



| | | | |
|----------------------------------|------------------------|-------|---------|
| Disciplina | Departamento | Turno | Período |
| Programação Modular | Engenharia de Software | Noite | 2º |
| Professor | | | |
| Hugo de Paula (hugo@pucminas.br) | | | |

Trabalho Prático de Programação Modular sobre DAO mesmo grupo do TIS, ou grupos de até 4 pessoas

1 Requisitos do trabalho

O objetivo deste trabalho é que os conceitos aprendidos ao longo da disciplina de Programação Modular sejam aplicados em conjunto em um programa coeso. Baseado na especificação de um sistema orientado para objetos, o grupo deverá desenvolver um modelo de persistência em Java baseado no *design pattern* DAO (*Data Access Object*), utilizando arquivos. Seguindo o modelo de interdisciplinaridade, esse trabalho utilizará o problema modelado na disciplina Trabalho Interdisciplinar de Software II (TIS 2).

IMPORTANTE: os grupos devem ser os mesmos de TIS 2. Para os alunos que não estejam cursando TIS 2, eles deverão formar grupos de até 4 integrantes. Não é permitido misturar alunos de TIS 2 com alunos que não estejam cursando a disciplina.

Este trabalho deve conter os seguintes requisitos mínimos de programação:

1. O sistema deve ser desenvolvido em Java, utilizando arquivos (podem ser texto ou binário).
2. As cláusulas **import** devem ser usadas adequadamente, sem o uso de * e sem importar classes que não serão utilizadas.
3. Deverá ser elaborado um diagrama de classe representando a aplicação.
4. O módulo de persistência deverá ser modelado e inserido no diagrama de classes da aplicação.
5. Deverá ser criada uma classe de teste *JUnit* para cada uma das entidades de persistência do sistema, testando os métodos CRUD (inserir, consultar, atualizar e remover).
6. A documentação deve conter apenas a descrição das classes desenvolvidas, sem a necessidade de fazer revisão bibliográfica sobre o padrão DAO.
7. Observe atentamente os princípios de reusabilidade, coesão, acoplamento, abertura/fechamento e a utilização de padrões de projeto se forem pertinentes.

2 Especificação do sistema

2.1 Para trabalho integrado com o TIS 2

Para os grupos que irão fazer o trabalho de forma integrada com o TIS 2, o diagrama de classes que será persistido é o mesmo da aplicação que está sendo modelada no TIS 2. Esse trabalho é totalmente integrado com a disciplina de TIS.

2.2 Para grupos que não cursam TIS 2

Os grupos cujos integrantes não estejam cursando TIS 2 deverão utilizar como base os requisitos de sistema apresentados a seguir.

Deseja-se desenvolver um sistema para fazer a gestão da monitoria dos cursos do ICEI/PUC Minas. A monitoria ocorre em três etapas:

A 1a etapa é a etapa de SELEÇÃO DE MONITORES. O Coordenador de Monitoria de cada curso publica o Edital de Monitoria, constando as vagas disponíveis. Uma vaga de monitoria é definida pela Disciplina, o turno e a carga horária. Um exemplo de Edital de Monitoria encontra-se disponível no SGA. Esse Edital é então divulgado para os alunos. Os alunos se candidatam às vagas de monitoria, obedecendo aos requisitos do Edital. O Coordenador de monitoria seleciona os monitores baseados nos critérios do Edital e divulga aos candidatos o resultado final da seleção, e as instruções de contratação.

A 2a etapa é a fase de DEFINIÇÃO DOS ATENDIMENTOS. Os monitores encaminham propostas de horários de atendimento dentro das regras de atendimento definidas no edital. A Coordenação e alunos chegam a um consenso sobre o horário final. O Coordenador consolida o horário em um documento e divulga para os alunos. Os monitores se apresentam aos alunos da turma e preenchem um plano de monitoria juntamente com o professor da disciplina.

A 3a etapa consiste no atendimento aos alunos e gestão dos monitores. Os alunos comparecem à monitoria. Os monitores precisam registrar a sua presença diária (ponto) e os atendimentos para elaboração do relatório final de atendimentos. Os monitores podem interagir com alunos e professores.

3 Resultados esperados

Deverá ser entregue em um arquivo comprimido em formato .zip, contendo a Documentação e o código fonte. O nome do arquivo .zip deve ser a concatenação dos nomes dos integrantes do grupo.