# Testerský slovník časť 18:

(postupne budem zverejňovať pojmy, ktoré som nazbierala)

**Feature toggles/Feature flag**

sú prepínače, ktorými možno zapnúť alebo vypnúť konkrétnu funkcionalitu bez nutnosti meniť kód. Umožňujú napríklad nasadiť funkciu na všetky prostredia, ale aktivovať ju len v niektorých z nich (napr. len v teste).

**SMTP**

(Simple Mail Transfer Protocol) je protokol na odosielanie e-mailov z aplikácie – typicky sa používa na zasielanie notifikácií, potvrdení objednávky alebo resetovacích odkazov. V testovaní je dôležité overiť, či je SMTP správne nakonfigurovaný, aby e-maily odchádzali a neboli zablokované alebo označené ako spam.

**Build number**

je číselný identifikátor konkrétneho zostavenia (build-u) aplikácie. Pomáha rozlíšiť jednotlivé verzie softvéru – napríklad ak vývojári vykonali zmeny a znova aplikáciu zbuildovali, každé nové zostavenie dostane nové „build number“, čo umožňuje presne sledovať, ktorá verzia je nasadená v ktorom prostredí.

**Commit hash**

je jedinečný identifikátor (zvyčajne dlhý reťazec znakov), ktorý označuje konkrétny commit v systéme na správu verzií (napr. Git). Vďaka nemu vieš presne určiť, aký kód bol nasadený – aj keď dve verzie aplikácie vyzerajú rovnako, rôzny commit hash odhalí, že pochádzajú z iného stavu vývoja.

**Version endpoint**

je špeciálne API rozhranie (napr. /version), ktoré vracia aktuálne informácie o nasadenej verzii aplikácie – ako napr. číslo buildu alebo commit hash. Tester si tak vie jednoducho overiť, či je v danom prostredí nasadená správna verzia softvéru.

**Debug logy**

sú detailné výpisy z aplikácie určené najmä pre vývojárov – obsahujú technické informácie, ktoré pomáhajú pri hľadaní chýb (napr. hodnoty premenných, priebeh funkcií). V produkčnom prostredí by nemali byť zapnuté, pretože môžu spomaliť aplikáciu a odhaliť citlivé informácie.

**Latency**

je oneskorenie medzi požiadavkou a reakciou – teda čas, ktorý uplynie od odoslania požiadavky (napr. klik na tlačidlo) po prijatie odpovede zo servera. V testovaní je dôležité sledovať, či toto oneskorenie neprekračuje akceptovateľné limity, najmä pri prechode na produkčné prostredie, kde môže byť infraštruktúra odlišná.

**HTTP**

(Hypertext Transfer Protocol) je základný komunikačný protokol, pomocou ktorého prehliadač posiela požiadavky na webový server a prijíma odpovede (napr. HTML stránky, obrázky, API odpovede). Je to nezabezpečený protokol – preto sa v praxi často používa jeho bezpečná verzia HTTPS, ktorá navyše šifruje komunikáciu.

**HTTPS**

(Hypertext Transfer Protocol Secure) je zabezpečená verzia protokolu HTTP, ktorá šifruje komunikáciu medzi prehliadačom a serverom pomocou SSL/TLS certifikátu. Vďaka tomu sú údaje chránené pred odpočúvaním, manipuláciou a falošnými stránkami.

**TLS**

(Transport Layer Security) je kryptografický protokol, ktorý zabezpečuje šifrovanie a integritu dát pri prenose medzi klientom a serverom – je nástupcom SSL. Používa sa v spojení s HTTPS a zabezpečuje, že komunikácia nemôže byť odpočúvaná ani zmenená.

**DNS**

(Domain Name System) je systém, ktorý prekladá názvy domén (napr. example.com) na IP adresy serverov, kde beží aplikácia. Ak DNS nefunguje správne alebo je zle nastavený, aplikácia sa nemusí načítať, pretože sa klient nedostane na správny server.

**Redirect**

je automatické presmerovanie používateľa z jednej adresy na inú – napríklad keď zadáš example.com a prehliadač ťa presmeruje na https://www.example.com. V testovaní kontrolujeme, či sú tieto presmerovania správne nastavené (napr. HTTP → HTTPS alebo staré URL → nové) a či nevedú na chybné alebo neexistujúce stránky.

**SSL certifikáty**

slúžia na zabezpečenie šifrovanej komunikácie medzi klientom a serverom (napr. pri HTTPS). Overením certifikátu zisťujeme, či je platný, vydaný dôveryhodnou autoritou a či zodpovedá doméne, na ktorej beží aplikácia.

**CSP**

(Content-Security-Policy)je bezpečnostný HTTP header, ktorý určuje, z akých zdrojov môže webová stránka načítať obsah (napr. skripty, obrázky, štýly). Pomáha zabrániť útokom ako Cross-Site Scripting (XSS), pretože obmedzuje, čo sa môže na stránke spúšťať a odkiaľ.

**X-Frame-Options**

je bezpečnostný HTTP header, ktorý určuje, či sa môže stránka zobraziť v rámci iného webu (napr. v <iframe>). Slúži ako ochrana pred útokmi typu „clickjacking“, kde by útočník mohol cez neviditeľné prekrytie manipulovať so stránkou bez vedomia používateľa.

**CORS**

(Cross-Origin Resource Sharing)je bezpečnostný mechanizmus, ktorý určuje, ktoré domény môžu pristupovať k zdrojom (napr. API) na inej doméne. Ak nie je CORS správne nastavený, aplikácia môže buď zbytočne blokovať požiadavky, alebo naopak umožniť prístup neoprávneným doménam, čo je bezpečnostné riziko.

Týmto nekončíme, len začíname. Dajte do komentov termíny alebo slovné spojenia, s ktorými ste sa pri testovaní stretli. Tie, ktoré vás zaujali, nevedeli ste, čo znamenajú a pod.

**#tester #testovacislovnik #manualnytester #automatizovanytester #ITtester #QA #ITkariera**

# Testerský slovník časť 19:

(postupne budem zverejňovať pojmy, ktoré som nazbierala)

**Integrity constraints**

Pravidlá v databáze, ktoré zabezpečujú správnosť a konzistenciu dát. Napríklad obmedzenie, že hodnota v stĺpci nemôže byť NULL alebo že musí byť jedinečná.

**Orphan records**

Záznamy v databáze, ktoré nemajú nadväzujúci rodičovský záznam (napr. objednávka priradená k neexistujúcemu zákazníkovi). Často ide o dôsledok chýb v mazacích alebo aktualizačných operáciách.

**Stavový diagram**

Grafické znázornenie stavov systému alebo objektu a prechodov medzi nimi. Pomáha testerom pochopiť logiku správania a identifikovať možné scenáre.

**UML**

(Unified Modeling Language) štandardizovaný jazyk na vizuálne modelovanie softvéru. Používa sa na tvorbu diagramov procesov, tried, sekvencií či stavov.

**Session management**

Správa používateľských relácií v aplikácii – od prihlásenia až po odhlásenie alebo vypršanie časového limitu. Nesprávne nastavenie môže viesť k bezpečnostným rizikám.

**Batch processing**

Spracovanie väčšieho množstva dát naraz v dávkach, často plánovane mimo špičky. V testovaní sa overuje správnosť, výkon a spracovanie chýb.

**Cache invalidácia**

Spracovanie väčšieho množstva dát naraz v dávkach, často plánovane mimo špičky. V testovaní sa overuje správnosť, výkon a spracovanie chýb.

**Force refresh**

Spracovanie väčšieho množstva dát naraz v dávkach, často plánovane mimo špičky. V testovaní sa overuje správnosť, výkon a spracovanie chýb.

**HSTS**

(HTTP Strict Transport Security) bezpečnostný mechanizmus, ktorý núti prehliadače používať HTTPS namiesto HTTP. Pomáha chrániť pred útokmi typu downgrade a cookie hijacking.

**Maintenance mode**

Režim, v ktorom je aplikácia dočasne nedostupná pre používateľov z dôvodu údržby. Tester kontroluje, či sa zobrazuje správna informačná stránka a či sa neodošlú nežiadané požiadavky.

**Fuzzy match**

Metóda porovnávania reťazcov, ktorá umožňuje malé odchýlky (napr. preklepy). Používa sa pri vyhľadávaní alebo kontrole dát, kde sa očakávajú nepresnosti.

**Spätná kompatibilita**

Schopnosť novej verzie systému fungovať s dátami, rozhraniami alebo komponentmi zo starších verzií. V testovaní je kľúčové overiť, že existujúci používatelia nebudú mať problémy.

Týmto nekončíme, len začíname. Dajte do komentov termíny alebo slovné spojenia, s ktorými ste sa pri testovaní stretli. Tie, ktoré vás zaujali, nevedeli ste, čo znamenajú a pod.

**#tester #testovacislovnik #manualnytester #automatizovanytester #ITtester #QA #ITkariera**

**UTC**

(Coordinated Universal Time) svetový časový štandard používaný na synchronizáciu systémov. Tester overuje správnu konverziu na lokálny čas a spracovanie časových zón.

**Responzivita**

Schopnosť používateľského rozhrania prispôsobiť sa rôznym veľkostiam obrazovky a zariadeniam. V testovaní sa kontroluje čitateľnosť, dostupnosť funkcií a zachovanie dizajnu.

**Trigger**

Automatická akcia v databáze, ktorá sa spustí pri určitej udalosti (napr. vloženie alebo úprava záznamu). Testuje sa, či vykonáva očakávané operácie bez nežiaducich vedľajších efektov.

**REST API**

(Representational State Transfer API) architektonický štýl pre tvorbu webových služieb. Používa HTTP metódy (GET, POST, PUT, DELETE) a pracuje s dátami vo formátoch ako JSON alebo XML.

**SOAP API**

(Simple Object Access Protocol) protokol pre výmenu dát medzi aplikáciami cez XML. Je prísnejší než REST, často používaný vo firemných systémoch.

**OAuth**

Otvárací štandard pre bezpečnú delegovanú autentifikáciu. Umožňuje používateľovi prihlásiť sa do aplikácie pomocou účtu z inej služby (napr. Google, Facebook) bez zdieľania hesla.

**Deprecated polia**

Polia v API alebo databáze, ktoré sú označené ako zastarané a plánuje sa ich odstránenie. Tester má overiť, že ich odstránenie neovplyvní funkčnosť.

**Screenshot**

Obrázok zachytávajúci aktuálny stav obrazovky alebo aplikácie. Používa sa ako dôkaz pri bug reportoch a dokumentácii testov.

**Layout**

Rozloženie prvkov na obrazovke alebo v dokumente. V testovaní sa kontroluje, či sa prvky zobrazujú správne na rôznych zariadeniach a rozlíšeniach.

**UI**

(User Interface) používateľské rozhranie aplikácie – všetko, s čím používateľ priamo interaguje. Tester overuje funkčnosť, vzhľad a intuitívnosť.

**UX**

(User Experience) celkový zážitok používateľa pri používaní aplikácie. Zahŕňa použiteľnosť, prístupnosť, rýchlosť a spokojnosť používateľa.

**Token**

Bezpečnostný reťazec znakov používaný na autentifikáciu a autorizáciu. Často nahrádza heslá pri komunikácii medzi klientom a serverom.

**Token**

V AI predstavuje token najmenšiu jednotku textu, s ktorou model pracuje – môže to byť celé slovo, časť slova alebo aj interpunkcia. Model rozdeľuje vstupy a výstupy na tokeny, čo ovplyvňuje dĺžku spracovania, cenu a rýchlost́ odpovede. Počet tokenov zároveň určuje, koľko textu model dokáže naraz spracovať.

**Multibyte znaky**

Znaky, ktoré sa ukladajú pomocou viac ako jedného bajtu (napr. á, 😊, 中文). Tester overuje, či aplikácia správne spracuje a zobrazí tieto znaky.

**Forgot password flow**

Proces, ktorým si používateľ obnoví zabudnuté heslo. Tester overuje bezpečnosť, funkčnosť a použiteľnosť tohto postupu.

**MVP**

(Minimum Viable Product) minimálna životaschopná verzia produktu s kľúčovými funkciami. Slúži na rýchle uvedenie na trh a zistenie spätnej väzby.

**MAU**

(Monthly Active Users) metrika počtu jedinečných používateľov, ktorí aplikáciu použili aspoň raz za mesiac. Pomáha sledovať popularitu produktu.

**PaaS**

(Platform as a Service) cloudová služba, ktorá poskytuje vývojové prostredie vrátane infraštruktúry, nástrojov a knižníc. Umožňuje rýchle nasadenie a škálovanie aplikácií.

**SEO**

(Search Engine Optimization) optimalizácia webovej stránky pre vyhľadávače. Tester môže overovať, či sú implementované správne meta tagy, URL a štruktúra obsahu.

**iOS**

Operačný systém od Apple určený pre mobilné zariadenia ako iPhone a iPad. Tester overuje kompatibilitu aplikácie s rôznymi verziami iOS a zariadeniami.

**Android**

Operačný systém od Google pre mobilné zariadenia, používaný širokou škálou výrobcov. Testovanie zahŕňa rôzne verzie systému, nadstavby výrobcov a veľkosti obrazoviek.

**BFF**

(Backend for Frontend) Architektonický vzor, pri ktorom sa vytvára špeciálna backendová vrstva prispôsobená potrebám konkrétneho frontendového klienta. Pomáha optimalizovať dáta a znížiť zložitosť na strane klienta.

**BFF (ohľadne contract testov)**

Backend for Frontend v kontexte contract testovania – testovanie, či API medzi BFF a frontendom spĺňa dohodnuté kontrakty.

**API Gateway**

Služba alebo komponent, ktorý sprostredkováva komunikáciu medzi klientmi a mikroservismi. Poskytuje centralizované riadenie autentifikácie, autorizácie, logovania a limitov požiadaviek.

**Rate-limit**

Mechanizmus obmedzujúci počet požiadaviek, ktoré môže klient odoslať na server v danom časovom intervale. Slúži na ochranu pred zneužitím a preťažením systému.

**ML pipeline**

Sústava krokov na spracovanie dát a trénovanie modelov strojového učenia – od získania dát až po nasadenie modelu. V testovaní sa overuje správnosť jednotlivých fáz a ich integrácia.

**NSFW/duplikáty**

Označenie pre obsah „Not Safe For Work“ (nevhodný do práce) alebo pre detekciu duplicitného obsahu. Používa sa pri filtrovaní a moderovaní dát.

**Funnel**

Model, ktorý znázorňuje, ako používatelia prechádzajú jednotlivými krokmi procesu (napr. nákupný proces). Tester môže sledovať, kde dochádza k odpadnutiu používateľov.

**Cohorty**

Skupiny používateľov rozdelené podľa spoločnej charakteristiky (napr. dátum registrácie) na účely analýzy správania. Pomáhajú pri hodnotení úspešnosti zmien alebo kampaní.

**Elasticsearch**

Vyhľadávací a analytický nástroj založený na Lucene. Používa sa na rýchle fulltextové vyhľadávanie a analýzu veľkých objemov dát.

**OpenSearch**

Open-source vyhľadávací a analytický nástroj, ktorý vznikol ako vetva Elasticsearch. Používa sa na podobné účely – indexovanie a rýchle vyhľadávanie dát.

**Meilisearch**

Ľahký a rýchly open-source vyhľadávací engine s jednoduchou integráciou. Hodí sa pre aplikácie, kde je potrebné rýchle a relevantné vyhľadávanie.

**CDN**

(Content Delivery Network) sieť serverov, ktoré distribuujú obsah bližšie k používateľom podľa ich geografickej polohy. Znižuje latenciu a zlepšuje dostupnosť obsahu.

**Cache feed**

Uložená verzia dátového kanála (feedu), ktorá umožňuje rýchlejšie načítanie obsahu. Tester overuje správnu aktualizáciu cache a jej vyprázdnenie pri zmene obsahu.

**RabbitMQ**

Open-source systém na riadenie správ (message broker), ktorý sprostredkováva komunikáciu medzi rôznymi časťami aplikácie. Tester kontroluje správne odosielanie, prijímanie a spracovanie správ.

**SQS**

(Amazon Simple Queue Service) spravovaná služba od AWS na riadenie front správ medzi aplikáciami. Umožňuje asynchrónnu komunikáciu a škálovanie bez potreby spravovať vlastný message broker.

**Remote config**

Mechanizmus, ktorý umožňuje meniť nastavenia aplikácie na diaľku bez potreby vydania novej verzie. Tester overuje, či sa nové hodnoty správne načítajú a aplikujú.

**LaunchDarkly**

Platforma na správu feature flagov a experimentov. Umožňuje vývojárom nasadzovať a zapínať/vypínať funkcie bez zásahu do kódu.

**Observabilita**

Schopnosť systému poskytovať dostatok informácií (logy, metriky, trasy požiadaviek) na pochopenie jeho stavu a príčin problémov. Tester ju využíva pri analýze chýb a monitorovaní výkonu.

**OIDC**

(OpenID Connect) autentifikačný protokol nad OAuth 2.0, umožňujúci prihlásenie cez externých poskytovateľov identity, ako sú Apple alebo Google.

**Krátke JWT + refresh tokeny v secure úložisku klienta**

(JSON Web Token) implementácia autentifikácie, kde sa používa krátko platný JSON Web Token a dlhšie platný refresh token, uložený bezpečne na strane klienta (napr. Secure Enclave, Keychain).

**Signed URLs pre upload/download médií**

Predpísané (podpísané) URL adresy, ktoré umožňujú dočasný prístup k nahraniu alebo stiahnutiu súboru bez priameho sprístupnenia súboru verejne.

**WAF**

(Web Application Firewall) bezpečnostná vrstva chrániaca webové aplikácie pred útokmi ako SQL injection, XSS či DDoS. Filtruje a blokuje škodlivé požiadavky.

**Mikroslužby (scale-out)**

Architektonický prístup, kde sa aplikácia skladá z viacerých nezávislých služieb. „Scale-out“ znamená horizontálne škálovanie pridaním ďalších inštancií služby.

**JVM**

(Java Virtual Machine) virtuálne prostredie, ktoré spúšťa Java aplikácie a zabezpečuje ich prenositeľnosť medzi platformami.

**Kotlin Multiplatform Mobile (KMM)**

Rámec od JetBrains umožňujúci zdieľanie logiky aplikácie medzi Androidom a iOS-om pri zachovaní natívnych UI.

**SwiftUI Compose bridge**

Prepojenie medzi SwiftUI (Apple) a Jetpack Compose (Google) na zdieľanie alebo premostenie komponentov pri multiplatformovom vývoji.

**NFR**

(Non-Functional Requirements-nefunkčné požiadavky) špecifikácie, ktoré neurčujú funkcie systému, ale jeho kvalitatívne vlastnosti, ako je výkon, bezpečnosť alebo škálovateľnosť.

**Monorepo**

Vývojová stratégia, pri ktorej sa kód viacerých projektov alebo služieb udržiava v jednom spoločnom repozitári.

**Polyrepo**

Opak monorepa – každý projekt alebo služba má vlastný samostatný repozitár. Umožňuje nezávislé riadenie a nasadzovanie.

**SAST**

(Static Application Security Testing) metóda testovania bezpečnosti aplikácie analýzou jej zdrojového kódu alebo binárky bez spustenia. Pomáha odhaliť zraniteľnosti už počas vývoja

**Canary/blue-green nasadenia**

Strategické spôsoby nasadzovania nových verzií aplikácií. Canary nasadenie nasadzuje zmenu najprv malej skupine používateľov, blue-green má dve prostredia, medzi ktorými sa prepína.

**Terraform**

Nástroj na deklaratívnu správu infraštruktúry ako kódu. Umožňuje definovať a automatizovať vytváranie cloudových a on-premise prostredí.

**Vault/SM**

Služby a nástroje na bezpečné ukladanie a správu citlivých údajov (tajomstiev), ako sú heslá, API kľúče či certifikáty.

**Guardrail na postupné rollouty**

Mechanizmus, ktorý zabraňuje rýchlemu alebo nekontrolovanému nasadeniu funkcie všetkým používateľom naraz, aby sa minimalizovalo riziko chýb.

**RPS**

(Requests Per Second) metrika udávajúca, koľko požiadaviek spracuje server za sekundu. Používa sa na hodnotenie výkonu a kapacity systému.

**Cold/warm cache**

Cold cache je prázdna vyrovnávacia pamäť (bez uložených dát), warm cache už obsahuje uložené dáta a poskytuje rýchlejšie odpovede.

**DAST**

(Dynamic Application Security Testing) testovanie bezpečnosti aplikácie počas jej behu. Simuluje útoky z pohľadu útočníka, aby sa identifikovali zraniteľnosti.

**Mobilné MASVS checky**

Kontrolný zoznam Mobile Application Security Verification Standard – slúži na overenie bezpečnostných opatrení mobilných aplikácií.

**MASVS**  
(Mobile Application Security Verification Standard) je štandard pre overovanie bezpečnosti mobilných aplikácií. Obsahuje úrovne a kontrolné body, ktoré pomáhajú testerom systematicky hodnotiť bezpečnosť aplikácií.

**ASVS**  
(Application Security Verification Standard) je štandard OWASP pre testovanie bezpečnosti webových aplikácií. Poskytuje zoznam požiadaviek a odporúčaní na rôznych úrovniach zabezpečenia.

**Scope creep**

Postupné rozširovanie rozsahu projektu nad pôvodný plán bez adekvátnej úpravy času a zdrojov. V testovaní to môže viesť k nárastu práce a oneskoreniu dodania.

**Nice-to-have**

Požiadavky alebo funkcie, ktoré nie sú nevyhnutné, ale bolo by dobré ich mať. Majú nižšiu prioritu než „must-have“ funkcie.

**Must-have**

označuje funkcie alebo požiadavky, ktoré sú nevyhnutné na správne fungovanie produktu alebo splnenie cieľov projektu. Bez ich implementácie by bol produkt nekompletný alebo nepoužiteľný. V testovaní majú najvyššiu prioritu a ich chýbanie môže zastaviť nasadenie.

**JS/TS doména**

Časť projektu alebo kódu implementovaná v JavaScripte alebo TypeScripte, ktorá má konkrétnu zodpovednosť (napr. správa UI, dátová vrstva).

**SDK**

(Software Development Kit) balík nástrojov, knižníc a dokumentácie na vývoj aplikácií pre konkrétnu platformu alebo službu.

**IDE**

(Integrated Development Environment) integrované vývojové prostredie poskytujúce editor kódu, ladiace nástroje a ďalšie pomôcky na uľahčenie vývoja.

**RPC**

(Remote Procedure Call) je mechanizmus, ktorý umožňuje programu spúšťať funkcie alebo procedúry na inom počítači či serveri, akoby boli lokálne. Skrýva detaily sieťovejkomunikácie, takže vývojár pracuje s volaním vzdialených služieb podobne ako s bežnými funkciami v kóde. Používa sa na jednoduchú integráciu distribuovaných systémov.

**gRPC**

Vysokovýkonné, open-source RPC rozhranie od Google založené na HTTP/2 a Protobuf. Používa sa na komunikáciu medzi službami, pričom poskytuje rýchly prenos dát a podporu pre viacero jazykov.

**ACID**

Súbor vlastností databázových transakcií – Atomicita, Konzistencia, Izolácia a Trvácnosť (Durability). Zabezpečujú, že transakcie budú spracované spoľahlivo a predvídateľne.

**MVCC**

(Multi-Version Concurrency Control) mechanizmus riadenia súbehu v databázach, ktorý umožňuje viacerým transakciám čítať dáta bez blokovania zápisov, vďaka čomu sa znižuje riziko konfliktov.

**JSONB**

Dátový typ v PostgreSQL, ktorý umožňuje efektívne ukladanie a vyhľadávanie JSON dát vo formáte binárneho úložiska. Zlepšuje výkon oproti bežnému JSON.

**ORM**

(Object-Relational Mapping) technika, ktorá mapuje objekty v kóde na tabuľky v relačnej databáze. Uľahčuje prácu s databázami bez písania SQL dotazov.

**TypeORM/Prisma**

Moderné ORM nástroje pre JavaScript/TypeScript aplikácie. Umožňujú jednoduchú prácu s databázami a generovanie typovo bezpečných dotazov.

**PITR**

(Point In Time Recovery) proces obnovy databázy do konkrétneho okamihu v minulosti. Používa sa pri haváriách alebo chybách na minimalizáciu straty dát.

**PostgreSQL**

Výkonný open-source relačný databázový systém s podporou rozšírených dátových typov a pokročilých funkcií. Je obľúbený pre svoju stabilitu a flexibilitu.

**MySQL**

Populárny open-source relačný databázový systém. Často používaný v webových aplikáciách, známy pre svoju jednoduchosť a širokú podporu.

**MSSQL**

(Microsoft SQL Server) komerčný relačný databázový systém od Microsoftu. Poskytuje široké možnosti integrácie s produktmi Microsoftu.

**SQL**

(Structured Query Language) štandardizovaný jazyk na správu a manipuláciu s dátami v relačných databázach. Používa sa na vytváranie, čítanie, aktualizáciu a mazanie dát.

**MariaDB**

Open-source vetva MySQL vyvíjaná komunitou, zameraná na vyšší výkon, otvorenosť a kompatibilitu s MySQL.

**CockroachDB**

**Distribuovaná SQL databáza odolná voči výpadkom. Navrhnutá na horizontálne škálovanie a automatickú replikáciu dát.**

**SQLLite**

**Ľahká embedded SQL databáza uložená v jednom súbore. Nepotrebuje samostatný server, často používaná v mobilných a desktopových aplikáciách.**

**SQLAlchemy**

Populárna Python knižnica na ORM a prácu s databázami. Poskytuje vysokú flexibilitu pri tvorbe dotazov a správe schém.

**Kafka**

Distribuovaná platforma na spracovanie prúdových dát v reálnom čase. Umožňuje publikovať, odoberať a uchovávať veľké objemy správ s vysokou priepustnosťou a nízkou latenciou.

**Tosca**

Nástroj na automatizované testovanie podnikových aplikácií, známy pre modelovo-riadený prístup. Umožňuje testovať webové, desktopové, mobilné aj API rozhrania bez potreby rozsiahleho programovania.

**Jira**

Populárny nástroj na riadenie projektov a sledovanie chýb. Používa sa na plánovanie úloh, sledovanie progresu a komunikáciu v tíme, často v spojení s metodikami Agile/Scrum.

**Xray**

Rozšírenie pre Jira (platené) určené na správu testov a testovacích cyklov. Umožňuje prepojiť požiadavky, testovacie prípady, výsledky testov a reportovanie.

**AWS**

(Amazon Web Services) cloudová platforma poskytujúca širokú škálu služieb – od výpočtových kapacít cez databázy až po AI/ML nástroje. Používa sa na nasadzovanie a prevádzku aplikácií v cloude.

**Microsoft Azzure**

Cloudová platforma od Microsoftu poskytujúca infraštruktúru, platformu aj softvérové služby. Podporuje integráciu s produktmi Microsoftu a hybridné riešenia.

**Thumbnail**

Zmenšená verzia obrázka alebo videa, používaná ako náhľad. V testovaní sa overuje kvalita, rozmery a správne načítanie v rôznych zariadeniach.

**NSFW klasifikátor**

Algoritmus na detekciu obsahu „Not Safe For Work“ (nevhodného na pracovisku), ako je napríklad nahota alebo násilie. Používa sa na filtrovanie alebo označovanie obsahu.

**AWS Cognito**

Služba od AWS na správu používateľských účtov, autentifikácie a autorizácie. Podporuje registráciu, prihlásenie aj viacfaktorové overenie.

**Ory/Keycloak (self-host)**

Open-source riešenia na správu identít a prístupov, ktoré si organizácia môže prevádzkovať vo vlastnej infraštruktúre. Podporujú OIDC, OAuth2 a ďalšie štandardy.

**Sociálne prihlásenia**

Možnosť prihlásiť sa do aplikácie pomocou účtu z inej služby (napr. Facebook, Google, LinkedIn). Zjednodušuje registráciu, no vyžaduje správne zabezpečenie.

**Active Directory**

(AD) je adresárová služba od Microsoftu na správu používateľov, skupín, počítačov a ďalších zdrojov v rámci siete organizácie. Umožňuje centralizovanú autentifikáciu a autorizáciu, často v kombinácii so Single Sign-On (SSO), aby sa používatelia po prihlásení do domény mohli automaticky dostať aj k ďalším interným službám. Používa sa najmä vo firemných a školských prostrediach.

**Docker**

Platforma na tvorbu, distribúciu a spúšťanie aplikácií v kontajneroch. Kontajnery izolujú aplikáciu a jej závislosti, čo uľahčuje nasadzovanie a škálovanie.

**GitHub**

Online platforma na správu kódu a verzií pomocou Git. Ponúka nástroje na spoluprácu, správu projektov, CI/CD a bezpečnostné funkcie.

**GitLab**

Platforma na správu kódu a DevOps procesov. Okrem Git repozitára obsahuje nástroje na plánovanie, CI/CD, monitoring a bezpečnostné analýzy.

**Bitbucket**  
Platforma od Atlassianu na správu Git repozitárov, úzko integrovaná s nástrojmi Jira, Confluence a Trello. Podporuje súkromné aj verejné repozitáre a ponúka zabudovaný CI/CD systém Bitbucket Pipelines.

**Azure DevOps Repos**  
Súčasť balíka Azure DevOps od Microsoftu, poskytuje Git repozitáre aj nástroje na riadenie projektov, testovanie a CI/CD. Je silno prepojená s ďalšími službami Azure a vhodná pre enterprise prostredia.

**AWS CodeCommit**  
Plne spravovaná cloudová služba od Amazonu na hosťovanie súkromných Git repozitárov. Ponúka vysokú bezpečnosť, škálovateľnosť a integráciu s ďalšími AWS službami, ako sú CodeBuild či CodePipeline.

**SourceForge**  
Jedna z najstarších platforiem na hostovanie open-source projektov, podporuje Git, SVN aj Mercurial. Poskytuje úložisko kódu, správu chýb, diskusné fóra a distribúciu binárnych súborov.

**Gitea**  
Ľahká a rýchla open-source platforma na správu Git repozitárov, ktorú si môžeš prevádzkovať sama. Ponúka webové rozhranie, správu úloh a jednoduché CI integrácie.

**Gogs**  
Minimalistická open-source Git platforma navrhnutá na nízku spotrebu zdrojov a jednoduchú inštaláciu. Vhodná pre malé tímy alebo projekty, ktoré potrebujú základné funkcie Git servera.

**Phabricator**  
Open-source balík nástrojov od Facebooku na správu kódu, úloh a dokumentácie. Hoci jeho vývoj bol ukončený, stále sa používa v niektorých organizáciách, ktoré si ho prevádzkujú samostatne.

**Assembla**  
Platforma zameraná na komerčné tímy, podporuje Git, Subversion (SVN) a Perforce. Ponúka hostovanie kódu, správu úloh a nástroje pre spoluprácu, vhodné pre väčšie vývojové tímy.

**Grafana**

Open-source platforma na vizualizáciu a monitoring dát z rôznych zdrojov (databázy, metriky, logy). Umožňuje vytvárať interaktívne dashboardy, ktoré pomáhajú pri sledovaní výkonu aplikácií a infraštruktúry.

**GDPR**

(General Data Protection Regulation) nariadenie Európskej únie o ochrane osobných údajov. Stanovuje pravidlá pre spracovanie, ukladanie a zdieľanie osobných údajov občanov EÚ.

**AI Act**

Pripravované nariadenie EÚ, ktoré má regulovať používanie umelej inteligencie. Zameriava sa na bezpečnosť, transparentnosť a zodpovednosť pri vývoji a nasadzovaní AI systémov.

**WCAG**

(Web Content Accessibility Guidelines) súbor odporúčaní na zlepšenie prístupnosti webového obsahu pre osoby so zdravotným znevýhodnením. Stanovuje zásady, ako má byť obsah vnímateľný, ovládateľný, zrozumiteľný a robustný.

**EAA**

(European Accessibility Act) európska smernica, ktorá stanovuje požiadavky na prístupnosť produktov a služieb pre osoby so zdravotným postihnutím. Platí pre rôzne odvetvia, vrátane IT a e-commerce.

**Izolácia tajomstiev**

(Secrets Manager/Vault) **m**echanizmus na bezpečné ukladanie a správu citlivých údajov, ako sú API kľúče, heslá a certifikáty. Zabraňuje ich úniku a umožňuje riadiť prístup len oprávneným používateľom alebo službám.

**TypeScript**

Programovací jazyk nadstavujúci JavaScript o statické typovanie. Pomáha predchádzať chybám počas vývoja a uľahčuje udržiavateľnosť kódu vo väčších projektoch.

**Python**

Vysokoúrovňový, interpretovaný programovací jazyk známy pre svoju jednoduchosť a čitateľnosť. Používa sa na webový vývoj, analýzu dát, strojové učenie aj automatizáciu.

**Java**

Objektovo orientovaný programovací jazyk navrhnutý na prenositeľnosť medzi platformami. Využíva sa pri vývoji enterprise aplikácií, mobilných aplikácií (Android) a backend systémov.

**JavaScript**

Skriptovací jazyk používaný primárne na tvorbu interaktívneho obsahu na webových stránkach. Spúšťa sa v prehliadači aj na serveri (Node.js).

**Ruby**

Dynamický, interpretovaný programovací jazyk známy pre svoju jednoduchosť a produktivitu. Často sa používa s frameworkom Ruby on Rails na vývoj webových aplikácií.

**C**

Jazyk nižšej úrovne vhodný na systémové programovanie a vývoj aplikácií s vysokou výkonnosťou. Mnohé moderné jazyky (napr. C++, Java) z neho čerpajú základy.

**C++**

Rozšírenie jazyka C o objektovo orientované prvky. Používa sa na vývoj softvéru s vysokými nárokmi na výkon, ako sú hry alebo embedded systémy.

**C#**

Programovací jazyk od Microsoftu, ktorý kombinuje jednoduchosť C a C++ s modernými funkciami. Často sa používa v .NET prostredí na vývoj webových, desktopových a mobilných aplikácií.

**Asembler**

Nízkoúrovňový jazyk, ktorý priamo zodpovedá inštrukciám procesora. Používa sa na vývoj systémového softvéru a optimalizáciu výkonu na hardvérovej úrovni.

**Golang (Go)**

Moderný, kompilovaný programovací jazyk vyvinutý v Google, známy pre svoju jednoduchosť, rýchle kompilácie a podporu paralelizmu cez gorutiny. Používa sa na vývoj backendových služieb, cloudovej infraštruktúry a nástrojov.

**MS-DOS**(Microsoft Disk Operating System) starší operačný systém s textovým rozhraním, používaný najmä v 80. a 90. rokoch. Slúžil ako základ pre spúšťanie aplikácií pred nástupom grafických systémov ako Windows.

**Basic**

(Beginner’s All-purpose Symbolic Instruction Code) jednoduchý programovací jazyk navrhnutý na výučbu programovania. V minulosti často používaný na domácich počítačoch v 80. rokoch.

**Rust**

Moderný programovací jazyk zameraný na výkon, bezpečnosť pamäte a prevenciu chýb pri súbežnom spracovaní. Vhodný na systémové programovanie aj webové aplikácie.

**Fortran**

(Formula Translation) jeden z najstarších programovacích jazykov, stále používaný v oblasti vedeckých a numerických výpočtov. Známy pre vysoký výkon pri práci s číselnými dátami.

**COBOL**

(Common Business-Oriented Language) programovací jazyk navrhnutý pre obchodné aplikácie a spracovanie veľkých objemov dát. Dodnes používaný v bankových a poisťovacích systémoch.

**Pascal**

Programovací jazyk vytvorený na výučbu štruktúrovaného programovania. V minulosti sa používal aj na vývoj desktopových aplikácií.

**PHP**

(PHP: Hypertext Preprocessor) skriptovací jazyk určený na tvorbu dynamických webových stránok a backendových aplikácií. Beží na strane servera a často sa kombinuje s databázami ako MySQL.

**Podpora ICU kolácií**

Možnosť databázy alebo aplikácie používať ICU (International Components for Unicode) na správne triedenie a porovnávanie textu podľa jazykových pravidiel. Je dôležité pri práci s viacerými jazykmi a znakmi.

**YugabyteDB**

Distribuovaná SQL databáza s kompatibilitou s PostgreSQL a Cassandra API. Navrhnutá na vysokú dostupnosť, horizontálne škálovanie a prácu v cloude.

**MongoDB**

Dokumentovo orientovaná NoSQL databáza, ktorá ukladá dáta vo formáte podobnom JSON. Vhodná na flexibilné schémy a rýchly vývoj aplikácií.

**DynamoDB**

Plne spravovaná NoSQL databáza od AWS optimalizovaná na rýchlosť a škálovanie. Podporuje kľúčovo-hodnotové aj dokumentové úložisko.

**Neo4j**

Grafová databáza optimalizovaná na ukladanie a dotazovanie vzťahov medzi entitami. Používa sa v odporúčacích systémoch, sieťových analýzach a detekcii podvodov.

**ClickHouse**

Rýchla open-source analytická databáza pre spracovanie veľkých objemov dát v reálnom čase. Vhodná na OLAP dotazy a business intelligence.

**Stemming**

Technika spracovania textu, pri ktorej sa slová redukujú na ich kmeň (napr. „bežal“ → „bež“). Používa sa vo vyhľadávačoch a NLP na zlepšenie zhody výsledkov.

**Syntetické TPS grafy**

Vizualizácie zobrazujúce počet transakcií za sekundu (Transactions Per Second) vytvorených umelo, napr. simuláciou záťaže. Slúžia na testovanie výkonu systému bez použitia reálnych dát. Tester sleduje tvary kriviek, výkyvy a limity systému.

**Anti-abuse**

Opatrenia a mechanizmy na ochranu systému pred zneužitím, ako sú spam, podvodné registrácie či automatizované útoky. Môžu zahŕňať detekciu anomálií, blokovanie IP alebo obmedzovanie rýchlosti požiadaviek.

**Bayesovou korekciou**

Štatistická metóda úpravy výsledkov (napr. hodnotenia) na základe Bayesovej pravdepodobnosti. Pomáha znížiť vplyv malých vzoriek – napr. nová položka s jedným 5★ hodnotením nebude hneď na vrchole rebríčka.

**Wilsonovo skóre**

Výpočet intervalov spoľahlivosti pre podiely, napr. pre percento pozitívnych hodnotení. Používa sa na spravodlivejšie zoradenie položiek pri nerovnakom počte hodnotení.

**Shadow-ban**

Tichý zákaz, pri ktorom používateľ stále vidí svoj obsah, ale iní ho nevidia. Slúži na potlačenie škodlivého správania bez upozornenia pôvodcu.

**Decay**

Mechanizmus postupného znižovania váhy starších dát alebo hodnotení. Umožňuje algoritmom preferovať novšie a relevantnejšie informácie.

**Idempotentnosť**

Vlastnosť operácie, pri ktorej jej opakované vykonanie vedie k rovnakému výsledku ako prvé vykonanie. Dôležitá pri API, aby nedošlo k duplicitám (napr. opakovaná platba).

**Re-compute job po batchi zmien**

Úloha, ktorá sa spustí po hromadnej zmene dát a znovu prepočíta odvodené hodnoty alebo agregáty. Tester overuje, že výpočty prebehli správne a sú aktuálne.

**DAU**

(Daily Active Users) – počet unikátnych používateľov, ktorí použili aplikáciu aspoň raz za deň. Sleduje sa ako metrika zapojenia a rastu.

**Survivorship bias**

Kognitívne skreslenie, keď vyhodnocujeme len „preživšie“ prípady a ignorujeme tie, ktoré zlyhali. V testovaní môže viesť k nesprávnym záverom, ak analyzujeme len úspešné scenáre.

**Redis cache agregátov**

Ukladanie vypočítaných agregovaných hodnôt (napr. počtu lajkov) v Redis databáze pre rýchly prístup. Tester sleduje, či sa cache aktualizuje pri zmene dát.

**Antimissuse**

Opatrenia proti nesprávnemu použitiu systému, aj keď nejde o úmyselné zneužitie. Môže zahŕňať ochranu pred chybnými vstupmi, neštandardnými požiadavkami alebo nadmerným zaťažením.

**Facetové štatistiky**

Rozdelenie dát podľa kategórií alebo filtrov (facets) – napr. počet produktov podľa značky, farby či ceny. Tester overuje správnosť počtov a reakcie na kombinácie filtrov.

**E2E testy**

(End-to-End) testy, ktoré overujú celú cestu používateľa alebo procesu od začiatku po koniec, vrátane všetkých integrovaných komponentov. Cieľom je simulovať reálne použitie.

**Managed IdP**

(Managed Identity Provider) – služba tretej strany, ktorá spravuje autentifikáciu a identitu používateľov. Umožňuje organizáciám využiť cloudové riešenia namiesto vlastnej infraštruktúry na správu účtov a prístupov.

**Rotácia signing keys v IdP**  
Pravidelná výmena podpisových kľúčov v poskytovateľovi identity na zvýšenie bezpečnosti. Zabraňuje zneužitiu kompromitovaných kľúčov.

**Authorization Code Flow**

Autentifikačný tok používaný v OAuth 2.0, pri ktorom klient najprv získa autorizačný kód a až následne ho vymení za prístupový token. Zvyšuje bezpečnosť, pretože token sa neposiela cez prehliadač používateľa.

**PKCE (S256)**

(Proof Key for Code Exchange) je rozšírenie OAuth 2.0 na bezpečnejšie získanie prístupového tokenu v klientskej aplikácii. Metóda S256 využíva SHA-256 hashovanie pre overenie výmeny kódu.

**URI**

(Uniform Resource Identifier) je reťazec, ktorý jednoznačne identifikuje zdroj na internete alebo v systéme. Môže byť vo forme URL (adresy) alebo URN (názvu zdroja).

**URN**  
(Uniform Resource Name) je typ URI, ktorý slúži na jednoznačnú identifikáciu zdroja podľa názvu, nie podľa jeho umiestnenia. Na rozdiel od URL neobsahuje informáciu o tom, kde sa zdroj nachádza, ale len jeho trvalý identifikátor.

**Bearer access token**

Typ prístupového tokenu, ktorý poskytuje držiteľovi („bearerovi“) plný prístup k zdrojom, na ktoré bol vydaný. Musí byť chránený, pretože ktokoľvek s tokenom môže získať prístup.

**Hash RT**

Hashovanie refresh tokenu pred uložením do databázy na ochranu pred jeho zneužitím. Aj keby databáza unikla, útočník nedostane platný token.

**Passwordless**

Metóda prihlasovania, ktorá nevyžaduje heslo, ale používa alternatívne mechanizmy ako e-mailové odkazy alebo biometrické overenie. Znižuje riziko odcudzenia hesiel.

**Magic link**

Prihlasovací odkaz zaslaný používateľovi (napr. e-mailom), po kliknutí na ktorý sa okamžite prihlási. Jednoduché na použitie, ale časovo obmedzené kvôli bezpečnosti.

**Passkeys**

Moderná náhrada hesiel založená na kryptografických kľúčoch uložených v zariadení používateľa. Umožňuje bezpečné a rýchle prihlásenie bez zadávania hesla.

**Argon2id**

Pokročilá hashovacia funkcia pre heslá, odolná voči útokom hrubou silou aj útokom pomocou GPU. Kombinuje vlastnosti algoritmov Argon2i a Argon2d.

**Bcrypt s costom**

Hashovacia funkcia pre heslá, pri ktorej parameter „cost“ určuje náročnosť výpočtu. Vyšší cost zvyšuje bezpečnosť, ale aj čas spracovania.

**Unikátne salting**

Pridanie náhodnej hodnoty (saltu) ku každému heslu pred jeho hashovaním, aby sa zabránilo útokom s rainbow tabuľkami. Každý používateľ má vlastný unikátny salt.

**Rainbow tabuľky**  
Predvypočítané tabuľky hashov pre veľké množstvo možných hesiel, používané na rýchle prelomenie hashovaných údajov. Útoky s rainbow tabuľkami sa dajú efektívne zmierniť použitím saltovania pri hashovaní hesiel.

**Minimá + haveibeenpwned kontrola reťazcov**

Overenie, či heslo spĺňa minimálne požiadavky na dĺžku a komplexitu, doplnené o kontrolu proti databáze uniknutých hesiel služby haveibeenpwned. Zvyšuje to odolnosť voči opätovnému použitiu kompromitovaných hesiel.

**MFA**

(Multi-Factor Authentication) je viacfaktorová autentifikácia vyžadujúca kombináciu aspoň dvoch rôznych overovacích faktorov. Zvyšuje bezpečnosť prístupu k účtom.

**TOTP**

(Time-based One-Time Password) je jednorazové heslo generované na základe času a zdieľaného tajomstva. Používa sa najmä v aplikáciách pre dvojfaktorovú autentifikáciu.

**WebAuthn**

(Web Authentication) je webový štandard pre bezheslové prihlasovanie pomocou verejnej kryptografie. Umožňuje overenie pomocou biometriky, bezpečnostných kľúčov alebo zariadení.

**RBAC**

(Role-Based Access Control) je model riadenia prístupu, kde sa oprávnenia prideľujú podľa role používateľa. Zjednodušuje správu prístupov a znižuje riziko neoprávneného prístupu.

**ABAC**

(Attribute-Based Access Control) je model riadenia prístupu, ktorý rozhoduje na základe atribútov používateľa, zdroja a kontextu. Umožňuje jemnejšie nastavenie pravidiel než RBAC.

**Claimy do access tokenu**

Dáta (claimy) vložené do prístupového tokenu, ktoré obsahujú informácie o používateľovi alebo relácii. Používajú sa na autorizáciu a overovanie požiadaviek.

**Exponenciálny backoff**

Technika riadenia opakovaných pokusov o spojenie, pri ktorej sa interval medzi pokusmi postupne zvyšuje. Znižuje zaťaženie systému pri dočasných problémoch.

**IP allow/deny list pre admin endpoints**

Bezpečnostné opatrenie, ktoré povoľuje alebo blokuje prístup k administrátorskym rozhraniam podľa IP adries. Pomáha obmedziť prístup len na dôveryhodné adresy.

**PII**

(Personally Identifiable Information) sú údaje, ktoré môžu identifikovať konkrétnu osobu. Zahŕňajú napríklad meno, adresu, rodné číslo alebo e-mail.

**JWKS podpisu**

(JSON Web Key Set) je formát na distribúciu verejných kľúčov používaných na overenie podpisu tokenov. Umožňuje klientom automaticky získavať aktuálne kľúče od servera.

**Bruteforce scenáre**

Útoky, pri ktorých útočník systematicky skúša rôzne kombinácie prihlasovacích údajov, kým nenájde správnu. Testovanie zahŕňa overenie, či systém takéto pokusy deteguje a blokuje.

**CSRF**

(Cross-Site Request Forgery) je útok, pri ktorom útočník prinúti používateľa vykonať nechcenú akciu na webovej aplikácii, kde je prihlásený. Prevencia zahŕňa kontrolu tokenov a pôvodu požiadaviek.

**SSRF/redirect hijack**

(Server-Side Request Forgery) je útok, pri ktorom útočník prinúti server vykonať požiadavku na iný server. Redirect hijack zneužíva presmerovania na získanie citlivých údajov alebo presmerovanie používateľa.

**Pre-podpísané URL**

Odkaz s dočasnou platnosťou, ktorý umožňuje prístup k súboru alebo zdroju bez ďalšieho overovania. Používa sa napríklad na bezpečné zdieľanie súborov v cloude.

**MIME**

(Multipurpose Internet Mail Extensions) je štandard, ktorý určuje typ obsahu súborov prenášaných cez internet. Pomáha klientom správne interpretovať text, obrázky, video alebo iné dáta.

**POP3**  
(Post Office Protocol version 3) je protokol na sťahovanie e-mailov zo servera do poštového klienta. Správy sa zvyčajne po stiahnutí zo servera odstránia, čo šetrí miesto na serveri.

**IMAP**  
(Internet Message Access Protocol) je protokol na prácu s e-mailami priamo na serveri. Umožňuje synchronizáciu priečinkov a správ medzi viacerými zariadeniami bez nutnosti sťahovať ich lokálne.

**Pipeline**

Sekvencia krokov, cez ktoré prechádzajú dáta alebo kód od vstupu po výsledok. V softvérovom vývoji sa používa na automatizáciu buildov, testovania a nasadzovania.

**EXIF**

(Exchangeable Image File Format) je dátový formát pre ukladanie metadát v obrázkoch, napríklad dátum, čas alebo GPS súradnice. Často sa používa pri digitálnej fotografii.

**HEIC**

(High Efficiency Image Coding) je moderný formát obrázkov založený na kodeku HEVC, ktorý ponúka vysokú kvalitu pri menšej veľkosti súboru. Bežne sa používa v zariadeniach Apple.

**AVIF/WebP**

Formáty obrázkov optimalizované pre web, ktoré poskytujú menšiu veľkosť súboru pri zachovaní kvality. AVIF využíva kodek AV1, WebP vyvinul Google pre rýchlejšie načítanie webu.

**JPEG**

(Bežne Joint Photographic Experts Group) je rozšírený formát kompresie obrázkov s miernou stratou kvality. Je vhodný na fotografie, ale menej na obrázky s ostrými hranami.

**AV sken**

Antivírusová kontrola súborov alebo systémov na prítomnosť škodlivého kódu. Môže byť vykonaná manuálne alebo automaticky v rámci bezpečnostných nástrojov.

**ClamAV**

Open-source antivírusový program určený na detekciu trójskych koní, vírusov a iného malvéru. Často sa používa na serveroch a v e-mailových bránach.

**Managed AV**

Antivírusová ochrana poskytovaná a spravovaná treťou stranou. Umožňuje organizáciám outsourcovať správu bezpečnostného softvéru a jeho aktualizácií.

**pHash**

(Perceptual Hash) je metóda hashovania, ktorá vytvára odtlačok obrázka podľa jeho vizuálnych vlastností. Umožňuje porovnávať obrázky aj pri malých zmenách, napríklad pri zmene veľkosti.

**Vendor lock-in**

Situácia, keď je zákazník závislý na jednom dodávateľovi technológie alebo služby a prechod ku konkurencii je nákladný alebo zložitý. Znižuje flexibilitu a môže zvyšovať náklady.

**Smart crop**

Automatické orezanie obrázka tak, aby sa zachovala najdôležitejšia časť. Využíva analýzu obsahu, napríklad detekciu tvárí alebo objektov.

**DPA**

(Data Processing Agreement) je zmluva medzi prevádzkovateľom a spracovateľom údajov, ktorá definuje podmienky spracúvania osobných údajov. Je požadovaná legislatívou, napríklad GDPR.

**GPS**

(Global Positioning System) je satelitný navigačný systém umožňujúci určenie polohy kdekoľvek na Zemi. Používa sa v navigácii, lokalizačných službách a sledovaní zariadení.

**SSIM/PSNR**

(Structural Similarity Index / Peak Signal-to-Noise Ratio) sú metriky na hodnotenie kvality obrázka po kompresii alebo spracovaní. SSIM sa zameriava na vizuálnu podobnosť, PSNR na pomer signálu k šumu.

**Metadáta**  
Údaje o iných údajoch, ktoré opisujú obsah, pôvod, štruktúru alebo formát dátového súboru. Využívajú sa napríklad na vyhľadávanie, organizáciu alebo správu obsahu.

**Malvér**  
(Škodlivý softvér) je softvér navrhnutý s cieľom poškodiť, zneužiť alebo neoprávnene získať prístup k systémom či dátam. Patrí sem napríklad vírus, trójsky kôň, ransomware alebo spyware.

**SLA**  
(Service Level Agreement) je zmluvná dohoda medzi poskytovateľom služby a zákazníkom, ktorá stanovuje úroveň poskytovaných služieb. Obsahuje metriky, ako napríklad dostupnosť, čas odozvy a sankcie pri nedodržaní.

**MD (man day)**  
Jednotka odhadu práce vyjadrujúca množstvo práce, ktoré vykoná jedna osoba za jeden pracovný deň. Používa sa pri plánovaní projektov a odhadovaní nákladov.

**NDA**  
(Non-Disclosure Agreement) je dohoda o mlčanlivosti, ktorá zaväzuje strany zachovať dôverné informácie v tajnosti. Často sa podpisuje pred začiatkom spolupráce alebo výmenou citlivých údajov.

**PM (Project Manager)**  
Projektový manažér zodpovedný za plánovanie, koordináciu a riadenie projektu. Sleduje časový harmonogram, rozpočet, kvalitu a komunikáciu medzi členmi tímu a zainteresovanými stranami.

**BA (Business Analyst)**  
Business analytik je zodpovedný za zber, analýzu a dokumentovanie požiadaviek od zainteresovaných strán. Prekladá obchodné potreby do technických špecifikácií pre vývojový tím a pomáha zabezpečiť, aby výsledné riešenie spĺňalo očakávania.

**QA (Quality Assurance)**  
Špecialista na zabezpečenie kvality má na starosti procesy, ktoré zaručujú, že produkt spĺňa stanovené štandardy kvality. Zahŕňa to tvorbu testovacích plánov, vykonávanie testov a reportovanie chýb.

**DEV (Developer)**  
Vývojár je zodpovedný za návrh, implementáciu a údržbu softvéru. Pracuje s rôznymi programovacími jazykmi a technológiami s cieľom vytvoriť funkčné a efektívne riešenia.

**UX (User Experience Designer)**  
Dizajnér používateľskej skúsenosti sa zameriava na návrh rozhraní a interakcií, ktoré sú intuitívne a príjemné pre používateľa. Využíva výskum, testovanie a prototypovanie na optimalizáciu zážitku z používania produktu.

**DBA (Database Administrator)**  
Databázový administrátor je zodpovedný za správu, údržbu a optimalizáciu databáz. Zabezpečuje dostupnosť, bezpečnosť a výkon databázových systémov a vykonáva zálohovanie a obnovu dát.

**KPI**  
(Key Performance Indicator) je kľúčový ukazovateľ výkonnosti používaný na meranie úspešnosti procesov alebo projektov. Pomáha sledovať pokrok voči stanoveným cieľom a identifikovať oblasti na zlepšenie.

**ROI**

(Return on Investment) je ukazovateľ, ktorý vyjadruje ziskovosť investície. Vypočíta sa ako pomer čistého zisku k investovaným nákladom a používa sa na hodnotenie efektívnosti projektov.

**Item-to-item**

Metóda odporúčacích systémov, ktorá vyhľadáva podobné položky na základe histórie správania používateľov. Často sa využíva v e-shopoch a streamingových službách.

**Save-rate@K**

Metrika vyjadrujúca percento položiek z odporúčacieho zoznamu dĺžky *K*, ktoré si používateľ uloží na neskoršie použitie. Hodí sa na meranie užitočnosti odporúčaní.

**CTR@K**

(Click-Through Rate) pri *K* odporúčaných položkách udáva percento kliknutí na položky zo zoznamu. Používa sa na hodnotenie atraktívnosti odporúčaní.

**Cook-proxy@K**

Špecifická metrika alebo mechanizmus proxy merania v odporúčacom systéme, ktorý sa aplikuje na *K* návrhov. Slúži na testovanie výkonu algoritmov nepriamo cez náhradné ukazovatele.

**Churn guardrails**

Opatrenia a metriky, ktoré bránia prudkému poklesu aktívnych používateľov (tzv. odlivu). Pomáhajú udržať stabilitu používateľskej základne pri zavádzaní zmien.

**Precision@K**

Metrika presnosti odporúčaní – vyjadruje podiel relevantných položiek medzi prvými *K* odporúčaniami. Vhodná na hodnotenie kvality návrhov.

**Recall@K**

Vyjadruje, akú časť zo všetkých relevantných položiek sa podarilo odporučiť medzi prvými *K* výsledkami. Dôležitá pre systémy, kde je prioritou úplnosť odporúčaní.

**nDCG@K**

(Normalized Discounted Cumulative Gain) je metrika hodnotiaca kvalitu zoradených odporúčaní, pričom zohľadňuje pozíciu relevantných položiek v zozname. Normalizácia umožňuje porovnanie medzi rôznymi zoznamami.

**Coverage**

Pokrytie – vyjadruje, akú časť dostupných položiek systém vôbec odporučí. Vyššie pokrytie znamená, že odporúčací systém ponúka väčšiu rozmanitosť.

**Intra-List Diversity (ILD) a novelty**

ILD meria rôznorodosť položiek v zozname odporúčaní, zatiaľ čo *novelty* hodnotí, do akej miery sú položky pre používateľa nové alebo neznáme. Obe metriky prispievajú k lepšej používateľskej skúsenosti.

**Latency P95**

Meria čas odozvy systému, pod ktorý sa zmestí 95 % všetkých požiadaviek. Používa sa na hodnotenie rýchlosti a konzistencie systému.

**Serendipity proxy**

Metrika alebo mechanizmus na meranie prekvapivosti a príjemnej neočakávanosti odporúčaní. Pomáha odlíšiť systém, ktorý neponúka len očakávané položky.

**Popularity-bias index**

Index merajúci, do akej miery odporúčací systém uprednostňuje populárne položky pred menej známymi. Vysoká hodnota môže signalizovať nedostatok diverzity odporúčaní.

**Temporal split**

Metóda delenia dát na trénovacie a testovacie časti na základe času. Používa sa najmä pri modeloch, kde poradie udalostí zohráva významnú úlohu.

**Golden set párov**

Súbor referenčných alebo „zlatých“ dát použitých na porovnanie a validáciu modelov či algoritmov. Slúži ako etalón, voči ktorému sa meria presnosť alebo výkon systému.

**Skóre cosine embeddings**

Metrika založená na kosínovej podobnosti, ktorá vyjadruje, ako sú si dva vektorové reprezentácie (embeddings) podobné. Hodnota blízka 1 znamená vysokú podobnosť.

**TF-IDF/Jaccard**

TF-IDF je metóda váženia slov podľa ich frekvencie v dokumente a zriedkavosti v celom korpuse. Jaccardov index meria podobnosť množín na základe prieniku a zjednotenia prvkov.

**Top-K**

Označuje výber *K* najlepších alebo najrelevantnejších položiek podľa určitého hodnotenia. Bežne sa používa v odporúčacích systémoch a vyhľadávaní.

**Cold-start item**

Situácia, keď systém má odporučiť novú položku bez dostatočných historických dát. Vyžaduje špeciálne metódy, ako napríklad využitie metadát.

**Cold-start user**

Podobne ako cold-start item, ide o nového používateľa bez histórie správania. Systém mu môže odporúčať na základe všeobecných trendov alebo dotazníkov.

**ILD**

(Intra-List Diversity) meria rozmanitosť položiek v zozname odporúčaní. Vyššia hodnota ILD znamená, že zoznam obsahuje rôznorodé položky.

**Popularity-penalty**

Technika, ktorá znižuje váhu príliš populárnych položiek v odporúčaniach. Cieľom je podporiť diverzitu a objavovanie nových možností.

**Ablácie**

Testovacia metóda, pri ktorej sa zámerne odstraňujú alebo menia časti modelu či systému, aby sa zistil ich vplyv na výsledok. Pomáha odhaliť dôležitosť jednotlivých komponentov.

**IPS/DR**

(Inverse Propensity Scoring / Doubly Robust) sú štatistické metódy používané na vyhodnocovanie odporúčacích systémov z neúplných dát. Pomáhajú eliminovať skreslenie pri hodnotení.

**Team-Draft Interleaving**

Technika porovnávania dvoch vyhľadávacích alebo odporúčacích algoritmov tak, že sa ich výsledky striedavo miešajú do jedného zoznamu. Používatelia interagujú s týmto zoznamom a ich správanie sa vyhodnocuje.

**Save-rate@K**

Percento položiek z odporúčaného zoznamu dĺžky *K*, ktoré si používateľ uloží. Používa sa na meranie užitočnosti odporúčaní.

**Guardrails**

Bezpečnostné a kontrolné mechanizmy, ktoré zabraňujú nežiaducemu alebo riskantnému správaniu systému. Môžu byť implementované v kóde alebo v procesoch.

**CUPED**

(Control Using Pre-Experiment Data) je štatistická metóda znižovania variability výsledkov experimentu. Využíva historické dáta na presnejšie odhady efektu.

**p-hacking**

Praktika manipulácie s experimentálnymi alebo štatistickými postupmi tak, aby výsledky vyšli štatisticky významné. Považuje sa za neetické a vedie k nespoľahlivým záverom.

**Simpsonov paradox**

Štatistický jav, pri ktorom sa trend pozorovaný v jednotlivých skupinách môže po ich spojení zmeniť alebo úplne zmiznúť. Poukazuje na dôležitosť správneho rozdelenia a interpretácie dát.

**Shilling útoky**

Úmyselné manipulácie s odporúčacím systémom pomocou falošných recenzií alebo hodnotení s cieľom zvýšiť či znížiť popularitu konkrétnych položiek. Používa sa v konkurenčnom boji alebo pri propagácii vlastných produktov.

**Trust score**

Metrika hodnotiaca dôveryhodnosť používateľa, zariadenia alebo transakcie. Zohľadňuje históriu správania, geolokáciu, reputáciu a ďalšie signály.

**Low-data mód**

Režim aplikácie, ktorý minimalizuje prenos dát a využíva menej náročné zdroje. Je určený pre pomalé alebo drahé internetové pripojenie.

**BlurHash/ThumbHash**

Algoritmy na generovanie malých vizuálnych náhľadov obrázkov v podobe farebných vzorov alebo rozmazaných blokov. Slúžia na rýchle načítanie náhľadu ešte pred stiahnutím originálneho obrázka.

**Lazy loading**

Technika načítavania obsahu až v momente, keď je potrebný, napríklad pri zobrazení na obrazovke. Znižuje čas načítania stránky a šetrí zdroje.

**TTFR**

(Time To First Render) meria čas od začiatku načítania stránky alebo aplikácie po prvé zobrazenie obsahu. Je kľúčovým ukazovateľom rýchlosti a používateľského komfortu.

**Error budget**

Povolená miera chýb alebo výpadkov, ktorú si tím stanoví bez porušenia dohodnutých úrovní služieb (SLA). Pomáha vyvážiť spoľahlivosť systému a rýchlosť dodávania nových funkcií.

**Android Studio/ADB**

Android Studio je oficiálne IDE pre vývoj Android aplikácií. ADB (Android Debug Bridge) je nástroj na komunikáciu s Android zariadením na účely testovania, ladenia a nasadzovania aplikácií.

**Network Link Conditioner**

Nástroj na simulovanie rôznych sieťových podmienok, ako je nízka rýchlosť, vysoká latencia či strata paketov. Používa sa pri testovaní odolnosti a výkonu aplikácií.

**Chrome DevTools Throttling**

Funkcia v Chrome DevTools umožňujúca obmedziť rýchlosť siete alebo výkon procesora pri testovaní webových aplikácií. Pomáha identifikovať problémy v rôznych podmienkach.

**Chatty API**

API, ktoré pri bežnej interakcii vyžaduje príliš veľa volaní, čím zvyšuje zaťaženie siete a latenciu. Optimalizácia zvyčajne zahŕňa zlučovanie požiadaviek.

**Protobuf/MessagePack**

Formáty binárnej serializácie dát, ktoré sú efektívnejšie než JSON alebo XML. Umožňujú rýchlejší prenos a menšiu veľkosť dát.

**AARRR**

Model merania rastu produktu – Acquisition, Activation, Retention, Referral, Revenue. Pomáha sledovať životný cyklus používateľa a optimalizovať stratégie.

**WAU**

(Weekly Active Users) metrika, ktorá udáva počet jedinečných používateľov, ktorí sa zapojili do aplikácie aspoň raz za týždeň. Slúži na sledovanie zapojenia a rastu.

**Diversity (ILD)**

Ukazovateľ, ako rôznorodé sú odporúčané položky. Vyššia hodnota znamená väčšiu variabilitu odporúčaní.

**haveibeenpwned API**

API služby Have I Been Pwned umožňuje overiť, či bol e-mail alebo heslo kompromitované pri úniku dát. Slúži na zvýšenie bezpečnosti používateľov.

**IDOR**

(Insecure Direct Object Reference) zraniteľnosť, pri ktorej aplikácia umožňuje prístup k objektom (napr. súborom, záznamom) na základe predvídateľných identifikátorov bez riadneho overenia oprávnení.

**UUID**

(Universally Unique Identifier) 128-bitový identifikátor, ktorý je prakticky jedinečný v čase a priestore. Používa sa na označovanie objektov, transakcií či relácií.

**MITM útoky**

(Man-in-the-Middle) útoky, pri ktorých útočník zachytáva a mení komunikáciu medzi dvoma stranami bez ich vedomia. Používajú sa na krádež údajov alebo manipuláciu prenosu.

**DoS**

(Denial of Service) útok, ktorého cieľom je zneprístupniť službu pre oprávnených používateľov preťažením systému. Často sa realizuje zahltením siete alebo servera požiadavkami.

**Credential stuffing**

Útok, pri ktorom útočník využíva zoznamy kompromitovaných prihlasovacích údajov z iných služieb na prihlásenie sa do cieľovej aplikácie. Spolieha sa na opakované používanie hesiel.

**Scam odkazy**

Podvodné odkazy, ktoré sa vydávajú za legitímne a snažia sa získať citlivé údaje alebo peniaze od používateľov. Často vedú na phishingové stránky.

**Model poisoning**

Útok na strojové učenie, pri ktorom útočník manipuluje trénovacie dáta s cieľom zmeniť správanie modelu. Môže viesť k nesprávnym predikciám alebo zníženiu výkonu.

**opt-in na tracking**

Mechanizmus, pri ktorom používateľ výslovne súhlasí so sledovaním svojho správania. Vyžaduje sa v mnohých legislatívach na ochranu súkromia.

**Logický COT**

(Chain of Thought) logický reťazec uvažovania používaný pri riešení problémov alebo rozhodovaní. V AI kontexte označuje generovanie medzikrokov vedúcich k finálnej odpovedi.

**Spinner**

**maintenance mód**

**fraud detection**

**IBAN**

**BIC**