

UNIVERSIDADE LUTERANA DO BRASIL

ULBRA – *CAMPUS* GUAÍBA

CURSO DE SISTEMAS DE INFORMAÇÃO



**TRABALHO DE CONCLUSÃO DE CURSO I**  
**GESTÃO DO CONHECIMENTO EM UM**  
**PROCESSO DE DESENVOLVIMENTO DE**  
**SOFTWARE**

CARLOS EDUARDO SPOLAVORI MARTINS

Anderson Yanzer  
Orientador

Guaíba, Julho de 2008.

## **DADOS DE IDENTIFICAÇÃO**

Acadêmico(a): 001102358-9

E-mail: cspolavori@hotmail.com

Professor(a) Orientador(a): Anderson Yanzer

E-mail: andersonyanzer@gmail.com

Título do Projeto: GESTÃO DO CONHECIMENTO EM UM PROCESSO DE  
DESENVOLVIMENTO DE SOFTWARE

Período de realização: 1º sem. 2008

## SUMÁRIO

<b>1</b>	<b>DEFINIÇÃO DO TEMA</b> .....	<b>5</b>
1.1	Tema.....	5
1.2	Delimitação do Tema .....	5
<b>2</b>	<b>PROBLEMA DE PESQUISA</b> .....	<b>6</b>
<b>3</b>	<b>HIPÓTESES DE SOLUÇÃO</b> .....	<b>6</b>
<b>4</b>	<b>OBJETIVOS</b> .....	<b>7</b>
4.1	Objetivos específicos: .....	7
<b>5</b>	<b>JUSTIFICATIVA</b> .....	<b>7</b>
<b>6</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b> .....	<b>8</b>
6.1	Conhecimento e Gestão o Conhecimento .....	8
6.2	Metodologias de GC.....	10
6.3	Técnicas e Ferramentas de GC.....	12
6.4	Desenvolvimento de Software .....	14
<b>7</b>	<b>DESCRIÇÃO DO PROCESSO ATUAL DE DESENVOLVIMENTO DE SOFTWARE</b> .....	<b>16</b>
7.1	Fluxos Iniciais.....	16
7.1.1	<i>Solicitação do Cliente</i> .....	16
7.1.2	<i>Análise da Task List</i> .....	18
7.2	Tipos de Projetos .....	20
7.2.1	<i>Novos - PPN</i> .....	20
7.2.1.1	Gerenciamento de Projetos .....	20
7.2.1.1.1	Planejar o Projeto .....	21
7.2.1.1.2	Planejar a Iteração .....	24
7.2.1.1.3	Gerenciar Iteração.....	25
7.2.1.1.4	Monitorar e Controlar o Projeto .....	26
7.2.1.1.5	Gerar Baseline .....	28
7.2.1.1.6	Fechar Projeto.....	28
7.2.1.2	Ambiente.....	29
7.2.1.3	Configuração e Mudanças (CVS) .....	30
7.2.1.4	Requisitos .....	30
7.2.1.5	Análise & Projeto.....	34
7.2.1.6	Implementação.....	37
7.2.1.7	Testes .....	39

7.2.1.8	Implantação:.....	43
<b>8</b>	<b>METODOLOGIA.....</b>	<b>45</b>
8.1	Pesquisa bibliográfica de Gestão do Conhecimento:.....	45
8.2	Pesquisa bibliográfica sobre Desenvolvimento de SW .....	45
8.3	Definição de Metodologias de GC para serem usadas como referência ...	45
8.4	Levantamento e descrição do processo de desenvolvimento de sw a ser utilizado neste projeto .....	46
8.5	Apontar e descrever técnicas, ferramentas ou metodologias que serão utilizadas para implantar GC no ciclo de DS .....	46
8.6	Experimento ou validação da proposta de GC que será utilizada no ciclo de DS .....	46
8.7	Elaboração de um relatório final com os resultados obtidos .....	46
<b>9</b>	<b>RESULTADOS .....</b>	<b>47</b>
<b>10</b>	<b>TRABALHO DE CONCLUSÃO DE CURSO - II.....</b>	<b>47</b>
<b>11</b>	<b>REFERÊNCIAS .....</b>	<b>48</b>

## **1 DEFINIÇÃO DO TEMA**

### **1.1 Tema**

Gestão do conhecimento em um processo de desenvolvimento de software não é exatamente um processo novo, mas ainda pouco utilizado pela maioria das empresas na área de TI.

A tecnologia da Informação vem se desenvolvendo de tal forma que surge nas empresas uma enorme necessidade de fazer uso destas tecnologias para o seu benefício, objetivando crescimento e vantagem competitiva. Pensando neste crescimento, as empresas da área de TI vêem o processo de desenvolvimento de software com auxílio da gestão do conhecimento como um precioso e valioso diferencial para se manter competitiva. A gestão do conhecimento atribui valores aos ativos intangíveis das organizações.

Mesclando o uso dos Sistemas de Informação ao processo da gestão do conhecimento, surgem ferramentas que visam auxiliar este processo, na criação de um melhor ambiente para o gerenciamento do conhecimento, criando, adquirindo, colaborando e compartilhando conhecimento. O apoio deste processo ao desenvolvimento de software, estabelece uma base sólida para projetos maduros e bem planejados, tornando-se um produto confiável e estável ajudando a manter o conhecimento gerado e adquirido no ambiente organizacional.

### **1.2 Delimitação do Tema**

Este trabalho irá analisar um ciclo de desenvolvimento de software baseado na metodologia RUP (*Rational Unified Process*), neste ciclo serão identificados pontos em que possam ser aplicadas técnicas de gestão do conhecimento para melhorar aspectos de aquisição, armazenamento e compartilhamento de conhecimento de interesse à equipe de desenvolvimento e clientes.

É inviável apresentar todos os passos propostos pela metodologia RUP, serão apresentados processos já definidos pelo modelo da empresa estudada, para desenvolver um software baseado nas fases propostas pela metodologia

como Iniciação, Elaboração, Construção e Transição, focando em pontos específicos para aquisição e gerenciamento do conhecimento.

Neste trabalho não será proposta melhoria ou alguma inovação na metodologia de desenvolvimento de sw apresentada, mas sim apontar em determinados pontos do processo onde poderemos adicionar aspectos de gestão do conhecimento para dinamizar e otimizar tais processos, afim de facilitar a comunicação e o acesso a informação pelos *stakeholders*.

## **2 PROBLEMA DE PESQUISA**

Atualmente a maioria das empresas tem a preocupação quanto à informação adquirida ou gerada em seu ambiente, mas como armazená-la ou gerencia-la? De tal maneira que todos possam usufruir estes dados positivamente para crescimento e amadurecimento da empresa. Analisando este cenário surge a necessidade de estudar tecnologias que auxiliem nesta tarefa para facilitar o controle deste fluxo de informação que aumenta a cada dia juntamente com o conhecimento dentro da organização que não podem ser desperdiçados.

Existem diversas ferramentas *free* no mercado para auxiliar a gestão do conhecimento, o problema é encontrar a ferramenta adequada que cumpra o objetivo descrito neste trabalho. O desenvolvimento de software possibilita a extração de conhecimento organizacional tanto quanto o conhecimento do negócio a ser desenvolvido, faz-se necessário organizar e utilizar esta informação em prol da organização.

Algumas barreiras ainda existem quanto às pessoas envolvidas nos projetos na área de TI, o compartilhamento de informação, o trabalho em equipe, a aceitação das metodologias utilizadas, isto dificulta a utilização na prática, mas com o empenho da organização esta nova cultura pode começar a dar resultados com ajuda das equipes.

## **3 HIPÓTESES DE SOLUÇÃO**

Definida a pesquisa a ser realizada neste trabalho, serão apresentadas algumas hipóteses de como por em prática a gestão do conhecimento no ambiente de desenvolvimento de software para o ambiente proposto.

- Hipótese 1: Analisar durante as fases do processo alguns pontos importantes para captura do conhecimento, abordando algumas técnicas de aquisição manuais como entrevistas estruturadas, que são bastante adequadas para capturar esse tipo de conhecimento.
- Hipótese 2: Durante a execução dos processos de desenvolvimento, diversos problemas surgem a todo instante e diversas soluções podem ser propostas para o mesmo problema. Desta forma, cada membro termina adquirindo um conjunto de lições aprendidas extremamente útil para a organização, solução será documentar e disponibilizar este conhecimento para compartilhar com o grupo e armazená-la para futuras consultas em outros projetos da organização.

## **4 OBJETIVOS**

O objetivo do trabalho será apresentar técnicas de gestão do conhecimento a serem empregadas em um ciclo de desenvolvimento de software baseado na metodologia RUP, focado em um projeto de pequeno porte.

### **4.1 Objetivos específicos:**

- Identificar técnicas e metodologias já existentes para Gestão do Conhecimento;
- Identificar o cenário favorável para a aplicação de técnicas de Gestão do Conhecimento dentro das organizações, focando empresas de desenvolvimento de software.
- Demonstrar em um projeto de desenvolvimento de software a captura e administração do conhecimento através da GC.

## **5 JUSTIFICATIVA**

Pelo volume de informação e a importância do conhecimento gerado e adquirido dentro das organizações, atualmente a GC aparece como um forte aliado das empresas para evitar uma série de problemas provenientes da falta de uma política de GC.

Para Rus & Lindvall (2002) o maior problema do capital intelectual é que “ele tem pernas e vai para casa todos os dias”. A frase deste autor demonstra um dos problemas que pode ocorrer pela falta da GC em uma empresa, que é o fato do conhecimento estar retido quase que somente nos empregados e, se estes deixam a empresa o conhecimento vai junto.

O modelo de desenvolvimento de software baseado no RUP, um modelo internacionalmente conhecido e utilizado por empresas conceituadas no mercado não aborda aspectos de gestão do conhecimento, portanto torna-se interessante apontar para este modelo algumas propostas para GC dentro do ciclo de desenvolvimento de sw. Cabe ressaltar que neste trabalho a metodologia de desenvolvimento de sw é uma instância do RUP adaptada para uma empresa de médio porte.

Devido a um acontecimento real na empresa em que se baseia este trabalho, grande parte do conhecimento de processos, conhecimento tácito, e conhecimento do próprio software desenvolvido foi perdido com a saída de vários funcionários em um curto prazo de tempo. Por iniciativa própria foi necessário estudar, pesquisar a GC para implementar no atual processo de desenvolvimento de software da organização, garantido ou permitindo que o conhecimento adquirido e gerado permaneçam para serem compartilhados.

## **6 FUNDAMENTAÇÃO TEÓRICA**

### **6.1 Conhecimento e Gestão o Conhecimento**

NONAKA e TAKEUCHI (1997) discorrem sobre os significados de conhecimento e informação, os diferenciam conforme as seguintes proposições: o conhecimento é função de uma atitude, perspectiva ou intenção específica e está relacionado à ação, ao contrário da informação; ambos, conhecimento e informação, dizem respeito ao significado, sendo específicos ao contexto e relacionais, na medida em que dependem da situação e são criados de forma dinâmica na interação social entre as pessoas; o conhecimento é um processo humano dinâmico de justificar a crença pessoal com relação à "verdade"; a informação é um meio ou material necessário para extrair e construir o conhecimento, acrescentando-lhe algo ou reestruturando-o; a informação é um produto capaz de gerar conhecimento,

sendo o conhecimento identificado com a crença produzida (ou sustentada) pela informação; a informação é um fluxo de mensagens, enquanto o conhecimento é criado por esse próprio fluxo de informação, ancorado nas crenças e compromissos de seu detentor.

Estes mesmos autores classificam o conhecimento humano em dois tipos: o tácito e o explícito. O conhecimento explícito é o conhecimento "articulado na linguagem formal, expressões matemáticas, especificações, manuais e assim por diante, podendo ser transmitido, formal e facilmente, entre os indivíduos" (NONAKA e TAKEUCHI, 1997, p. XIII). O conhecimento tácito, por sua vez, é "difícil de ser articulado na linguagem formal, sendo conhecimento formal incorporado à experiência individual e envolve fatores intangíveis como, por exemplo, crenças pessoais, perspectivas e sistemas de valor" (p. XIII). Ambos são considerados unidades estruturais básicas que se complementam mutuamente.

A partir da interação que podem se verificar entre estes dois tipos de conhecimento, os autores sugerem diversos modos de conversão do conhecimento, o que por sua vez produziria mais conhecimento. São eles a socialização (conversão de conhecimento tácito para conhecimento tácito), a externalização (conhecimento tácito para explícito), a combinação (conhecimento explícito para explícito) e a internalização (conhecimento explícito para tácito). A criação do conhecimento organizacional (que produz a inovação), seria segundo estes autores, uma interação contínua e dinâmica entre o conhecimento tácito e o conhecimento explícito, bem como através dos diversos níveis ontológicos (indivíduo, grupo, organização e inter-organização).

DAVENPORT e PRUSAK(1999), por sua vez, buscam uma conceituação de conhecimento, procurando oferecer uma definição funcional do termo, e consideram que o conhecimento deriva da informação da mesma forma como a informação deriva dos dados:

*"Conhecimento é uma mistura fluida de experiência condensada, valores, informação contextual e insight experimentado, a qual proporciona uma estrutura para avaliação e incorporação de novas experiências e informações. Ele tem origem e é aplicado na mente dos conhecedores. Nas organizações, ele costuma estar embutido não só em documentos ou repositórios, mas também em rotinas, processos, práticas e normas organizacionais"*

(DAVENPORT e PRUSAK, 1999, p. 6)

Para DAVENPORT e PRUSAK (1999), a gestão do conhecimento compõe-se de, pelo menos, três etapas, não necessariamente consecutivas ou ordenadas: geração, codificação e transferência do conhecimento.

Um conceito estreitamente relacionado à gestão do conhecimento é o de capital intelectual. Neste caso, STEWART(1998) considera que o capital intelectual pode ser encontrado nas pessoas (o capital humano, ou a "fonte da inovação e renovação"), nas estruturas (o capital estrutural) e nos clientes (o capital do cliente, considerando "o valor dos relacionamentos de uma empresa com as pessoas com as quais faz negócios", que também pode incluir os relacionamentos com os parceiros e fornecedores – p. 69)<sup>1</sup>. No entanto, o autor enfatiza que "o capital intelectual não é criado a partir de partes distintas de capital humano, estrutural e do cliente, mas do intercâmbio entre eles" (p. 70).

NONAKA e TAKEUCHI (1997) enfatizam a gestão do conhecimento como um processo interativo de criação do conhecimento organizacional, definindo-o como "a capacidade que uma empresa tem de criar conhecimento, disseminá-lo na organização e incorporá-lo a produtos, serviços e sistemas".

## 6.2 Metodologias de GC

Nesta seção serão apresentadas três metodologias encontradas na literatura sobre o assunto e após a apresentação destas, faremos a proposta de uma metodologia para ser aplicada na solução proposta neste trabalho de TCC.

A primeira a ser apresentada será de NONAKA e TAKEUCHI(1997), onde o conhecimento ocorre em quatro modos de conversão no que diz respeito ao ciclo da gestão do conhecimento:

**Socialização:** refere-se ao compartilhamento do conhecimento, informações e experiências, ou seja, um processo ao qual sugere que os colaboradores da empresa interajam entre si, realizando a troca de idéias, intenções e visões.

**Externalização:** refere-se a transformação do conhecimento tácito em conceitos explícitos, ou seja, transportar o conhecimento da mente para algo concreto, capaz de ser justificado, categorizado e contextualizado. Este processo pode ser realizado através da escrita, por exemplo.

**Combinação:** processo que realiza a sistematização de conceitos de um sistema de conhecimento, ou seja, a associação e identificação de conceitos. Este processo identifica e agrupa os conceitos semelhantes em conjuntos de conhecimento explícito.

**Internalização:** é a incorporação do conhecimento explícito em tácito, assegurando a expansão da experiência prática, ou seja, a captação individual do conhecimento que foi extraído para a organização. É quando o conhecimento explícito se torna ferramenta de aprendizagem através de documentos ou manuais, onde através destes, volta a ocupar a mente humana, tendo um conceito pessoal para cada indivíduo.

Já conforme ARONSON apud TURBAN(2004), o ciclo de sistema de gestão do conhecimento contempla seis etapas conforme abaixo:

**Criação do conhecimento:** O conhecimento é criado quando as pessoas apresentam novas formas de fazer ou desenvolver know-how. A criação do conhecimento também pode ser a importação do meio externo de conhecimentos.

**Captura do conhecimento:** A partir do momento em que o conhecimento é criado, ele deve ser reconhecido em uma empresa como valor.

**Refinamento do conhecimento:** Um novo conhecimento deve ser colocado no contexto organizacional para se tornar utilizável. É nessa etapa que o conhecimento tácito deve ser capturado junto com o conhecimento explícito.

**Armazenamento de conhecimento:** O conhecimento útil deve ser armazenado em formato razoável em um repositório de conhecimento com fácil acesso.

**Gerenciamento do conhecimento:** Fazer com que os conhecimentos armazenados estejam sempre atualizados.

**Disseminação do conhecimento:** Proliferar o conhecimento à todos, disponibilizando este em um ambiente de fácil acesso. Fazer com que todos conheçam o conhecimento adquirido.

Conforme DAVENPORT e PRUSAK(1999) o procedimento para se obter eficácia no gerenciamento do conhecimento empresarial pode ser sintetizado em três passos básicos, conforme segue:

**Geração do conhecimento:** Novos conhecimentos podem ser gerados pelas empresas através da aquisição pura e simples dos mesmos ou por desenvolvimento destes. Existem cinco formas de gerar conhecimento: aquisição, recursos dirigidos, fusão, adaptação e rede do conhecimento.

**Compartilhamento do conhecimento:** A cultura organizacional deve ser favorável ao compartilhamento do conhecimento, pois de pouco adianta o conhecimento gerado nas empresas se estes não são compartilhados. “É vital que o conhecimento seja compartilhado e distribuído dentro de uma organização para que informações ou experiências possam ser usadas por toda a empresa” (PROBST, RAUB e ROMHARDT, 2002). Para o sucesso do compartilhamento do conhecimento existem algumas estratégias, conforme segue: contratar as pessoas certas, realizar o rodízio de funções, estimular as conversas livres, realizar fóruns e feiras do conhecimento e leituras no ambiente do trabalho.

**Registro do conhecimento:** Registrar o conhecimento de forma clara e objetiva para quem o precisar, ou seja, transportar o conhecimento para um código inteligível.

### 6.3 Técnicas e Ferramentas de GC

Ferramentas de colaboração são ferramentas que permitem às pessoas criarem, partilharem, colaborarem e comentarem as informações, ou seja, facilitam o trabalho conjunto. Estas ferramentas são necessárias para que não se crie “ilhas de informações”, retendo conhecimentos de indivíduos que podem sair da empresa e compartilhando conhecimentos de novos funcionários em um único repositório. Estas ferramentas tecnológicas são de grande valia, pois podem propiciar um meio de aliviar a difícil tarefa de gerir o conhecimento nas organizações.

Segundo DAVENPORT e PRUSAK(1998), o objetivo das ferramentas de gestão do conhecimento é modelar parte do conhecimento que existe na cabeça das pessoas e no documentos corporativos, disponibilizando-o para toda a organização.

Para as empresas que estão prontas para desenvolverem um compromisso com o desenvolvimento do conhecimento, existem várias ferramentas que facilitam a rapidez de criação e partilha de informações, porem existem grandes questões a

serem verificadas no que diz respeito a escolha destas ferramentas, como por exemplo:

- Qual ferramenta é mais adequada para um determinado processo de gestão do conhecimento?

- Qual ferramenta pode extrair o máximo de um determinado processo de gestão do conhecimento?

- Quais elementos do ciclo de processo da gestão do conhecimento estão sendo tratados em uma determinada ferramenta de colaboração?

Deve-se lembrar que o uso de ferramentas de tecnologia sem uma metodologia bem trabalhada e sem a cultura organizacional voltada para o processo de gestão do conhecimento, de nada serve. Segundo DAVENPORT e PRUSAK (1998), a função mais valiosa da tecnologia na gestão do conhecimento é expandir o alcance e potencializar a velocidade de transferência do conhecimento.

Para a gestão do conhecimento, uma tecnologia pode ser considerada útil se promover a integração das pessoas, eliminar fronteiras de comunicação e permitir a criação de redes globais para partilha do conhecimento.

Abaixo são descritas algumas tecnologias e ferramentas de colaboração que serão analisadas neste trabalho, dentre as muitas tecnologias existentes no mercado:

- *Intranet*: Nada mais que redes privadas, é um ambiente ideal para o compartilhamento de informações dinâmicas e interligadas. DAVENPORT e PRUSAK (1998) consideram as tecnologias baseadas na *Web* muito intuitivas, pois lidam facilmente com representações do conhecimento. Os sistemas baseados na *intranet* privilegiam a informação interna à organização. Dessa forma, a *intranet* está se tornando um importante veículo de informação interna entre a empresa e o funcionário. No entanto esta comunicação é passiva, no sentido de que a informação esta na *intranet* e o colaborador deve ir busca-lá.
- Repositório de Dados, Informações e Conhecimento: A criação de diretórios disponíveis em servidores e compartilhados em rede é estruturada por projetos e por tipos de informações, utilizando o cvs para controle de versões

de documentos, código fonte, artefatos identificando os devidos responsáveis pelas alterações e divulgação de eventuais atualizações de documentos. É uma forma interessante de compartilhar informações e possivelmente contribuir para armazenar conhecimento.

- Lições Aprendidas: refere-se às experiências adquiridas pelos funcionários no desenvolvimento de suas atividades e que podem ser úteis para situações semelhantes que venham a ser vivenciadas por algum funcionário da empresa. Estas lições aprendidas podem ser armazenadas em um local comum e acessível a todos (usualmente no Repositório de Dados, Informações e Conhecimento) ou através de ciclos de palestras proferidas pelos próprios funcionários aos colegas. Nestas lições aprendidas podem estar incluídas também relatórios de eventos, visitas e cursos realizados. Uma boa prática sempre muito útil é de no final de cada projeto de desenvolvimento de software levantar, entre todos os membros da equipe, as lições aprendidas durante o projeto, de modo a servir de base para experiências futuras, listando inclusive falhas no projeto e oportunidades de melhoria. Cabe ressaltar que, segundo RUS et. al. (2001, p. 19), as lições aprendidas devem ser codificadas e transferidas, já que o conhecimento foi criado anteriormente de forma tácita dentro das pessoas.

#### **6.4 Desenvolvimento de Software**

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

Sem um processo bem definido, sua equipe irá desenvolver de maneira designada, com o sucesso baseado nos esforços heróicos de alguns dedicados colaboradores individuais. Então esta não é uma condição sustentável. O *Rational Unified Process* é um processo que descreve quem está fazendo o quê, como e quando.

- Quem: Trabalhadores
- Como: Atividade
- O quê: Artefatos
- Quando: Fluxos

O *Rational Unified Process*® (também chamado de processo RUP®) é um processo de engenharia de software. Ele oferece uma abordagem baseada em disciplinas para atribuir tarefas e responsabilidades dentro de uma organização de desenvolvimento. Sua meta é garantir a produção de software de alta qualidade que atenda às necessidades dos usuários dentro de um cronograma e de um orçamento previsíveis.

A figura abaixo mostra a arquitetura geral do RUP.

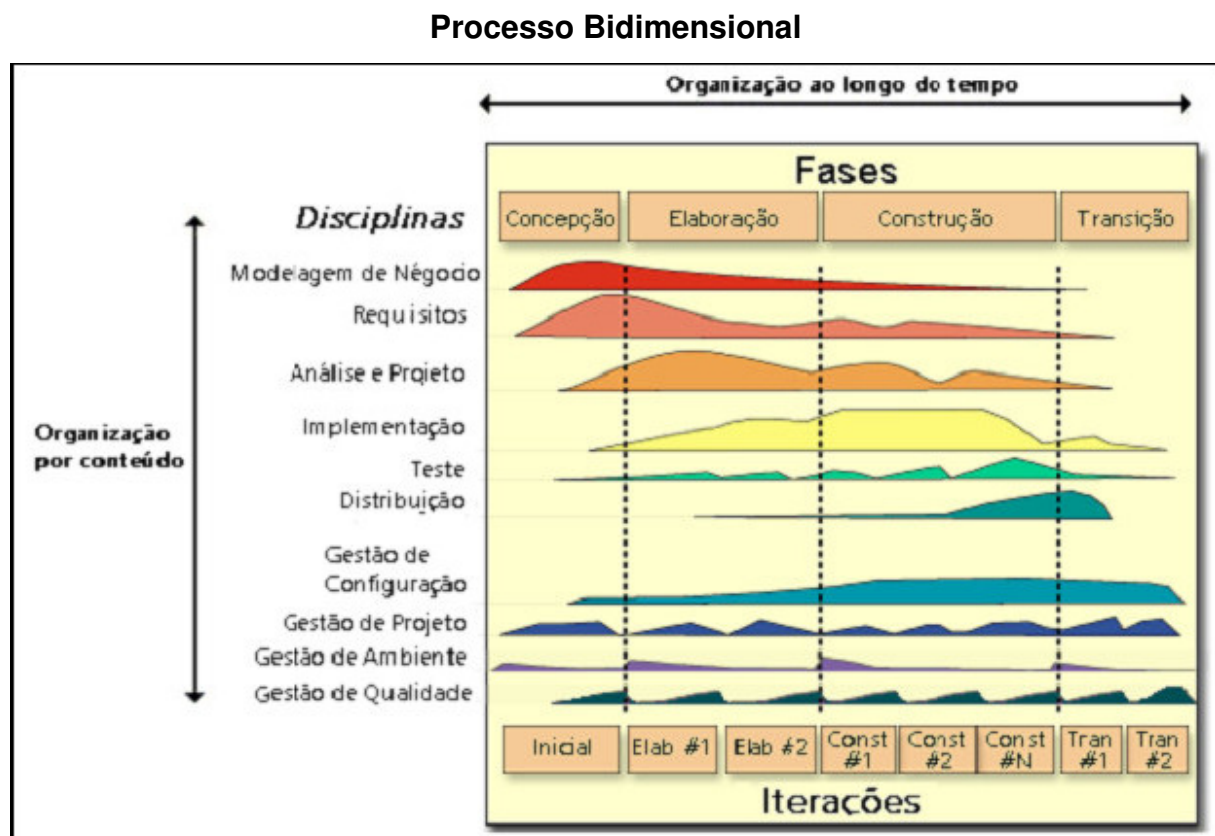


Figura 1 – Arquitetura geral do RUP

O RUP tem duas dimensões:

O eixo horizontal representa o tempo e mostra os aspectos do ciclo de vida do processo à medida que se desenvolve.

O eixo vertical representa as disciplinas, que agrupam as atividades de maneira lógica, por natureza.

A primeira dimensão representa o aspecto dinâmico do processo quando ele é aprovado e é expressa em termos de fases, iterações e marcos.

A segunda dimensão representa o aspecto estático do processo, como ele é descrito em termos de componentes, disciplinas, atividades, fluxos de trabalho, artefatos e papéis do processo.

O gráfico mostra como a ênfase varia através do tempo. Por exemplo, nas iterações iniciais, dedicamos mais tempo aos requisitos. Já nas iterações posteriores, gastamos mais tempo com implementação.

## **7 DESCRIÇÃO DO PROCESSO ATUAL DE DESENVOLVIMENTO DE SOFTWARE**

Nesta etapa será descrito o processo de desenvolvimento de software implantado na empresa estudada em questão.

### **7.1 Fluxos Iniciais**

O objetivo dos Fluxos Iniciais é receber, analisar, classificar as solicitações de acordo com os tipos de projetos (manutenções normais, manutenções urgentes, novos) e gerenciar a *Task List*.

#### **7.1.1 Solicitação do Cliente**

O objetivo do *Workflow* Solicitação do Cliente é receber e processar a solicitação do cliente, analisando o tipo de processo que deverá ser executado. Essa solicitação pode ser de novos sistemas, de novas funcionalidades ou de manutenções.

- Rec. Processar a Solicitação Cliente:

Na atividade Receber e Processar a Solicitação Cliente o Gerente Comercial recebe uma solicitação do Cliente e formaliza essa solicitação utilizando o artefato Solicitação do Cliente. Pode ser uma solicitação para a inclusão de novas funcionalidades, para a correção de *bugs*, para modificações na aplicação ou para a solicitação de um novo sistema.

#### Artefatos Entrada:

\* Uma solicitação do cliente. Pode ser uma solicitação via telefone, msn, e-mail, documento, etc.

#### Artefatos Saída:

\* Solicitação do Cliente

- Desenvolver Proposta Comercial:

Na atividade Desenvolver Proposta Comercial o Gerente de Projetos, em conjunto com o Gerente Comercial, define o cronograma, o custo do projeto, o escopo e a forma como o projeto será conduzido. Essas informações serão documentadas no artefato Proposta Comercial que é avaliada e aprovada pelo cliente.

#### Artefatos Entrada:

\* Solicitação do Cliente

#### Artefatos Saída:

\* Proposta Comercial

\* Entrevista

- Revisar Proposta Comercial:

Na atividade Revisar Proposta Comercial o Cliente analisa a Proposta Comercial e solicita alterações. O Gerente Comercial implementa as modificações conforme solicitação do cliente.

#### Artefatos Entrada:

\* Proposta Comercial

\* Solicitação do Cliente

\* Entrevista

#### Artefatos Saída:

\* Proposta Comercial

- Atualizar *Task List*:

Na atividade *Atualizar Task List* o Gerente de Projetos atualiza a lista de tarefas (*Task List*) com as novas funcionalidades solicitadas pelo cliente para implementá-las posteriormente.

Artefatos Entrada:

- Solicitação do Cliente
  - \* *Task List*

Artefatos Saída:

- \* *Task List*
- Armazenar Proposta Comercial

Na atividade *Armazenar Proposta Comercial* o Gerente Comercial guarda a Proposta Comercial, bem como todos os documentos que tenham sido gerados até o momento.

Artefatos Entrada:

- \* Proposta Comercial
- \* Solicitação do Cliente
- \* Entrevista

Artefatos Saída:

- \* Não tem

### **7.1.2 Análise da Task List**

O objetivo do *Workflow Análise da Task List* é o gerenciamento, feito pelo Gerente de Projetos, das solicitações não urgentes feitas pelo cliente, agrupando solicitações para criar projetos de manutenção.

- Organizar; Agrupar; Priorizar TL:

Na atividade *Organizar, Agrupar e Priorizar Task List* o Gerente de Projetos analisa as tarefas na *Task List*, organiza de acordo com as prioridades do cliente e define o escopo para um projeto de manutenção normal.

Artefatos Entrada:

- \* *Task List*

Artefatos Saída:

\* *Task List*

- Desenvolver Proposta Comercial:

Na atividade Desenvolver Proposta Comercial o Gerente de Projetos, em conjunto com o Gerente Comercial, define o cronograma, o custo do projeto, o escopo e a forma como o projeto será conduzido. Essas informações serão documentadas no artefato Proposta Comercial que será avaliada e aprovada pelo cliente.

Artefatos Entrada:

\* *Task List*

Artefatos Saída:

\* Proposta Comercial

\* Entrevista (opcional)

- Revisar Proposta Comercial:

Na atividade Revisar Proposta Comercial o Cliente analisa a Proposta Comercial podendo aprová-la, dando início ao projeto; podendo rejeitá-la ou então solicitar alterações. No caso de alterações, o Gerente de Projetos modifica a Proposta Comercial conforme solicitação do cliente.

Artefatos Entrada:

\* Proposta Comercial

\* Entrevista (opcional)

Artefatos Saída:

\* Proposta Comercial

- Armazenar Proposta Comercial:

Na atividade Armazenar Proposta Comercial o Gerente de Projetos armazena a Proposta Comercial, bem como todos os documentos que tenham sido gerados até o momento.

Artefatos Entrada:

- \* Proposta Comercial
- \* Entrevista (opcional)

Artefatos Saída:

- \* Não tem

## 7.2 Tipos de Projetos

Recebida a Solicitação do Cliente, deve-se analisá-la e classificá-la como um dos seguintes tipos de projeto:

- \* Manutenções Urgentes (PPMU)
- \* Novos (PPN)

### 7.2.1 Novos - PPN

O objetivo do Processo Projetos Novos (PPN) é demonstrar o processo de projetos nos quais, o escopo do projeto é o desenvolvimento de um módulo ou sistema totalmente novo ou quando não é possível ter um conhecimento prévio a cerca dos requisitos envolvidos. Além disso, o PPN pode ser usado, de forma customizada, para projetos de manutenção normal.

O PPN está organizado em oito disciplinas, que podem ser entendidas como *containers*. Essas disciplinas representam um particionamento, de todos os papéis e atividades do processo, em agrupamentos lógicos de áreas de interesse ou especialidade.

#### 7.2.1.1 Gerenciamento de Projetos

A Disciplina de Gerenciamento de Projetos trata da aplicação de conhecimento, competências, ferramentas e técnicas de planejamento e gerenciamento às diversas atividades do projeto visando alcançar ou exceder as necessidades e expectativas dos *stakeholders* do projeto. A figura abaixo mostra o *workflow* da disciplina.

- Iniciar Projeto:

Na atividade Iniciar Projeto se organizam os documentos para o projeto que se inicia a partir dos artefatos gerados nos fluxos iniciais. Nessa atividade o Gerente de Projetos utiliza os artefatos Proposta Comercial, Entrevista e Solicitação do Cliente

para definir um Plano de Projeto inicial e também iniciar a elaboração do Documento de Visão.

Artefatos Entrada:

- \* Proposta Comercial
- \* Solicitação do Cliente
- \* Entrevista

Artefatos Saída:

- \* Documento de Visão
- \* Plano de Iteração (Gantt)
- \* Plano de Projeto

#### **7.2.1.1.1. Planejar o Projeto**

Esse grupo de atividades desenvolve todos os planos necessários ao projeto como, Plano de Iteração, Plano de Riscos, Plano de Gerenciamento de Configuração, Plano de Ambiente, entre outros. Refina o Plano de Projeto a cada iteração, possibilita acompanhar e documentar os riscos e problemas que ocorrem durante o projeto. Além disso, define a equipe do projeto no início do mesmo, bem como redefine a equipe a cada iteração (se necessário).

- Refinar Plano de Projeto

Nesta atividade são feitos refinamentos a cada iteração no artefato Plano de Projeto. Na primeira iteração é melhorado o Plano de Projeto inicial, que foi elaborado com base nos artefatos dos fluxos iniciais. Nas iterações seguintes os refinamentos relacionam-se aos recursos, ao cronograma e ao orçamento do projeto.

Os refinamentos se fazem necessários, pois o projeto é dinâmico, diversos problemas podem ocorrer, forçando o Gerente de Projetos a revisar e melhorar o artefato constantemente.

Artefatos Entrada:

- \* Documento de Visão
- \* Plano de Projeto

- \* Plano de Riscos

Artefatos Saída:

- \* Plano de Projeto

- Definir Org. do Projeto e Equipe:

Nesta atividade o Gerente de Projetos pensa na equipe do projeto, nos recursos que serão necessários para desenvolver a aplicação e no perfil de cada recurso, para alocar o par recurso-tarefa de forma eficiente e produtiva.

Artefatos Entrada:

- \* Documento de Visão
- \* Plano de Riscos
- \* Plano de Projeto

Artefatos Saída:

- \* Plano de Projeto

- Desenvolver Lista de Problemas:

Nesta atividade o Gerente de Projetos documenta os problemas que ocorrem durante o projeto. Na primeira iteração, esse artefato é instanciado com os dados do projeto, mas à medida que o projeto for evoluindo, problemas podem surgir, e esses devem estar documentados no artefato Lista de Problemas. Na atividade também são feitos refinamentos nos problemas e informadas quais as ações serão tomadas e quem será o responsável para corrigir cada problema.

Artefatos Entrada:

- \* Documento de Visão

Artefatos Saída:

- \* Lista de Problemas

- Planejar Configuração:

Nesta atividade o Gerente de Projetos cria a estrutura de repositório (CVS) do projeto e define as políticas de acesso a esses diretórios. Essas informações são

documentadas no artefato Plano de Gerenciamento de Configuração que é constantemente avaliado e atualizado.

Artefatos Entrada:

- \* Documento de Visão

Artefatos Saída:

- \* Plano de Gerenciamento de Configuração

- Planejar Ambiente:

Nesta atividade o Gerente de Projetos faz um levantamento de todos os recursos de hardware e software necessários para o projeto e para a aplicação criando o artefato Plano de Ambiente. Além disso, define os prazos (cronograma) que esses recursos serão necessários. O Gerente de Projetos faz um acompanhamento nesse artefato durante todo o projeto.

Artefatos Entrada:

- \* Documento de Visão

Artefatos Saída:

- \* Plano de Ambiente

- Desenv. Plano Aceitação Entrega:

Nesta atividade o Gerente de Projetos define critérios que deverão ser atendidos, por parte do cliente e da equipe do projeto, para determinar a aceitabilidade das entregas do projeto.

Artefatos Entrada:

- \* Documento de Visão

Artefatos Saída:

- \* Plano de Aceitação

- Desenvolver Plano de Riscos

Nesta atividade o Gerente de Projetos instancia, na primeira iteração, o artefato Plano de Riscos com os dados do projeto e da aplicação e documenta todos os riscos que julga que possam vir a ocorrer no projeto, bem como descreve as ações

que serão tomadas. À medida que o projeto evolui, essa atividade será utilizada para acompanhar os riscos, refinar o artefato e tomar as decisões que foram descritas caso um risco se torne real.

Artefatos Entrada:

- \* Documento de Visão
- \* Plano de Projeto

Artefatos Saída:

- \* Plano de Riscos
- Planejar Implantação

Nesta atividade o Gerente de Projetos define o processo de implantação do produto (os responsáveis pela implantação, quando o processo vai ocorrer e quais as ferramentas necessárias). Ele organiza essas informações no artefato Plano de Implantação. Além disso, acompanha e atualiza o processo de implantação da aplicação com base nas mudanças ocorridas no Plano de Projeto.

Artefatos Entrada:

- \* Plano de Iteração (Gantt)
- \* Plano de Projeto

Artefatos Saída:

- \* Plano de Implantação

#### **7.2.1.1.2. Planejar a Iteração**

Esse grupo de atividades permite ao Gerente de Projetos dividir as tarefas do projeto em um conjunto de iterações e atribuir recursos humanos e tempo a cada tarefa da iteração.

- Desenvolver Plano de Iteração:

Nesta atividade o Gerente de Projetos divide as atividades do projeto em um conjunto de iterações, cria um diagrama de Gantt no artefato Plano de Projeto, por meio da alocação de recursos para cada tarefa.

Artefatos Entrada:

- \* Documento de Visão
- \* Modelo de Casos de Uso
- \* Plano de Projeto
- \* Plano de Riscos

Artefatos Saída:

- \* Plano de Iteração (Gantt)
- Alocar Equipe:

Nesta atividade o Gerente de Projetos aloca recursos humanos às tarefas do cronograma e define o tempo necessário para finalizar cada atividade, criando assim, o cronograma (diagrama de Gantt).

Artefatos Entrada:

- \* Plano de Iteração (Gantt)
- \* Plano de Projeto

Artefatos Saída:

- \* Plano de Iteração (Gantt)
- \* Plano de Projeto

#### **7.2.1.1.3. Gerenciar Iteração**

Esse grupo de atividades representa a execução de um ciclo de tarefas (iteração) que estão descritas no Plano de Projeto. Neste grupo, o gerente define as tarefas no sistema *Application* e avalia o resultado de cada iteração.

- Iniciar Iteração:

Nesta atividade o Gerente de Projetos aloca os recursos humanos necessários a cada tarefa no sistema *Application*, permitindo que os membros da equipe possam iniciar suas atividades.

Artefatos Entrada:

- \* Plano de Iteração (Gantt)
- \* Plano de Projeto

Artefatos Saída:

\* *Application*

- Avaliar Iteração:

Nesta atividade, ao final de cada iteração, o Gerente de Projetos faz uma reunião com toda a equipe do projeto para discutir o andamento do mesmo, avaliar a iteração que foi finalizada, discutir os problemas que ocorreram e conversar a respeito da próxima iteração.

Artefatos Entrada:

- \* Avaliação do Projeto
- \* Documento de Visão
- \* Lista de Problemas
- \* Plano de Iteração (Gantt)
- \* Plano de Projeto
- \* Plano de Testes

Artefatos Saída:

- \* Avaliação do Projeto

#### **7.2.1.1.4. Monitorar e Controlar o Projeto**

Esse grupo de atividades representa as ações que o gerente de projetos deve conduzir quando ocorrem exceções ou imprevistos no projeto.

- Agendar e Redefinir Trabalho

Nesta atividade o Gerente de Projetos realoca recursos ou redistribui tarefas, caso ocorra algum problema com relação aos recursos do projeto. Por exemplo, se algum colaborador ficar doente, as tarefas terão que ser redistribuídas.

Artefatos Entrada:

- \* *Application*
- \* Plano de Iteração (Gantt)
- \* Plano de Projeto

Artefatos Saída:

- \* *Application*

- \* Plano de Iteração (Gantt)
- \* Plano de Projeto
- Monitorar Status do Projeto:

Nesta atividade o Gerente de Projetos acompanha o andamento do projeto referente à ocorrência ou não dos riscos, a existência de problemas, possíveis atrasos no cronograma, a coleta de métricas e documenta essas questões no artefato Avaliação do Projeto. Esse artefato serve como base para a reunião de avaliação das iterações. Além disso, o Gerente de Projetos pode comparar o status atual do projeto com as questões que foram definidas nos planos e traçar suas estratégias para a solução de eventuais problemas.

Artefatos Entrada:

- \* *Application*
- \* Lista de Problemas
- \* Planilha de Medições do Projeto
- \* Plano de Iteração (Gantt)
- \* Plano de Medição
- \* Plano de Projeto
- \* Plano de Riscos

Artefatos Saída:

- \* *Application*
- \* Avaliação do Projeto
- \* Lista de Problemas
- \* Planilha de Medições do Projeto
- Relatar Status do Projeto

Nessa atividade o Gerente de Projetos preenche a Planilha de Medições do Projeto (com as métricas coletadas) e mantém a equipe informada sobre o status atual do projeto.

Artefatos Entrada:

- \* Avaliação do Projeto
- \* Planilha de Medições do Projeto

Artefatos Saída:

- \* Avaliação do Projeto

#### **7.2.1.1.5. Gerar Baseline**

Esta atividade deve sempre ser executada ao final de cada iteração, para gerar um marco no projeto. Como todos os artefatos (código fonte, padrões, documentação, diagramas) estão no CVS, o Gerente de Projetos gera uma *baseline* com uma versão de todos esses documentos e cria um marco do projeto. Sendo assim, nessa atividade busca-se “tirar uma fotografia” do estado do projeto.

Artefatos Entrada:

- \* Repositório do Projeto

Artefatos Saída:

- \* Repositório do Projeto (Com *Baseline*)

#### **7.2.1.1.6. Fechar Projeto**

Nesta atividade se documenta todas as informações referentes ao projeto que se está concluindo além da realização de uma reunião formal para avaliação final do projeto.

Artefatos Entrada:

- \* Avaliação do Projeto
- \* Lista de Problemas
- \* Plano de Projeto

Artefatos Saída:

- \* Avaliação do Projeto
- \* Lista de Problemas
- \* Plano de Projeto

### 7.2.1.2 Ambiente

O objetivo da Disciplina de Ambiente é oferecer à organização o ambiente de desenvolvimento de software (processos, ferramentas e infra-estrutura), que darão suporte à equipe de desenvolvimento. Ela descreve as atividades para o desenvolvimento das diretrizes de suporte de um projeto.

- Selecionar e Adquirir Ferramenta:

Essa atividade é realizada pelo Gerente de Projetos que analisa as necessidades do projeto e faz uma lista com todas as ferramentas que serão utilizadas no decorrer do projeto. Com essa atividade, o gerente pode fazer uma lista de itens que serão solicitados ao Suporte Infra-estrutura. Caso a gerência permita a aquisição das ferramentas, o Suporte Infra-estrutura irá comprá-las.

Artefatos Entrada:

- \* Plano de Ambiente
- \* Plano de Projeto

Artefatos Saída:

- \* Ferramentas (Solicitadas)
- \* Plano de Ambiente

- Instalar Ferramentas:

Com base na lista de ferramentas selecionadas pelo Gerente de Projetos, o Suporte Infra-estrutura instala as ferramentas nas máquinas dos colaboradores do projeto. Nessa atividade, o Suporte Infra-estrutura deixa o *workspace* de cada recurso do projeto preparado para desenvolver suas atividades.

Artefatos Entrada:

- \* Ferramentas (Solicitadas)

Artefatos Saída:

- \* Ferramentas (Instaladas)

- Suporte ao Ambiente:

Essa atividade é realizada pelo Suporte Infra-estrutura onde ele acompanha e garante que o ambiente de trabalho dos membros da equipe esteja funcionando corretamente.

Artefatos Entrada:

- \* Não tem

Artefatos Saída:

- \* Infra-Estrutura (Revisada)

### **7.2.1.3 Configuração e Mudanças (CVS)**

O objetivo da Disciplina Configuração e Mudanças (CVS) é gerenciar e controlar a evolução do software através do controle formal de versão e de solicitações de mudanças, mantendo a integridade dos artefatos do projeto;

- Criar Repositório do Projeto:

Nessa atividade o Suporte Infra-estrutura cria a estrutura padrão de diretórios no cvs para o projeto em questão e implementa as políticas de acesso e de backup no servidor.

Artefatos Entrada:

- \* Plano de Gerenciamento de Configuração

Artefatos Saída:

- \* Repositório do Projeto

### **7.2.1.4 Requisitos**

Na disciplina de Requisitos busca-se capturar e gerenciar requisitos do sistema para desenvolvê-los de forma efetiva, visando atender às necessidades dos usuários e *stakeholders*. Do ponto de vista do analista/projetista, o propósito é criar uma especificação em alto nível do que o sistema deve fazer.

- Entrevistar *Stakeholders*:

Nessa atividade o Gerente de Projetos realiza uma entrevista com os *stakeholders* envolvidos no projeto para identificar suas necessidades, os requisitos da aplicação, os problemas e também as possíveis soluções. A atividade Entrevistar *Stakeholders* pode ser em grupo ou de forma individual.

Artefatos Entrada:

- \* Documento de Visão

Artefatos Saída:

- \* Entrevista

- Realizar Análise de Impacto:

Essa atividade é executada quando o projeto tiver o objetivo de realizar modificações na aplicação. Assim, quando a alteração é solicitada pelo usuário, o gerente de projetos, em conjunto com o analista de sistemas, deve avaliar o impacto da mudança na aplicação e escolher a melhor estratégia para resolver o problema.

- Refinar Requisitos:

Nesta atividade o Gerente de Projetos utiliza os artefatos Entrevista e o Documento de Visão para identificar, descrever e documentar os requisitos da aplicação. Para documentar os requisitos deve utilizar a ferramenta EA, criando uma lista dos requisitos e definindo características como prioridade, complexidade e estado do requisito.

Artefatos Entrada:

- \* Documento de Visão

- \* Entrevista

Artefatos Saída:

- \* Requisitos

- Refinar o Documento de Visão:

Nesta atividade busca-se revisar e refinar o Documento de Visão com base nas informações levantadas com o artefato Entrevista, bem como em dados de iterações anteriores.

Artefatos Entrada:

- \* Documento de Visão

- \* Entrevista

Artefatos Saída:

- \* Documento de Visão

- Capturar Vocabulário:

Nessa atividade o Analista de Sistemas analisa os artefatos Entrevista e Documento de Visão buscando identificar e descrever termos relacionados à aplicação e ao negócio. Além disso, também identifica e descreve as regras de negócio para o desenvolvimento do sistema. A atividade resulta na criação do artefato Glossário que, durante o andamento do projeto, irá ser refinado.

Artefatos Entrada:

- \* Documento de Visão

- \* Entrevista

Artefatos Saída:

- \* Glossário

- Refinar Modelo de Negócio:

Na atividade Refinar Modelo de Negócio o Analista de Negócios abstrai o funcionamento do negócio criando um Modelo de Negócio (BPMN), independente do sistema a ser desenvolvido. Com esse modelo é possível ter uma visão do contexto do negócio no qual a aplicação irá funcionar, bem como entender suas interfaces de comunicação.

Artefatos Entrada:

- \* Documento de Visão

- \* Entrevista

- \* Glossário

Artefatos Saída:

- \* BPMN

- \* Glossário

- Definir Caso de Uso:

Nesta atividade o Analista de Sistemas identifica os casos de uso da aplicação, bem como os atores que interagem com a mesma. Para realizar essa atividade, o

Analista de Sistemas baseia-se no Documento de Visão e na descrição dos requisitos da aplicação.

Artefatos Entrada:

- \* Documento de Visão
- \* Requisitos

Artefatos Saída:

- \* Casos de Uso (Identificados)
- Priorizar Casos de Uso:

Na atividade Priorizar Casos de Uso o Gerente de Projetos e o Arquiteto de Software devem se reunir, e com base nos casos de uso identificados na atividade anterior, devem priorizar os casos de uso para permitir que os mesmos sejam implementados em iterações diferentes. Essa priorização pode ser construída por meio de uma lista onde o gerente mostra a ordem dos casos de uso dizendo quais são os que devem ser implementados primeiro.

Artefatos Entrada:

- \* Casos de Uso (Identificados)
- \* Documento de Visão

Artefatos Saída:

- \* Casos de Uso (Priorizados)
- Detalhar Caso de Uso:

Nessa atividade o Analista de Sistemas detalha cada caso de uso do Modelo de Casos de Uso, seguindo a ordem definida na priorização dos mesmos. Para detalhar o caso de uso, o Analista de Sistemas define os cenários, as pré e pós-condições e linka o caso de uso com os requisitos que ele implementa.

Artefatos Entrada:

- \* Casos de Uso (Identificados)
- \* Casos de Uso (Priorizados)
- \* Documento de Visão

- \* Requisitos

Artefatos Saída:

- \* Modelo de Casos de Uso

- Criar Cenários Típicos de Uso:

Na atividade Criar Cenários Típicos de Uso o Analista de Sistemas, com base no BPMN, nos requisitos e nos casos de uso, identifica e descreve os Cenários Típicos de Uso da aplicação. Esse artefato define a seqüência dos casos de uso que refletem uma ação freqüentemente realizada pelo usuário junto à aplicação.

Artefatos Entrada:

- \* BPMN

- \* Documento de Visão

- \* Modelo de Casos de Uso

Artefatos Saída:

- \* Cenários Típicos de Uso

#### **7.2.1.5 Análise & Projeto**

O objetivo da Disciplina de Análise & Projeto é traduzir os requisitos para uma especificação que descreva como implementar o sistema.

O propósito da análise é transformar requisitos do sistema em um modelo que sirva de mapa para que o projetista de software projete um conjunto de classes e subsistemas. Esta transformação é dirigida pelos Casos de Uso e remodelada posteriormente pelos requisitos não-funcionais do sistema. A análise assegura que os requisitos funcionais do sistema sejam implementados. Por questões de simplificação, ela ignora muitos requisitos não-funcionais do sistema e também as restrições de implementação do ambiente. Como resultado, a análise expressa uma figura quase ideal do sistema. O principal trabalho em análise inicia no final da fase de Iniciação e é o foco principal da fase de Elaboração, junto com os requisitos.

Por outro lado, o propósito do projeto é adaptar o resultado da análise levando em consideração as restrições impostas pelos requisitos não-funcionais, o ambiente de implementação, requisitos de performance, entre outros. O projeto é um refinamento da análise, focando na otimização do projeto do sistema e assegurando

uma cobertura total aos requisitos definidos. No projeto os projetistas decidem as questões estratégicas do projeto, como a persistência e distribuição de um objeto e criam um modelo do projeto. O gerente do projeto e o arquiteto também devem criar políticas para o entendimento de questões táticas do projeto.

Observando a relação das atividades da disciplina nas fases do RUP, nota-se que a maioria das atividades na fase de Elaboração consistem em criar modelos que capturam o comportamento desejado do sistema. O trabalho de análise ocorre em grande parte, juntamente com a captura de requisitos, na verdade estas duas atividades freqüentemente andam juntas, é necessário realizar algum tipo de análise nos requisitos para esclarecê-los e resolver qualquer problema com os requisitos e até mesmo a falta de requisitos.

O trabalho de projeto é realizado durante a última parte da fase de Elaboração e a primeira metade da fase de Construção. O principal foco no início das iterações iniciais é a parte de requisitos e análise, e como a atividade de análise se torna mais completa, o foco da modelagem junta-se ao projeto. A análise e o projeto podem ocorrer em paralelo.

- Analisar Casos de Uso:

Na atividade Analisar Casos de Uso o Analista de Sistemas realiza uma análise no Modelo de Casos de Uso a fim de descobrir e organizar as classes da aplicação.

Artefatos Entrada:

- \* Documento de Visão
- \* Glossário
- \* Modelo de Casos de Uso

Artefatos Saída:

- \* Modelo de Classes
- \* Modelo de Casos de Uso

- Projetar Arquitetura:

Na atividade Projetar Arquitetura o Arquiteto de Software analisa documentos já existentes, como casos de uso e modelo de classes e define a arquitetura e os

padrões arquiteturais para a aplicação. Ele descreve como será a subdivisão do sistema em componentes e pacotes.

Artefatos Entrada:

- \* Modelo de Classes

Artefatos Saída:

- \* Documento de Visão (Seção Arquitetura da Aplicação)

- **Projetar Interface do Usuário:**

Na atividade **Projetar Interface do Usuário** cria-se um protótipo da interface gráfica da aplicação para validar o projeto da interface do usuário com os requisitos funcionais e de usabilidade. Além disso, constrói-se o mapa de menus da aplicação. No caso das aplicações desenvolvidas com o suporte da MTF, o Projetista de Interface deve estereotipar os casos de uso para formar a hierarquia de menus.

Artefatos Entrada:

- \* Documento de Visão
- \* Modelo de Classes
- \* Modelo de Casos de Uso

Artefatos Saída:

- \* Modelo de Casos de Uso (Com *Profile* MTF)
- \* Projeto da Interface

- **Projetar Banco de Dados:**

Na atividade **Projetar Banco de Dados** o Projetista da Base de Dados cria a estrutura de banco de dados da aplicação, garantindo a persistência dos dados além de elaborar o modelo ER. Se a aplicação for desenvolvida com o suporte da MTF, o projetista do banco deve primeiro, estereotipar as classes do modelo de classes para que o *framework* possa gerar a persistência dos dados. Assim, o modelo ER (para aplicações Java) será construído via engenharia reversa do banco de dados.

Artefatos Entrada:

- \* Modelo de Classes

Artefatos Saída:

- \* Modelo de Classes (Com *Profile* MTF)
- \* Modelo ER
- Construir Caso de Uso Real:

Nessa atividade o analista de sistema vai criar a ordem de serviço para o desenvolvedor. Para tanto, o mesmo deve criar o artefato caso de uso real, realizando a ligação entre a interface na qual o caso de uso é acessado, e especificando o relacionamento entre o caso de uso e as classes que implementam seu comportamento.

Artefatos Entrada:

- \* Modelo de Casos de Uso
- \* Modelo de Classes
- \* Projeto da Interface

Artefatos Saída:

- \* Casos de Uso Real
- Refinar Arquitetura:

Nesta atividade o Arquiteto de Software executa refinamentos na arquitetura da aplicação, identificando problemas, sugerindo melhorias e atualizando-a conforme alterações nos artefatos do projeto.

Artefatos Entrada:

- \* Diagrama de Implantação

Artefatos Saída:

- \* Diagrama de Implantação
- \* Documento de Visão (Seção Arquitetura da Aplicação)

#### **7.2.1.6 Implementação**

O objetivo da Disciplina de Implementação é definir a organização do código (implementação de subsistemas organizados em camadas), implementar os elementos do projeto (implementação de elementos, arquivos, executáveis e outros),

testar os componentes desenvolvidos como unidades e integrar os componentes produzidos em um sistema executável.

- Gerar Aplicação:

Na atividade Gerar Aplicação o Arquiteto de Software gera uma versão inicial da aplicação, com base no modelo de classes e no modelo de casos de uso. Em aplicações Java, essa atividade é realizada com o suporte das tecnologias MTF e GEN.

Artefatos Entrada:

- \* Modelo de Classes (XML)
- \* Modelo de Casos de Uso (XML)

Artefatos Saída:

- \* Código Fonte

- Atualizar ER e Banco de Dados:

Nessa atividade o DBA deve fazer melhorias na base de dados visando performance. Para tanto, o DBA pode criar *index*, *constraints*, fatorar tabelas, entre outras atividades. Muitas vezes essas alterações irão refletir no Modelo de Classes, que deverá ser alterado. Outro objetivo dessa atividade é manter o modelo ER sempre atualizado com o modelo de classes. Assim, o DBA utiliza uma ferramenta de modelagem de base de dados para realizar um complete compare e gerar os scripts para atualizar a base de dados.

Artefatos Entrada:

- \* Modelo de Classes
- \* Modelo ER
- \* Script (Gerado pela MTF)

Artefatos Saída:

- \* Modelo de Classes
- \* Modelo ER

- Implementar:

Nessa atividade os Desenvolvedores irão implementar, a partir da versão inicial do código fonte, uma nova versão com as funcionalidades que a aplicação deve possuir e com as interfaces gráficas que não foram geradas a partir do GEN. Dessa forma, é na atividade Implementar, que a aplicação vai ser desenvolvida.

Artefatos Entrada:

- \* Caso de Uso Real
- \* Código Fonte

Artefatos Saída:

- \* Código Fonte
- Gerar Build

Na atividade Gerar *Build* o Arquiteto de Software gera uma versão executável da aplicação com a documentação associada (*build*). Essa atividade é realizada conforme o cronograma definido no Plano de Projeto.

Artefatos Entrada:

- \* Código Fonte
- \* Plano de Projeto

Artefatos Saída:

- \* *Build*

### **7.2.1.7 Testes**

O objetivo da Disciplina de Testes é avaliar a qualidade do produto, utilizando as seguintes práticas:

- \* Encontrar e documentar defeitos na qualidade do software;
- \* Advertir a respeito da percepção da qualidade do software;
- \* Validar as afirmações feitas nas especificações do projeto e requisitos através de demonstrações concretas;
- \* Validar se o software trabalha da forma que foi projetado;
- \* Validar se os requisitos foram implementados apropriadamente;
- \* Validar se as regras de negócio foram atendidas;

\* Validar se a interface gráfica está em conformidade com o padrão da empresa.

- Criar Plano de Teste:

Na da atividade Criar Plano de Teste, o Gerente de Testes planeja testes e levanta os casos de uso de cada iteração para identificar os cenários de teste. Além disso, descreve os recursos de software e hardware, os recursos humanos, os riscos que podem ocorrer, identifica as telas para os testes de interface, os cenários típicos de uso e define o cronograma de execução dos testes.

Artefatos Entrada:

- \* Cenários Típicos de Uso
- \* Documento de Visão
- \* Modelo de Casos de Uso
- \* Plano de Projeto
- \* Projeto da Interface

Artefatos Saída:

- \* Plano de Testes

- Criar Casos de Teste:

Nesta atividade o Analista de Testes descreve passo a passo as ações de entrada que o testador deve executar, bem como define as saídas esperadas do sistema. Utiliza os casos de uso e as regras de negócio para criar os casos de teste.

Artefatos Entrada:

- \* Caso de Uso Real
- \* Glossário (Regras de Negócio)
- \* Plano de Testes

Artefatos Saída:

- \* Caso de Teste

- Preparar Ambiente de Teste:

Nesta atividade o Suporte Infra-estrutura prepara o ambiente para a execução de testes, de acordo com o que foi descrito no Plano de Testes.

Artefatos Entrada:

- \* Plano de Testes

Artefatos Saída:

- \* Ambiente de Testes

- Aplicar Testes:

Nesta atividade o Testador executa os cenários dos casos de teste para validar a aplicação.

Artefatos Entrada:

- \* Caso de Teste

Artefatos Saída:

- \* Não tem

- Registrar Erros:

Nessa atividade os Testadores reportam, na ferramenta de *Bugtracker*, os erros identificados na execução dos casos de teste.

Artefatos Entrada:

- \* Não tem

Artefatos Saída:

- \* *Log* de Testes (Mantis)

- Executar Testes de Integração:

Nesta atividade o Analista de Testes valida o sistema de um modo geral, verificando se o comportamento das funcionalidades relacionadas está em conformidade com as especificações dos cenários típicos que o usuário executa no seu negócio.

Artefatos Entrada:

- \* Cenários Típicos de Uso

Artefatos Saída:

\* Não tem

- Executar Testes de Interface:

Nesta atividade os Testadores validam a interface gráfica da aplicação (aparência e comportamento) para verificar se a mesma está implementada conforme o padrão de interface descrito no documento da empresa para Interface *Desktop*.

Artefatos Entrada:

\* Padrão da organização para Interfaces *Desktop*

Artefatos Saída:

\* Não tem

- Executar Testes de Aceitação:

Nesta atividade se gera um roteiro de teste embasado no artefato: Cenários Típicos de Uso. Esse roteiro é executado pelo cliente em seu ambiente de trabalho, antes que a versão final da aplicação entre em produção.

Artefatos Entrada:

\* Cenários Típicos de Execução

Artefatos Saída:

\* *Bugs* Reportados pelo Cliente

- Avaliar Resultados dos Testes:

Nesta atividade se avalia os resultados do teste, com todas as informações disponíveis, como por exemplo, *log* de teste, relatório de execução de teste, métricas de teste. Essa atividade é realizada por meio de uma reunião com todos os participantes do projeto, onde se busca identificar as falhas no processo de testes e apontar melhorias.

Artefatos Entrada:

\* *Log* de Testes (Mantis)

Artefatos Saída:

- \* Avaliação do Projeto

### **7.2.1.8 Implantação:**

O objetivo da Disciplina de Implantação é descrever as atividades associadas com:

- \* A garantia de que o produto de software seja disponibilizado a seus usuários finais;

- \* A gerência dos testes de aceitação e de homologação;

- \* A migração e/ou conversão das aplicações e bases de dados legados;

- \* O empacotamento, a distribuição e a instalação do produto;

- \* O treinamento dos usuários.

- **Projetar Material de Suporte:**

Com essa atividade pretende-se definir os materiais de suporte que serão entregues ao usuário, bem como projetar como serão esses materiais. O Analista de Negócios deve pensar na forma como os conteúdos (*help* do sistema, material de treinamento) serão construídos para que atendam as necessidades reais dos usuários. Deve também, identificar as necessidades do usuário em termos de *help* e treinamento e refletir as soluções dessas necessidades na forma de Manual de Uso da Aplicação e do Material para Treinamento e suporte.

Artefatos Entrada:

- \* *Build*

- \* Cenários Típicos de Uso

- \* Documento de Visão

- \* Projeto da Interface

- \* Manual de Uso da Aplicação (Projetado)

- \* Material de Treinamento (Projetado)

Artefatos Saída:

- \* Manual de Uso da Aplicação (Projetado)

- \* Material de Treinamento (Projetado)

- Desenvolver Material de Suporte:

Na atividade Desenvolver Material de Suporte o Editor Técnico elabora a documentação do sistema relacionada ao processo de implantação como, manuais do usuário, manuais de instalação, notas de versão (release notes), entre outros, em conformidade com o que foi definido na atividade Projetar Material de Suporte.

Artefatos Entrada:

- \* *Build*
- \* Cenários Típicos de Uso
- \* Documento de Visão
- \* Manual de Uso da Aplicação (Construído)
- \* Material de Treinamento (Construído)
- \* Projeto da Interface

Artefatos Saída:

- \* Manual de Uso da Aplicação (Construído)
- \* Material de Treinamento (Construído)

- Produzir Unidade de Implantação:

Nessa atividade o Analista de Negócios cria um pacote (coleção) de componentes executáveis, de documentos associados (manuais, release notes, etc) e instaladores. Esse pacote de arquivos e documentos compõe o produto final que deve ser entregue ao cliente.

Artefatos Entrada:

- \* *Build*
- \* Manual de Uso da Aplicação (Construído)
- \* Material de Treinamento (Construído)
- \* Plano de Implantação

Artefatos Saída:

- \* Unidade de Implantação

## **8 METODOLOGIA**

Neste Capítulo será apresentada a metodologia utilizada para desenvolver este trabalho de pesquisa, que tem como objetivo demonstrar os passos e a forma utilizada para obter o melhor resultado possível nesta pesquisa.

### **8.1 Pesquisa bibliográfica de Gestão do Conhecimento:**

Para desenvolver este trabalho, foram pesquisados autores conhecidos com várias publicações sobre Gestão do Conhecimento conforme citações feitas, tornando as informações utilizadas mais confiáveis e disponibilizando uma variedade de técnicas que serão identificadas e será escolhida a que melhor se adequar ao atual processo da empresa.

### **8.2 Pesquisa bibliográfica sobre Desenvolvimento de SW**

O atual processo de desenvolvimento de software na organização é baseado na metodologia RUP descrito neste trabalho. Utilizando esta fonte de pesquisa podemos ter conhecimento de todos os processos utilizados para desenvolver os softwares dentro da organização utilizando documentos e etapas ditadas pela metodologia RUP.

### **8.3 Definição de Metodologias de GC para serem usadas como referência**

Como citado no outro tópico, a definição da metodologia de GC utilizada neste trabalho tem como referência os seguintes autores:

A primeira a ser apresentada é de NONAKA e TAKEUCHI(1997), onde o conhecimento ocorre em quatro modos de conversão no que diz respeito ao ciclo da gestão do conhecimento.

Já conforme ARONSON apud TURBAN(2004), o ciclo de sistema de gestão do conhecimento contempla seis etapas conforme descritos neste trabalho.

Conforme DAVENPORT e PRUSAK(1999) o procedimento para se obter eficácia no gerenciamento do conhecimento empresarial pode ser sintetizado em três passos básicos.

Estas metodologias podem ser utilizadas em conjunto ou separadamente ajustadas de forma que venham a contribuir para o processo de desenvolvimento de

software administrando e armazenando o conhecimento nas etapas onde serão implementados.

#### **8.4 Levantamento e descrição do processo de desenvolvimento de sw a ser utilizado neste projeto**

O processo de desenvolvimento de software apresentado neste trabalho foi trabalhado e desenvolvido pela empresa citada com apoio de consultores e pesquisadores da área de TI da PUC/RS, entidade educacional reconhecida nacionalmente. Baseado na metodologia RUP, foram criadas etapas e documentos em um processo sólido e com qualidade para ajudar as equipes e formar um processo único para a organização.

#### **8.5 Apontar e descrever técnicas, ferramentas ou metodologias que serão utilizadas para implantar GC no ciclo de DS**

Além das técnicas e ferramentas propostas pelos autores pesquisados, foram identificadas ferramentas no atual ambiente organizacional que podem colaborar com o processo de implantação de GC no ambiente de DS. Dentre estas ferramentas estão a *intranet* e repositório de dados em servidores na rede que poderão ser aproveitados para o trabalho em questão.

#### **8.6 Experimento ou validação da proposta de GC que será utilizada no ciclo de DS**

Para validação da experiência proposta neste trabalho, serão consultados alguns funcionários envolvidos no atual processo de desenvolvimento de software, estes serão questionados pela resposta positiva ou não da implementação de GC nos pontos importantes selecionados no processo de DS.

Dependendo da resposta destes *stakeholders* saberemos se o trabalho realizado está no caminho correto, caso não esteja, será reavaliado e sugerido uma nova proposta podendo dar seqüência ao trabalho até a sua conclusão.

#### **8.7 Elaboração de um relatório final com os resultados obtidos**

Ao final do trabalho, no TCC II será apresentado um relatório com os resultados obtidos que servirão para concluir a avaliação e ter uma maior clareza do

processo executado demonstrando as mudanças positivas, ou não, e os benefícios alcançados com este trabalho.

## 9 RESULTADOS

Com este trabalho foi possível mapear o processo de desenvolvimento de software que será utilizado e também foram realizados vários estudos sobre Gestão do Conhecimento, suas técnicas, ferramentas e metodologias que unidas ou separadamente serão utilizadas para tratar os pontos identificados na próxima etapa de experiências no TCC II.

O resultado esperado considera os benefícios que a metodologia de GC aplicada ao processo de DS baseado na metodologia RUP, sejam positivos e que o conhecimento mapeado possa agregar os processos utilizados nas tarefas diárias e nos projetos desenvolvidos, tornando o conhecimento disponível a todos interessados principalmente a organização.

## 10 TRABALHO DE CONCLUSÃO DE CURSO - II

Depois de mapeado o processo de DS a ser utilizado, serão identificados os pontos onde poderão ser utilizados aspectos de GC para melhoria deste processo de DS. Após identificados estes pontos serão indicadas ferramentas ou técnicas de GC para melhorar a aquisição, administração, compartilhamento e uso das informações e conhecimentos que estão associados a estes pontos do processo de GC.

Para identificar estes pontos dentro do processo de DS a serem melhorados, serão considerados os seguintes aspectos:

- Tarefas ou atividades que envolvem mais de um *stakeholder*.
- Tarefas ou atividades que ainda não possuam artefatos definidos.
- Tarefas onde o conhecimento é compartilhado somente de forma tácita.
- Pontos que podem ser fontes de conhecimento para um banco de lições aprendidas (Boas práticas), tanto do ponto de vista de quem gera ou de quem irá consultar.

Através dos pontos identificados anteriormente pretende-se avaliar as situações mais críticas para a empresa em uma situação de saída de funcionários. Após identificadas estes pontos serão escolhidos as formas para melhor tratar o conhecimento nestas situações considerando o tipo de conhecimento, técnica e ferramentas estudadas e pessoas envolvidas.

Estas sugestões que serão apresentadas p/ melhorar a GC no ciclo de DS deverão impactar o mínimo possível nas tarefas diárias dos funcionários, pois se alterar muito a rotina dos profissionais existe a tendência destes acabarem rejeitando e não realizando as tarefas associadas à GC.

## **11 REFERÊNCIAS**

DAVENPORT, Thomas H; PRUSAK, Laurence. Conhecimento Empresarial. Rio de Janeiro: Editora Campus, 1999.

NONAKA, Ikujiro; TAKEUCHI, Hirotaka. Criação de conhecimento na empresa: como as empresas japonesas geram a dinâmica da inovação. Rio de Janeiro: Editora Campus, 1997.

PROBST, Gilbert; RAUB, Steffen e ROMHARDT, Kai. Gestão do Conhecimento: os elementos construtivos do sucesso. Porto Alegre: Bookman, 2002.

RATIONAL UNIFIED PROCESS®. Versão 2002.05.00, Copyright © 1987 – 2001, Rational Software Corporation.

Rus, Ioana; Lindvall, Mikael (2002). "Guest Editors' Introduction: Knowledge Management in Software Engineering" In:IEEE Software, vol. 19, no. 3, pp. 26-38, May/Jun, 2002.

STEWART, Thomas. Capital intelectual: a nova vantagem competitiva das empresas. Rio de Janeiro: Campus, 1998.

TURBAN, Efraim; MClean, Ephraim;Wetherbe, James. Tecnologia da Informação para Gestão. Porto Alegre: Bookman, 2004.

The Unified Modeling Language User Guide, Booch, Rumbaugh, Jacobson, Addison-Wesley, 1999