# 1. ANÁLISE DE REQUISITOS

**Requisitos são solicitações, desejos, necessidades**. Um requisito é a propriedade que um software exibe para solucionar problemas reais. Quando se trata de um [software sob demanda](https://codificar.com.br/servicos/desenvolvimento-de-software/), por exemplo, **um requisito é uma maneira** pelo qual o sistema oferecido deve fazer, ou **um condicionamento no desenvolvimento** **do sistema**.

Ele um dos principais responsáveis pelo sucesso dos softwares, os requisitos, são a base para modelagem, projeto, execução, testes e até mesmo para a manutenção dos mesmos.  **Assim, os requisitos estão presentes ao longo de todo o ciclo de vida de um software.**  Ao começar um projeto, os requisitos já devem ser levantados, entendidos e documentados.

Existem diversas formas para se extrair os requisitos, tais formas serão vistas na próxima aula.

**Tipos de requisitos:**

# Requisitos Funcionais

Materialização de uma necessidade ou solicitação realizada por um software. Porém, vários Requisitos Funcionais podem ser realizados dentro de uma mesma funcionalidade. São variadas as funções e serviços que um sistema pode fornecer ao seu cliente, descrevemos abaixo algumas das inúmeras funções que os softwares podem executar:

# Exemplos

Observação: CRUD (Create, Retrieve, Update e Delete) quatro operações básicas, utilizadas em banco de dados relacionais, que permite que o usuário possa criar, consultar, atualizar e deletar dados.

**RF1.- Cadastro de Fornecedores**

CRUD de Fornecedores.

**RF2. Consulta a Fornecedores**

Permitir consultar Fornecedores.

**RF3. Emitir Relatório de Fornecedores**

Emitir relatório de Fornecedores.

**RF4. Cadastro de Clientes**

CRUD de Clientes.

**RF5. Consulta de Clientes**

Permitir consultar Clientes.

**RF6. Emitir Relatório de Clientes**

Emitir relatórios de Clientes.

**RF7. Cadastro de Contas a Pagar**

CRUD de Contas a Pagar.

**RF8. Consulta de Contas a Pagar**

Permitir consultar Contas a Pagar.

**RF9. Emitir Relatório de Conta a Pagar**

Emitir relatórios de Contas a Pagar.

**RF10. Cadastro de Contas a Receber**

CRUD de Contas a Receber.

**RF11. Consulta de Contas a Receber**

Permitir consultar Contas a Receber.

**RF12. Emitir Relatório de Conta a Receber**

Emitir Relatório de Contas a Receber.

**RF13 Baixa de Contas**

Permite ao Usuário encerrar a pendência de uma Conta no sistema.

* 1. **Requisitos Não-Funcionais**

Uma vez que os Requisitos Funcionais definem o que o sistema fará, a Engenharia de Software afirma que os Requisitos Não Funcionais são aqueles que não interferem diretamente no desenvolvimento do sistema. Os Requisitos não Funcionais não estão relacionados diretamente às funcionalidades de um sistema, mas sim a possíveis melhorias e a um “plus” nas suas funcionalidades.

**Exemplos**

**RNF1. Manter histórico das operações armazenado no sistema**

Registrar as operações realizadas no sistema em um arquivo armazenado no sistema

**RNF2. O sistema pode ser acessível aos usuários pela Internet usando HTTPS**

O sistema online deve utilizar o protocolo HTTPS, pois permite transmissão de dados criptografados.

**RNF3. O sistema dever permitir portabilidade para vários browsers**

O sistema online deve ser compatível nos seguintes browsers:

Internet Explorer 8, Mozilla FireFox 7 e Google Chrome 14.0

**RNF4. Controle de acessos por perfis**

Manter cadastro de perfis de usuários para o sistema e definir as permissões para cada tipo de perfil

**RNF5 . Documento atualizado da implantação do sistema**

Documentação atualizada para auxiliar na implantação do sistema

**RNF6. Mensagens claras de tratamento de erros**

As mensagens de erros devem ser claras para facilitar a compreensão do usuário

**RNF7. Banco de dados MySQL 5.5.9**

Banco de dados MySQL, pois permite fácil instalação.

**RNF8. Programação em Java**

O sistema deve ser programado exclusivamente em Java, pois é uma linguagem orientada ao objeto.