



WIRTSCHAFTSKAMMER OBERÖSTERREICH
Lehrlingsstelle und Meisterprüfungsstelle

THEMENKATALOG

für die Vorbereitung

auf die

Lehrabschlussprüfung

Applikationsentwicklung - Coding

Im Wesentlichen ausgearbeitet

Inhaltsverzeichnis

Lehrabschlussprüfung Applikationsentwicklung - Coding	3
Theoretische Prüfung	3
Praktische Prüfarbeit	3
Fachgespräch	3
1) Grundlagen in der Informationstechnik	4
2) Betriebssysteme und Software	4
3) Betreuung von mobiler Hardware	4
4) Technische Dokumentationen/Projektarbeit/Schulungen	4
5) Gesetzliche Bestimmungen im Zusammenhang mit Applikationsentwicklung – Coding	5
6) Netzwerkdienste	5
7) IT-Security und Betriebssicherheit	5
8) Informatik und Gesellschaft	6
9) Ergonomische Gestaltung eines Arbeitsplatzes	6
10) Fachberatung, Planung	6
11) Informatik	6
12) Projektmanagement	7
13) Projektmethoden, Tools	8
14) Qualitätssicherung	8
15) Grundkenntnisse des Programmierens	8
16) Kenntnis und Verwendung von Datenbanken, Datenmodellen und Datenstrukturen	9
17) Systementwicklung/Testkonzepte	10
18) Übungsbeispiel	11

Lehrabschlussprüfung Applikationsentwicklung - Coding

Theoretische Prüfung

Die theoretische Prüfung entfällt, wenn der Prüfungskandidat das Erreichen des Lehrzieles durch ein anerkanntes positives Zeugnis (z.B. 4. Klasse Berufsschule) nachgewiesen hat.

Datentechnik und Systemmanagement

Unterschiede Cloud- und On-Premise-Lösungen, Client- und Serverbetriebssysteme, Kenntnis der Möglichkeiten des Datenaustausches, Urheberrecht und Datenschutz

Angewandte Mathematik

Dateigrößen und Datenkapazitäten, Datendurchsatz, Gleichungen, Zahlensysteme
Das Verwenden von Rechenhilfen, Formeln und Tabellen ist zulässig.

Applikationsentwicklung

Sortieralgorithmen, Suchalgorithmen, objektorientierte Programmierung, relationales Datenbankdesign, Grundlagen des Internets

Praktische Prüfarbeit

Die gesamte Prüfarbeit ist zu dokumentieren (u. a. auch Sourcecode).

1. Teil der Prüfarbeit (Vormittag) - Arbeitszeit 3,5 Stunden

Folgende Punkte können beim 1. Teil der Prüfarbeit abgeprüft werden:

Datenbank/Konzeption eines Tests:

- Entwicklung einer Datenbank (ER-Diagramm in der 3. Normalform)
- Beziehungen (1:1, 1:n)
- SQL-Datenbankerzeugung (MySQL)
- Softwaretest

2. Teil der Prüfarbeit (Nachmittag) - Arbeitszeit 3,5 Stunden

Folgende Punkte können beim 2. Teil der Prüfarbeit abgeprüft werden:

Anpassung einer Bedienoberfläche:

- Softwareinstallation und -konfiguration einer beliebigen Freeware (z. B. PHP, MySQL)
- Erstellen von Applikationen mit React Native
- Datenbankerzeugung anhand einer vorgegebenen Datenbank inkl. Testeinträge
- Programmieren einer Bedienoberfläche
- Suche und Ausgabe von Inhalten über eine Bedienoberfläche
- Berechnungen
- Fehlermeldungen
- Einsatz von Frameworks (z. B. Bootstrap) zur Gestaltung von Oberflächen und clientseitigen Skriptsprachen (z. B. JavaScript)/Programmiersprachen

Fachgespräch

Inhalte gemäß Themenkatalog

Themen und Fragen werden von der Prüfungskommission ausgewählt

Dauer ca. 15 - 25 Minuten pro Prüfungskandidat

1) Grundlagen in der Informationstechnik

Kenntnis des Zeichensatzes ASCII

ASCII (American Standard Code for Information Interchange) ist ein Zeichensatz, der 128 Zeichen umfasst, darunter Buchstaben, Zahlen, Satzzeichen und Steuerzeichen. Er verwendet 7 Bits pro Zeichen und wird in Computern und Kommunikationssystemen weit verbreitet.

Kenntnis der Einheiten Bit, Byte

Ein Bit (Binary Digit) ist die kleinste Informationseinheit und kann entweder 0 oder 1 sein. Ein Byte besteht aus 8 Bits und ist die grundlegende Maßeinheit für Datengrößen in Computern.

Kenntnis der Begriffe Gigabyte, Terabyte, Petabyte, Exabyte

- Gigabyte (GB): 1 GB = 1.000 Megabyte (MB), entspricht etwa 1 Milliarde Bytes.
- Terabyte (TB): 1 TB = 1.000 Gigabyte (GB), entspricht etwa 1 Billion Bytes.
- Petabyte (PB): 1 PB = 1.000 Terabyte (TB), entspricht etwa 1 Billiarde Bytes.
- Exabyte (EB): 1 EB = 1.000 Petabyte (PB), entspricht etwa 1 Trillion Bytes.

Kenntnis der Begriffe Gibibyte, Tebibyte, Pebibyte, Exbibyte

- Gibibyte (GiB): 1 GiB = 1.024 Mebibyte (MiB) = 1.073.741.824 Bytes.
- Tebibyte (TiB): 1 TiB = 1.024 Gibibyte (GiB) = 1.099.511.627.776 Bytes.
- Pebibyte (PiB): 1 PiB = 1.024 Tebibyte (TiB) = 1.125.899.906.842.624 Bytes.
- Exbibyte (EiB): 1 EiB = 1.024 Pebibyte (PiB) = 1.152.921.504.606.846.976 Bytes.

Kenntnis der gebräuchlichen Zahlensysteme in der IT und Umwandlung zwischen Binär-, Dezimal- und Hexadezimalzahlen

- Binär (Basis 2): Verwendet nur die Ziffern 0 und 1. Es ist die Grundlage der digitalen Datenverarbeitung.
- Dezimal (Basis 10): Das alltägliche Zahlensystem, das die Ziffern 0 bis 9 umfasst.
- Hexadezimal (Basis 16): Verwendet die Ziffern 0-9 und die Buchstaben A-F. Es ist eine kompakte Möglichkeit, Binärzahlen darzustellen.
- Oktal (Basis 8): Verwendet die Ziffern 0-7. Weniger gebräuchlich, wurde früher oft in der Programmierung verwendet.

Kenntnis der Logik-Schaltungen (AND, OR, XOR, NOT) und deren Wahrheitstabellen

2) Betriebssysteme und Software

Fachbegriff Betriebssystem

Ein Betriebssystem (OS) ist eine Software, die die Hardware eines Computers verwaltet und eine Schnittstelle zwischen Benutzer und Hardware bietet. Es ermöglicht das Ausführen von Programmen, die Verwaltung von Ressourcen (wie Prozessor, Speicher und Peripheriegeräte) und sorgt für eine benutzerfreundliche Umgebung zur Interaktion mit

dem System. Bekannte Betriebssysteme sind Windows, macOS, Linux und Android.

Kenntnis der am Markt führend verbreiteten Betriebssysteme

- Microsoft Windows: Das meistgenutzte Desktop-Betriebssystem weltweit, besonders auf PCs und Laptops.
- macOS: Das Betriebssystem von Apple, vor allem auf Macs (Desktop und Laptop) verbreitet.
- Linux: Ein Open-Source-Betriebssystem, das auf Servern und als Desktop-Betriebssystem in verschiedenen Distributionen wie Ubuntu, Fedora und Debian genutzt wird.
- Android: Ein Open-Source-Betriebssystem von Google, das vor allem auf Smartphones und Tablets verbreitet ist.
- iOS: Das mobile Betriebssystem von Apple, das auf iPhones und iPads läuft.

Kenntnisse über Desktop-Betriebssysteme

Siehe vorhergehende Frage

Fachbegriff Firmware

Firmware ist eine spezielle Art von Software, die in Hardware-Geräten eingebettet ist und deren grundlegende Funktionen steuert. Sie befindet sich in nicht-flüchtigem Speicher (wie z. B. Flash-Speicher oder ROM) und wird in der Regel nicht häufig geändert. Firmware stellt die Verbindung zwischen der Hardware und der darüberliegenden Software her, indem sie Hardwarekomponenten wie Prozessoren, Speicher oder Peripheriegeräte steuert. Beispiele für Firmware sind die BIOS/UEFI-Software auf Computermotherboards, die Software in Druckern oder auch in Routern.

Fachbegriffe Systemprogramm, Anwendungsprogramm

Systemprogramm: Ein Systemprogramm ist Software, die das Betriebssystem und die Verwaltung der Hardware unterstützt. Es ermöglicht die Ausführung und Verwaltung von Anwendungsprogrammen und stellt grundlegende Funktionen für den Betrieb eines Computers zur Verfügung. Beispiele für Systemprogramme sind Betriebssysteme (wie Windows, macOS, Linux), Compiler, Treiber und Dienstprogramme (wie Dateimanager oder Backup-Software).

Anwendungsprogramm: Ein Anwendungsprogramm ist Software, die für den Endanwender bestimmte Aufgaben oder Funktionen ausführt, wie Textverarbeitung, Tabellenkalkulation, Bildbearbeitung oder Web-Browsing. Diese Programme werden direkt vom Benutzer genutzt und beruhen auf der Unterstützung durch das Betriebssystem und Systemprogramme. Beispiele sind Microsoft Word, Excel, Photoshop und Webbrowser wie Google Chrome.

Fachbegriff Multitasking-Betriebssystem

Ein Multitasking-Betriebssystem ist ein Betriebssystem, das in der Lage ist, mehrere Aufgaben (Prozesse) gleichzeitig auszuführen. Dies bedeutet, dass das System zwischen verschiedenen Programmen oder Prozessen hin- und herwechseln kann, sodass der Eindruck entsteht, dass alle gleichzeitig laufen.

Es gibt zwei Hauptarten von Multitasking:

- **Kooperatives Multitasking:** Hierbei müssen Programme freiwillig die Kontrolle abgeben, damit das Betriebssystem andere Prozesse ausführen kann.
- **Präemptives Multitasking:** Das Betriebssystem übernimmt die Kontrolle und unterbricht laufende Prozesse, um anderen Prozessen Rechenzeit zu geben, ohne dass diese freiwillig stoppen müssen.

Moderne Betriebssysteme wie Windows, macOS und Linux unterstützen präemptives Multitasking, um eine effizientere und reaktionsschnellere Nutzung zu gewährleisten.

Fachbegriffe Single-User-System, Multi-User-System

Single-User-System: Ein Single-User-System ist ein Betriebssystem, das für die Nutzung durch einen einzelnen Benutzer zu einem bestimmten Zeitpunkt ausgelegt ist. Es ermöglicht einem einzigen Benutzer den Zugriff auf die Ressourcen des Systems (wie CPU, Speicher und Peripheriegeräte) und wird auf einem einzelnen Gerät genutzt. Beispiele sind Personal Computer mit Windows oder macOS.

Multi-User-System: Ein Multi-User-System ist ein Betriebssystem, das mehreren Benutzern gleichzeitig den Zugriff auf dieselben Ressourcen eines Computers ermöglicht. Es verwaltet die gleichzeitige Nutzung und sorgt dafür, dass die Benutzer voneinander getrennt arbeiten können, ohne sich gegenseitig zu stören. Beispiele sind Server-Betriebssysteme wie Linux, Unix und Windows Server, die es mehreren Benutzern ermöglichen, gleichzeitig auf das System zuzugreifen.

Kenntnis über die Powershell (inkl. einfacher Befehle)

Die PowerShell ist eine leistungsstarke Kommandozeilen-Schnittstelle und Skriptsprache, die von Microsoft entwickelt wurde. Sie ist besonders nützlich für die Automatisierung von Verwaltungsaufgaben und für die Interaktion mit dem Betriebssystem, Servern und anderen Anwendungen. PowerShell kombiniert eine interaktive Shell mit einer Skriptsprache und bietet tiefgreifende Systemverwaltungsfunktionen.

Einige einfache PowerShell-Befehle:

Get-Help - Zeigt die Hilfe zu einem Befehl an.

Beispiel: Get-Help Get-Process

Kenntnisse über grafische Oberflächen unter Linux

Unter Linux gibt es mehrere grafische Benutzeroberflächen (GUIs), die den Zugriff auf das System erleichtern und eine benutzerfreundliche Interaktion ermöglichen. Die bekanntesten und am weitesten verbreiteten GUIs unter Linux sind sogenannte Desktop-Umgebungen. Jede Desktop-Umgebung bietet eine eigene Art der Benutzeroberfläche und Funktionalität, aber alle erfüllen den gleichen Zweck: Sie ermöglichen eine grafische Steuerung des Systems. Beispiele GNOME, KDE Plasma, Xfce (leichtgewichtig)

Fachbegriff Dateisystem

Ein **Dateisystem** ist eine Methode und Struktur, mit der Daten auf Speichermedien (wie Festplatten, SSDs oder USB-Sticks) organisiert und verwaltet werden. Es legt fest, wie Dateien gespeichert, benannt, verwaltet und abgerufen werden. Ein Dateisystem sorgt dafür, dass Daten effizient gespeichert und der Zugriff darauf korrekt und schnell erfolgt.

Wichtige Funktionen eines Dateisystems:

- **Speicherung und Organisation von Dateien:** Es verwaltet, wo und wie Daten auf dem Speichermedium abgelegt werden.
- **Zugriffssteuerung:** Es sorgt dafür, dass Benutzer oder Anwendungen die richtigen Berechtigungen haben, um auf bestimmte Dateien zuzugreifen.
- **Fehlerbehandlung:** Es bietet Mechanismen, um mit beschädigten oder verlorenen Dateien umzugehen.

Beispiele sind FAT32, NTFS (Windows) ext4 (Linux) APFS (Apple)

3) Betreuung von mobiler Hardware

Technische Merkmale von Smartphones

Die technischen Merkmale von Smartphones variieren je nach Modell und Hersteller, aber hier sind die wichtigsten und häufigsten technischen Merkmale, die man in modernen Smartphones findet:

- **Prozessor (CPU)**
 - Qualcomm Snapdragon, Apple A-Serie oder Samsung Exynos.
 - Kerne: Die Anzahl der Kerne (z. B. 4, 6, 8) beeinflusst die Fähigkeit zur Multitasking-Leistung.
- **RAM (Arbeitsspeicher)**
 - Der RAM speichert temporäre Daten und Programme, die gerade verwendet werden. Mehr RAM ermöglicht besseres Multitasking und schnellere Reaktionszeiten.
 - Typische Größen: 4 GB, 6 GB, 8 GB, 12 GB oder mehr.

- Speicher (ROM)
 - Der interne Speicher (ROM) wird zum Speichern von Apps, Fotos, Videos und anderen Dateien verwendet.
- Display
 - IPS, OLED oder AMOLED für bessere Farben und Kontraste. AMOLED ist für tiefere Schwarztöne und bessere Energieeffizienz bekannt.
- Kamera
- Akku
 - Kapazität und Ladegeschwindigkeit
- Betriebssystem
 - Android, iOS, Harmony-OS usw.
- Netzwerk und Konnektivität
 - 5G, WIFI (ac), Bluetooth, NFC (kontaktloses Bezahlen)
- Sensoren
 - Fingerabdruckscanner, Barometer, Lichtsensoren, Beschleunigungssensoren
- Design
 - Materialien, Schutz IP67, faltbare Displays,
- Sicherheitsfunktionen
 - Fingerprint, Face-ID, Verschlüsselung

Technische Merkmale von Tablets

Siehe Smartphone

Kenntnisse über Android

Android ist ein Open-Source-Betriebssystem, das von Google entwickelt wurde, hauptsächlich für mobile Geräte wie Smartphones und Tablets, aber auch für Smartwatches, Fernseher, Autos und andere IoT-Geräte. Linux-Kernel

Kenntnisse über iOS

iOS ist das mobile Betriebssystem von Apple, das exklusiv auf Apple-Geräten wie iPhones, iPads und iPods läuft. Es wurde erstmals 2007 zusammen mit dem iPhone eingeführt und hat sich seitdem zu einem der bekanntesten und am weitesten verbreiteten mobilen Betriebssysteme entwickelt. iOS bietet eine benutzerfreundliche, intuitive Oberfläche und ist bekannt für seine hohe Sicherheit, Stabilität und die enge Integration mit der Apple-Hardware.

Fachbegriff QR-Code

Ein QR-Code (Quick Response Code) ist ein zweidimensionaler Barcode, der entwickelt wurde, um schnell und einfach Informationen zu speichern und zu lesen. Er wird in verschiedenen Bereichen angewendet, einschließlich Marketing, Zahlungsabwicklung,

Veranstaltungen und vielem mehr. Merkmale: Struktur (Quadrat), Datenkapazität (4296 Zeichen), Fehlerkorrektur (3 Stufen L,M, H), Sicherheit (praktisch aber Sicherheitsrisiken)

Vor- und Nachteile von geschlossenen Systemen mit Betriebssystem und App-Store
Vorteile:

Sicherheit und Datenschutz, zentralisierte Kontrolle, App-Zertifizierung, regelmäßige Updates, Integration von Hardware,

Nachteile: eingeschränkte Anpassungsmöglichkeiten, höhere Kosten, abhängig vom Anbieter, eingeschränkte Interoperabilität

4) Technische Dokumentationen/Projektarbeit/Schulungen

Aufgabe und Strukturierung von Testläufen

Die Aufgabe und Strukturierung von Testläufen ist ein wesentlicher Bestandteil des Softwaretests und der Qualitätssicherung. Testläufe helfen, die Funktionalität und die Leistung eines Systems oder einer Anwendung zu überprüfen, bevor sie in den produktiven Einsatz gehen. Sie stellen sicher, dass die Software den Anforderungen entspricht und keine schwerwiegenden Fehler oder Probleme vorliegen.

Aufgabe: Fehleridentifikation, Verifikation von Anforderungen, Vertrauen in die Qualität, Regressionsprüfung, Dokumentation von Problemen und Verbesserungen

Protokollieren technischer Arbeiten

Protokollieren technischer Arbeiten ist ein wichtiger Bestandteil in vielen technischen Bereichen, insbesondere in der Softwareentwicklung, Systemadministration, Ingenieurwesen und Forschung. Ein gut strukturiertes Protokoll hilft dabei, den Fortschritt, durchgeführte Arbeiten, Ergebnisse und auftretende Probleme zu dokumentieren. Dies fördert nicht nur die Nachvollziehbarkeit und Transparenz, sondern hilft auch bei der Fehlerbehebung und zukünftigen Optimierungen.

Inhalt einer technischen Dokumentation/technisches Protokoll (z.B. FAQ, ...)

Eine technische Dokumentation oder ein technisches Protokoll dient dazu, technische Informationen strukturiert und nachvollziehbar zu präsentieren. Sie enthält alle relevanten Details über Produkte, Systeme, Software oder Arbeitsprozesse, die für Benutzer, Entwickler, Techniker oder andere Stakeholder von Bedeutung sind.

Titel, Ziel und Zweck, Einführung, Anforderungen und Voraussetzungen, Vorgehensweise, Ergebnisse, Lösungen, Fazit, FAQ

Aufbereitung einer technischen Dokumentation/technisches Protokoll

Planung und Struktur, Inhaltliche Aufbereitung, Erstellung von Anleitungen, Überprüfung, .

Kenntnis über Abläufe und Prozessschritte zum Roll-out von Applikationen (z.B. Einführungsvorgehen, Sicherheitsanforderungen, evtl. Abbruch und Rückführung, Datenmigration/Konvertierung, Anwenderschulung, Übergabe, Abnahme)
Gestaltung und Vorbereitung von Präsentationen

Planungsphase (Projekt), Sicherheitsanforderungen(Datenschutz, Backup, Berechtigungen), Abbruchstrategie(Sicherstellen das die Applikation nach Fehlern zurückgesetzt werden kann), Datenmigration(Analyse, Migrationstests, Validierung), Anwenderschulung(Planung, Ressourcen bereitstellen, Feedback einholen), Übernahme(Testphase, Übergabe und Betrieb, Abnahmeprotokoll), Monitoring(Support, Fehlerbehebung, usw)

5) Gesetzliche Bestimmungen im Zusammenhang mit Applikationsentwicklung - Coding

Kenntnis über DSGVO (Datenschutzgrundverordnung)

Die Hauptziele der DSGVO sind:

- Schutz personenbezogener Daten: Gewährleistung, dass personenbezogene Daten geschützt werden, und zwar durch angemessene technische und organisatorische Maßnahmen.
- Stärkung der Rechte der betroffenen Personen: Sicherstellung, dass Einzelpersonen mehr Kontrolle über ihre eigenen Daten haben.
- Harmonisierung der Datenschutzbestimmungen: Vereinheitlichung der Datenschutzregelungen innerhalb der EU, sodass Unternehmen die gleichen Standards einhalten müssen.

Fachbegriff "Datenminimierung" im Zusammenhang der DSGVO

müssen personenbezogene Daten adäquat, relevant und auf das notwendige Maß beschränkt sein. Zweckbindung, Menge der Daten, Speicherfrist

Fachbegriffe "betroffene Personen", Verantwortlicher, Auftragsverarbeiter

Eine **betroffene Person** ist jede natürliche Person, deren personenbezogene Daten verarbeitet werden. Das bedeutet, dass es sich um die Person handelt, deren Daten erfasst, gespeichert, genutzt oder auf andere Weise verarbeitet werden.

Der **Verantwortliche** ist die natürliche oder juristische Person, Behörde, Einrichtung oder andere Stelle, die allein oder gemeinsam mit anderen über die Zwecke und Mittel der Verarbeitung personenbezogener Daten entscheidet.

Ein **Auftragsverarbeiter** ist eine natürliche oder juristische Person, Behörde, Einrichtung oder andere Stelle, die personenbezogene Daten im Auftrag des Verantwortlichen verarbeitet.

Kenntnis über Rechte von "betroffene Personen" lt. DSGVO

Recht auf Auskunft, Recht auf Berichtigung, Recht auf Löschung

Fachbegriff "personenbezogene und sensible Daten" lt. DSGVO

Personenbezogene Daten sind alle Informationen, die sich auf eine identifizierte oder identifizierbare natürliche Person beziehen. Name, Adresse, Geburtsdatum, usw...

Bedeutung von Kopplungsverbot beim DSGVO

Das Kopplungsverbot besagt, dass eine Einwilligung zur Verarbeitung personenbezogener Daten nicht an die Erfüllung eines anderen Vertrages gekoppelt werden darf, wenn die Verarbeitung der Daten für den Vertrag nicht erforderlich ist.

Datenschutzbeauftragter lt. DSGVO und dessen Funktion

Der Datenschutzbeauftragte (DSB) ist eine zentrale Rolle im Datenschutzmanagement eines Unternehmens oder einer Organisation gemäß der Datenschutz-Grundverordnung. Der Datenschutzbeauftragte sorgt dafür, dass die Organisation die datenschutzrechtlichen Anforderungen einhält und unterstützt die betroffenen Personen und das Unternehmen bei der Umsetzung der DSGVO-Vorgaben.

Pflichten für Unternehmen bei bekannt gewordenen Datendiebstahl lt. DSGVO

Meldung an Aufsichtsbehörde (72 Stunden), Benachrichtigung der betroffenen Personen, Dokumentation des Vorfalls, Schutzmaßnahmen ergreifen, Sicherheitsvorkehrungen

Kenntnisse über Grundbegriffe und Gültigkeitsbereich des Urheberrechtes

Das Urheberrecht schützt die geistigen Schöpfungen von Schöpfern und Künstlern und stellt sicher, dass sie für ihre Werke vergütet werden, wenn diese verwendet oder verbreitet werden. Urheber->natürliche Person; Werk -> kreative Schöpfung; Verwertungsrechte-> Veröffentlichung und Vervielfältigung; Persönlichkeitsrechte -> Verfälschung oder Verfremdung

Kenntnis gesetzlicher Gewährleistungs- und Garantiebestimmungen und deren unterschiedlicher Anwendung bei Hardware- und Softwareproblemen

Die Gewährleistung und Garantie sind zwei verschiedene rechtliche Konzepte, die den Schutz von Konsumenten beim Kauf von Produkten, sowohl in Bezug auf Hardware als auch Software, betreffen.

Gewährleistung (2Jahre): gesetzlich vorgeschrieben; Ein Verkäufer muss für Mängel haften, wenn eine Ware defekt oder nicht funktionsfähig ist, wenn der Käufer nachweisen kann hat er Anspruch auf Nachbesserung, Ersatzlieferung, Minderung, usw.

Garantie: freiwillig vom Hersteller oder Verkäufer. Bedingung und Dauer wird individuell festgelegt

Kenntnisse über umweltgerechte Entsorgung von Elektronikschrott, Toner, Akkus oder Batterien

Allgemein: umweltgerechte Entsorgung sicherstellen, wegen schädlicher Stoffe wie Quecksilber, Fluorkohlenwasserstoffe, usw...

Elektronikschrott umfasst alle nicht mehr funktionierenden oder veralteten elektronischen Geräte.

Kenntnisse über das E-Commerce-Gesetz (ECG)

Das E-Commerce-Gesetz (ECG) regelt in vielen Ländern den rechtlichen Rahmen für den elektronischen Geschäftsverkehr. Es betrifft Unternehmen und Privatpersonen

Kenntnisse über das Telekom-Gesetz (TKG)

Das Telekommunikationsgesetz (TKG) ist ein zentrales Gesetz, das den Rechtsrahmen für die Telekommunikationsbranche regelt. Es schafft die Grundlage für die Telekommunikationsdienste, einschließlich der Bereitstellung von Telefonie, Internet und Mobilfunk.

Kenntnisse über Pflichtangaben eines Homepage-Betreibers (Impressum)

Ein Impressum ist für alle geschäftsmäßigen, also nicht rein privaten, Webseiten verpflichtend, um Transparenz zu gewährleisten und den rechtlichen Anforderungen zu entsprechen. Beinhaltet: Name und Anschrift, Kontaktmöglichkeit, Firmenbuchnummer, Vertretungsberechtigte, berufsrechtliche Regelungen

Kenntnisse über Pflichtangaben beim E-Mail-Verkehr von Unternehmen

Name des Unternehmens, Rechtsform, Firmenanschrift, Firmenbuchnummer, Vertretungsberechtigte, usw.

Kenntnisse über die gesetzliche Einhaltung von Bildschirmspausen

In Österreich fällt die Bildschirmarbeit unter die allgemeinen Arbeitszeitregelungen. Es gibt eine Verordnung für Bildschirmarbeitsplätze die Ergonomie am Arbeitsplatz sowie nach ca. 60 Minuten eine Tätigkeitswechsel empfiehlt.

6) Netzwerkdienste

Fachbegriffe Domain, Sub-Domain und Top-Level-Domain

Eine Domain ist der Hauptteil einer Webadresse (salzburg.at), Sub-Domain Unterteilung einer Domain (lbs4.salzburg.at), Top-Level-Domain (at) TLD: generische und länderspezifische

Kenntnis der Web-Protokolle HTTP und HTTPS

http (Hypertext Transfer Protocol) ist ein unverschlüsseltes Kommunikationsprotokoll (Port 80), um Daten zwischen Client und Server auszutauschen

HTTPS (Hypertext Transfer Protocol Secure) verwendet TLS zur Absicherung des Datenverkehrs (Zertifikat) Port 443

Funktionsprinzip eines Mail-Servers

Ein Mail-Server ist ein Computer, der für den Versand, Empfang und die Speicherung von E-Mails verantwortlich ist. Er verwendet spezielle Protokolle zur Kommunikation mit anderen Mail-Servern und den E-Mail-Clients der Benutzer.

Kenntnis des Mail-Protokolls POP3/POP3S

POP3 (Post Office Protocol 3) ermöglicht es einem Benutzer, E-Mails vom Mail-Server herunterzuladen und sie lokal auf seinem Gerät zu speichern Port 110

POP3S (Post Office Protocol 3 Secure) verschlüsselte Kommunikation für den Abruf von Mails

Kenntnis des Mail-Protokolls IMAP/IMAPS

Das IMAP-Protokoll- Port143 - (Internet Message Access Protocol) und die IMAPS-Version (sichere Variante von IMAP) - Port 993 - sind Protokolle, die zum Abrufen und Verwalten von E-Mails von einem Mail-Server verwendet werden. IMAP bietet im Vergleich zu POP3 eine größere Flexibilität und ermöglicht eine synchronisierte Verwaltung von E-Mails über mehrere Geräte hinweg.

Kenntnis des Mail-Protokolls SMTP/SMTPS

Das SMTP-Protokoll (Simple Mail Transfer Protocol Port 25) und seine sichere Variante SMTPS - Port 587 - sind für das Senden von E-Mails zwischen Mail-Servern verantwortlich.

Kenntnisse über FTP/FTPS

FTP (File Transfer Protocol Port 21) und FTPS (File Transfer Protocol Secure) sind Protokolle, die zum Übertragen von Dateien zwischen Computern über ein Netzwerk, insbesondere über das Internet, verwendet werden. Während FTP die grundlegende Form des Dateiübertragungsprotokolls darstellt, bietet FTPS zusätzliche Sicherheitsfunktionen durch die Integration von Verschlüsselung.

Kenntnisse über SSL

SSL (Secure Sockets Layer) ist ein veraltetes, aber bedeutendes Kryptografie-Protokoll, das ursprünglich entwickelt wurde, um eine sichere Verbindung zwischen einem Webbrowser und einem Webserver herzustellen. Es wird heutzutage größtenteils durch TLS (Transport Layer Security) ersetzt, das jedoch oft immer noch als "SSL" bezeichnet wird, da SSL in der Praxis der Begriff ist, der allgemein verwendet wird. TLS1.3

Fachbegriff Cloud-Computing

Cloud Computing bezeichnet die Bereitstellung von IT-Ressourcen (wie Rechenleistung, Speicherplatz, Anwendungen und Datenbanken) über das Internet. Anstatt diese Ressourcen lokal auf eigenen Servern oder Computern zu betreiben, können Unternehmen und Privatpersonen die benötigten Dienste von einem externen Anbieter (Cloud-Anbieter) über das Internet beziehen.

Kenntnisse über Private/Public/Hybrid-Cloud

In der Public Cloud werden Cloud-Dienste von einem Drittanbieter über das öffentliche Internet bereitgestellt und von mehreren Kunden (auch "Mietern" genannt) gemeinsam genutzt.

Eine Private Cloud ist eine Cloud-Infrastruktur, die nur von einem einzigen Unternehmen oder einer Organisation verwendet wird. Sie kann entweder intern (im eigenen Rechenzentrum) oder von einem Drittanbieter in einer externen Umgebung gehostet werden.

Die Hybrid-Cloud ist eine Kombination aus Public Cloud und Private Cloud, die es Unternehmen ermöglicht, ihre Ressourcen zwischen den beiden zu verschieben.

Fachbegriffe IaaS, PaaS, SaaS

IaaS bezeichnet die Bereitstellung von Grundlagen der IT-Infrastruktur (wie Rechenleistung, Speicher und Netzwerke) über das Internet (Amazon Web Services).

PaaS stellt eine Plattform bereit, die Entwicklern die Möglichkeit gibt, Anwendungen zu erstellen, zu entwickeln und zu betreiben, ohne sich um die zugrunde liegende Infrastruktur kümmern zu müssen (Azure App Services).

SaaS ist ein Cloud-Service-Modell, bei dem der Anbieter vollständig verwaltete Software über das Internet bereitstellt. Die Nutzer greifen über den Webbrowser auf die Anwendung zu, und der Anbieter übernimmt die gesamte Infrastruktur, Plattform und Softwareverwaltung (Office 365).

Beispiele für marktbekannte Cloud-Dienste

Kriterien und Voraussetzungen für den Einsatz von Cloud-Diensten

Sicherheitsanforderungen: Datenschutz, Verschlüsselung, ..

Verfügbarkeit und Zuverlässigkeit: Service Level Agreements (SLAs), Redundanz, Backup,

Kosten und Wirtschaftlichkeit: Preismodelle, Kostenoptimierung,

Skalierbarkeit: API-Unterstützung, Kompatibilität

Verwaltung und Kontrolle: Zugriffsmanagement, Monitoring

7) IT-Security und Betriebssicherheit

Kenntnisse über Gefahren von Viren, Würmern, Trojanern, Spyware, Hackern und Phishing

- Viren: Verbreiten sich und beschädigen Daten.
- Würmer: Selbstvermehrend, oft zur Netzwerksabotage.
- Trojaner: Tarnen sich als harmlose Software, öffnen Hintertüren.
- Spyware: Spioniert Nutzer aus, stiehlt Daten.
- Hacker: Dringen in Systeme ein, oft mit kriminellen Absichten.
- Phishing: Täuschung, um persönliche Daten zu stehlen.

Fachbegriff Zero-Day-Exploit

Ein Zero-Day-Exploit ist ein Angriff, der eine bisher unbekannte Sicherheitslücke ausnutzt, bevor der Hersteller sie schließen kann.

Kenntnisse über Einschränkungsmöglichkeiten bei Benutzerkonten

- Rechteverwaltung: Nur notwendige Berechtigungen vergeben.
- Gastkonten: Eingeschränkter Zugriff für temporäre Nutzer.
- Gruppenrichtlinien: Regeln für Nutzerverhalten festlegen.
- Softwareeinschränkungen: Installation und Nutzung kontrollieren.
- Netzwerkzugriff: Bestimmte Bereiche sperren.
- Zeitliche Beschränkung: Nutzungszeiten festlegen.
- Zwei-Faktor-Authentifizierung: Sicherheit erhöhen.

Funktion einer Software-Firewall

Eine Software-Firewall überwacht und kontrolliert den ein- und ausgehenden Datenverkehr auf einem Gerät. Sie blockiert unerwünschte Verbindungen und schützt vor Angriffen aus dem Internet.

Kenntnisse über Möglichkeiten Client-PCs vor Missbrauch zu schützen

- Antivirensoftware: Schädliche Programme erkennen und entfernen.
- Firewall: Netzwerkzugriffe kontrollieren und unerwünschte Verbindungen blockieren.
- Benutzerrechte einschränken: Nur notwendige Berechtigungen vergeben.
- Software-Updates: Sicherheitslücken schließen.
- Zwei-Faktor-Authentifizierung: Zugriff absichern.
- Sichere Passwörter: Komplexe, einzigartige Passwörter verwenden.
- Phishing-Schutz: Verdächtige Links und E-Mails meiden.
- Geräteverschlüsselung: Daten vor unbefugtem Zugriff schützen.

Kenntnisse über sichere Planung von Backups

- 3-2-1-Regel: 3 Kopien, 2 verschiedene Medien, 1 extern.
- Automatische Backups: Regelmäßige Sicherungen einplanen.
- Verschlüsselung: Schutz vor unbefugtem Zugriff.
- Versionierung: Mehrere Backup-Stände speichern.
- Speichermedien prüfen: Regelmäßige Tests auf Integrität.
- Offline-Backups: Schutz vor Ransomware.
- Cloud-Backups: Zusätzliche Sicherheit und Verfügbarkeit.

Kenntnisse über verschiedene Backup-Prinzipien

- Voll-Backup: Gesamte Daten sichern.
- Inkrementelles Backup: Nur geänderte Daten seit dem letzten Backup speichern.
- Differenzielles Backup: Alle Änderungen seit dem letzten Voll-Backup sichern.
- Spiegelung (Mirroring): Echtzeitkopie der Daten erstellen.
- Snapshots: Zustand eines Systems zu einem bestimmten Zeitpunkt sichern.
- Cloud-Backup: Externe Sicherung in der Cloud
- Türme von Hanoi: rotierende Backupstrategie mit mehreren Speichermedien nach einem festgelegten Muster

Das **Generationenprinzip** basiert auf der Idee, verschiedene Backup-Stufen zu erstellen, um Daten langfristig zu sichern und gleichzeitig Speicherplatz zu sparen. Es besteht aus drei Hauptkategorien:

- **Sohn-Backup:** Tägliche, inkrementelle oder differenzielle Backups, die nur die

zuletzt geänderten Daten sichern

- **Vater-Backup:** Wird wöchentlich erstellt und enthält eine Auswahl der wichtigsten Daten.
- **Großvater-Backup:** Wird selten durchgeführt, meist monatlich. Es enthält alle Daten und wird langfristig aufbewahrt.

Durch dieses System können Benutzer verschiedene Versionen ihrer Daten behalten und bei Bedarf auf ältere Versionen zugreifen.

Kenntnisse über Backup-Medien und deren richtiger Lagerung

Einige wichtige Backup-Medien und deren richtige Lagerung:

- Externe Festplatten: Schnell und zuverlässig, sollten an einem sicheren, kühlen und trockenen Ort gelagert werden, fern von extremen Temperaturen und Feuchtigkeit.
- USB-Sticks: Geeignet für kleinere Datenmengen, jedoch weniger langlebig. Lagern in einer schützenden Hülle und fern von magnetischen Feldern.
- Optische Medien (CD/DVD): Gute langfristige Aufbewahrung, sollten vor Kratzern und direkter Sonneneinstrahlung geschützt werden.
- Bandlaufwerke: Geeignet für große Datenmengen, erfordern eine kühle und trockene Lagerung, idealerweise in einem Brandschutzschrank.
- Cloud-Speicher: Sicher und zugänglich, aber vom Anbieter abhängig. Redundante Speicherung und regelmäßige Überprüfungen sind wichtig.

Lagerung: Medien sollten regelmäßig überprüft werden, um ihre Funktionsfähigkeit sicherzustellen und idealerweise an zwei oder mehr geografisch getrennten Orten aufbewahrt werden, um vor physischen Schäden wie Feuer oder Überschwemmungen geschützt zu sein.

8) Informatik und Gesellschaft

Fachbegriff Big-Data

Big Data bezeichnet große, komplexe Datenmengen, die mit traditionellen Methoden der Datenverarbeitung nicht effizient verarbeitet werden können. Sie zeichnen sich durch mindestens 3 Vs aus:

- **Volumen:** Große Datenmengen.
- **Varietät:** Unterschiedliche Datenarten (z.B. Texte, Bilder, Videos).
- **Velocity:** Schnelle Datenströme und -verarbeitung.

Big Data wird oft für **Datenanalyse**, **Mustererkennung** und **Vorhersagemodelle** genutzt.

Fachbegriff Web 2.0

Web 2.0 bezeichnet die Weiterentwicklung des Internets, bei der der Fokus auf Interaktivität, Benutzergenerierten Inhalten und sozialer Vernetzung liegt. Es betont die aktive Teilnahme der Nutzer, z.B. durch soziale Netzwerke, Wikis und Cloud-Dienste.

Fachbegriff Industrie 4.0

Industrie 4.0 bezeichnet die vierte industrielle Revolution, bei der intelligente und vernetzte Systeme in der Produktion zum Einsatz kommen. Sie basiert auf IoT (Internet of Things), Künstlicher Intelligenz, Big Data und Automatisierung, um die Effizienz und Flexibilität in Fertigungsprozessen zu steigern. Ziel ist die Smart Factory, in den Maschinen, Produkte und Systeme miteinander kommunizieren und autonom Entscheidungen treffen können.

Fachbegriff IoT

IoT (Internet of Things) bezeichnet die Vernetzung von Alltagsgeräten und Objekten über das Internet, sodass sie Daten sammeln, austauschen und miteinander kommunizieren können. Beispiele sind Smartphones, Smart Homes, Wearables und vernetzte Maschinen. Ziel ist es, durch diese Vernetzung Prozesse zu automatisieren und die Effizienz zu steigern.

Kenntnisse über Vor- und Nachteile bei Nutzung von Sprachassistenten

Vorteile von Sprachassistenten: Komfort, Zeitersparnis, Zugänglichkeit (beeinträchtigte Menschen), Smart Home Integration
Nachteile: Datenschutz, Fehleranfälligkeit, Sicherheitsrisiken, Abhängigkeit (Ausfälle)

Kenntnisse über e-Government, digitale Signatur und Handy-Signatur

e-Government:

e-Government bezeichnet die Nutzung digitaler Technologien und Internetdienste zur Erbringung von Verwaltungsleistungen und zur Kommunikation zwischen Bürgern, Unternehmen und staatlichen Stellen. Ziel ist die Effizienzsteigerung, Bürgerfreundlichkeit und Transparenz in der Verwaltung.

Digitale Signatur:

Eine digitale Signatur ist eine elektronische Methode, um die Authentizität und Integrität von digitalen Dokumenten zu bestätigen. Sie basiert auf Asymmetrischer Kryptografie und stellt sicher, dass das Dokument nicht verändert wurde und der Unterzeichner eindeutig identifizierbar ist.

Handy-Signatur:

Die Handy-Signatur ist eine mobile Lösung zur digitalen Unterschrift, die über das Handy funktioniert. Sie ermöglicht die rechtsverbindliche elektronische Unterzeichnung von Dokumenten und wird durch die österreichische Signaturkarte bzw. Apps wie ID Austria unterstützt. Sie bietet eine einfache, sichere Möglichkeit zur Authentifizierung und Unterschrift direkt über das Mobilgerät.

Schutzmöglichkeiten vor Cookie-Tracking und Cookieless-Tracking

Cookies ablehnen: In den meisten Browsern können Cookies manuell abgelehnt oder blockiert werden.

Private/Inkognito-Modi: Diese Modi speichern keine Cookies und löschen sie nach dem Schließen des Browsers.

Browsererweiterungen: Tools wie Ghostery, uBlock Origin (Werbeblocker) oder Privacy

Badger (Trackingblocker) blockieren Cookies und Tracking-Skripte.

Kenntnisse über die Gefahr von Identitätsdiebstahl

Identitätsdiebstahl ist eine ernsthafte Gefahr, bei der Kriminelle persönliche Informationen stehlen, um sich als jemand anderes auszugeben. Dies kann zu finanziellen Verlusten, Rufschädigung und rechtlichen Problemen führen. Rufschädigung, finanzielle Verluste, Verletzung der Privatsphäre

Fachbegriff Netzneutralität

Netzneutralität bezeichnet das Prinzip, dass Internetdienstanbieter (ISPs) den gesamten Internetverkehr gleich behandeln sollen, ohne bestimmte Daten, Dienste oder Anwendungen zu bevorzugen oder zu blockieren. Das bedeutet, dass keine Webseiten, Dienste oder Anwendungen aufgrund ihrer Art, Herkunft oder Zielsetzung langsamer gemacht oder bevorzugt werden dürfen. Ziel ist es, Chancengleichheit und Innovationsfreiheit im Internet zu gewährleisten.

Kenntnisse über Vor- und Nachteile bei Nutzung von biometrischen Daten

Inhalte von Unternehmensrichtlinien für Nutzung von sozialen Netzwerken

9) Ergonomische Gestaltung eines Arbeitsplatzes

Kenntnisse über die ergonomische Einrichtung eines Bildschirmarbeitsplatzes

Kenntnisse über den optimalen Aufstellungsort von Bildschirmen (Lichteinfall)

Kenntnisse der gesetzlichen Bestimmungen von Pausen bei Bildschirmarbeit

Kenntnisse über die ideale Höhe von Tisch/Tastatur, Bildschirmoberkante und Bildschirmabstand zum Benutzer

Kenntnisse über Schutzmaßnahmen zur Vorbeugung körperlicher Schäden bei sitzender Tätigkeit

Kenntnisse über körperliche Entspannungsübungen bei sitzender Tätigkeit

10) Fachberatung, Planung

Führen von fachspezifischen Verkaufsgesprächen, Produktberatung

1. Führen von fachspezifischen Verkaufsgesprächen:

Dabei geht es darum, ein Verkaufsgespräch gezielt auf die Branche und das Produkt abzustimmen. Man nutzt Fachwissen, um dem Kunden kompetent zu begegnen, auf individuelle Bedürfnisse einzugehen und passende Lösungen anzubieten - mit dem Ziel, Vertrauen aufzubauen und den Verkauf abzuschließen.

2. Produktberatung:

Produktberatung bedeutet, den Kunden über die Eigenschaften, Vorteile und Anwendungsmöglichkeiten eines Produkts zu informieren. Dabei steht im Fokus, das passende Produkt für den Bedarf des Kunden zu finden, Fragen zu beantworten und eventuelle Alternativen aufzuzeigen.

Kompetenz, technische Zusammenhänge beratend erklären zu können

Fachwissen verständlich und kundenorientiert vermitteln.

Beratung und Erstellen kundenorientierter Softwarelösungen

Das bedeutet, gemeinsam mit dem Kunden Anforderungen zu analysieren, passende Softwarelösungen zu planen und umzusetzen.

Dabei steht der individuelle Bedarf des Kunden im Mittelpunkt - die Lösung soll funktional, benutzerfreundlich und effizient sein.

Kenntnisse über richtigen Umgang bei Reklamationen

Das heißt, Reklamationen professionell, freundlich und lösungsorientiert zu bearbeiten. Wichtig sind dabei: dem Kunden zuhören, Verständnis zeigen, ruhig bleiben und schnell eine faire Lösung anbieten - immer mit dem Ziel, die Kundenzufriedenheit wiederherzustellen.

Richtiger Kundenumgang bei folgenreichen technischen Problemen

In solchen Situationen ist ein besonders professionelles und empathisches Auftreten gefragt. Wichtig ist: schnell reagieren, ehrlich kommunizieren, Verantwortung übernehmen, transparente Lösungen anbieten und den Kunden regelmäßig über den Stand informieren - so zeigt man Zuverlässigkeit trotz Krise

11) Informatik

Fachbegriff Informatik

Informatik ist die Wissenschaft von der systematischen Verarbeitung, Speicherung und Übertragung von Informationen - insbesondere mithilfe von Computern. Sie umfasst Bereiche wie Softwareentwicklung, Netzwerke, Datenbanken, künstliche Intelligenz und vieles mehr.

Typen von Webseiten (statische, dynamische Webseiten)

Statische Webseiten:

Der Inhalt ist fest im HTML-Code gespeichert und ändert sich nur, wenn er manuell angepasst wird. Jeder Besucher sieht denselben Inhalt.

Beispiel: einfache Firmen-Homepage oder digitale Visitenkarte.

Dynamische Webseiten:

Der Inhalt wird bei jedem Aufruf neu generiert - oft abhängig von Benutzereingaben oder Datenbanken. Sie reagieren interaktiv und sind flexibel.

Beispiel: Online-Shops, soziale Netzwerke, Newsportale.

Fachbegriffe Weblog, Webshop, Web-Plattform

1. Weblog (Blog):

Ein Weblog ist eine Art Online-Tagebuch oder Magazin, in dem regelmäßig Beiträge (Posts) zu bestimmten Themen veröffentlicht werden. Leser können oft kommentieren.

Beispiel: Reiseblog, Technikblog.

2. Webshop:

Ein Webshop ist ein Online-Geschäft, in dem Produkte oder Dienstleistungen über das Internet verkauft werden. Nutzer können Artikel ansehen, in den Warenkorb legen und online bezahlen.

Beispiel: Amazon, Zalando.

3.

Web-Plattform:

Eine Web-Plattform ist eine digitale Umgebung, die verschiedene Dienste oder Funktionen bereitstellt - z. B. für Kommunikation, Handel oder Zusammenarbeit. Oft bringen Plattformen Anbieter und Nutzer zusammen.

Beispiel: YouTube, eBay, LinkedIn.

Auszeichnungssprachen HTML, XML - Fachbegriff und Einsatzgebiet

1. HTML (HyperText Markup Language):

Fachbegriff: Auszeichnungssprache zur Strukturierung von Webseiten.

Einsatzgebiet: HTML wird verwendet, um Inhalte wie Texte, Bilder, Links und Überschriften auf Webseiten darzustellen und zu strukturieren. Es ist die Basis jeder Webseite im Internet.

2. XML (eXtensible Markup Language):

Fachbegriff: Auszeichnungssprache zur Speicherung und zum Austausch von Daten in strukturierter Form.

Einsatzgebiet: XML wird genutzt, um Daten zwischen verschiedenen Systemen oder Anwendungen auszutauschen - z. B. in Webservices, Konfigurationsdateien oder Datenbanken.

Kurz:

XML = Sprache für den strukturierten Datenaustausch.

Kenntnisse über das HTML5-Grundgerüst mit den wichtigsten Bestandteilen

<!DOCTYPE html>

Dieser Tag gibt an, dass es sich um ein HTML5-Dokument handelt. Er ist wichtig, um sicherzustellen, dass der Browser die Seite korrekt darstellt.

<html lang="de">

Das <html>-Tag umschließt den gesamten HTML-Inhalt und der lang="de"-Attribut gibt an, dass die Seite in deutscher Sprache ist.

<head>

Der <head>-Bereich enthält Meta-Informationen über die Seite, die nicht direkt sichtbar sind. Hier werden zum Beispiel der Zeichensatz und der Titel der Seite festgelegt.

- **<meta charset="UTF-8">:** Legt die Zeichencodierung auf UTF-8 fest, was für die richtige Darstellung von Sonderzeichen sorgt.
- **<title>:** Hier wird der Titel der Seite definiert, der im Browser-Tab angezeigt wird.

<body>

Der <body>-Bereich enthält den sichtbaren Inhalt der Webseite - alles, was der Nutzer auf der Seite sieht, wie Texte, Bilder und Links.

Fachbegriff Meta-Element/Metadaten

Meta-Element (Meta-Tag):

Ein Meta-Element ist ein HTML-Tag, das im **<head>-Bereich** einer Webseite eingebunden wird und **Metadaten** über die Seite bereitstellt. Metadaten sind Informationen, die nicht direkt auf der Seite angezeigt werden, aber für die Suchmaschinenoptimierung (SEO), das Laden der Seite oder die Anzeige im Browser wichtig sind.

Metadaten sind Daten, die andere Daten beschreiben oder erklären. In Bezug auf Webseiten umfassen sie Informationen wie den Zeichensatz (charset), die Beschreibung der Seite (description), Schlüsselwörter (keywords), die Autorenschaft (author) und vieles mehr.

Fachbegriff SEO und Maßnahmen

SEO steht für Search Engine Optimization (Suchmaschinenoptimierung) und bezeichnet den Prozess, eine Webseite so zu gestalten und zu optimieren, dass sie in den organischen (unbezahlten) Suchergebnissen von Suchmaschinen wie Google, Bing oder Yahoo besser platziert wird. Ziel ist es, die Sichtbarkeit und Reichweite einer Webseite zu erhöhen, um mehr Traffic (Besucher) zu generieren.

Fachbegriff Cascading StyleSheets und deren Einsatz

CSS steht für **Cascading Style Sheets** (kaskadierende Formatvorlagen) und ist eine Stylesheet-Sprache, die dazu verwendet wird, das **Design und Layout** von HTML-Dokumenten zu definieren. Mit CSS können Farben, Schriftarten, Abstände, Layouts und andere visuelle Aspekte einer Webseite gestaltet werden, ohne den Inhalt in HTML zu verändern.

Scripting (clientseitiges Scripting, serverseitiges Scripting)

Definition:

Clientseitiges Scripting bezeichnet das Ausführen von Skripten auf dem **Rechner des Benutzers (Client)**, also direkt im Webbrowser. Der Code wird heruntergeladen und im Browser des Nutzers ausgeführt.

Einsatzgebiete:

- Interaktive Elemente wie Formulare, Dropdown-Menüs, Popups und Animationen.
- Validierung von Benutzereingaben ohne erneutes Laden der Seite.
- Dynamisches Nachladen von Inhalten (z. B. mit JavaScript und AJAX).

Definition:

Serverseitiges Scripting bedeutet, dass der Code auf dem **Webserver** ausgeführt wird, bevor die Webseite an den Benutzer (Client) gesendet wird. Der Server verarbeitet Anfragen, führt Logik aus und gibt die Ergebnisse an den Client zurück.

Einsatzgebiete:

- Benutzeranmeldung und -authentifizierung.
- Datenbankzugriffe, um personalisierte Inhalte oder dynamische Webseiten zu erzeugen.
- Generierung von HTML-Seiten auf Grundlage von Benutzereingaben oder Datenbankinformationen.

PHP , Node.js, ASP.NET, .

Software zum Erstellen und Betrachten von Webseiten (Code-Editoren, Web-Browser, FTP-Programme, Grafikprogramme, Serversoftware)

Code-Editoren sind Programme, mit denen Entwickler den HTML-, CSS-, JavaScript- und anderen Code für Webseiten schreiben können. Sie bieten Funktionen wie Syntax-Hervorhebung, Autovervollständigung und Fehlererkennung.

Web-Browser werden verwendet, um Webseiten anzuzeigen. Sie interpretieren HTML, CSS und JavaScript, um Webseiten für den Endbenutzer darzustellen. Sie bieten auch Entwicklertools, mit denen Webseiten analysiert und debuggt werden können.

FTP-Programme (File Transfer Protocol) werden verwendet, um Dateien zwischen dem lokalen Computer und einem Webserver zu übertragen. Sie sind besonders nützlich, um die entwickelten Webseiten auf einen Server hochzuladen.

Grafikprogramme werden genutzt, um visuelle Elemente für Webseiten zu erstellen (wie Logos, Icons, Bilder, etc.). Sie sind ein wichtiger Bestandteil des Webdesigns.

Serversoftware wird benötigt, um Webseiten auf einem Webserver bereitzustellen und dynamische Funktionen zu ermöglichen (z. B. bei Verwendung von Datenbanken oder serverseitigem Scripting)

Fachbegriff CMS (Einsatzgebiet, notwendige Voraussetzungen, existierende Systeme am Markt)

Ein Content Management System (CMS) ist eine Software, mit der Benutzer Inhalte für eine Webseite erstellen, verwalten und bearbeiten können, ohne tiefgehende Programmierkenntnisse zu benötigen. CMS bieten eine benutzerfreundliche Oberfläche, die das Erstellen und Bearbeiten von Texten, Bildern, Videos und anderen Medien auf einer Webseite ermöglicht.

Unterschied LIFO/FIFO-Prinzip

Definition:

Beim LIFO-Prinzip wird das zuletzt hinzugefügte Element zuerst entnommen. Es funktioniert ähnlich wie ein Stapelturm: Das letzte Objekt, das auf den Stapel gelegt wird, ist das erste, das wieder entfernt wird.

Einsatzgebiete:

Stapel (Stack): In der Informatik wird LIFO häufig in Stack-Datenstrukturen verwendet. Hier werden Operationen wie „Push“ (Hinzufügen) und „Pop“ (Entnehmen) durchgeführt.

Lagerhaltung: Manchmal wird LIFO in Lagerverwaltungssystemen verwendet, bei denen das neueste Produkt (z. B. eine Lieferung) zuerst verkauft oder genutzt wird, wie bei verderblichen Waren oder bei der Verarbeitung von Aufträgen.

Fachbegriffe Stack und Queue

In der Informatik sind **Stack** und **Queue Datenstrukturen**, die zur Verwaltung von Elementen in einer bestimmten Reihenfolge verwendet werden. Sie haben jeweils unterschiedliche Prinzipien, wie Daten hinzugefügt und entnommen werden.

Fachbegriff Userinterface (Arten, Regeln für Entwurf, Gestaltungshilfen/Toolkits/Frameworks)

Ein **User Interface (UI)** ist die Schnittstelle zwischen einem Benutzer und einem Computerprogramm oder einer Website. Es umfasst alle Elemente, mit denen der Benutzer interagiert, wie Schaltflächen, Menüs, Fenster und Eingabefelder. Das Ziel eines UI-Designs ist es, die Interaktion so benutzerfreundlich und effizient wie möglich zu gestalten (Benutzerzentriert, Konsistenz, Minimierung, Reaktionszeit (Systemstatus), Barrierefreiheit, ...)

Fachbegriff Zeichencodierung (ASCII, ISO-Latin, Unicode, ... - Unterschiede und Verwendung)

ASCII ist eine Zeichencodierung, die ursprünglich in den 1960er Jahren entwickelt wurde und **7-Bit-Codierung** verwendet. Sie stellt **128 Zeichen** dar, darunter englische Buchstaben (Groß- und Kleinbuchstaben), Zahlen, Satzzeichen und Steuerzeichen (wie Zeilenumbruch oder Wagenrücklauf).

ISO-Latin-1 (oder **ISO 8859-1**) ist eine **8-Bit-Codierung**, die eine erweiterte Zeichenmenge im Vergleich zu ASCII bietet. Sie stellt **256 Zeichen** dar und umfasst alle Zeichen von ASCII sowie zusätzliche Zeichen für westeuropäische Sprachen (z. B. akzentuierte Buchstaben wie é, à, ñ, usw.).

Unicode ist eine **universelle Zeichencodierung**, die darauf abzielt, alle Schriftzeichen aus allen Sprachen der Welt in einem einheitlichen System darzustellen. Unicode verwendet verschiedene **Codepoints** (Nummerierungen) für Zeichen und kann mehrere **Bitlängen** verwenden (z. B. 16-Bit, 32-Bit), um eine größere Anzahl von Zeichen darzustellen.

Standards ANSI, ISO, IEEE

Das **American National Standards Institute (ANSI)** ist eine **US-amerikanische Organisation**, die für die Festlegung von **nationalen Standards** verantwortlich ist. Es koordiniert die Entwicklung von Standards in den USA und arbeitet mit internationalen Gremien zusammen, um global einheitliche Standards zu schaffen.

ISO (International Organization for Standardization) ist eine **weltweit tätige Standardisierungsorganisation**, die **internationale Standards** für alle Arten von Produkten, Dienstleistungen und Systemen festlegt. Die Organisation hat über 160 Mitgliedsländer.

Das **IEEE (Institute of Electrical and Electronics Engineers)** ist eine **weltweite Organisation**, die Standards in den Bereichen **Elektrotechnik, Elektronik, Telekommunikation, Computertechnik** und verwandten Feldern entwickelt.

Fachbegriff Frame

In der Webentwicklung bezeichnet **Frame** ein **HTML-Element**, das es ermöglicht, den Inhalt einer Webseite in mehrere, separate Bereiche (Fenster) zu unterteilen.

Fachbegriff Webservices (verteiltes System für heterogene Systeme, ...)

Ein **Webservice** ist eine **Software-Anwendung**, die über das **Internet** oder ein **Intranet** zugänglich ist und standardisierte **Kommunikationsprotokolle** verwendet, um Daten und Funktionen zwischen verschiedenen, oft heterogenen Systemen auszutauschen.

Webservices ermöglichen die Interoperabilität zwischen verschiedenen Anwendungen, die auf unterschiedlichen Plattformen und in unterschiedlichen Programmiersprachen laufen.

Kenntnisse über Standards (SOAP, WSDL, ...)

SOAP ist ein **Protokoll**, das für den Austausch von strukturierten Informationen zwischen Computern über das **Internet** verwendet wird. SOAP basiert auf **XML** und verwendet standardisierte Nachrichtenformate für die Kommunikation zwischen Anwendungen.

WSDL ist eine **XML-basierte Sprache**, die dazu verwendet wird, Webservices zu **beschreiben** und zu definieren. Mit WSDL können Entwickler den Webservice selbst sowie seine verfügbaren Methoden, Parameter, Rückgabewerte und Transportprotokolle beschreiben

Fachbegriff Rest API

Eine **REST API** (Representational State Transfer Application Programming Interface) ist eine **Schnittstelle** für die Kommunikation zwischen verschiedenen Softwareanwendungen über das **Web**, die auf den Prinzipien von **REST** basiert. REST ist ein Architekturstil, der die Kommunikation über das **HTTP-Protokoll** nutzt und auf Einfachheit und Skalierbarkeit ausgelegt ist.

Fachbegriff JSON

JSON (JavaScript Object Notation) ist ein leichtgewichtiges, **textbasiertes** Format zur **Datenrepräsentation**. Es wird verwendet, um strukturierte Daten in einem einfach lesbaren und schreibbaren Format zu speichern und zwischen Systemen zu übertragen. JSON basiert auf einer Teilmenge der **JavaScript-Syntax** und ist sowohl von Menschen als auch von Maschinen gut lesbar.

Fachbegriff Agile Softwareentwicklung

Die **agile Softwareentwicklung** ist ein Ansatz zur Softwareentwicklung, der auf Flexibilität, Zusammenarbeit und kontinuierlicher Verbesserung basiert. Sie fokussiert sich darauf, Software iterativ und inkrementell zu entwickeln, anstatt sie in großen, starren Phasen zu planen. Der Begriff „agil“ bezieht sich auf die Fähigkeit, schnell und flexibel auf Veränderungen zu reagieren, sowohl bei den Anforderungen als auch bei den technologischen Herausforderungen.

Fachbegriff Reaktive Programmierung

Die **Reaktive Programmierung (Reactive Programming)** ist ein Paradigma der Programmierung, das auf der asynchronen und ereignisgesteuerten Verarbeitung von Datenströmen basiert. Sie zielt darauf ab, dass Programme auf **Datenänderungen oder Ereignisse** reagieren und automatisch darauf reagieren, ohne dass explizite Anweisungen oder Polling benötigt werden.

Kenntnisse über Frameworks

Ein **Framework** ist eine Sammlung von vorgefertigten Softwarekomponenten, Bibliotheken, Tools und Best Practices, die Entwicklern helfen, Anwendungen schneller und effizienter zu erstellen. Es bietet eine grundlegende Struktur oder „Gerüst“, auf dem Entwickler ihre spezifischen Anwendungen aufbauen können, ohne das Rad jedes Mal neu erfinden zu müssen.

Einsatzgebiete Angular JS

AngularJS ist ein populäres **JavaScript-Framework**, das ursprünglich von Google entwickelt wurde, um dynamische, interaktive Webanwendungen zu erstellen. Es verwendet eine **Model-View-Controller (MVC)**-Architektur und bietet zahlreiche Tools und Funktionen, um moderne Single-Page-Applications (SPAs) effizient zu entwickeln.

Einsatzgebiete Bootstrap

Bootstrap ist ein weit verbreitetes, **Open-Source-Framework** für die Entwicklung von **responsiven** und **mobilfreundlichen Webanwendungen**. Es wurde ursprünglich von Twitter entwickelt und bietet eine Sammlung von **CSS-, JavaScript- und HTML-Komponenten**, die Entwicklern dabei helfen, schnell ansprechende und funktionale Webseiten und Anwendungen zu erstellen.

Einsatzgebiet jQuery

jQuery ist eine weit verbreitete JavaScript-Bibliothek, die ursprünglich entwickelt wurde, um die Arbeit mit HTML-Dokumenten, Ereignissen, Animationen und AJAX-Anfragen zu vereinfachen. Sie wurde so konzipiert, dass sie die Interaktion mit dem DOM (Document Object Model) in Webanwendungen vereinfacht und die Code-Entwicklung für den Browser flexibler und plattformübergreifend macht.

Kenntnisse über den Zugriff PHP auf MySQL-Datenbank (Dienste Server/Client)

Der Zugriff auf eine MySQL-Datenbank mit PHP ist ein grundlegender Bestandteil vieler dynamischer Webanwendungen. MySQL ist ein weit verbreitetes relationales Datenbanksystem, und PHP ist eine weit verbreitete Server-seitige Programmiersprache, die es ermöglicht, Webanwendungen dynamisch mit Daten zu versorgen. Der Zugriff erfolgt über spezifische Datenbankdienste auf der Server-Seite und wird durch die Kommunikation zwischen PHP und MySQL ermöglicht (MySQLi, PDO)

Fachbegriff Multitasking

Multitasking bezeichnet die Fähigkeit eines Computersystems oder eines Betriebssystems, mehrere Prozesse oder Aufgaben gleichzeitig zu verwalten und auszuführen. Dabei können verschiedene Programme oder Anwendungsprozesse parallel oder in schnellen Wechseln (zeitlich versetzt) ausgeführt werden, sodass der Benutzer den Eindruck hat, dass mehrere Aufgaben gleichzeitig erledigt werden (Preemptive - wechseln, Kooperativ - Programm wechseln)

Kenntnisse über mobile Webseiten/Optimierung für Smartphones

Die **Optimierung von Webseiten für Smartphones** ist ein wesentlicher Bestandteil der modernen Webentwicklung. Da immer mehr Benutzer über mobile Geräte wie Smartphones und Tablets auf das Internet zugreifen, ist es entscheidend, dass Webseiten auf diesen Geräten **benutzerfreundlich**, **schnell** und **effizient** funktionieren Responsive, Ladegeschwindigkeit, Navigation, ...)

Fachbegriff Responsive Webdesign, Umsetzung

Responsive Webdesign (RWD) bezeichnet einen Ansatz in der Webentwicklung, bei dem Webseiten so gestaltet werden, dass sie sich automatisch an die Bildschirmgröße und das Gerät des Benutzers anpassen. Dies bedeutet, dass eine Webseite sowohl auf großen Desktops als auch auf kleinen Smartphones und Tablets benutzerfreundlich und funktionsfähig bleibt, ohne dass separate Versionen für verschiedene Geräte erstellt werden müssen.

Kenntnisse über Konzept Mobile First

Mobile First ist ein Design- und Entwicklungsansatz in der Webentwicklung, bei dem eine Webseite oder Anwendung **zuerst für mobile Endgeräte** (Smartphones und Tablets) entworfen wird, bevor sie für größere Bildschirme (wie Tablets und Desktops) angepasst wird. Der Grundgedanke hinter diesem Ansatz ist, dass immer mehr Benutzer mit mobilen Geräten auf das Internet zugreifen, weshalb die mobile Benutzererfahrung im Zentrum der Gestaltung steht.

Kenntnisse über aktuelle Programmiersprachen

Aktuelle **Programmierersprachen** bieten unterschiedliche Funktionen und Vorteile, die je nach Einsatzzweck und Technologie-Stack verwendet werden. Hier sind einige der bekanntesten und am häufigsten genutzten Programmiersprachen in der heutigen Softwareentwicklung

Kenntnisse über Programmiersprachen für mobile Anwendungen/Internet

Es gibt verschiedene **Programmierersprachen** und **Frameworks**, die speziell für die Entwicklung von **mobilen Anwendungen** und **Internet-basierten Anwendungen** geeignet sind. SWIFT (iOS), Kotlin (Android), React Native (iOS und Android)

Kenntnisse über die Anwendung von JAVA-Technologien im Web (Servlets, Java-Server-Pages)

Java-Technologien im Web bieten leistungsstarke Lösungen für die Entwicklung dynamischer und interaktiver Webanwendungen. Zwei der bekanntesten Technologien, die für die Entwicklung von Webanwendungen mit Java verwendet werden, sind **Servlets** und **JavaServer Pages (JSP)**.

Grundkenntnisse über die Anwendung der .NET-Technologien im Web (ASP.NET)

ASP.NET ist ein Open-Source-Webframework von Microsoft für die Entwicklung von Webanwendungen und -diensten. Es ist eine der bekanntesten Technologien für die Erstellung von dynamischen und skalierbaren Webanwendungen auf der **.NET-Plattform**

Prinzipien der Softwareentwicklung: KISS, DRY

KISS ist ein Prinzip, das besagt, dass Lösungen so einfach wie möglich gehalten werden sollten. Es ermutigt Entwickler, unnötige Komplexität zu vermeiden und den Code klar, verständlich und wartbar zu gestalten.

DRY ist ein Prinzip, das darauf abzielt, Code-Wiederholungen zu vermeiden. Es fordert Entwickler auf, logische Einheiten von Code nur einmal zu definieren und diese dann bei Bedarf wiederzuverwenden. Wiederholungen führen zu unübersichtlichem Code und machen spätere Änderungen schwieriger und fehleranfälliger.

Kenntnisse über Coding-Standards/Code-Konventionen

Coding-Standards sind eine Sammlung von Best Practices, die Entwicklern helfen, Code zu schreiben, der sowohl funktional als auch qualitativ hochwertig ist. Sie umfassen Regeln zu Namensgebung, Strukturierung des Codes, Kommentierung, Fehlerbehandlung, Testbarkeit und mehr (Lesbarkeit, Wartbarkeit, Fehlerreduktion, ...). Code-Konvention (Variablen, Namen, Klassen, ...)

Fachbegriff Cross Plattform Entwicklung

Cross-Plattform-Entwicklung bezeichnet die Praxis, Software zu entwickeln, die auf mehreren Betriebssystemen oder Plattformen ohne signifikante Änderungen am Code ausgeführt werden kann. Ziel ist es, eine Anwendung einmal zu entwickeln und sie dann auf verschiedenen Geräten oder Betriebssystemen (wie Windows, macOS, Linux, Android, iOS und Web) lauffähig zu machen.

Fachbegriff Continuous Integration (CI)

Continuous Integration (CI) ist eine Softwareentwicklungspraktik, bei der Codeänderungen regelmäßig und automatisch in das gemeinsame Code-Repository integriert werden. Das Ziel von CI ist es, häufige, kleine Änderungen am Code zu integrieren, um Konflikte frühzeitig zu erkennen und zu beheben, anstatt große Änderungen über längere Zeiträume hinweg zu sammeln.

Fachbegriff Continuous Delivery bzw. Continuous Deployment (CD)

Continuous Delivery (CD) und Continuous Deployment (CD) sind eng miteinander verwandte Praktiken in der Softwareentwicklung, die auf der Idee basieren, Änderungen schnell und zuverlässig in die Produktion zu bringen. Beide Konzepte sind Erweiterungen der Continuous Integration (CI)-Praktik, aber sie gehen einen Schritt weiter, indem sie den Prozess der Bereitstellung und Veröffentlichung von Software automatisieren.

CI/CD Vorgaben bei der Applikationsentwicklung

Continuous Integration stellt sicher, dass der Code regelmäßig integriert und überprüft wird. Hier sind einige wichtige Vorgaben und Best Practices:

Häufige Commits, automatische Builds, automatisierte Tests, clean Builds, konsistentes Feedback

Continuous Delivery ermöglicht es, Software in einer staging-Umgebung bereitzustellen, sodass sie für die Produktion bereit ist, aber noch nicht automatisch ausgerollt wird. Es handelt sich um eine Erweiterung von CI, die den Code durch alle notwendigen Test- und Qualitätssicherungsprozesse führt, um ihn bereit für die Produktion zu machen.

Automatische Bereitstellung, Feature Flags, manuelle Freigabe für Produktion

12) Projektmanagement

Fachbegriff Projektmanagement

Projektmanagement bezeichnet den Prozess der Planung, Organisation, Durchführung und Überwachung von Projekten, um festgelegte Ziele innerhalb eines vorgegebenen Zeitrahmens und Budgets zu erreichen. Es umfasst alle Aktivitäten, die notwendig sind, um ein Projekt erfolgreich zu steuern, von der Initiierung über die Planung und Durchführung bis hin zum Abschluss.

Definition von Projekten

Projekt: Ein Projekt ist eine einmalige, zeitlich begrenzte und zielgerichtete Aufgabe oder ein Vorhaben, das dazu dient, ein bestimmtes Ziel oder Ergebnis zu erreichen. Ein Projekt hat definierte Anfangs- und Endpunkte sowie festgelegte Ziele, Ressourcen und oft auch einen spezifischen Budgetrahmen.

Fachbegriff Pflichtenheft und notwendiger Inhalt

Pflichtenheft: Ein Pflichtenheft ist ein Dokument im Projektmanagement und in der Softwareentwicklung, das die Anforderungen, Erwartungen und spezifischen Ziele eines Projekts aus Sicht des Auftraggebers detailliert beschreibt.

Einleitung, detaillierte Anforderungen, Daten und Informationen, Benutzerschnittstellen, Leistungsanforderungen, usw

Fachbegriff Lastenheft und notwendiger Inhalt

Lastenheft: Ein Lastenheft ist ein Dokument im Projektmanagement und in der Produktentwicklung, das die Anforderungen und Erwartungen des Auftraggebers an ein Projekt oder Produkt beschreibt. Es legt fest, was erreicht werden soll, ohne jedoch festzulegen, wie dies technisch umgesetzt werden soll.

Kenntnisse über Spannungsfelder in einem Projekt

Spannungsfelder in einem Projekt entstehen durch unterschiedliche, oft widersprüchliche Anforderungen, Erwartungen oder Interessen von verschiedenen Projektbeteiligten. Diese Spannungsfelder können den Erfolg eines Projekts gefährden, wenn sie nicht richtig gemanagt werden.

Kenntnisse über den Fachbegriff Primäres Projektziel

Primäres Projektziel: Das primäre Projektziel bezeichnet das Hauptziel oder den Kernzweck eines Projekts, der die gesamte Ausrichtung und den Erfolg des Projekts bestimmt. Es ist der zentrale Punkt, auf den alle Projektaktivitäten ausgerichtet sind und zu dessen Erreichung alle Ressourcen und Anstrengungen mobilisiert werden.

Kenntnisse über Vor- und Nachteile einer Projektorganisation

Projektorganisation bezeichnet die Struktur und die Art und Weise, wie ein Projekt innerhalb einer Organisation organisiert wird. Dabei werden Aufgaben, Zuständigkeiten und Ressourcen so verteilt, dass das Projektziel effizient erreicht wird.

Vorteil: Fokussierung, Klarheit der Verantwortlichen, effiziente Nutzung Ressourcen

Nachteil: Ressourcenintensiv (Teams), doppelte Strukturen, Konflikte zwischen Linie und Projekt

Ziel einer Projektdokumentation

Ziel einer Projektdokumentation ist es, alle relevanten Informationen eines Projekts systematisch zu erfassen, zu speichern und zu organisieren, sodass das Projekt jederzeit nachvollziehbar, überprüfbar und transparent bleibt. Eine Projektdokumentation dient sowohl als Kommunikationsmittel innerhalb des Projektteams als auch als Nachweis und Referenz für zukünftige Projekte oder Audits.

Fachbegriff Struktogramm

Ein **Struktogramm** (auch **Nassi-Shneiderman-Diagramm** genannt) ist ein grafisches Werkzeug zur Darstellung von Algorithmen oder Programmlogik. Es wird verwendet, um die Struktur und den Ablauf eines Programms übersichtlich darzustellen. Besonders in der Softwareentwicklung und Programmierung wird das Struktogramm genutzt, um komplexe Algorithmen in einer klaren und nachvollziehbaren Weise zu visualisieren.

Fachbegriff Ablaufdiagramm (Flowchart)

Ein **Ablaufdiagramm** (auch **Flowchart** genannt) ist ein grafisches Werkzeug zur Darstellung von Prozessen, Algorithmen oder Arbeitsabläufen. Es wird verwendet, um die Reihenfolge von Aktionen und Entscheidungen in einem System oder Programm visuell darzustellen. Ablaufdiagramme helfen dabei, komplexe Prozesse in übersichtliche, leicht verständliche Schritte zu unterteilen.

Kenntnisse über wesentliche Schritte einer Projektplanung

Die **Projektplanung** ist eine entscheidende Phase im Projektmanagement, die den gesamten Verlauf eines Projekts bestimmt. Sie umfasst eine Reihe wesentlicher Schritte, um sicherzustellen, dass das Projekt effizient, im Zeitrahmen und im Budgetrahmen erfolgreich abgeschlossen wird.

Ziele definieren, Projektumfang festlegen, PSP erstellen, Zeitplanung und Meilensteine, Ressourcenplanung, Budgetplanung, Risikomanagement, Teamerstellung, Kommunikationsplan, Projektüberwachung, Abschluss

Kenntnisse über Eigenschaften eines Projektleiters

Ein Projektleiter spielt eine zentrale Rolle im erfolgreichen Abschluss eines Projekts. Die Eigenschaften eines guten Projektleiters sind entscheidend für den Projekterfolg, da er die Verantwortung für die Planung, Durchführung und den Abschluss des Projekts trägt.

Aufgaben eines Projektleiters

Führungsaufgaben, Kommunikationswege definieren,

Kenntnisse über Dokumentationen eines Projektes

Projektinitiierung: Definieren des Projekts, Kick-off-Meeting, Aufbau der Projektorganisation.

Projektplanung: Erstellen des Projektplans, Festlegung des Projektumfangs, Ressourcenplanung.

Teamaufbau und -management: Teamzusammenstellung, Definition von Rollen und Verantwortlichkeiten, Motivation.

Projektüberwachung und -steuerung: Fortschrittsüberwachung, Problemlösung, Anpassung des Plans.

Risikomanagement: Identifikation und Management von Risiken.

Kommunikation und Stakeholder-Management: Informationsaustausch, Stakeholder-Pflege, Konfliktlösung.

Qualitätsmanagement: Festlegung und Überwachung von Qualitätsstandards.

Budget- und Kostenmanagement: Budgetplanung und -überwachung.

Änderungsmanagement: Verwaltung von Änderungen im Projekt.

Projektabschluss: Abnahme, Dokumentation, Lessons Learned.

Abschlussgespräche und Übergabe: Übergabe der Ergebnisse, Nachbereitung.

Fachbegriff Projektauftrag

Der **Projektauftrag** ist ein formelles Dokument, das die Grundlage für ein Projekt bildet. Er dient als **Vertrag** zwischen dem Auftraggeber und dem Projektteam bzw. dem Projektleiter und definiert die wichtigsten Rahmenbedingungen und Ziele eines Projekts.

Fachbegriff Projektstrukturplan

Ein **Projektstrukturplan (PSP)** ist ein Hilfsmittel im Projektmanagement, das die **hierarchische Gliederung** eines Projekts in verschiedene Teilprojekte und Arbeitspakete darstellt. Der PSP dient dazu, das Projekt in gut strukturierte Einheiten zu zerlegen, sodass der Projektumfang und die Arbeitsschritte systematisch erfasst werden.

Fachbegriff Arbeitspaket

Ein **Arbeitspaket** ist ein **konkretes, abgegrenztes und eigenständiges Arbeitselement** innerhalb eines Projekts, das im Rahmen eines Projektstrukturplans (PSP) definiert wird. Es stellt eine spezifische Aufgabe oder eine Reihe von Aufgaben dar, die zusammen ein definiertes **Ziel** oder **Deliverable** (Ergebnis) liefern.

Fachbegriff Meilenstein

Ein **Meilenstein** ist ein **wichtiger Zeitpunkt** oder ein **Zwischenergebnis** im Verlauf eines Projekts, das eine besondere Bedeutung hat und den Fortschritt oder die Erreichung eines bestimmten Ziels markiert. Meilensteine dienen als **Kontrollpunkte**, um den Projektfortschritt zu überprüfen und sicherzustellen, dass das Projekt auf Kurs ist.

Unterschiede internes/externes Projekt

Der Unterschied zwischen einem **internen (innerhalb des Unternehmens)** und einem **externen Projekt (außerhalb des Unternehmens)** bezieht sich auf die **Herkunft** und die **Beteiligten** eines Projekts sowie auf den Kontext, in dem es durchgeführt wird.

Kenntnis Projektkostenplanung

Die **Projektkostenplanung** ist ein wesentlicher Bestandteil der Projektplanung und dient dazu, die voraussichtlichen Kosten eines Projekts zu ermitteln, zu überwachen und zu steuern. Ziel der Projektkostenplanung ist es, sicherzustellen, dass das Projekt innerhalb des festgelegten Budgets abgeschlossen wird.

13) Projektmethoden, Tools

Kenntnisse über Softwareprozessmodelle

Das **Wasserfallmodell** ist eines der ältesten und bekanntesten Softwareprozessmodelle. Es beschreibt eine lineare und sequenzielle Vorgehensweise, bei der jede Phase erst abgeschlossen sein muss, bevor die nächste beginnt.

Das **Spiralmodell** ist ein iteratives und inkrementelles Softwareprozessmodell, das von Barry Boehm entwickelt wurde. Es kombiniert Elemente des Wasserfallmodells mit einer **Risikomanagement-Orientierung** (jede Iteration eine vollständige Entwicklung - Planung, Risikoanalyse, Design, Implementierung und Test)

Kenntnisse über Agiles Projektmanagement/Methoden

Agiles Projektmanagement ist eine flexible und iterative Methode zur Durchführung von Projekten, bei der der Fokus auf Zusammenarbeit, kontinuierlicher Verbesserung und schnellen Anpassungen an Veränderungen liegt. Es wurde ursprünglich für Softwareentwicklungsprojekte entwickelt, hat aber mittlerweile auch in vielen anderen Bereichen Anwendung gefunden.

Fachbegriff DevOps

DevOps ist ein Fachbegriff, der die Kombination von **Development (Entwicklung)** und **Operations (Betrieb)** beschreibt. Es handelt sich um eine Methodik und eine Kultur, die darauf abzielt, die **Zusammenarbeit** und **Kommunikation** zwischen den Entwicklungs- und Betriebsteams zu verbessern, um Software schneller und effizienter zu entwickeln, bereitzustellen und zu betreiben.

Fachbegriff Scrummaster

Der **Scrum Master** ist eine der zentralen Rollen im Scrum-Framework, einer weit verbreiteten agilen Methode zur Softwareentwicklung und Projektmanagement. Die Hauptaufgabe des Scrum Masters ist es, das Scrum-Team zu unterstützen, den Scrum-Prozess zu verstehen und zu befolgen und Hindernisse zu beseitigen, die das Team daran hindern könnten, effektiv zu arbeiten.

Fachbegriff Productowner

Der **Product Owner (PO)** ist eine zentrale Rolle im **Scrum-Framework** und verantwortlich für die Verwaltung und Priorisierung der **Produktanforderungen**. Der Product Owner stellt sicher, dass das Scrum-Team an den wichtigsten Aufgaben arbeitet, die den größten Mehrwert für den Kunden und das Unternehmen bieten.

Fachbegriff Backlog

Backlog ist ein Fachbegriff aus dem agilen Projektmanagement, insbesondere im **Scrum-Framework**, und bezeichnet eine priorisierte Liste von Aufgaben, Anforderungen oder Features, die im Rahmen eines Projekts umgesetzt werden sollen. Product Backlog und Sprint Backlog

Fachbegriff Sprint

Der **Sprint** ist ein zentraler Begriff im **Scrum-Framework** und bezeichnet einen festgelegten, wiederholbaren Zeitraum, in dem ein Scrum-Team an einem bestimmten Set von Aufgaben aus dem **Product Backlog** arbeitet, um ein funktionsfähiges, potenziell auslieferbares **Inkrement** (Produktverbesserung) zu erstellen. Ein Sprint ist also eine **Entwicklungsiteration**, die einen bestimmten Zeitraum umfasst, in dem ein klar definiertes Ziel erreicht werden soll.

Fachbegriff Stakeholder

Der **Stakeholder** ist ein Fachbegriff aus dem Projektmanagement und bezieht sich auf **alle Personen, Gruppen oder Organisationen**, die ein **Interesse** an einem Projekt haben oder von dessen Ergebnis betroffen sind. Stakeholder können sowohl **intern** als auch **extern** zum Unternehmen oder Projektteam gehören. Diese gibt es intern (Projektteam, Führungskräfte, Investoren) und extern (Kunden, Lieferanten, Regulierungsbehörden, ..)

Fachbegriff Daily Scrum/Daily Standup

Der **Daily Scrum** (auch **Daily Standup** genannt) ist ein wichtiger Bestandteil des **Scrum-Frameworks**, der sich auf ein kurzes tägliches Meeting konzentriert, in dem das Scrum-Team den aktuellen Stand der Arbeit bespricht, Hindernisse identifiziert und sich auf die nächsten Schritte vorbereitet.

Fachbegriff User Story/Story Board

User Story und **Story Board** sind zwei wesentliche Begriffe im agilen Projektmanagement, insbesondere im Scrum- und Kanban-Framework. Sie beziehen sich auf die Darstellung und Planung von Anforderungen und Aufgaben in einem Projekt.

Probleme, die beim Wasserfallmodell auftreten können

Das **Wasserfallmodell** in der Softwareentwicklung ist ein sequentielles, lineares Modell, das die Phasen der Softwareentwicklung klar voneinander trennt. Nachteile: mangelnde Flexibilität, späte Fehlererkennung, unklare Anforderungen am Beginn, fehlendes kontinuierliches Feedback

Kenntnisse über den Aufbau des V-Modells

Das **V-Modell** ist ein Softwareentwicklungsprozess, der eine Weiterentwicklung des Wasserfallmodells darstellt. Der Hauptunterschied zum Wasserfallmodell liegt darin, dass das V-Modell den **Testprozess parallel zur Entwicklung** betrachtet. Das „V“ steht dabei symbolisch für die **Verifizierung und Validierung**, die auf unterschiedlichen Ebenen und zu verschiedenen Zeitpunkten des Entwicklungsprozesses durchgeführt werden.

Kenntnisse über Vor- und Nachteile des V-Modells

Frühzeitige Fehlererkennung: durch parallele Führung, klarer Prozess, Verifikation und Validierung sorgt dafür, dass die Software technisch korrekt ist.

Nachteile: wenig flexibel, hoher Aufwand - parallele Tests,

Fachbegriff Softwareentwurf

Der **Softwareentwurf** (auch **Software Design**) ist eine der wichtigsten Phasen im Softwareentwicklungsprozess, in der das System und seine Komponenten auf einer abstrakten Ebene geplant werden. Ziel des Softwareentwurfs ist es, aus den definierten Anforderungen eine konkrete Struktur zu entwickeln, die als Grundlage für die spätere Implementierung dient.

Fachbegriff Prototyp

Ein **Prototyp** in der Softwareentwicklung ist eine **frühzeitige, funktionale Version** einer Software oder eines Systems, die entwickelt wird, um bestimmte Aspekte der Software zu testen, zu validieren oder zu demonstrieren. Der Prototyp kann eine vollständige oder vereinfachte Version der endgültigen Software sein und wird verwendet, um Anforderungen zu überprüfen, Feedback zu sammeln und die Benutzerfreundlichkeit zu testen, bevor die endgültige Implementierung erfolgt.

Fachbegriff Soll-Ist-Analyse

Die **Soll-Ist-Analyse** ist ein wichtiger Begriff im Bereich der **Anforderungsanalyse** und **Projektplanung**. Sie beschreibt den Vergleich zwischen dem **sollten Zustand (Soll)** und dem **aktuell vorhandenen Zustand (Ist)** eines Systems, Prozesses oder Projekts. Ziel dieser Analyse ist es, Abweichungen zu identifizieren und konkrete Maßnahmen abzuleiten, um die Lücken zwischen dem aktuellen Zustand und den gewünschten Zielen zu schließen.

Fachbegriff Versionsverwaltung

Versionsverwaltung (auch **Version Control**) ist ein System, das die Verwaltung von Änderungen an Dateien und Code über die Zeit hinweg ermöglicht. Sie wird hauptsächlich in der Softwareentwicklung eingesetzt, um die Historie von Quellcode, Konfigurationsdateien oder anderen digitalen Dokumenten zu verfolgen, zu speichern und zu verwalten. Mit einer Versionsverwaltung können Entwickler Änderungen rückgängig machen, verschiedene Versionen eines Projekts miteinander vergleichen und zusammenarbeiten, ohne die Arbeit der anderen zu überschreiben.

14) Qualitätssicherung

Kenntnisse über den Zweck von Code-Reviews

Ein Code-Review ist ein wichtiger Prozess in der Softwareentwicklung, bei dem der Quellcode von einem oder mehreren Entwicklern überprüft wird, bevor er in die Hauptcodebasis integriert wird. Der Zweck eines Code-Reviews ist es, die Qualität des Codes zu verbessern, Fehler zu entdecken und Best Practices zu fördern.

Fachbegriff Schreibtischtest

Der **Schreibtischtest** ist ein Begriff aus der Softwareentwicklung und bezeichnet eine **theoretische Überprüfung** oder **Durchsicht** von Code, Prozessen oder Lösungen ohne praktische Ausführung. Dabei wird der Code oder das Konzept auf einem „Schreibtisch“ (also ohne eine tatsächliche Testumgebung oder Simulation) überprüft. Der Schreibtischtest wird häufig von Entwicklern oder Architekten durchgeführt, um potenzielle Fehler, Designprobleme oder Verbesserungspotenziale zu erkennen, bevor eine tatsächliche Implementierung oder ein praktischer Test stattfindet.

Kenntnisse über Black-Box-Test/White-Box-Test, wesentliche Unterschiede

Der **Black-Box-Test** (auch als **Verhaltenstest** bezeichnet) fokussiert sich auf die **Funktionalität** der Software aus der Sicht des Endbenutzers, ohne sich mit der internen Struktur oder dem Code zu beschäftigen.

Der **White-Box-Test** (auch als **struktureller Test** oder **Glasbox-Test** bezeichnet) prüft die **interne Struktur** der Software. Der Tester hat vollständigen Zugang zum Quellcode und überprüft, ob der Code korrekt, effizient und sicher ist. Beim White-Box-Test geht es darum, die interne Funktionsweise zu testen, nicht nur das äußere Verhalten der Software.

Kenntnisse über wichtige Qualitätsmerkmale der Softwarefunktionalität

Qualitätsmerkmale der Softwarefunktionalität beziehen sich auf die Fähigkeit einer Software, die gewünschten Funktionen und Anforderungen korrekt und effizient zu erfüllen. Korrektheit, Vollständigkeit, Richtigkeit, Kompatibilität, Fehlerbehandlung, Wartbarkeit, ...

Kenntnisse über Changemanagement

Changemanagement bezieht sich auf den strukturierten Ansatz, der verwendet wird, um Veränderungen innerhalb einer Organisation zu planen, zu implementieren, zu überwachen und erfolgreich durchzuführen. Das Ziel des Changemanagements ist es, sicherzustellen, dass Änderungen effektiv und reibungslos eingeführt werden, ohne negative Auswirkungen auf den laufenden Betrieb zu haben. Ziel: Minimierung von Widerstand, Effizienz, Optimierung der Auswirkungen

Fachbegriff Versionierung und deren Nutzen

Verfolgen von Änderungen, Rückverfolgbarkeit von Fehlerbehebung, Teamarbeit, erleichterte Wartung, vorsorge gegen Datenverlust, Verwaltung von Releases

Kenntnisse über Problemmanagement

Problemmanagement ist ein Prozess im IT-Service-Management, der sich mit der Identifikation, Analyse und Behebung von zugrunde liegenden Ursachen von wiederkehrenden oder signifikanten IT-Problemen beschäftigt. Das Ziel des

Problemmanagements ist es, Probleme dauerhaft zu lösen, um ihre Wiederholung zu vermeiden und die Servicequalität langfristig zu verbessern. Es handelt sich dabei um eine präventive Maßnahme, die in enger Verbindung mit dem Incident Management (Störungsmanagement) steht.

15) Grundkenntnisse des Programmierens

Stadien der Softwareentwicklung

Anforderungsanalyse - Sammeln und Dokumentieren der Anforderungen.

System- und Softwaredesign - Entwurf der Architektur und des Designs.

Implementierung - Programmierung der Software.

Testen - Überprüfung und Sicherstellung der Qualität.

Deployment - Bereitstellung der Software für den Einsatz.

Wartung und Support - Weiterentwicklung, Fehlerbehebung und Anpassungen.

Rückzugsphase - Das Ende des Lebenszyklus der Software, wenn sie nicht mehr unterstützt wird.

Fachbegriffe Prozedurale Programmierung, Objektorientierte Programmierung, Unterschiede

Prozedurale Programmierung ist ein Programmierparadigma, das auf der Durchführung von Anweisungen (Prozeduren oder Funktionen) basiert, um Aufgaben zu erledigen. Der Code wird dabei in eine Reihenfolge von aufeinander folgenden Schritten (Anweisungen) unterteilt.

Objektorientierte Programmierung ist ein Paradigma, das die Daten in sogenannten Objekten zusammen mit den dazugehörigen Methoden (Funktionen) kapselt. In OOP wird die Software als Sammlung von Objekten modelliert, die miteinander interagieren.

Fachbegriff Algorithmus

Ein Algorithmus ist eine klar definierte, endliche Abfolge von Anweisungen oder Regeln, die Schritt für Schritt ein bestimmtes Problem löst oder eine Aufgabe ausführt. Er ist präzise, wiederholbar und führt bei korrekter Anwendung immer zum gleichen Ergebnis.

Fachbegriff Pseudocode

Pseudocode ist eine vereinfachte, sprachähnliche Darstellung eines Algorithmus. Er verwendet Elemente natürlicher Sprache und Programmiersprachen, um Abläufe logisch und strukturiert darzustellen, ohne sich an eine konkrete Programmiersyntax halten zu müssen.

Kenntnisse über Sortieralgorithmen (Bubblesort, Quicksort)

Bubblesort: Ein einfacher Sortieralgorithmus, der benachbarte Elemente vergleicht und vertauscht, wenn sie in der falschen Reihenfolge sind. Dieser Vorgang wird wiederholt, bis die Liste sortiert ist. Nachteil: langsam bei großen Datenmengen ($O(n^2)$).

Quicksort: Ein effizienter, rekursiver Sortieralgorithmus. Er wählt ein "Pivot"-Element, teilt die Liste in kleinere (kleiner als Pivot) und größere Teile und sortiert diese rekursiv. Sehr schnell bei großen Datenmengen (durchschnittlich $O(n \log n)$).

Kenntnisse über Suchalgorithmen (sequentielle Suche, binäre Suche)

Sequentielle Suche: Durchsucht eine Liste Element für Element von vorne bis hinten.

Einfach, aber langsam bei großen Listen ($O(n)$).

Binäre Suche: Funktioniert nur bei sortierten Listen. Teilt die Liste wiederholt in zwei Hälften und vergleicht das mittlere Element mit dem Suchwert. Sehr schnell ($O(\log n)$).

Ablauf der Programmentwicklung

Problemdefinition - Verstehen und Beschreiben der Aufgabe.

Anforderungsanalyse - Festlegen, was das Programm leisten soll.

Entwurf - Planung der Programmlogik (z. B. mit Pseudocode oder Flussdiagramm).

Implementierung - Schreiben des Quellcodes in einer Programmiersprache.

Testen und Debugging - Überprüfen auf Fehler und Korrektur.

Dokumentation - Beschreibung der Funktionsweise und Nutzung.

Wartung - Pflege und Weiterentwicklung nach Fertigstellung.

Fachbegriffe zum Aufbau einer Programmiersprache (Syntax, Semantik, Kommentare, Schlüsselwörter, Anweisung)

Syntax: Regeln für den korrekten Aufbau von Code (z. B. richtige Schreibweise, Satzzeichen).

Semantik: Bedeutung des Codes - was der Code tatsächlich ausführt.

Kommentare: Notizen im Code, die vom Programm ignoriert werden und zur Erklärung dienen.

Schlüsselwörter: Reservierte Wörter der Programmiersprache (z. B. if, while, return).

Variablen: Speicherorte für Daten mit einem Namen.

Datentypen: Arten von Daten (z. B. Ganzzahl, Text, Wahrheitswert).

Operatoren: Zeichen zur Verarbeitung von Daten (z. B. +, -, ==).

Fachbegriffe Interpreter und Compiler (Unterschiede, Vor- und Nachteile)

Interpreter: Übersetzt und führt den Quellcode zeilenweise direkt aus.

Vorteile: Einfach zu testen und schnelleres Feedback.

Nachteile: Langsamer in der Ausführung.

Compiler: Übersetzt den gesamten Quellcode in eine ausführbare Datei, bevor das Programm läuft.

Vorteile: Schnelle Ausführung, Code muss nicht mitgeliefert werden.

Nachteile: Fehler erst nach dem Kompilieren sichtbar, weniger flexibel beim Testen.

Fachbegriff Debugger (Einsatz)

Ein Debugger ist ein Werkzeug zur Fehlersuche im Programm.

Er hilft, den Code Schritt für Schritt auszuführen, Variablenwerte zu beobachten und Programmabläufe zu analysieren, um logische oder syntaktische Fehler zu finden und zu beheben.

Fachbegriff Assembler

Assembler ist eine Programmiersprache, die eine symbolische Darstellung von Maschinenbefehlen bietet. Sie ermöglicht es, direkt auf die Hardware eines Computers zuzugreifen, indem sie mnemonische Codes verwendet, die den maschinenlesbaren Code (Maschinensprache) repräsentieren. Ein Assembler-Programm wird durch einen Assembler-Compiler in Maschinensprache übersetzt.

Fachbegriff Rekursive Funktionen

Rekursive Funktionen sind Funktionen, die sich selbst aufrufen, um ein Problem zu lösen. Sie bestehen aus zwei Hauptteilen: einer Basisbedingung, die den rekursiven Aufruf stoppt, und einem rekursiven Schritt, der das Problem in kleinere Teilprobleme aufteilt.

Beispiel: Die Berechnung der Fakultät einer Zahl ($n!$) kann rekursiv durch $n * (n-1)!$ erfolgen.

Kenntnisse über ASCII-Tabellen

Die ASCII-Tabelle (American Standard Code for Information Interchange) ist ein Zeichensatz, der Zeichen (Buchstaben, Zahlen, Sonderzeichen) in numerische Codes übersetzt, die von Computern verarbeitet werden können. Sie verwendet 7 Bit und enthält 128 Zeichen, darunter:

- Zeichen 0-31: Steuerzeichen (z. B. Wagenrücklauf, Zeilenumbruch).
- Zeichen 32-47: Sonderzeichen (z. B. Leerzeichen, Satzzeichen).
- Zeichen 48-57: Ziffern (0-9).
- Zeichen 65-90: Großbuchstaben (A-Z).
- Zeichen 97-122: Kleinbuchstaben (a-z).

Es gibt auch erweiterte ASCII-Zeichensätze mit 256 Zeichen, die zusätzliche Symbole und Zeichen enthalten.

Kenntnisse über Variablenarten, Datentypen und Definitionen

Variablenarten:

Lokale Variablen: Nur innerhalb einer Funktion oder eines Blocks sichtbar und verfügbar.

Globale Variablen: Über das gesamte Programm hinweg zugänglich, unabhängig von Funktionen oder Blöcken.

Statische Variablen: Behalten ihren Wert zwischen Funktionsaufrufen bei, auch wenn sie lokal sind.

Datentypen:

Primitive Datentypen:

- Integer (int): Ganze Zahlen (z. B. 1, -5, 42).
- Float/Double: Gleitkommazahlen (z. B. 3.14, -0.001).
- Char: Einzelne Zeichen (z. B. 'A', '3').
- Boolean (bool): Wahrheitswerte (true/false).

Komplexe Datentypen:

- Strings: Zeichenketten (z. B. "Hallo").
- Arrays: Sammlungen von Werten gleichen Typs (z. B. [1, 2, 3]).
- Listen, Tupel, Sets (je nach Sprache): Flexiblere Datenstrukturen, die verschiedene Datentypen speichern können.

Fachbegriff Schleifen, Beispiele für Schleifen

Schleifen sind Kontrollstrukturen, die es ermöglichen, einen Codeblock wiederholt auszuführen, solange eine bestimmte Bedingung erfüllt ist.

FOR-schleife - festgelegte Wiederholungen

WHILE-Schleife - Abbruchbedingung kopfgesteuert

DO-WHILE - Abbruchbedingung am Ende der Schleife wird mindestens einmal durchlaufen - fußgesteuert

Kenntnisse über Verzweigungen und Fallunterscheidungen

Verzweigungen und Fallunterscheidungen sind Kontrollstrukturen, die es ermöglichen, den Programmablauf basierend auf Bedingungen zu steuern.

IF-ELSE - führt eine Aktion aus, wenn eine Bedingung wahr ist, und eine andere, wenn sie falsch ist

SWITCH-CASE - Fallunterscheidungen ermöglichen es, verschiedene Fälle basierend auf dem Wert einer Variablen zu behandeln. Sie sind in vielen Programmiersprachen eine Alternative zu langen if-else-Ketten.

Kenntnis der objektorientierten Programmierung (Klassen, Objekte, Vererbung, ...)

Objektorientierte Programmierung (OOP) ist ein Programmierparadigma, das auf dem Konzept von "Objekten" basiert. Objekte sind Instanzen von Klassen, die sowohl Daten (Eigenschaften) als auch Methoden (Funktionen) enthalten, die auf diese Daten zugreifen und sie manipulieren

Klassen definieren den Bauplan von Objekten.

Objekte sind Instanzen von Klassen.

Vererbung ermöglicht es, neue Klassen basierend auf bestehenden Klassen zu erstellen.

Polymorphismus erlaubt unterschiedliche Implementierungen von Methoden in verschiedenen Klassen.

Kapselung sorgt dafür, dass die Daten einer Klasse sicher und nur über definierte Schnittstellen zugänglich sind.

16) Kenntnis und Verwendung von Datenbanken, Datenmodellen und Datenstrukturen

Fachbegriff Datenbanksysteme (Traditionelle Datenbanken (RDB), Objektorientierte Datenbanken, Multimedia-Datenbanken (GIS), Data-Warehouse und OLAP)

Datenbanksysteme (DBS) sind Softwarelösungen zur Speicherung, Verwaltung und Abfrage großer Datenmengen. Sie bestehen aus einer Datenbank (Daten) und einem Datenbankmanagementsystem (DBMS), dass die Daten organisiert und den Zugriff steuert.

1. Traditionelle relationale Datenbanken (RDB):

- Speichern Daten in Tabellen (Zeilen & Spalten).

- Nutzen SQL zur Datenabfrage.

- Beziehungen zwischen Tabellen (z. B. Fremdschlüssel).

- Beispiel: MySQL, PostgreSQL, Oracle.

2. Objektorientierte Datenbanken (OODB):

- Speichern Objekte direkt (wie in OOP-Sprachen).

- Unterstützen Vererbung, Kapselung, Polymorphismus.

- Kein Umwandeln in Tabellen nötig.

- Beispiel: db4o, ObjectDB.

3. Multimedia-Datenbanken (z. B. GIS):

- Speichern Multimediadaten wie Bilder, Videos, Ton, Karten.

- GIS (Geoinformationssysteme): Speichern und analysieren geografische Daten (z. B. für Navigation, Stadtplanung).

- Brauchen spezielle Formate, große Speicherkapazität und schnelle Suche.

4. Data-Warehouse:

- Zentrale Sammelstelle für Daten aus verschiedenen Quellen.

- Daten sind oft zusammengefasst und vorgefiltert.

- Unterstützen langfristige Analysen und Entscheidungen im Unternehmen.

5. OLAP (Online Analytical Processing):

- Methode zur mehrdimensionalen Analyse von Daten im Data-Warehouse.

- Nutzer können Daten aus verschiedenen Perspektiven betrachten (z. B. nach Zeit, Region, Produkt).

- Unterstützt schnelle, interaktive Auswertungen

Fachbegriffe zu Datenbankabfragen (z.B.: SQL, SQL/XML)

SQL (Structured Query Language):

Standardsprache für Abfragen in relationalen Datenbanken.

Ermöglicht Erstellen, Ändern, Löschen und Abfragen von Daten.

SQL/XML:

Erweiterung von SQL zur Verarbeitung von **XML-Daten** in relationalen Datenbanken.

Erlaubt Kombination von **strukturierter** (Tabellen) und **semi-strukturierter** (XML) Daten.

Nutzt Funktionen wie XMLQUERY, XMLELEMENT, XMLTABLE.

Praktisch für Webanwendungen oder den Datenaustausch zwischen Systemen.

Fachbegriff Datenbankmanagementsystem (DBMS)

Ein **DBMS** ist eine Software, die das **Erstellen, Verwalten, Speichern** und **Abrufen** von Daten in einer **Datenbank** ermöglicht.

Hauptaufgaben:

- **Verwaltung der Datenstruktur** (z. B. Tabellen, Beziehungen)
- **Zugriffskontrolle und Sicherheit**
- **Mehrbenutzerbetrieb** (gleichzeitiger Zugriff)
- **Datenintegrität** (Richtigkeit & Vollständigkeit)
- **Datenabfragen mit SQL**

Fachbegriff Content Management System (CMS)

Ein **CMS** ist eine Software zur **Erstellung, Bearbeitung** und **Verwaltung** von **Webinhalten**, ohne dass man Programmierkenntnisse benötigt.

Hauptfunktionen:

- Inhalte (Texte, Bilder, Videos) leicht einfügen und ändern
- Benutzerverwaltung
- Design-Vorlagen (Themes)
- Plugins zur Erweiterung

Fachbegriff Integrität im Zusammenhang mit Datenbanken

Integrität in Datenbanken bedeutet die **Korrektheit, Vollständigkeit** und **Widerspruchsfreiheit** der gespeicherten Daten.

Entity-Integrität:

- Jede Tabelle hat einen **Primärschlüssel**, der eindeutig ist und **nicht null** sein darf.
- Beispiel: Jede Kundennummer darf nur einmal vorkommen.

Referentielle Integrität:

- Beziehungen zwischen Tabellen bleiben konsistent.
- Ein **Fremdschlüssel** verweist auf einen gültigen Eintrag in einer anderen Tabelle.
- Beispiel: Eine Bestellung darf nur existieren, wenn der Kunde existiert.

Datenintegrität:

- Regeln zur Sicherstellung gültiger Werte (z. B. Zahlenbereiche, Datentypen, Pflichtfelder).
- Beispiel: Alter darf nicht negativ sein.

Fachbegriff Redundanz im Zusammenhang mit Datenbanken

Redundanz in Datenbanken bedeutet, dass **die gleichen Daten mehrfach** gespeichert sind.

Warum problematisch?

- **Speicherplatzverschwendung**
- **Inkonsistenzen** möglich (z. B. wenn nur eine Kopie geändert wird)
- **Wartung wird aufwendiger**

Vorgangsweise bei der Datenmodellierung (RDB)

Die **Datenmodellierung** bei relationalen Datenbanken (RDB) ist ein strukturierter Prozess zur Planung der Datenstruktur. Anforderungsanalyse, erstellen ERM, festlegen von Schlüsseln, Überführung in Relationenmodell (Tabelle, Spalten, Schlüssel), Normalisierung, Implementierung

Kenntnisse über die ersten drei Normalformen im Zusammenhang mit Datenbanken

Die ersten drei Normalformen (1NF, 2NF, 3NF) helfen dabei, eine Datenbank strukturiert und fehlerfrei aufzubauen, indem sie Redundanzen und Anomalien vermeiden.

NF1: Alle Attribute enthalten nur atomare (unteilbare) Werte.

NF2: erfüllt 1NF und alle Nicht-Schlüsselattribute hängen vollständig vom Primärschlüssel ab.

NF3: erfüllt 2NF und keine transitive Abhängigkeit - Nicht-Schlüsselattribute dürfen nicht voneinander abhängen.

Fachbegriffe Primärschlüssel, Fremdschlüssel, Relationen

Kenntnis über Vor- und Nachteile bei Verwendung eines Indexes

Index in Datenbanken = spezielle Datenstruktur zur **Beschleunigung von Abfragen** (ähnlich wie ein Inhaltsverzeichnis im Buch). Schnelle Suchabfragen aber mehr Speicherplatz wird benötigt

Vor- und Nachteile von Freeware Datenbanken

Kenntnisse über Sicherungsmethoden

Sicherungsmethoden in Datenbanken beziehen sich auf Techniken und Verfahren, die verwendet werden, um die Datenintegrität und Verfügbarkeit der Daten zu gewährleisten, insbesondere im Falle von Fehlern, Ausfällen oder Datenverlusten. Vollständiges Backup (Dump), Transaktionsprotokolle (Änderungen werden protokolliert), Replikation, Clustering (mehrerer Datenbankserver zusammenführen (Lastverteilung, Fehlertoleranz)

Fachbegriff Sperrtabelle und Sperrverhalten

Sperrtabelle und Sperrverhalten beziehen sich auf Mechanismen in Datenbanksystemen, die zur Verhinderung von gleichzeitigen Konflikten bei Datenzugriffen verwendet werden, insbesondere in Multi-User-Datenbanken. Sie stellen sicher, dass Daten in einer Transaktion konsistent bleiben und verhindern Datenverlust oder Datenkorruption durch konkurrierende Zugriffe.

Fachbegriff BIS (Betriebliches Informationssystem)

Ein **Betriebliches Informationssystem (BIS)** ist ein **Software-gestütztes System**, das in Unternehmen eingesetzt wird, um **Daten zu sammeln, zu verarbeiten, zu speichern und bereit zu stellen**, sodass die Geschäftsprozesse effizienter gestaltet und Entscheidungen besser unterstützt werden können.

Kenntnisse/Fachbegriff ERP Systeme

Ein **ERP-System** ist eine integrierte Softwarelösung, die Unternehmen hilft, **verschiedene Geschäftsprozesse zu verwalten und zu optimieren**. Es vereint verschiedene **Bereiche** eines Unternehmens, wie z. B. **Finanzen, Personalwesen, Produktion, Lagerverwaltung und Vertrieb**, in einer einzigen Software-Plattform.

Kenntnisse/Fachbegriff BI/BW Systeme

BI/BW Systeme (Business Intelligence / Business Warehouse Systeme)

Business Intelligence (BI) und Business Warehouse (BW) Systeme sind Softwarelösungen, die Unternehmen dabei unterstützen, Daten zu sammeln, zu analysieren und in wertvolle Informationen umzuwandeln, um bessere Geschäftsentscheidungen zu treffen.

Kenntnisse der Abläufe und Prozessschritte (Auswählen DBMS, Erstellen des physischen Modells, Performance- und Stresstests, Datensicherheit, Datenschutz, Datenverschlüsselung - Kryptografie, Datenmigration) zum Umsetzen von Datenmodellen in eine Datenbank

Die Umsetzung eines Datenmodells in eine Datenbank umfasst mehrere wichtige Schritte, die sowohl technische als auch sicherheitsrelevante Aspekte berücksichtigen. Dazu gehören:

- **Auswahl des passenden DBMS** für die spezifischen Anforderungen.
- **Erstellung eines physischen Modells**, das auf dem logischen Modell basiert.
- **Durchführung von Performance- und Stresstests**, um sicherzustellen, dass das System auch unter hoher Last funktioniert.
- **Gewährleistung von Datensicherheit und Datenschutz**, insbesondere durch Zugriffskontrollen und Verschlüsselung.
- **Datenmigration** aus bestehenden Systemen und Sicherstellung der Datenkonsistenz und -integrität.

All diese Schritte sind entscheidend, um ein stabiles, performantes und sicheres Datenbanksystem zu gewährleisten.

Kenntnisse der Abläufe und Prozessschritte (Zugriffsschnittstelle, Zugriffstechnologie, Transaktionskonzept, Programmierung, Testreihen, Benutzerabnahmetest, Ergebnisprüfung)

Die Schritte zur Umsetzung eines Datenmodells in eine Datenbank umfassen nicht nur die technische Implementierung der Datenbank selbst, sondern auch die **Integration mit Anwendungen**, die **Programmierlogik**, **Testreihen** sowie die **Abnahme** und **Ergebnisprüfung**. Jeder dieser Schritte stellt sicher, dass die Datenbank den Anforderungen hinsichtlich **Performance**, **Zugriffssicherheit**, **Fehlerfreiheit** und **Benutzerfreundlichkeit** entspricht.

Wichtige Phasen im Prozess:

Zugriffsschnittstelle (APIs, Treiber),

- **Zugriffstechnologie** (SQL, ORM),
- **Transaktionskonzept** (ACID, Isolationsebenen),
- **Programmierung** (Abfragen, Stored Procedures),
- **Testreihen** (Unit- und Integrationstests),
- **Benutzerabnahmetests** (UAT),
- **Ergebnisprüfung** (Validierung und Go-Live).

17) Systementwicklung/Testkonzepte

Fachbegriff Programmspezifikation

Programmspezifikation ist der Prozess und das Dokument, in dem die **Anforderungen und Funktionsweise** eines Softwareprogramms detailliert festgelegt werden. Sie dient als **Basis** für die spätere Entwicklung und Implementierung der Software. In der Programmspezifikation werden sowohl **technische** als auch **funktionale** Anforderungen beschrieben, um sicherzustellen, dass das entwickelte Programm die gewünschten Funktionen und Qualitätsmerkmale erfüllt.

Fachbegriff Datenmodell

Datenmodell bezeichnet die abstrakte Darstellung der Struktur und Organisation von Daten, die in einem Informationssystem gespeichert und verarbeitet werden. Es definiert die **Beziehungen, Datenstrukturen und Datenflüsse** innerhalb eines Systems und dient als Grundlage für die **Datenbankgestaltung** und die **Datenverarbeitung**. Ein gutes Datenmodell hilft dabei, die Komplexität von Daten zu reduzieren und ermöglicht eine effiziente Speicherung, Verarbeitung und Abfrage von Daten.

Kenntnisse über wichtige Datentypen und Datenstrukturen

Kenntnisse über Funktionen (Definition, Schnittstelle, Parameter, Rückgabewert, Aufruf)

Unterschiede zwischen Call-By-Value und Call-By-Reference

- **Call-by-Value** wird oft verwendet, wenn es wichtig ist, dass der ursprüngliche Wert des Arguments nicht verändert wird.
- **Call-by-Reference** ist nützlich, wenn große Datenstrukturen übergeben werden oder wenn die Funktion den ursprünglichen Wert des Arguments verändern soll.

Kenntnisse über Klassen (Datenelemente, Konstruktor, Destruktor, Methoden, Zugriffsmodifikatoren)

- **Datenelemente** Variablen innerhalb der Klasse, die den Zustand des Objekts darstellen.
- **Konstruktor** Eine Methode zur Initialisierung eines Objekts der Klasse.
- **Destruktor** Eine Methode, die beim Zerstören eines Objekts automatisch aufgerufen wird
- **Methoden** Funktionen innerhalb der Klasse, die das Verhalten des Objekts definieren.
- **Zugriffsmodifikatoren** bestimmen, wie und ob Mitglieder einer Klasse von außen zugänglich sind.

Kenntnisse über das Prinzip der Vererbung

Fachbegriff Standardbibliothek

Standardbibliothek bezeichnet eine Sammlung von vorgefertigten Funktionen, Klassen, Datenstrukturen und Modulen, die in einer Programmiersprache integriert sind und die Entwicklung von Software erleichtern. Sie stellt grundlegende und oft wiederverwendbare Funktionen zur Verfügung, die von der Programmiersprache selbst bereitgestellt und gepflegt werden.

Kenntnisse über Testkonzepte

Auswertung eines Softwaretests

Die Auswertung dient dazu, die Qualität der Software zu beurteilen. Dabei werden Testergebnisse gesammelt, mit Anforderungen verglichen und Fehler analysiert. Wichtige Punkte sind:

- **Fehlerstatistik:** Anzahl, Art und Schwere der Fehler
- **Testabdeckung:** Wurden alle Funktionen geprüft?
- **Erfolgsquote:** Wie viele Tests wurden bestanden?
- **Testbericht:** Dokumentation der Ergebnisse
- **Entscheidung:** Freigabe, Nachbesserung oder erneute Tests

Ziel: Bewertung der Softwarequalität und Entscheidung über den nächsten Schritt.

Kriterien für den Test von Datenbankfeldern unterschiedlicher Typen (Mail, Datum, ...)

Unterschiede zwischen einem reproduzierbaren/nicht-reproduzierbaren Fehler

Kenntnisse über Möglichkeiten zur Automatisierung von Tests

Kriterien für den Test von Datenbankfeldern unterschiedlicher Typen:

- **E-Mail:** Formatprüfung (z. B. name@domain.com), ungültige Zeichen, Länge
- **Datum:** Gültiges Format (z. B. YYYY-MM-DD), Zukunft/Vergangenheit, Schaltjahre
- **Zahlen:** Bereich (z. B. keine negativen Werte bei Alter), Dezimalstellen
- **Textfelder:** Maximale Länge, Sonderzeichen, leere Eingabe

Reproduzierbar: Tritt unter gleichen Bedingungen immer auf → leichter zu beheben

Nicht-reproduzierbar: Tritt sporadisch oder unter unbekannten Bedingungen auf → schwer zu analysieren

18) Übungsbeispiel

Anforderung:

- Erstellung eines HTML-Formulars für die Eingabe von Stammdaten
- Umsetzung eines responsiven Designs
- Coding einer Gültigkeitsprüfung einer SV-Nummer
- Generierung eines QR-Codes
- Prüfung des QR-Codes

Benötigte Hardware/Software:

- PC mit Windows 11 Installation
- QR-Code Scanner, zB Code Two QR Code Desktop Reader
- Entwicklungsumgebung, zB Xampp
- Code Editor
- Browser

1. Eingabe-Formular mit Framework

Erstellen Sie ein HTML-Formular, um eine Stammdaten-Abfrage durchführen zu können.

Verwenden Sie nach Möglichkeit ein Framework (Bootstrap, oä.) und stellen Sie sicher, dass das Formular sowohl am PC als auch am Handy verwendbar ist (Stichwort: Responsive).

Folgende Informationen sollten zumindest im Formular zu finden sein:

- Vorname
- Nachname
- SV-NR

Folgender Lösungsvorschlag dient nur zur Orientierung (Lösung muss nicht ident aussehen):

QR-Code mit Versicherungsnummerncheck

Anrede

Herr

Vorname

Nachname

SV-NR

Erstellen

Stellen Sie sicher, dass die relevanten Felder auch wirklich eingegeben wurden (Überprüfung bei Betätigen des Senden-Buttons). Ein Speichern der Daten ist NICHT notwendig.

2. Gültigkeitsprüfung der SV-Nummer

Überprüfen Sie anhand der eingegebenen Werte, ob es sich beim Feld SV-NR um eine gültige österreichische Sozialversicherungsnummer handelt. Recherchieren Sie im Internet, wie man die Gültigkeit einer österreichischen Sozialversicherungsnummer prüfen kann.

Sie können diese Beispiel-Daten nutzen, um die Funktionalität Ihres Codes zu überprüfen:

Beispiele für **gültige** SV-Nummern:

4422 180599

3567 010705

5884 050902

Beispiele für **ungültige** SV-Nummern:

2511 010100

5255 121299

4999 070700

3. Ausgabe eines QR-Codes

Nach der Eingabe aller Daten muss ein QR-Code generiert werden. Der Inhalt des QR-Codes soll nur den Text "Richtig" (bei korrekter SV-Nr) oder "Falsch" (bei ungültiger SV-Nr) enthalten. Suchen Sie selbst im Internet nach einer frei verfügbaren Programm-Bibliothek oder einem Framework, womit Sie den QR-Code generieren können.

Beispiel für die Formular-Ausgabe:

QR-Code mit Versicherungsnummerncheck

Anrede

Vorname

Nachname

SV-NR



Hinweis: Bei der Lehrabschlussprüfung gilt striktes Handy-Nutzungsverbot. Suchen Sie im Internet die Freeware "CodeTwo QR Code Desktop Reader", damit können Sie den QR-Code aus Bildschirmausschnitten abschnappen und Ihre selbst generierten QR-Codes überprüfen.