

— Exame —

Desenvolvimento de Sistemas de Informação

LESI/LMCC
Época Especial - 2006/07
19/09/2007

Duração máxima: 2h00

**Leia o exame com atenção e responda utilizando UML 2.0.
Cada grupo deve ser respondido em folhas diferentes.**

Grupo I

1. Considere os seguintes *Use Case* em que se modela a inscrição de um aluno num curso (note que neste caso o *Sistema* é a Universidade e não o Software):

Use Case Inscrição

Comportamento Normal:

1. O Aluno preenche o formulário de inscrição.
2. O Aluno entrega o formulário ao funcionário dos Serviços Académicos.
3. O Funcionário inspecciona o formulário e considera-o correcto.
4. «include» Criar Aluno
5. «include» Pagar Propinas

Comportamento Alternativo

4. O Funcionário inspecciona o formulário e considera-o incorrecto.
5. O Funcionário devolve o formulário ao aluno.
6. Use case termina

Use Case Criar Aluno

Comportamento Normal:

1. O Funcionário clica no ícone Criar Aluno.
2. O Software apresenta o écran de criação de alunos.
3. O Funcionário introduz o nome, endereço e número de telefone do aluno.

Use Case Pagar Propinas

Comportamento Normal:

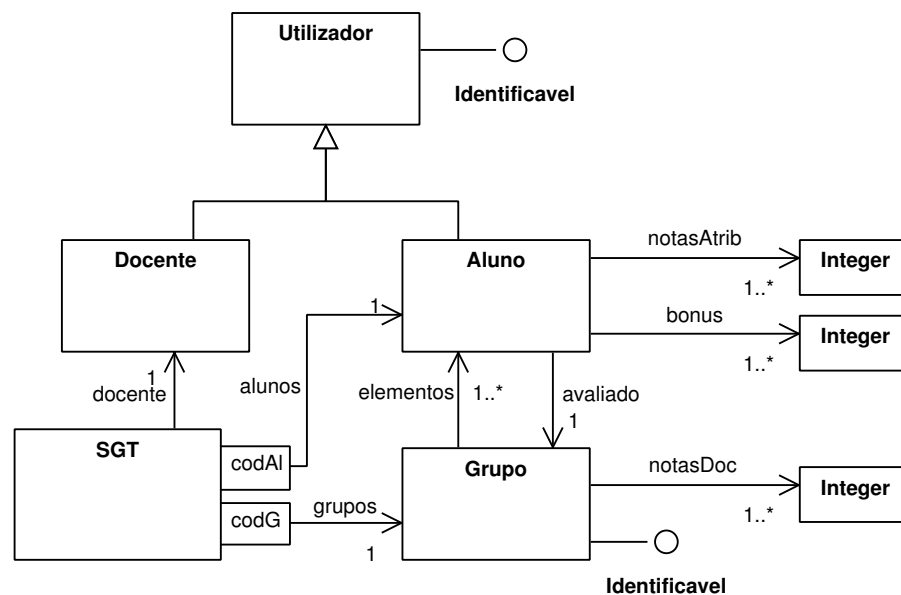
1. O Software apresenta o écran de valor total de propinas.
2. O Funcionário pede ao aluno o valor das propinas.
3. O Aluno paga as propinas.
4. O Funcionário regista o pagamento.
5. O Software emite um recibo.
6. O Funcionário entrega o recibo ao aluno.

Comportamento Alternativo

3. O Aluno não paga as propinas.
4. O Funcionário regista falta de pagamento.
5. O Software coloca o aluno em situação irregular.
6. Use case termina.

Sabendo que se pretendem modelar os comportamentos alternativos com *<< extends >>*, desenhe o **Diagrama de Use Case** correspondente à descrição fornecida.

2. Considere a proposta de arquitectura apresentada na figura, que relembra o processo de atribuição de notas dos trabalhos práticos. De uma forma geral, um dos aspectos menos conseguidos nos trabalhos entregues tem a ver com a componente de avaliação dos trabalhos dos colegas por parte dos alunos. Considere a proposta de arquitectura apresentada na figura (para simplificar foi considerado que cada aluno dá uma nota a cada um dos relatórios de um dado grupo definido no início do semestre).



Created with Poseidon for UML Community Edition. Not for Commercial Use.

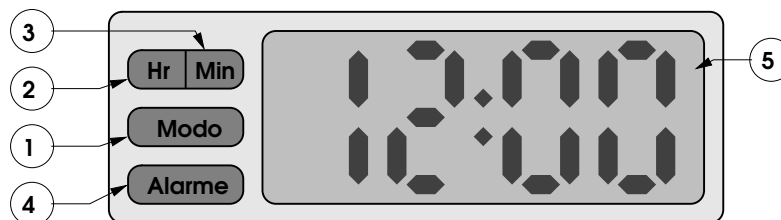
- (a) Escreva o código Java (esqueleto de declaração das classes) para as classes SGT (Sistema de Gestão de Trabalhos), Aluno e Grupo.
- (b) Considere agora que os requisitos do sistema foram alterados e que:
- em vez de um docente o sistema deverá registar informação sobre vários docentes (existindo um que é o regente da disciplina),
 - cada grupo deverá ter um docente como supervisor.

Redesenhe o **Diagrama de Classes** tendo em conta estas alterações.

- (c) Desenhe um **Diagrama de Sequência** para o método `List<Aluno> alunosComNotaMaiorQue(int nota)`, que determina a lista dos alunos em que o somatório das várias notas atribuídas pelo docente (para as várias fases) é superior à nota passada como parâmetro.

Grupo II

3. Considere o seguinte excerto de um manual de instruções para um relógio despertador:



- ① Botão "Mode" — prima este botão para percorrer repetidamente os diferentes modos de funcionamento do relógio:
Apresentação da hora actual → Regulação da hora actual → Regulação da hora do alarme → ... (volta para Apresentação da hora actual)
- ② Botão "Hr (hora)" — prima este botão para incrementar a hora quando em modo de regulação de hora actual/alarme (não produz qualquer efeito em modo de apresentação da hora actual).
- ③ Botão "Min (minutos)" — prima este botão para incrementar os minutos quando em modo de regulação de hora actual/alarme (não produz qualquer efeito em modo de apresentação da hora actual).
- ④ Botão "Alarme" — prima este botão para activar/desactivar o alarme (se está activo desactiva e vice-versa). O relógio começa a tocar quando o alarme está activo, o relógio está em modo de apresentação da hora, e a hora actual é igual à hora do alarme (o alarme desliga-se automaticamente após 3 minutos).

- ⑤ Écran — apresenta a hora correspondente ao modo em que o relógio se encontra.

Desenhe um **Diagrama de Estados** que represente o funcionamento do relógio tal como descrito.

4. Escreva um **Diagrama de Actividades** que represente um processo de encomenda numa loja online, com o seguinte comportamento:

Um formulário de encomenda online é normalmente utilizado para realizar a encomenda. O formulário de encomenda passa por vários estados, desde o pedido inicial até ao seu arquivo. Após preencher este formulário, o cliente envia-o para a empresa para processamento.

Se a quantia for pequena (menos de EUR1000.00), o funcionário despacha o pedido e encaminha este para o serviço de aprovisionamento. Ao receber a encomenda o serviço de aprovisionamento verifica se tem stock. Em caso afirmativo a encomenda é despachada à cobrança. Se não existirem em stock os itens necessários para responder ao pedido é necessário pedir à fábrica que produza os itens em falta e aguardar a recepção dos mesmos, seguindo-se o envio da encomenda para o cliente. Após o pagamento do cliente o pedido de encomenda é arquivado.

Se a quantia for elevada (EUR1000.00 ou mais), o funcionário requer ao cliente que efectue uma transferência do montante da encomenda antes do seu envio. À parte do pagamento, o resto do processo é idêntico.