend auf den Erkenntnissen des fachlichen Requirements Engineerings, bei der fachliche Anforderungen ermittelt, dokumentiert, geprüft und abgestimmt wurden, wird im Rahmen der Spezifikation eine stark technisch ausgeprägte Dokumentation des zu erstellenden Systems erstellt.*

*Dabei handelt es sich bezogen auf die Aktivitäten des Requirements Engineerings um eine Erweiterung und Detaillierung der Dokumentation von Anforderungen. Hinsichtlich Ermittlungstechniken oder Prüftechniken unterscheiden sich RE und Spezifikation nicht.*

1 Explain the difference between software requirements specification (SRS) and requirements engineering (RE).

*SRS activities are aimed at the technical documentation of external requirements placed on a software system. Building on the results of requirements engineering, which identifies, documents, tests, and coordinates the conceptual requirements, SRS prepares highly technical documentation for the system under development.*

*Compared with RE, SRS is a broader and more detailed approach to the documentation of requirements. The investigation and test methods used in RE and SRS are identical.*

1. Nennen Sie die Aktivitäten eines Softwareprozesses, in denen Spezifikationsdokumente verwendet werden.

*Die Spezifikation ist die Grundlage für die Umsetzung des Systems in Aktivitäten zum Design und der Implementierung als auch für die Formulierung von Testfällen in allen Teststufen sowie die Durchführung und Auswertung von Tests.*

2 List the activities in a software process that uses SRS documents.

*Specification provides the basis for translating the system into design and implementation activities, as well as for the formulation of test cases at all testing levels and the execution and evaluation of testing.*

1. Grenzen Sie die Begriffe Spezifikation und Design voneinander ab.

*Die Spezifikation eines Systems gibt aus fachlicher Sicht den technisch detaillierten Rahmen für Designentscheidungen vor. Dabei trifft eine Spezifikation keine Entscheidung darüber, wie das System intern konstruiert werden muss, sondern beschreibt stets nur die nach außen sichtbaren Systemeigenschaften. Erst beim Design werden konkrete Elemente des Systems gestaltet. Dabei muss sich an die Vorgaben der Spezifikation gehalten werden.*

3 Explain the difference between specification and design.

*From a conceptual perspective, specification of a system provides a detailed technical framework for design decisions. Specification does not decide on the internal design of the system, and is confined to a description of those system characteristics which are externally visible. Specific elements of the system are not defined until the design phase, but the guidelines set out in the specification must be observed.*

**Unit 1.2**

1. Nennen und beschreiben Sie die zur Spezifikation von GUIs relevanten Aspekte.
* *Inhalte und Aufbau von einzelnen Dialogmasken:*

*Detaillierte Vorgabe zu Art, Größe, Position, Farbe und Inhalt von Elementen einer Bildschirmseite, z. B. von Eingabefeldern, Texten, Schaltflächen, Bildern*

* *Validierung von Daten:*

*Spezifikation der Regeln, um Eingabefelder auf fachliche Plausibilität zu prüfen*

* *Dialogfluss:*

*Spezifikation der Führung des Anwenders durch die Oberfläche in Abhängigkeit von eingegebenen Daten und Aktionen des Anwenders*

1. Describe which aspects are relevant for the specification of GUIs.
* *Content and structure of individual dialog boxes:*

*Detailed specification of the nature, size, position, color, and content of elements on a screen page, such as input boxes, texts, buttons, and images*

* *Validation of data:*

*Specification of rules used to verify the conceptual plausibility of input boxes*

* *Dialog flow:*

*Specification of user guidance through the interface depending on the user’s actions and data entries*

1. Nennen und beschreiben Sie die zur Spezifikation von technischen Systemschnittstellen relevanten Aspekte.
* *Der Zweck der Schnittstelle auf einer fachlichen Ebene. Beispiel: Übertragung der Fondskurse in die Fondsverwaltung, Validierung von Adressdaten auf Gültigkeit.*
* *Das detaillierte Verhalten bzw. das technische Protokoll der Kommunikation sowie die Regeln, nach denen das System mit seinem Umfeld kommuniziert. Beispiel: HTTP, FTP.*
* *Die Datenstruktur der Nachrichten, die an der Schnittstelle ausgetauscht werden. Beispiel: XML, CSV*
1. Describe which aspects are relevant for the specification of technical system interfaces.
* *The interface’s intended purpose at a conceptual level. For example: transferring share prices into the fund management system or validating address data.*
* *The detailed behavior and technical protocol of communication and the rules used by the system to communicate with its environment. For example: HTTP, FTP.*
* *The data structure of messages exchanged at the interface. Examples: XML, CSV.*

**Unit 1.3**

1. Erläutern Sie den Zusammenhang zwischen Klassendiagrammen und Objektdiagrammen und nennen Sie je eine Situation, in der Sie diese Diagrammtypen einsetzen würden.

*Das UML-Objektdiagramm wird zur Darstellung eines ganz bestimmten Datensatzes auf Basis der in einem Klassendiagramm vorgegebenen Struktur erstellt. Vergleichbar mit dem Sequenzdiagramm können mit einem Objektdiagramm Sachverhalte exemplarisch dargestellt und veranschaulicht werden.*

*Einsatz: z. B. zur Veranschaulichung konkreter Daten aus einem Klassendiagramm zur Abstimmung mit Fachexperten ODER zur Überprüfung der Eignung des Klassendiagramms anhand von konkreten Datensätzen.*

1. Explain the correlation between class diagrams and object diagrams and name a situation in which you would use each of these diagram types.

*The UML object diagram is used to denote a particular data record based on the structure prescribed in a class diagram. Like a sequence diagram, an object diagram is suitable for modeling and illustrating facts by using examples.*

*It may be used, for example, to illustrate actual data from a class diagram for coordinating with subject experts OR to review the suitability of the class diagram using actual data records.*

1. Nennen Sie drei weitere für eine Spezifikation typische Dokumentationsformen, die es neben Text und UML-Diagrammen noch gibt.
* GUI-Prototypen zur Spezifikation von Benutzerschnittstellen,
* XML-Sprachen zur Spezifikation von Datenstrukturen an technischen Systemschnittstellen,
* Entscheidungstabellen zur Spezifikation von Systemverhalten sowie Geschäftsregeln, die je nach Situation zur Spezifikation von Datenstrukturen oder Systemverhalten eingesetzt werden können.
1. Name three further typical documentation formats used in specifications, other than text and UML diagrams.
* GUI prototypes for the specification of user interfaces,
* XML languages for the specification of data structures at technical system interfaces,
* Decision tables for the specification of system behavior and business rules which may be used to specify data structures or system behavior, depending on the situation.

**Unit 2.1**

1. Erläutern Sie kurz, welchen Zweck GUIs für industrielle Informationssysteme erfüllen.

*Industrielle Informationssysteme stellen in der Regel eine GUI, d. h. grafische Benutzeroberfläche, zur Verfügung. Das bedeutet die Information wird visuell dargestellt und vom Nutzer mit den Augen wahrgenommen. Die GUIs industrieller Informationssysteme werden genutzt, um durch den fachlichen Prozess zu führen bzw. zu navigieren. Dabei werden den Nutzern Eigenschaften von Fachobjekten angezeigt und Fachobjekte von Nutzern bearbeitet. Außerdem werden von Nutzern Entscheidungen in das System eingegeben, die zur Ablaufsteuerung durch den Geschäftsprozess erforderlich sind.*

1. Briefly explain the purpose of GUIs for industrial information systems.

*Industrial information systems generally provide a GUI (graphical user interface) which presents information in a visual format so that it can be seen by the user. In industrial information systems, GUIs help to guide users as they navigate through the process. Users can see and process the characteristics of conceptual objects. They also enter decisions which are required for operational control through the business process.*

1. Nennen und beschreiben Sie kurz zwei Elemente von GUI-Spezifikationen.
* *Bildschirmdialog*

*Seitenabfolge (auch: Dialog), die den Nutzer durch das System führt und ihn bei der Bearbeitung seiner Aufgaben unterstützt. Ein Bildschirmdialog besteht aus einer oder mehreren Dialogmasken sowie einem festgelegten Dialogfluss.*

* *Dialogmaske*

*Menge angemessener Eingabe- und Ausgabeelemente, die zu einem Interaktionsschritt bezogen auf eine vom System unterstützte Fachfunktion benötigt werden.*

* *Konvertierung*

*Umwandlung von Datensätzen der Datenbank in ein lesbares Format und zurück.*

* *Validierung*

*Regel zur Prüfung von Nutzereingaben, die sicherstellen, dass die eingegebenen Daten im System korrekt gespeichert werden.*

* *Dialogfluss*

*Die Navigationsmöglichkeiten zwischen Dialogmasken. Ein Dialogfluss wird verwendet, um in einem Informationssystem einen Arbeitsablauf abzubilden, dessen Abarbeitung mehrere Masken nacheinander erfordert.*

1. Briefly describe two elements of GUI specifications.
* *Screen dialog*

*A sequence of pages (dialog) which guides users through the system and assists with task processing. A screen dialog is comprised of one or more dialog boxes and a predefined dialog flow.*

* *Dialog box*

*A set of appropriate input and output elements needed for an interaction step in relation to a function supported by the system.*

* *Conversion*

*Conversion of data records from the database into a readable format and vice versa.*

* *Validation*

*Rule for checking user inputs to ensure that the entered data is correctly stored in the system.*

* *Dialog flow*

*The navigation options between dialog boxes. A dialog flow is used to model a work operation in an information system whereby multiple boxes are processed in sequence.*

1. Nennen Sie drei Dokumentationsformen für GUI-Spezifikationen.
* *Beispiele für visuelle Dokumentationsformen:*
	+ *Skizzen,*
	+ *Screenshots von GUI-Prototypen,*
	+ *Diagramme mit der Reihenfolge der Dialogmasken*
* *Beispiele für Dokumentationsformen mit Text:*
	+ *Regeln zur Aktivierung und Deaktivierung von GUI‐Elementen,*
	+ *Validierungsregeln,*
	+ *Dialogflussbedingungen.*
1. Name 3 types of documentation for GUI specifications.
* *Examples of visual documentation forms:*
	+ *Sketches,*
	+ *Screenshots of GUI prototypes,*
	+ *Diagrams showing the sequence of dialog boxes*
* *Examples of text-based documentation formats:*
	+ *Rules for the activation and deactivation of GUI elements*
	+ *Validation rules*
	+ *Dialog flow conditions.*

**Unit 2.2**

1. Nennen Sie je ein GUI-Element und einen Anwendungsfall für atomare GUI-Elemente, Komposit-Elemente und komplexe GUI-Elemente.
* *Atomares GUI-Element: z. B. Textfeld, Eingabe einer Email-Adresse;*
* *Komposit-Element: z. B. Optionsfelder, Auswahl von 1 aus N vorgegebenen Optionen, z. B. Wahl ob Hin- und Rückfahrt oder nur Hinfahrt bei der Ticketbuchung;*
* *Komplexes GUI-Element: Datumseingabe per grafischen Kalender, Eingabe des Reisedatums bei einer Ticketbuchung.*
1. Name one GUI element and one application case for atomic GUI elements, composite elements, and complex GUI elements.
* *Atomic GUI element: e.g. text box, entry of an email address;*
* *Composite element: e.g. option boxes, selection of 1 of N predefined options, e.g. choice of return or one way when booking a ticket;*
* *Complex GUI element: Entering the date on a graphical calendar, entering the date of travel when booking a ticket.*
1. Beschreiben Sie kurz anhand eines selbstgewählten Anwendungsbeispiels (nicht das KZF-Kennzeichen), wozu eine Konvertierung benötigt wird.

*Beispiel: Eingabe einer IP-Adresse der Form XXX.XXX.XXX.XXX, wobei XXX jeweils einer Zahl von 0–255 entspricht. In der Datenbank wird nur ein String gespeichert, die Eingabe soll jedoch in 4 Textfeldern erfolgen, die nebeneinander angezeigt werden.*

1. Using an application example you have thought up yourself (not the car number plate), briefly explain why conversion is necessary.

*Example: Enter an IP address in the format XXX.XXX.XXX.XXX, whereby each XXX corresponds to a number between 0 and 255. Only a string is saved in the database, but the data is entered in 4 adjacent text boxes.*

**Unit 2.3**

1. Nennen Sie vier verschiedene Transaction Levels von GUI-Validierungen und geben Sie zu jedem Transaction Level ein selbstgewähltes Beispiel für eine Validierung an.
* *Verlassen des GUI-Elements: Prüfen des richtigen Datentyps bei der Eingabe einer ganzen Zahl;*
* *Verlassen der Bildschirmseite: Prüfung, ob alle Elemente einer Adresse eingeben wurden;*
* *Zwischenspeichern der Daten, z. B. um die Bearbeitung zu einem späteren Zeitpunkt fortzusetzen: Prüfen, ob alle bereits eingegebenen Daten im richtigen Format eingegeben wurden;*
* *Senden der Daten an das System und Abschließen des Eingabevorgangs: Prüfen, ob das Ende einer Vertragslaufzeit ein Zeitpunkt nach Beginn der Vertragslaufzeit ist.*
1. Name four different transaction levels of GUI validations and give an example of validation for each level that you have thought of yourself.
* *Exit the GUI element: Check for the correct data type when entering a whole number;*
* *Exit the screen page: Check whether all elements of an address have been entered;*
* *Data cache storage e.g. in order to resume processing at a later date: Check whether all data already entered is in the correct format;*
* *Send data to the system and finalize the input process: Check that a contract‘s end is after its start date.*
1. Beschreiben Sie kurz, auf welche Weise dem Nutzer eine fehlgeschlagene Validierungsregel mitgeteilt werden kann.
	* *Anzeigen einer Fehlernachricht;*
	* *Umranden der betroffenen Eingabefelder;*
	* *Einfärben der Beschriftung der Eingabeelemente;*
	* *Einschränken von Navigationsmöglichkeiten.*
2. Briefly describe how to tell the user about a failed validation rule.
	* *Show an error message;*
	* *Draw a frame around the affected input boxes;*
	* *Change the font color for the input elements;*
	* *Limit navigation options.*

**Unit 2.4**

1. Nennen Sie die Elemente, die bei der Spezifikation von Dialogflüssen berücksichtigt werden müssen.
	* + *Jede im Arbeitsablauf verwendete Maske;*
		+ *Die Festlegung der Reihenfolge in der die Masken durchlaufen werden;*
		+ *Manuelle Navigationsmöglichkeiten zwischen den Masken (zurück, weiter); sowie*
		+ *Dialogflussbedingungen zur Steuerung des Durchlaufs (Überspringen von Masken).*
2. List the elements which must be taken into account when specifying dialog flows.
	* + *Each box used in the workflow;*
		+ *Specification of the order in which boxes must be completed;*
		+ *Manual navigation options between boxes (Back, Next); and*
		+ *Dialog flow conditions to control the cycle (skipping boxes).*
3. Wie können solche Navigationsmöglichkeiten übersichtlich dokumentiert werden, die es von bzw. zu allen Bildschirmmasken gleichermaßen gibt?

*Häufige oder für alle Dialogmasken mögliche Navigationen können exemplarisch für eine Dialogmaske modelliert und dann im beschreibenden Text für alle weiteren Masken verallgemeinert werden.*

2 How can you clearly document navigation options which apply from and to all screen masks?

*Common navigations and those supported by all dialog boxes can be modeled for one dialog box as an example and then generalized for all subsequent masks in the descriptive text.*

**Unit 3.1**

1. Beschreiben Sie kurz, was eine Komponente ist. Gehen Sie dabei auf den Zusammenhang zwischen System und Komponente ein.

*Komplexe Softwaresysteme werden in Komponenten aufgeteilt. Eine Komponente ist eine unabhängige Softwareeinheit, die aufgrund vereinbarter Schnittstellen mit anderen Komponenten zu einem Softwaresystem zusammengestellt werden kann.*

1 Briefly describe what is meant by a component. Examine the relationship between systems and components.

*Complex software systems are divided into components. A component is an independent software unit that can be combined with other components via agreed interfaces to create a software system.*

1. Nennen Sie drei Qualitätsziele, die durch den Einsatz von Komponenten erreicht werden sollen.
* *Einfache Erweiterbarkeit,*
* *bessere Testbarkeit sowie*
* *bessere Wartbarkeit.*

2. Name three quality targets that can be met using components.

* *Easily extendible,*
* *Better testability and*
* *Better maintainability.*
1. Nennen und beschreiben Sie die Elemente, die zu einer Komponente spezifiziert werden müssen.
* *Internes Verhalten, beispielsweise Laufzeitverlängerung des Vertrags, Berechnung der Gesamtsumme;*
* *Abläufe und logische Struktur der ausgetauschten Nachrichten an Schnittstellen von Komponenten.*
1. Describe which elements must be specified for a component.
* *Internal behavior, such as extending the term of the contract, calculating the overall total;*
* *Operations and logical structure of exchanged interfaces at component interfaces.*

**Unit 3.2**

1. Welcher Diagrammkategorie ist das Komponentendiagramm zugeordnet und was sind dessen Hauptelemente?

*Das UML-Komponentendiagramm (engl.: component diagram, Abkürzung: cod) ist ein Strukturdiagramm. Die Hauptelemente sind Komponenten, Schnittstellen sowie Beziehungen von Komponenten zu Schnittstellen.*

1. Which diagram category do component diagrams belong to, and what are their principal elements?

*The UML component diagram (cod) is a structural diagram. Its principal elements are components, interfaces, and relationships between components and interfaces.*

1. Wofür werden UML-Komponentendiagramme schwerpunktmäßig eingesetzt?

*Der Schwerpunkt der Modellierung liegt auf dem inneren Aufbau und den Elementen eines Systems. Die modellierten Komponenten können fachliche Komponenten sein. Deren Beziehungen stellen dann fachliche Abhängigkeiten dar, die grundsätzlich auch ohne die Unterstützung von IT-Systemen bestehen. Darüber hinaus kann das Komponentendiagramm auch zur Spezifikation von technischen Komponenten eingesetzt werden. So können beispielsweise Softwarearchitekten die konkreten Komponenten eines Systems sowie deren Schnittstellen festlegen.*

2 What are the main application areas for UML component diagrams?

*Modeling focuses on a system’s internal structure and elements. The modeled components may be conceptual components. The relationships between them represent conceptual dependencies which generally exist even without the support of IT systems. Component diagrams can also be used in the specification of technical components. For example, software architects can specify particular components of a system and their interfaces.*

**Unit 3.3**

1. Beschreiben Sie den Einsatz von Partitionen im UML-Aktivitätsdiagramm.

*Eine Partition im Aktivitätsdiagramm grenzt den Einflussbereich der an dem Ablauf dargestellten Akteuren ab. Als Akteur können sowohl menschliche Nutzer als auch technische Systeme eingesetzt werden. Partitionen dienen der Spezifikation und der Darstellung der Verantwortlichkeit für fachliche Funktionen und Entscheidungen. Aus diesen Verantwortlichkeiten leiten sich die konkreten Anforderungen an Schnittstellen und benötigten Geschäftsobjekten ab.*

1. Describe the use of partitions in UML activity diagrams.

*In an activity diagram, a partition delimits the actors‘ sphere of influence modeled on the operation. Actors may be either human users or technical systems. Partitions are used to specify and represent responsibility for conceptual functions and decisions. The specific requirements placed on interfaces and the required business objects can be derived from these responsibilities.*

1. Grenzen Sie die Modellierungstechniken Entscheidungstabellen und Zustandstabellen voneinander ab.

*Entscheidungstabellen sind eine Technik zur Spezifikation von Komponentenverhalten. Sie bieten eine Möglichkeit komplexe Geschäftsregeln übersichtlich zu dokumentieren. Die Entscheidungstabelle eignet sich gut zur Veranschaulichung von komplexeren Regelzusammenhängen. Sie hilft außerdem eine Menge von Geschäftsregeln auf Vollständigkeit zu prüfen.*

*Eine Zustandstabelle wird eingesetzt, um mögliche Aktionen oder Funktionen in Abhängigkeit des aktuellen Zustands von Geschäftsobjekten, Komponenten oder Prozessen zu ermöglichen. Sie zeigt an, welche Aktionen in welchen Zuständen erlaubt bzw. nicht erlaubt sind.*

2 Explain the distinction between the modeling techniques used in decision tables and state transition tables.

*Decision tables are a technique used in the specification of component behavior. They allow complex business rules to be clearly documented. Decision tables are ideal for illustrating more complex rule correlations and are also useful for checking sets of business rules for completeness.*

*State transition tables are used to facilitate potential actions or functions depending on the current state of business objects, components, or processes. They indicate which actions are admissible / inadmissible in which states.*

**Unit 4.1**

1. Beschreiben Sie kurz, wie der Objektknoten im UML-Aktivitätsdiagramm die Spezifikation von Verhalten an Systemschnittstellen unterstützt.

*Ein Objektknoten im Aktivitätsdiagramm wird eingesetzt, um darzustellen, welche Art von Information eine Aktion benötigt, um ausgeführt zu werden, und welche Art Information nach der Ausführung einer Aktion ausgegeben wird. Mit Hilfe von Partitionen und Objektknoten im Aktivitätsdiagramm können auf eine einfache Art und Weise komplexe Abläufe unter Einbeziehung der zwischen Systemen ausgetauschten Nachrichten modelliert werden. Somit lässt sich ein einfacher Überblick über die benötigten Nachrichtentypen darstellen.*

1. Briefly describe how the object node in a UML activity diagram supports the specification of behavior at system interfaces.

*In activity diagrams, object nodes are used to show which type of information an action needs in order to be executed, and which type of information is output after executing an action. The use of partitions and object nodes in an activity diagram enables the easy modeling of complex operations, including the messages exchanged between systems. This produces a straightforward overview of the required message types.*

1. Nennen Sie Stärken und Schwächen des UML-Sequenzdiagramms bei der Spezifikation von Systemschnittstellen.
* Stärken:
	1. Darstellung komplizierter Interaktionen und technischer Interaktionskaskaden in ihrer genauen Reihenfolge;
	2. Verbindung zwischen Ablauf und Datenstruktur.
* Schwächen:
	1. Fokus liegt auf konkreten Abläufen;
	2. Ablauf isoliert vom fachlichen Anwendungsfall;
	3. Wird sehr schnell sehr komplex und technisch;
	4. Nur mit fortgeschrittenen UML-Kenntnissen lesbar.

2 List the strengths and weaknesses of UML sequence diagrams in the specification of system interfaces.

* Strengths:

a Modeling complex interactions and interaction cascades in precise order

b Link between operation and data structure.

* Weaknesses:

c Focus on specific operations

d The operation is isolated from the conceptual application

e Very quickly becomes highly complex and technical

f Only readable with advanced knowledge of UML.

**Unit 4.2**

1. Nennen Sie drei mögliche Startpunkte für die Identifikation konkreter Funktionen von Systemschnittstellen.
* Jeder Aufrufbeziehung von Komponenten zu Schnittstellen im Komponentendiagramm muss eine oder mehrere Funktionen der Schnittstelle zugeordnet werden können.
* Jedem Aufruf im Sequenzdiagramm muss eine entsprechende Funktion der Schnittstelle zugeordnet werden können.
* Jeder Schnittpunkt von Kontrollfluss und Partitionsgrenze im Aktivitätsdiagramm ist ein Indiz für eine Funktion einer Schnittstelle
1. Name three possible starting points for identifying the particular functions of system interfaces.
* Every call relationship between components and interfaces in the component diagram must be assigned to one or more interface functions.
* Every call in the sequence diagram must be assigned to a corresponding interface function.
* Every intersection between the control flow and the partition boundary in the activity diagram indicates an interface function
1. Erläutern Sie, wie Sie prüfen können, ob Sie alle benötigten Nachrichtentypen an einer Schnittstelle identifiziert haben.

*Alle benötigten und gelieferten Nachrichten, die als Objektknoten im Aktivitätsdiagramm modelliert werden, lassen sich auch in der Schnittstellen identifizieren. Es dürfen in der Schnittstellenbeschreibung nicht weniger Parameter und Ergebnisse spezifiziert werden, als aus der Summe der bereits verfügbaren Aktivitätsdiagramme ablesbar sind.*

*Darüber hinaus muss geprüft werden, ob jeder Nachricht im Sequenzdiagramm ein Nachrichtentyp der Schnittstellenspezifikation zugeordnet werden kann.*

2 Explain how to check whether you have identified all required message types at an interface.

*All required and delivered messages that are modeled as object nodes in the activity diagram can also be identified in the interfaces. The interface description should not specify fewer parameters and results than can be read from the sum total of activity diagrams already available.*

*It is also important to check whether a message type in the interface specification can be assigned to every message in the sequence diagram.*

1. Nennen Sie vier Quellen für detaillierte Attribute, die für Datenstrukturen in Systemschnittstellen benötigt werden.
	* *Bereits dokumentierte Anforderungen;*
	* *Schnittstellenbeschreibungen bereits existierender Komponenten;*
	* *Industriestandards, Normen;*
	* *Unternehmensrichtlinien.*

3 Name four sources of detailed attributes required for data structures in system interfaces.

* + *Existing documented requirements*
	+ *Interface descriptions of existing components*
	+ *Industry standards & guidelines*
	+ *Company policies*

**Unit 5.1**

1. Nennen Sie drei Bereiche einer Spezifikation, in denen fachlichen Datenmodelle beschrieben werden.

*Fachlichen Entitäten zur Darstellung, welche Informationen das System intern verarbeiten beziehungsweise unterstützen muss.*

*Fachliches Modell eines Geschäftsobjektes, dessen konkrete Daten über einen Bildschirmdialog vom Nutzer eingegeben werden müssen, z. B. der Antrag auf eine KFZ-Haftpflichtversicherung.*

*Fachliches Modell einer Nachricht, die über eine technische Systemschnittstelle ausgetauscht wird, z. B. der Datensatz einer gewünschten Flugreise zur Abfrage an alle Buchungssysteme für Fluglinien.*

2 Cite three areas of a specification which describe conceptual data models.

*Conceptual entities to denote which information the system must process and/or support internally.*

*Conceptual model of a business object where data is entered by the user using a screen dialog, e.g. an application for a vehicle insurance policy*

*Conceptual model of a message exchanged via a technical system interface, e.g. the data record for a flight request sent to all airline booking systems.*

1. Beschreiben Sie kurz, wie das fachliche Datenmodell zur Prüfung der spezifizierten Anforderungen eingesetzt werden kann.

*Das fachliche Datenmodell kann bei der Ermittlung von Anforderungen als eine Art Checkliste helfen: Jedes relevante Element des Datenmodells muss durch das Softwaresystem unterstützt werden.*

3 Briefly describe how conceptual data models are used to check the specified requirements.

*The conceptual data model can help to identify requirements by acting as a kind of checklist: Each relevant element in the data model must be supported by the software system*

**Unit 5.2**

1. Erläutern Sie, welche Probleme es bei der eindeutigen Identifikation von Geschäftsobjekten geben kann und wie diese bei der Spezifikation von Datenmodellen berücksichtigt werden können.

*Attribute von Geschäftsobjekten können sich verändern, bzw. können fachliche Werte von Attributen identisch sein, jedoch verschiedene Objekte bezeichnen. Daher lassen sich Geschäftsobjekte nicht immer auf Grundlage der fachlichen Datensätze zuverlässig unterscheiden. Mit der Ergänzung eines ID-Attributes wird das Datenmodell um eine Eigenschaft ergänzt, mit der sich Datensätze auch noch dann unterscheiden bzw. identifizieren lassen, wenn die fachlichen Attribute eines Geschäftsobjektes dazu nicht ausreichen.*

1 Explain which problems can arise with the unique identification of business objects and how allowance can be made for this when specifying data models.

*Attributes of business objects may change, or the values of attributes may be identical but refer to different objects. For this reason, conceptual data records do not always allow a reliable distinction between business objects. Adding an ID attribute to the data model allows individual data records to be distinguished and identified if the functional attributes of a business object are insufficient for this purpose.*

**Unit 5.3**

1. Begründen Sie kurz, warum sich das Objektdiagramm zur Prüfung von fachlichen Datenmodellen eignet.

*Mit Hilfe von Objektdiagrammen lassen sich bestimmte komplexe Systemzustände darstellen. Mit dieser Darstellung lassen sich zum einen die fachliche Korrektheit von Klassendiagramm prüfen und zum anderen der aktuelle Zustand von Datensätzen bewerten. Insbesondere Mengenangaben lassen sich mit dem Objektdiagramm gut darstellen. Durch diese Darstellung von Datensätzen auf Instanzebene, und nicht wie im Klassendiagramm auf Typebene, ist für das Verständnis des Objektdiagramms kein so großes Abstraktionsvermögen erforderlich, wie für das Verständnis von Klassendiagrammen.*

1 Briefly justify why object diagrams are suitable for checking conceptual data models.

*Object diagrams can be used to denote certain complex system states. This model allows us to check both the functional accuracy of class diagrams and assess the current status of data records. In particular, the object diagram is ideal for modeling quantities. Denoting data records at instance level in this way, rather than at type level in a class diagram, does not require as much abstract reasoning capacity.*

1. Begründen Sie kurz, warum ein Objektdiagramm mit möglichst echten Datensätzen modelliert werden sollte.

*Kann für jeden relevanten Datensatz ein entsprechendes Objektdiagramm erstellt werden, so ist das ein Indiz für die Eignung des spezifizierten Datenmodells. Häufig werden bei jedoch erst bei der Befüllung von Objektdiagrammen mit echten Daten Lücken und Schwachstellen im Datenmodell identifiziert, die anschließend noch behoben werden müssen.*

2 Briefly justify why object diagrams should be modeled with real-life data records wherever possible.

*If an equivalent object diagram can be prepared for every relevant data record, this indicates the suitability of the specified data model. Often, however, gaps and weaknesses in data models do not become apparent until real-life data is added, and will then need to be rectified.*

1. Begründen Sie kurz, warum es sinnvoll sein kann, im Objektdiagramm nicht alle Attribute von Klassen zu modellieren.

*Oft wird ein Objektdiagramm eingesetzt, um ganz konkrete Wertekonstellationen oder Zustände zu veranschaulichen oder zu diskutieren. Der Ersteller eines Objektdiagramms sollte daher nur die aktuell für das Verständnis relevanten Werte angeben. Alle übrigen Werte müssen nicht belegt werden.*

3 Briefly justify why it sometimes makes sense not to model all attributes of classes in an object diagram.

*Object diagrams are often used to illustrate specific value constellations or states. As such, the authors of object diagrams should only specify the values which are currently relevant as an aid to comprehension. All other values need not be documented.*

**Unit 5.4**

1. Erläutern Sie kurz, was Entitäten sind und wozu sie eingesetzt werden.

*Entitäten sind Elemente des Datenmodells, die eine fachliche Identität haben. Entitäten unterliegen häufig einem Lebenszyklus, der mit einem Zustandsdiagramm beschrieben werden kann. Die Eigenschaften einer Entität können sich über den Lebenszyklus ändern.*

1 Briefly explain what entities are and what they are used for.

*Entities are elements of a data model with a conceptual identity. Entities are often subject to a lifecycle which can be described in a state diagram. The properties of an entity may change over the course of its lifecycle.*

1. Erläutern Sie kurz, was Werteobjekte sind und wozu sie eingesetzt werden.

*Werteobjekte (engl.: value objects) sind Datenobjekte ohne eigene Identität. Sie haben keinen eigenen Lebenszyklus und auch kein fachliches ID-Attribut. Wertobjekte werden unter anderem eingesetzt um zusätzliche Informationen zu Entitäten zu speichern.*

2 Briefly explain what value objects are and how they are used.

*Value objects are data objects without a separate identity. They do not have their own lifecycle nor do they have a conceptual ID attribute. Value objects are used, among other things, to save additional information about entities.*

1. Erläutern Sie kurz, was Dienste sind und wozu sie eingesetzt werden.

*Als Dienste (auch: Service) werden zustandslose Fachfunktionen bezeichnet, die nicht direkt Entitäten oder Werteobjekten zugeordnet werden können. Zustandslos bedeutet, dass ein Dienst selber keine Attribute hat und damit über keinen inneren Zustand verfügt. Zustandslose Funktionen liefern bei gleichen Parametern immer das gleiche Ergebnis zurück.*

3 Briefly explain what services are and how they are used.

*Services refer to stateless functions which cannot directly be ascribed to entities or value objects. Stateless means that a service itself does not have any attributes and hence no inner state. Stateless functions will always deliver the same result with the same parameters.*

**Unit 6.1**

1. Welche Anforderungen sind bei der Übersetzung einer internen Datenstruktur in ein Austauschformat zu beachten?
* *Erhaltung der Datenstruktur;*
* *Einfache Lesbarkeit von Menschen;*
* *Effiziente Speicherung der Information.*

1 Which requirements must be met when converting an internal data structure into an exchange format?

* *Preserve the data structure*
* *Be easily readable by humans*
* *Store information efficiently.*
1. Nennen Sie die Vor- und Nachteile von binären Nachrichten.

*Vorteile:*

*Die technische Datenstruktur bleibt dabei erhalten und kann direkt rekonstruiert werden. Die Speicherung der Daten kann in der Regel sehr effizient erfolgen.*

*Nachteile:*

*Struktur und Inhalte von Binärnachrichten sind nicht direkt von Menschen lesbar und schreibbar.*

2 Outline the pros and cons of binary messages.

*Pros:*

*The technical data structure is preserved and can be directly reconstructed. Data storage is generally very efficient.*

*Cons:*

*The structure and content of binary messages are cannot be directly read or written by humans.*

1. Nennen Sie die Vor- und Nachteile von strukturierten Textnachrichten.

*Vorteile:*

*Strukturierte Textnachrichten können unter Beachtung der festgelegten Struktur von Menschen gelesen und auch geschrieben werden. Mögliche Syntaxfehler werden sehr einfach erkannt, da jede Nachricht automatisch vor der Verarbeitung auf korrekte Form hin geprüft wird.*

*Nachteile:*

*Umwandlung der technischen Datenstruktur in die Nachricht erforderlich. In der Regel größerer Speicherbedarf als binäre Daten.*

3 Outline the pros and cons of structured text messages.

*Pros:*

*Structured text messages can be read and written by humans provided the specified structure is observed. Potential syntax errors are very easily identified because every message is automatically checked for correct formatting prior to processing.*

*Cons:*

*The technical data structure must be converted into the message. The storage requirements are usually larger than for binary data.*

**Unit 6.2**

1. Worin unterscheiden sich Attribute und Elemente in XML-Dateien?

*Die Namen der XML-Elemente werden zwischen spitzen Klammern „<“ und „>“ dargestellt und Tag genannt. Der Inhalt eines XML-Elements steht zwischen dem öffnenden Tag und dem schließenden Tag. Attribute sind dazu geeignet, XML-Elemente mit Eigenschaften zu versehen. Sie dürfen ausschließlich im Anfangs-Tag des Elements stehen.*

1 What is the difference between attributes and elements in XML files?

*The names of XML elements are written between pointed brackets (< and >) and are referred to as tags. The content of an XML element is located between the opening tag and the closing tag. Attributes are used for ascribing characteristics to XML elements and must only be positioned in the element‘s initial tag.*

1. Welche verschiedenen Elementtypen gibt es in XML? Geben Sie zu jedem ein selbstgewähltes Beispiel in XML an.

***Einfache Elemente*** *enthalten nur Zeichenketten. Im unten aufgeführten Beispiel ist die Bestellnummer ein einfaches Element:*

<bestellnr>48729</bestellnr>

***Strukturierte Elemente*** *enthalten weitere Elemente. Im Beispiel trifft das auf jede Position der Bestellung zu:*

 <position>

 <id>1</id>

 <artikelnr>57823566</artikelnr>

 <menge>4</menge>

 <preisEUR>15,90</preisEUR>

 </position>

***Leere Elemente*** *besitzen keinen Inhalt. Die Information steckt also lediglich in der Präsenz des Elements und – falls vorhanden – in dessen Attributen. Das HTML-Element für Zeilenumbrüche ist hierfür ein Beispiel:*

<br />

2 What are the different element types in XML? Give an example of each that you have thought of yourself.

***Simple elements*** *only contain strings of characters. In the example below, the order number is a simple element:*

<OrderNo>48729</OrderNo>

***Structured elements*** *contain further elements. In this example it applies to every position in the order:*

 <position>

 <id>1</id>

 <itemno>57823566</itemno>

 <quantity>4</quantity>

 <priceEUR>15,90</priceEUR>

 </position>

***Empty elements*** *have no content. The presence of the element itself and – if applicable – its attributes are the only information they contain. The HTML element for line breaks is an example of this:*

<br />

1. Worin unterscheiden sich die Eigenschaften Gültigkeit und Wohlgeformtheit von XML-Dokumenten?

*Wohlgeformtheit bedeutet, dass alle allgemeinen Syntaxregeln eingehalten wurden. Gültigkeit bedeutet, dass auch die Regeln eines zu der XML-Datei angegebenen Schemas erfüllt werden.*

3 What is the difference between the properties “valid” and “well-formed” for XML documents?

*Well-formed means that all general syntax rules have been observed. Valid means that the rules of a schema specified for the XML file have been observed.*

**Unit 6.3**

1. Spezifizieren Sie das folgende einfache XML-Element mit XML-Schema:

**<lieferdatum>2014-05-23</lieferdatum >**

*Lösung:*

<xs:element name="deliverydate" type="xs:date" />

1 Specify the following simple XML element with an XML schema:

**<deliverydate>2014-05-23</deliverydate >**

*Solution:*

<xs:element name="deliverydate type="xs:date" />

**Unit 6.4**

1. Erläutern Sie, wie Kompositoren aus einem XML-Schema in ein UML-Klassendiagramm übertragen werden können.

*Alle Kompositoren sowie das mehrfache Auftreten von Elementen werden durch Assoziationen ausgedrückt. Die Richtung der Assoziation ergibt sich dabei aus der Eltern-Kind-Abhängigkeit der Elemente. Die maxOccurs- und minOccurs-Attribute werden hierbei mit berücksichtigt, auch wenn deren Standardwert gilt. Sie werden durch Multiplizitäten im Klassendiagramm dargestellt. Die Abbildung des Kompositors von xs:choice in das Klassendiagramm ist nur bedingt möglich. Für den Fall, dass sich die einzelnen Elemente innerhalb von xs:choice sinnvoll unter einem Oberbegriff verallgemeinern lassen, kann im Klassendiagramm die Vererbung eingesetzt werden. Geht das nicht, ist eine Darstellung in UML nicht direkt möglich. Für diesen Fall sollte eine Notiz im Diagramm vermerkt werden, dass jeweils nur eines aus der Menge der Elemente auftreten kann.*

1 Explain how compositors from an XML schema can be transferred into a UML class diagram.

*All compositors and multiple occurrences of elements are expressed by associations. The direction of association is derived from the parent/child dependency of the elements. The maxOccurs und minOccurs attributes are also taken into account, even when the default value applies. They are represented in the class diagram by multiplicities. The compositor of xs:choice can only be modeled in the class diagram under certain circumstances. If it is expedient to generalize the individual elements within xs:choice under an umbrella term, inheritance may be used in the class diagram. If this is not possible, it cannot be directly modeled in UML. In such cases, a note should be added to the diagram that only one element from the set may occur.*

1. Erläutern Sie, warum die XML-Datentypen mit in das Klassendiagramm übertragen werden müssen.

*Die tatsächlichen technischen Datentypen der Attribute werden dabei übernommen, damit im späteren Verlauf des Projektes die Typsicherheit gewährleistet werden kann.*

2 Explain why XML data types must also be transferred into the class diagram.

*The actual technical data types of attributes are transferred in order to be able to guarantee type safety over the subsequent course of the project.*

**Unit 6.5**

1. Nennen Sie die XML-Sprache mit der Webservices typischerweise spezifiziert werden und nennen Sie die typischen Elemente der Spezifikation von Webservices.

*Eine XML-Sprache zur Spezifikation von Webservices ist WDSL. In einer WDSL-Datei werden zu einem Webservice folgende Elemente definiert:*

* *Funktionen des Webservices mit deren Ein- und Ausgabenachrichten;*
* *Adresse des Services (URL); sowie*
* *weitere technische Angaben, technischer Zugriff und Deployment.*

1 Name the XML language typically used to specify Web services and name the typical elements used in the specification of Web services.

*WDSL is an XML language used in the specification of Web services. The following elements are defined in a WDSL file about a Web service:*

* *Functions of the Web service with its input and output messages;*
* *Address of the service (URL);*
* *Other technical information, technical access, and deployment.*
1. Welche Elemente müssen zu jeder Funktion eines Webservice spezifiziert werden?
* *Name*
* *Beschreibung*
* *Eingabenachricht*
* *Ausgabenachricht*

2 Which elements must be specified for each function of a Web service?

* *Name*
* *Description*
* *Input message*
* *Output message*
1. Beschreiben Sie den Ablauf der Kommunikation zwischen einem Client und einem Server, die über einen Webservice und per XML-Nachrichten miteinander kommunizieren.

*Der Client sendet eine XML-Nachricht an den Webservice. Dieser validiert zunächst, ob die Nachricht wohlgeformt und gültig ist. Anschließend werden die Parameter ausgelesen und auf deren Basis das Ergebnis ermittelt. Das Ergebnis wird in eine XML-Nachricht geschrieben, die dem Client als Antwort übermittelt wird. Der Client prüft nun die Nachricht auf Wohlgeformtheit und Gültigkeit. Besteht die XML-Nachricht beide Prüfungen, wird das Ergebnis verarbeitet.*

3 Describe the communication procedure between a client and a server communicating with one another via a Web service using XML messages.

*The client sends an XML message to the Web service, which first validates whether the message is well-formed and valid. It then reads the parameters and calculates the result on that basis. The result is written into an XML message sent to the client as a response. The client then checks that the message is well-formed and valid. If the XML message passes both tests, the result will be processed*

**Unit 7.1**

1. Nennen und erläutern Sie, was die Anwendung des SMART-Modells auf die Qualitätseigenschaften bedeutet.

*Die formulierten Qualitätseigenschaften für Softwaresysteme sollten nach dem Schema SMART beschrieben werden. SMART ist ein Akronym, das wie folgt aufgelöst werden kann:*

* + ***S*** *= spezifisch: Qualitätseigenschaften sollen deutlich beschrieben und definiert sein.*
	+ ***M*** *= messbar (qualitativ, quantitativ): Die Umsetzung der Qualitätseigenschaften muss geprüft werden können.*
	+ ***A*** *= akzeptabel: Die Qualitätseigenschaften müssen von den Stakeholdern des Projekts akzeptiert werden, dazu zählt auch die Akzeptanz der Prüfmethode.*
	+ ***R*** *= realistisch: Qualitätseigenschaften müssen unter Berücksichtigung der fachlichen und organisatorischen Rahmenbedingungen erreicht werden können.*
	+ ***T*** *= terminiert: Qualitätseigenschaften müssen innerhalb eines vorgesehenen Zeitrahmens, z. B. in einem bestimmten Release umgesetzt werden.*

1 Explain what is meant by application of the SMART model to quality characteristics.

*The formulated quality characteristics for software systems should be described according to the SMART model. SMART is an acronym as follows:*

* + ***S*** *= Specific: Quality characteristics should be clearly described and defined.*
	+ ***M*** *= Measurable (qualitative, quantitative): It must be possible to verify implementation of the quality characteristics.*
	+ ***A*** *= Acceptable: The quality characteristics, including the testing methods used, must be accepted by the project’s stakeholders.*
	+ ***R*** *= Realistic: Quality characteristics must be achievable, with due regard for functional and organizational framework conditions.*
	+ ***T*** *= Time-bound: Quality characteristics must be implemented within a set time frame, e.g. in a certain release.*
1. Was ist die Grundidee eines Qualitätsmodells?

*Die Grundidee eines Qualitätsmodells ist die Zergliederung eines abstrakteren Oberbegriffs in verschiedene Unterbegriffe. Diese Unterbegriffe werden wiederum so lange zergliedert, bis eine konkret messbare Qualitätseigenschaft formuliert werden kann. Ein Qualitätsmodell besteht daher aus einer Hierarchie von Begriffen, die bei der Spezifikation konkreter Qualitätseigenschaften helfen können.*

2 What is the basic precept of a quality model?

*The basic precept of a quality model is that the more abstract, overall concept of quality is broken down into various sub-concepts. These sub-concepts, in turn, are broken down still further until it becomes possible to formulate a specific, measurable quality characteristic. Quality models are therefore comprised of a hierarchy of concepts that are useful when specifying tangible quality characteristics.*

**Unit 7.2**

1. Beschreiben Sie kurz was die GQM-Methode ist und wozu diese eingesetzt wird.

*Die GQM-Methoden ist ein Vorgehensmodell zur Spezifikation von prüfbaren Qualitätseigenschaften und der Durchführung ihrer Messungen. Ziel von GQM ist die Entwicklung von speziell auf ein spezifisches Softwaresystem angepasste Qualitätseigenschaften.*

1 Briefly describe the GQM method and where it is used.

*The GQM method is a procedural model for specifying and measuring verifiable quality characteristics. GQM aims to develop quality characteristics that are tailored to a specific software system.*

1. Welche Schritte der GQM-Methode müssen zur Spezifikation von Qualitätseigenschaften durchgeführt werden?

*Die GQM-Methode lässt sich zur Spezifikation von Qualitätseigenschaften in die folgenden Schritte einteilen:*

1. *Definition der Auswertungsziele (engl.: Goals)*
2. *Verfeinerung der Auswertungsziele mit einem Qualitätsmodell*
3. *Ableitung der Fragestellungen (engl.: Questions)*
4. *Ableitung der Messewerte und Indikatoren (engl.: Metrics)*
5. *Festlegen von geforderten konkreten Messwerten*
6. *Vorgehen und Mechanismen zum Messen festlegen*
7. *Formulierung von prüfbaren Qualitätseigenschaften*

2 When applying the GQM method, which steps must be implemented to specify quality characteristics?

*The GQM method used to specify quality characteristics can be broken down into the following steps:*

1. *Define the evaluation goals*

*2 Refine the evaluation goals with a quality model*

*3 Derive the questions*

*4 Derive the metrics*

*5 Define the specific measurements required*

*6 Define the measurement procedure and mechanisms*

*7 Formulate verifiable quality characteristics*