

# UC1: Analisar requisitos e funcionalidades da aplicação

Docente: Mateus de Paula Pereira

Turma: Téc. desenvolvimento de Sistemas 0041 - 2025/2026

# Unidade 1 — Compreensão de Projetos e Briefing

## Objetivos:

- Apresentar conceitos fundamentais de projetos de software.
- Interpretar e construir briefings.
- Definir objetivos com base em clientes e público-alvo.

# Introdução a Projetos de Software

# O que é um Projeto?

- Esforço **temporário** para desenvolver um **produto, serviço ou resultado exclusivo**.
- Tem **início, meio e fim** definidos.
- Exige **planejamento, execução e controle**.
- Exemplo: Desenvolver um sistema de agendamento de consultas.

# O que é um Projeto de Software?

- Conjunto de **atividades** para criar ou evoluir um sistema.
- Resolve **um problema real** de um cliente ou organização.
- **Características:**
  - Temporário
  - Iterativo e incremental
  - Requisitos bem definidos
  - Envolve design, análise, programação e testes.

# Componentes de um Projeto de Software

- Objetivo claro
- Público-alvo
- **Requisitos (funcionais e não funcionais)**
- Recursos e equipe
- Tecnologia utilizada
- Cronograma de desenvolvimento
- Critérios de entrega e validação

# Tipos de Projetos de Software

<b>Tipo</b>	<b>Características principais</b>
<b>Web</b>	Acesso via navegador
<b>Desktop</b>	Instalável, independente da internet
<b>Mobile</b>	Android/iOS
<b>Híbrido</b>	Multiplataforma (ex.: PWA)
<b>Embarcado</b>	Integrado a hardware, IoT, tempo real

# Ciclo de Vida de um Projeto de Software

- **Levantamento de Requisitos**
- **Análise**
- **Modelagem**
- Desenvolvimento
- Testes
- Entrega e Manutenção



# Projeto x Produto de Software

Projeto	Produto
Processo temporário (início e fim)	Resultado do projeto
Foco em <b>escopo, cronograma e requisitos</b>	Foco em <b>qualidade e evolução contínua</b>
Exemplo: Desenvolvimento de um app	Exemplo: O app sendo utilizado pelos usuários
Pode originar vários produtos	Pode evoluir por meio de novos projetos

# Atividade 01 - AVA

## Objetivos:

- Identificar e discutir exemplos de projetos e produtos de software conhecidos.
- Debate sobre como essas ferramentas surgiram a partir de projetos.

# Briefing e Escopo

# O que é Briefing?

- Documento com informações **iniciais fornecidas pelo cliente.**
- Alinha necessidades, expectativas e objetivos.
- **Base para levantamento de requisitos e escopo.**

# Importância do Briefing



# Importância do Briefing

- Reduz **ruídos de comunicação**.
- Evita **retrabalhos**.
- Direciona **requisitos técnicos** e modelagem.
- **Garante alinhamento** entre cliente e equipe.

# Estrutura do Briefing

- Dados do cliente/empresa
- Problema a ser resolvido
- Objetivo do sistema
- Público-alvo
- Funcionalidades esperadas
- Referências/sistemas semelhantes
- Restrições (tempo/orçamento/tecnologia)
- Critérios de sucesso

# Como obter o Briefing?

- Entrevistas estruturadas ou semiestruturadas
- Formulários de briefing
- Reuniões de levantamento inicial
- Análise de documentos existentes

## **Importante:**

O briefing **não substitui** o levantamento completo de requisitos.



# Atividade 02 - AVA

- Desenvolver um formulário que possa ser utilizado na coleta inicial de informações com um cliente que deseja obter um sistema de software.

Cliente, Usuários e Necessidades

# Cliente x Usuário: Funções Diferentes

Cliente	Usuário
Contrata/financia o projeto	Utiliza o sistema no dia a dia
Foco em resultados, prazos, ROI	Foco em usabilidade, praticidade
Define escopo e recursos	Aponta dores e necessidades

Exemplo:

Cliente (Clínica médica) x Usuários (Recepcionistas, médicos, pacientes)

# Tipos de Necessidades

Tipo	Definição	Exemplo
<b>Explícitas</b>	Declaradas pelo cliente/usuário	“Quero login com CPF e senha.”
<b>Implícitas</b>	Inferidas pelo contexto	Segurança de dados, backup
<b>Latentes</b>	Não percebidas, mas essenciais	Acessibilidade, responsividade
<b>Emergentes</b>	Surgem com o uso	Integração com WhatsApp

**Reflexão:** Quais necessidades você já percebeu ao usar sistemas no seu dia a dia?

## **Atividade para reflexão:**

Em relação à sua experiência com a plataforma Rede de Carreiras, quais funcionalidades você sentiu falta?

Listar e classificar essas necessidades

# Levantamento de Necessidades (Boas Práticas)

- Entrevistas e observação direta
- Questionários abertos e fechados
- Análise de processos atuais
- Benchmarking do mercado
- Validação constante com cliente

# Erros Comuns

- Pressupor que o cliente sabe tudo o que precisa
- Ignorar o ponto de vista do usuário
- Não validar interpretações com o cliente
- Traduzir desejos diretamente em requisitos sem análise

**Dica: Escutar mais e confirmar entendimentos ajuda a evitar ruídos.**



# Necessidade → Requisito

- Necessidade: Ex.: “Evitar esquecimentos de consultas.”
- Requisito: Ex.: “O sistema deve enviar notificações para o paciente com 24 horas de antecedência.”
- A necessidade é o **“o que o cliente / usuário precisa”**.
- O requisito é o **“como o sistema deve fazer”** (funcionamento técnico).

# Atividade 03 - AVA

Estudo de caso:

- Identificar cliente e usuários
- Classificar necessidades (explícitas, implícitas, latentes)
- Propor requisitos com base nas necessidades levantadas

# Requisitos e Funcionalidades da Aplicação

# O que são requisitos?

Requisitos são descrições claras do que o sistema deve fazer.

Eles **guiam o processo de desenvolvimento**.

Existem dois tipos de requisitos:

- **Funcionais**
- **Não funcionais**

# Requisitos Funcionais

**Definição:** descrevem comportamentos do sistema sob determinadas condições.

Exemplos:

- O usuário pode fazer login com e-mail e senha.
- O sistema deve exibir relatórios mensais de vendas.
- O cliente pode adicionar produtos ao carrinho

# Requisitos Não Funcionais

**Definição:** definem restrições de desempenho, segurança, usabilidade, entre outros atributos de qualidade.

Exemplos:

- O sistema deve estar disponível 99% do tempo.
- A interface deve ser acessível para daltônicos.
- As respostas do sistema devem ocorrer em até 2 segundos.

# Diferenças entre Requisitos Funcionais e Não Funcionais

Aspecto	Requisitos Funcionais	Requisitos Não Funcionais
O que descrevem	Comportamento do sistema	Qualidades e restrições do sistema
Exemplos	Login, cadastro, relatórios	Tempo de resposta, segurança
Avaliação	Testes funcionais	Testes de desempenho e qualidade

# Como identificar requisitos em um briefing

- Identificar **quem são os usuários** e **quais suas necessidades**.
- Traduzir as necessidades (**o que o cliente / usuário precisa**) em funcionalidades (**o que o sistema deve fazer**).
- Investigar quais restrições se aplicam (ex: segurança, tempo, acessibilidade).



# Exemplo de Requisitos

## **Sistema de Agendamento de Consultas Médicas:**

Sistema web que permite a pacientes agendarem, visualizarem e cancelarem consultas médicas online. Possui cadastro de pacientes, login de administradores e listagem de consultas diárias. É acessível por dispositivos móveis, garante segurança dos dados e realiza backups automáticos.

# Requisitos Funcionais (RF)

Código	Requisito Funcional	Descrição
RF01	Cadastro de pacientes	O sistema deve permitir o cadastro de pacientes com nome, CPF, telefone, e-mail e data de nascimento.
RF02	Agendamento de consulta	O sistema deve permitir que o paciente agende uma consulta selecionando o médico, data e horário disponíveis.
RF03	Cancelamento de consulta	O sistema deve permitir que o paciente cancele uma consulta com pelo menos 24h de antecedência.
RF04	Login de administradores	O sistema deve permitir que os administradores façam login para acessar funções de controle.
RF05	Listagem de consultas	O sistema deve exibir uma lista de todas as consultas agendadas para o dia.

# Requisitos Não Funcionais (RNF)

Código	Requisito Não Funcional	Descrição
RNF01	Compatibilidade com dispositivos móveis	O sistema deve ser responsivo, funcionando em celulares, tablets e desktops.
RNF02	Tempo de resposta	O sistema deve responder às solicitações do usuário em até 2 segundos.
RNF03	Segurança dos dados	Todos os dados sensíveis (como CPF e e-mail) devem ser armazenados de forma segura e criptografada.
RNF04	Disponibilidade	O sistema deve estar disponível para acesso 24 horas por dia, 7 dias por semana.
RNF05	Backup automático	O sistema deve realizar backup automático dos dados uma vez por dia.

# Boas práticas para escrever requisitos

- Clareza e objetividade
- Evitar ambiguidade (ex: "sistema rápido")
- Utilizar linguagem compreensível
- Priorizar requisitos essenciais
- Identificar requisitos **obrigatórios** e **desejáveis**

# Atividade 04 - AVA

- Com base nos estudos de caso da atividade 04 do AVA, faça:
  - Identifique cliente e usuários
  - Classifique necessidades (explícitas, implícitas, latentes)
  - Identifique os requisitos (funcionais e não funcionais)

# Conectando Briefing, Necessidades e Requisitos (Integração)

# Recapitulando

- Projeto x Produto
- Briefing: estrutura e importância
- Cliente x Usuário
- Necessidades x Requisitos

**Como esses elementos se conectam?**

# Fluxo de Desenvolvimento Inicial

- **Briefing:** Alinhamento com cliente
- **Identificação de necessidades:** Cliente e usuários
- **Transformação de necessidades em requisitos**
  - Requisitos funcionais
  - Requisitos não funcionais
- Modelagem do projeto
- Prototipagem



# Encerramento da Unidade 1

- Conteúdos aprendidos:
  - Importância do briefing na análise de requisitos
  - Diferença entre cliente e usuário
  - Identificação de necessidades e transformação em requisitos
- Preparação para a próxima unidade: **Modelagem de Aplicações.**