



# Engenharia de Software

**Prof. Alex Mateus Porn, Me.**

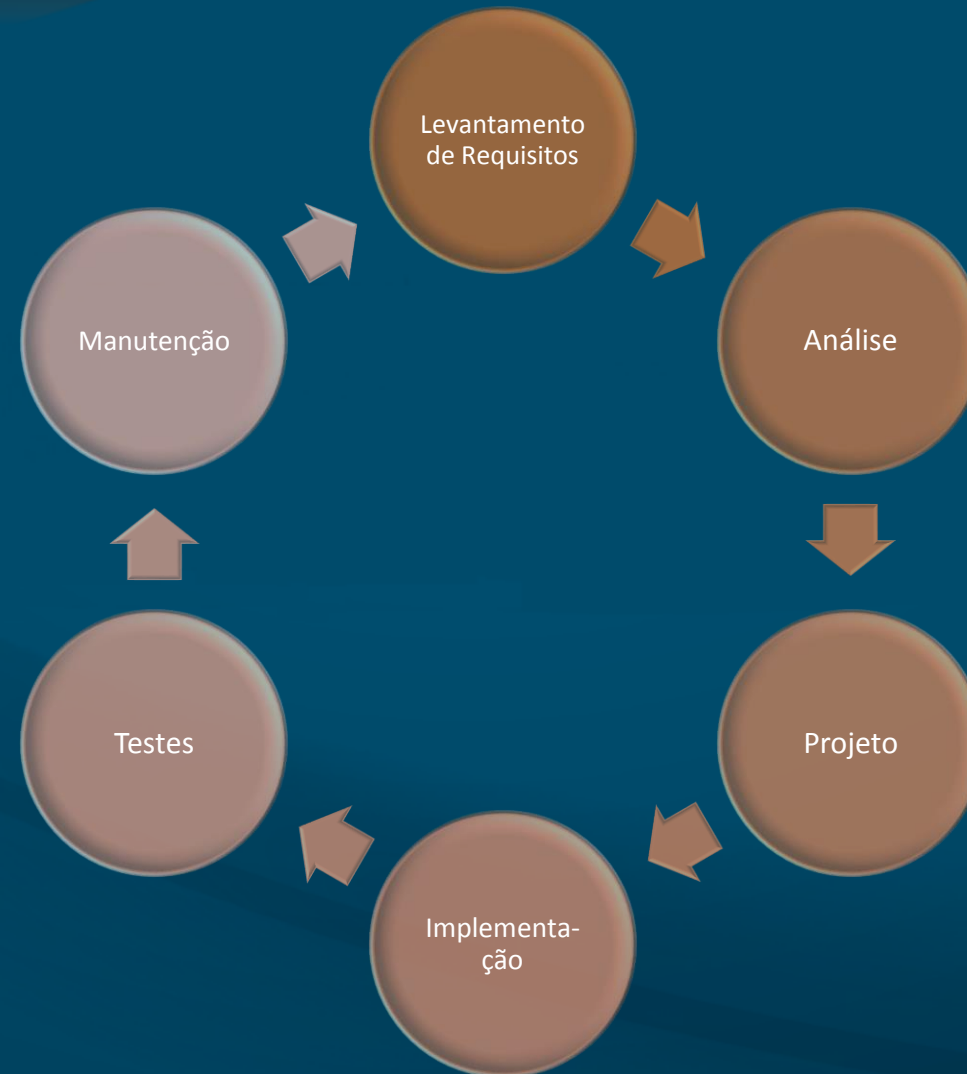
# Organização da Aula

- Referente ao conteúdo abordado nas Aulas 1 e 2 de Engenharia de Software, abordaremos assuntos complementares aos seguintes conteúdos:
  - Ciclo de Vida de Sistemas;
  - Processos de Software;
  - Análise de um estudo de caso.

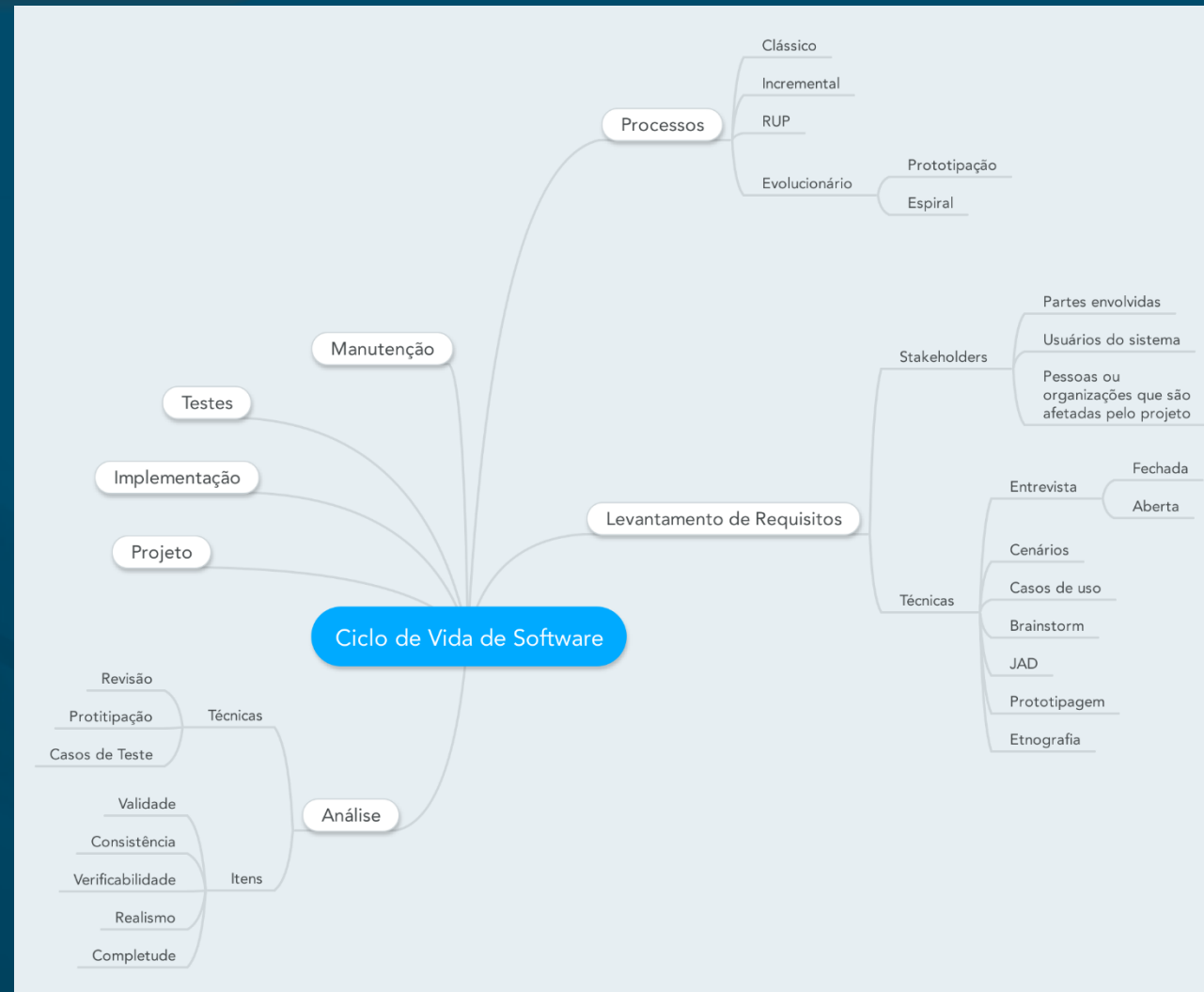


# Ciclo de Vida de Sistemas

# Ciclo de Vida de Sistemas



# Mapa Mental do Ciclo de Vida de Sistemas





# Ciclo de Vida de Sistemas

## 1 - Levantamento de requisitos



# Levantamento de Requisitos

- É a etapa de reunir informações sobre o sistema requerido e os sistemas existentes e separar dessas informações os requisitos de usuário e de sistema.



# Levantamento de Requisitos

- Fontes de informação de requisitos incluem documentação, *stakeholders* e especificações de sistemas similares.
- ➔ Os *stakeholders* variam desde os usuários finais, passando pelos gerentes do sistema até *stakeholders* externos, como reguladores, que certificam a aceitabilidade do sistema.



# Levantamento de Requisitos

- Possíveis técnicas a serem utilizadas:
  - Entrevistas;
  - Cenários;
  - Casos de Uso;
  - Etnografia;
  - Brainstorming;
  - JAD;
  - Prototipagem.

# Levantamento de Requisitos

## ■ Técnicas:

- ➡ Entrevistas: Questionamentos com os *stakeholders* sobre o sistema atual e o sistema que será desenvolvido.
  - Entrevistas fechadas: Conjunto de questões predefinidas.
  - Entrevistas abertas: Série de questões abertas para compreensão das necessidades.

# Levantamento de Requisitos

## ■ Técnicas:

- Cenários: Descrições de sessões de interação do sistema:
  - Descrição do que o sistema e os usuários esperam quando o cenário se iniciar;
  - Descrição do fluxo normal e alternativo de eventos;
  - Descrição do estado do sistema quando o cenário acaba.

# Exemplo prático de cenário

## Quadro 4.4 Cenário para a coleta do histórico médico em MHC-PMS.

### Suposição inicial:

O paciente é atendido em uma clínica médica por uma recepcionista; ela gera um registro no sistema e coleta suas informações pessoais (nome, endereço, idade etc.). Uma enfermeira é conectada ao sistema e coleta o histórico médico do paciente.

### Normal:

A enfermeira busca o paciente pelo sobrenome. Se houver mais de um paciente com o mesmo sobrenome, o nome e a data de nascimento são usados para identificar o paciente.

A enfermeira escolhe a opção do menu para adicionar o histórico médico.

A enfermeira segue, então, uma série de *prompts* do sistema para inserir informações sobre consultas em outros locais, os problemas de saúde mental (entrada de texto livre), condições médicas (enfermeira seleciona condições do menu), medicação atual (selecionado no menu), alergias (texto livre) e informações da vida doméstica (formulário).

### O que pode dar errado:

O prontuário do paciente não existe ou não pôde ser encontrado. A enfermeira deve criar um novo registro e registrar as informações pessoais.

As condições do paciente ou a medicação em uso não estão inscritas no menu. A enfermeira deve escolher a opção 'outros' e inserir texto livre com descrição da condição/medicação.

O paciente não pode/não fornecerá informações sobre seu histórico médico. A enfermeira deve inserir um texto livre registrando a incapacidade/relutância do paciente em fornecer as informações. O sistema deve imprimir o formulário-padrão de exclusão afirmando que a falta de informação pode significar que o tratamento será limitado ou postergado. Este deverá ser assinado e entregue ao paciente.

### Outras atividades:

Enquanto a informação está sendo inserida, o registro pode ser consultado, mas não editado por outros agentes.

### Estado do sistema na conclusão:

O usuário está conectado. O prontuário do paciente, incluindo seu histórico médico, é inserido no banco de dados e um registro é adicionado ao *log* do sistema, mostrando o tempo de início e fim da sessão e a enfermeira envolvida.

# Levantamento de Requisitos

## ■ Técnicas:

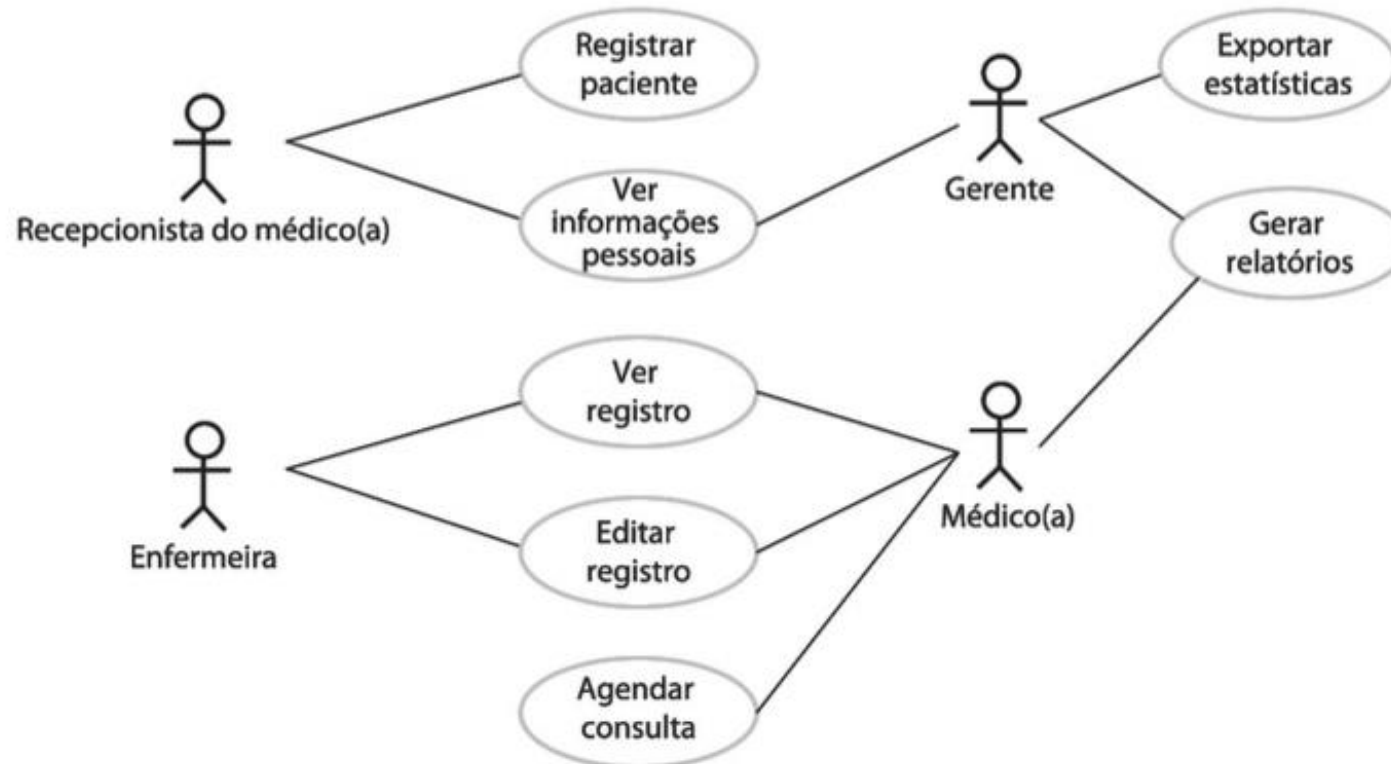
### ➤ Casos de uso:

- Identifica os atores envolvidos;
- Nomeia o tipo de interação;
- As informações podem ser descrições textuais ou gráficos, como os diagramas UML que serão abordados nas próximas aulas.

# Exemplo de caso de uso

**Figura 4.6**

Casos de uso para o MHC-PMS.





# Levantamento de Requisitos

## ■ Técnicas:

### ➤ Etnografia:

- Técnica de observação in loco;
- Usada para compreender os processos operacionais;
- Ajuda a extrair os requisitos de apoio para os processos;

# Levantamento de Requisitos

- Técnicas:

- Brainstorming: Reuniões onde os participantes sugerem ou exploram ideias;



# Levantamento de Requisitos

## ■ Técnicas:

### ➡ JAD (Joint Application Development):

- Usada para promover a cooperação, entendimento e trabalho em grupo entre os usuários e desenvolvedores;
- Utiliza dinâmicas de grupos, utilização de documentos padrão, entre outros.

# Levantamento de Requisitos

## ■ Técnicas:

- Prototipagem: Implementa de forma rápida um pequeno subconjunto de funcionalidades do produto, com o objetivo de explorar aspectos críticos.





# Ciclo de Vida de Sistemas

## 2 - Análise

# Análise

- Etapa pela qual se verifica se os requisitos definem o sistema que o cliente realmente quer, onde se sobrepõe a atividade de validação dos requisitos.





- Diferentes tipos de verificações devem ser executadas com os requisitos encontrados:
  - Verificações de validade;
  - Verificações de consistência;
  - Verificações de completude;
  - Verificações de realismo;
  - Verificabilidade.

- Tipos de verificações:
  - Validade: Maior reflexão e análise mais aprofundada nos requisitos dos usuários podem revelar que alguns requisitos podem ser desnecessários ou até mesmo a falta de outros requisitos.

- Tipos de verificações:

- ▶ Consistência: Requisitos não devem estar em conflito, não devem haver restrições contraditórias ou descrições diferentes do mesmo sistema.

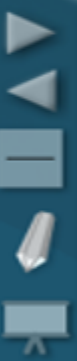
- Tipos de verificações:
  - Completude: Os requisitos devem definir todas as funções e as restrições pretendidas pelos usuários.



# Análise

- Tipos de verificações:

- Realismo: Os requisitos devem ser analisados de modo a garantir que realmente podem ser implementados, com relação às tecnologias existentes.



- Tipos de verificações:
  - Verificabilidade: Abrange a possibilidade de gerar um conjunto de testes que demonstrem que o sistema atende a cada requisito especificado.





- Para a validação dos requisitos algumas técnicas também podem ser utilizadas, tais como:
  - Revisão dos requisitos;
  - Prototipação;
  - Geração de casos de teste.

- Técnicas para validação de requisitos:
  - Revisão dos requisitos: Uma equipe de revisores verifica sistematicamente os requisitos em busca de erros e inconsistências.



- Técnicas para validação de requisitos:
  - Prototipação: Assim como usado no levantamento dos requisitos, um modelo executável é demonstrado para os usuário finais e clientes.



- Técnicas para validação de requisitos:
  - Geração de casos de teste: Os requisitos devem ser testáveis, se for difícil desenvolver testes para os requisitos, provavelmente eles serão difíceis de serem implementados.



# Ciclo de Vida de Sistemas

## 3 - Projeto

# Projeto

- Nesta fase é abordada praticamente toda a estrutura a ser utilizada para a implementação do sistema:
  - Tecnologias de hardware;
  - Rede de computadores;
  - Nível de processamento;
  - Bases de dados;
  - Linguagens de programação, etc.





# Ciclo de Vida de Sistemas

## 4 - Implementação

# Implementação

- Com base nos requisitos levantados e analisados, assim como em toda a estrutura necessária definida, esta fase do ciclo de vida aborda justamente a codificação do sistema como um todo.





# Ciclo de Vida de Sistemas

## 5 - Testes

# Testes

- Diversas técnicas de teste podem ser utilizadas, tais como as técnicas de caixa branca (onde considera-se analisar o código fonte do programa em teste), e as técnicas de caixa preta (onde consideram-se testar as funcionalidades do sistema sem avaliar o seu código fonte ou como foi desenvolvido).



# Ciclo de Vida de Sistemas

## 6 - Manutenção

# Manutenção

- Analisando o resultado de todas as fases anteriores, assim como defeitos encontrados na fase de teste, correções e melhorias devem ser implementadas no sistema.



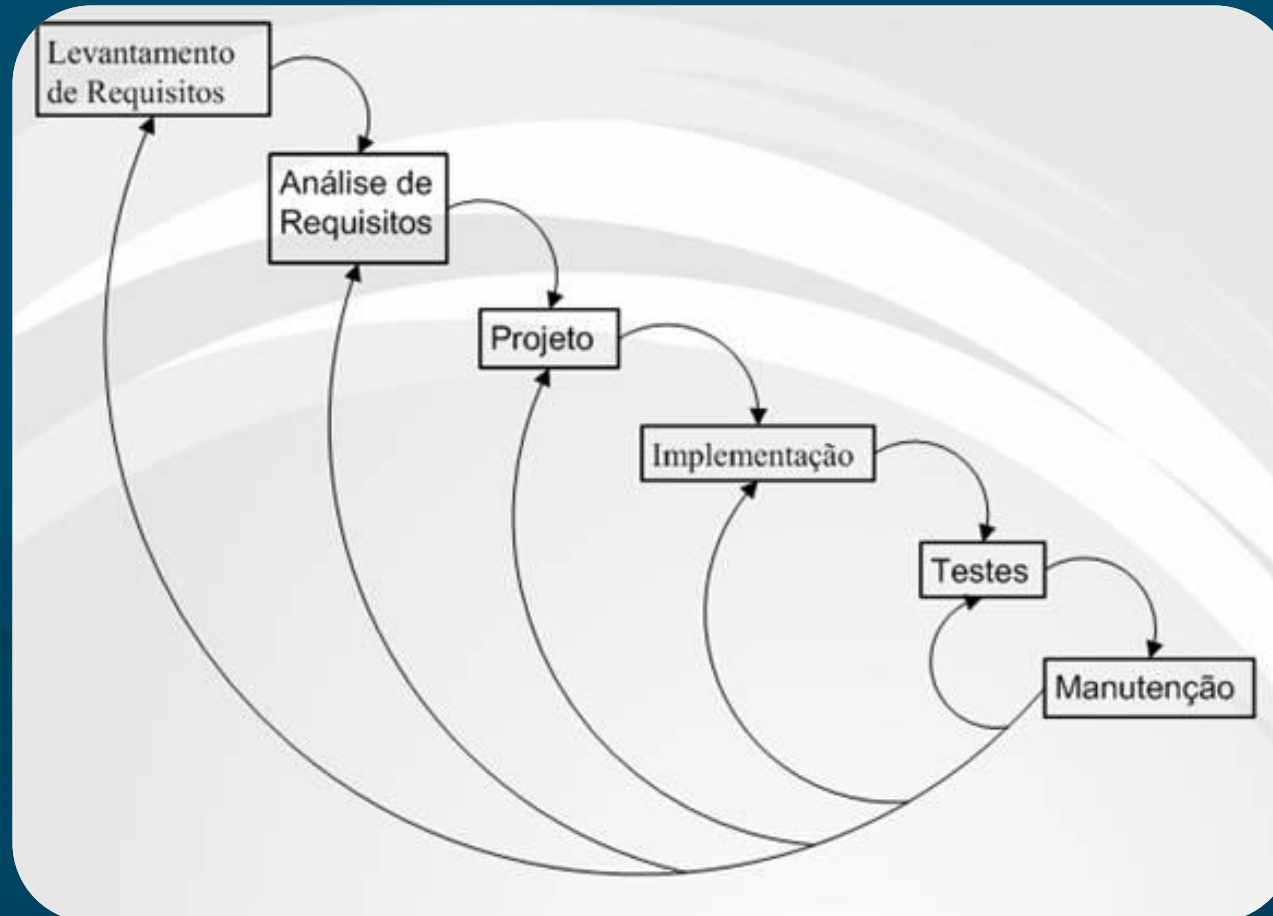




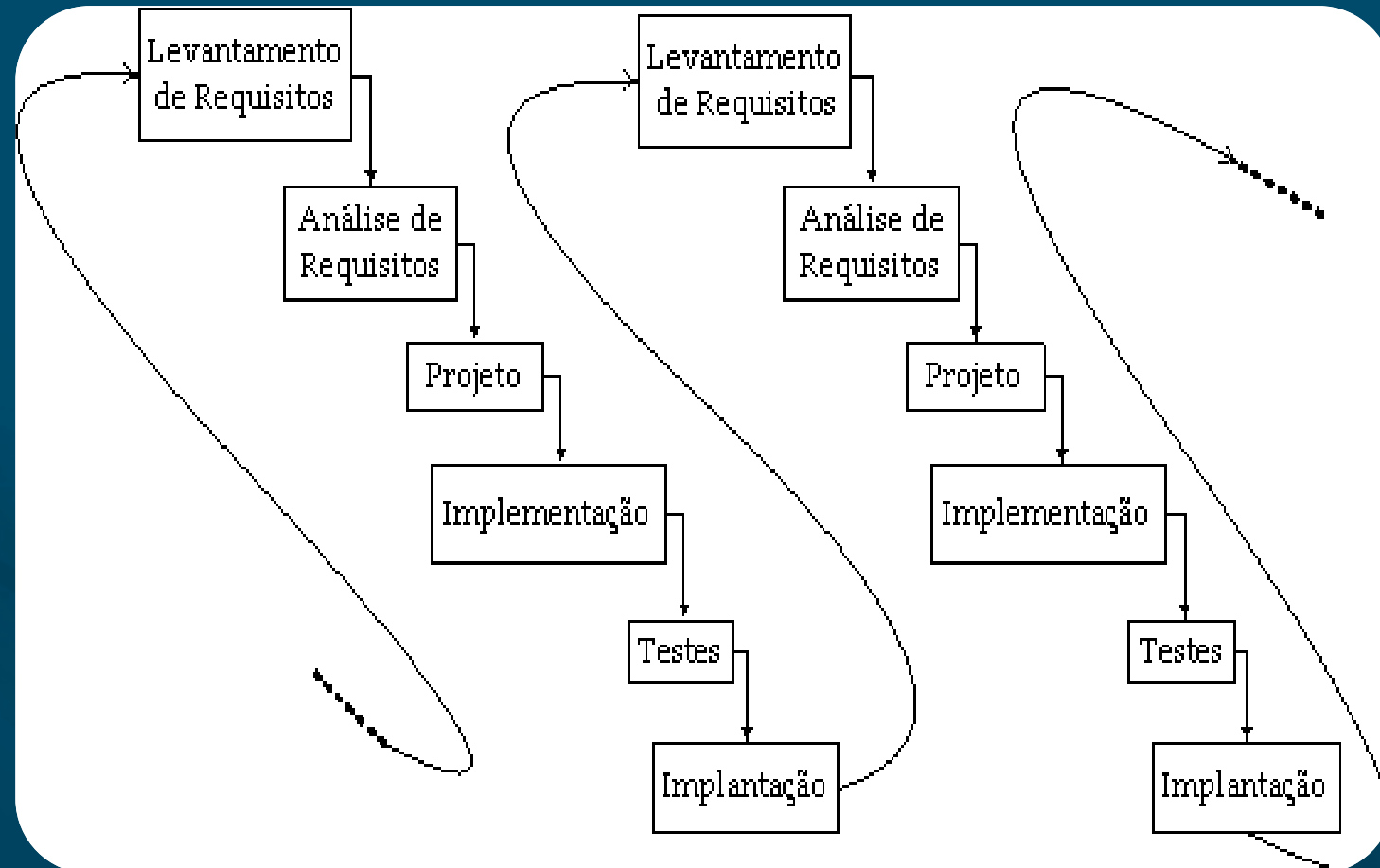
# Processos de software



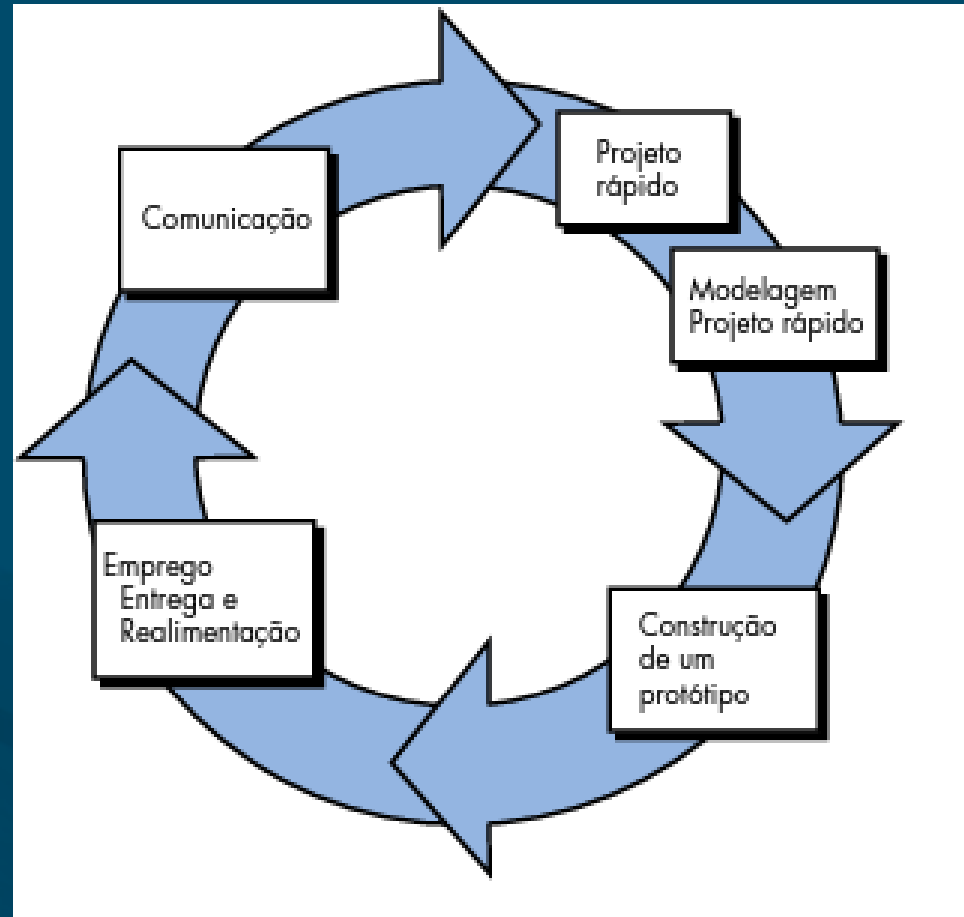
# Clássico (Cascata)



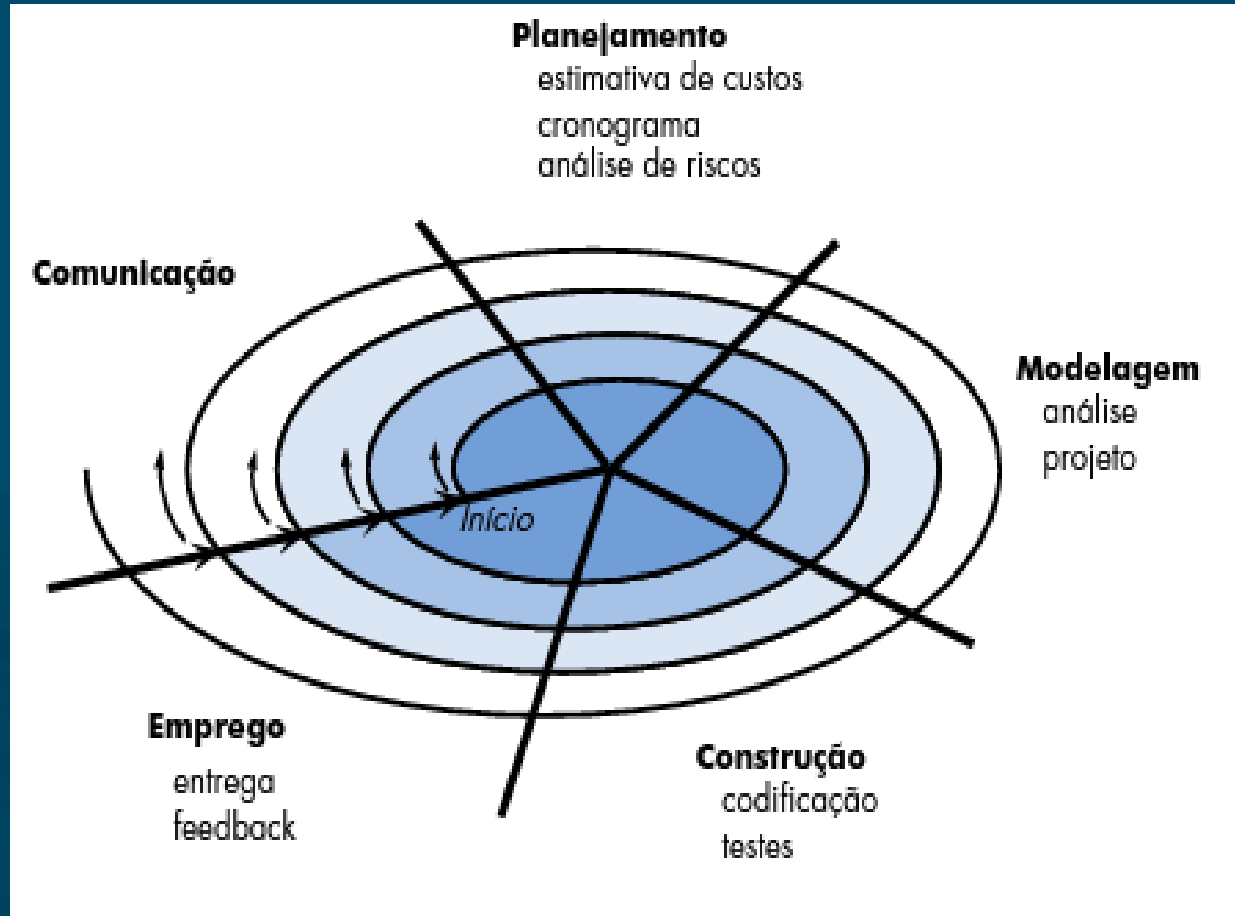
# Incremental



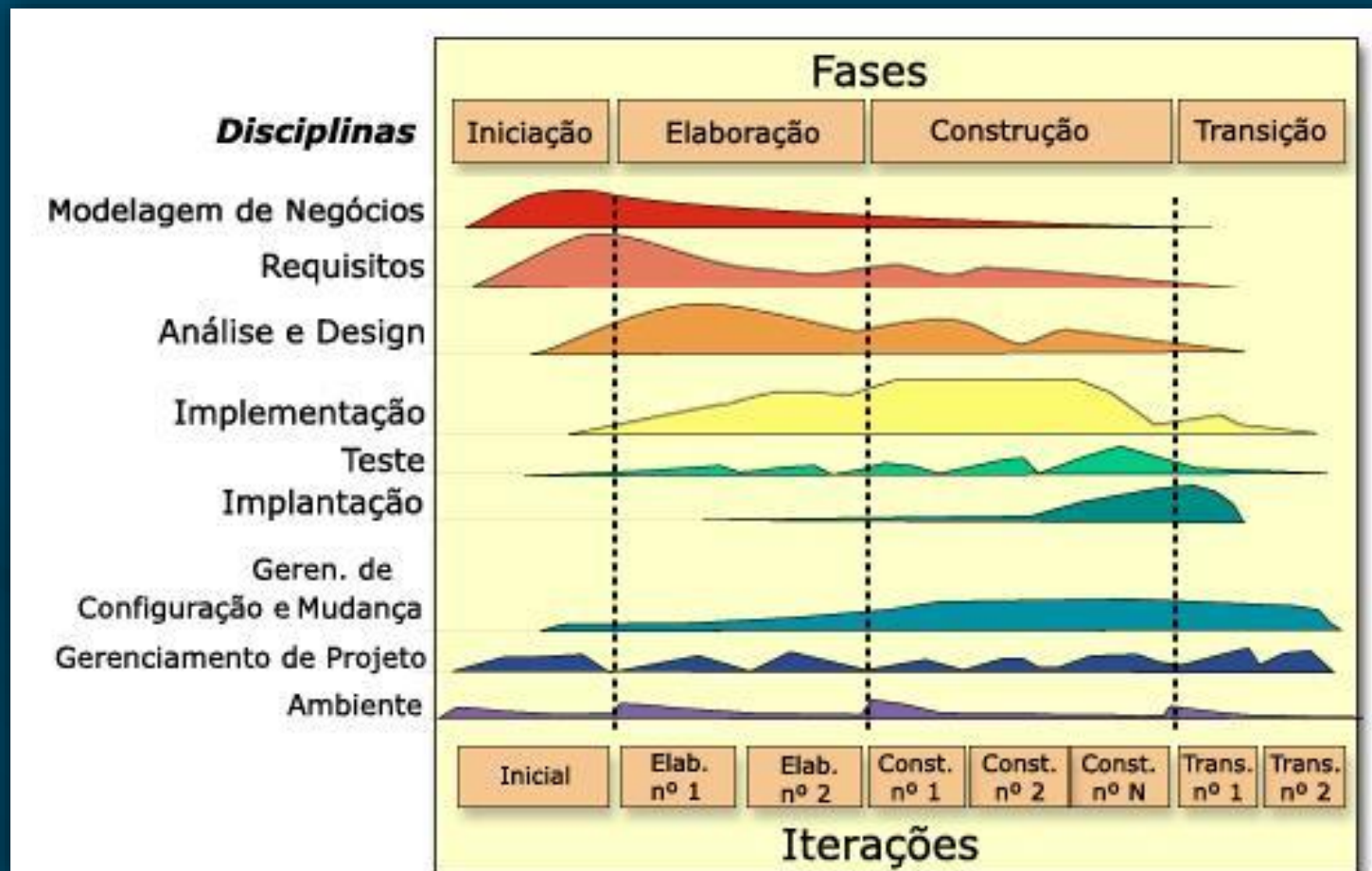
# Evolucionário - Prototipação



# Evolucionário - Espiral



# RUP (Rational Unified Process)





# Prontuário Médico Eletrônico

Estudo de caso...

# Contexto

- Durante uma consulta típica com um médico, normalmente observam-se prateleiras cheias de pastas e papéis dedicados à armazenagem de registros médicos.
- Toda vez que vamos ao médico, nossos registros são criados ou modificados, e muitas vezes cópias duplicadas são geradas durante todo o curso de uma visita ao médico ou a um hospital.



# Contexto

- Atualmente, diversos registros médicos ainda são baseados em papel, tornando esses registros muito difíceis de serem acessados e compartilhados.
- Em 2012, os custos com manutenção de registros médicos atingiram 2,8 trilhões de dólares, representando 18% do produto interno bruto (PIB) norte-americano.

# Contexto

- Estimativas indicam que se nada for feito, até 2037 os custos de saúde subirão para 25% do PIB e consumirão aproximadamente 40% dos gastos federais.
- Aperfeiçoar os sistemas de manutenção de registros médicos tem se tornado uma meta importante a fim de reduzir os custos e até mesmo obter uma maior qualidade do serviço.

# Contexto

- Trata-se dos Sistemas de Prontuário Médico Eletrônicos (EMR – *Eletronic Medical Records*).



# Levantamento de Requisitos

- No caso do sistema de prontuário médico eletrônico, incluem-se como *stakeholders*:
  - Pacientes cujas informações estão registradas no sistema;
  - Médicos responsáveis pela avaliação e tratamento dos pacientes;
  - Enfermeiros que coordenam as consultas e administram tratamentos;

# Levantamento dos Requisitos

## ■ continuação...

- Recepcionistas dos médicos, que gerenciam as consultas dos pacientes;
- Equipe de TI, responsável pela instalação e manutenção do sistema;
- Gerente de ética médica, que deve garantir que o sistema atenda às diretrizes éticas atuais no atendimento ao paciente;

# Levantamento de Requisitos

## ■ continuação...

- Gerentes de saúde, que obtêm informações de gerenciamento a partir do sistema;
- A equipe de registros médicos, responsável por garantir que as informações do sistema sejam mantidas e preservadas, e que os procedimentos de manutenção dos registros sejam devidamente implementados.

# Dados necessários (Requisitos)

- O sistema deve conter todos os dados médicos vitais de uma pessoa, incluindo:
  - Informações pessoais;
  - Histórico médico completo;
  - Resultado dos exames;
  - Diagnósticos;
  - Tratamentos;
  - Prescrição de medicamentos;
  - Efeitos desses tratamentos.



# Dados necessários (Requisitos)

- O sistema deve possibilitar ao médico a capacidade de acessar diretamente e de imediato as informações necessárias, sem ter que examinar arquivos de papel.



# Dados necessários (Requisitos)

- Quando um paciente for ao hospital, os registros desse paciente e os resultados de quaisquer exames por ele realizados até aquele momento ficarão disponíveis on-line.
- Ter “em mãos” um conjunto completo de informações sobre o paciente ajudará o médico a evitar as interações entre remédios prescritos e exames redundantes.

# Dados necessários (Requisitos)

- Muitos especialistas acreditam que o sistema de prontuário médico eletrônico reduzirá os erros médicos e melhorará o atendimento, criará menos burocracia e prestará um serviço mais rápido, resultando em uma economia dramática no futuro.

# Decisão...

- Tanto para o desenvolvimento de uma nova aplicação quanto para a implantação de um sistema existente, um processo de software adaptado ao ciclo de vida de sistemas deverá ser utilizado.



# Mapa Mental do Ciclo de Vida do Prontuário Médico Eletrônico



# Referências

- Sommerville, I. Engenharia de Software. 9ª ed. Pearson: São Paulo, 2011.
- Laudon, K. C.; Laudon J. P. Sistemas de Informação Gerenciais. 11ª ed. Pearson: São Paulo, 2014.