

Analise e Modelagem de Sistemas

# **Elicitação, especificação e validação de requisitos**

Fonte: WERLICH, C. *Análise e Modelagem de Sistemas*.  
Londrina: EDE SA, 2020. ISBN 978-85-522-1683-4.

- Requisitos
  - Descrições formais das necessidades, funcionalidades e restrições que um sistema de software deve atender
  - Lista de tudo que os stakeholders precisam e desejam
- A partir dos problemas e das necessidades dos stakeholders, a elicitação de requisitos visa
  - Especificar o domínio do problema do sistema
  - Verificar as possibilidades de reutilização de alguma solução já realizada
  - Identificar os stakeholders diretamente envolvidos pelo sistema
  - Elicitar e qualificar os requisitos do sistema
  - Análise dos requisitos elicitados
  - Validação dos requisitos elicitados

- Procedimentos básicos de levantamento e análise de requisitos de um sistema
  - Concepção ou compreensão do domínio
    - Compreensão exata do que será realizado e os limites
  - Coleta de requisitos e classificação dos requisitos (elicitação)
    - Todas as atividades
  - Negociação dos requisitos
    - O que está correto, o que é conflito, etc.
  - Validação dos requisitos
    - “termo de aceite” em que todas as partes envolvidas concordam e validam os requisitos
  - Documentação
    - Especificações dos requisitos funcionais e requisitos não funcionais
    - Meio de comunicação entre analista/engenheiro e o cliente

# Solução de conflito

- Chamar as partes envolvidas para resolver o conflito
  - Verificação de requisitos
    - Todos os requisitos são analisados e verificados se estão de acordo com as exigências dos stakeholders
  - Definição das prioridades
    - Determinação dos requisitos mais importantes e que merecem total atenção para o sucesso do projeto

# Levantamento e análise de requisitos



- A negociação serve para estabelecer os ajustes necessários e ajudar a determinar o que será prioritário no desenvolvimento
- Na validação é realizada uma checagem geral de todos os requisitos, tendo como objetivo a documentação
- É um processo de validação continuada

# Atividades do processo de descoberta de requisitos

6/17

- Descoberta de requisitos
  - Com os stakeholders, é um processo interativo, e os usuários finais do sistema devem entrar com sua expertise para ajudar na coleta dos requisitos do sistema
- Classificação e organização de requisitos
  - Agrupamento - repetem ou se fazem parte de subsistemas
- Priorização e negociação de requisitos
  - Com os stakeholders é estabelecida a priorização de cada requisito (essencial, importante ou desejável)
- Especificação de requisitos
  - Documentação dos requisitos, formal ou informal

# Técnicas

- Pesquisa
  - Como funciona a rotina dos processos do sistema e de outros softwares utilizados, procurando requisitos ocultos
- Entrevista
  - Guiada por questionário: necessidades que o sistema deverá suprir
  - Saber ouvir e marcar o máximo de informações obtidas
- Reuniões
  - Brainstorming: requisitos ocultos e resolver conflitos
- Documentos
  - Coleta: relatórios, planilhas, papéis, etc.
- Etnografia
  - Observação e análise de como os usuários finais realmente trabalham (diferente do que venham a dizer)

# Especificação de requisitos

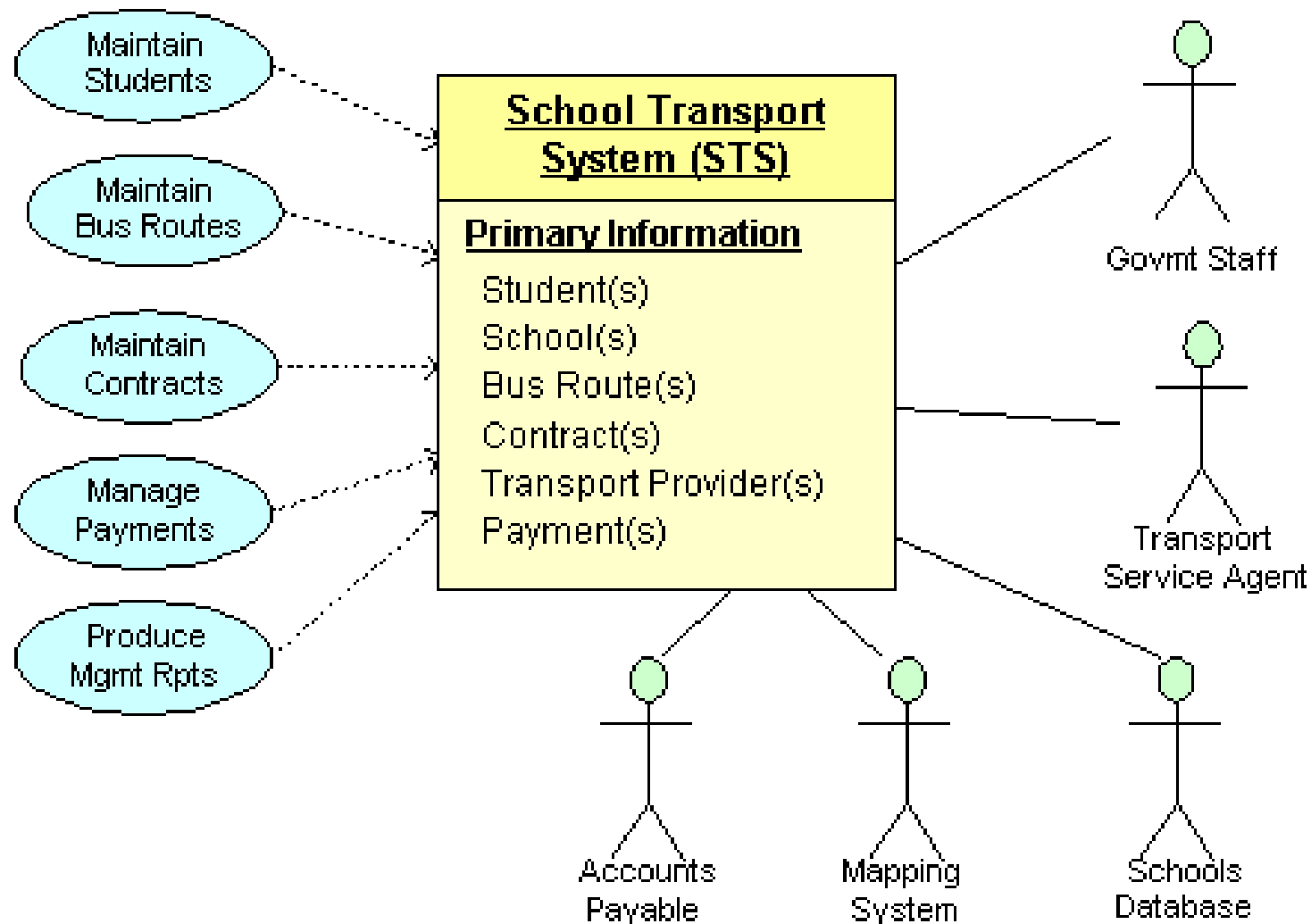
- Sistematização e abstração do que o software deverá realizar a partir de descrições claras e objetivas
- Caracteriza o problema a ser solucionado e os resultados
- Geração de um documento de especificação de requisitos
- A melhor forma de começar a especificação é usando a forma hierárquica, começando dos atributos gerais do sistema, passando por subníveis até chegar aos atributos mais específicos



# Especificação de requisitos

- Meio de comunicação entre o analista de sistemas e os programadores que desenvolverão o software
- Não pode haver duplicidade de interpretações, pois o programador utilizará a especificação gerada para programar exatamente o que estará na especificação
- Descreve todas as funcionalidades e suas restrições dos requisitos funcionais e dos requisitos não funcionais (geralmente em tabelas ou documentos específicos)
- Pode utilizar diagramas de caso de uso para ajudar na comunicação
- Pode fazer uso da prototipagem

- UML (Unified Modeling Language)
  - Ferramenta de comunicação nas atividades do desenvolvimento



# Prototipagem

- Criação de uma versão menor do sistema a ser desenvolvido
- Verificação de custo-benefício, em que a experiência do usuário é uma parte fundamental do desenvolvimento do protótipo
- Descoberta de problemas com antecedência
- Verificar se os requisitos do software satisfazem às necessidades dos clientes
- Melhora a comunicação com o cliente ao apresentar o progresso do software
- Testes de aceitação e opinião do cliente
- Prevê vários problemas e a medição do tempo para resolvê-los

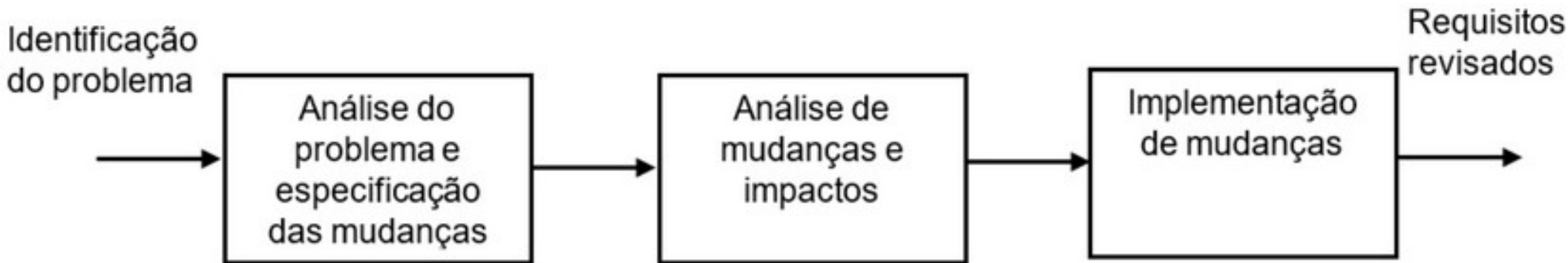
# Prototipagem - abordagens

- Protótipo descartável
  - Aprender sobre o problema a ser resolvido
  - Explorar mais sobre a viabilidade da solução
- Protótipo evolutivo
  - Elimina dúvidas sobre alguma parte do software (geralmente a interface gráfica)
  - O protótipo depois é utilizado como parte do software final
- Prototipagem rápida
  - Partes do software para verificar a viabilidade dos requisitos
  - Verifica se a realização do projeto é viável

# Negociação e Monitoramento

- Negociação de requisitos
  - Desenvolver um plano de projeto que atenda às demandas dos envolvidos
  - Analisar as restrições de orçamento, pessoal, tecnologia, tempo, impostas à equipe de desenvolvimento do software
- Monitoramento de requisitos
  - Garante que o escopo do software desenvolvido seja realizado
  - Deve garantir a rastreabilidade das alterações
    - Uso de ferramentas
    - Dependências (quais requisitos dependem do requisitos em questão), o status do requisito, quem alterou o requisito, quem aprovou o requisito, as datas

# Gerenciamento de mudança de requisitos



- Evitar duplicidade
- Rastreabilidade
- Avaliar os impactos
- Viabilidade técnico-financeira
- Não alterar sem planejar e documentar

# Validação de requisitos

- Validade: se contemplam os objetivos do projeto
- Consistência: um requisito não entre em conflito com outro
- Completude: garantir as funcionalidades esperadas pelos usuários do sistema
- Realismo: ter certeza que a tecnologia utilizada atenda as demandas do sistema projetado
- Ambiguidade: não pode ter mais de uma interpretação
- Verificável: deve ser passível de verificação através de testes
- Rastreabilidade: origem clara e bem definida

# Algumas perguntas ao validar

- Os requisitos estão de acordo com objetivos globais do sistema?
  - O requisito foi bem especificado, fornecendo detalhes apropriados de forma clara e concisa?
  - O requisito é realmente necessário? Está com a sua classificação correta?
  - O requisito não está em conflito com outro requisito?
  - O modelo de requisito reflete adequadamente a informação, comportamento e funcionalidade do sistema?
- 
- As perguntas podem ser mais ou menos elaborado e aprofundado, conforme o sistema a ser desenvolvido
  - Pode necessária mais investigação e mais perguntas para determinar se os requisitos estão ou não corretos
  - Um erro causa retrabalho (custo)



# Validação

- Ferramenta “*checklist*”
  - Auxilia na garantia da qualidade da validação
  - É uma lista preparada de perguntas ou itens de verificação
  - Análise de cada requisito do sistema
  - Descoberta de erros em vários níveis: função, lógica, implementação
  - Verifica se o sistema possui os requisitos especificados
  - Garante que o software desenvolvido foi implementado de acordo com padrões previamente impostos