

Processo Unificado

Agenda

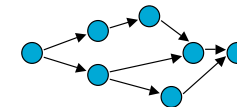
- Introdução
 - Motivação para Processos de Software
- Processo Unificado (USDP)
 - Definições
 - RUP x USDP
 - Características do Processo Unificado
- Descrição detalhada do Processo Unificado
- Processos Derivados
- Templates simplificados
- Conclusões

Introdução

- Comparando o processo de software com outros processos industriais
- Crescente preocupação com processos
- Modelos de Processos Populares

Processo de Software

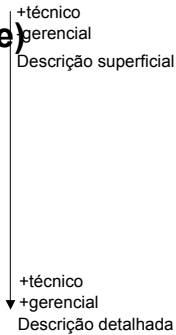
- Definição para Processo de Software
 - “Todos os elementos do mundo real envolvidos no desenvolvimento e manutenção de um produto de software”
 - Inclui os recursos, ferramentas, atividades, artefatos e organização (Derniame, 1998 apud GDPA)



Processo de Software

- Processos vem sendo propostos pela indústria, países e academia

- **Análise Estruturada (Yourdon, Gane)**
- Método de Jackson
- Objectory (Jacobson)
- V-Model (Alemanha)
- Catalysis
- **Rational Unified Process - RUP**
- XP - eXtreme Programming



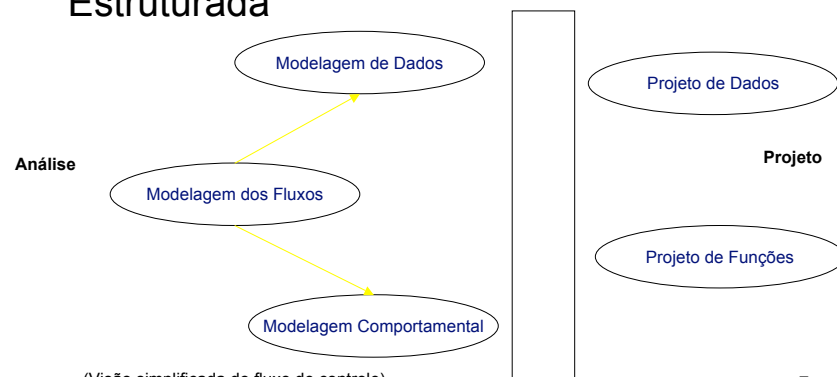
Processo de Software

- Exemplo de Processo: Análise Estruturada

- Proposta por uma diversidade de autores nas décadas de 1980 e 1990
- Fundamentação:
 - Programação Estruturada
 - Estruturas de repetição, decisão e seqüência
 - Projeto Estruturado
 - Principal mecanismo de abstração: decomposição funcional
 - Ciclo de Vida Cascata
 - Processo Seqüencial Linear

Processo de Software

- Exemplo de Processo: Análise Estruturada



(Visão simplificada do fluxo de controle)

Processo Unificado

- Histórico e Definições
- RUP x USDP
- Características do Processo Unificado

Processo Unificado

- Definição principal
 - O processo “oficial” definido para apoiar o uso da UML
 - Necessidade a partir do sucesso da UML como padrão de fato para especificação de software

Processo Unificado

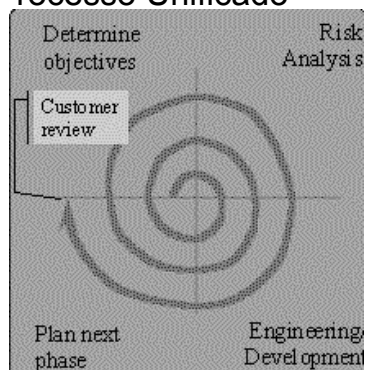
- Histórico: UML
 - Unified Modeling Language (UML)
 - Linguagem visual para sistemas orientados a objetos
 - Unified Method 0.8: 1995
 - Padrão de fato e de direito
 - UML foi proposta somente como uma linguagem, sem orientação de uso (i.e., sem um processo)



Processo Unificado

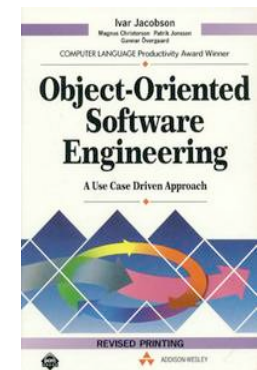
- Histórico: Processo Unificado
 - Bases históricas do Processo Unificado

- Processo Espiral
 - Iteratividade
 - Gerência de riscos



Processo Unificado

- Histórico: Processo Unificado
 - Bases históricas do Processo Unificado
 - Processo *Objectory*
 - Proposto por Jacobson et al
 - Processo direcionado pelos Casos de Uso

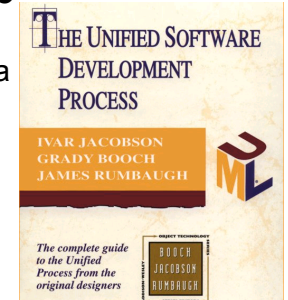


Processo Unificado

- O que é o Processo Unificado?
 - Pode ter 2 respostas:
 - Modelo de Processo Padrão
 - Produto comercial da IBM/Rational

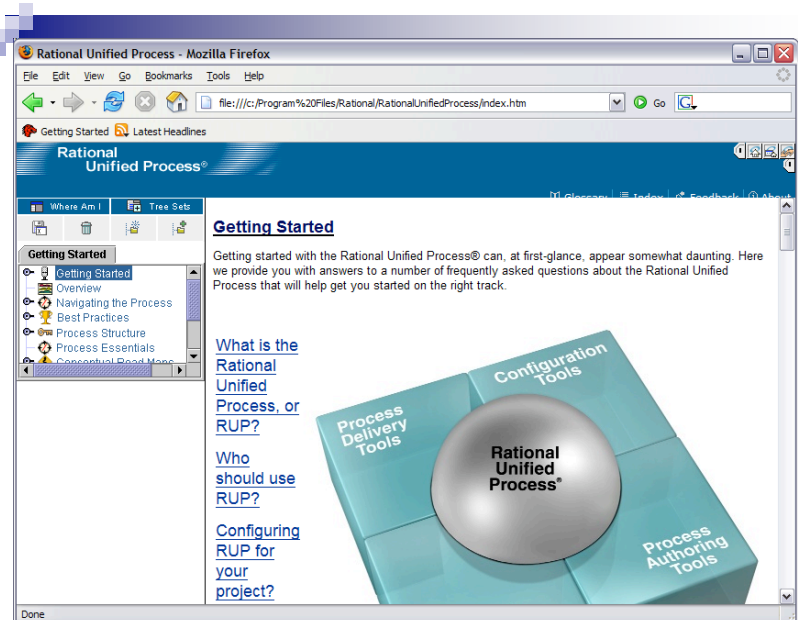
Processo Unificado: Introdução

- Definições: o que é Processo Unificado
 - ...**Modelo de Processo Padrão**
 - Descrição de atividades que compõem um processo que adota UML
 - Mais simples que a proposta da Rational



Processo Unificado: Introdução

- Definições: O que é Processo Unificado
 - ...**Produto comercial**
 - Desenvolvido e mantido pela Rational
 - Integrado a suite de produtos
 - Disponível em CD-ROM / Internet
 - Conhecido como Rational Unified Process
 - E-coach: treinamento a distância
 - <http://www.rational.com/rup>
 - Para o treinamento online, clicar em "Trials & Betas"



Artifact: Vision

Defines the stakeholders view of the product to be developed, specified in terms of the stakeholders key needs and features. Containing an outline of the envisioned core requirements, it provides the contractual basis for the more detailed technical requirements.

Role: [System Analyst](#)

Optionality/Occurrence: Created early in the Inception phase. Evolving during the earlier portion of the lifecycle.

Templates and Reports:

- [Template: Vision](#)
- [Template: Vision \(Informal\)](#)

Examples:

- [CREG Vision - Inception Phase](#)
- [CSPS Vision - Inception Phase](#)

UML Representation: Not applicable.

More Information:

- [Checklist: Requirements Attributes](#)
- [Guideline: Requirements Management Plan](#)
- [Checklist: Stakeholder Requests](#)
- [Checklist: Vision](#)

Descrição do artefato Vision

quites@computer.org

17

<Project Name> Vision

- Introduction**
- Positioning**

- 2.1 Problem Statement**
[Provide a statement summarizing the problem being solved by this project. The following format may be used:]

The problem of	[describe the problem]
affects	[the stakeholders affected by the problem]
the impact of which is	[what is the impact of the problem?]
a successful solution would be	[list some key benefits of a successful solution]

- 2.2 Product Position Statement**
[Provide an overall statement summarizing, at the highest level, the unique position the product intends to fill in the marketplace. The following format may be used:]

For	[target customer]
Who	[statement of the need or opportunity]
The (product name)	is a [product category]
That	[statement of key benefit, that is, the compelling reason to buy]
Unlike	[primary competitive alternative]
Compared to	[statement of primary differentiator]

Template para o artefato Vision

quites@computer.org


O gerenciamento de Riscos deve ser feito continuamente

- A cada iteração (novos) riscos devem ser identificados e tratados;
- Isto “garante” que o desenvolvimento terá sucesso;

quites@computer.org

20

“The Spirit of RUP”



Foco em Funcionalidades para o Cliente

- Especificação, organização e documentação dos requisitos é facilitada através dos diagramas de casos de uso;
- Casos de uso guiam todo o processo de desenvolvimento
 - O que desenvolver, testar e validar em cada iteração;
- Casos de uso são funcionalidades para o cliente;



Foco em Software Executável

- Artefatos são construídos para facilitar e documentar o processo de desenvolvimento;
- Mas, **não** é necessário construir todos os artefatos indicados pelo RUP;



Aprenda a lidar com Mudanças

- Mudanças são inevitáveis no processo de desenvolvimento;
- Adote estratégias para gerenciar mudanças
 - Tomada de decisão sobre uma mudança;
 - Impacto desta mudança no sistema;
 - Minimizar o custo desta mudança;



Defina uma Arquitetura estável cedo

- Uma arquitetura do sistema é definida, implementada e testada no início do processo (Elaboração) para garantir que o sistema atenderá aos requisitos funcionais e não-funcionais;
- Com a arquitetura definida, o processo de construção é mais simples;

Considere continuamente a Qualidade

- O controle de qualidade deve ser feito desde o início do processo de desenvolvimento
 - Inspeção de software;
 - Teste dos casos de uso implementados;
 - Definição de casos de teste a partir dos casos de uso;

Desenvolvimento Iterativo

- “Impossível” desenvolver o sistema em uma única iteração;
- A cada iteração mais detalhes são adicionados;
- Diversas vantagens:
 - Redução da Complexidade;
 - Facilidade para lidar com mudanças nos requisitos, cronograma, etc.

Estrutura do Processo Unificado

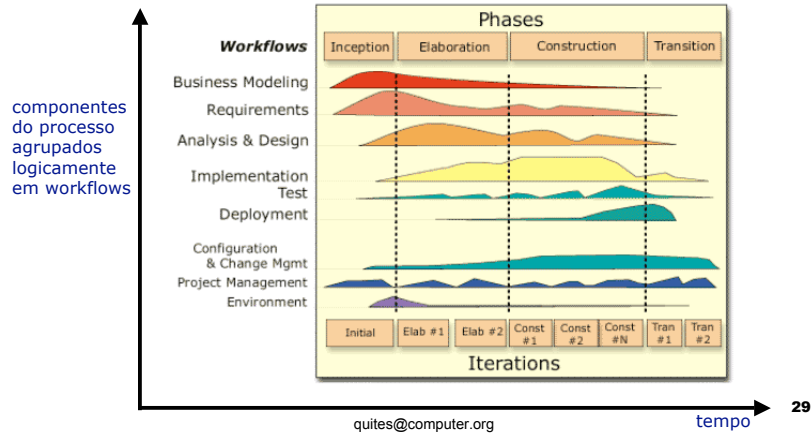
Processo Unificado

- Estrutura do Processo Unificado
 - Processo Iterativo, baseado no modelo Espiral
 - Iterativo: baseado em sucessivas versões
 - Espiral:



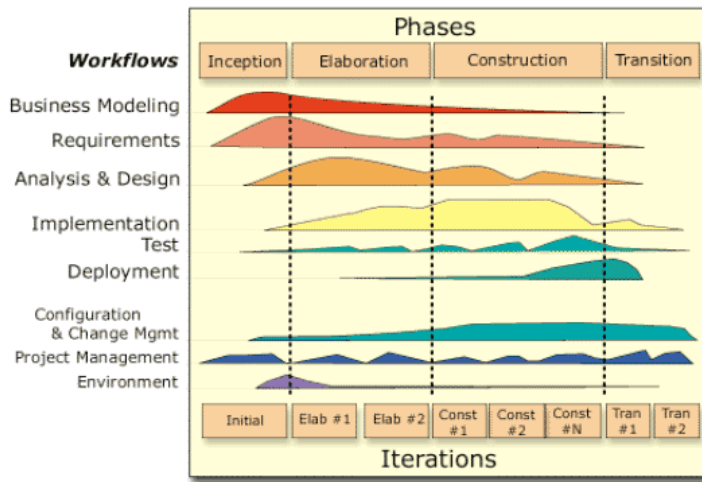
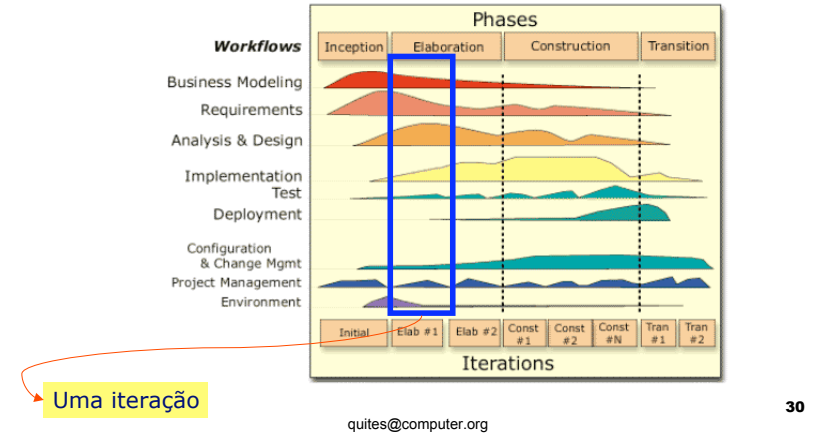
Processo Unificado

■ Estrutura do Processo Unificado

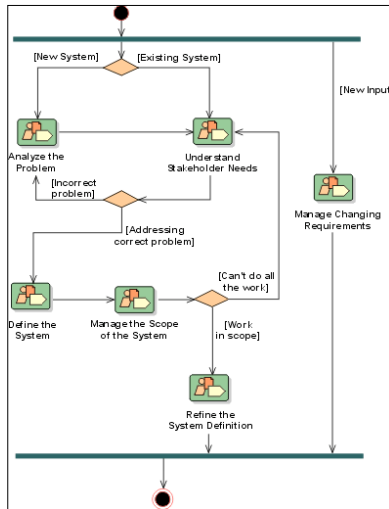


Processo Unificado

■ Estrutura do Processo Unificado

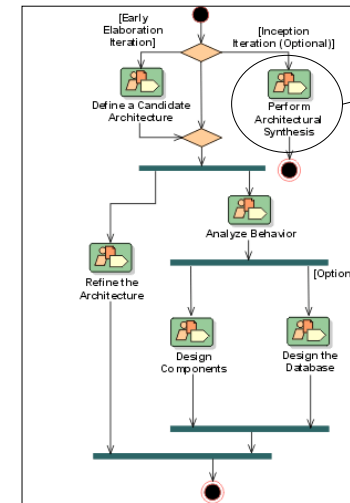


Workflow de Requisitos



33

Workflow de Análise e Projeto

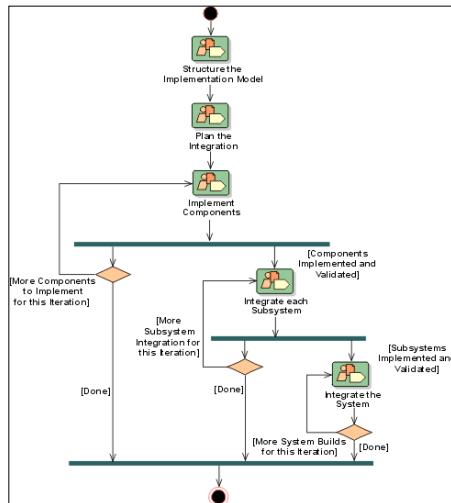


Detalhado a seguir



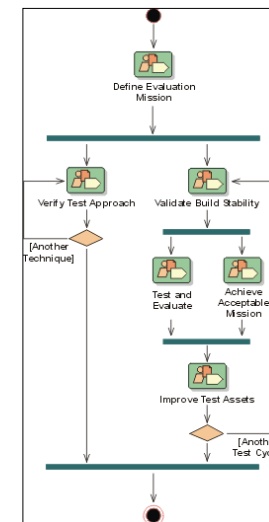
34

Workflow de Implementação



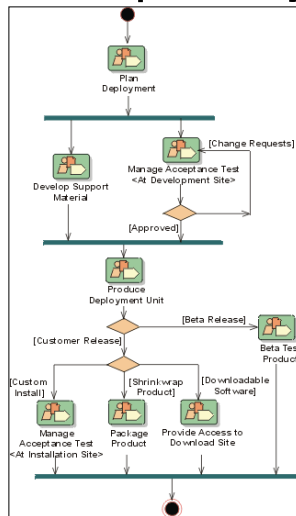
35

Workflow de Testes



36

Workflow de Implantação



37

Processo Unificado: detalhamento das etapas

Processo Unificado: detalhamento das fases

Concepção

Elaboração

Construção

Transição

Processo Unificado: detalhamento das fases

■ Concepção

Concepção

Elaboração

Construção

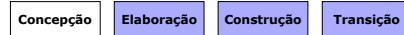
Transição

□ Objetivos

- Estabelecer escopo do projeto e condições de fronteira
- Descrever os casos de uso críticos do sistema
- Descrever pelo menos uma arquitetura candidata para os principais casos de uso
- Estimar o custo e cronograma para a Elaboração
- Estimar riscos (fontes de incerteza)

Processo Unificado: detalhamento das fases

■ Conceção

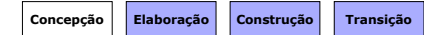


□ Atividades

- Descrever o escopo do projeto
 - Capturar o contexto na forma de requisitos e restrições para determinar um critério de aceitação do produto final
- Planejar e preparar o Plano de Negócios
 - Avaliação de riscos, staff, plano de projeto e relações entre custo, cronograma e lucro
- Preparar uma arquitetura candidada
 - Avaliar alternativas de projeto (atividade pode ser suprimida se o sistema não possui novidades ou possui uma arquitetura bem conhecida)
- Preparar o ambiente de projeto (environment)
 - Escolha de recursos físicos e humanos, e ferramentas de software
- Obs: Geralmente a concepção é completada em dois dias ou menos para sistemas pequenos

Processo Unificado: detalhamento das fases

■ Conceção

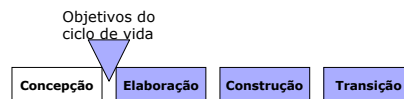


□ Artefatos produzidos

- O documento de **Visão**, isto é, a visão geral dos requisitos principais do sistema, incluindo funcionalidades principais e restrições
- O modelo de caso de uso, listando todos os casos de uso e atores que podem ser identificados neste início (10% a 20% do total)
- Um glossário inicial do projeto
- Um plano de negócios inicial, contendo:
 - Contexto do negócio, Critério de sucesso (projeção de lucro, reconhecimento do mercado, etc), Provisionamento Financeiro
- Análise de Riscos Inicial
- Um plano de projeto (para etapa de Elaboração)
- Um ou mais protótipos

Processo Unificado: detalhamento das fases

■ Conceção



□ Milestone: Objetivos do ciclo de vida

- Acordo com stakeholder acerca da definição de escopo, e estimativas de custo e cronograma
- Entendimento dos requisitos (evidenciado pelos principais casos de uso)
- Estimativas reais de custo e cronograma, prioridades, riscos e processo
- Protótipo de Arquitetura do software

Processo Unificado: detalhamento das fases

■ Elaboração



□ Objetivos

- Definir e validar uma arquitetura *baseline*
 - Baseline - release estável que serve como ponto de partida e referência no desenvolvimento futuro
- Gerar uma Visão baseline
- Gerar um plano detalhado para a fase de construção
- Demonstrar que a arquitetura *baseline* irá atender a revisão no custo e tempo estimados

Processo Unificado: detalhamento das fases

■ Elaboração



□ Atividades

- Elaborar a visão: entendimento sólido dos casos de uso mais críticos (que determinam as decisões arquiteturais e de planejamento)
- A arquitetura é elaborada e componentes de software são selecionados
 - Componentes potenciais são avaliados segundo decisões *make/buy/reuse* para determinar custo e estimativa
 - Lições obtidas podem servir para gerar o novo projeto da arquitetura do sistema

Processo Unificado: detalhamento das fases

■ Elaboração

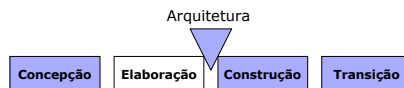


□ Artefatos produzidos

- Um modelo de caso de uso (pelo menos 80% dos casos de uso)
- Requisitos suplementares que capturem requisitos não-funcionais e requisitos que não estão associados com um caso de uso específico
- Uma descrição da arquitetura de software
- Um protótipo arquitetural executável
- Uma lista revisada dos riscos e plano de negócios
- Um plano para as próximas iterações
- Um manual do usuário preliminar

Processo Unificado: detalhamento das fases

■ Elaboração



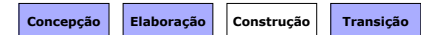
□ Milestone: Arquitetura

■ Perguntas:

- A visão do produto é estável?
- A arquitetura é estável?
- O plano para Construção está suficientemente detalhado e correto?
 - Iterações x Releases
- O cliente está de acordo com a visão?
- A alocação de recursos está de acordo com o previsto?

Processo Unificado: detalhamento das fases

■ Construção



□ Atividades:

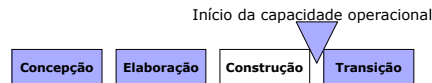
- Gerenciamento de recursos
- Desenvolver e testar os componentes
- Avaliar e, eventualmente, prosseguir para a próxima iteração

□ Artefatos

- Produto de software integrado na plataforma de hardware
- Manuais de usuário
- Descrição dos releases

Processo Unificado: detalhamento das fases

■ Construção



- Milestone: Início da Capacidade Operacional
 - O release está maduro e estável para ser usado?
 - Todos os *stakeholders* estão prontos para a transição?
 - O consumo de recursos é aceitável?

Processo Unificado: detalhamento das fases

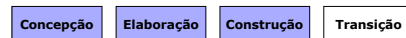
■ Transição



- Objetivo geral:
 - Garantir que o software esteja disponível para usuários finais
- Atividades
 - Finalizar o material de apoio ao usuário final
 - Testar o produto entregue
 - Simular o ambiente do cliente (se possível) ou instalar o software no cliente
 - Realizar um ajuste fino do produto com base no feedback
 - Entregar o produto final para o usuário

Processo Unificado: detalhamento das fases

■ Transição



- Artefatos
 - Release Notes
 - É raro o produto que não possui instruções e modificações de "último-minuto"
 - Material de treinamento e documentação

Processos
Derivados

Processos Derivados

- Grande número de processos surgiram para customizar ou estender o Processo Unificado
- Experiências na indústria e academia
- Há uma verdadeira coqueluche em adaptações de RUP para empresas específicas

Templates Simplificados

2.1 Fase de Concepção

2.1.1 Iteração 1 - <<data>>

2.1.1.1. Planejamento do Projeto

Descrever fases, prazos e alocação de alunos às atividades na etapa de Concepção, descrevendo ainda um planejamento inicial das próximas etapas (Elaboração, Construção e Transição)

2.1.1.2. Fluxo de Requisitos

Fornecer os <<Diagramas de caso de uso>> (**explicados**)

Todos os casos de uso essenciais do sistema devem ser encontrados na etapa de Concepção

2.1.1.3. Fluxo de Análise e Projeto

Fornecer os << diagramas de classes e pacotes >> (**explicados**)

Para cada caso de uso deve ser fornecido um diagrama de classes.

O diagrama de pacotes pode ser produzido para agrupar as classes do sistema de forma separada. Quanto ao diagrama de pacotes, recomenda-se:

-Apresentá-lo somente na segunda iteração do sistema (ainda na Concepção)

-Identificar todas as classes que compõem um pacote, se esta informação não for fornecida graficamente pela ferramenta CASE

2.1.1.4. Fluxo de Implementação

Incluir, por exemplo, telas do usuário e restrições adicionais para implementação. As telas podem ser construídas à mão ou com um editor. Restrições incluem descrição de características de implementação não documentadas nas seções anteriores.

2.1.2 Iteração 2 (se necessário) - <<data>>

Se houver necessidade de revisão no modelo (novas classes, novos casos de uso, modificação de rótulos, etc), fazer novas iterações (contendo os fluxos descritos acima).

2.2. Fase de Elaboração

2.2.1 Iteração 1 - <<data>>

2.2.1.1. Planejamento do Projeto

2.2.1.2. Fluxo de Requisitos

Acrescentar casos de uso que não foram encontrados na Concepção

2.2.1.3. Fluxo de Análise e de Projeto

Identificar as classes persistentes. Se for usado BD Relacional, fazer o mapeamento para tabelas

2.2.1.4. Fluxo de Implementação

Descrever detalhes de implementação com DTE (por classe), Diagrama de Seqüência (por caso de uso) e Diagrama de atividades (por caso de uso)

2.2.1.5. Fluxo de Testes

Descrever testes a serem realizados no software. Todos os testes devem ser agrupados por casos de uso.

2.2.1.6. Fluxo de Gerenciamento de Projetos (Planejamento da Construção)

Descrever quais versões serão produzidas
Associar versões aos casos de uso encontrados

Continua...

Continuação

2.2.2 Iteração 2 (se necessário) - <<data>>

...

2.2.3 Iteração 3 (se necessário) - <<data>>

...

2.3. Considerações sobre a Fase de Construção

2.4. Considerações sobre a Fase de Transição