

Processos de Software

Prof. Allan Garcia

O que veremos?

- * O que é um software?
- * Como transformar necessidades em software?
- * O que é processo de software?
- * Processos de software

O que é um software?

- * Software compreende tudo que é necessário para um sistema computacional funcionar, tais como:
 - * Programa de computador
 - * Documentação
 - * Arquivos de configuração
 - * Entre outros...

Como transformar necessidades em software?

- * Devem ser consideradas as atividades:

- * Entender as necessidades do cliente

- * Planejar solução

- * Implementar solução

- * Validar solução

- * Entregar o produto

Como transformar necessidades em software?

- * Estas atividades são executadas
 - * Ordenadas ou não-ordenadas
 - * Formalmente ou informalmente

O que é um processo de software?

- * Processo de software é um conjunto de atividades que leva a produção de um produto de software
- * Complexo, e depende do julgamento humano

O que é um processo de software?

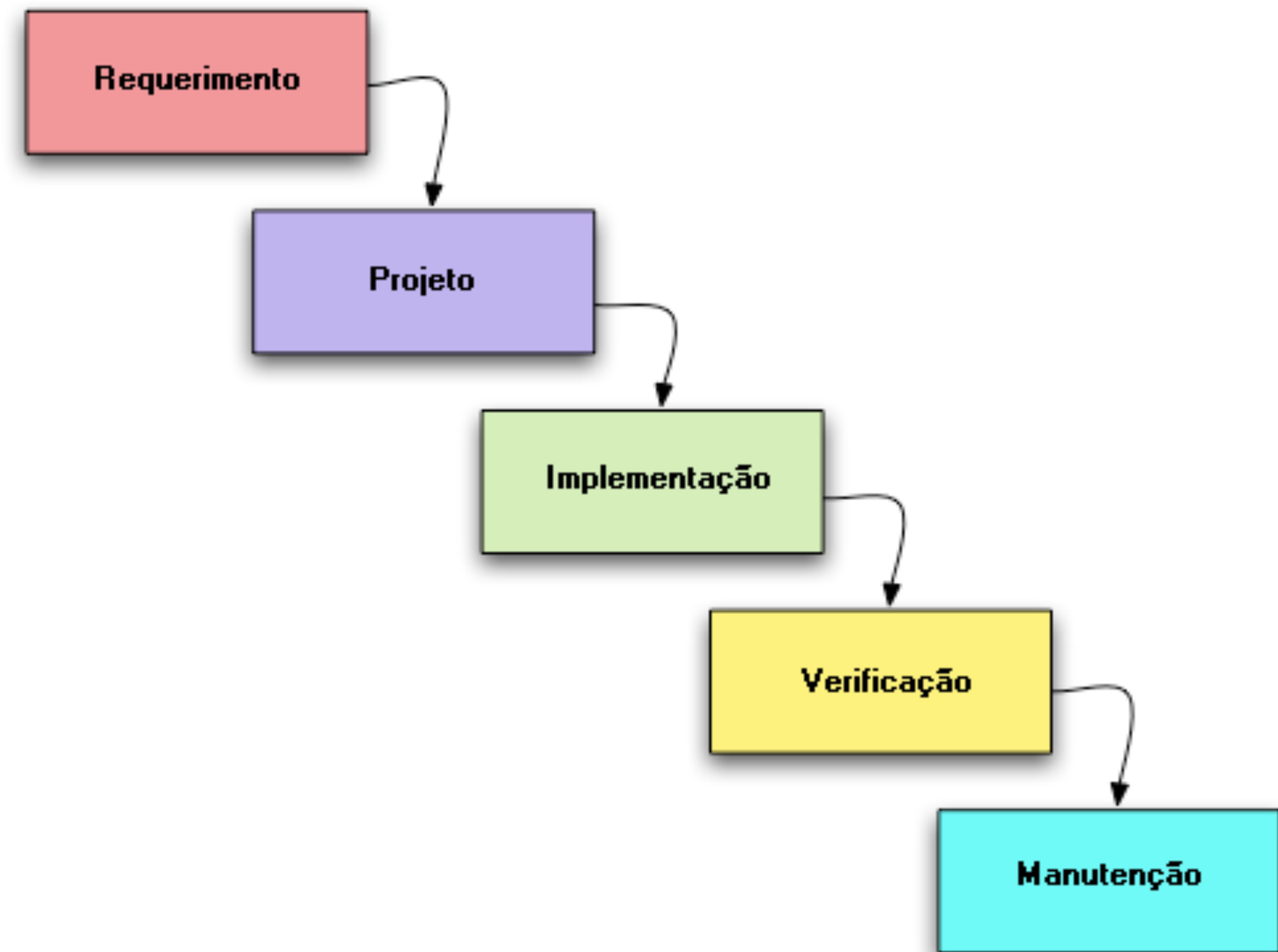
- * Atividades fundamentais
 - * Especificação do software
 - * Projeto e implementação
 - * Validação do software
 - * Evolução do software

Processos de software

- * Todo processo de transformação tem início e fim
- * Essa variável temporal, denominada de ciclo de vida, determina as fases do desenvolvimento de um software
- * Um modelo de processo de desenvolvimento de software é uma representação abstrata de um processo
- * Deve determinar quais atividades devem ser realizadas, quando, como e por quem

Modelo em cascata

- * Primeiro modelo de processo de desenvolvimento de software
- * As atividades são estruturadas numa cascata onde a saída de uma é a entrada para a próxima
- * Cada atividade só pode ser iniciada após a anterior ter sido terminada e verificada



Problemas do modelo em cascata

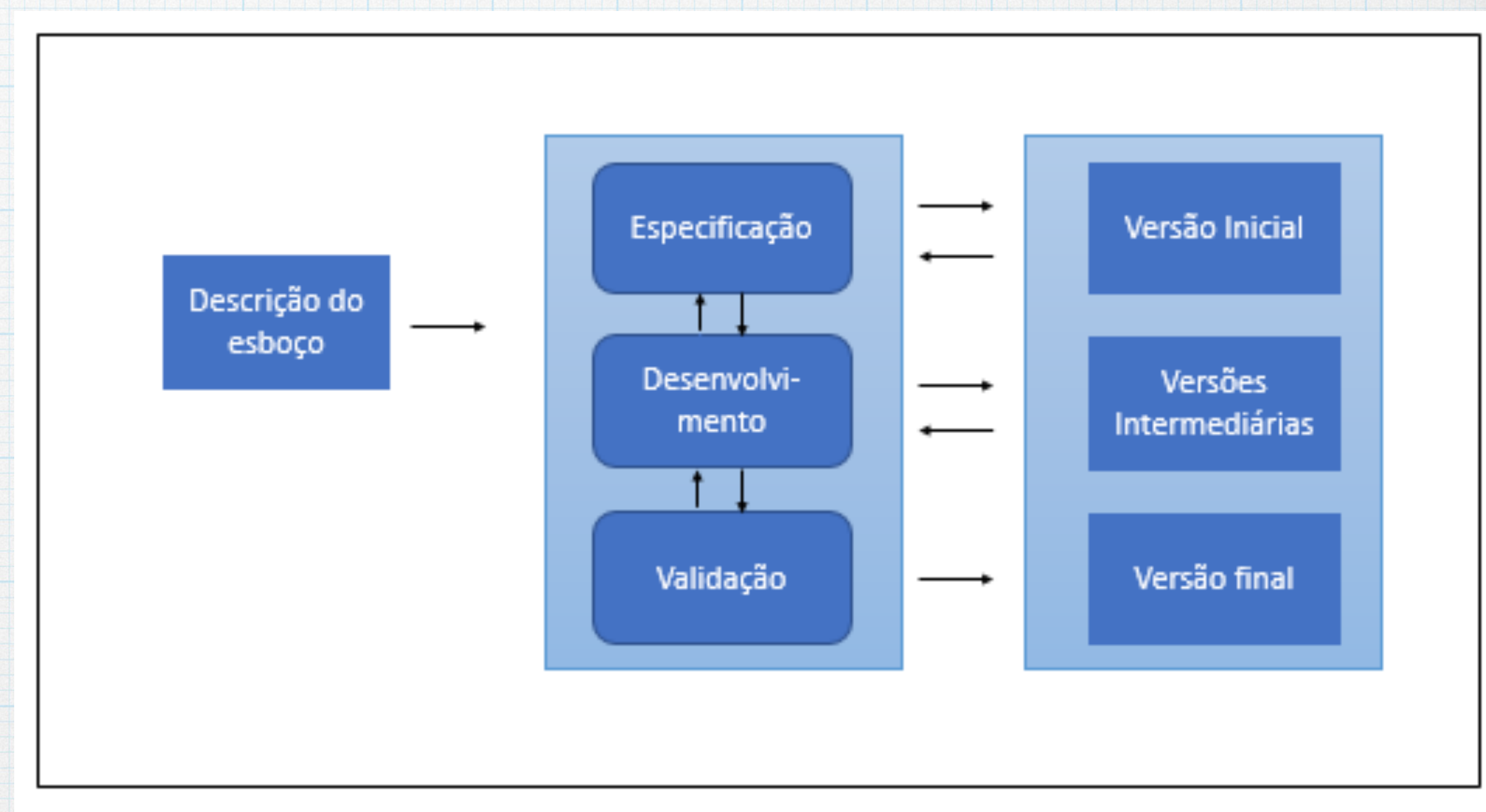
- * Projetos reais raramente seguem o fluxo sequencial que o modelo propõe
- * Não introduzir a participação de clientes e usuário durante as atividades do desenvolvimento
- * Durante a fase final, erros e omissões nos requisitos originais são descobertos

Indicação do modelo em cascata

- * Só deve ser usado quando os requisitos forem bem compreendidos e houver pouca probabilidade de mudanças radicais durante o desenvolvimento

Modelo evolutivo ou evolucionário

- * Baseia-se na ideia de desenvolvimento de uma implementação inicial, expondo o resultado aos comentários do usuário e refletindo esse resultado por meio de várias versões até que seja desenvolvido um sistema adequado
- * A especificação pode ser desenvolvida de forma incremental



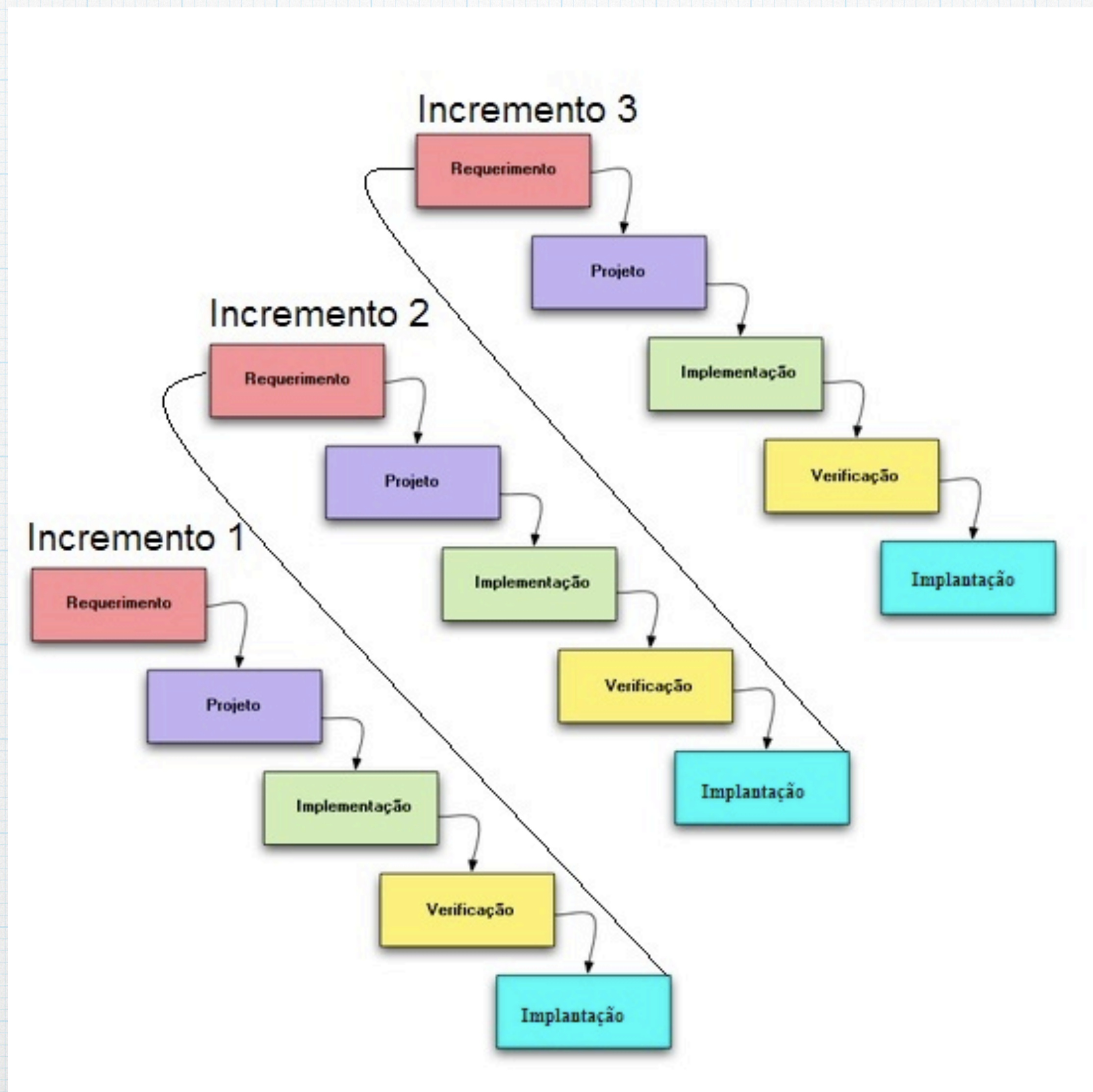
Problemas do modelo evolutivo

- * O processo não é visível
- * Os sistemas são frequentemente mal estruturados

Modelo iterativo

- * O pressuposto básico do modelo iterativo é que o processo de software não é uma execução única
- * A mudança é inevitável em todos os projetos
- * Os requisitos mudam
- * Em razão do surgimento de novas tecnologias os projetos e implementações mudam

Modelo iterativo



Modelo iterativo

- * No processo iterativo, a especificação é desenvolvida conjuntamente com o software
- * Não existe uma especificação completa de software até o incremento final seja software até o incremento final seja especificado
- * O modelo iterativo ainda contém dois sub-modelos: a **Entrega Incremental** e o **Desenvolvimento em Espiral**

Entrega incremental

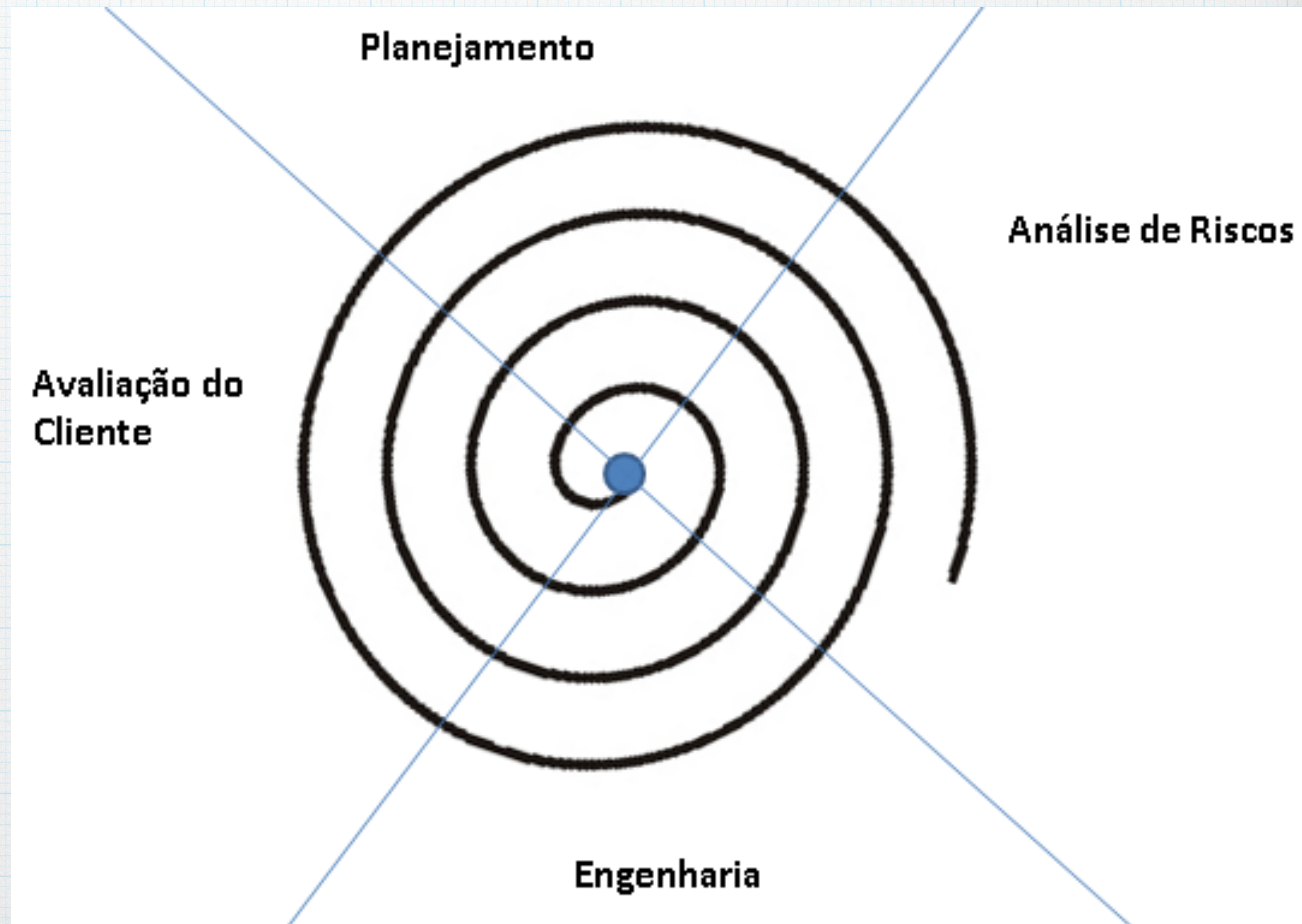
- * Clientes não precisam esperar até a entrega do sistema inteiro para se beneficiarem dele
- * Clientes podem usar os incrementos iniciais como protótipos e ganhar experiência
- * Existe risco menor de falha geral do projeto
- * Os serviços mais importantes recebem mais testes

Entrega incremental

- * Abordagem intermediária entre o modelo cascata e evolutivo
- * O cliente identifica, em linhas gerais, os serviços a serem fornecidos pelo sistema (quais são mais importantes e quais são menos)
- * Um número de incrementos de entrega é definido
- * Após a identificação dos incrementos, os requisitos dos serviços a serem entregues no 1º incremento são definidos detalhadamente e ele é desenvolvido

Desenvolvimento em espiral

- * Cada loop na espiral representa uma fase do processo de software processo de software



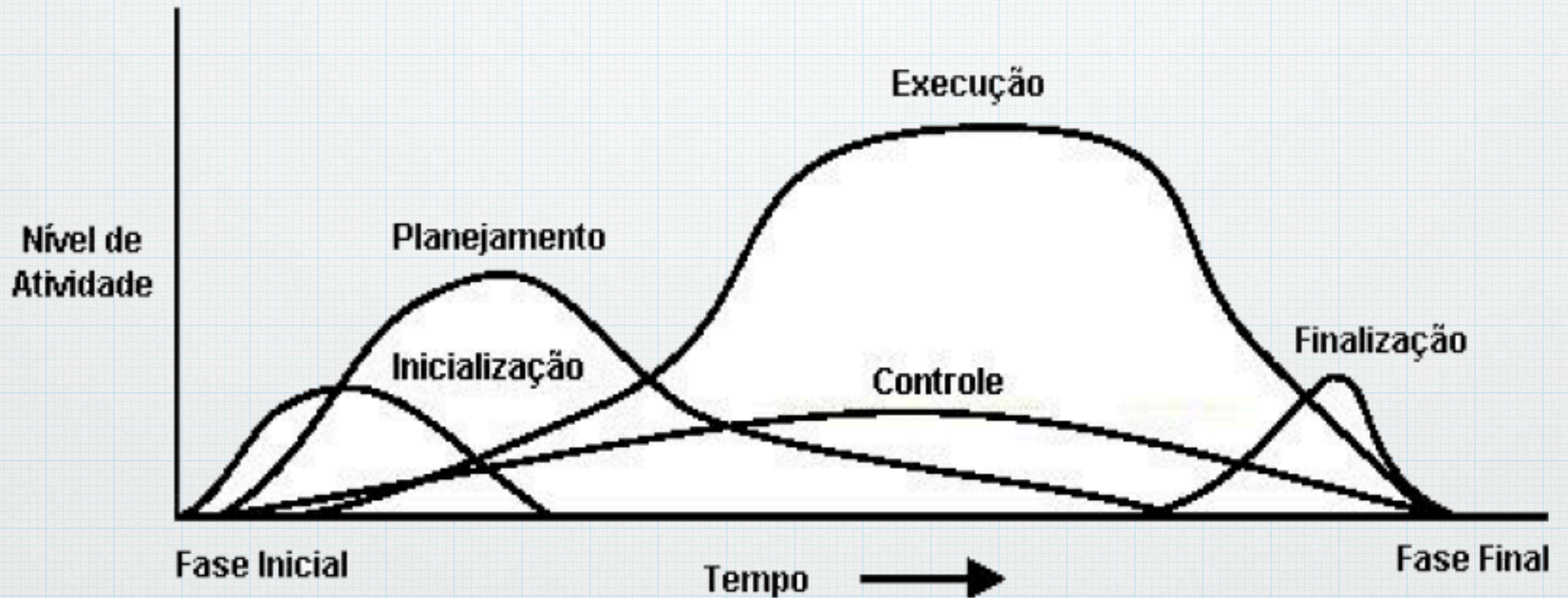
Desenvolvimento em espiral

- * O loop mais interno por estar relacionado com a viabilidade do sistema; o próximo loop, à definição de requisitos; o próximo ao projeto de sistema e assim por diante
- * Cada loop é dividido em 4 setores
- * O diferencial desse modelo é o reconhecimento do risco

Processo de software

- * Como o processo de desenvolvimento de software se distribui/concentra no tempo dentro de um projeto?

Processo de software



**Na próxima unidade veremos
o processo unificado...**