

BACEN

BANCO CENTRAL DO BRASIL

Analista – Área: Tecnologia
da Informação

CÓD: SL-083JL-26
7901183002115

Língua Portuguesa

1. Compreensão e interpretação de textos de gêneros variados	9
2. Reconhecimento de tipos e gêneros textuais	12
3. Domínio da ortografia oficial	18
4. Domínio dos mecanismos de coesão textual.....	22
5. Emprego de elementos de referência, substituição e repetição, de conectores e de outros elementos de sequenciação textual.....	25
6. Emprego de tempos e modos verbais	25
7. Domínio da estrutura morfosintática do período. Relações de coordenação entre orações e entre termos da oração. Relações de subordinação entre orações e entre termos da oração. Reorganização da estrutura de orações e de períodos do texto.....	26
8. Emprego das classes de palavras	30
9. Emprego dos sinais de pontuação	39
10. Concordância verbal e nominal	42
11. Regência verbal e nominal.....	45
12. Emprego do sinal indicativo de crase.....	49
13. Colocação dos pronomes átonos	52
14. Reescrita de frases e parágrafos do texto. Reescrita de textos de diferentes gêneros e níveis de formalidade.....	53
15. Significação das palavras.....	54
16. Substituição de palavras ou de trechos de texto	57

Noções de Lógica e Estatística

1. Raciocínio lógico. Estruturas lógicas	67
2. Lógica de argumentação: analogias, inferências, deduções e conclusões.....	70
3. Lógica sentencial (ou proposicional). Proposições simples e compostas. Tabelas-verdade. Equivalências. Leis de Morgan; problemas	73
4. Noções de estatística. População e amostra.....	79
5. Histogramas e curvas de frequência	80
6. Medidas de posição: média, moda, mediana e separatrizes	84
7. Medidas de dispersão absoluta e relativa.....	86
8. Probabilidade condicional, independência	89
9. Variável aleatória e funções de distribuição	91

Fundamentos de Macroeconomia e Microeconomia

1. Contas nacionais. Agregados monetários. Multiplicador monetário, criação e destruição de moeda. Contas do sistema monetário. Balanço de pagamentos	99
2. MICROECONOMIA: Estrutura de mercado. Formas de organização da atividade econômica, o papel dos preços, custo de oportunidade e fronteiras das possibilidades de produção. Oferta e demanda. Curvas de indiferença. Restrição orçamentária. Equilíbrio do consumidor. Efeitos preço, renda e substituição. Curva de demanda. Elasticidade da demanda	105

Ciência de Dados

1. Aprendizado de Máquina.....	115
2. Deep learning.....	118
3. Processamento de linguagem natural	121
4. Big data	124
5. Qualidade de Dados.....	127
6. Tipos de Aprendizado: Supervisionado, Não Supervisionado, Semi Supervisionado, Por Reforço, Por Transferência	130
7. Grandes Modelos de Linguagem (LLM), IA Generativa.....	134
8. Redes Neurais	137
9. MLOps: Gestão de código, treinamento, implantação, monitoramento e versionamento de modelos, automação do ciclo de produção	140
10. Governança e Ética na IA: Transparência, Responsabilidade, Explicabilidade, Privacidade, Segurança, Viés.....	143

Segurança da Informação

1. Gestão de Identidades e Acesso: Autenticação e Autorização, Single Sign-On (SSO), Security Assertion Markup Language (SAML), OAuth2 e OpenId Connect	149
2. Privacidade e segurança por padrão.....	154
3. Principais tipos de ataques e vulnerabilidades	158
4. Controles e testes de segurança para aplicações Web e Web Services	160
5. Múltiplos Fatores de Autenticação (MFA).....	168
6. Soluções para Segurança da Informação: Firewall, Intrusion Detection System (IDS), Intrusion Prevention System (IPS), Security Information and Event Management (SIEM), Proxy, Identity Access Management (IAM), Privileged Access Management (PAM), Antivírus, Antispam	172
7. Frameworks de segurança da informação e segurança cibernética, como, por exemplo, MITRE ATT&CK, CIS Controls e NIST CyberSecurity Framework (NIST CSF)	183
8. Tratamento de Incidentes Cibernéticos	196
9. Assinatura e certificação digital, criptografia e proteção de dados em trânsito e em repouso.....	204
10. Segurança em nuvens e de contêineres	220

Engenharia de Software

1. Arquitetura de sistemas web: protocolo HTTP, HTTP/2, gRPC, WebSockets, TLS, servidores proxy, cache, DNS, balanceamento de carga, tolerância a falhas e escalabilidade em sistemas web.....	269
2. Princípios e práticas de DevOps e DevSecOps, integração e entrega contínuas (CI/CD)	280
3. Técnicas de desenvolvimento seguro.....	287
4. Testes de software: Testes unitários, Testes de Integração, TDD, BDD	292
5. Arquiteturas em camadas, baseada em serviços, microsserviços (orquestração de serviços e API gateway), orientação a eventos, cliente-servidor, serverless.....	297
6. Práticas de UX e UI design	304
7. Programação assíncrona	311
8. RESTful e GraphQL	317
9. Web services.....	324

1. Padrões: GoF e GRASP	330
2. Git	337
3. Python e Java	344
4. Transações distribuídas	348
5. Distributed Ledger Technology (DLT)	352

Infraestrutura em TI

1. Conceitos de infraestrutura como código e automação de infraestrutura de TI	333
2. Docker, Kubernetes: boas práticas para infraestrutura e orquestração de containers	335
3. Serviços de Rede Microsoft Windows Server: DNS, DHCP, Radius, Autenticação, Certificados, Active Directory (AD)	338
4. Monitoração, observabilidade e análise de sistemas em produção por meio do uso de ferramentas de monitoramento e logging, como o Nagios, Prometheus, Grafana, Elasticsearch, Kibana, Application Performance Monitoring (APM), entre outras	341
5. Protocolos da camada de aplicação, como Simple Mail Transfer Protocol (SMTP), Hypertext Transfer Protocol (HTTP), Hypertext Transfer Protocol (HTTPS), SSL/TLS, Lightweight Directory Access Protocol (LDAP), Network File System (NFS), Server Message Block (SMB)	343
6. Tolerância a falhas e continuidade de operação	346
7. Implantação e administração de serviços de nuvem (IaaS, PaaS e SaaS)	349
8. Administração e gerenciamento de ambientes de virtualização	351
9. Administração e operação de sistemas operacionais Microsoft Windows Server e Linux	354
10. Conceitos de LAN, WAN e SDN	357
11. Conceitos e ferramentas de orquestração e automação de infraestrutura: Puppet, Ansible	359

Bancos de Dados

1. SGBDs SQL e NOSQL	365
2. Modelagens de dados: relacional, multidimensional, nosql	368
3. SQL (Procedural Language / Structured Query Language)	370
4. Arquitetura de Inteligência de Negócio: DataWarehouse, DataMart, DataLake, DataMesh	374

Gestão em TI

1. Kanban	381
2. Scrum	388
3. Governança de Dados	394
4. ITIL v4	396

Material Digital Atualidades

1. Tópicos relevantes e atuais de diversas áreas, tais como segurança, transportes, política, economia, sociedade, educação, saúde, cultura, tecnologia, energia, relações internacionais, desenvolvimento sustentável e ecologia 4

Direito Administrativo

1. Administração pública: princípios básicos 5
2. Administração direta e indireta; Organização administrativa brasileira; Autarquias, agências reguladoras, agências executivas, fundações públicas, empresas públicas, sociedades de economia mista 9
3. Poderes administrativos: Espécies de poder: hierárquico, disciplinar, regulamentar, de polícia e normativo; Uso e abuso do poder 13
4. Serviços públicos: conceito e princípios 15
5. Ato administrativo: Conceito, requisitos e atributos; Comunicação dos atos administrativos; Anulação, revogação e convalidação; Discricionariedade e vinculação..... 18
6. Servidores públicos: cargo, emprego e função públicos: Lei nº 8.112/1990 (Regime Jurídico dos Servidores Públicos Civis da União) e alterações; Disposições preliminares; Provimento, vacância, remoção, redistribuição e substituição; Direitos e vantagens; Regime disciplinar; Seguridade social do servidor: aposentadoria e pensão civil 22
7. Processo administrativo disciplinar 51
8. Improbidade administrativa..... 57
9. Código de Ética Profissional do Servidor Público Civil do Poder Executivo Federal (Decreto nº 1.171/1994)..... 66
10. Código de Conduta da Alta Administração Federal..... 69
11. Lei de conflito de interesses (Lei nº 11.813/2013)..... 72
12. Lei de Acesso à Informação (Lei nº 12.527/2011)..... 74
13. Lei nº 13.709/2018 (Lei Geral de Proteção de Dados Pessoais)..... 81

Atenção

▪ Para estudar o Material Digital acesse sua “Área do Aluno” em nosso site ou faça o resgate do material seguindo os passos da página 2.

<https://www.editorasolucao.com.br/customer/account/login/>

LÍNGUA PORTUGUESA

COMPREENSÃO E INTERPRETAÇÃO DE TEXTOS DE GÊNEROS VARIADOS

A compreensão e a interpretação de textos são habilidades essenciais para que a comunicação alcance seu objetivo de forma eficaz. Em diversos contextos, como na leitura de livros, artigos, propagandas ou imagens, é necessário que o leitor seja capaz de entender o conteúdo proposto e, além disso, atribuir significados mais amplos ao que foi lido ou visto.

Para isso, é importante distinguir os conceitos de compreensão e interpretação, bem como reconhecer que um texto pode ser verbal (composto por palavras) ou não-verbal (constituído por imagens, símbolos ou outros elementos visuais).

Compreender um texto implica decodificar sua mensagem explícita, ou seja, captar o que está diretamente apresentado. Já a interpretação vai além da compreensão, exigindo que o leitor utilize seu repertório pessoal e conhecimentos prévios para gerar um sentido mais profundo do texto. Dessa forma, dominar esses dois processos é essencial não apenas para a leitura cotidiana, mas também para o desempenho em provas e concursos, onde a análise de textos e imagens é frequentemente exigida.

Essa distinção entre compreensão e interpretação é crucial, pois permite ao leitor ir além do que está explícito, alcançando uma leitura mais crítica e reflexiva.

CONCEITO DE COMPREENSÃO

A compreensão de um texto é o ponto de partida para qualquer análise textual. Ela representa o processo de decodificação da mensagem explícita, ou seja, a habilidade de extrair informações diretamente do conteúdo apresentado pelo autor, sem a necessidade de agregar inferências ou significados subjetivos. Quando compreendemos um texto, estamos simplesmente absorvendo o que está dito de maneira clara, reconhecendo os elementos essenciais da comunicação, como o tema, os fatos e os argumentos centrais.

► A Compreensão em Textos Verbais

Nos textos verbais, que utilizam a linguagem escrita ou falada como principal meio de comunicação, a compreensão passa pela habilidade de ler com atenção e reconhecer as estruturas linguísticas. Isso inclui:

- **Vocabulário:** O entendimento das palavras usadas no texto é fundamental. Palavras desconhecidas podem comprometer a compreensão, tornando necessário o uso de dicionários ou ferramentas de pesquisa para esclarecer o significado.
- **Sintaxe:** A maneira como as palavras estão organizadas em frases e parágrafos também influencia o processo de compreensão. Sentenças complexas, inversões sintáticas ou o

- uso de conectores como conjunções e preposições requerem atenção redobrada para garantir que o leitor compreenda as relações entre as ideias.

- **Coesão e coerência:** são dois pilares essenciais da compreensão. Um texto coeso é aquele cujas ideias estão bem conectadas, e a coerência se refere à lógica interna do texto, onde as ideias se articulam de maneira fluida e compreensível.

Ao realizar a leitura de um texto verbal, a compreensão exige a decodificação de todas essas estruturas. É a partir dessa leitura atenta e detalhada que o leitor poderá garantir que absorveu o conteúdo proposto pelo autor de forma plena.

► A Compreensão em Textos Não-Verbais

Além dos textos verbais, a compreensão se estende aos textos não-verbais, que utilizam símbolos, imagens, gráficos ou outras representações visuais para transmitir uma mensagem. Exemplos de textos não-verbais incluem obras de arte, fotografias, infográficos e até gestos em uma linguagem de sinais.

A compreensão desses textos exige uma leitura visual aguçada, na qual o observador decodifica os elementos presentes, como:

- **Cores:** As cores desempenham um papel comunicativo importante em muitos contextos, evocando emoções ou sugerindo informações adicionais. Por exemplo, em um gráfico, cores diferentes podem representar categorias distintas de dados.
- **Formas e símbolos:** Cada forma ou símbolo em um texto visual pode carregar um significado próprio, como sinais de trânsito ou logotipos de marcas. A correta interpretação desses elementos depende do conhecimento prévio do leitor sobre seu uso.
- **Gestos e expressões:** Em um contexto de comunicação corporal, como na linguagem de sinais ou em uma apresentação oral acompanhada de gestos, a compreensão se dá ao identificar e entender as nuances de cada movimento.

► Fatores que Influenciam a Compreensão

A compreensão, seja de textos verbais ou não-verbais, pode ser afetada por diversos fatores, entre eles:

- **Conhecimento prévio:** Quanto mais familiarizado o leitor estiver com o tema abordado, maior será sua capacidade de compreender o texto. Por exemplo, um leitor que já conhece o contexto histórico de um fato poderá compreender melhor uma notícia sobre ele.

▪ **Contexto:** O ambiente ou a situação em que o texto é apresentado também influencia a compreensão. Um texto jornalístico, por exemplo, traz uma mensagem diferente dependendo de seu contexto histórico ou social.

▪ **Objetivos da leitura:** O propósito com o qual o leitor aborda o texto impacta a profundidade da compreensão. Se a leitura for para estudo, o leitor provavelmente será mais minucioso do que em uma leitura por lazer.

► **Compreensão como Base para a Interpretação**

A compreensão é o primeiro passo no processo de leitura e análise de qualquer texto. Sem uma compreensão clara e objetiva, não é possível seguir para uma etapa mais profunda, que envolve a interpretação e a formulação de inferências. Somente após a decodificação do que está explicitamente presente no texto, o leitor poderá avançar para uma análise mais subjetiva e crítica, onde ele começará a trazer suas próprias ideias e reflexões sobre o que foi lido.

A compreensão textual é um processo que envolve a decodificação de elementos verbais e não-verbais, permitindo ao leitor captar a mensagem essencial do conteúdo. Ela exige atenção, familiaridade com as estruturas linguísticas ou visuais e, muitas vezes, o uso de recursos complementares, como dicionários. Ao dominar a compreensão, o leitor cria uma base sólida para interpretar textos de maneira mais profunda e crítica.

TEXTOS VERBAIS E NÃO-VERBAIS

Na comunicação, os textos podem ser classificados em duas categorias principais: verbais e não-verbais. Cada tipo de texto utiliza diferentes recursos e linguagens para transmitir suas mensagens, sendo fundamental que o leitor ou observador saiba identificar e interpretar corretamente as especificidades de cada um.

► **Textos Verbais**

Os textos verbais são aqueles constituídos pela linguagem escrita ou falada, onde as palavras são o principal meio de comunicação. Eles estão presentes em inúmeros formatos, como livros, artigos, notícias, discursos, entre outros. A linguagem verbal se apoia em uma estrutura gramatical, com regras que organizam as palavras e frases para transmitir a mensagem de forma coesa e compreensível.

Características dos Textos Verbais:

▪ **Estrutura Sintática:** As frases seguem uma ordem gramatical que facilita a decodificação da mensagem.

▪ **Uso de Palavras:** As palavras são escolhidas com base em seu significado e função dentro do texto, permitindo ao leitor captar as ideias expressas.

▪ **Coesão e Coerência:** A conexão entre frases, parágrafos e ideias deve ser clara, para que o leitor compreenda a linha de raciocínio do autor.

Exemplos de textos verbais incluem:

▪ **Livros e artigos:** Onde há um desenvolvimento contínuo de ideias, apoiado em argumentos e explicações detalhadas.

▪ **Diálogos e conversas:** Que utilizam a oralidade para interações mais diretas e dinâmicas.

▪ **Panfletos e propagandas:** Usam a linguagem verbal de forma concisa e direta para transmitir uma mensagem específica.

A compreensão de um texto verbal envolve a decodificação de palavras e a análise de como elas se conectam para construir significado. É essencial que o leitor identifique o tema, os argumentos centrais e as intenções do autor, além de perceber possíveis figuras de linguagem ou ambiguidades.

► **Textos Não-Verbais**

Os textos não-verbais utilizam elementos visuais para se comunicar, como imagens, símbolos, gestos, cores e formas. Embora não usem palavras diretamente, esses textos transmitem mensagens completas e são amplamente utilizados em contextos visuais, como artes visuais, placas de sinalização, fotografias, entre outros.

Características dos Textos Não-Verbais:

▪ **Imagens e símbolos:** Carregam significados culturais e contextuais que devem ser reconhecidos pelo observador.

▪ **Cores e formas:** Podem ser usadas para evocar emoções ou destacar informações específicas. Por exemplo, a cor vermelha em muitos contextos pode representar perigo ou atenção.

▪ **Gestos e expressões:** Na comunicação corporal, como na linguagem de sinais ou na expressão facial, o corpo desempenha o papel de transmitir a mensagem.

Exemplos de textos não-verbais incluem:

▪ **Obras de arte:** Como pinturas ou esculturas, que comunicam ideias, emoções ou narrativas através de elementos visuais.

▪ **Sinais de trânsito:** Que utilizam formas e cores para orientar os motoristas, dispensando a necessidade de palavras.

▪ **Infográficos:** Combinações de gráficos e imagens que transmitem informações complexas de forma visualmente acessível.

A interpretação de textos não-verbais exige uma análise diferente da dos textos verbais. É necessário entender os códigos visuais que compõem a mensagem, como as cores, a composição das imagens e os elementos simbólicos utilizados. Além disso, o contexto cultural é crucial, pois muitos símbolos ou gestos podem ter significados diferentes dependendo da região ou da sociedade em que são usados.

► **Relação entre Textos Verbais e Não-Verbais**

Embora sejam diferentes em sua forma, textos verbais e não-verbais frequentemente se complementam. Um exemplo comum são as propagandas publicitárias, que utilizam tanto textos escritos quanto imagens para reforçar a mensagem. Nos livros ilustrados, as imagens acompanham o texto verbal, ajudando a criar um sentido mais completo da história ou da informação.

NOÇÕES DE LÓGICA E ESTATÍSTICA

RACIOCÍNIO LÓGICO. ESTRUTURAS LÓGICAS

ASSOCIAÇÃO DE INFORMAÇÕES

Aqui veremos questões que envolvem correlação de elementos, pessoas e objetos fictícios, através de dados fornecidos. Vejamos o passo a passo:

01. Três homens, Luís, Carlos e Paulo, são casados com Lúcia, Patrícia e Maria, mas não sabemos quem é casado com quem. Eles trabalham com Engenharia, Advocacia e Medicina, mas também não sabemos quem faz o quê. Com base nas dicas abaixo, tente descobrir o nome de cada marido, a profissão de cada um e o nome de suas esposas.

- a) O médico é casado com Maria.
- b) Paulo é advogado.
- c) Patrícia não é casada com Paulo.
- d) Carlos não é médico.

Vamos montar o passo a passo para que você possa compreender como chegar a conclusão da questão.

▪ **1º passo** – Construir a tabela dos dados. Vamos montar uma tabela para facilitar a visualização da resolução, a mesma deve conter as informações prestadas no enunciado, nas quais podem ser divididas em três grupos: homens, esposas e profissões.

	Medicina	Engenharia	Advocacia	Lúcia	Patrícia	Maria
Carlos						
Luís						
Paulo						
Lúcia						
Patrícia						
Maria						

Também criamos abaixo do nome dos homens, o nome das esposas.

▪ **2º passo** – Construir a tabela gabarito. Essa tabela não servirá apenas como gabarito, mas em alguns casos ela é **fundamental** para que você enxergue informações que ficam meio escondidas na tabela principal. Uma tabela complementa a outra, podendo até mesmo que você chegue a conclusões acerca dos grupos e elementos.

HOMENS	PROFISSÕES	ESPOSAS
Carlos		
Luís		
Paulo		

AMOSTRA

▪ **3º passo** preenchimento de nossa tabela, com as informações mais óbvias do problema, aquelas que não deixam margem a nenhuma dúvida. Em nosso exemplo:

O médico é casado com Maria: marque um “S” na tabela principal na célula comum a “Médico” e “Maria”, e um “N” nas demais células referentes a esse “S”.

	Medicina	Engenharia	Advocacia	Lúcia	Patrícia	Maria
Carlos						
Luís						
Paulo						
Lúcia	N					
Patrícia	N					
Maria	S	N	N			

ATENÇÃO: se o médico é casado com Maria, ele **NÃO PODE** ser casado com Lúcia e Patrícia, então colocamos “N” no cruzamento de Medicina e elas. E se Maria é casada com o médico, logo ela **NÃO PODE** ser casada com o engenheiro e nem com o advogado (logo colocamos “N” no cruzamento do nome de Maria com essas profissões).

Paulo é advogado: Vamos preencher as duas tabelas (tabela gabarito e tabela principal) agora.

Patrícia não é casada com Paulo: Vamos preencher com “N” na tabela principal

Carlos não é médico: preenchamos com um “N” na tabela principal a célula comum a Carlos e “médico”.

	Medicina	Engenharia	Advocacia	Lúcia	Patrícia	Maria
Carlos	N		N			
Luís	S	N	N			
Paulo	N	N	S		N	
Lúcia	N					
Patrícia	N					
Maria	S	N	N			

Notamos aqui que Luís então é o médico, pois foi a célula que ficou em branco. Podemos também completar a tabela gabarito.

Novamente observamos uma célula vazia no cruzamento de Carlos com Engenharia. Marcamos um “S” nesta célula. E preenchemos sua tabela gabarito.

	Medicina	Engenharia	Advocacia	Lúcia	Patrícia	Maria
Carlos	N	S	N			
Luís	S	N	N			
Paulo	N	N	S		N	
Lúcia	N					
Patrícia	N					
Maria	S	N	N			

HOMENS	PROFISSÕES	ESPOSAS
Carlos	Engenheiro	
Luís	Médico	
Paulo	Advogado	

FUNDAMENTOS DE MACROECONOMIA E MICROECONOMIA

CONTAS NACIONAIS. AGREGADOS MONETÁRIOS. MULTIPLICADOR MONETÁRIO, CRIAÇÃO E DESTRUIÇÃO DE MOEDA. CONTAS DO SISTEMA MONETÁRIO. BALANÇO DE PAGAMENTOS

CONTAS NACIONAIS

As contas nacionais formam um sistema de registro da atividade econômica de um país em determinado período. Elas organizam informações sobre produção, renda, consumo, investimento, gastos do governo, exportações e importações. Sua principal finalidade é medir o desempenho da economia de forma agregada, permitindo conhecer o valor dos bens e serviços produzidos, a renda gerada e a forma como essa renda foi utilizada.

O indicador mais conhecido das contas nacionais é o Produto Interno Bruto, chamado de PIB. Ele representa o valor de todos os bens e serviços finais produzidos dentro do território econômico de um país durante determinado período, geralmente um trimestre ou um ano. A expressão bens e serviços finais é importante porque evita dupla contagem. Não se deve contar separadamente o valor de um insumo e depois o valor do produto final que já incorpora esse insumo.

Por exemplo, se a farinha é usada para produzir pão, o valor da farinha já está incluído no preço do pão. Se ambos fossem somados integralmente, haveria dupla contagem. Por isso, o PIB considera bens e serviços finais ou, de outra forma, o valor adicionado em cada etapa produtiva. O valor adicionado corresponde à diferença entre o valor da produção e o valor dos insumos utilizados.

As contas nacionais podem medir a atividade econômica por três óticas: produto, renda e despesa. Pela ótica do produto, mede-se o valor da produção de bens e serviços finais. Pela ótica da renda, mede-se a remuneração dos fatores de produção, como salários, lucros, juros, aluguéis e rendimentos mistos. Pela ótica da despesa, mede-se o gasto realizado na compra dos bens e serviços finais produzidos.

Essas três óticas estão relacionadas. O que é produzido gera renda para alguém; essa renda permite despesas; e as despesas sustentam a venda da produção. Por isso, produto, renda e despesa são formas diferentes de observar o mesmo processo econômico. Em uma economia bem registrada, os valores tendem a ser equivalentes, respeitadas as metodologias e ajustes estatísticos.

Na ótica da despesa, o PIB costuma ser representado pela soma do consumo das famílias, investimento, gastos do governo e exportações líquidas. As exportações líquidas correspondem às exportações menos as importações. Essa identidade mostra quem comprou a produção da economia: famílias, empresas, governo ou estrangeiros.

O consumo das famílias representa os gastos das pessoas com bens e serviços, como alimentos, transporte, moradia, saúde, educação, lazer e outros itens. Em muitas economias, o consumo é a maior parcela da demanda agregada. Ele depende da renda, do emprego, dos preços, do crédito, das expectativas e de outros fatores.

O investimento representa a formação de capital, como aquisição de máquinas, equipamentos, construção, ampliação de capacidade produtiva e variação de estoques. É importante distinguir investimento econômico de aplicação financeira. Na contabilidade nacional, investimento está ligado à produção de bens de capital e aumento da capacidade produtiva.

Os gastos do governo incluem despesas com bens e serviços necessários à prestação de serviços públicos e ao funcionamento da administração. Transferências, como benefícios pagos às famílias, não entram diretamente como gasto de governo no PIB, pois não correspondem à compra de bem ou serviço final naquele momento. No entanto, podem influenciar o consumo das famílias.

As exportações representam bens e serviços produzidos internamente e vendidos ao exterior. Elas entram no PIB porque correspondem à produção nacional. As importações são subtraídas porque representam bens e serviços produzidos em outros países, embora consumidos internamente. Subtrair importações evita contar como produção interna aquilo que foi produzido fora.

As contas nacionais também permitem calcular o PIB nominal e o PIB real. O PIB nominal é medido a preços correntes, ou seja, pelos preços do próprio período analisado. O PIB real elimina o efeito da variação de preços, permitindo observar o crescimento da produção em termos de quantidade. Essa distinção é importante porque o PIB nominal pode aumentar apenas por causa da inflação.

Outro conceito importante é a renda nacional, que mede a renda recebida pelos residentes de um país, considerando também relações com o exterior. Enquanto o PIB está ligado ao território onde a produção ocorre, a renda nacional considera a renda dos residentes, podendo incluir rendimentos recebidos do exterior e excluir rendimentos enviados a não residentes.

As contas nacionais são fundamentais para analisar crescimento econômico, recessão, composição da demanda, participação dos setores, produtividade e políticas econômicas. Governos, empresas, pesquisadores e instituições utilizam esses dados para tomar decisões e avaliar a situação econômica.

PRINCIPAIS AGREGADOS DAS CONTAS NACIONAIS

Os principais agregados das contas nacionais são indicadores que resumem a atividade econômica de uma sociedade. Eles permitem analisar produção, renda, despesa, consumo, poupança,

investimento e relações com o exterior. Esses agregados são fundamentais porque transformam milhões de transações individuais em informações organizadas e comparáveis.

O Produto Interno Bruto é o principal agregado macroeconômico. Ele mede o valor dos bens e serviços finais produzidos dentro do território econômico de um país em determinado período. O termo interno indica que o critério é territorial. Assim, considera-se o que foi produzido dentro do país, independentemente da nacionalidade dos proprietários dos fatores de produção.

O Produto Nacional Bruto, por sua vez, utiliza o critério da nacionalidade ou residência dos agentes econômicos. Ele considera a produção ou renda gerada pelos residentes de um país, ainda que parte dela venha do exterior, e exclui a renda gerada internamente que pertence a não residentes. A diferença entre produto interno e produto nacional está relacionada aos fluxos de renda enviados e recebidos do exterior.

A renda nacional representa a soma das rendas recebidas pelos fatores de produção pertencentes aos residentes. Ela inclui remunerações como salários, lucros, juros, aluguéis e outros rendimentos associados à produção. Esse agregado mostra como o valor produzido se transforma em renda para os agentes econômicos.

A renda disponível é a renda que os agentes, especialmente as famílias, têm efetivamente disponível para consumir ou poupar. Ela considera impostos, contribuições, transferências e outras movimentações que alteram a renda recebida. Esse conceito é importante para entender o comportamento do consumo e da poupança.

O consumo agregado corresponde ao gasto das famílias com bens e serviços finais. Ele é influenciado por renda disponível, preços, taxa de juros, crédito, expectativas e distribuição de renda. Quando o consumo cresce, pode haver estímulo à produção e ao emprego, desde que exista capacidade de oferta.

A poupança representa a parte da renda que não é consumida. Em termos simples, quando famílias, empresas ou governo não gastam toda a renda disponível, há poupança. Essa poupança pode financiar investimentos, diretamente ou por meio do sistema financeiro. A relação entre poupança e investimento é um tema importante da Macroeconomia.

O investimento agregado corresponde à formação de capital na economia. Inclui máquinas, equipamentos, construções, instalações e variações de estoques. O investimento é essencial para ampliar a capacidade produtiva futura. Economias que investem pouco podem ter dificuldade para crescer de forma sustentada.

A absorção interna é a soma do consumo, do investimento e dos gastos do governo. Ela representa a demanda realizada pelos residentes por bens e serviços, independentemente de serem produzidos internamente ou importados. Quando a absorção interna supera a produção doméstica, a diferença tende a aparecer como importações líquidas ou necessidade de financiamento externo.

A despesa agregada é outra forma de observar a atividade econômica. Ela mostra o total gasto na compra de bens e serviços finais. Na identidade macroeconômica básica, a despesa total inclui consumo das famílias, investimento, gastos do governo e exportações líquidas. Essa ótica permite analisar os componentes da demanda.

É importante compreender a relação entre produto, renda e despesa. O produto mede o valor gerado pela produção. A renda mede a remuneração dos fatores usados na produção. A despesa mede a compra dessa produção. Em uma economia, o valor produzido se transforma em renda e é adquirido por algum componente da demanda. Por isso, essas três medidas estão interligadas.

Outro agregado relevante é o produto líquido, obtido ao descontar a depreciação do produto bruto. A depreciação representa o desgaste do capital utilizado na produção. Quando se considera o produto líquido, busca-se medir o acréscimo efetivo de riqueza, depois de descontada a perda de valor dos bens de capital.

Também é importante distinguir valores nominais e reais. Agregados nominais são medidos a preços correntes. Agregados reais descontam os efeitos da inflação, permitindo avaliar variações na quantidade produzida ou consumida. Essa distinção é essencial para não confundir aumento de preços com aumento real da produção.

Os agregados das contas nacionais ajudam a avaliar a saúde econômica. Crescimento do PIB real indica aumento da produção. Crescimento do consumo pode sinalizar maior renda ou crédito. Aumento do investimento pode indicar expansão futura. Desequilíbrios externos podem aparecer nas exportações líquidas. Cada agregado revela uma parte do funcionamento econômico.

AGREGADOS MONETÁRIOS

Os agregados monetários são medidas da quantidade de moeda existente em uma economia. Eles organizam diferentes formas de ativos conforme seu grau de liquidez, isto é, conforme a facilidade com que podem ser usados como meio de pagamento ou convertidos em moeda. Esses agregados são importantes porque ajudam a analisar a oferta de moeda, o comportamento do crédito e os efeitos da política monetária.

A moeda cumpre três funções básicas. A primeira é ser meio de troca, facilitando compras e vendas. A segunda é ser unidade de conta, permitindo expressar preços, salários, dívidas e valores em uma mesma medida. A terceira é ser reserva de valor, permitindo transferir poder de compra para o futuro, embora essa função seja prejudicada quando há inflação elevada.

O conceito mais restrito de moeda costuma incluir o papel-moeda em poder do público e os depósitos à vista. O papel-moeda em poder do público corresponde ao dinheiro físico mantido por pessoas e empresas fora dos bancos. Os depósitos à vista são saldos em contas bancárias que podem ser utilizados imediatamente para pagamentos, transferências e saques.

A soma do papel-moeda em poder do público e dos depósitos à vista forma os meios de pagamento em sentido restrito. Esse agregado representa a moeda mais líquida da economia. Ele indica a quantidade de recursos imediatamente disponíveis para realizar transações.

Além dos meios de pagamento estritos, existem agregados mais amplos que incluem ativos financeiros com alta liquidez, embora não sejam usados diretamente como moeda em todas as transações. Esses ativos podem ser chamados, em termos gerais, de quase moeda. Eles incluem aplicações que podem ser convertidas em meios de pagamento com relativa facilidade.

CIÊNCIA DE DADOS

APRENDIZADO DE MÁQUINA

Aprendizado de máquina é uma área da inteligência artificial dedicada ao desenvolvimento de sistemas capazes de identificar padrões em dados e utilizá-los para realizar previsões, classificações, recomendações ou decisões. Em vez de receber todas as regras explicitamente programadas, o sistema ajusta seu comportamento a partir de exemplos.

Na programação tradicional, o desenvolvedor fornece regras e dados para produzir uma saída. No aprendizado de máquina, dados históricos e resultados conhecidos são utilizados para construir um modelo, que posteriormente recebe novos dados e produz inferências.

Abordagem	Entradas principais	Resultado
Programação tradicional	Dados e regras definidas	Respostas calculadas
Aprendizado de máquina	Dados e exemplos de respostas	Modelo capaz de generalizar
Inferência	Modelo treinado e novos dados	Previsão ou decisão

► Elementos essenciais

Dados, atributos e rótulos

Cada registro utilizado no treinamento é chamado de exemplo ou instância. Os atributos representam características observáveis, como idade, renda, temperatura ou frequência de acesso. O rótulo corresponde à resposta que se deseja prever em problemas supervisionados.

O treinamento ajusta os parâmetros internos do modelo. Após essa etapa, o sistema entra em inferência, utilizando o conhecimento aprendido para analisar dados ainda não vistos. A qualidade dessa generalização é mais importante que o desempenho obtido apenas sobre os exemplos de treinamento.

Parâmetros e hiperparâmetros

Parâmetros são valores aprendidos automaticamente pelo modelo. Hiperparâmetros são definidos antes ou durante os experimentos e controlam aspectos como profundidade de uma árvore, número de vizinhos ou taxa de aprendizagem.

► Paradigmas de aprendizagem

Principais modalidades

As modalidades são diferenciadas pelo tipo de informação disponível durante o treinamento:

- Aprendizado supervisionado utiliza exemplos com respostas conhecidas.
- Aprendizado não supervisionado procura estruturas em dados sem rótulos.
- Aprendizado semissupervisionado combina poucos dados rotulados com muitos dados não rotulados.
- Aprendizado por reforço utiliza recompensas e penalidades para orientar decisões sucessivas.

Classificação prevê categorias, como fraude ou não fraude. Regressão estima valores numéricos, como preço ou consumo. Agrupamento reúne registros semelhantes. Redução de dimensionalidade representa dados complexos por um número menor de variáveis.

► Aplicações e limitações

Usos práticos

Aprendizado de máquina é empregado em reconhecimento de imagens, tradução automática, sistemas de recomendação, análise de crédito, manutenção preditiva, diagnóstico auxiliado por computador e previsão de demanda.

Seu uso não garante decisões corretas. Modelos podem reproduzir vieses, falhar diante de mudanças nos dados ou produzir resultados difíceis de explicar. O desempenho depende da definição adequada do problema, da representatividade dos dados e da avaliação contínua.

Relação com outras áreas

Aprendizado profundo utiliza redes neurais com muitas camadas para lidar com imagens, áudio e linguagem. Inteligência artificial generativa utiliza modelos capazes de criar textos, imagens, sons e outros conteúdos. Ambos pertencem ao campo mais amplo do aprendizado de máquina, mas exigem grande volume de dados e capacidade computacional.

DADOS, PREPARAÇÃO E ENGENHARIA DE ATRIBUTOS

► Coleta e compreensão dos dados

Tipos e fontes

Os dados podem ser estruturados, como tabelas; semiestruturados, como arquivos JSON; ou não estruturados, como textos, imagens e áudios. A origem pode incluir bancos de dados, sensores, formulários, sistemas corporativos, dispositivos móveis e fontes públicas.

Antes do desenvolvimento do modelo, é necessário definir a variável-alvo, a unidade de análise e o período coberto pelos dados. A coleta deve refletir o ambiente no qual o modelo será utilizado.

► **Análise exploratória e qualidade**

Compreensão inicial

A análise exploratória busca identificar distribuições, padrões, relações, valores atípicos e inconsistências. Essa etapa permite reconhecer problemas antes que eles afetem o treinamento.

Os principais aspectos de qualidade incluem:

- Completude, relacionada à presença das informações necessárias.
- Consistência, referente à ausência de contradições entre registros.
- Precisão, ligada à proximidade entre o dado armazenado e a realidade.
- Representatividade, relacionada à cobertura adequada da população de interesse.
- Atualidade, referente à compatibilidade temporal dos dados com o problema.

Valores ausentes podem ser removidos, preenchidos ou tratados como categoria específica. Duplicidades devem ser investigadas. Valores extremos podem representar erros ou ocorrências legítimas e não devem ser eliminados automaticamente.

► **Transformações e engenharia de atributos**

Preparação para o treinamento

Variáveis categóricas precisam ser convertidas em representações numéricas. Variáveis numéricas podem ser normalizadas ou padronizadas para evitar que escalas muito diferentes prejudiquem determinados algoritmos.

Técnica	Finalidade
Codificação categórica	Transformar categorias em números
Padronização	Centralizar e ajustar a escala
Normalização	Colocar valores em intervalo definido
Seleção de atributos	Remover variáveis pouco úteis
Extração de atributos	Criar representações mais informativas
Redução de dimensionalidade	Diminuir a quantidade de variáveis

Engenharia de atributos consiste em criar novas variáveis a partir de conhecimento do domínio. Datas podem gerar dia da semana, mês ou duração. Valores financeiros podem originar proporções e médias. Em textos, palavras podem ser convertidas em vetores numéricos.

► **Divisão dos dados**

Treinamento, validação e teste

O conjunto de treinamento é usado para ajustar o modelo. O conjunto de validação auxilia na escolha de hiperparâmetros. O conjunto de teste deve permanecer separado até a avaliação final.

Validação cruzada divide os dados em diferentes partes e repete o treinamento, produzindo estimativas mais estáveis. Em séries temporais, a ordem cronológica precisa ser preservada.

Vazamento de dados ocorre quando o modelo recebe, durante o treinamento, informações que não estariam disponíveis no uso real. Isso gera desempenho artificialmente elevado.

► **Governança e proteção**

Rastreabilidade e privacidade

Dados, transformações e versões devem ser documentados. Informações pessoais ou sensíveis exigem controle de acesso, anonimização quando aplicável e finalidade claramente definida. A qualidade técnica do modelo não justifica coleta excessiva ou uso incompatível com os direitos dos titulares.

MODELOS, TREINAMENTO E AVALIAÇÃO

► **Escolha do algoritmo**

Adequação ao problema

A escolha depende do tipo de tarefa, volume de dados, necessidade de interpretação, custo computacional e formato das informações. Não existe algoritmo universalmente superior.

Modelo	Uso frequente	Característica
Regressão linear	Previsão numérica	Simple e interpretável
Regressão logística	Classificação	Produz probabilidades
Árvore de decisão	Classificação e regressão	Regras facilmente visualizadas
Floresta aleatória	Classificação e regressão	Combina várias árvores
K-vizinhos	Classificação e regressão	Baseado em proximidade
Máquina de vetores de suporte	Classificação	Busca separação entre classes
K-means	Agrupamento	Forma grupos por similaridade
Rede neural	Tarefas complexas	Aprende representações não lineares

► **Processo de treinamento**

Função de custo e otimização

O treinamento procura reduzir uma função de custo, que mede a diferença entre as previsões e os resultados esperados. Em muitos modelos, o gradiente descendente ajusta os parâmetros na direção que reduz o erro.

A taxa de aprendizagem controla o tamanho de cada ajuste. Valores muito altos podem impedir a convergência, enquanto valores muito baixos tornam o treinamento lento.

SEGURANÇA DA INFORMAÇÃO

GESTÃO DE IDENTIDADES E ACESSO: AUTENTICAÇÃO E AUTORIZAÇÃO, SINGLE SIGN-ON (SSO), SECURITY ASSERTION MARKUP LANGUAGE (SAML), OAUTH2 E OPENID CONNECT

A Gestão de Identidades e Acessos, conhecida pela sigla IAM — Identity and Access Management — reúne políticas, processos e tecnologias destinados a administrar quem pode acessar os recursos de uma organização, em quais condições e com quais permissões. Seu objetivo é assegurar que cada usuário, dispositivo, aplicação ou serviço possua acesso adequado às informações e aos sistemas necessários para executar suas atividades.

A identidade digital representa uma entidade em um ambiente computacional. Ela pode corresponder a uma pessoa, a uma conta de serviço, a uma aplicação, a um equipamento ou a outro componente autorizado a interagir com o sistema. Essa identidade é normalmente associada a atributos, como nome, cargo, departamento, função, localização e nível de privilégio.

É importante distinguir alguns conceitos relacionados. A conta é o registro utilizado para representar a identidade em determinado sistema. A credencial é a evidência apresentada para comprovar essa identidade, como uma senha, um certificado digital ou um token. O perfil reúne características e permissões atribuídas ao usuário, enquanto o privilégio corresponde a uma autorização específica para realizar determinada ação.

► Autenticação

Autenticação é o processo de verificar se uma entidade é realmente quem afirma ser. Essa verificação ocorre mediante a apresentação de um ou mais fatores de autenticação.

Os fatores mais utilizados podem ser compreendidos da seguinte forma:

- Fator de conhecimento: algo que o usuário sabe, como senha ou código pessoal.
- Fator de posse: algo que o usuário possui, como celular, cartão ou token físico.
- Fator inerente: algo que pertence ao usuário, como impressão digital ou reconhecimento facial.
- Fator comportamental: característica relacionada ao modo de agir, como padrão de digitação.

Quando apenas um fator é exigido, ocorre autenticação de fator único. Na autenticação multifator, dois ou mais fatores de categorias diferentes são combinados, reduzindo o risco de acesso indevido em caso de comprometimento de uma senha.

► Autorização

Autorização é o processo que determina quais recursos e operações uma identidade autenticada pode utilizar. Enquanto a autenticação responde à pergunta “quem é você?”, a autorização responde “o que você pode fazer?”.

As permissões podem ser atribuídas diretamente ao usuário ou vinculadas a funções, grupos, atributos e políticas. Em ambientes corporativos, a autorização deve respeitar o princípio do menor privilégio, segundo o qual cada identidade recebe somente os acessos estritamente necessários ao desempenho de suas atividades.

► Ciclo de vida das identidades

A gestão de identidades deve abranger todo o período de relacionamento da entidade com a organização. Isso inclui criação da conta, concessão inicial de acessos, alterações decorrentes de mudanças de função, revisões periódicas, suspensão e exclusão.

Falhas nesse processo podem gerar contas órfãs, privilégios excessivos e acessos indevidos. Por esse motivo, autenticação, autorização, auditoria e revisão de permissões devem funcionar de forma integrada, garantindo rastreabilidade, conformidade e proteção dos ativos de informação.

SINGLE SIGN-ON E FEDERAÇÃO DE IDENTIDADES

► Conceito e finalidade do Single Sign-On

Single Sign-On, normalmente identificado pela sigla SSO, é um modelo de autenticação no qual o usuário realiza uma única autenticação principal e, a partir dela, obtém acesso a diferentes sistemas integrados sem precisar informar novamente suas credenciais em cada aplicação. A tradução mais comum é “autenticação única”, embora o conceito não signifique necessariamente que o usuário possua apenas uma conta, mas que diversas aplicações confiem em uma mesma fonte de autenticação.

Em um ambiente sem SSO, cada aplicação pode manter seu próprio cadastro de usuários, suas próprias senhas e suas próprias regras de autenticação. Esse modelo aumenta a complexidade administrativa, favorece a reutilização de senhas e dificulta a aplicação uniforme de controles de segurança. No SSO, a autenticação é centralizada em um componente confiável, permitindo que as aplicações deleguem a ele a verificação da identidade.

O funcionamento geral do SSO envolve uma relação de confiança entre os sistemas participantes. Quando o usuário tenta acessar uma aplicação, ela verifica se existe uma sessão válida. Caso não exista, o usuário é direcionado ao serviço central de identidade. Após a autenticação, esse serviço emite uma declaração, resposta ou token que informa à aplicação que a identidade foi validada. A aplicação analisa essa informação e, se ela for legítima, concede o acesso.

► Componentes de uma arquitetura de SSO

Uma arquitetura de SSO normalmente envolve dois componentes principais: o provedor de identidade e o provedor de serviço. O provedor de identidade, conhecido como Identity Provider ou IdP, é responsável por autenticar o usuário e emitir informações confiáveis sobre sua identidade. O provedor de serviço, chamado Service Provider ou SP, representa a aplicação que oferece o recurso solicitado.

O processo pode ser compreendido pelas seguintes etapas fundamentais:

- O usuário solicita acesso a uma aplicação protegida.
- A aplicação verifica se existe uma sessão de acesso válida.
- Na ausência de sessão, o usuário é redirecionado ao provedor de identidade.
- O provedor de identidade solicita e valida as credenciais.
- Após a autenticação, uma resposta confiável é enviada à aplicação.
- A aplicação valida a resposta e cria uma sessão para o usuário.

É importante observar que a senha do usuário não precisa ser enviada a todas as aplicações. As credenciais são apresentadas ao provedor de identidade, enquanto os demais sistemas recebem apenas a confirmação da autenticação e, quando necessário, determinados atributos da identidade.

► SSO, sessão e tokens

O SSO depende da existência de sessões e de artefatos de segurança capazes de transportar informações entre os sistemas. Esses artefatos podem assumir a forma de assertions, códigos de autorização, tokens ou respostas assinadas digitalmente, conforme o protocolo utilizado.

A sessão do provedor de identidade representa o estado de autenticação central do usuário. Já a sessão da aplicação representa o acesso específico àquele sistema. Essas sessões podem possuir tempos de validade diferentes. Por isso, encerrar a sessão em uma aplicação nem sempre significa encerrar automaticamente a sessão existente no provedor de identidade ou nas demais aplicações.

O encerramento centralizado de todas as sessões é conhecido como Single Logout. Embora seja desejável em muitos ambientes, sua implementação pode ser complexa, pois exige integração adequada entre todos os serviços. Uma aplicação que não processe corretamente a solicitação de encerramento poderá manter a sessão ativa mesmo depois de o usuário sair de outro sistema.

► SSO e sincronização de senhas

O SSO não deve ser confundido com sincronização de senhas. Na sincronização, diferentes sistemas recebem ou mantêm senhas iguais, mas cada aplicação continua realizando sua própria autenticação. No SSO, as aplicações confiam em um serviço central, não sendo necessário replicar a credencial principal entre os sistemas.

Também existe diferença entre SSO e cofres de credenciais. Um cofre pode armazenar senhas e preenchê-las automaticamente em diferentes aplicações. Apesar de melhorar a experiência do

usuário, esse mecanismo não estabelece necessariamente uma federação de identidade nem elimina a autenticação independente de cada sistema.

► Federação de identidades

Federação de identidades é o estabelecimento de relações de confiança que permitem a utilização de uma identidade administrada em um domínio para acessar recursos pertencentes a outro domínio. Nesse modelo, organizações ou ambientes distintos concordam em confiar nas informações de autenticação emitidas por um provedor de identidade.

Um exemplo ocorre quando um funcionário utiliza sua conta corporativa para acessar uma plataforma externa contratada pela empresa. A plataforma não precisa armazenar a senha corporativa do usuário. Ela confia na declaração emitida pelo provedor de identidade da organização e cria o acesso com base nos atributos recebidos.

O SSO pode existir dentro de uma única organização sem que haja federação. A federação, por sua vez, geralmente viabiliza o SSO entre domínios administrativos diferentes. Assim, os conceitos são relacionados, mas não são equivalentes.

► Benefícios e riscos

O SSO reduz a quantidade de credenciais administradas pelo usuário, facilita a aplicação de autenticação multifator e permite centralizar políticas de senha, bloqueio, monitoramento e resposta a incidentes. Também simplifica a revogação de acessos, pois a suspensão da identidade central pode impedir o acesso a diversos serviços integrados.

Entretanto, a centralização cria um ponto de alta criticidade. Se a conta central for comprometida, o invasor poderá alcançar múltiplos sistemas. Da mesma forma, uma falha ou indisponibilidade no provedor de identidade pode impedir o acesso a todas as aplicações dependentes.

Para reduzir esses riscos, alguns controles são especialmente importantes:

- Exigir autenticação multifator, principalmente para contas administrativas.
- Aplicar políticas de sessão compatíveis com o risco de cada aplicação.
- Utilizar autenticação adaptativa baseada em contexto e comportamento.
- Proteger chaves, certificados e segredos utilizados nas integrações.
- Monitorar tentativas anômalas, acessos incomuns e emissão irregular de tokens.
- Revisar periodicamente as relações de confiança e os atributos compartilhados.

Quando corretamente implementados, SSO e federação de identidades melhoram simultaneamente a experiência do usuário, a governança de acessos e a aplicação centralizada de controles de segurança. Contudo, sua adoção exige arquitetura resiliente, validação rigorosa de mensagens e proteção reforçada do provedor de identidade.

ENGENHARIA DE SOFTWARE

ARQUITETURA DE SISTEMAS WEB: PROTOCOLO HTTP, HTTP/2, GRPC, WEBSOCKETS, TLS, SERVIDORES PROXY, CACHE, DNS, BALANCEAMENTO DE CARGA, TOLERÂNCIA A FALHAS E ESCALABILIDADE EM SISTEMAS WEB

FUNDAMENTOS DA COMUNICAÇÃO WEB

► Caminho completo de uma requisição

Quando um usuário informa um endereço no navegador, a resposta exibida na tela é resultado de uma sequência de operações distribuídas entre diferentes componentes. A primeira etapa consiste em interpretar o endereço informado, normalmente uma URL. Essa URL contém o protocolo, o nome do host, o caminho do recurso e, eventualmente, parâmetros adicionais.

Antes de enviar qualquer requisição, o cliente precisa descobrir qual endereço de rede corresponde ao nome informado. Essa resolução normalmente ocorre por meio do DNS. Depois de obter o endereço IP, o cliente estabelece uma conexão com o destino. Em conexões baseadas em TCP, ocorre inicialmente a negociação da conexão de transporte. Quando o acesso utiliza HTTPS, há também a negociação TLS, responsável pela proteção criptográfica da comunicação.

Somente após essas etapas o cliente envia a requisição da aplicação. Em HTTP, a mensagem contém método, caminho, cabeçalhos e, em alguns casos, corpo. Essa requisição pode não chegar diretamente ao servidor da aplicação. É comum que passe antes por um proxy reverso, balanceador de carga, cache intermediário ou mecanismo de proteção.

Cada intermediário pode executar tarefas específicas. Um balanceador escolhe uma instância de destino. Um proxy pode encerrar a conexão TLS, aplicar políticas e reencaminhar a mensagem. Um cache pode devolver uma resposta armazenada e evitar que a aplicação seja consultada.

Depois de receber a requisição, o servidor executa o processamento necessário e produz uma resposta. Essa resposta retorna pelo caminho inverso, podendo ser modificada ou armazenada pelos intermediários.

► Relação entre cliente, rede, servidores e intermediários

O cliente é o componente que inicia a comunicação. Pode ser um navegador, aplicativo móvel, sistema corporativo ou qualquer programa capaz de se comunicar pelos protocolos utilizados.

O servidor é o componente que recebe a solicitação e produz uma resposta. Em arquiteturas reais, “servidor” não representa necessariamente uma única máquina. Pode representar um conjunto de instâncias distribuídas, acessadas por meio de uma mesma entrada lógica.

A rede é responsável por transportar os dados. Sua qualidade influencia diretamente o tempo de resposta. Latência, perda de pacotes, largura de banda e distância geográfica afetam a comunicação mesmo quando a aplicação está funcionando corretamente.

Os intermediários ocupam posição estratégica. Eles separam o cliente da infraestrutura interna e permitem aplicar funções sem alterar necessariamente a aplicação. Um proxy pode esconder os servidores reais. Um balanceador pode distribuir conexões. Um cache pode reduzir o tempo de resposta. Um terminador TLS pode centralizar certificados.

Essa separação facilita a evolução da infraestrutura. Servidores podem ser substituídos, ampliados ou reorganizados sem que o cliente conheça os detalhes internos.

► Latência, conexão, transmissão e processamento

O tempo percebido pelo usuário é composto por várias parcelas. A latência de rede representa o tempo necessário para que os dados percorram o caminho entre as partes. A criação de conexões adiciona outras etapas, especialmente quando há negociação TLS.

A transmissão depende do tamanho das mensagens e da largura de banda disponível. Conteúdos maiores levam mais tempo para serem transferidos, principalmente em redes lentas.

O processamento corresponde ao tempo gasto pelos componentes para interpretar a requisição, executar operações e preparar a resposta. Um sistema pode apresentar baixa latência de rede e ainda responder lentamente por causa de processamento excessivo.

A arquitetura precisa observar o tempo total, e não apenas uma etapa isolada. Melhorar o servidor pode produzir pouco efeito quando o principal problema está na rede. Da mesma forma, aumentar largura de banda não resolve uma aplicação que processa cada requisição de maneira ineficiente.

O desempenho final depende do equilíbrio entre transporte, processamento, armazenamento e intermediação.

PROTOCOLO HTTP

► Estrutura de requisições e respostas

HTTP é um protocolo de aplicação baseado na troca de mensagens entre cliente e servidor. O cliente envia uma requisição, e o servidor devolve uma resposta.

Em HTTP/1.1, uma requisição textual contém uma linha inicial, cabeçalhos, uma linha em branco e, opcionalmente, um corpo. A linha inicial informa o método, o recurso solicitado e a versão do protocolo. Os cabeçalhos carregam metadados. O corpo transporta os dados da operação quando necessário.

Uma resposta segue estrutura semelhante. Sua linha inicial contém a versão do protocolo, o código de status e uma descrição textual. Depois aparecem os cabeçalhos e, quando aplicável, o corpo.

O protocolo organiza a comunicação, mas não define a lógica interna do sistema. Ele indica como a mensagem deve ser estruturada e interpretada.

► Métodos, cabeçalhos, corpo e códigos de status

Os métodos representam a intenção da operação. GET é usado para obter uma representação. POST normalmente envia dados para processamento. PUT costuma indicar substituição integral. PATCH representa alteração parcial. DELETE solicita remoção. HEAD devolve cabeçalhos equivalentes aos de uma resposta GET, mas sem corpo. OPTIONS informa possibilidades de comunicação.

Alguns métodos são classificados como seguros, pois não deveriam alterar o estado do servidor. GET e HEAD são exemplos. Outros podem ser idempotentes, isto é, múltiplas execuções equivalentes devem produzir o mesmo efeito final que uma única execução. PUT e DELETE são normalmente tratados dessa forma.

Os cabeçalhos permitem transportar informações como tipo do conteúdo, tamanho, idioma, autenticação, cache, cookies e origem da requisição. Eles também possibilitam negociação de representação.

O corpo pode conter diferentes formatos, como texto, formulários, arquivos ou estruturas serializadas.

Os códigos de status são agrupados em categorias. A faixa 1xx é informativa. A faixa 2xx indica sucesso. A faixa 3xx trata de redirecionamentos. A faixa 4xx representa falhas associadas à requisição. A faixa 5xx indica erro no processamento do servidor.

Essa classificação permite ao cliente distinguir falha de autenticação, recurso inexistente, redirecionamento ou erro interno.

► Persistência de conexões

Criar uma conexão para cada requisição adiciona custo. Por isso, HTTP/1.1 utiliza conexões persistentes por padrão. O cliente pode enviar várias requisições sucessivas pela mesma conexão.

A reutilização reduz o número de negociações de transporte e TLS. Contudo, no HTTP/1.1, o paralelismo dentro de uma mesma conexão é limitado. Na prática, navegadores costumavam abrir várias conexões em paralelo para contornar essa limitação.

Essa estratégia melhora o uso da rede, mas também aumenta a quantidade de conexões mantidas pelo servidor.

► Ausência de estado e manutenção de sessões

HTTP é considerado sem estado porque cada requisição pode ser tratada de forma independente. O protocolo não exige que o servidor lembre interações anteriores.

Aplicações frequentemente precisam preservar contexto, como autenticação, preferências ou carrinho de compras. Para isso, utilizam mecanismos adicionais.

Cookies permitem que o servidor envie dados ao cliente, que os devolve em requisições futuras. Um cookie pode conter o próprio estado ou um identificador associado a dados mantidos no servidor.

Quando a sessão é armazenada localmente em uma instância, surge dependência daquele servidor. Isso afeta balanceamento e escalabilidade. Uma alternativa é utilizar armazenamento compartilhado ou transportar informações verificáveis em cada requisição.

► Controle de conteúdo e cache no protocolo

HTTP possui mecanismos próprios para armazenamento e validação de respostas. O cabeçalho `Cache-Control` pode definir se a resposta é pública, privada, reutilizável ou não armazenável.

O cabeçalho `Expires` informa uma data de expiração. Já mecanismos como `ETag` e `Last-Modified` permitem validação condicional.

Quando o cliente possui uma versão armazenada, pode perguntar se ela ainda é válida. Se não houver alteração, o servidor responde com o status 304, sem retransmitir o corpo completo.

Esses recursos reduzem tráfego e tempo de resposta, mas exigem configuração adequada. Uma política incorreta pode fazer o cliente reutilizar conteúdo desatualizado ou armazenar informações que não deveriam ser compartilhadas.

HTTP/2

► Limitações do HTTP/1.1

HTTP/1.1 foi amplamente utilizado, mas apresenta limitações quando uma página depende de muitos recursos. Navegadores precisam solicitar documentos, scripts, imagens, fontes e folhas de estilo. Embora conexões persistentes reduzam parte do custo, uma mesma conexão não oferece multiplexação plena.

Como resposta prática, clientes costumavam abrir várias conexões paralelas para o mesmo destino. Isso aumenta uso de recursos e repete custos de estabelecimento.

Outro problema ocorre quando uma resposta lenta bloqueia requisições posteriores na mesma conexão. Essa limitação é conhecida como bloqueio na camada de aplicação.

HTTP/2 foi projetado para reduzir esses problemas sem abandonar a semântica do HTTP.

► Estrutura binária

Diferentemente do formato textual do HTTP/1.1, HTTP/2 utiliza uma camada binária de enquadramento. As mensagens são divididas em unidades chamadas frames.

Esses frames possuem campos estruturados que identificam tipo, tamanho, sinalizadores e fluxo ao qual pertencem. Isso torna a interpretação mais eficiente e menos ambígua para máquinas.

A mudança para formato binário não altera o significado de métodos, códigos de status e cabeçalhos. A aplicação continua trabalhando com os mesmos conceitos. A diferença está na forma como os dados são transportados entre as partes.

► Streams e frames

Um stream é um fluxo lógico bidirecional dentro de uma conexão HTTP/2. Cada stream possui um identificador e transporta uma requisição e sua resposta.

As mensagens são fragmentadas em frames. Frames de cabeçalho transportam metadados. Frames de dados transportam o corpo. Outros tipos controlam prioridade, encerramento, janelas de fluxo e estado da conexão.

INFRAESTRUTURA EM TI

CONCEITOS DE INFRAESTRUTURA COMO CÓDIGO E AUTOMAÇÃO DE INFRAESTRUTURA DE TI

Infraestrutura como código, frequentemente abreviada como IaC, é a prática de definir servidores, redes, armazenamento, bancos de dados, permissões, serviços de nuvem e outros recursos por meio de arquivos versionáveis. Em vez de configurar cada componente manualmente, a equipe descreve a infraestrutura desejada e utiliza ferramentas para criá-la, alterá-la ou removê-la de forma automatizada.

Essa abordagem aproxima a administração de infraestrutura das práticas de engenharia de software. As definições podem ser armazenadas em repositórios, revisadas por outras pessoas, testadas antes da aplicação e associadas a versões específicas.

A automação de infraestrutura é um conceito mais amplo. Ela inclui IaC, scripts, gerenciamento de configuração, orquestração, implantação de aplicações, execução de tarefas operacionais e resposta automática a eventos.

► Infraestrutura física, virtual e em nuvem

Recursos administrados pela automação

A infraestrutura pode incluir equipamentos físicos, máquinas virtuais, contêineres, redes, balanceadores, bancos de dados e serviços gerenciados. Em ambientes locais, a automação pode configurar servidores e equipamentos de rede. Em nuvem, APIs permitem provisionar recursos sob demanda.

Os principais objetivos da automação são apresentados a seguir:

- Padronizar ambientes e reduzir diferenças entre configurações.
- Reproduzir rapidamente recursos em desenvolvimento, homologação e produção.
- Diminuir erros causados por atividades manuais repetitivas.
- Registrar alterações e facilitar auditorias.
- Aumentar a velocidade de implantação e recuperação.
- Permitir escalabilidade e criação de ambientes temporários.

► Abordagens declarativa e imperativa

Estado desejado e sequência de comandos

Na abordagem declarativa, o código descreve o estado final esperado. A ferramenta compara o estado atual com o desejado e calcula as operações necessárias. Na abordagem imperativa, o código informa uma sequência específica de comandos.

A abordagem declarativa costuma ser adequada ao gerenciamento contínuo de recursos. A abordagem imperativa é útil quando a ordem das operações ou uma lógica procedural precisa ser controlada diretamente.

Idempotência significa que uma operação pode ser executada repetidamente sem produzir alterações adicionais quando o estado desejado já foi atingido. Essa característica é essencial para evitar duplicação ou inconsistência.

► Infraestrutura mutável e imutável

Alteração ou substituição dos recursos

Na infraestrutura mutável, um servidor existente recebe atualizações e alterações de configuração. Na infraestrutura imutável, uma nova imagem ou instância é criada e substitui a anterior. A imutabilidade reduz diferenças acumuladas, embora possa exigir maior planejamento para dados persistentes.

Drift é a divergência entre o código declarado e o ambiente real, normalmente causada por mudanças manuais. A detecção de drift permite identificar recursos alterados fora do processo aprovado.

IaC melhora reprodutibilidade e governança, mas não elimina riscos. Um erro no código pode ser propagado em larga escala. Por isso, revisão, testes, segurança e aprovação continuam indispensáveis.

FERRAMENTAS, LINGUAGENS E ORGANIZAÇÃO DOS PROJETOS

► Categorias de ferramentas

Provisionamento, configuração e empacotamento

Ferramentas de infraestrutura como código possuem finalidades distintas. Algumas criam recursos, outras configuram sistemas já existentes e outras produzem imagens reutilizáveis.

A divisão mais comum pode ser compreendida da seguinte forma:

Categoria	Finalidade
Provisionamento	Criar redes, máquinas, bancos e serviços
Configuração	Instalar pacotes e ajustar sistemas
Imagens	Produzir modelos prontos de máquinas ou contêineres
Orquestração	Coordenar serviços, contêineres e dependências
Pipelines	Validar e implantar alterações automaticamente

Terraform e OpenTofu utilizam uma abordagem declarativa e trabalham com diferentes provedores. CloudFormation e Bicep são voltados a ambientes específicos de nuvem. Pulumi permite definir infraestrutura com linguagens de programação. Ansible

é frequentemente utilizado para configuração e automação de tarefas. Packer cria imagens de máquinas, enquanto Docker empacota aplicações em contêineres. Kubernetes administra a execução distribuída desses contêineres.

► Critérios para escolha

Adequação ao ambiente

A escolha deve considerar portabilidade, suporte ao provedor, curva de aprendizagem, integração com pipelines, comunidade, segurança e capacidade de manutenção. Uma ferramenta amplamente adotada nem sempre é a melhor para todos os contextos.

Scripts em Shell, PowerShell ou Python continuam úteis para tarefas específicas. Contudo, scripts extensos podem perder idempotência, documentação e controle de estado, tornando-se difíceis de manter.

► Organização do repositório

Módulos, ambientes e parâmetros

Projetos de IaC devem separar componentes reutilizáveis, configurações por ambiente e variáveis sensíveis. Módulos agrupam recursos relacionados e facilitam padronização.

Uma organização adequada costuma contemplar:

- Módulos reutilizáveis para redes, servidores e bancos de dados.
- Configurações específicas para desenvolvimento, homologação e produção.
- Variáveis para valores que mudam entre ambientes.
- Saídas para compartilhar identificadores e endereços entre componentes.
- Documentação sobre pré-requisitos, dependências e formas de uso.

Módulos devem possuir versões. Alterações incompatíveis não devem ser introduzidas sem controle, pois vários ambientes podem depender da versão anterior.

► Gestão do estado

Relação entre código e recursos reais

Algumas ferramentas mantêm um arquivo de estado que relaciona recursos declarados aos objetos existentes. Esse estado pode conter identificadores, endereços e informações sensíveis.

Em equipes, o estado deve ser armazenado remotamente, protegido por criptografia e controle de acesso. O bloqueio impede que duas execuções alterem o mesmo ambiente simultaneamente.

Recursos criados manualmente podem ser importados, mas isso exige cuidado para que o código passe a representar corretamente a infraestrutura existente. Estados excessivamente grandes aumentam acoplamento e risco operacional. A separação por domínio, ambiente ou ciclo de vida reduz esse problema.

PIPELINES, TESTES, SEGURANÇA E IMPLANTAÇÃO AUTOMATIZADA

► Ciclo de desenvolvimento da infraestrutura

Da alteração à aplicação

Uma mudança em infraestrutura deve seguir processo semelhante ao desenvolvimento de software. O código é escrito, formatado, validado, revisado e testado antes de alcançar o ambiente real.

O fluxo pode incluir as seguintes etapas:

- Validação sintática e verificação da estrutura dos arquivos.
- Análise estática para identificar configurações inseguras.
- Geração de um plano que mostre recursos criados, alterados ou destruídos.
- Revisão técnica e aprovação das mudanças.
- Aplicação automatizada no ambiente autorizado.
- Verificação posterior de disponibilidade, segurança e desempenho.

A geração do plano é particularmente importante porque permite visualizar efeitos antes da execução. Alterações destrutivas devem receber atenção especial.

► Testes de infraestrutura

Validação em diferentes níveis

Testes unitários verificam módulos isolados. Testes de integração confirmam a comunicação entre recursos. Testes de conformidade avaliam políticas de segurança e padrões organizacionais. Ambientes temporários permitem validar a infraestrutura de forma próxima à realidade.

Testes também devem abranger destruição e recriação. Uma infraestrutura que pode ser criada, mas não removida corretamente, produz custos e resíduos operacionais.

► Implantação e recuperação

Estratégias de mudança segura

Alterações podem ser implantadas gradualmente. Rolling deployment substitui instâncias aos poucos. Blue-green mantém dois ambientes e alterna o tráfego. Canary direciona apenas pequena parcela das requisições à nova versão.

Rollback em infraestrutura nem sempre é simples. Recursos com estado, especialmente bancos de dados, podem sofrer alterações irreversíveis. Por isso, infraestrutura e dados persistentes devem ter ciclos de vida separados.

Backups, replicação e planos de recuperação precisam ser automatizados e testados. Criar backups sem validar restauração não garante continuidade.

► Segredos, acessos e cadeia de suprimentos

Proteção do processo automatizado

Senhas, tokens, chaves e certificados não devem ser incluídos diretamente no código. Cofres de segredos permitem armazenamento, rotação e entrega controlada de credenciais.

BANCOS DE DADOS

SGBDS SQL E NOSQL

Banco de dados é uma coleção organizada de informações relacionadas, estruturada para permitir armazenamento, consulta, atualização e compartilhamento. O Sistema Gerenciador de Banco de Dados, ou SGBD, é o software responsável por administrar essa coleção, oferecendo mecanismos de segurança, integridade, concorrência, recuperação e controle de acesso.

A aplicação utiliza o SGBD para manipular as informações, mas não precisa conhecer todos os detalhes físicos do armazenamento. Essa separação promove independência entre dados e programas, permitindo alterações na estrutura interna sem modificar completamente os sistemas consumidores.

As funções centrais de um SGBD incluem:

- Definir estruturas, tipos de dados, relacionamentos e restrições.
- Executar consultas, inserções, atualizações e exclusões.
- Controlar acessos simultâneos realizados por diferentes usuários.
- Impedir operações que violem regras de integridade.
- Recuperar informações após falhas de hardware ou software.
- Gerenciar usuários, permissões, auditoria, cópias e restauração.

► Modelo relacional

Tabelas, linhas e colunas

O modelo relacional representa os dados por meio de relações, normalmente implementadas como tabelas. Cada linha corresponde a uma ocorrência ou registro, enquanto as colunas representam atributos. O domínio determina os valores permitidos para cada atributo.

Esquema é a definição estrutural do banco, incluindo tabelas, colunas, tipos e restrições. Instância é o conjunto de dados existente em determinado momento.

Elemento	Função
Tabela	Representar uma entidade ou associação
Linha	Armazenar uma ocorrência
Coluna	Representar uma característica
Domínio	Definir valores permitidos
Esquema	Descrever a estrutura do banco
Instância	Representar o conteúdo atual

► Chaves e integridade

Identificação e relacionamento

A chave primária identifica cada linha de forma única e não deve possuir valor nulo. Uma tabela pode ter várias chaves candidatas, mas apenas uma é escolhida como primária. Chaves compostas utilizam mais de uma coluna.

A chave estrangeira referencia uma chave existente em outra tabela, permitindo representar relacionamentos. A integridade referencial impede, por exemplo, que um pedido seja associado a um cliente inexistente.

A integridade de entidade garante a identificação dos registros. A integridade de domínio verifica tipos, formatos e intervalos. Restrições adicionais podem assegurar unicidade, obrigatoriedade ou condições específicas.

► Normalização

Redução de redundâncias e anomalias

Normalização organiza os dados para reduzir duplicidades e dependências inadequadas. A primeira forma normal exige valores atômicos. A segunda elimina dependências parciais de chaves compostas. A terceira reduz dependências transitivas entre atributos não identificadores.

Uma modelagem inadequada pode provocar anomalias de inserção, atualização e exclusão. Contudo, em sistemas analíticos ou de alta leitura, pode-se aplicar desnormalização controlada para reduzir junções e melhorar desempenho.

Entre os SGBDs relacionais mais conhecidos estão PostgreSQL, MySQL, MariaDB, Oracle Database, Microsoft SQL Server e SQLite. Suas principais vantagens são consistência, transações robustas e linguagem padronizada. Como limitações, podem apresentar maior rigidez estrutural e desafios de escalabilidade horizontal em determinados cenários.

LINGUAGEM SQL, TRANSAÇÕES E DESEMPENHO

► Organização da linguagem SQL

Categorias de comandos

SQL é a linguagem utilizada para definir, consultar, manipular e controlar bancos relacionais. Embora existam diferenças entre produtos, seus princípios fundamentais são amplamente compartilhados.

As categorias tradicionais organizam os comandos conforme sua finalidade:

- DDL define estruturas por meio de comandos como CREATE, ALTER e DROP.
- DML manipula registros com INSERT, UPDATE e DELETE.
- DQL consulta informações, principalmente por meio de SELECT.

- DCL controla permissões com comandos como GRANT e REVOKE.
- TCL administra transações com COMMIT, ROLLBACK e SAVEPOINT.

Restrições como NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY e CHECK ajudam a impedir dados inválidos já no momento da gravação.

► Consultas e combinações

Filtragem, agrupamento e junções

SELECT define as colunas recuperadas. WHERE filtra linhas, ORDER BY organiza o resultado, GROUP BY agrupa registros e HAVING restringe grupos após a agregação. Funções como COUNT, SUM, AVG, MIN e MAX resumem valores.

Junções conectam tabelas relacionadas:

Junção	Resultado
INNER JOIN	Apenas correspondências existentes nos dois lados
LEFT JOIN	Todas as linhas da esquerda e correspondências da direita
RIGHT JOIN	Todas as linhas da direita e correspondências da esquerda
FULL JOIN	Todas as linhas de ambas as tabelas

Subconsultas permitem incorporar uma consulta dentro de outra. Common Table Expressions organizam consultas complexas em etapas nomeadas. Visões apresentam resultados armazenados logicamente, sem necessariamente duplicar os dados.

Procedimentos, funções e gatilhos podem centralizar regras no banco, mas o uso excessivo aumenta acoplamento e dificulta manutenção.

► Transações e propriedades ACID

Confiabilidade das operações

Transação é uma unidade lógica composta por uma ou mais operações. As propriedades ACID asseguram comportamento confiável:

- Atomicidade garante que todas as operações sejam concluídas ou desfeitas.
- Consistência preserva regras e restrições após a transação.
- Isolamento controla interferências entre transações concorrentes.
- Durabilidade mantém alterações confirmadas mesmo após falhas.

Níveis de isolamento controlam fenômenos como leituras sujas, leituras não repetíveis e leituras fantasmas. Quanto maior o isolamento, maior a proteção, mas também podem aumentar bloqueios e custos.

SGBDs utilizam bloqueios ou controle multiversão para administrar concorrência. O controle multiversão permite que leituras utilizem versões consistentes sem bloquear necessariamente as escritas.

► Índices e otimização

Acesso eficiente aos dados

Índices aceleram consultas ao criar estruturas auxiliares de localização. Podem ser simples ou compostos, agrupados ou não agrupados. Entretanto, cada índice consome espaço e aumenta o custo de inserções e atualizações.

Planos de execução mostram como o SGBD pretende acessar tabelas, aplicar filtros e realizar junções. Consultas eficientes evitam leituras desnecessárias, utilizam índices adequados e reduzem operações de alto custo.

Particionamento divide tabelas extensas em segmentos. Replicação mantém cópias para disponibilidade e leitura. Backups completos, incrementais e registros de transação permitem recuperação após falhas. Segurança deve incluir papéis, privilégios mínimos, criptografia, auditoria e proteção das credenciais.

SGBDs NOSQL E SEUS PRINCIPAIS MODELOS

► Conceito e motivações

Flexibilidade e distribuição

NoSQL designa uma família de bancos não relacionais desenvolvidos para cenários que exigem esquema flexível, escalabilidade horizontal, distribuição geográfica ou processamento de grandes volumes. O termo não significa ausência completa de consultas ou estruturas, mas afastamento do modelo relacional como única forma de organização.

Esses bancos são frequentemente adotados em aplicações com elevada taxa de escrita, dados heterogêneos, estruturas variáveis ou necessidade de expansão por adição de servidores.

► Bancos chave-valor e orientados a documentos

Agregados de acesso rápido

Bancos chave-valor armazenam cada conteúdo associado a uma chave única. São adequados para cache, sessões, contadores e configurações. Redis é um exemplo conhecido, oferecendo também estruturas como listas, conjuntos e filas.

Bancos de documentos armazenam registros em estruturas semelhantes a JSON. Um documento pode conter campos aninhados e listas, reduzindo a necessidade de junções. MongoDB é um exemplo amplamente utilizado.

Modelo	Unidade principal	Aplicação típica
Chave-valor	Chave e conteúdo	Cache e sessões
Documentos	Documento estruturado	Catálogos e aplicações web
Colunar	Famílias de colunas	Escrita distribuída
Grafos	Nós e arestas	Relacionamentos complexos
Séries temporais	Eventos com tempo	Sensores e monitoramento

GESTÃO EM TI

KANBAN

INTRODUÇÃO AO KANBAN

► Conceito de Kanban

Significado do termo

Kanban é uma palavra de origem japonesa que pode ser traduzida como “cartão”, “sinal visual” ou “placa de sinalização”. O termo está relacionado ao uso de elementos visuais para representar a existência, a movimentação ou a necessidade de determinado trabalho. Em sua forma mais simples, um cartão Kanban indica que uma atividade precisa ser realizada, está em execução ou já foi concluída.

Com o tempo, o termo passou a designar não apenas o cartão utilizado para sinalização, mas também um método de gestão do fluxo de trabalho. Nesse método, as atividades são representadas visualmente e movimentadas por diferentes etapas, de acordo com o avanço de sua execução.

O Kanban permite que as pessoas observem o estado atual do trabalho sem depender exclusivamente de relatórios extensos ou reuniões constantes. Ao olhar para o quadro, é possível identificar quais atividades ainda aguardam início, quais estão sendo realizadas, quais estão bloqueadas e quais já foram concluídas.

Finalidade do método

A principal finalidade do Kanban é melhorar o fluxo de trabalho. Isso significa organizar a passagem das atividades pelas diferentes etapas do processo, reduzindo atrasos, acúmulos, interrupções e desperdícios.

O método procura tornar o trabalho visível para que problemas possam ser percebidos com maior facilidade. Quando muitas atividades se acumulam em uma mesma etapa, por exemplo, o quadro evidencia a existência de um possível gargalo. Da mesma forma, quando uma tarefa permanece parada por muito tempo, sua posição no quadro permite identificar que existe um bloqueio ou uma dependência não resolvida.

O Kanban também busca evitar que as pessoas iniciem mais atividades do que conseguem concluir. Em muitos ambientes, o excesso de tarefas simultâneas provoca perda de foco, aumento de erros e demora nas entregas. Por essa razão, o método utiliza limites para controlar a quantidade de trabalho em andamento.

A proposta não é manter todos ocupados o tempo inteiro, mas fazer com que as atividades avancem de maneira contínua e equilibrada. O foco está na conclusão do trabalho e não apenas em seu início.

► Origem e evolução

Surgimento no contexto industrial

O Kanban surgiu no ambiente industrial japonês, especialmente associado ao sistema de produção desenvolvido pela Toyota. Nesse contexto, os cartões eram utilizados para sinalizar a necessidade de reposição de materiais e componentes ao longo da linha de produção.

Em vez de produzir grandes quantidades antecipadamente, cada etapa solicitava apenas aquilo que precisava para continuar operando. Quando determinada quantidade de peças era consumida, um cartão indicava a necessidade de reposição. Dessa forma, a produção era orientada pela demanda real, e não apenas por previsões.

Esse funcionamento contribuiu para reduzir estoques excessivos, desperdícios e produção desnecessária. Os cartões atuavam como mecanismos de comunicação entre as etapas do processo, permitindo que materiais fossem produzidos ou movimentados no momento adequado.

A lógica industrial do Kanban está diretamente relacionada ao controle do fluxo. Cada etapa deve receber a quantidade de trabalho compatível com sua capacidade. Quando o sistema recebe mais itens do que consegue processar, formam-se filas, estoques e atrasos.

Expansão para ambientes de serviços e conhecimento

Posteriormente, os princípios do Kanban foram adaptados para atividades que não envolvem necessariamente a fabricação de produtos físicos. O método passou a ser utilizado em serviços administrativos, atendimento ao cliente, comunicação, educação, tecnologia, manutenção, análise de documentos e diversas outras áreas.

Nesses ambientes, os cartões não representam peças ou materiais, mas tarefas, solicitações, projetos, documentos ou demandas. O quadro mostra como esses itens se movimentam desde o momento em que surgem até sua conclusão.

Em um setor de atendimento, por exemplo, um cartão pode representar uma solicitação recebida. Em um departamento jurídico, pode representar um processo em análise. Em uma equipe de comunicação, pode representar a elaboração de uma peça publicitária. Embora as atividades sejam diferentes, a lógica de gestão do fluxo permanece semelhante.

► Características gerais

Gestão visual

A gestão visual é uma das características mais reconhecidas do Kanban. O trabalho é representado em um quadro dividido em etapas, normalmente por meio de colunas. Cada atividade aparece em um cartão, que se desloca conforme sua evolução.

Essa representação permite que o processo seja compreendido de maneira rápida. A posição de cada cartão mostra o estado da atividade, enquanto cores, etiquetas e outros elementos podem fornecer informações adicionais.

Controle do fluxo de trabalho

O Kanban não se limita a exibir tarefas. Ele permite controlar como essas tarefas entram, avançam e saem do sistema. O método procura reduzir variações excessivas, impedir acúmulos e criar um ritmo sustentável de entrega.

O controle ocorre principalmente pela observação do quadro, pela limitação do trabalho em andamento e pelo acompanhamento dos tempos de execução.

Melhoria contínua

O Kanban adota uma abordagem de melhoria gradual. Em vez de exigir uma transformação completa e imediata, o método pode ser aplicado inicialmente ao processo existente. A partir da observação do fluxo, são identificados problemas e oportunidades de ajuste.

As mudanças são realizadas com base em evidências. Se uma etapa acumula atividades com frequência, por exemplo, pode ser necessário revisar sua capacidade, suas regras ou sua forma de funcionamento. O processo evolui conforme a equipe aprende com os resultados obtidos.

FUNDAMENTOS DO KANBAN

► Visualização do trabalho

Representação das etapas do processo

Visualizar o trabalho significa transformar atividades, etapas e regras em informações que possam ser observadas diretamente. No Kanban, essa visualização é realizada principalmente por meio de um quadro.

As colunas do quadro representam os estados pelos quais uma atividade passa. Um fluxo simples pode conter as etapas “A fazer”, “Em andamento” e “Concluído”. Processos mais complexos podem incluir análise, aprovação, execução, revisão, validação e entrega.

A estrutura do quadro deve corresponder ao funcionamento real do processo. Criar colunas que não representam atividades efetivamente realizadas pode gerar confusão. Por outro lado, omitir etapas importantes impede que o quadro mostre onde o trabalho realmente está.

A visualização também ajuda a revelar tarefas que antes permaneciam ocultas. Atividades informais, correções, retrabalhos e solicitações urgentes podem consumir grande parte da capacidade de uma equipe sem aparecer em planejamentos formais. Quando esses itens são representados no quadro, torna-se mais fácil compreender a carga total de trabalho.

Visibilidade sobre tarefas e responsabilidades

Os cartões podem apresentar informações como título da atividade, pessoa responsável, data de entrada, prazo, prioridade e eventuais bloqueios. Essas informações permitem que todos saibam quem está atuando em cada item e qual é sua situação.

A visibilidade reduz a dependência de perguntas individuais. Em vez de solicitar atualizações a cada pessoa, é possível consultar o quadro. Isso favorece a comunicação e diminui a perda de informações.

Entretanto, a transparência exige atualização constante. Um quadro desatualizado transmite uma visão incorreta do processo. Por isso, a movimentação dos cartões deve acompanhar a movimentação real das atividades.

► Sistema puxado

Início de atividades conforme a capacidade disponível

O Kanban utiliza a lógica de sistema puxado. Nesse sistema, uma nova atividade é iniciada quando existe capacidade disponível para executá-la. A etapa seguinte “puxa” o trabalho da etapa anterior conforme consegue recebê-lo.

Suponha que uma equipe tenha capacidade para executar apenas três atividades simultaneamente. Quando as três posições estão ocupadas, uma quarta atividade não deve ser iniciada até que pelo menos uma das anteriores avance ou seja concluída.

Esse mecanismo evita o excesso de tarefas abertas. Em vez de iniciar tudo o que chega, a equipe seleciona novos itens apenas quando possui condições reais de trabalhar neles.

Diferença entre fluxo puxado e fluxo empurrado

No fluxo empurrado, atividades são enviadas para as etapas seguintes independentemente de sua capacidade. Uma área pode encaminhar diversos itens para outra, ainda que essa segunda área já esteja sobrecarregada. Como resultado, o trabalho se acumula em filas.

No fluxo puxado, a transferência ocorre conforme a disponibilidade. A etapa seguinte recebe um novo item quando possui espaço para processá-lo. O objetivo é equilibrar a quantidade de trabalho entre as etapas.

Um sistema puxado não significa ausência de planejamento ou prioridade. As atividades podem ser organizadas de acordo com critérios previamente definidos. A diferença está no momento do início: a demanda só entra em execução quando existe capacidade.

► Limitação do trabalho em andamento

Conceito de WIP

WIP é a sigla de Work in Progress, expressão que corresponde ao trabalho em andamento. O WIP representa a quantidade de itens que foram iniciados, mas ainda não foram concluídos.

No Kanban, podem ser estabelecidos limites máximos de WIP para determinadas colunas ou conjuntos de colunas. Se uma etapa possui limite igual a quatro, ela não pode conter mais de quatro cartões ao mesmo tempo.

Esses limites devem considerar a capacidade real das pessoas e do processo. Um limite muito alto pode permitir excesso de tarefas simultâneas. Um limite muito baixo pode causar ociosidade ou bloquear o fluxo sem necessidade. Por isso, os limites devem ser observados e ajustados ao longo do tempo.