|  |
| --- |
| Calandra soluções  Padrão |
| PC 003 |
| Modelo de Desenvolvimento de Sistemas de Informação |
| Primeira edição |
| 00/00/0000 |
|  |

|  |
| --- |
| [Resumo] |

Conteúdo

[1. Introdução 2](#_Toc404868309)

[2. Visão geral 2](#_Toc404868310)

[2.1. Escopo 2](#_Toc404868311)

[2.2. Propósito 2](#_Toc404868312)

[2.3. Limitações 2](#_Toc404868313)

[2.4. Dependências 2](#_Toc404868314)

[3. Termos e definições 2](#_Toc404868315)

[4. Aplicação deste documento 4](#_Toc404868338)

[4.1. Conceitos de sistema 4](#_Toc404868339)

[4.1.1. Sistema 4](#_Toc404868340)

[4.1.2. Relação entre eistema e ETL 5](#_Toc404868341)

[4.1.3. Estrutura de sistema 6](#_Toc404868342)

[4.1.4. Ambiente operacional 6](#_Toc404868343)

[4.1.5. Sistemas habilitadores 7](#_Toc404868344)

[4.2. Conceitos de ciclo de vida 7](#_Toc404868345)

[4.2.1. Modelo de ciclo de vida 7](#_Toc404868346)

[4.2.2. Estágios do ciclo de vida 7](#_Toc404868347)

[4.3. Conceitos de Processo 8](#_Toc404868348)

[4.3.1. Critério para classificação 8](#_Toc404868349)

[4.3.2. Descrição do processo versus implementação de processo 8](#_Toc404868350)

[4.3.3. Descrição do processo 8](#_Toc404868351)

[4.3.4. Implementação do processo 8](#_Toc404868352)

[4.3.5. Aplicação de processos 9](#_Toc404868353)

[4.3.6. Artefatos 10](#_Toc404868354)

[4.3.7. Exceção 12](#_Toc404868355)

[4.4. Organização deste padrão 12](#_Toc404868356)

[4.5. Uso 14](#_Toc404868357)

[5. Descrição de Processos de Ciclo de Vida de ETL 14](#_Toc404868358)

[5.1. Processos de Projeto 14](#_Toc404868359)

[5.1.1. Processo de Planejamento de Projeto 14](#_Toc404868360)

[5.1.2. Processo de Avaliação e Controle de Projeto 15](#_Toc404868361)

[5.2. Processos Primários 16](#_Toc404868362)

[5.2.1. Processo de Definição de Requisitos do Cliente 16](#_Toc404868363)

[5.2.2. Processo de Análise de Requisitos de Sistema 17](#_Toc404868364)

[5.2.3. Processo de Implementação 18](#_Toc404868365)

[5.2.4. Processo de Implementação de ETL 19](#_Toc404868366)

[5.2.5. Processo de Análise de Requisito de ETL 19](#_Toc404868367)

[5.2.6. Processo de Design Detalhado de ETL 20](#_Toc404868368)

[5.2.7. Processo de Construção de ETL 21](#_Toc404868369)

[5.2.8. Processo de Integração de ETL 22](#_Toc404868370)

[5.2.9. Processo de Teste de Qualificação de ETL 23](#_Toc404868371)

[5.3. Ciclo de vida 24](#_Toc404868372)

[5.3.1. Mecânica do ciclo de vida 24](#_Toc404868373)

[5.3.2. Das tarefas 25](#_Toc404868374)

[5.3.3. Dos estágios 25](#_Toc404868375)

# Introdução

# Visão geral

## Escopo

## Propósito

## Limitações

## Dependências

# Termos e definições

## ambiente operacional

ambiente que suporta o sistema-de-interesse durante a sua operação mas não faz parte de sua especificação

**NOTA -** Um sistema sempre é desenvolvido para operar em uma ambiente operacional definido.

## arquitetura

organização fundamental de um sistema incorporada em seus componentes, as relações entre eles e o com o ambiente, e os princípios que guiam seu design e evolução.

## artefato

corpo de informações tangível e separadamente identificável produzido durante o ciclo de vida de um sistema

## baseline

especificação ou produto que foi formalmente revisto e aceito, que serve como base para progressão do desenvolvimento, e que só pode ser alterado através de procedimentos formais de controle de mudança.

## cliente

indivíduo ou organização tendo direito, parte, reivindicação ou interesse em um sistema, ou no seu conjunto de características, que atendam suas necessidades e expectativas

## elemento de sistema

membro de um conjunto de elementos que constituem um sistema

## ETL

produto de software especializado para soluções de extração, transformação e carga de dados

## estágio

período dentro do ciclo de vida de uma entidade que se relaciona com o estado de sua descrição ou realização

## item de informação

artefato de uso humano

## item de configuração

entidade dentro de uma configuração que satisfaz uma função de uso final e que pode ser identificada univocamente em um dado ponto de referência

## item de software

código fonte, código de objeto, código de controle, dados de controle, ou um conjunto desses itens

**NOTA -** Um item de software pode ser visto como um elemento de sistema.

## produto de software

conjunto de programas de computador, procedimentos e, possivelmente, documentação e dados associados

## processo

conjunto de atividades inter-relacionadas que transformam entradas em saídas

## propósito do processo

objetivo de alto nível da execução de um processo e os resultados esperados da implementação efetiva do processo

## recurso

ativo que é utilizado ou consumido durante a execução de um processo

## resultado do processo

resultado observável da realização bem sucedida do propósito de um processo

## sistema

combinação de elementos que se relacionam organizados a fim de atender á um ou mais propósitos declarados

**NOTA -** Um sistema pode ser considerado como um produto ou como o serviço que ele oferece.

## sistema-de-interesse

sistema cujo ciclo de vida está sob consideração no contexto deste documento

## sistema habilitador

sistema que suporta o sistema-de-interesse durante o seu ciclo de vida mas que não contribui necessariamente, de forma direta, com o seu funcionamento durante operação

## teste de qualificação

teste conduzido pelo desenvolvedor, e testemunhado pelo adquirente, para demonstrar que um produto de software atende sua especificações e está pronto para uso no seu ambiente operacional ou para integração com o sistema que o contém

## unidade de ETL

pedaço de ETL separável e funcional

## unidade de software

pedaço de código separadamente compilável

# Aplicação deste documento

Está cláusula apresenta uma visão geral do Modelo de Desenvolvimento de Sistemas de Informação que pode ser usado para adquirir, fornecer, desenvolver, operar, manter, gerenciar e controlar sistema de informação. O objetivo é fornecer um mapa para os usuários deste modelo para que eles possam se orientar e aplicá-lo.

## Conceitos de sistema

### Sistema

Os sistemas considerados neste modelo são artefatos, criados e utilizados para prover produtos e/ou serviços em um ambiente definido para o benefício de usuários e outras partes interessadas. Estes sistemas podem estar configurados com um ou mais dos seguintes elementos de sistema: hardware, software, dados, humanos, processos (e.g., processo para prover um serviço à um usuário), procedimento (e.g., instruções de operação), instalações, materiais e entidades naturais. Na prática são vistos como produtos ou serviços.

A percepção e definição de um sistema em particular, sua arquitetura, e seus elementos de sistema, dependem dos interesses e responsabilidades do observador. O sistema-de-interesse duma pessoa pode ser visto com um elemento de sistema na percepção doutra pessoa. Ou, além disso, um sistema-de-interesse pode ser visto como parte do ambiente operacional do sistema-de-interesse doutra pessoa.

Seguem pontos-chave concernentes às características do sistema-de-interesse

* fronteiras definidas encapsulam necessidades significativas e soluções práticas;
* existe uma relação, hierárquica ou outra, entre os elementos de sistema;
* uma entidade em qualquer nível do sistema-de-interesse pode ser vista como um sistema;
* um sistema compreende um conjunto definido e integrado de elementos de sistema subordinados;
* propriedades características na fronteira do sistema surgem das interações entre os elementos de sistema;
* humanos podem ser vistos como usuários externos ao sistema e como elementos de sistema (i.e., operadores) dentro do sistema;
* um sistema pode ser visto isoladamente como uma entidade, i.e., um produto, ou como uma coleção de funções capazes de interagir com seu ambiente circundante, i.e., um conjunto de serviços.

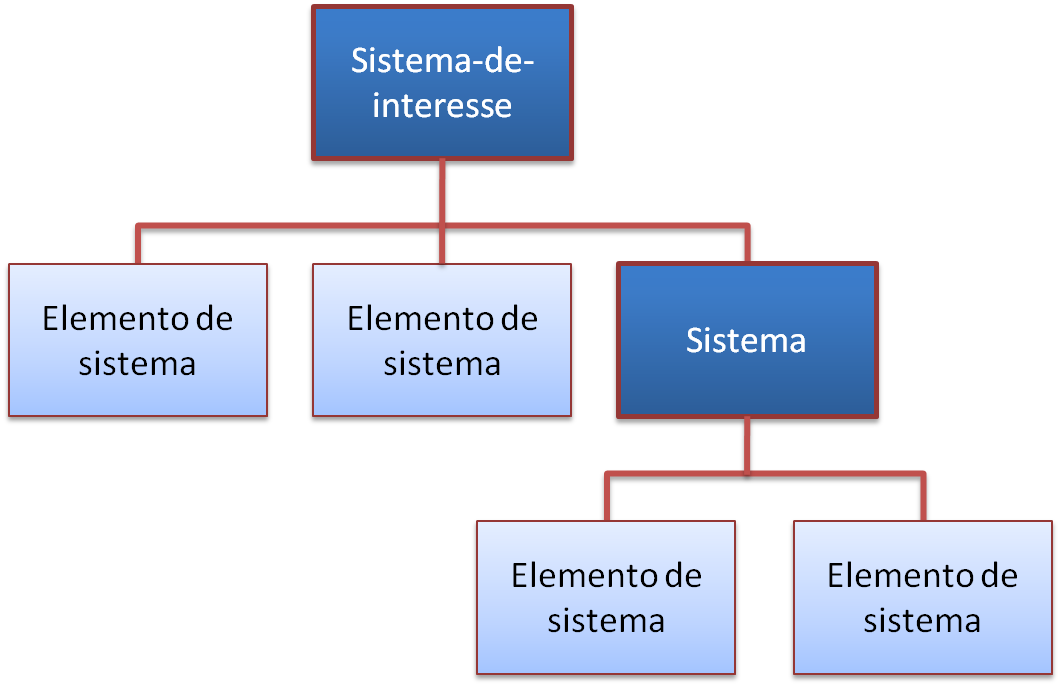


Figura - Estrutura do sistema-de-interesse

### Relação entre sistema e software

Este modelo estabelece uma forte relação entre sistema e software. É com base num princípio geral da engenharia de sistemas. Software é visto como uma parte integrante do total do sistema e executa certas funções nesse sistema. Isto é realizado através da extração de requisitos de software dos requisitos de sistema, produção do software e integração do software com o restante do sistema. É uma premissa fundamental deste modelo que um software sempre existe no contexto de um sistema, mesmo que o sistema seja somente o computador no qual o software é executado. Portanto, um software é sempre tratado como um elemento de sistema em um sistema.

Este padrão foi concebido no âmbito de atender sistemas em que os elementos não-software sejam mínimos e que não seja necessário sua implementação.

### Estrutura de sistema

Este modelo é descrito com relação à um sistema que é composto de um conjunto de elementos de sistema inter-relacionados, no qual cada um pode ser implementado para cumprir seus respectivos requisitos. A responsabilidade de implementar qualquer elemento de sistema pode, portando, ser delegada para outra organização através de um acordo (e.g., a implementação de um algoritmo de clusterização).

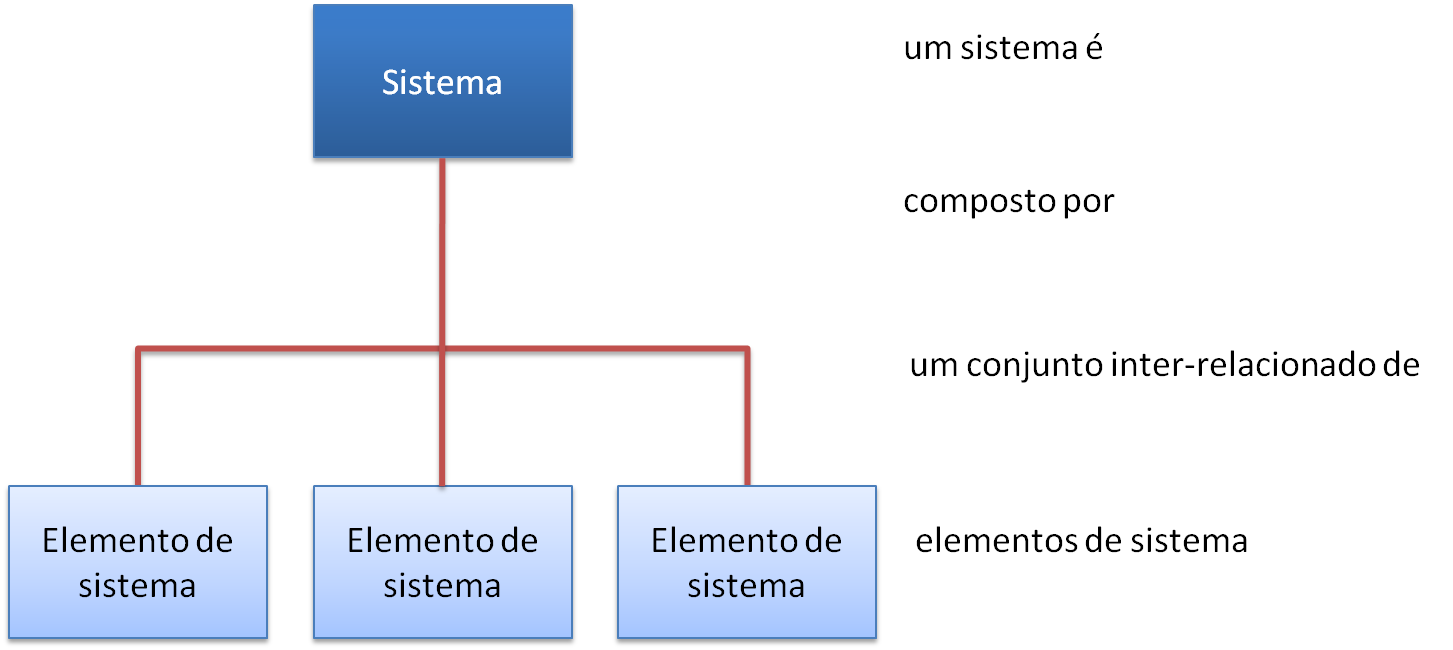


Figura - Relacionamento entre sistema e elemento de sistema

### Ambiente operacional

Todo sistema é desenvolvido para operar em um ambiente operacional específico. O desenvolvimento do ambiente operacional não faz parte do escopo do desenvolvimento do sistema. Porém, pode ser necessário a especificação do ambiente operacional do sistema-de-interesse. Para um sistema operar é necessário a instanciação de seu ambiente operacional correspondente e a implantação do sistema neste ambiente.

O ambiente operacional de um elemento de sistema sempre está contido no sistema que o contém.

Figura - Sistema e seu ambiente operacional

### Sistemas habilitadores

Durante o ciclo de vida de um sistema-de-interesse, serviços essenciais são necessários de sistemas que não fazem parte diretamente do ambiente operacional (e.g., Redmine), ou do sistema-de-interesse em si. Cada um desses sistema habilitam que uma parte, e.g. um estágio do ciclo de vida, seja conduzida. Nomeados de "sistemas habilitadores", ele facilitam a progressão do sistema-de-interesse durante seu ciclo de vida.

A relação entre os serviços entregues ao ambiente operacional pelo sistema-de-interesse e os serviços entregues pelos sistemas habilitadores para o sistema-de-interesse são mostrados na Figura 3. Sistemas habilitadores contribuem indiretamente para os serviços providos pelo sistema-de-interesse. O relacionamento pode ser bidirecional ou unidirecional.

Figura - Interação do sistema-de-interesse com o ambiente operacional e sistemas habilitadores

## Conceitos de ciclo de vida

### Modelo de ciclo de vida

Todo sistema tem um ciclo de vida. O ciclo de vida pode ser descrito usando um modelo abstrato funcional, como o deste documento, que represente a concepção, realização, utilização e evolução do sistema.

Um sistema progride pelo ciclo de vida como conseqüência de ações, executada,s e gerenciadas por pessoas, usando processos para a execução dessas ações. O detalhe no modelo de ciclo de vida é expressado em termo de processos, seus resultados, relacionamentos e seqüência. Este documento define um modelo de ciclo de vida, um conjunto de processos e artefatos utilizados, e seus relacionamentos.

### Estágios do ciclo de vida

Ciclos de vida variam de acordo com a natureza, propósito, uso e circunstâncias prevalentes do sistema. Cada estágio tem um propósito e contribuição distinta para o ciclo de vida e são considerados durante o planejamento, gerenciamento e execução do ciclo de vida do sistema.

**NOTA -** A definição de estágio do ciclo de vida é conceitualmente similar ao termo "marco crítico". No contexto deste documento, este termo não será usado para manter coerência com os padrões e literatura sobre engenharia de sistemas.

Os estágios representam os períodos do ciclo de vida mais relevantes associados com um sistema e eles informam o estado da descrição do sistema ou do próprio sistema. Eles geram pontos de decisão primários do ciclo de vida. Esses pontos de decisão são usados por organizações para entender e gerenciar as incerteza e riscos inerentes associadas com o custo, cronograma e funcionalidade quando criando ou utilizando um sistema. Portanto, os estágio contribuem com um framework que provê ao gerenciamento da organização uma visibilidade e controle de alto nível sobre projeto.

## Conceitos de Processo

### Critério para classificação

Este modelo estabelece um framework de processos para o ciclo de vida de sistemas de informação. O ciclo de vida se inicia com uma idéia ou necessidade que pode ser atendida integral ou parcialmente por um ETL e termina com retirada do ETL. A arquitetura é construída com um conjunto de processos e inter-relações entre esses processos. A determinação dos processos de ciclo de vida é baseada em dois princípios básicos: coesão e responsabilidade.

**Coesão**: Os processos do ciclo de vida são coesos e acoplados na medida ótima considerada prática e viável.

**Responsabilidade**: Um processo é colocado sobre a responsabilidade de uma organização durante o ciclo de vida de sistemas de informação.

### Descrição do processo versus implementação de processo

Este padrão separa a descrição do processo da implementação do processo. O objetivo desta separação é isolar o propósito do processo, ou seja, o que o processo se dispõe a fazer, da forma como o processo vai atender a esse propósito. Isso facilita caso o usuário deste padrão queira estender, modificar ou reusar este padrão.

### Descrição do processo

A descrição do processo diz o que o processo deve fazer, e quais são os objetivos do processo. A descrição do processo não diz como o processo deve ser executado. Os processos neste modelo são descritos em termos dos seguintes atributos:

* O título transmite o escopo do processo como um todo
* O propósito descreve os objetivos da execução do processo
* Os resultados expressam as realizações observáveis esperadas de uma execução bem sucedida do processo
* As atividades são lista de ações que são usadas para atingir os resultados do processo

### Implementação do processo

A implementação do processo diz detalhadamente como um processo deve ser executado. As atividades e tarefas expostas na descrição do processo não implicam em um mapeamento unívoco com a implementação do processo. O usuário é desencorajado a tentar estabelecer relações entre as atividades e tarefas expostas na descrição do processo com as atividades e tarefas expostas na implementação do processo. O usuário deste padrão é encorajado porém, a recorrer a descrição para entendimento do escopo do processo e para avaliação de conformidade de processo.

A implementação do processo segue as seguintes regras:

* Cada processo consiste de uma ou mais atividades.
* Cada atividade contida em um processo fornece um serviço, e tem somente um ponto de entrada.
* As atividades de um processo podem instanciar outras atividades do mesmo processo.
* Um processo pode conter sub-processos.
* Todo processo é sub-processo, em algum nível, do Desenvolvimento de Sistemas de Informação.
* Um processo pode instanciar atividades de um sub-processo.
* As atividades de um processo podem levantar eventos para o processo invocador.
* As atividades de um processo podem capturar eventos do processo invocador.
* Uma atividade ou tarefa consome artefatos de entrada e fornece artefatos de saída.
* O Desenvolvimento de Sistemas de Informação estabelece uma relação seqüencial entre as atividades de seus sub-processos.
* O Desenvolvimento de Sistema de Informação representa o projeto e, no momento de sua instanciação, deve ser renomeado para Projeto [*nome do projeto*].

### Aplicação de processos

A ordem em que os processos são apresentadas neste documento não implicam nenhuma ordem prescritiva no seu uso. A ordem dos processos e suas atividades e tarefas estão arranjadas numa forma apropriada para exposição. Porém, as relações seqüências são introduzidas pelo modelo de ciclo de vida. A hora de ativação e propósito detalhado desses processos durante o ciclo de são influenciados por vários fatores, incluindo sociais, mudanças, considerações organizacionais e técnicas, onde cada fator pode variar durante o ciclo de vida de um sistema. O ciclo de vida individual de um sistema é, portanto, um sistema complexo de processos que irá normalmente apresentar características de concorrência, iteratividade, recursividade e dependências temporais.

Uso concorrente de processos pode existir dentro de um projeto (e.g., uma unidade ETL começar a ser construída antes de design estar pronto), e entre projetos (e.g., quando elementos de sistemas são desenhados ao mesmo tempo sobre diferentes projetos).

Quando a aplicação do mesmo processo ou atividades ou conjunto de processo é repetida no mesmo sistema, a aplicação é dita como iterativa. O uso iterativo de processos e atividades é importante para o refinamento progressivo das saídas dos processos ou atividades, e.g., a iteração entre ações de integração e teste de qualificação podem construir confiança e conformidade no produto incrementalmente. Iteração não é somente apropriada como também esperada. Novas informações são criadas pela aplicação de um processo ou um conjunto de processos. Normalmente estas informações tem a forma de questionamentos (e.g., sobre o design), conflitos de requisitos, riscos ou oportunidades.

O uso recursivo de processos, i.e., a aplicação repetida do mesmo processo ou conjunto de processos aplicados à níveis sucessíveis de elementos de sistema na estrutura do sistema, é um aspecto chave deste modelo. As saídas dos processos em qualquer nível, seja informação, artefatos, ou serviços, são entradas para os mesmos processos usados no nível abaixo (e.g., design top-down) ou no nível acima (e.g., durante a integração do sistema). Os resultados de uma aplicação são usados como entradas no próximo nível mais baixo (ou alto) de sistema na estrutura do sistema para alcançar um nível mais detalhado ou maduro de resultados. Tal abordagem acrescenta valor para os sistemas sucessíveis na estrutura do sistema.

### Artefatos

Artefatos são produtos de trabalho associados com a execução de uma atividade ou tarefa. Existem dois tipos genéricos de artefatos: software (e.g. programa de computador, documentos, informação, conteúdo); hardware (e.g. computador, dispositivo).

Neste padrão, os artefatos podem estar em três estados distintos e excludentes: ativo, representado ou congelado.

1. Artefato ativo

Este estado representa que o artefato é acessível porém sujeito a evolução. A atividade ou tarefa responsável pela criação do artefato ainda está em execução. O artefato pode não estar completo. O artefato sempre se apresenta neste estado, a não ser que seja explicitamente dito o contrário.

1. Artefato congelado

Este estado representa que artefato é acessível porém a atividade responsável por ele foi finalizada. O artefato só pode ser modificado através de procedimentos formais de controle de alteração, que incluem a reabertura da atividade responsável.

1. Artefato representado

Este estado significa que o artefato existe virtualmente através de um representante humano. O propósito declarado do artefato deve ser completamente atendido por este representante em qualquer instante de tempo.

**NOTA -** Somente itens de informação podem ser encontrados no estado representado.

O objetivo dos estados de artefato ativo e realizado é permitir agilidade e flexibilidade durante o ciclo de vida sem infringir a conformidade com este modelo. Porém, o consumo de artefatos nestes estados não é recomendado por este padrão e deve ser adotado criteriosamente durante o ciclo de vida.

No contexto deste documento a categorização de artefatos de software é estendida em dois subconjuntos principais: itens de informação e itens de software.

#### Itens de Informação

Itens de informação são produtos de processo de particular interesse para o gerenciamento do ciclo de vida. Itens de informação são corpos de informação separadamente identificáveis que são produzidos e armazenados para uso humano durante o ciclo de vida de um sistema. Em outras palavras, eles são entradas e saídas dos processos de ciclo de vida, e são transformados por estes processos.

A descrição de um item de informação consiste de um nome, um conjunto de características, opcionalmente um template, e um item de informação parente, de onde ele herda suas características.

**Nome do item de informação:** O nome serve como um identificador do tipo de item de informação que um processo pode produzir.

**Características do item de informação:** As características potencias associadas com um tipo de item de informação. Características podem descrever o propósito e uso de um item de informação, seu conteúdo, formato e qualidade.

**Template do item de informação:** Define a estrutura, formato e conteúdo parcial de um item de informação. Uma instância de um item de informação deve preencher o restante do template nas zonas indicadas. O início e o fim dessas zonas são indicados por "[" "]", respectivamente.

**Parente do item de informação:** Um item de informação pode herdar características e template de outro item de informação. O parente de um item de informação é o nome do item de informação do qual ele herda suas características e seu template, podendo estes, serem estendidos. Todos os itens de informação herdam diretamente, ou indiretamente de item de informação.

Este documento define e usa tipos genéricos de informação que são estendidos conforme necessário. Qualquer item de informação usado durante o ciclo de vida definido por este padrão é uma instância de um desses tipos genéricos ou uma instância de um tipo especializado que tem um desses tipos genéricos como parente.

|  |  |
| --- | --- |
| **Nome do item de informação** | **Características do item de informação** |
| Descrição | Um relato ou representação de um conceito ou objeto proposto ou atual. Pode ser textual, gráfico ou matemático. Pode ser um formulário padronizado para uso humano ou interpretação de máquina. Pode ser estático ou um modelo dinâmico ou uma simulação. Pode estabelecer ordem, estrutura, agrupamento ou classificação. |
| Plano | Um esquema proposto ou fluxo de ações sistemático com o fim de atingir um propósito declarado. Prevê como alcançar com sucesso objetivos em termos de ações específicas, executadas em tempos definidos que utilizam recursos definidos. Podem ser ações técnicas, de projeto, ou empresariais. Em um alto nível de abstração, pode seu uma política ou uma estratégia. |
| Procedimento | Uma forma declarada de conduzir um fluxo de ações rotineiro. Define um jeito ou modo estabelecido e aprovado de conduzir um negócio em uma organização. Pode detalhar métodos permissíveis ou recomendados a fim de atingir metas ou resultados técnicos ou gerencias. |
| Registro | Uma forma permanente e legível de informação ou conhecimento. Evidência, mantida e acessível, da existência ou ocorrência de fatos, eventos ou transações. Pode ter a forma de um jornal cronológico, um registro, ou um arquivo. |
| Relatório | Um relato preparado para organizações interessadas, a fim de comunicar estado, resultados ou conseqüências. É resultado da coleta, observação, investigação ou avaliação de informações e pode transmitir situação, impacto, progresso ou realização. Serve como fonte de informação para que decisões ou atos subseqüentes possam ser acionados. |
| Requisição | Uma comunicação que inicia um curso de ação definido ou mudança a fim de atender uma necessidade. Pode originar ou controlar uma ação em curso com base em um plano ou procedimento acordado. Pode tomar a forma de uma solicitação, requisição, instrução, ou demanda por um recurso, serviço ou ato de aprovação. |
| Especificação | Critérios ou condições que colocam limites ou restrições em ações, atributos ou qualidades. Pode estabelecer medidas ou qualidades que determinam aceitação, conformidade ou mérito. Pode ser requisitada como parte de um acordo ou contrato. |

### Exceção

Exceções são eventos que podem ocorrer durante a execução de um processo, que necessitam de tratamento especializado. Este documento não define qual plano de ação deve ser executado para o tratamento da exceção. Este documento prevê, porém, quais ações são necessárias para o gerenciamento e controle de um fluxo de exceção.

Toda instância de uma exceção consiste de um tipo, uma tarefa aonde a exceção foi detectada, e uma descrição. Este documento define um conjunto de tipos de exceção. Qualquer exceção que não se encaixe nesse conjunto deve ser considerada uma exceção do tipo genérica.

#### Tipos de exceção

##### Exceção de artefato:

Ocorre quando um artefato que é consumido por um processo ou tarefa não atende ao seu propósito declarado.

##### Exceção de conformidade:

Ocorre quando uma falta de conformidade com este padrão é detectada durante o ciclo de vida.

##### Exceção conceitual:

Ocorre quando um conflito conceitual é detectado. Pode ser causada por incongruência, inconsistência, ausência, incompletude ou ambigüidade.

##### Exceção de viabilidade:

Ocorre quando um processo ou tarefa não poderá ser concluído, ou será consideravelmente atrasado, devido a um impasse técnico ou gerencial não previsto.

## Organização deste padrão

* Processos de Suporte
  + Processo de Gerenciamento de Infraestrutura
    - Estabelecimento de Infraestrutura
    - Manutenção de Infraestrutura
* Processos de Planejamento
  + Processo de Planejamento
    - Inicialização de Projeto
    - Planejamento de Projeto
    - Identificação de Participante
    - Ativação de Plano de Projeto
  + Processo de Avaliação e Controle de Projeto
    - Controle de Alteração
    - Rastreamento de Projeto
    - Análise de Exceção
    - Ativação de Plano de Intervenção
  + Processo de Gerenciamento de Configuração
    - Identificação de Ambiente
    - Versionamento
* Processo Técnicos
  + Processo de Definição de Requisitos de Cliente
    - Definição de Requisitos de Cliente
    - Consolidação de Alterações
  + Processo de Implementação de ETL
    - Construção de ETL
    - Design Arquitetural de ETL
    - Integração de ETL
    - Análise de Requisitos de Qualificação de ETL
    - Análise de Requisitos de ETL
  + Processo de Análise de Requisitos de Sistema
    - Definição de Requisitos de Sistema
  + Processo de Operação de Sistema
    - Depuração de Dados
    - Harmonização de Dados
    - Depuração de Sistema
    - Carga de Dados
    - Construção de Alvo
    - Carga de Metas
    - Carga de Plano de Ação
    - Assessoria
  + Processo de Operação de ETL
    - Carga de Dados
  + Processo de Implantação de ETL
  + Processo de Implantação de Sistema
  + Processo de Implementação de Sistema
  + Processo de Resolução de Problemas
    - Resolução de Problemas

## Uso

# Descrição de Processos de Ciclo de Vida de ETL

## Processos de Projeto

### Processo de Planejamento de Projeto

#### Propósito

O propósito do Processo de Planejamento de Projeto é produzir e comunicar planos de projetos efetivos e executáveis.

#### Resultados

1. o escopo do trabalho para o projeto é definido;
2. viabilidade de atingir os objetivos do projeto, com as limitações e os recursos disponíveis, é avaliada;
3. tarefas e recursos necessários para completar o trabalhado são medidos e estimados;

#### Atividades e tarefas

##### Inicialização de projeto:

###### Uma vez estabelecido os requisitos do projeto, o gerente deve estabelecer a viabilidade do projeto verificando se os recursos (pessoas, materiais, tecnologia, e ambiente) requeridos para a execução e gerenciamento do projeto estão disponíveis, são adequados, são apropriados, e que o cronograma para a conclusão do projeto é possível.

###### Conforme necessário, os requisitos do projeto podem ser modificados neste ponto para atender o critério de completude.

##### Planejamento de projeto:

###### O gerente deve preparar planos de execução para o projeto. Os planos associados com a execução do projeto devem conter descrições das atividades e tarefas associadas e identificação dos sistemas que serão entregues. Estes planos devem incluir, pelo menos:

1. Cronograma.
2. Estimativa de esforço.
3. Recursos necessários.
4. Alocação das tarefas.
5. Atribuição de responsabilidades.
6. Fornecimento de infra-estrutura e ambiente.
7. Definição do modelo de ciclo de vida.

##### Ativação de projeto:

###### O gerente deve obter autorização para o projeto.

###### O gerente deve requisitar os recursos necessários para a execução do projeto.

###### O gerente deve iniciar a implementação do plano(s) de projeto e exercer controle sobre o projeto.

### Processo de Avaliação e Controle de Projeto

#### Propósito

O propósito do Processo de Avaliação e Controle de Projeto é determinar o estado do projeto e garantir que a execução do projeto está de acordo com os planos e cronogramas, e dentro do custo, e que satisfaz objetivos técnicos.

Este processo inclui redirecionamento de atividades de projeto, conforme apropriado, para corrigir desvios identificados e variações de outros processos. Redirecionamento pode incluir replanejamento conforme necessário.

#### Resultados

1. progresso do projeto é monitorado e reportado;
2. ações para corrigir desvios do plano e prevenir recorrência de problemas identificados no projeto são tomadas quando objetivos de projeto não são alcançados; e
3. objetivos de projeto são alcançados e registrados.

#### Atividades e tarefas

##### Monitoramento de projeto:

###### O gerente deve monitorar a execução geral do projeto, providenciando relatórios internos de andamento do projeto e externos para o adquirente conforme definido em contrato.

##### Controle de projeto:

###### O gerente deve investigar, analisar, e resolver os problemas descobertos durante a execução do projeto. A resolução de problemas pode resultar em alteração do planos. É responsabilidade do gerente garantir que o impacto de qualquer mudanção é determinado, controlado e monitorado. Problemas e suas soluções devem ser documentados.

###### O gerente deve reportar, em pontos e acordados, o progresso do projeto, declarando aderência com os planos e resolvendo faltas de progresso. Isto inclui relatos internos e externos, conforme definido por procedimentos organizacionais e o contrato.

###### 

## Processos Primários

### Processo de Definição de Requisitos do Cliente

#### Propósito

O propósito do Processo de Definição de Requisitos do Cliente é definir os requisitos para um sistema que possa fornecer os serviços necessários aos usuários e outras partes interessadas, em um ambiente definido.

Identifica as partes interessadas, ou classes destas, envolvidas no ciclo de vida do sistema, suas necessidades e seus desejos. Analisa e transforma estes em um conjunto comum de requisitos de cliente que expressam a interação pretendida que o sistema vai ter com o seu ambiente operacional e que servem de referência para validar cada serviço operacional resultante, confirmando assim, que o sistema atende as necessidades.

#### Resultados

1. As características requeridas e o contexto de uso do serviço são especificados;
2. As limitações na solução do sistema são definidas;
3. A base para definição de requisitos de sistema é descrita; e
4. A base para validação de conformidade do serviço é definida.

#### Atividades e tarefas

##### Identificação de requisitos:

###### O projeto deve extrair os requisitos.

Requisitos descrevem as necessidades, desejos e expectativas dos clientes e as limitações percebidas, ou impostas, por eles.

O projeto deve definir as limitações da solução que são conseqüências inevitáveis de acordos existentes, decisões gerenciais e decisões técnicas.

Isso pode ser conseqüência de:

1. Domínio definido pelo cliente;
2. Decisões de implementação feitas em níveis mais altos da estrutura hierárquica do sistema
3. Sistemas habilitadores predefinidos, recursos e pessoal.

##### Avaliação de requisitos:

###### O projeto deve analisar o conjunto completo de requisitos extraídos.

Análise inclui identificar e priorizar os requisitos conflitantes, faltantes, incompletos, ambíguos, inconsistentes, incongruentes ou inverificáveis.

##### Aprovação de requisitos:

###### O projeto deve resolver problemas de requisitos.

Isso inclui requisitos inexeqüíveis ou impraticáveis.

###### O projeto deve atualizar o cliente com o resultado da análise a fim de garantir que suas necessidades e desejos foram adequadamente capturados e expressados.

Explicar e obter aprovação para as propostas que resolvem requisitos conflitantes, impraticáveis e inexeqüíveis.

##### Registro de requisitos:

###### O projeto deve registrar os requisitos em uma forma apropriada para o gerenciamento durante o ciclo de vida do sistema.

### Processo de Análise de Requisitos de Sistema

#### Propósito

O propósito do Processo de Análise de Requisitos de Sistemas é transformar os requisitos de cliente em um conjunto de requisitos técnicos de sistema que irá guiar o design do sistema.

#### Resultados

1. Um conjunto definido de requisitos de sistema, funcionais e não-funcionais, que descrevem o problema a ser resolvido, é estabelecido;
2. Os requisitos de sistemas são analisados a fim de correção e testabilidade;
3. Os requisitos são priorizados, aprovados e atualizados conforme necessário;
4. Mudanças no baseline são avaliadas a fim de custo, cronograma e impacto técnico; e
5. Os requisitos de sistema são passados para todas as partes interessadas e congelados;

#### Atividades e Tarefas

##### Especificação de requisitos:

###### O uso específico pretendido do sistema deve ser analisado para especificar os requisitos do sistema. Os requisitos do sistema devem descrever: funções e capacidades do sistema; requisitos de negócio, organizacionais e de usuário; limitações no design e requisitos de qualificação. A especificação de requisitos do sistema deve ser documentada.

Requisitos de sistema devem ser priorizados, aprovados, congelados e informados para todos os interessados. Mudanças no baseline devem ser avaliadas a fim de custo, cronograma e impacto técnico.

##### Avaliação de requisitos:

###### Os requisitos de sistema devem ser avaliados considerando os critérios abaixo. O resultado da avaliação deve ser documentado.

###### Rastreabilidade para os requisitos do cliente;

###### Consistência com os requisitos do cliente;

###### Testabilidade;

###### Viabilidade do design arquitetural do sistema;

###### Viabilidade de operação e manutenção.

### Processo de Implementação

#### Propósito

O propósito do Processo de Implementação é realizar um elemento de sistema especificado.

**NOTA 1 - Este padrão não contempla a realização de elementos de sistema, ou software, que não sejam ETL (software especializado). O Processo de Implementação de ETL (cláusula ) é uma especialização deste processo e contempla a realização de elementos de sistema que sejam software especializado para ETL.**

**NOTA 2 - O usuário deste padrão tem a liberdade para estender este Processo de Implementação, se necessário, de acordo com escopo, magnitude e complexidade do projeto.**

**NOTA 3 - Para estender o Processo de Implementação e aclamar conformidade com este padrão é necessário que o processo criado atenda os critérios estabelecidos no .**

### Processo de Implementação de ETL

#### Propósito

O propósito do Processo de Implementação de ETL é produzir um elemento de sistema especificado, implementado como um produto de software especializado para ETL.

Este processo transforma comportamentos, interfaces e restrições de implementação em ações que criam um elemento de sistema implementado como um produto de software especializado para ETL, ou simplesmente ETL. Este processo resulta em um ETL que satisfaz os requisitos de design arquitetural através de verificação e requisitos de cliente através de validação.

#### Resultados

1. Uma estratégia de implementação é definida;
2. Restrições tecnológicas são identificadas no design;
3. Um ETL é realizado.

O Processo de Implementação de ETL contém os seguintes sub-processos:

* Processo de Análise de Requisito de ETL
* Processo de Design de ETL
* Processo de Construção de ETL
* Processo de Integração de ETL

#### Atividades e tarefas

##### Estratégia de Implementação de ETL:

###### O desenvolvedor deve:

1. Colocar as saídas sob controle do e executar controle de mudança de acordo com esse.
2. Documentar e resolver problemas e não conformidades encontradas no produto de ETL e tarefas, de acordo com o .

###### Itens não entregáveis podem ser usados no desenvolvimento ou manutenção de produtos de ETL. Contudo, deve ser garantido que a operação e manutenção do produto de ETL, depois de sua entrega, é independente de tais itens; caso contrário, esses itens devem ser considerados como entregáveis.

### Processo de Análise de Requisito de ETL

**NOTA -** O Processo de Análise de Requisito de software, neste padrão, é um sub-processo do Processo de Implementação de ETL.

#### Propósito

O propósito do Processo de Análise de Requisitos de ETL é estabelecer os requisitos de ETL do sistema.

#### Resultados

1. Os requisitos alocados para o ETL são definidos;
2. Requisitos de ETL são analisados a fim de correção e testabilidade;
3. Priorização para implementação dos requisitos de ETL é definida;
4. Os requisitos de ETL são aprovados e atualizados conforme necessário;
5. Mudanças nos requisitos de ETL são avaliadas a fim de custo, cronograma e impacto técnico; e
6. Os requisitos de ETL são congelados e comunicados para todas as partes interessadas.

#### Atividades e tarefas

##### Análise de requisitos de ETL:

###### O desenvolvedor deve estabelecer e documentar os requisitos de ETL, descritos abaixo.

1. Interfaces externas ao ETL.
2. Requisitos de qualificação.
3. Definição de dados e requisitos de banco de dados.
4. Requisitos de documentação de usuário.

###### O desenvolvedor deve avaliar os requisitos de ETL considerando a lista de critérios abaixo. O resultado da avaliação deve ser documentado.

1. Consistência externa com os requisitos de sistema.
2. Consistência interna.
3. Testabilidade.
4. Viabilidade do design.
5. Viabilidade de operação e manutenção.

### Processo de Design Detalhado de ETL

**NOTA -** O Processo de Design Detalhado de ETL neste documento é um sub-processo do Processo de Implementação de ETL

#### Propósito

O propósito do Processo de Design Detalhado de ETL é providenciar um design para o ETL que está sendo implementado que possa ser verificado contra os requisitos e a arquitetura de sistema e é suficientemente detalhado para permitir codificação e teste.

#### Resultados

1. as unidades de ETL são identificadas e descritas; e
2. interfaces externas de cada unidade de ETL é definida;

#### Atividades e tarefas

##### Design detalhado de ETL:

###### O desenvolvedor desenvolver um design detalhado para o ETL. O ETL deve ser refinado em níveis granulares contendo unidades de ETL que podem ser codificadas e testadas. Deve-se garantir que os requisitos de ETL estão alocados nas unidades de ETL. O design detalhado deve ser documentado.

###### O desenvolvedor deve desenvolver e documentar um design detalhado para o banco de dados.

###### O desenvolvedor deve atualizar a documentação de usuário conforme necessário.

###### O desenvolvedor deve avaliar o design detalhado de ETL considerando os critérios abaixo. O resultado da avaliação deve ser documentado.

1. Rastreabilidade para os requisitos de ETL;
2. Consistência externa com o design arquitetural de sistema;
3. Consistência interna entre as unidades de ETL;
4. Viabilidade de teste;
5. Viabilidade de operação e manutenção.

### Processo de Construção de ETL

**NOTA -** O Processo de Construção de ETL neste documento é um sub-processo do Processo de Implementação de ETL.

#### Propósito

O propósito do Processo de Construção de ETL é produzir unidades de ETL que refletem apropriadamente o design de ETL.

#### Resultados

1. unidades de ETL definidas no design são produzidas;
2. consistência e rastreabilidade entre as unidades de ETL e requisitos e design; e
3. verificação das unidades de ETL contra os requisitos e design é realizada.

#### Atividades e tarefas

##### Construção de ETL:

###### O desenvolvedor deve desenvolver e documentar o seguinte:

1. Cada unidade de ETL e banco de dados.
2. Procedimentos de teste e dados para teste para cada unidade de ETL e banco de dados.

###### O desenvolvedor deve testar cada unidade de ETL e banco de dados garantindo que satisfazem os requisitos.

###### O desenvolvedor deve atualizar a documentação de usuário conforme necessário.

###### O desenvolvedor deve avaliar a unidade de ETL e resultados dos testes considerando os critérios listados abaixo.

1. Rastreabilidade para os requisitos e design do ETL.
2. Consistência externa com os requisitos e design do ETL.
3. Consistência interna entre os requisitos de unidade.
4. Conformidades com os métodos de codificação e padrões usados.
5. Viabilidade da integração e teste de ETL.
6. Viabilidade de operação e manutenção.

### Processo de Integração de ETL

**NOTA -** O Processo de Integração de ETL neste documento é um sub-processo do Processo de Implementação de ETL.

#### Propósito

O propósito do Processo de Integração de ETL é combinar as unidades de ETL, produzindo um item de ETL integrado, consistente com o design de ETL, que demonstra que os requisitos funcionais e não-funcionais são satisfeitos em um ambiente operacional equivalente.

#### Resultados

1. critérios de verificação para o ETL são desenvolvidos que garante conformidade com os requisitos de ETL alocados para o ETL;
2. ETL é produzido;

#### Atividades e tarefas

##### Integração de ETL:

###### O desenvolvedor deve integrar as unidades de ETL e testar conforme o agregado é desenvolvido.

###### O desenvolvedor deve atualizar a documentação de usuário conforme necessário.

###### O desenvolvedor deve desenvolver e documentar para cada requisito de qualificação do ETL um conjunto de testes, casos de teste (entradas, saídas, critério de teste), e procedimentos de teste para conduzir Teste de Qualificação de ETL. O desenvolvedor deve garantir que o ETL integrado esta pronto para Teste de Qualificação de ETL.

###### O desenvolvedor deve avaliar a integração, design, código, testes, resultados de testes, e documentação de usuário considerando os critérios abaixo.

1. Rastreabilidade para os requisitos de sistema.
2. Consistencia externa com os requisitos de sistema.
3. Consistência interna.
4. Cobertura dos testes sobre os requisitos de ETL.
5. Conformidade dos padrões de teste e métodos usados.
6. Conformidade com os resultados esperados.
7. Viabilidade de teste de qualificação de ETL
8. Viabilidade de operação e manutenção

### Processo de Teste de Qualificação de ETL

**NOTA -** O Processo de Teste de Qualificação de ETL neste documento é um sub-processo do Processo de Implementação de ETL.

#### Propósito

O propósito do Processo de Teste de Qualificação de ETL é confirmar que o ETL integrado atendo seus requisitos definidos.

#### Resultados

1. critérios para o ETL integrado são desenvolvidos que demonstram conformidade com os requisitos de ETL;
2. o ETL integrado é verificado usando os critérios definidos;
3. uma estratégia de regressão é desenvolvida e aplicada para re-testar o ETL integrado quando uma mudança ocorrer em alguma unidade de ETL.

#### Atividades e tarefas

##### Teste de qualificação de ETL:

###### O desenvolvedor deve conduzir teste de qualificação de acordo com os requisitos de qualificação para o ETL. Deve-se garantir que cara requisito de ETL é testado para conformidade.

###### O desenvolvedor deve atualizar a documentação de usuário conforme necessário.

###### O desenvolvedor deve avaliar o design, código, testes, resultado de destes e documentação de usuário considerando os critérios abaixo.

1. Cobertura dos teste sobre os requisitos de ETL.
2. Conformidade com o resultado esperado.

## Ciclo de vida

## Mecânica do ciclo de vida

### Dos processos

1. **Processos** são compostos por **atividades** ou outros processos.
2. Processos compostos por outros processos são definidos como **processo composto**.
3. Processos que compões outros processos são definidos como **sub-processo** do **processo compositor**.
4. Um processo composto pode usar seus sub-processos para atingir seu propósito.
5. Um processo não pode usar atividades ou sub-processos de outros processos para atingir seu propósito.
6. O usuário deste modelo deve instanciar processos para poder usar suas atividades. A entidade resultante da ação de instanciar um processo é definido como **instância de processo**.
7. A **ordem** e **cronograma** criados pelo o usuário que usam as atividades dos processos instanciados é definido como **plano de atividades**.
8. A instância de um processo apresenta comportamentos diferentes de acordo com as **características de processo** definida aqui:
   1. ***Multiplicidade****: a multiplicidade de um processo diz respeito à quantas instâncias concorrentes de um processo um projeto pode conter. A multiplicidade assume um de dois valores possíveis:*
      1. **Múltiplo**: o projeto pode usar múltiplas instâncias paralelas do mesmo processo. Cada instância deve ter um identificador único. A instância que possui um identificador vazio é definida como instância principal.
      2. **Único**: o projeto só pode ter uma instância do processo.
   2. **Recursividade**: a recursividade de um processo diz se um processo pode usar a si mesmo como sub-processo para o atendimento de seu propósito. Um processo poder **recursivo** ou não.

### Das atividades

1. Uma atividade é composta por **tarefas**.
2. Uma atividade defini uma **ordem e seqüenciamento condicional** de suas tarefas através de **diagramação BPMN 2.0**
3. O usuário deste modelo deve instanciar atividades de acordo com as necessidades de projeto ou com o plano de atividades. A entidade resultante da ação de instanciar uma atividade é definida como **instância de atividade.**
4. A instância de uma atividade apresenta comportamentos diferentes de acordo com as características da atividades, definidas aqui:
   1. **Multiplicidade:**  a multiplicidade de uma atividade diz respeito à quantos instâncias de uma atividade uma instância de processo pode conter. A multiplicidade assum um de dois valores possíveis
      1. **Múltipla**: o processo pode usar múltiplias instâncias da mesma atividade. Cada instância deve ter um identificador único. A instância que possui um identificador vazio é definida como instância principal da atividade.
      2. **Única**: o processo só pode conter uma intância da atividade.
   2. **Formalidade:** a formalidade de uma atividade diz respeito à objetividade técnica da implementação de uma atividade. A formalidade de uma atividade assume um de dois valores, que, quando não dito o contrário assume-se o valor de **objetiva**. Esses valores são aqui definidos:
      1. Objetiva: a implementação da atividade é diagramada de acordo com o padrão BPMN 2.0 e seus artefatos associados estão definidos neste modelo.
      2. **Ad hoc**: a implementação da atividade pode não estar diagramada de acordo com o padrão BPMN 2.0 ou seus artefatos associados podem não estar definidos neste modelo. Fica a critério do usuário o método que vai ser usado para alcançar o objetivo da atividade.
   3. **Iteratividade:** a iteratividade de uma atividade diz respeito à possibilidade de evoluir os artefatos associados à uma instância de uma atividade após a conclusão da atividade. Um atividade pode ser **iterativa** ou não. Uma atividade iterativa pode ser re-instanciada após sua conclusão, onde a nova instância irá refinar o atendimento aos objetivos da atividade à partir de um contexto mais aprimorado.
   4. **Modo de instanciamento:** o modo de instanciamento diz respeito à como uma atividade pode ser instanciada. Quando não dito o contrário, assume-se que o modo de instanciamento é **manual**. Essa característica pode assumir um de dois valores, aqui definidos:
      1. Manual**:** uma instância de uma atividade só é criada à partir da intervenção de um usuário envolvido no projeto.
      2. **Automática:** uma instância de uma atividade é criada automaticamente à partir de **eventos** que ocorram durante um projeto.

### Das tarefas

### Dos estágios

1. Atividades de um processos podem instanciar atividades do mesmo processos ou de sub-processos.
2. Uma atividade tem somente um ponto de entrada.
3. Uma atividade se encontra em um dos cinco estados:
   1. Planejada: a atividade foi criada a priori de sua instanciação devido a decisões de planejamento.
   2. Em andamento: a atividade está sendo executada.
   3. Concluída: a atividade está concluída.
   4. Suspensa: a atividade foi suspensa e por decisão gerencial, impasse técnico, ou impasse organizacional.
   5. Cancelada: Atividade foi cancelada por decisão gerencial.
4. Um processo se encontra em um de dois estados:
   1. Aberto: o processo está disponível para prestar serviços.
   2. Fechado: o processo não está disponível para prestar serviços
5. Uma atividade pode lançar eventos.
6. Uma atividade pode capturar eventos lançados pelo processo invocador.
7. Um processo pode capturar eventos de suas atividades ou de seus sub-processos.

[N]ot everything imaginable is feasible, not even in software, evanescent as it may appear

to outsiders. On the contrary, software feasibility has four solid dimensions: Technology—

Is a project technically feasible? Is it within the state of the art? Can defects be reduced to

a level matching the application’s needs? Finance—Is it financially feasible? Can development be completed at a cost the software organization, its client, or the market can

afford? Time—Will the project’s time-to-market beat the competition? Resources—Does

the organization have the resources needed to succeed?