

# Processo unificado

## Apresentação

Processo unificado, também conhecido como PU, é um modelo de processo de *software* que visa à construção de *software* orientado a objetos e que pode ser personalizado de acordo com as necessidades e recursos disponíveis para cada projeto. É um processo interativo e adaptativo que permite conduzir de forma organizada e consistente o projeto de *software* por meio de um conjunto de atividades que especificam o processo, as quais vão transformar as necessidades de um cliente em um *software*.

Nesta Unidade de Aprendizagem, você aprenderá os conceitos gerais do processo unificado. Também será possível identificar as suas fases, assim como relacionar seus diferentes desenvolvimentos, iterativos ou incrementais.

Bons estudos.

**Ao final desta Unidade de Aprendizagem, você deve apresentar os seguintes aprendizados:**

- Explicar os conceitos gerais do processo unificado (PU);
- Relacionar o desenvolvimento iterativo com o incremental.
- Identificar as fases do PU.

# **Desafio**

---

Gerenciar riscos significa identificá-los e traçar planos para minimizar os seus efeitos sobre o projeto. Para identificar os riscos, podemos contar com a experiência da equipe obtida em projetos anteriores.

Identificar e classificar os riscos é importante para garantir a fluidez e o sucesso do projeto.

Baseado nesses fatos, considere o seguinte cenário:



Você é o gestor de TI de uma pequena empresa e acabou de sair de uma reunião com o cliente que deseja comprar um software para gerenciamento de sua escola.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

O cliente comentou na reunião que são oferecidos diversos cursos contendo disciplinas diversas. Para fazer as matrículas, os alunos precisam ser cadastrados com nome, endereço, telefone e um código da matrícula para manter o controle. Além disso, é necessário manter o registro dos professores que vão ministrar as disciplinas aos alunos.



O cliente explicou as necessidades e agora você precisa dar início ao projeto de acordo com a fase concepção do processo unificado, ou seja, fazer a identificação de riscos e uma listagem inicial dos requisitos.

1) Preencha a tabela abaixo com os requisitos levantados:

REQUISITO	DESCRIÇÃO

2) Preencha a tabela abaixo com os riscos identificados:

CÓDIGO	SEVERIDADE	DESCRIÇÃO DO RISCO	PROBABILIDADE	IMPACTO

A probabilidade pode ser: 1 – muito baixa; 2 – baixa; 3 – média; 4 – alta; 5 – muito alta.

O impacto pode ser: 1 – muito baixo; 2 – baixo; 3 – médio; 4 – alto; 5 – muito alto.

**1. Preencha a tabela abaixo com os requisitos levantados:**

<b>REQUISITO</b>	<b>DESCRIÇÃO</b>

**2. Preencha a tabela abaixo com os riscos identificados:**

<b>CÓDIGO</b>	<b>SEVERIDADE</b>	<b>DESCRIÇÃO DO RISCO</b>	<b>PROBABILIDADE</b>	<b>IMPACTO</b>

A probabilidade pode ser: 1 – muito baixa; 2 – baixa; 3 – média; 4 – alta; 5 – muito alta.

O impacto pode ser: 1 – muito baixo; 2 – baixo; 3 – médio; 4 – alto; 5 – muito alto.

# Infográfico

Os desenvolvimentos incremental e iterativo foram propostos como melhorias para os problemas encontrados no modelo em cascata.

Esse tipo de processo desenvolve um produto de *software* em ciclos. O processo iterativo acaba sendo também incremental e vice-versa, pois ao desenvolver as partes no processo incremental, é feita uma iteração.

Veja, no Infográfico, as diferenças, vantagens e desvantagens do desenvolvimento incremental e do iterativo.

## Processo unificado

As principais diferenças entre desenvolvimento iterativo e desenvolvimento incremental

### O desenvolvimento iterativo:

Aqui a ideia é **melhorar ou refinar o software pouco a pouco**, por meio de iterações.

A cada iteração **são identificados e especificados os requisitos mais relevantes**, implementando por meio de componentes.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

### VANTAGENS

Baseia-se na **participação e na boa comunicação** entre desenvolvedores e usuários.

### DESVANTAGENS

As adaptações feitas durante o processo de desenvolvimento **podem tornar o projeto totalmente diferente** da ideia original.

A **cada ciclo** do sistema os usuários e clientes **podem utilizar-lo**.

Durante o desenvolvimento, **novos requisitos podem aparecer**, dando a sensação de que o projeto nunca finalizará.

Ao final de cada iteração, **pode-se ter um feedback** de como está o projeto.

Inexperiência com a forma de trabalhar do modelo iterativo **pode levar a problemas no gerenciamento do processo**.

Alterações nos requisitos **podem ser rapidamente incorporadas** na processo de desenvolvimento.

### Desenvolvimento incremental:

Aqui a ideia é **aumentar o sistema pouco a pouco**, com sucessivos **incrementsos a partir da primeira versão** levantada.

Um projeto inicialmente simples **pode se tornar mais detalhado e sofisticado**.



### VANTAGENS

As versões são **fornecidas após cada iteração** (ciclo) do modelo incremental.

### DESVANTAGENS

Cada fase de uma iteração é **rígida** e **não sobrepõe** uma a outra.

É **flexível e fácil** de gerenciar.

O modelo incremental **precisa ser relativamente pequeno**.

O gerenciamento de riscos é facilitado, pois o **cliente usará o sistema após cada versão liberada**.

**Difícil mapear os requisitos dos usuários** dentro de incrementos.

**Os testes são simples**.

**Baixo risco de falha** geral do projeto.

# **Conteúdo do Livro**

---

O processo unificado (UP – *Unified Process*) é um processo de desenvolvimento de *software* baseado em componentes interconectados. É um processo iterativo e adaptativo que permite conduzir de forma organizada e consistente o projeto de *software*.

No Capítulo Processo unificado, do Livro *Engenharia de software* você vai aprender os conceitos gerais do processo unificado, bem como as suas fases.

Boa leitura.

# ENGENHARIA DE SOFTWARE

A photograph showing three people working together at a table. A woman with curly hair is leaning over a tablet, pointing at it while speaking. A man is seated next to her, looking down at the screen. A woman is also visible, partially seen from behind, looking at the tablet. They are surrounded by papers, books, and a coffee cup. The background is blurred.

Adriana de Souza  
Vettorazzo

# Processo Unificado

## Objetivos de aprendizagem

Ao final deste texto, você deve apresentar os seguintes aprendizados:

- Explicar os conceitos gerais do Processo Unificado (PU).
- Identificar as fases do PU.
- Relacionar o desenvolvimento iterativo e incremental.

## Introdução

Processo Unificado (PU) é um modelo de processo de *software* que visa à construção de *software* orientado a objetos. Ele pode ser personalizado de acordo com as necessidades e recursos disponíveis para cada projeto. Neste capítulo, você aprenderá os conceitos gerais do PU, identificando suas fases.

## Processo Unificado (PU)

O PU foi proposto pela *Rational* e tornou-se RUP (*Rational Unified Process*), com a intenção de apoiar o desenvolvimento de *software* orientado a objeto, proporcionando vantagens na utilização da UML (*Unified Modeling Language*).

O RUP é um *framework* que pode ser melhorado para uma ampla classe de sistemas de *software* e aplicado em diferentes áreas, tipos de organização e tamanhos de projetos. Seu objetivo principal é atender às necessidades dos usuários e ao mesmo tempo garantir a produção de um *software* de qualidade dentro do cronograma esperado com o orçamento estipulado.

O RUP apresenta como será a construção do sistema já na fase de implementação, mostrando o modelo do projeto e definindo o responsável em cada tarefa e como deverão ser executadas cada uma delas, por meio da descrição de metas. Esse *framework* reconhece que os modelos convencionais de processo apresentam uma visão única de processo. Por outro lado, o RUP é geralmente descrito a partir de três perspectivas (SOMMERVILLE, 2011):

1. perspectiva dinâmica, que mostra as fases do modelo ao longo do tempo;
2. perspectiva estática, que mostra as atividades realizadas no processo;
3. perspectiva prática, que sugere as boas práticas a serem usadas durante o processo.

O PU visa a tornar claras as tarefas dos grupos ou indivíduos que estão envolvidos no desenvolvimento de um projeto. Quanto antes se definir quais as etapas (iterações) e os artefatos que serão envolvidos durante o processo, melhor.

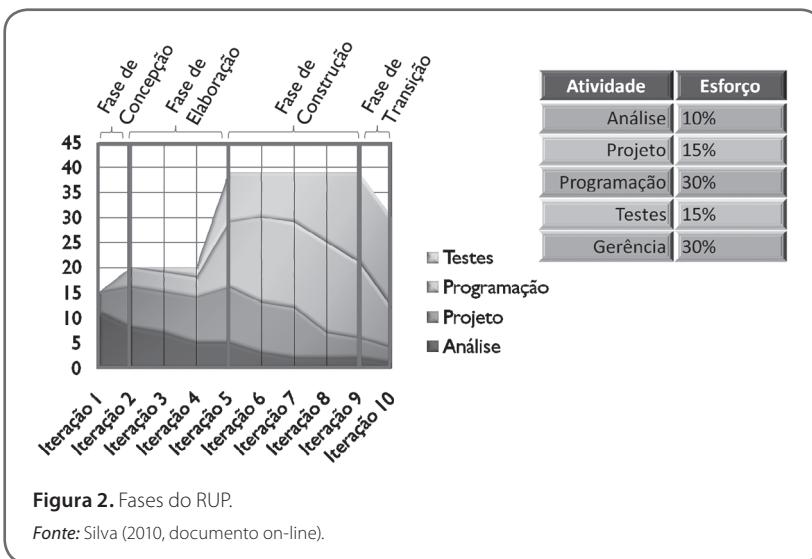
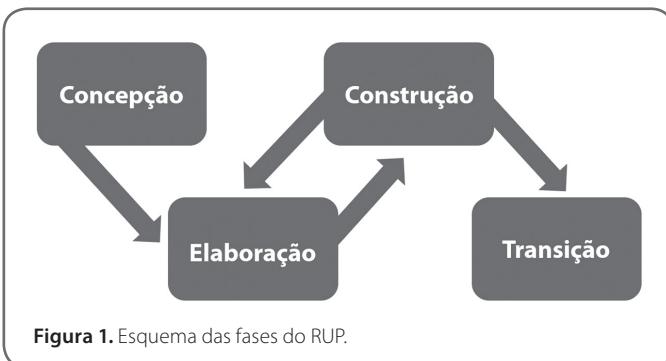
Por essas características, conclui-se que o PU é um modelo que pode ser configurado ou ajustado de acordo com o tipo de projeto a ser desenvolvido.

## Fases do RUP

O RUP está dividido em quatro fases, a fim de organizar as questões de planejamento, levantamento de requisitos, análise, implementação, teste e implantação do sistema. Cada uma dessas fases possui um papel importante para que o objetivo final seja alcançado.

- **Fase de concepção/iniciação:** nesta fase, estão incluídas as tarefas de comunicação com o cliente e também o planejamento do projeto. Planos de avaliação de riscos, de estimativa de custos e prazos, as prioridades e o levantamento de requisitos são elaborados para serem analisados previamente. Desta forma, as partes interessadas estarão cientes da complexidade do projeto e opinarão quanto à continuidade do desenvolvimento.
- **Fase de elaboração:** nesta fase, apresenta-se o modelo genérico do processo de desenvolvimento de *software*. Seu principal objetivo é fazer a análise do projeto de forma mais detalhada, revisando todos os riscos levantados.
- **Fase de construção:** são desenvolvidos os componentes do *software*. Esta fase tem como principal objetivo a construção do sistema com foco no desenvolvimento dos componentes e dos recursos que serão utilizados.
- **Fase de transição:** o objetivo principal desta fase é a entrega do *software* ao usuário e o início da fase de testes, incluindo atividades de treinamento.

As Figuras 1 e 2 ilustram as fases do RUP.



As características básicas do PU são descritas a seguir.

1. O PU é direcionado por casos de uso: o processo deve ser iniciado com a utilização de casos de uso e com a finalidade de definir uma linguagem entre o usuário e o sistema, facilitando a especificação de requisitos.
2. O processo é centrado na arquitetura: ela é modelada por meio de aspectos estáticos e dinâmicos do projeto.
3. O PU é iterativo e incremental: os projetos são divididos em projetos menores que, por sua vez, possuem iteração que quase sempre abrange todo o fluxo de trabalho. Essa iteração resulta em um incremento para o projeto e são planejadas de acordo com os casos de uso.

## Vantagens e desvantagens do PU

O PU possui vantagens e desvantagens no seu uso. Entre as vantagens apresentadas pelo PU, destacam-se o fato de que ele é um processo robusto e bem definido, que gera artefatos importantes. É um método evolucionário de desenvolvimento de *software*, em que o usuário não espera até a conclusão do projeto para ter contato com o *software*, devido ao modelo incremental. Além disso, os maiores riscos são atacados primeiro, diminuindo as chances de fracasso do projeto. Após o término do desenvolvimento, é muito difícil encontrar novos erros.

Entre as desvantagens, são destacados alguns aspectos importantes, como mostra o Quadro 1. O PU pode ser considerado complexo e trabalhoso para projetos de pequeno porte. Além disso, sua utilização provoca o aumento de gastos, devido à implantação da versão a cada incremento. Outro aspecto a ser considerado é que podem ocorrer divergências entre a documentação e o *software*. Pode entrar em *loop* devido ao modelo iterativo e incremental, dependendo do cliente para chegar ao fim do projeto. Por fim, ele exige experiência da equipe.

**Quadro 1.** Workflows estáticos no RUP

Workflow	Descrição
Modelagem de negócios	Os processos de negócios são modelados usando casos de uso de negócios.
Requisitos	Os agentes que interagem com o sistema são identificados e os casos de uso são desenvolvidos para modelar os requisitos de sistema.
Análise e projeto	Um modelo de projeto é criado e documentado usando modelos de arquitetura, modelos de componente, modelos de objeto e modelos de sequência.
Implementação	Os componentes de sistema são implementados e estruturados em subsistemas de implementação. A geração automática de código com base nos modelos de projeto ajuda a acelerar esse processo.
Teste	O teste é um processo iterativo realizado em conjunto com a implementação. O teste de sistema segue o término da implementação.
Implantação	Uma versão do produto é criada, distribuída aos usuários e instalada no local de trabalho.

(Continua)

(Continuação)

**Quadro 1.** Workflows estáticos no RUP

Workflow	Descrição
Gerenciamento de configuração e mudanças	Este workflow de apoio gerencia o desenvolvimento do sistema.
Ambiente	Este workflow está relacionado à disponibilização de ferramentas apropriadas de <i>software</i> para a equipe de desenvolvimento.

**Fonte:** Adaptado de Sommerville (2011).



### Fique atento

Os modelos de processo de *software* existem para auxiliar no desenvolvimento com a utilização das melhores práticas.

Porém, a metodologia escolhida poderá ser adaptada de acordo com a necessidade do projeto, da empresa e até mesmo dos recursos disponíveis.

## Desenvolvimento iterativo e incremental

O desenvolvimento iterativo e incremental trouxe agilidade e flexibilidade para as equipes de desenvolvimento. O desenvolvimento aumentou a capacidade de atender às demandas do mercado, atingindo melhores resultados em menor prazo.

Entre as principais características do desenvolvimento iterativo e incremental, destaca-se o fato de que esse desenvolvimento se baseia nas metodologias ágeis, pautadas por vários ciclos curtos em que são inclusas funcionalidades, além de coletar *feedbacks* e revisar os requisitos.

Neste modelo, pode-se atingir maior nível de satisfação do cliente e garantir que o resultado final esteja dentro do esperado.

## Vantagens do desenvolvimento iterativo e incremental

Adotar este tipo de processo pode trazer uma série de vantagens, tais como:

- criação de um fluxo de entrega de *software* em que os requisitos são apresentados em partes funcionais menores;
- facilidade de acompanhar a evolução do desenvolvimento e de identificar erros e falhas durante o desenvolvimento;
- redução de riscos na etapa do projeto;
- maior capacidade de modificar a direção do projeto;
- desenvolvimento de escopo de *software* mais flexível;
- maior agilidade e produtividade da equipe.

Para adotar o desenvolvimento iterativo e incremental, é necessário adotar processos iterativos e incrementais e também mudanças de rotina de criação de *software*, o que pode ser feito em três passos:

1. definir corretamente as prioridades;
2. distribuir responsabilidades entre os times;
3. distribuir as entregas.



### Referências

SILVA, V. T. *Processo unificado*. 2010. Disponível em: <<http://www2.ic.uff.br/~viviane.silva/2010.1/es1/util/aula3.pdf>>. Acesso em: 26 out. 2018.

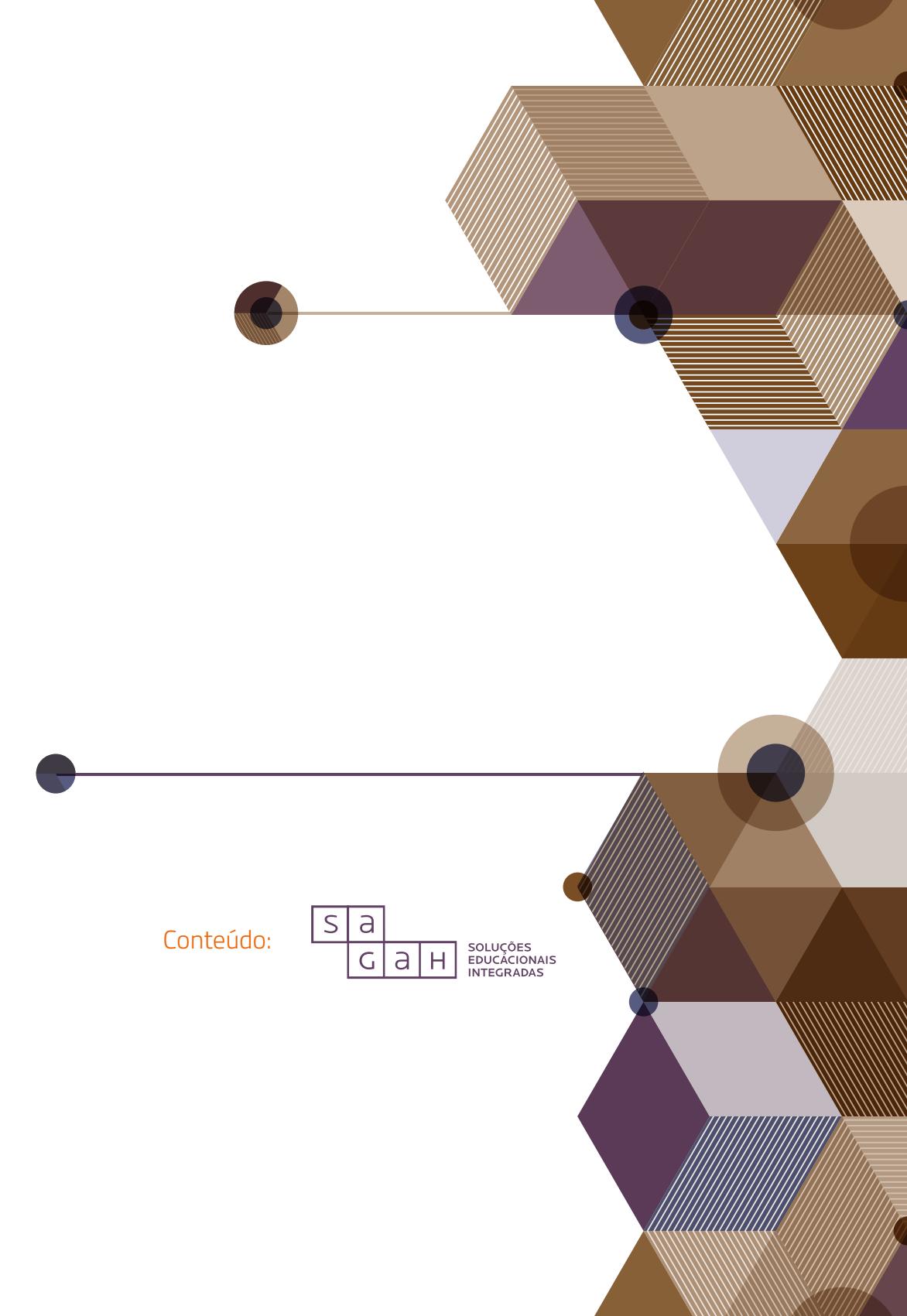
SOMMERVILLE, I. *Engenharia de software*. 9. ed. São Paulo: Pearson, 2011.

### Leituras recomendadas

PRESSMAN, R. S.; MAXIM, B. R. *Engenharia de software: uma abordagem profissional*. 8. ed. Porto Alegre: Penso, 2016.

STEFFEN, J. B. *O que são essas tais de metodologias Ágeis?* 23 jan. 2012. Disponível em: <[https://www.ibm.com/developerworks/community/blogs/rationalbrasil/entry/mas\\_o\\_que\\_s\\_c3\\_a3o\\_essas\\_tais\\_de\\_metodologias\\_\\_c3\\_a1geis?lang=en](https://www.ibm.com/developerworks/community/blogs/rationalbrasil/entry/mas_o_que_s_c3_a3o_essas_tais_de_metodologias__c3_a1geis?lang=en)>. Acesso em: 19 out. 2018.

Encerra aqui o trecho do livro disponibilizado para esta Unidade de Aprendizagem. Na Biblioteca Virtual da Instituição, você encontra a obra na íntegra.



Conteúdo:



SOLUÇÕES  
EDUCACIONAIS  
INTEGRADAS

# Dica do Professor

---

Na fase *Elaboração* do processo unificado, uma das tarefas é gerenciar os riscos encontrados.

O gerenciamento de riscos é o processo de manter a rastreabilidade dos riscos identificados, monitorar riscos residuais e identificar novos riscos. Assegurar a execução dos planos de risco e avaliar a sua efetividade na redução dos riscos.

A Dica do Professor a seguir apresenta dicas para fazer o gerenciamento dos riscos encontrados.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

# Exercícios

- 1) Processo unificado é um processo de desenvolvimento ligado à orientação a objetos, mas que também pode ser utilizado em projeto estruturado, sem perder suas características básicas.

Baseado nesta afirmativa, analise as opções abaixo e assinale a que corresponde ao princípio fundamental do PU.

- A) Ser concentrado em arquitetura.
- B) Ser desenvolvido em cascata.
- C) Empregar *times* autodirigidos e auto-organizados.
- D) Ser programado em pares.
- E) Ter a propriedade coletiva do código-fonte.

- 2) Uma das práticas do processo unificado é dividir grandes projetos em miniprojetos. Eles possuem uma iteração, que abrange quase todo o fluxo de trabalho, o que resulta em um incremento para os projetos, além de serem planejados de acordo com os casos de uso.

Dessa forma, analise as alternativas abaixo e assinale a que caracteriza o processo unificado.

- A) Ciclo de desenvolvimento de *software* em cascata.
- B) Ciclo de desenvolvimento de *software* sequencial.
- C) Processo de *software* específico para reengenharia.
- D) Processo de *software* iterativo e incremental.
- E) Processo de *software* iterativo.

- 3) O processo unificado é um modelo configurável, ou seja, ele pode ser ajustado de acordo com os tipos de projeto que será desenvolvido. Esse modelo possui quatro fases no seu desenvolvimento.

**Analise, dentre as alternativas a seguir, a que apresenta as atividades que correspondem às fases:**

**1. Concepção e 2. Transição do PU**

- A)** 1. Essa fase tem como objetivo verificar a viabilidade do projeto, bem como os riscos e definir os casos de uso mais críticos obtendo as funções-chave do sistema.
2. O objetivo dessa fase é garantir que todos os requisitos do projeto foram atendidos e implementados corretamente.
- B)** 1. O objetivo dessa fase é garantir que todos os requisitos do projeto foram atendidos e implementados corretamente.
2. A fusão de vários artefatos de *software*, possibilitando que o sistema seja implementado quase que completamente.
- C)** 1. Essa fase tem como objetivo verificar a viabilidade do projeto, bem como os riscos e definir os casos de uso mais críticos, obtendo as funções-chave do sistema.
2. A fusão de vários artefatos de *software*, possibilitando que o sistema seja implementado quase que completamente.
- D)** 1. Durante essa fase, a maioria dos casos de uso são especificados e detalhados.
2. Essa fase tem como objetivo verificar a viabilidade do projeto, bem como os riscos e definir os casos de uso mais críticos, obtendo as funções-chave do sistema.
- E)** 1. Essa fase tem como objetivo verificar a viabilidade do projeto, bem como os riscos e definir os casos de uso mais críticos obtendo as funções-chave do sistema.
2. A fusão de vários artefatos de *software*, possibilitando que o sistema seja implementado quase que completamente.

O processo unificado consiste da repetição de uma série de ciclos durante a vida de um sistema e, a cada ciclo concluído, uma nova versão do sistema é disponibilizada para utilização do usuário. Esses ciclos consistem em quatro fases.

Tendo em vista as fases do PU, analise as afirmativas abaixo e assinale a que corresponde à ordem cronológica das fases.

- A) *Concepção, Requisitos, Implementação e Testes.*
  - B) *Concepção, Elaboração, Construção e Implementação.*
  - C) *Concepção, Elaboração, Construção e Transição.*
  - D) *Elaboração, Concepção, Construção e Transição.*
  - E) *Elaboração, Construção, Implementação e Transição.*
- 5) As fases do processo unificado são quatro: *Concepção, Elaboração, Construção e Transição*. Cada uma dessas fases possui um conjunto de atividades que ajudarão na fase seguinte. Portanto, elas seguem uma ordem.

Dentre as quatro fases do PU, quais são as que correspondem às de atividade de desenvolvimento de software mais acentuada?

- A) *Elaboração e Construção*
- B) *Elaboração e Concepção*
- C) *Concepção e Transição*
- D) *Construção e Transição*
- E) *Construção e Concepção*

## Na prática

Uma das características do processo unificado é o fato de o modelo estar direcionado por casos de uso. No início do processo, a utilização dos casos de uso ajuda a definir uma linguagem entre os usuários e o sistema. Isso facilita a especificação dos requisitos.

Veja, Na Prática, um exemplo da utilização de casos de uso no processo de desenvolvimento de um *software*.

## Casos de Uso

Neste exemplo prático, será utilizada a descrição de caso de uso "Matricular aluno".

Na utilização de casos de uso são necessários descrição, objetivo e ator envolvido.

### Primeiro passo: Definir atores e objetivo

#### "Matricular aluno"



#### Descrição:

Possibilitar que ocorra a matrícula de alunos em turmas.

#### Objetivo:

Possibilidade que ocorra a matrícula de alunos em turmas.

#### Ator envolvido:



### Segundo passo: Descrição da iteração entre o ator e o sistema

SECRETARIA	SISTEMA
Solicita a matrícula de alunos em turmas	Exibe uma interface com uma lista de turmas cadastradas contendo descrição do curso, descrição da disciplina, ano, semestre e descrição da turma.
	Exibe uma lista de nomes de alunos cadastrados.
Seleciona uma turma e o aluno a ser matriculado	
	Armazena a matrícula.
	Retorna o resultado da operação.
	Fecha a interface.

### Terceiro passo: Declarar regras e exceções

#### Regras de negócio:

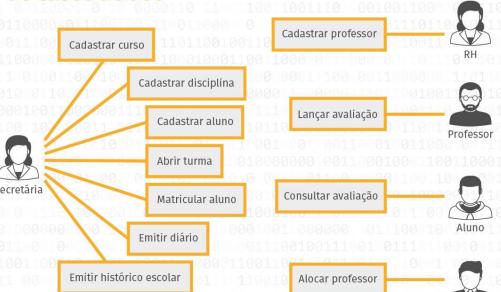
RN01 – o aluno somente pode ser matriculado na turma de uma disciplina que não possui pré-requisito(s) ou que, se tiver, o aluno deverá ter cursado-o anteriormente e ter obtido aprovação.

RN02 – o aluno somente pode ser matriculado em turmas de disciplinas do curso em que esteja matriculado.

#### Exceções:

EX01 – caso o aluno já se encontre matriculado na turma, a mensagem "Este aluno já possui matrícula na turma" deverá ser apresentada e a matrícula não será efetivada.

### Quarto passo: Montar o diagrama de casos de uso



Durante a análise, o projeto e a implementação, os casos de uso são levantados e, na fase de testes, é verificado se o sistema está seguindo o que está descrito no modelo de casos de uso.

Os casos de uso são usados no planejamento e no acompanhamento das iterações.



Aponte a câmera para o código e accese o link do conteúdo ou clique no código para accesar.

# Saiba mais

Para ampliar o seu conhecimento a respeito desse assunto, veja abaixo as sugestões do professor:

## **Introdução ao processo unificado**

Processo unificado é um processo iterativo e adaptativo de desenvolvimento e vem ganhando cada vez mais adeptos devido à maneira organizada e consistente que permite conduzir um projeto.



Aponte a câmera para o código e accese o link do conteúdo ou clique no código para acessar.

## **Engenharia de Software**

Leia a obra a seguir e aproveite para aprofundar seu conhecimento a respeito do assunto desta unidade de aprendizagem.

**Conteúdo interativo disponível na plataforma de ensino!**