

# **MODELAGEM DE SISTEMAS**

## **FUNDAMENTOS DA ENGENHARIA DE SOFTWARE - UNIDADE 01**

Na sociedade atual o uso de qualquer tipo de sistema tomou grandes proporções. Atualmente é comum o mercado de trabalho exigir que os profissionais tenham conhecimento em um determinado sistema ou então que as empresas treinem seus funcionários para usar.

### **PROCESSO EVOLUTIVO DOS SOFTWARES**

Conforme novas formas de uso para hardware e software se tornam acessíveis fica evidente a maior necessidade de melhorar os sistemas. As necessidades dos clientes passam a mudar e vão se alterando conforme o tempo. As mudanças sempre ocorrerão ao longo do tempo de criação e uso de um software. Sempre existe a necessidade de ajustes e correções ou ainda pode ocorrer à necessidade de adicionar novas funcionalidades. Um software passa por uma série de manutenções e após realizar diversos ajustes os mesmos podem levar a novos problemas. Algumas vezes se faz necessário até mesmo criar um novo sistema.

### **O PAPEL DO ANALISTA DE SISTEMAS**

O analista de sistemas possui um papel fundamental nos processos da engenharia de software pois é o responsável por realizar pesquisas e planejamentos além coordenar as equipes de desenvolvimento e recomendar de alternativas de software de acordo com as necessidades de desenvolvimento ou solução para problemas de negócios. Um analista de sistemas deve primeiro descobrir o que um sistema deverá fazer e depois entender e avaliar as necessidades e expectativas de cada usuário em relação a ele. Neste momento o profissional desse ramo precisa conhecer um pouco de cada área de negócio e caso não tenha o domínio necessário sobre algum tema deve ter a proatividade de procurar o máximo de conhecimento sobre a área que o software abrangerá. Uma característica importante do analista de sistemas é ter uma boa visão empresarial para ajudar nos processos gerenciais da produção do software e um conhecimento tecnológico atualizado além de um ótimo relacionamento interpessoal.

### **PRÁTICAS DA ENGENHARIA DE SOFTWARE**

- **QUALIDADE**
- **PROCESSO**
- **MÉTODOS**
- **FERRAMENTAS**

### **PRINCÍPIOS DA ANÁLISE DE SISTEMAS**

Os princípios da análise de sistemas se fundamentam na necessidade de realizar estudos de processos para encontrar a melhor solução para a criação de um sistema. A análise de sistemas se baseia em métodos de investigação para encontrar a melhor solução para algum problema a partir das funcionalidades levantadas pelo analista de sistemas.

- **ANALISAR**
- **PROJETAR**
- **IMPLEMENTAR**
- **TESTAR**
- **DOCUMENTAR**
- **ADAPTAR**

## **CATEGORIAS DE SOFTWARES**

**SISTEMA (SUPOORTAM OUTROS PROGRAMAS)**

**APLICATIVO (RESOLVEM DETERMINADO PROBLEMA)**

**ENGENHARIA (ENTENDIMENTO DE PROBLEMAS)**

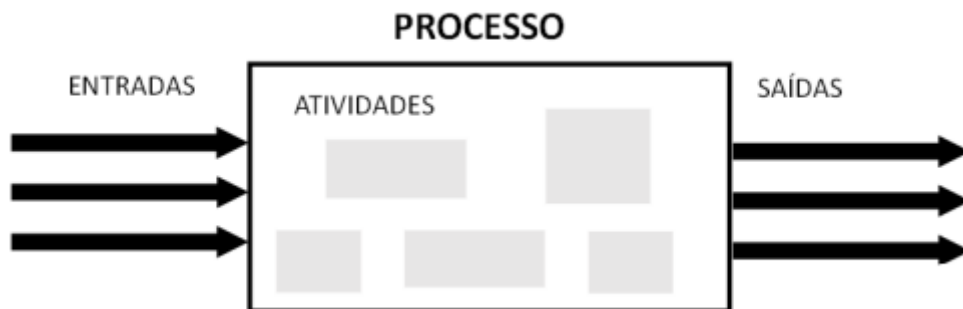
**LINHA (USO POR DIVERSOS CLIENTES)**

**EMBARCADOS (APENAS PARA DETERMINADO PRODUTO)**

**MÓVEIS (DISPOSITIVOS COM ACESSO A INTERNET)**

## **PROCESSO DE SOFTWARE**

Um Processo de Software é um conjunto de atividades e resultados que levam à produção de um determinado sistema. Na Engenharia de Software um processo não é uma determinação rigorosa sobre como ele deve ser desenvolvido mas sim uma abordagem adaptável que torna possível à equipe de desenvolvimento escolher os processos que melhor se enquadram na forma de trabalho da empresa. Um Processo de Software possui diversas entradas e saídas mas também é composto por processos que consistem em uma série de atividades que serão executadas de forma padronizada e agrupadas em fases que mudam conforme prosseguem.



## **PROCESSO GENÉRICO DE SOFTWARE**

### **ESPECIFICAR O SOFTWARE**

- **DEFINIR O QUE SERÁ DESENVOLVIDO (FUNCIONALIDADES)**

### **PROJETAR E IMPLEMENTAR O SOFTWARE**

- **PROJETO E DESENVOLVIMENTO (PROGRAMAR)**

### **VALIDAR O SOFTWARE**

- **OBSERVAR SE O QUE FOI CONSTRUÍDO ATENDE AOS REQUISITOS**

### **ADAPTAR O SOFTWARE**

- **ATUALIZAR PARA ACOMPANHAR AS NECESSIDADES DO CLIENTE**

## **PRINCIPAIS ATIVIDADES METODOLÓGICAS**

- **COMUNICAR**
- **PLANEJAR**
- **MODELAR**
- **CONSTRUIR**
- **ENTREGAR**

## **FLUXOS DE PROCESSO**

As atividades metodológicas citadas anteriormente devem ter uma série de tarefas que darão suporte no acompanhamento e controle do projeto onde a forma com que a sequência de cada atividade genérica é realizada pode ter formas distintas dependendo do projeto.

### **FLUXO DE PROCESSO LINEAR**

- **REALIZA AS CINCO ATIVIDADES EM FORMA SEQUENCIAL**

### **FLUXO DE PROCESSO INTERATIVO**

- **REPETE UMA OU MAIS ATIVIDADES ANTES DE PROSSEGUIR**

### **FLUXO DE PROCESSO EVOLUCIONÁRIO**

- **ATIVIDADES EXECUTADAS DE MODO CIRCULAR**
- **CICLOS QUE ENVOLVEM AS CINCO ATIVIDADES**

### **FLUXO DE PROCESSO PARALELO**

- **ATIVIDADES REALIZADAS DE FORMA PARALELA**
- **ATIVIDADES PODEM SER EXECUTADAS SIMULTANEAMENTE**

## **CONJUNTO DE TAREFAS**

Desenvolver um software requer muito planejamento e se deve ter mente que um software nunca é igual ao outro. Os Analistas de Sistemas determinam o conjunto de tarefas baseados nos problemas e nas características do projeto que será executado. Um conjunto de tarefas determina o que deverá ser feito para alcançar os objetivos de uma determinada ação dentro do processo. Um exemplo é a programação das telas de um sistema onde o programador recebe um conjunto de requisitos que as telas deverão possuir e após a conclusão da programação seu trabalho será avaliado conforme as especificações que ele recebeu.

## **PADRÕES DE QUALIDADE**

A existência de um Processo de Software não garante a qualidade do mesmo e muito menos que ele será entregue no prazo combinado. Também não é certo que suas funcionalidades estarão de acordo com o que o cliente demandou. A qualidade é diretamente influenciada pelos padrões impostos durante a produção sendo necessário estabelecer procedimentos para garantir a qualidade dos processos. Os erros que ocorrem durante o processo podem ser controlados usando uma abordagem metodológica. O analista de sistemas deve estar atento ao surgimento de novas metodologias e se forem apropriadas deve fazer seu uso durante o desenvolvimento sempre visando à qualidade e a aprovação do cliente.

## **MODELOS DE PROCESSOS DE SOFTWARE**

Para o gerenciamento das atividades são usados os Modelos de Processos de Software que possuem como objetivo controlar e organizar as atividades do processo. Consiste em uma representação dos objetos e atividades envolvidas no Processo de Software. Um Modelo de Processo de Software é um guia exclusivo para as atividades da Engenharia de Software e existem diversos Modelos de Processos de Softwares que possuem características diferentes.

### **MODELO DE PROCESSO PRESCRITIVO (TRADICIONAIS)**

Consiste em um conjunto de elementos do processo que garantem a qualidade e o controle de mudanças nos projetos de desenvolvimento. As tarefas ocorrem de forma sequencial e com regras bem definidas. Esse tipo de modelo prescreve os relacionamentos e tem como objetivo estruturar e ordenar o desenvolvimento de um software.

## **MODELO PRESCRITIVO INCRMENTAL**

O Modelo Incremental é um modelo que visa criar e entregar pequenas versões do software e posteriormente desenvolver novas versões até que o sistema esteja completo. Nesse modelo uma nova versão é chamada de incremento e cada incremento incorpora parte de uma funcionalidade desejada pelo cliente. Essas pequenas versões são chamadas de módulos que acrescentam ou melhoram as funcionalidades do sistema.

## **MODELO PRESCRITIVO EVOLUCIONÁRIO**

### **PROTOTIPAGEM**

Esse modelo começa a partir da comunicação onde se observa quais são os objetivos e as funcionalidades do software. No planejamento rápido são determinados os requisitos que serão modelados e que já podem ser construídos. Após a entrega o cliente fornece sua opinião e a equipe de desenvolvimento passa a aprimorar os requisitos.

### **ESPIRAL**

Esse modelo usa aspectos sistemáticos e controlados do Modelo Cascata e seu objetivo consiste em fornecer um rápido desenvolvimento de versão.

## **MODELO PRESCRITIVO CONCORRENTE**

O Modelo Concorrente permite que a equipe de desenvolvimento possa representar elementos concorrentes de qualquer um dos modelos. Modelos Concorrentes são usados em projetos que envolvem diferentes equipes de desenvolvimento onde os planos de projeto devem ser considerados documentos vivos. O Modelo Concorrente pode ser usado em diversos tipos de desenvolvimento porque não segue uma sequência de atividades e estabelece uma rede de atividades que se movimentam de uma atividade para outra.

## **MODELO DE PROCESSO ESPECIALIZADO**

- **MODELO BASEADO EM COMPONENTES**
- **MODELO DE MÉTODOS FORMAIS**
- **DESENVOLVIMENTO ORIENTADO A ASPECTO**
- **MODELO DE PROCESSO UNIFICADO**
- **MODELOS DE PROCESSOS PESSOAL**

## **MODELO DE DESENVOLVIMENTO ÁGIL**

As etapas de desenvolvimento geralmente são muito demoradas e esse modelo consiste em uma resposta ao rápido desenvolvimento dos softwares e aplicativos. Torna o desenvolvimento menos ferrenho e atende as necessidades do cliente com mais velocidade. Seu princípio é focado nas entregas e prioriza também a comunicação entre os envolvidos de forma ativa e contínua para realizar entregas. Reduz drasticamente a documentação e a burocracia presente em outros Modelos de Processos.

## **FUNDAMENTOS DE PROCESSOS DE NEGÓCIO - UNIDADE 02**

### **ÁREAS DO NEGÓCIO**

As áreas do negócio têm como objetivo dar prosseguimento à missão organizacional por meio da produção de bens que atenderão às necessidades externas.

Quando falamos em áreas de negócios não podemos nos esquecer do processo de negócio que consiste numa estruturada sequência de tarefas.

## **TIPOS DE PROCESSOS**

### **MACROPROCESSO (PROCESSOS ABRANGENTES)**

#### **PROCESSO (SIMPLES) (COMPLEXOS)**

#### **SUBPROCESSO (ETAPA DO PROCESSO)**

## **PROCESSO DE NEGÓCIO**

O processo de negócio representa a consolidação de atividades que visam atingir um resultado que demonstre valor agregado ao cliente realizando o sequenciamento de atividades com o objetivo de desenhar como o trabalho deverá ser executado.

## **PROCESSOS PRIMÁRIOS**

- **ATRELADOS AS ATIVIDADES ESSENCIAIS DA EMPRESA**
- **AGREGAM VALOR AO CLIENTE**
- **IMPACTO DIRETO AO CLIENTE (MAIOR IMPACTO)**

## **PROCESSOS DE SUPORTE**

- **AGREGAM VALOR AO PROCESSO**
- **SERVEM COMO SUPORTE PARA OS DEMAIS PROCESSOS**
- **IMPACTO INDERETO AO CLIENTE (MENOR IMPACTO)**

## **PROCESSO DE GERENCIAMENTO**

- **MONITORAMENTO E CONTROLE DAS ATIVIDADES ORGANIZACIONAIS**
- **ACOMPANHAMENTO DOS RESULTADOS**

## **MODELAGEM DE PROCESSOS DE NEGÓCIO**

A modelagem de processos envolve algumas técnicas que ajudam a entender os processos de negócios das empresas. A modelagem aplicada ao campo da informática trata da criação de modelos que consistem em representações em escala reduzida de algo real. O processo de negócio é traduzido como uma sequência de atividades executadas sequencialmente visando um determinado resultado que agregue valor ao cliente.

## **ABORDAGENS PARA MODELAGEM**

### **BOTTOM UP (MICRO PARA MACRO)**

- **PARTE DO DETALHAMENTO DE TAREFAS E ATIVIDADE**
- **DEPOIS SE ESTABELECE UMA VISÃO MACRO DA EMPRESA**

### **BOTTOM DOWN (MACRO PARA MICRO)**

- **PARTE DA VISÃO MACRO DA EMPRESA**
- **POSTERIORMENTE SE ATINGE A VISÃO DO PROCESSO**

## **BPMN**

Diagrama que permite desenhar os mais diversos tipos de modelagem de processo. Se mostrou como uma técnica de simples compreensão.

## **ELEMENTOS (BPMN)**

**ATIVIDADE (TRABALHO A SER REALIZADO)**

**EVENTO (INFLUENCIAM OUTROS ELEMENTOS)**

**GATEWAYS (CONTROLAM O FLUXO)**

**CONECTORES (DEMONSTAM CAMINHOS)**

**DADOS (AGREGAM DADOS AO PROCESSO)**



## **DIAGRAMA (REDUZIDO)**

- APRESENTA APENAS OS ELEMENTOS PRINCIPAIS
- OCULTA OS DETALHES RELATIVOS AO FLUXO DE TRABALHO

## **MAPA**

- AGREGA MAIS DETALHES ACERCA DO PROCESSO
- MOSTRA OS PRINCIPAIS COMPONENTES DO PROCESSO

## **MODELO (COMPLETO)**

- APRESENTA TODOS OS ENVOLVIDOS
- MOSTRA OS FATORES QUE AFETAM O COMPORTAMENTO

## **GERENCIAMENTO DE PROCESSOS DE NEGÓCIO (BPM)**

O gerenciamento é algo importante para todas as organizações e em todas as suas áreas pois permite descobrir lacunas que prejudicam os resultados e desenvolver planos de melhoria para acabar com as mesmas. Com o gerenciamento de processos de negócio os processos organizacionais são vistos de uma forma ampliada onde existe uma percepção horizontal de toda a cadeia envolvida para entregar um produto ou serviço. Se faz possível entender o gerenciamento de processos de negócio como a construção de processos ponta a ponta que integra as estratégias com os objetivos da empresa. Todas as ações são direcionadas pelas mesmas regras e com o intuito de se atingir os objetivos organizacionais. Em uma empresa a governança deve incorporar a ideia de controle e prestação de contas.

## **SISTEMA DE GERENCIAMENTO DE PROCESSOS DE NEGÓCIO (BPMS)**

O BPMS é uma ferramenta que permite executar e monitorar os processos funcionais visando automatizar as ações e o fluxo de informações existentes nos processos. Traz uma visão muito mais ampla e permite integrar sistemas antigos que permanecem em operação.

## **BPMS MAIS USADOS**

- CASEWISE
- ARIS PLATFORM
- AQUALOGIC STUDIO
- VISIO
- BONITA OPEN SOLUTION
- COMINDWARE

**SMART (DESENVOLVIMENTO DE METAS)**

**SPECIFIC (OBJETIVOS CLAROS)**

**MEASURABLE (OBJETIVOS COMENSURÁVEIS)**

**ATTAINABLE (OBJETIVOS REALISTAS)**

**RELEVANT (OBJETIVOS IMPORTANTES)**

**TIMELY (APURAR RESULTADOS DENTRO DE CERTO TEMPO)**

## **MODELO INTEGRADO DE CAPACIDADE DE MATURIDADE (CMMI)**

O planejamento do gerenciamento de processos de negócios está vinculado à compreensão da escala de maturidade em processos. A capacidade da empresa de entender e gerenciar seus processos determinará a forma como o gerenciamento acontecerá. O CMMI é extremamente relevante para o gerenciamento de processos pois colabora para o melhor gerenciamento de atividades que acarreta em produto padronizado e com menor risco de erros.

## **NIVEIS DE MATURIDADE (CMMI)**

- **INICIAL**
- **GERENCIADO**
- **DEFINIDO**
- **GERENCIADO QUANTITATIVAMENTE**
- **OTIMIZADO**

## **CULTURA ORGANIZACIONAL**

Atualmente um dos pontos de maior debate das organizações é a gestão de mudança um fator apontado por diversas pessoas como um dos grandes desafios contemporâneos. Mudar gera desconforto e traz impactos negativos aos processos de gestão organizacionais onde a aptidão para a mudança é extremamente necessária. Assim existem duas formas para essa mudança cultural acontecer. A primeira forma é fazer com que as pessoas comprem a ideia e passem a ter o mesmo pensamento a respeito do tema e a segunda forma envolve o recrutamento de novas pessoas para o compor a equipe.

## **O PROCESSO DE ENGENHARIA DE REQUISITOS - UNIDADE 03**

Um dos maiores problemas no desenvolvimento de um software é estabelecer em quanto tempo será concluído. Outro desafio é garantir que as necessidades sejam correspondidas pelo software entregue. Essas questões podem ser superadas com a engenharia de requisitos que consiste em um conjunto de atividades que colaboram para a produção e a manutenção de um documento conhecido como documento de requisitos. Esse processo descreve todas as funções que um sistema deve possuir bem como as restrições do mesmo proporcionando a todos os envolvidos no desenvolvimento do sistema uma mesma compreensão do problema.

## **ENUNCIADOS**

Criar os enunciados extensos ou com pouco designo para os requisitos não consiste em algo recomendado pois os mesmos devem ser objetivos e consistentes para que a compreensão do que será realizado seja possível por todas as partes envolvidas.

## **PRIORIDADES**

Destacar um fator no levantamento de requisitos determina sua precedência.

- REQUISITO ESSENCIAL
- REQUISITO IMPORTANTE
- REQUISITO DESEJÁVEL

## **ROTULAGEM DOS REQUISITOS**

### **REQUISITOS FUNCIONAIS**

- FUNCIONALIDADES QUE O SISTEMA DEVE ATENDER

### **REQUISITOS NÃO FUNCIONAIS**

- QUALIDADES QUE SISTEMA DEVE APRESENTAR

### **REQUISITOS DE DOMÍNIO**

- CONTROLAM OS REQUISITOS FUNCIONAIS

## **ATIVIDADES DO PROCESSO DA ENGENHARIA DE REQUISITOS**

- ENTENDER PROBLEMA
- COLETAR E CLASSIFICAR REQUISITOS
- NEGOCIAR REQUISITOS
- VALIDAR REQUISITOS
- DOCUMENTAR REQUISITOS

## **ELICITAR REQUISITOS**

Esse processo consiste em descobrir o máximo de informações para estabelecer os requisitos de determinado sistema. Envolve diversas pessoas conhecidas como stakeholders e a por meio dos problemas e das necessidades apontados por eles visa realizar as algumas tarefas.

### **ELENCAR (COLETAR REQUISITOS DO SISTEMA)**

### **CLASSIFICAR (DESCOBIR SE OS MESMOS SE REPETEM)**

### **PRIORIZAR (ESTABELECE A PRIORIDADE DE CADA REQUISITO)**

### **ESPECIFICAR (DOCUMENTAR OS REQUISITOS)**

## **TECNICAS PARA ELICITAR REQUISITOS**

- PESQUISA
- ENTREVISTA
- REUNIÕES
- DOCUMENTOS
- ETNOGRAFIA

## **ESPECIFICAR REQUISITOS**

Apontamento sobre o que determinado software deverá realizar por meio de descrições claras e objetivas. Procura caracterizar o problema a ser solucionado gerando um documento. Esse processo consiste no meio de comunicação entre o analista de sistemas e os programadores e assim é preciso que não haja margem para duplas interpretações visto que o programador programará exatamente o que consta no documento. Podemos usar diagramas apontando os caso de uso ou ainda fazer uso da prototipagem.



## **PROTOTIPAGEM**

Consiste na criação de uma versão menor do sistema a ser desenvolvido.

### **PROTOTIPAGEM DESCARTÁVEL**

- **CRIADO APENAS PARA ENTENDER O PROBLEMA APONTADO**
- **DESCARTADO POSTERIORMENTE**

### **PROTOTIPAGEM EVOLUTIVA**

- **CRIADO QUANDO CLIENTES TEM DÚVIDAS SOBRE O SOFTWARE**

### **PROTOTIPAGEM RÁPIDA**

- **PEQUENAS PARTES DO SOFTWARE**
- **OBSERVAM VIABILIDADE DOS REQUISITOS E DO PROJETO**

## **NEGOCIAR REQUISITOS**

Esse processo tem como objetivo desenvolver um plano de projeto que atenda às demandas dos envolvidos respeitando o orçamento e o tempo impostos à equipe de desenvolvimento.

## **MONITORAR REQUISITOS**

Esse processo garante o monitoramento de cada alteração de requisitos durante o processo de desenvolvimento do software fazendo o uso de alguma ferramenta de controle.

## **VALIDAR REQUISITOS**

Processo que assegura que os requisitos propostos atenderão às necessidades impostas.

## **MODELAGEM DE REQUISITOS**

Esse processo foca no que será desenvolvido e não em como será realizado.

## **REMO (MODELAGEM DE REQUISITOS)**

Geralmente usada na fase de extração essa técnica conecta a modelagem de processos de negócios ao desenvolvimento do sistema. Consente que a extração seja realizada por meio dos diagramas de processos de negócios e seu método é composto por duas fases. Na primeira o foco é no contexto para que se possa conhecer o problema e elaborar um documento sobre o mesmo. Na segunda fase o foco é levantar e descrever os requisitos do sistema.

## **DOCUMENTOS**

Os documentos na fase de extração registram os principais tópicos que dizem respeito ao que o sistema realizará e determinam em quais estados as soluções serão executadas. Os requisitos são descritos de forma clara e o detalhamento dos mesmos pode ser necessário caso deem margem para dúvidas. Importante ressaltar que processo de documentar começa nos registros das reuniões com todo o pessoal envolvido.

## **DIAGRAMAS DE CASOS DE USO**

Apontado como a parte mais importante para construção de um software. Esses diagramas detalham o que precisa ser programado e servem como meio de comunicação entre os envolvidos no projeto. Cada caso de uso representa uma determinada função ou tarefa do sistema que envolve agentes externos. Descrever um caso de uso tem como objetivo informar quais os atores interagem com as funções que estão sendo modeladas.

# **PARADIGMA ORIENTADO A OBJETOS - UNIDADE 04**

## **FUNDAMENTOS**

Consideramos um paradigma como um modelo que foi testado e segue alguns princípios para a resolução de um problema. Um modelo consiste em uma representação do mundo real e fazer o uso de um torna o desenvolvimento menos complexo. O paradigma orientado a objetos surgiu em 1966 onde se demonstrou o conceito de objeto e posteriormente passou a ser amplamente usado para modelagem. Com esse paradigma surgiu não só um novo padrão para o desenvolvimento mas também uma nova forma de pensar sobre os problemas do mundo.

## **CONCEITOS BÁSICOS (POO)**

### **INTERFACE**

Nesse processo nos referimos a um objeto como algo do mundo real e sem nos preocuparmos com os detalhes que compõe o mesmo.

### **CLASSES**

Momento em que se determinam as características que o objeto deverá ter e quais ações que ele poderá fazer. As características são chamadas de atributos e os comportamentos de métodos.

#### **CADEIRA**

- **COR (A)**
- **RODAS (A)**
- **PERNAS (A)**
- **ENCOSTO (A)**
- **ASSENTO (A)**
- **ANDAR (M)**
- **MOVER ENCOSTO (M)**

### **OBJETOS**

Um objeto consiste em uma instância da classe. São únicos e se distinguem pelos valores dos seus atributos comportamentos apesar de pertencerem a uma mesma classe.

#### **CADEIRA (PRETA)**

- **COR (PRETO) (A)**
- **PERNAS (A)**
- **ENCOSTO (A)**
- **ASSENTO (A)**
- **MOVER ENCOSTO (M)**
- **ELEVAR ASSENTO (M)**

### **LEGADO**

Podemos fazer classes abrangentes e durante o processo de modelagem elas serão melhoradas dando origem às chamadas subclasses. As mesmas poderão herdar os comportamentos e as características da classe abrangente que nesse caso são chamadas de superclasses.

## **CADEIRA (SUPERCLASSE)**

- **COR (A)**
- **PERNAS (A)**
- **ENCOSTO (A)**
- **ASSENTO (A)**
- **MOVER ENCOSTO (M)**
- **ELEVAR ASSENTO (M)**



## **CADEIRA GAMER (SUBCLASSE)**

- **RODAS (A)**
- **ANDAR (M)**
- **GIRAR (M)**

## **ENCAPSULAMENTO**

Pode esconder do usuário a forma que uma classe faz uma determinada ação ou como os dados são realmente representados. Processo de dissimulação de todos os detalhes de um objeto que não contribuem para suas características essenciais.

## **CADEIRA**

- **COR (A) (OCULTO) (-)**
- **PERNAS (A) (OCULTO) (-)**
- **ENCOSTO (A) (OCULTO) (-)**
- **ASSENTO (A) (OCULTO) (-)**
- **MOVER ENCOSTO (M) (+)**
- **ELEVAR ASSENTO (M) (+)**

## **POLIFORMISMO**

Um mesmo método pode atuar de modos diversos em classes diferentes.

## **ANÁLISE E PROJETO ORIENTADO A OBJETO**

Durante esse processo os conceitos apontados anteriormente são usados para análise e elaboração do projeto que são fases que antecedem a programação. Um projeto que venha a fazer o uso desse paradigma deve usar esta estratégia em todo processo de desenvolvimento.

## **MODELO DO PROCESSO UNIFICADO (PU)**

Um processo consiste na maneira como produzimos um software. Engloba metodologias e técnicas ao longo do processo. Existem alguns modelos podem ser usados para organizar o processo de desenvolvimento. Dentro desse universo temos o PU que consiste em um modelo adaptativo baseado no modelo incremental e que visa à construção evolutiva de um software.

## **PROCESSO UNIFICADO RACIONAL (RUP)**

Esse modelo consiste em um melhoramento do modelo anterior. Cobre conceitos de suporte voltados para a maturidade do produto e nos dias atuais tem a IBM como detentora.

# **ASPECTOS CHAVES DO PROCESSO UNIFICADO**

## **DIRIGIDO POR CASO DE USO**

Nesta fase devemos conhecer o quê os futuros usuários necessitam e desejam. Devemos nos atentar para o que cada usuário precisa e não apenas as funções do sistema.

## **CENTRADO NA ARQUITETURA**

Essa fase visa dar ao engenheiro de software uma visão abrangente sobre o sistema.

## **EVOLUTIVO E INCREMENTAL**

O processo de evolução consiste em segmentar o projeto em subprojetos menores gerando um acréscimo como resultado desse processo.

## **ELEMENTOS BÁSICOS DE UM PROCESSO (PU)**

### **PAPEL**

- **APONTAR AS RESPONSABILIDADES DE CADA MEMBRO**
- **UM TRABALHADOR PODE DESENVOLVER DIVERSAS ATIVIDADES**

### **ARTEFATO**

- **QUALQUER PRODUTO QUE TRABALHADOR GERA**

### **ATIVIDADE**

- **TAREFA EXECUTADA PELO TRABALHADOR**

### **DISCIPLINA (FLUXO DE TRABALHO)**

- **FLUXO DE ATIVIDADES QUE GERAM UM RESULTADO**

## **PERSPECTIVAS DO FLUXO DE TRABALHO**

- **MODELAGEM DE NEGÓCIO**
- **REQUISITOS**
- **ANÁLISE E PROJETO**
- **IMPLEMENTO**
- **TESTES**
- **ENTREGA**

## **MÉTODOS ORIENTADOS A OBJETOS**

Com o paradigma orientado a objeto foi observado à necessidade de métodos voltados para análise de projetos e assim surgiram vários que atendiam a esse fim. Mas o mercado ansiava por uma norma única que tornasse o processo de modelagem e manutenção menos complexo e em com esse contexto se desenvolveu uma linguagem para modelagem de sistemas que nomeada de UML que passou a ser usada na maioria das empresas. Surgia então um padrão para modelagem orientada a objetos.

## **UML**

Fazendo uso de diagramas dos mais diversos tipos nos permite enxergar de forma menos complexa os objetos e seus relacionamentos.

## **DIAGRAMAS (UML)**

## **DIAGRAMA DE CASO DE USO**

- FORNECE PONTO DE VISTA GERAL DOS OBJETIVOS DOS ATORES

## **DIAGRAMA DE CLASSE**

- DECRETA ESTRUTURA DAS CLASSES USADAS PELO SISTEMA

## **DIAGRAMA DE SEQUENCIA**

- MOSTRA O RELACIONAMENTO ENTRE OS PARTICIPANTES

## **DIAGRAMA DE ATIVIDADE**

- DESCREVE PASSOS A SEREM SEGUIDOS PARA EXECUTAR UMA TAREFA

## **DIAGRAMA DE ESTADO**

- MOSTRA PASSAGEM DE ESTADOS PELOS OBJETOS DO SISTEMA
- ENTRY (ASSUME UM ESTADO)
- DO (ENQUANTO EM UM ESTADO)
- EXIT (SAI DE UM ESTADO)

## **ESTEREÓTIPOS (ELEMENTOS QUE TEM ALGO COMUM)**

**ENTITY (ARMAZENAM DADOS RECEBIDOS PELO SISTEMA)**

**BOUNDARY (COMUNICAM ATORES EXTERNOS AO SISTEMA)**

**CONTROL (INDICAM REGRAS DO NEGÓCIO)**

## **TIPOS DE RELACIONAMENTOS (UML)**

- DEPENDENCIA
- ASSOCIAÇÃO
- AGREGAÇÃO
- COMPOSIÇÃO
- GENERALIZAÇÃO
- MULTIPLICIDADE