

Disciplina

# Engenharia de Software

**DISCIPLINAS DO RUP**

Professor: Gilmar Luiz de Borba

**2014 - 1**

# Conteúdo Programático

## **Disciplina Engenharia de Software (80 horas)**

### **1-Conceitos**

Engenharia de software, Ciência da Computação, Engenharia de Sistemas. Custos, desafios, principais atributos de um bom software. (SOMMERVILLE, 2001:2-17).

### **2- Modelos de Processos de Software (ou Modelos de ciclo de vida)**

Modelo Codifica-Remenda, Modelo em Cascata, Modelo em Espiral, Modelo de Prototipagem evolutiva, Modelo de Entrega Evolutiva, Modelo Dirigido por Prazo, Modelo Dirigido por Ferramenta. (SOMMERVILLE, 2001:18-37) e (PÁDUA, 2005:11-15).

### **3- Processos de Software**

Conceitos, UP (Unified Process), EUP (Enterprise Unified Process), PSP (Personal Software Process), TSP (Team Software Process), PRAXIS (PRocesso para Aplicativos eXtensíveis InterativoS), ICONIX (ICONIX Software Engineering), XP (Extreme Programming), RUP (Rational Unified Process). (PÁDUA, 2005:16-51).

### **4- Rational Unified Process (RUP)**

Modelo Dirigido por Ferramenta. Estrutura, Arquitetura, fluxos (disciplinas), fases, gráfico da baleia, Acesso e manutenção. Kruchten, (2003).

### **5- Prática**

Desenvolvimento de um projeto de software a partir das práticas do RUP.

### **6-Modelagem Ágil de Software e Métodos ágeis,**

Desenvolvimento Ágil, Extreme Programming, Valores, Princípios e Práticas. Discussão participativa em sala de aula. (SOMMERVILLE, 2001:38-51).

# Bibliografia básica

FILHO, Wilson de Pádua. **Engenharia de Software – Fundamentos, Métodos e Padrões**. LTC, 2001.

KRUCHTEN, Philippe. **Introdução ao RUP - Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003.

PRESSMAN, Roger S. **Engenharia de Software**. 6rd ed. McGraw-Hill, 2002.

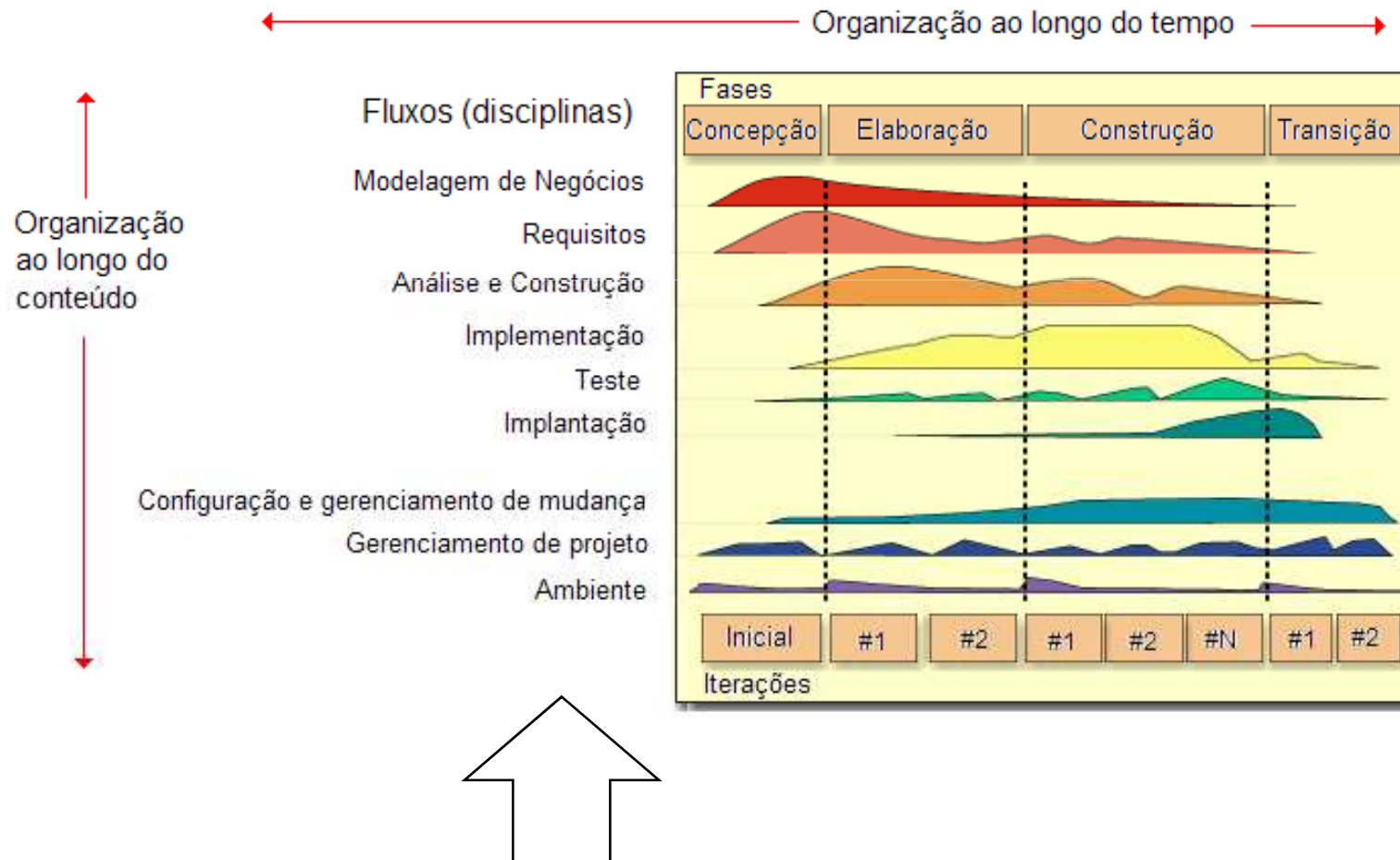
SOMMERVILLE, Ian. **Engenharia de Software**. 9ª. Ed. São Paulo. Pearson, Prentice Hall, 2011.

# Bibliografia complementar

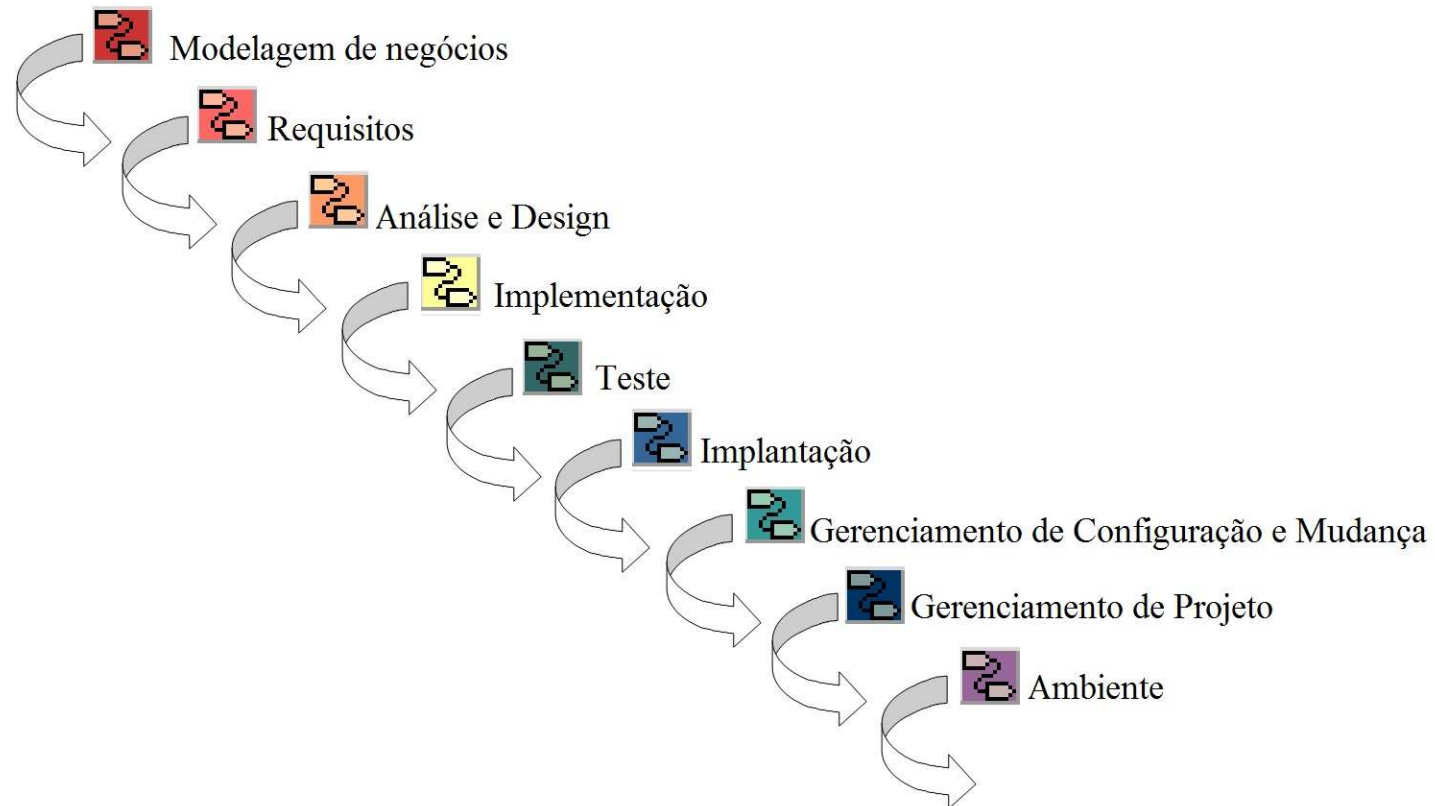
MACHADO, Felipe Rodrigues Nery. **Tecnologia e Projeto de Data Warehouse** – São Paulo SP: Érica: 2004.

OLIVEIRA, Wilson José. **Data Warehouse** – Florianópolis SC: Visual Books: junho de 2002.

# Disciplinas do RUP



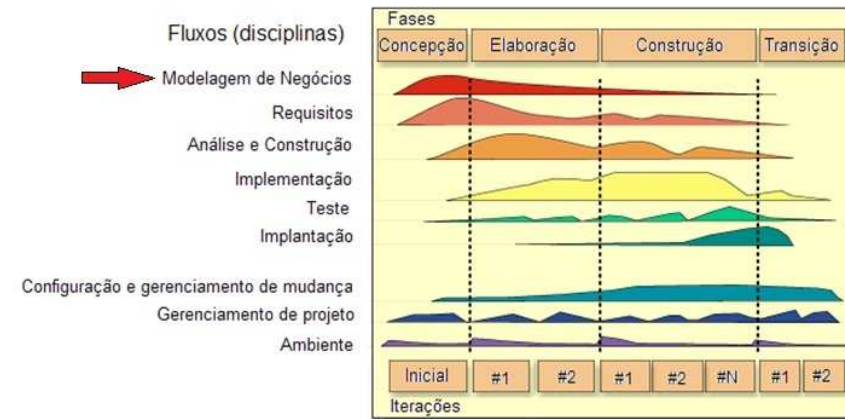
## Introdução - Disciplinas do RUP



# A disciplina de Modelagem de Negócios

A disciplina de modelagem de negócios consiste basicamente em:

- Entendimento da estrutura da organização (onde o sistema será implantado).
- Identificar as possibilidades de melhoria.
- Certificar que cliente e desenvolvedor tenha uma mesma visão do negócio.
- Definir processos, papéis e responsabilidades usando para isso Casos de Uso.



## PESSOAS

Nessa disciplina, o **Analista do Processo de Negócio** é responsável pelas atividades como: avaliar a organização, definir metas, identificar e se apropriar de um vocabulário comum, localizar os atores e casos de uso de negócio. Este trabalhador (Analista do Processo de Negócio) produz artefatos como: glossário de negócios, modelos de caso de uso de negócios e visão do negócio. Além do Analista do Processo de Negócio, nesta disciplina destacam-se os atores: **Designer de Negócios** e o **Revisor do Modelo de Negócios**. Veja o quadro resumo com as atividades e artefatos destes trabalhadores.

**Observação:** trataremos aqui o termo PESSOAS que pode ser entendido também como “papéis” ou “trabalhadores”.

# A disciplina de Modelagem de Negócios

## Modelagem de Negócios:



Disponível em: <http://www.wthree.com/rup/portugues/index.htm>  
 Acessado em: Novembro de 2012

## Trabalhadores (Pessoas)



# A disciplina de Modelagem de Negócios

## Alguns Conceitos:

### Arquitetura de negócios

Conjunto de elementos relacionados no sistema de forma a definir a funcionalidade a que se propõem. Essa arquitetura não deve ser completa mas sim abranger de uma maneira global a organização, de acordo com as principais visões desta. Essas visões são classificadas em: visão de processos de negócios, visão da estrutura organizacional, visão cultural, visão dos recursos humanos e visão de domínio.

Outros conceitos

### Custo Baseado em Atividades

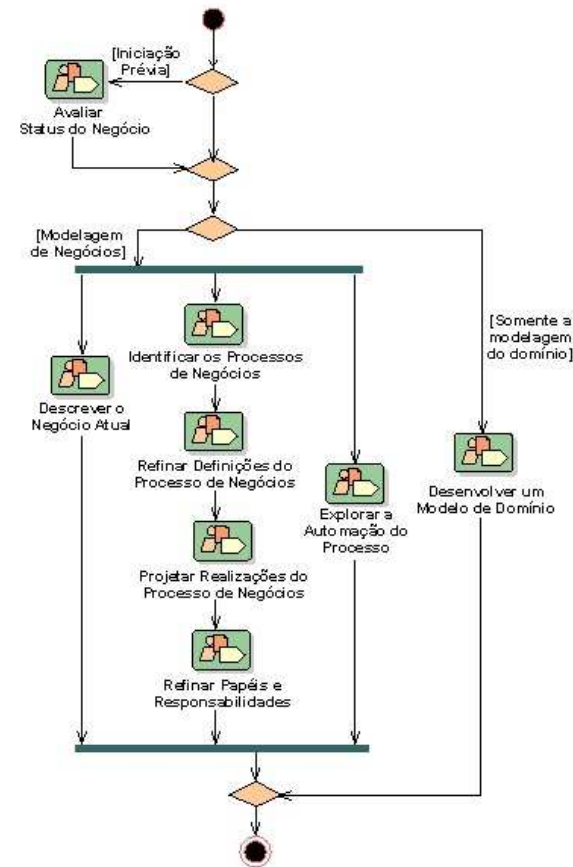
### Desenvolvimento de Comércio Eletrônico

### Escopo da Modelagem de Negócios

### Modelagem de Organizações de Grande Porte

### Padrões de Negócios

## Fluxo de Trabalho



# 1 - A disciplina de **Modelagem de Negócios**

## **Atividade em grupo: Disciplina Modelagem de Negócios**

- Por que modelar negócios?
- Como se relaciona com outras disciplinas
- Técnicas da Engenharia de Software para modelar negócios.
- Cenários da modelagem de negócio (organograma, modelagem de domínio, um negócio muitos sistemas ...)
- Explicar o papel dos trabalhadores e as suas respectivas atividades.
- Explicar o papel dos Trabalhadores e os respectivos artefatos gerados.
- Fluxo (diagrama de atividades) – Explicar o processo.
- Como modelar negócio de desenvolvimento de software
- Explicar os conceitos: arquitetura de negócios; custo baseado em atividades; desenvolvimento de comércio eletrônico; escopo da modelagem de negócios; modelagem de organizações de grande porte e padrões de negócios.
- Força nas fases do RUP
- Suporte de ferramenta
- Conclusão/Resumo.

### **Fontes de Consulta:**

KRUCHTEN, Philippe. **Introdução ao RUP - Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003. (disponível na biblioteca)  
<http://www.wthree.com/rup/portugues/index.htm>

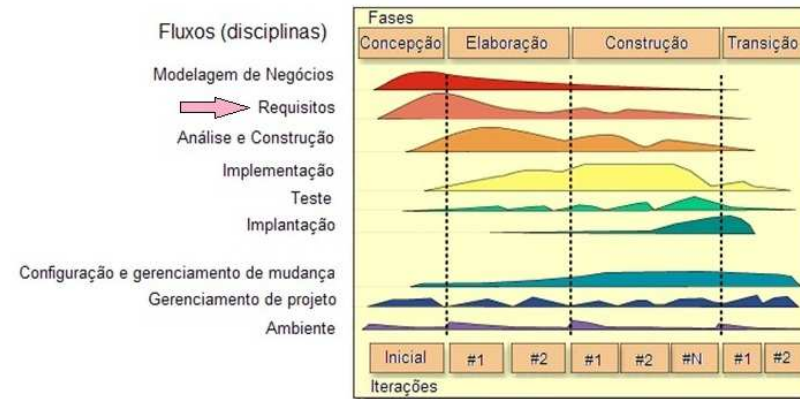
### **Apresentação:**

Apresentação com PowerPoint. Máximo 30 Slides e mínimo 17 Slides. Padrão 7 x 7. Tempo de apresentação 45 a 55 minutos.  
Aulas no laboratório serão usadas para preparação dos trabalhos.

# A disciplina de Requisitos

## A disciplina de Requisitos consiste basicamente em:

- Manter concordância entre clientes e desenvolvedores.
- Compreender melhor os requisitos do sistema.
- Delimitar o sistema.
- Estabelecer as bases para as iterações.
- Fornecer com mais detalhes a estimativa de custo do sistema.
- Fornecer uma estimativa de tempo para desenvolvimento do sistema.
- Definir uma interface de usuário inicial do sistema.



## PESSOAS:

Nessa disciplina, o **analista de sistemas** coordena o processo de identificação dos requisitos, a modelagem dos casos de uso e delimita as principais funcionalidades do sistema.

*"O arquiteto ideal deve ser uma pessoa erudita, um matemático, familiarizado com estudos históricos, um estudioso aplicado de filosofia, conhecedor de música, que não desconheça medicina, detentor de saber jurídico e familiarizado com astronomia e cálculos astronômicos." - Vitruvius, há aproximadamente 25 anos a.C.*

No contexto da Engenharia de software esse profissional (o **arquiteto de software**) deve ter grande conhecimento geral, maturidade, visão ampla e experiência para a identificar problemas, deve se expressar com sensatez, conhecer o domínio do problema e os requisitos, ser um líder nato, ter boa comunicação, ser proativo e focar sempre nos resultados. São algumas atividades do arquiteto: priorizar os casos de uso, construir e avaliar provas de conceito arquitetural, desenvolver guias de programação, desenvolver guias de design, fazer a análise arquitetural.

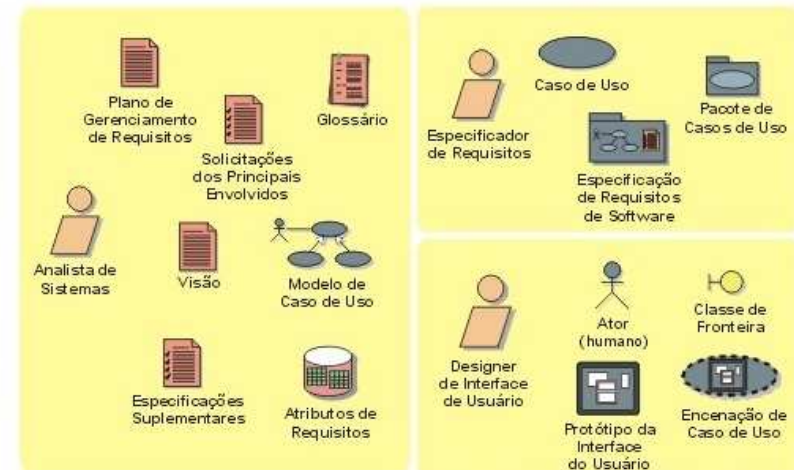
# A disciplina de Requisitos

## Requisitos

Trabalhadores (Pessoas)



Visão Geral das Atividades



Visão Geral dos Requisitos

Disponível em: <http://www.wthreex.com/rup/portugues/index.htm>

Acessado em: Novembro de 2012

# A disciplina de Requisitos

## Conceitos no contexto da disciplina de Requisitos

### Design Centrado no Usuário.

Sobre esse conceito não há um consenso explícito, porém algumas diretrizes foram desenvolvidas (na IBM), para se aproximar o mais possível dessa abordagem, entre muitos, destacam-se: converse com os usuários; visite os locais de trabalho dos usuários; observe as atividades dos usuários (ações); se possível grave em vídeo os usuários trabalhando etc.

Outros conceitos

**Gerenciamento de Requisitos.**

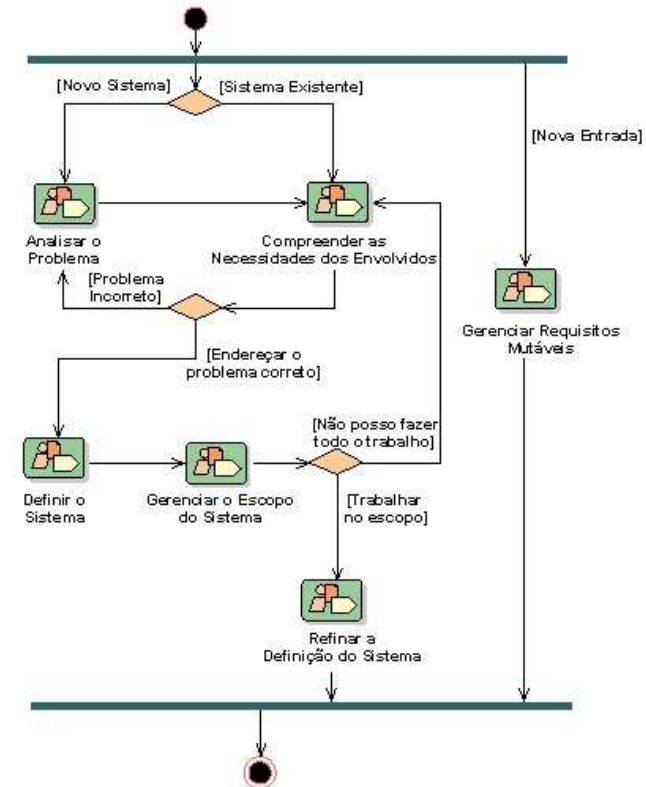
**Rastreabilidade.**

**Requisitos.**

**Tipos de Requisitos.**

**Visão de Casos de Uso.**

## Disciplina de Requisitos - Fluxo de Trabalho



Rational Unified Process: Visão Geral - RUP (documentação versão português)

Disponível em: <http://www.wthreex.com/rup/portugues/index.htm>

Acessado em: novembro de 2012.

## 2 - A disciplina de **Requisitos**

### **Atividade em grupo: Disciplina de Requisitos**

- O que é um requisito?
- Requisitos funcionais.
- Requisitos não funcionais.
- Solicitação X Necessidades.
- Como captar e administrar requisitos?
- Como projetar uma interface centrada no usuário?
- Como essa disciplina se relaciona com outras disciplinas?
- Fluxo de requisitos (diagrama de atividades)
- Trabalhadores em requisitos – suas atividades.
- Artefatos usados em requisitos.
- Força nas fases do RUP
- Suporte a ferramenta.
- Conclusão/Resumo.

### **Fontes de Consulta:**

KRUCHTEN, Philippe. **Introdução ao RUP - Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003. (disponível na biblioteca)  
<http://www.wthreex.com/rup/portugues/index.htm>

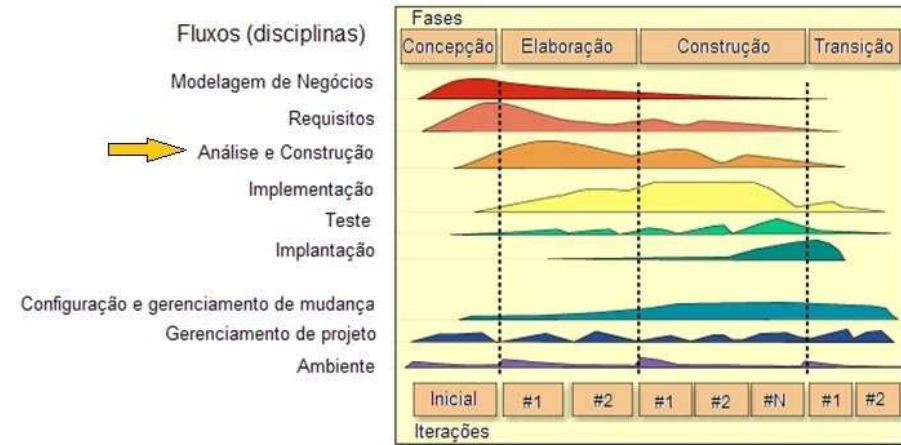
### **Apresentação:**

Apresentação com PowerPoint. Máximo 30 Slides e mínimo 17 Slides. Padrão 7 x 7. Tempo de apresentação 45 a 55 minutos.  
Aulas no laboratório serão usadas para preparação dos trabalhos.

# A disciplina de **Análise e Projeto**

**A disciplina de Análise e Design (Projeto) consiste basicamente em:**

- Fazer a transformação dos requisitos em um projeto.
- Aprimorar a arquitetura do sistema (tornando-a mais sofisticada).
- Fazer adaptações no projeto para a implementação.
- Relacionamento com Outras Disciplinas



**PESSOAS:**

Nessa disciplina o **arquiteto de software** coordena atividades e artefatos. Ele estabelece a estrutura geral de cada visão de arquitetura, como a decomposição desta visão e o agrupamento dos elementos do projeto e suas interfaces. O arquiteto deve ter uma visão global do projeto. O **revisor de arquitetura** conduz as revisões formais da arquitetura do sistema. Esse profissional possui o mesmo perfil do arquiteto, porém o seu enfoque é mais detalhado, contrabalançando riscos e custos e sempre que possível, deve levantar questões relevantes para discussão com outros atores do projeto. O **designer de banco de dados** define os objetos do banco, tais como: tabelas, índices, visões, restrições, triggers e procedures, todas essas informações são mantidas no Artefato: Modelo de Dados.

No slide seguinte, veja figura mostrando uma visão geral com trabalhadores , atividades e artefatos.



# A disciplina de **Análise e Projeto**

## Trabalhadores (Pessoas)

### Análise e Design



Visão Geral das atividades



Visão geral dos artefatos

Rational Unified Process: Visão Geral - RUP (documentação versão português)  
Disponível em: <http://www.wthreex.com/rup/portugues/index.htm>  
Acessado em: novembro de 2012.



# A disciplina de **Análise e Projeto**

## Alguns conceitos

### Arquitetura de Software

A arquitetura é um enfoque no projeto que concentra em alguns recursos específicos. O RUP, a arquitetura trata da estrutura dos componentes relevantes do sistema que podem interagir a partir de suas interfaces, com outros elementos.

### Outros conceitos

Visão de Implantação

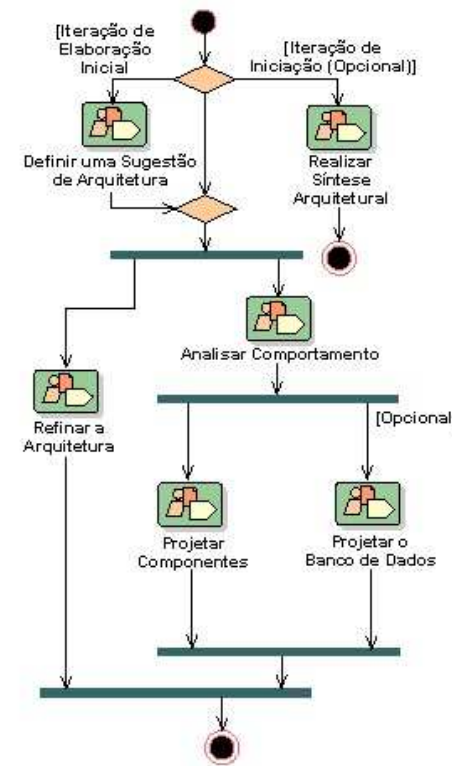
Visão de Processos

Divisão em camadas

Padrões de Distribuição

Padrões de Arquitetura WEB

## Fluxo de Trabalho - Análise e Design



Rational Unified Process: Visão Geral - RUP (documentação versão português)  
 Disponível em: <http://www.wthreex.com/rup/portugues/index.htm>  
 Acessado em: novembro de 2012.

## 3 - A disciplina de **Análise e Projeto**

### **Atividade em grupo: Disciplina de Análise e Projeto**

- Análise X Projeto (*Relação entre requisitos funcionais e não funcionais*)
- Até onde o projeto tem que ir?
- Trabalhadores e artefatos.
- O modelo de projeto.
- O modelo de análise.
- O papel das interfaces (nesse fluxo)
- Projetos baseados em componentes.
- Fluxo (diagrama de atividades)
- Força nas fases do RUP
- Suporte a ferramenta.
- Conclusão/Resumo.

### **Fontes de Consulta:**

KRUCHTEN, Philippe. **Introdução ao RUP - Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003. (disponível na biblioteca)

<http://www.wthreex.com/rup/portugues/index.htm>

### **Apresentação:**

Apresentação com PowerPoint. Máximo 30 Slides e mínimo 17 Slides. Padrão 7 x 7. Tempo de apresentação 45 a 55 minutos.

Aulas no laboratório serão usadas para preparação dos trabalhos.

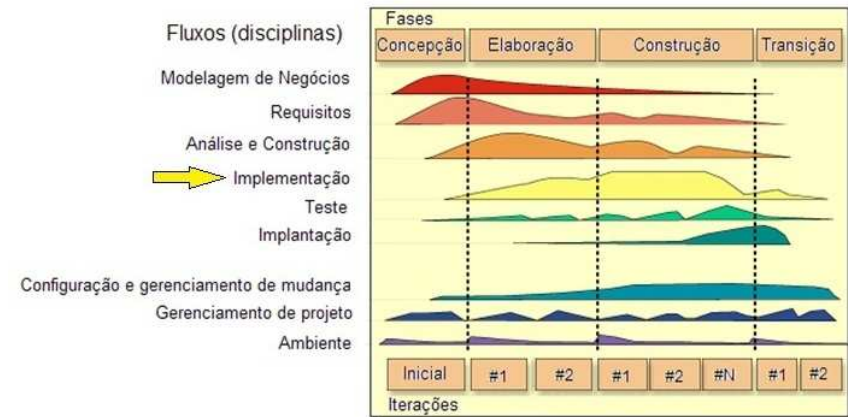
# A disciplina de **Implementação**

**A disciplina de implementação consiste basicamente em:**

- Realizar a Implementação de classes e objetos (arquivos-fonte, binários, executáveis e outros).
- Realizar os testes dos componentes desenvolvidos como unidades.
- Fazer a integração dos resultados das implementações ao sistema como um todo.

*Obs:*

*A disciplina Implementação se limita aos testes unidade. Os demais testes (de sistema e integração) são descritos na disciplina Teste.*

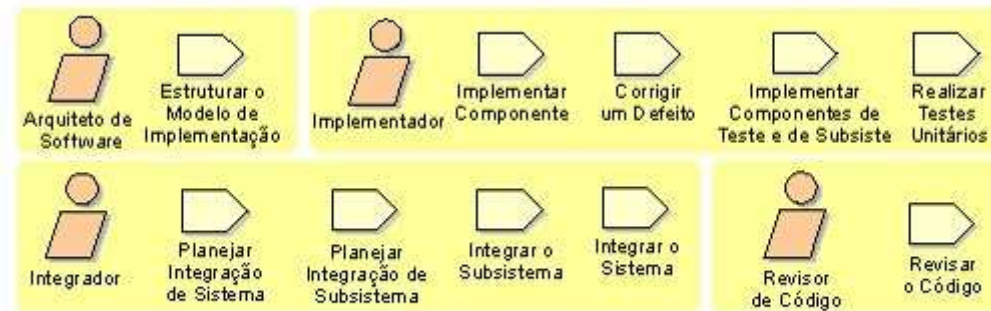


## PESSOAS

O **Implementador** é responsável construir e testar os componentes de software de acordo com as especificações adotados para o projeto. Se necessário o implementador deve desenvolver e testar os componentes necessários para teste e implementação do sistema. O **Arquiteto de Software** tem como atividades principais: a priorização dos casos de uso, realizar a análise arquitetural, estruturar o modelo de implementação, desenvolver guias de programação, fazer a descrição da arquitetura em tempo de execução. O principal artefato gerado por este trabalhador é o modelo de implementação. O papel dos **integradores** é combinar os componentes e subsistema para criar os builds do sistema. O **Revisor de Código** é responsável por conduzir as sucessivas revisões garantindo a qualidade do código.

# A disciplina de **Implementação**

Trabalhadores (Pessoas)



Implementação: Visão Geral da Atividade



Implementação: Visão Geral de Artefatos

Rational Unified Process: Visão Geral - RUP (documentação versão português)  
Disponível em: <http://www.wthreex.com/rup/portugues/index.htm>  
Acessado em: novembro de 2012.

# A disciplina de **Implementação**

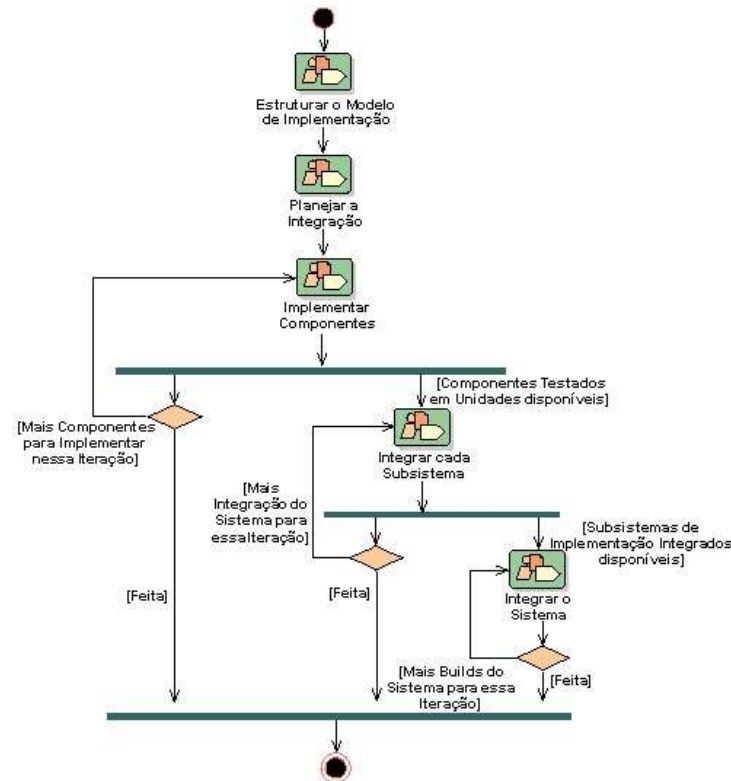
## Conceitos no contexto da disciplina de Implementação

### Build

Trata-se de uma versão operacional do sistema, demonstrando as capacidades oferecidas por esse, como produto final. Os builds são construídos dentro do processo iterativo, são na verdade o produto das tentativas contínuas de demonstrar as novas funcionalidades do sistema, recentemente implementadas.

### Outro conceitos

- Espaços de Trabalho de Desenvolvimento e Integração
- Integração do Software
- Mapeamento para Código a partir do Design.
- Stubs
- Teste do Desenvolvedor
- Visão da Implementação



Rational Unified Process: Visão Geral - RUP (documentação versão português)

Disponível em: <http://www.wthreex.com/rup/portugues/index.htm>

Acessado em: novembro de 2012.

## 4 - A disciplina de **Implementação**

### **Atividade em grupo: Disciplina de Implementação**

- Proposta desse fluxo.
- Construções.
- Integrações.
- Protótipos.
- Tipos de protótipos.
- Trabalhadores: Atividade e Artefatos.
- Conceitos (espaços de Trabalho; Integração do Software; mapeamento Design -> Código; stubs; testes do desenvolvedor; visão da implementação)
- Fluxo (diagrama de atividades)
- Suporte a ferramenta.
- Força nas fases do RUP
- Conclusão/Resumo.

### **Fontes de Consulta:**

KRUCHTEN, Philippe. **Introdução ao RUP - Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003. (disponível na biblioteca)

<http://www.wthreex.com/rup/portugues/index.htm>

### **Apresentação:**

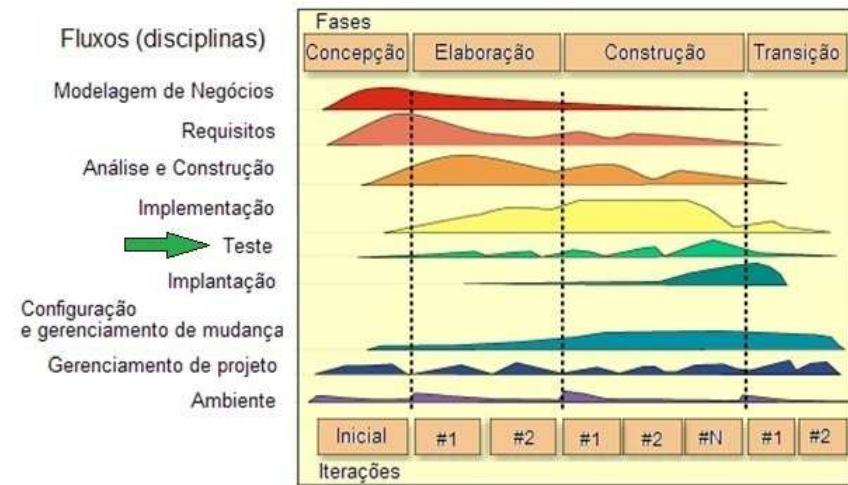
Apresentação com PowerPoint. Máximo 30 Slides e mínimo 17 Slides. Padrão 7 x 7. Tempo de apresentação 45 a 55 minutos.

Aulas no laboratório serão usadas para preparação dos trabalhos.

# A disciplina de **Teste**

**A disciplina de Testes consiste basicamente em:**

- Enfatizar principalmente a questão da qualidade do produto.
- Verificar se as funções do software estão de acordo com o projeto.
- Localizar e documentar os possíveis defeitos do software.
- Observar e apontar, de uma maneira abrangente, as questões relativas à qualidade do software.
- Localizar e expor os pontos fracos do software.
- Delegar pessoas qualificadas na área de teste e o uso de uma metodologia clara.



## PESSOAS

O **Gerente de Testes** é o responsável pelo êxito dos testes. O Gerente de Testes é o responsável pela qualidade, planejamento e gerenciamento dos testes. O **Analista de Teste** define e monitora os testes, esse monitoramento evidentemente está relacionado à qualidade final obtida, algumas funções do Analista de Teste também são realizadas pelo Designer de Teste. O **Designer de Teste** normalmente é o responsável em identificar as ferramentas e as técnicas adequadas para implementar os testes, esse papel é também conhecido como: Arquiteto de Testes ou Especialista em Automatização de Testes. O **Testador** é responsável em implementar efetivamente os testes, registrando os resultados e analisando os erros.



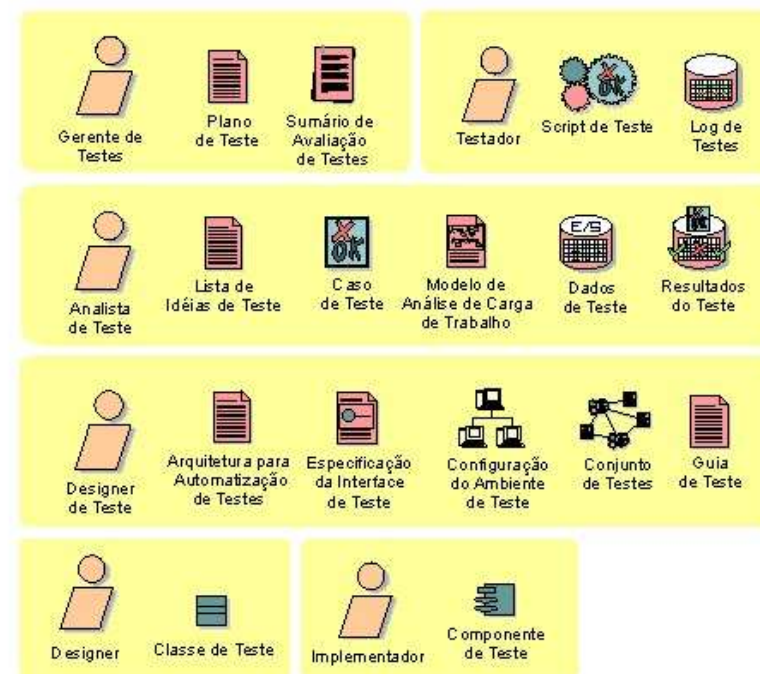
# A disciplina de **Teste**

## Trabalhadores (Pessoas)



## Visão Geral da Atividade

Rational Unified Process: Visão Geral - RUP (documentação versão português)  
 Disponível em: <http://www.wthreex.com/rup/portugues/index.htm>  
 Acessado em: novembro de 2012.



## Visão Geral de Artefatos



# A disciplina de **Teste**

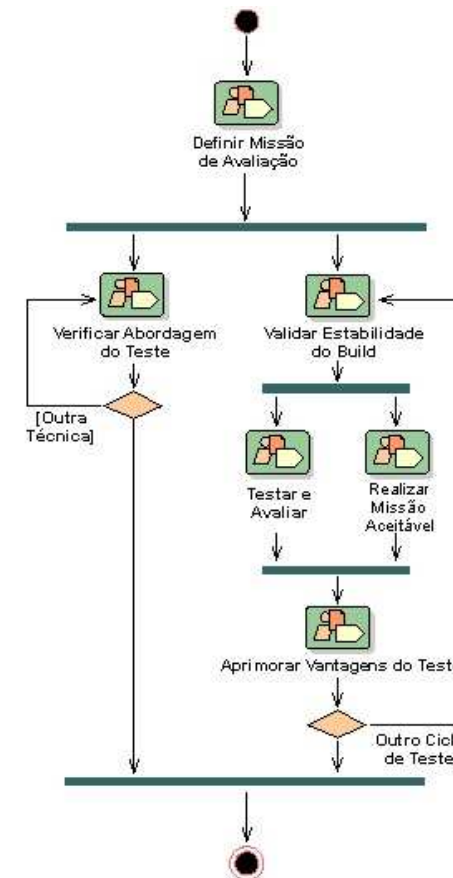
## Conceitos no contexto da disciplina de Implementação

### Testes de Aceitação

É a última fase do processo de teste antes da implantação do software. A idéia deste é certificar que o software está pronto para uso. O teste de aceitação pode ter três abordagens: a primeira é a **Formal**, que é um processo gerenciado, planejado e projetado como um teste de sistema; a segunda, a **Informal**, não há tanto rigor e é mais subjetiva do que o tipo formal. a última, o teste **Beta** é ainda mais informal e com menor controle do que a anterior e os testes são de inteira responsabilidade do testador.

### Outros Conceitos

- Principais Medidas de Teste (*baseada em requisitos e em códigos*).
- Teste de Desempenho (*avaliação de desempenho; teste de carga e teste de stress*)
- Estágios de Teste (*teste do Desenvolvedor; teste Independente e dos envolvidos; teste Unitário; teste de Integração; teste do Sistema; teste de Aceitação*)
- Caixa branca versus Caixa preta
- Teste de usabilidade



Rational Unified Process: Visão Geral - RUP (documentação versão português)  
 Disponível em: <http://www.wthreex.com/rup/portugues/index.htm>  
 Acessado em: novembro de 2012.

## 5 - A disciplina de **Teste**

### **Atividade em grupo: Disciplina de Teste**

- Proposta deste fluxo. Definir Qualidade no contexto desta disciplina (Testes).
- Conceitos (*principais medidas de teste, teste de desempenho, estágios de teste, caixa branca versus Caixa preta e teste de usabilidade*)
- Propriedades de qualidade do produto.
- O teste do ciclo de vida iterativo.
- Dimensões do teste.
- Dimensão da qualidade. Estágios do teste.
- Tipos de teste.
- Teste de regressão.
- Modelo de teste.
- Trabalhadores: Atividades e Artefatos. Fluxo (diagrama de atividades)
- Suporte a ferramenta. Força nas fases do RUP
- Conclusão/Resumo.

### **Fontes de Consulta:**

KRUCHTEN, Philippe. **Introdução ao RUP - Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003. (disponível na biblioteca)

<http://www.wthreex.com/rup/portugues/index.htm>

### **Apresentação:**

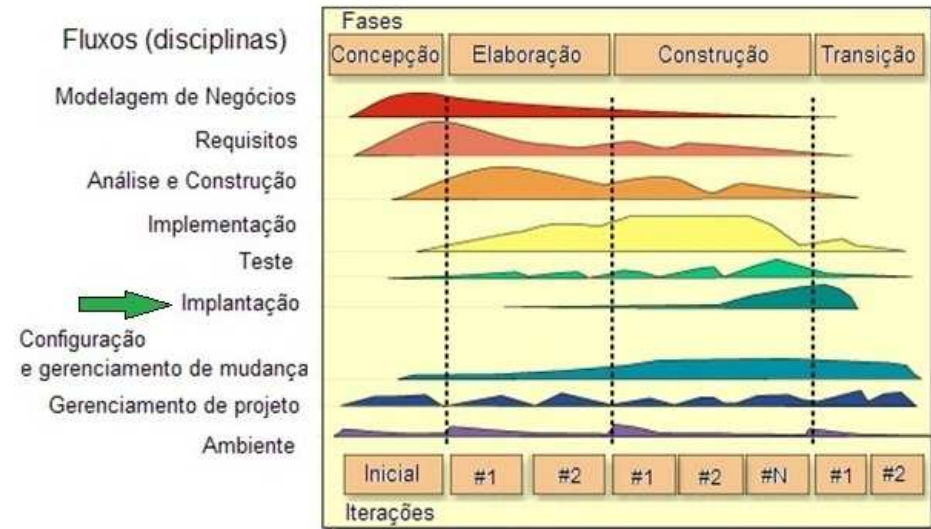
Apresentação com PowerPoint. Máximo 30 Slides e mínimo 17 Slides. Padrão 7 x 7. Tempo de apresentação 45 a 55 minutos.

Aulas no laboratório serão usadas para preparação dos trabalhos.

# A disciplina de **Implantação**

## A disciplina de Implantação consiste basicamente em:

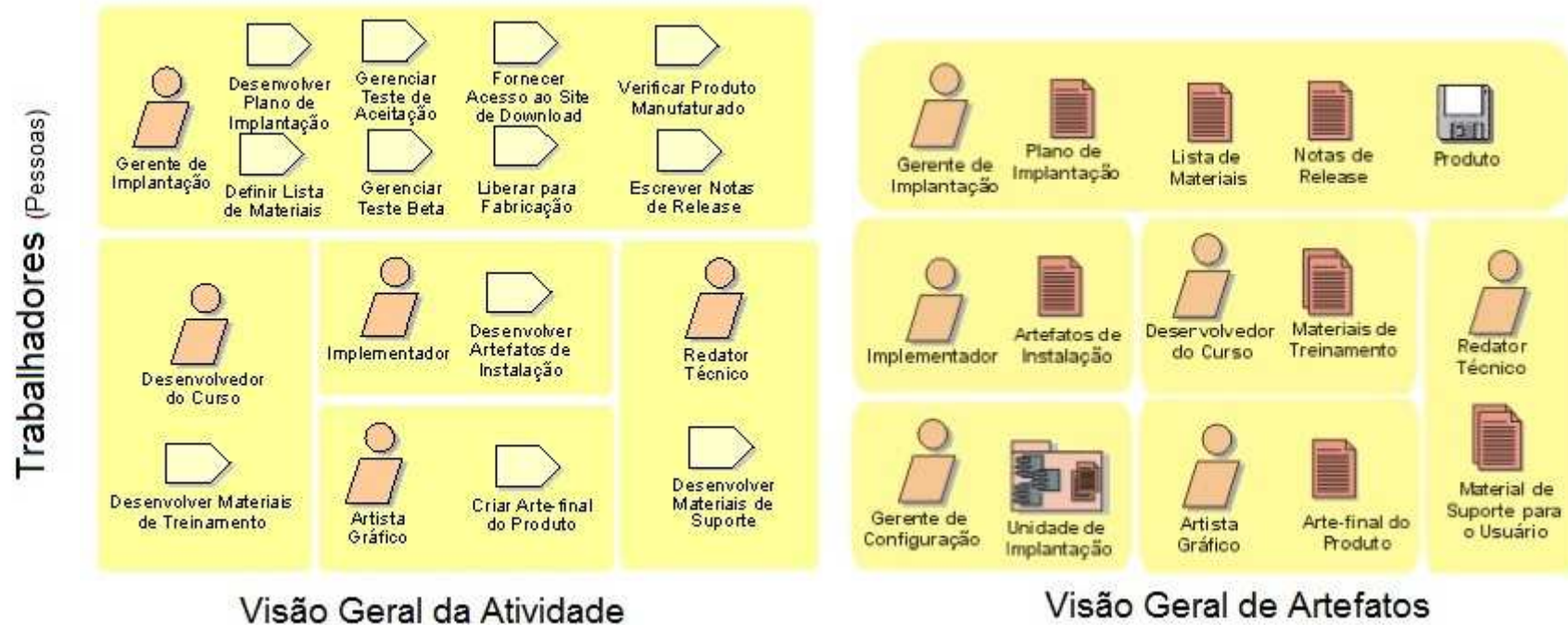
- Descrever as atividades que garantirão a disponibilização do produto para os usuários finais.
- Descrever os três modos de instalação do produto: instalação personalizada; produto em uma forma "compacta"; cesso ao produto por meio da Internet.
- Realizar o teste Beta (no local de desenvolvimento) antes de entregá-lo ao cliente.
- O pico (força) desta disciplina ocorre de maneira significativa na Fase de Transição.



## PESSOAS:

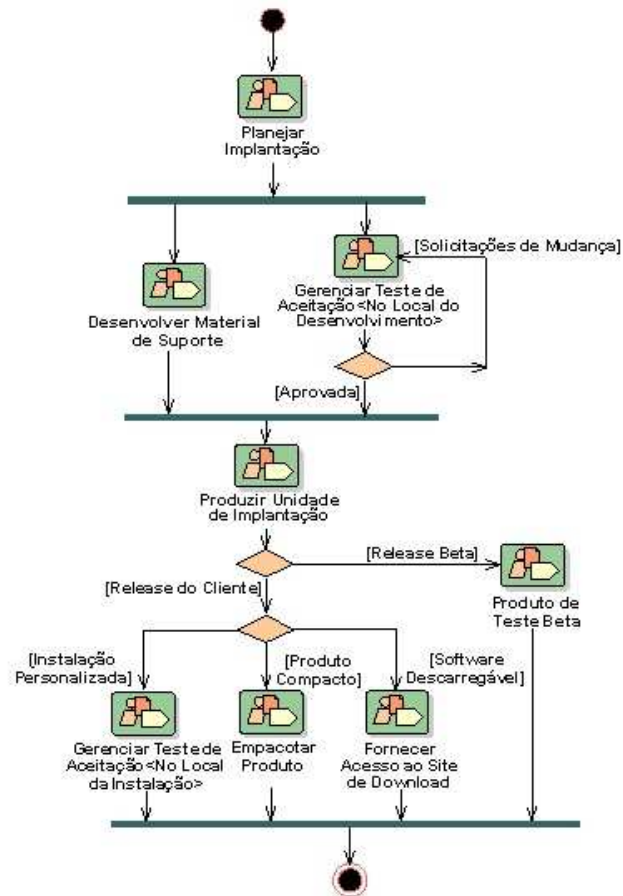
O **gerente de implantação** tem como função principal planejar a transição do produto para a comunidade de usuários. O **desenvolvedor do curso** tem como atividade principal o desenvolvimento do material de treinamento, ele produzirá apresentações, tutoriais etc. enfim, materiais para treinamento e suporte para os usuários finais. O **implementador** será responsável na implementação de componentes finais para implantação, tais como drivers e stubs para integração do sistema com subsistemas legados, garantindo a implantação correta do mesmo. A arte final do produto é desenvolvida pelo **Artista-Gráfico**. O material de suporte para o usuário final é escrito pelo Redator Técnico.

# A disciplina de **Implantação**



Rational Unified Process: Visão Geral - RUP (documentação versão português)  
 Disponível em: <http://www.wthree.com/rup/portugues/index.htm>  
 Acessado em: novembro de 2012.

# A disciplina de **Implantação**



Rational Unified Process: Visão Geral - RUP (documentação versão português)  
 Disponível em: <http://www.wthreex.com/rup/portugues/index.htm>  
 Acessado em: novembro de 2012.

## 6 - A disciplina de **Implantação**

### **Atividade em grupo: Disciplina de Implantação**

- Finalidade da disciplina.
- Definir implantação, no contexto desta disciplina.
- Definir o papel de cada um dos trabalhadores no que se refere as atividades realizadas.
- Definir o papel de cada um dos trabalhadores no que se refere aos artefatos gerados.
- Explicar a importância do teste nesta disciplina.
- Explicar o fluxo de trabalho desta disciplina.
- Explicar como essa disciplina se relaciona com a disciplina Requisitos.
- Explicar como essa disciplina se relaciona com a disciplina Testes.
- Explicar como essa disciplina se relaciona com a disciplina Gerenciamento e Configuração de Mudanças.
- Explicar como essa disciplina se relaciona com a disciplina de Ambiente.
- Suporte a ferramenta. Força nas fases do RUP
- Conclusão/Resumo.

### **Fontes de Consulta:**

KRUCHTEN, Philippe. **Introdução ao RUP - Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003. (disponível na biblioteca)

<http://www.wthreeex.com/rup/portugues/index.htm>

### **Apresentação:**

Apresentação com PowerPoint. Máximo 30 Slides e mínimo 17 Slides. Padrão 7 x 7. Tempo de apresentação 45 a 55 minutos.

Aulas no laboratório serão usadas para preparação dos trabalhos.



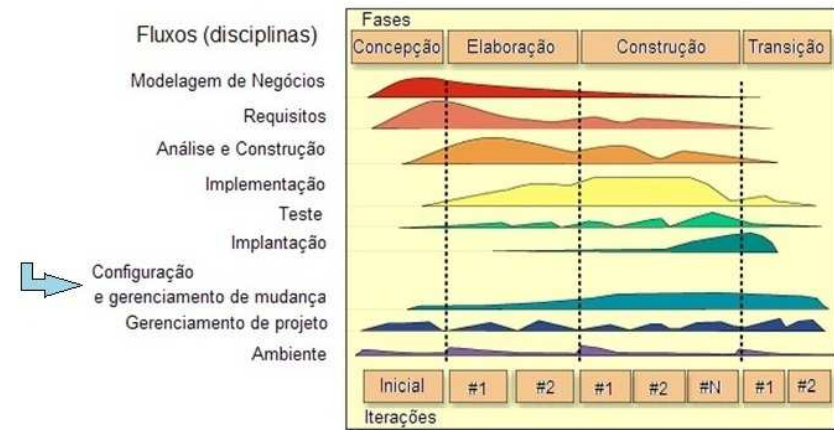
## A disciplina de Gerenciamento e Configuração de Mudança

**A disciplina de Gerenciamento e Configuração de Mudança consiste basicamente em:**

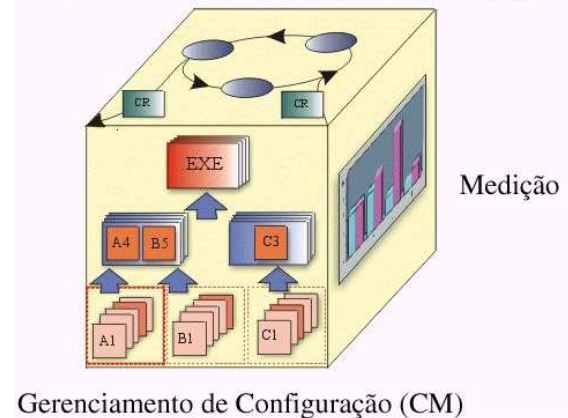
- Controlar mudanças feitas nos artefatos de um projeto e manter a integridade destes.
- identificar os itens de configuração e garantir a restrição de mudanças nestes.
- Auditar as mudanças feitas nesses itens.
- Garantir a integridade dos inúmeros artefatos pelas pessoas que trabalham no projeto (como controle de versão).
- Garantir a integridade entre os releases (quando pessoas trabalham em releases diferentes)

**A disciplina de Gerenciamento e Configuração de Mudança pode ser vista a partir de um Cubo denominado Cubo CCM:**

A face CM corresponde ao gerenciamento de configuração, está relacionada a estrutura do produto. A face Estado e Medida se relaciona ao controle do projeto (gráficos) e finalmente a face do Gerenciamento de Solicitação de Mudança lida com o processo da solicitação de uma mudança que consiste também na documentação desta (correção de defeitos, novas implementações etc.)



Gerenciamento de Solicitações de Mudança (CRM)



## A disciplina de Gerenciamento e Configuração de Mudança

### PESSOAS:

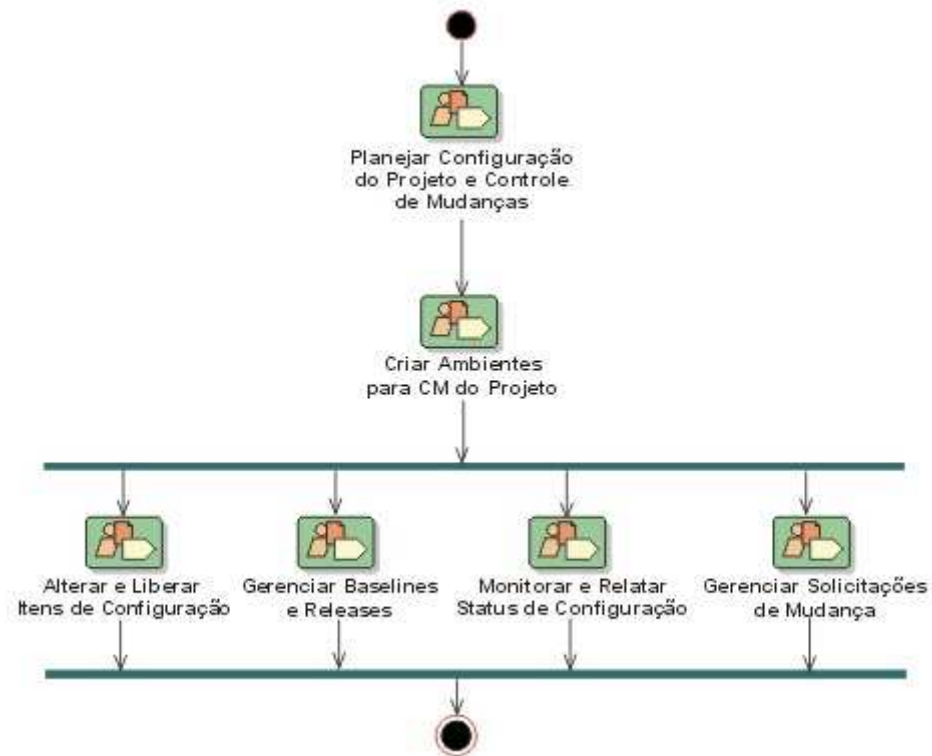
A disponibilização do ambiente e infra-estrutura geral de gerenciamento e configuração é feita pelo **gerente de configuração**. Este trabalhador assegura a facilidade das revisões do produto, este trabalhador também redige o Plano CM. A supervisão do processo de controle de mudanças é feita pelo **Gerente de Controle de Mudança**, este papel também pode ser feito por um comitê que inclui usuários finais, clientes e pessoal técnico. O principal artefato gerado nesta atividade é a Solicitação de Mudança. Os **implementadores** são responsáveis em liberar os componentes já testados em um espaço compartilhado de trabalho, os **integradores**, por sua vez, agrupam esses componentes criando build. **Qualquer Papel**, neste contexto, se refere a todos os papéis identificados no RUP a partir dos privilégios de acesso. Estes, podem fazer o 'check-in' e 'check-out' nos artefato relacionados ao produto que estejam trabalhando. Este também podem enviar solicitações de mudança e realizá-las, desde que sejam proprietários.



Rational Unified Process: Visão Geral - RUP (documentação versão português)  
Disponível em: <http://www.wthreex.com/rup/portugues/index.htm>  
Acessado em: novembro de 2012.



## A disciplina de Gerenciamento e Configuração de Mudança



Rational Unified Process: Visão Geral - RUP (documentação versão português)  
Disponível em: <http://www.wthreex.com/rup/portugues/index.htm>  
Acessado em: novembro de 2012.

## 7 - A disciplina de **Gerenciamento e Configuração de Mudança**

### **Atividade em grupo: Disciplina de Gerenciamento e Configuração de Mudança**

- Proposta deste fluxo.
- Explicar o cubo CCM.
- Definir Gerenciamento de solicitação de mudanças.
- Definir e diferenciar estado e medida.
- Explicar a função dos Trabalhadores e suas Atividades no contexto desta disciplina.
- Explicar a relação dos Trabalhadores com os artefatos que produzem.
- Explicar o Fluxo (diagrama de atividades)
- Suporte a ferramenta.
- Força nas fases do RUP
- Conclusão/Resumo.

### **Fontes de Consulta:**

KRUCHTEN, Philippe. **Introdução ao RUP - Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003. (disponível na biblioteca)

<http://www.wthreex.com/rup/portugues/index.htm>

### **Apresentação:**

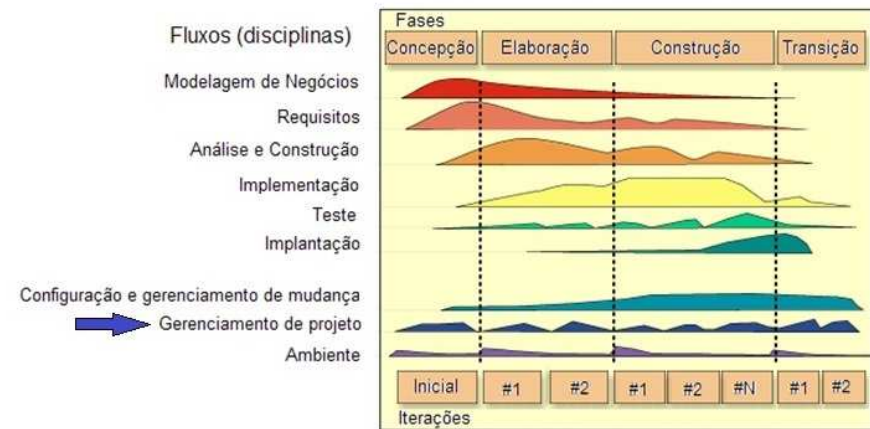
Apresentação com PowerPoint. Máximo 30 Slides e mínimo 17 Slides. Padrão 7 x 7. Tempo de apresentação 45 a 55 minutos.

Aulas no laboratório serão usadas para preparação dos trabalhos.

## A disciplina de Gerenciamento de Projeto

**A disciplina de Gerenciamento de Projeto consiste basicamente em:**

- Fornecer uma estrutura para gerenciar projetos de software.
- Fornecer diretrizes práticas para planejar, prover pessoal, executar e monitorar projetos.
- Fornecer uma estrutura para gerenciar riscos.



**Mas não abrange questões como:**

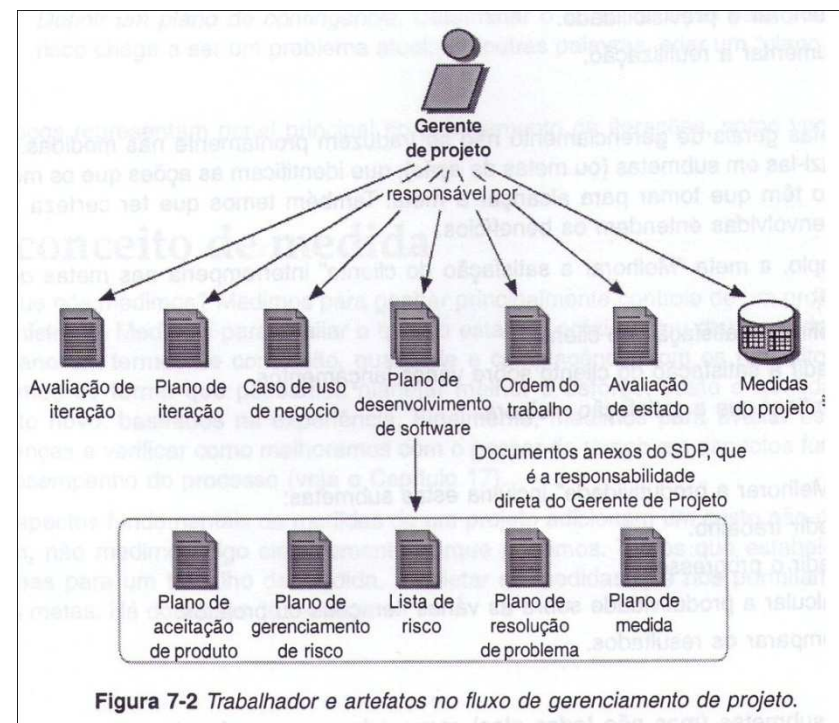
- Administrar pessoas (contratação, treinamento, etc.)
- Administrar orçamentos.
- Administrar contratos com fornecedores (provedores) e clientes.

# A disciplina de Gerenciamento de Projeto

## Pessoas (Workers)

### Gerente de Projeto

Produz o plano de desenvolvimento de software: plano de aceitação, plano de gerenciamento de risco, plano de resolução de problema e plano de medida.



# A disciplina de Gerenciamento de Projeto

## Fluxo

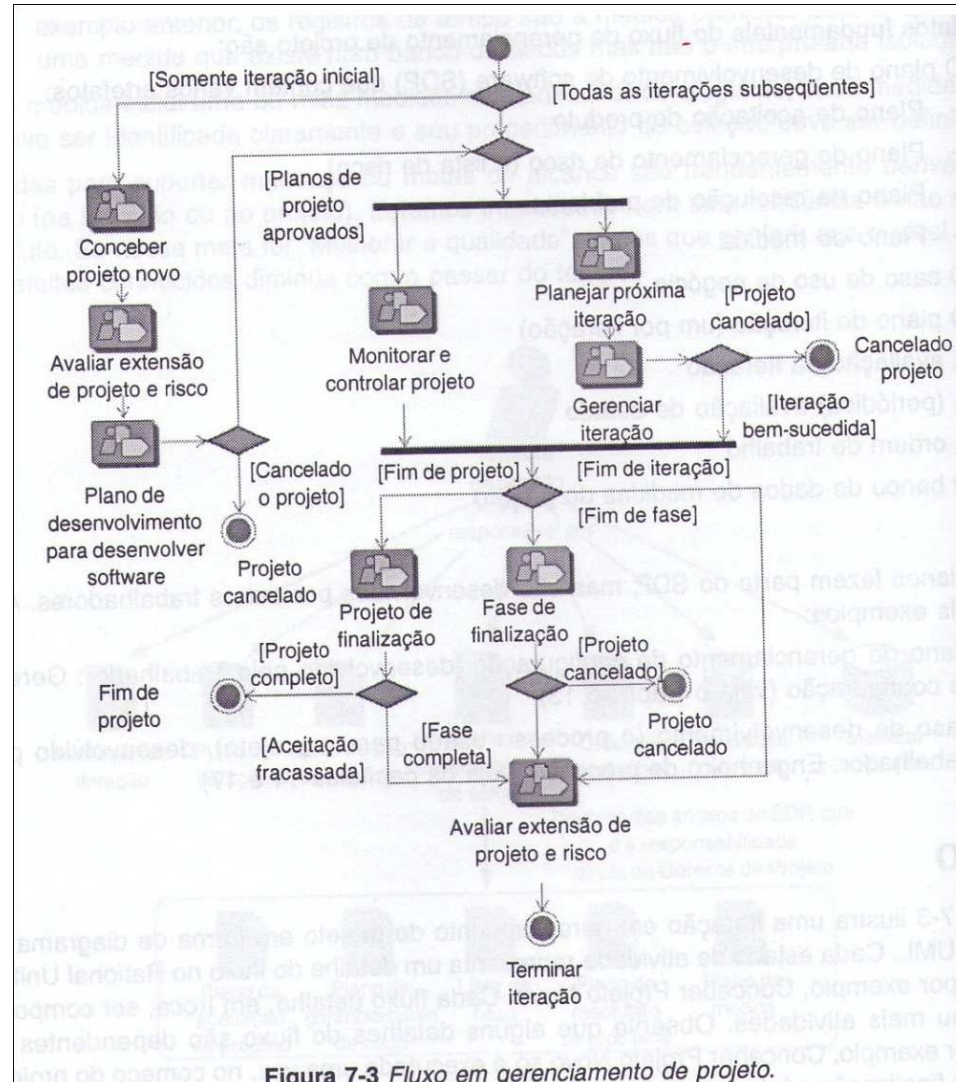


Figura 7-3 Fluxo em gerenciamento de projeto.

## 8 - A disciplina de **Gerenciamento de Projeto**

### **Atividade em grupo: Disciplina de Gerenciamento de Projeto**

- Explicar como é gerenciado o processo Iterativo (O plano de fase e o plano de iteração).
- Conceituar Risco.
- Conceituar Medida.
- Explicar a função dos principais trabalhadores desta disciplina.
- Explicar os principais artefatos gerados pelos trabalhadores desta disciplina.
- Explicar o fluxo em gerenciamento de projeto desta disciplina (diagrama de atividades).
- Explicar como é construído um plano de repetição:
  - Iteração na fase de Elaboração; Iteração na fase de construção; Iteração na fase de Transição;
- Força nas fases do RUP
- Conclusão/Resumo.

### **Fontes de Consulta:**

KRUCHTEN, Philippe. **Introdução ao RUP - Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003. (disponível na biblioteca)

<http://www.wthreex.com/rup/portugues/index.htm>

### **Apresentação:**

Apresentação com PowerPoint. Máximo 30 Slides e mínimo 17 Slides. Padrão 7 x 7. Tempo de apresentação 45 a 55 minutos.

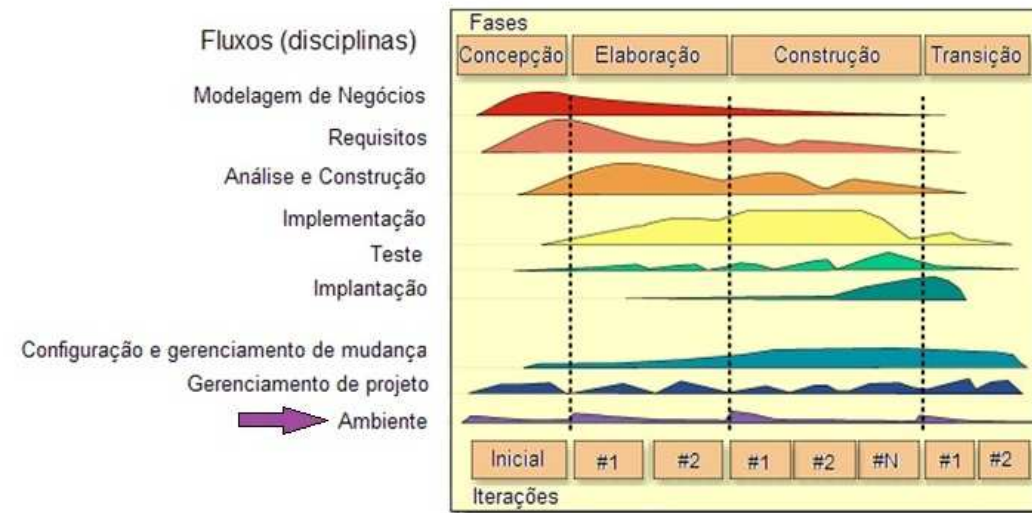
Aulas no laboratório serão usadas para preparação dos trabalhos.



## A disciplina de Ambiente

**A disciplina de Ambiente consiste basicamente em:**

- Selecionar as ferramentas (e aquisição)
- Configurar as ferramentas adquiridas de forma a harmonizar a organização.
- Configurar e melhorar os processos.
- Prover os serviços técnicos para suportar o processo: infra-estrutura da tecnologia da informação, administração de contas, backup e assim por diante. (Kruchten, 2003, página 187).

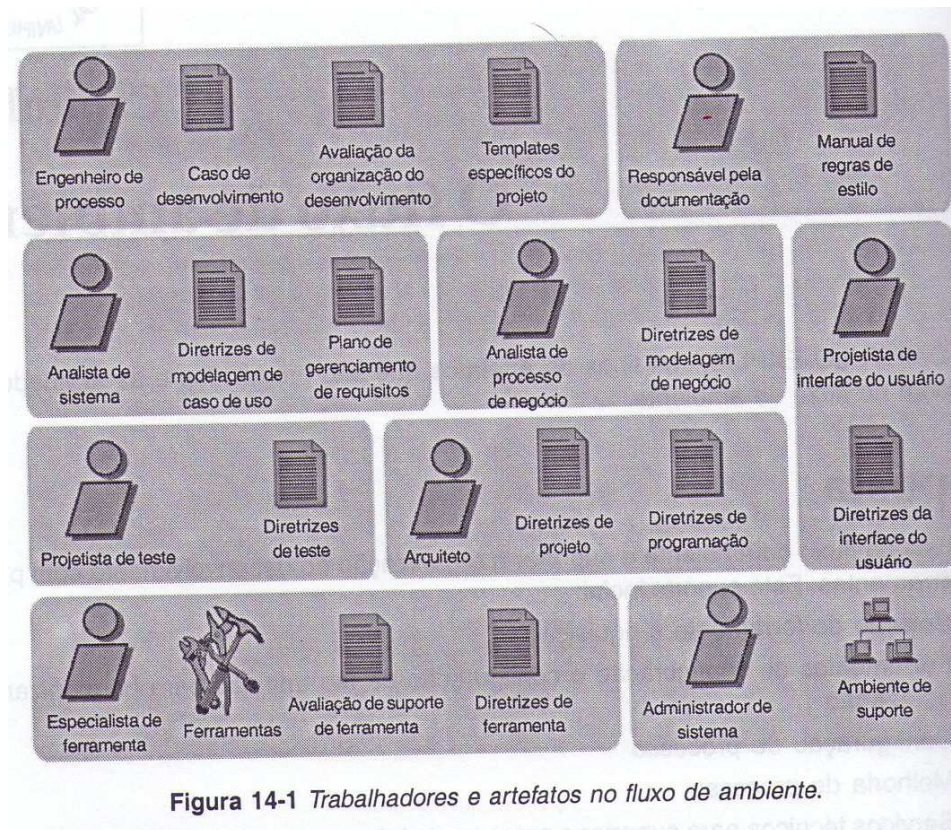


### Trabalhadores e artefatos

- Engenheiro de Processo
- Analista de Sistemas
- Projetista de teste
- Arquiteto.
- Analista de Processo de Negócios
- Projetista de interface de usuário
- Administrador de sistema.

## A disciplina de Ambiente

### Trabalhadores e artefatos





## A disciplina de Ambiente

### Fluxo

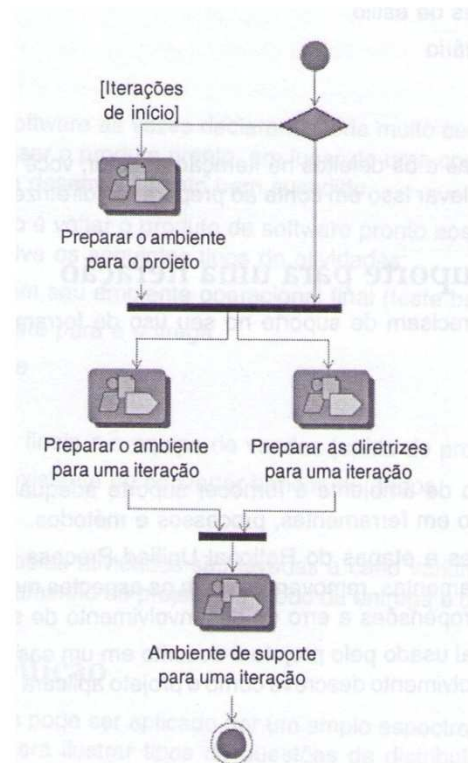


Figura 14-2 Um fluxo de ambiente.

## 9 - A disciplina de **Ambiente**

### **Atividade em grupo: Disciplina de Ambiente**

- Explicar a proposta desta disciplina.
- Explicar os trabalhadores e respectivos artefatos produzidos.
- Explicar o fluxo do processo.
- Explicar o propósito de preparar o ambiente para o projeto.
- Explicar o propósito de preparar o ambiente para uma iteração.
- Explicar o propósito de preparar as diretrizes para uma iteração.
- Força nas fases do RUP
- Conclusão/Resumo (Resumir, explicar a meta desta disciplina e porque automatizar algumas tarefas).

### **Fontes de Consulta:**

KRUCHTEN, Philippe. **Introdução ao RUP - Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003. (disponível na biblioteca)

<http://www.wthree.com/rup/portugues/index.htm>

### **Apresentação:**

Apresentação com PowerPoint. Máximo 30 Slides e mínimo 17 Slides. Padrão 7 x 7. Tempo de apresentação 45 a 55 minutos.

Aulas no laboratório serão usadas para preparação dos trabalhos.

## Referências Principais

FILHO, Wilson de Pádua. **Engenharia de Software – Fundamentos, Métodos e Padrões**. LTC, 2001.

KRUCHTEN, Philippe. **Introdução ao RUP - Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003.

PRESSMAN, Roger S. **Engenharia de Software**. 6rd ed. McGraw-Hill, 2002.

SOMMERVILLE, Ian. **Engenharia de Software**. 8ª edição, São Paulo : Pearson Addison-Wesley, 2007.

## Referências Relevantes

BEZERRA, E. **Princípios de Análise e projeto de sistemas com UML**. Rio de Janeiro: Campus, 2002.

MELO, Ana Cristina. **Desenvolvendo aplicações com UML**. Rio de Janeiro: Brasport, 2002.

TOM, PENDER. **UML: a Bíblia**. Rio de Janeiro : Elsevier , 2004.

GANE, CHRIS. **Análise Estruturada de Sistemas**. Rio de Janeiro: LTC – Livros Técnicos e Científicos Editora S/A., 1983.