

# Etapas do processo de extração de requisitos

**Entendimento do Domínio:** os programadores devem estudar, o melhor possível, a empresa e as atividades do cliente, a fim de poder entender melhor o escopo de suas solicitações.

**Extração e Análise de Requisitos:** através de interação com os usuários, os programadores são informados dos requisitos do sistema. Essas informações são classificadas e organizadas de acordo com suas prioridades, são determinadas inconsistências, conflitos e omissões

**Especificação dos Requisitos:** os requisitos são armazenados de uma ou mais formas, incluindo linguagem natural, linguagem semi-formal ou formal, representações simbólicas e gráficas.

**Validação dos Requisitos:** os desenvolvedores analisam os requisitos coletados para confirmar se eles estão completos e de acordo com as necessidades e solicitações dos usuários.

# PIECES - técnica de para extração de requisitos



**P - Desempenho (ou performance):** o desempenho de um sistema pode ser mensurado pela quantidade de tarefas executadas em um intervalo de tempo (throughput) e pelo tempo de resposta para a execução de uma tarefa. As perguntas sobre desempenho ajudam a identificar as tarefas a serem executadas pelo software, e assim pode-se calcular o throughput ou o tempo de resposta. *Exemplo:* Quantos atendimentos são efetuados por dia? Quantos usuários utilizam o sistema por turno? Quantos pedidos são efetuados em uma hora?

**I - Informação e Dados:** as informações a serem acessadas pelos usuários devem ser exatamente as que eles necessitam, em quantidades adequadas, nem informações de mais nem falta de informações. Elas também devem ser disponibilizadas de forma adequada ao usuário, evitando que a consulta a elas seja cansativa, frustrante ou difícil. *Exemplo:* A relação de funcionários da empresa é fornecida em que intervalo de tempo? Todos os dados de cada item do pedido devem constar da nota fiscal?

**E - Economia:** o nível de serviço e a capacidade de lidar com alta demanda são dois aspectos que influenciam no custo do desenvolvimento do software. O nível de serviço é medido pelo throughput ou pelo tempo de resposta, ou pelos dois. A capacidade de lidar com alta demanda pode ser necessária de tempos em tempos, necessitando de um tempo de resposta tão eficaz quanto se a demanda de dados fosse menor. Mas isso pode significar disponibilizar um sistema com capacidade maior do que o necessário na maior parte do tempo, encarecendo o projeto. Isso deve ser discutido com o usuário para que o desenvolvedor possa tomar decisões referentes ao nível de serviço e à capacidade de lidar com alta demanda sem assim indispor o usuário em relação ao software. *Exemplo:* Você recebe um maior número de pedidos em determinada época do mês, ou do ano? Existe um horário em que o registro de reclamações aumenta?

# PIECES - técnica de para extração de requisitos



**C - Controle:** o software é desenvolvido para funcionar de maneira prevista. Um desvio de qualquer função pode acarretar perda de controle da situação pelo sistema, e alguma providência deve ser tomada pelo próprio software. Outra forma de controle é a segurança, que pode impedir o acesso de certos usuários ou a certas informações, ou certos tipos de acesso podem ser permitidos ou não. O controle do sistema deve ser discutido com cuidado, pois uma extração de requisitos mal feita pode ocasionar em excesso de controle, impedindo o trabalho, ou controle de menos, prejudicando o sistema ou a base de dados. *Exemplo:* Todos os usuários têm acesso irrestrito à base de dados do sistema? Aos usuários deverá ser requisitada autorização de acesso?

**E - Eficiência:** é diferente de economia, pois economia é a redução da quantidade total de recursos utilizados, e eficiência é a redução da perda de recursos. Os recursos podem ser alocados ao sistema em quantidade maior do que a necessária, mas não podem ser recuperados. O que se gasta de recursos deve ser reduzido, e essa capacidade de redução de perda é a eficiência. Aspectos como entradas em duplicidade, processos sendo executados diversas vezes desnecessariamente, ou armazenar o mesmo dados em diversos espaços é perda de recursos desnecessária. Isso pode resultar de um algoritmo ou uma interface pobres. *Exemplo:* Todas as funções se utilizarão deste dado? Você vai entrar com esse valor uma única vez?

**S - Serviços:** um software é desenvolvido para prestar serviços ao usuário ou a outro software. Definir quais são esses serviços e de que forma eles devem ser disponibilizados ajudam a extração de requisitos a ser mais eficaz. *Exemplo:* O software emite ordem de serviço ? O software controla o estoque em tempo real ?