

Material de Estudo – Infraestrutura de Hardware e Software para SI

1. Conceito de Infraestrutura de TI

A **infraestrutura de TI** é o conjunto de recursos **hardware, software, redes e serviços** que sustentam o funcionamento de sistemas de informação em uma organização.

Ela deve equilibrar:

- **Custo (\$\$\$)** – investimento financeiro disponível.
- **Necessidade** – atender às demandas da empresa.

Exemplos:

- Uma **startup pequena** pode usar apenas **notebooks e serviços em nuvem** para reduzir custos.
 - Um **banco** precisa de **data centers robustos, servidores redundantes e alta segurança**.
-

2. Infraestrutura para Organizações

Elementos importantes:

- **Dimensionamento de Data Center:** quantidade de servidores, energia e refrigeração necessárias.
- **Dimensionamento de PCs:** escolher máquinas adequadas para a rotina dos usuários.
- **Redes e telecomunicações:** internet, firewalls, switches e roteadores.
- **Especificação de equipamentos:** definir padrões para evitar incompatibilidades.
- **Compra de softwares:** licenciamento, versões, atualizações.
- **Escolha do sistema operacional:** Windows, Linux, macOS, Android, iOS etc.

Processos de gestão de infraestrutura:

- Resolver problemas do dia a dia rapidamente.
 - Reduzir falhas com **manutenção preventiva**.
 - Monitorar desempenho dos sistemas.
 - Demonstrar resultados em **métricas claras** para gestores.
 - Criar **autonomia nos usuários** (ex.: treinamentos).
-

3. Infraestrutura para Desenvolvimento de Sistemas

3.1 Desenvolvimento Desktop

Essenciais:

- Computador, compilador/interpretador, editor de texto, SGBD.

Quase sempre necessários:

- IDE (ex.: Visual Studio, Eclipse).

Desejáveis:

- APIs e frameworks (ex.: .NET, JavaFX).

➔ **Exemplo prático:** Para desenvolver em **Java Desktop**, utiliza-se:

- **Eclipse IDE**
 - **JDK (compilador Java)**
 - **MySQL (SGBD)**
-

3.2 Desenvolvimento Web

Além dos recursos do desktop, precisa de:

- **Servidor Web** (Apache, Nginx, IIS).
- **Arquitetura cliente-servidor.**
- **Servidor HTTP** que interpreta URLs e responde com páginas web.

Arquitetura em 3 camadas (3-Tier):

1. **Apresentação** – interface (HTML, CSS, JS).
2. **Lógica de Negócio** – aplicação (Java, Python, Node.js, PHP).
3. **Dados** – banco de dados (MySQL, PostgreSQL).

➔ **Exemplo prático:**

Um sistema e-commerce em PHP:

- **Frontend:** HTML/CSS/JS.
 - **Backend:** PHP rodando no Apache.
 - **Banco de Dados:** MySQL.
-

3.3 Desenvolvimento Mobile

Além dos recursos do desktop:

- **Emuladores de smartphones** (Android Studio, Xcode).

- **Smartphones físicos para testes reais.**
- **APIs** para acesso a banco de dados e serviços.
- **Frameworks** para facilitar desenvolvimento.

Frameworks mais usados:

- **React Native (JS)** → híbrido.
- **Flutter (Dart)** → híbrido.
- **Ionic (HTML + JS + CSS)** → híbrido.
- **Xamarin (C#)** → focado em Windows Phone.

➔ Exemplo prático:

Criar um app de delivery:

- Backend em **Node.js**.
 - App em **React Native**.
 - API REST para integrar pedidos e pagamentos.
-

4. Infraestrutura em Cloud

Muitos acham que “nuvem” é só **armazenamento de arquivos**, mas vai muito além.
Há **modelos de serviço**:

SaaS – Software as a Service

- Software pronto para uso.
- Usuário não precisa instalar nem manter.
- Pagamento por assinatura/uso.
➔ **Exemplo:** Google Docs, Microsoft 365, Trello.

PaaS – Platform as a Service

- Plataforma completa para **desenvolvimento e implantação**.
- Não há necessidade de comprar servidores físicos.
➔ **Exemplo:** Heroku, Google App Engine.

IaaS – Infrastructure as a Service

- Infraestrutura (servidores, rede, armazenamento) fornecida pela nuvem.
 - Usuário gerencia o sistema operacional e os aplicativos.
➔ **Exemplo:** Amazon AWS EC2, Microsoft Azure VMs.
-

Resumindo

- **Infraestrutura de TI** sustenta as operações das empresas.
 - Para **desenvolvimento desktop, web e mobile** existem recursos e ferramentas específicas.
 - **Cloud computing (SaaS, PaaS, IaaS)** oferece flexibilidade, redução de custos e escalabilidade.
-

