



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

Trabalho Teórico de Programação Modular

1 Introdução

As linguagens de programação como o Java possuem estruturas semânticas e APIs que promovem a reutilização de código, a produtividade e favorecem a obtenção da qualidade de software conforme apresentado na disciplina. Essas estruturas e APIs podem impactar a forma como um programador codifica um programa ou resolve um problema.

2 Especificação do problema

Este trabalho deverá ser desenvolvido em grupos de até 4 (quatro) pessoas. Neste trabalho, o grupo deverá estudar o tema *Lambda Expressions and functional interfaces*. Em especial, o grupo deverá abordar os seguintes tópicos (mas não se limitar apenas a eles):

- O que são funções lambda.
- Como utilizá-las em diversos pontos do código, tais como implementar interfaces, substituir classes anônimas, criar funções de ordem superior, etc.
- Como utilizá-las em conjunto com a *Java Collections Framework*.
- Como utilizar a *Streams API*.

Uma boa referência e ponto de partida para analisar os aspectos mais avançados da linguagem podem ser encontrados no livro *Effective Java, 3rd Ed.*, Bloch [2018]. É importante que se use a terceira edição (disponível apenas em inglês), pois está atualizada para o Java 9.

O trabalho deve conter:

1. Introdução e contextualização do tema.
2. Descrição da tecnologia: como funciona, seus principais recursos de programação.
3. Vantagens da tecnologia para o desenvolvimento de um software modular. Análise do impacto do uso dessa tecnologia para a melhoria da qualidade de software.
4. Apresentação de um cenário de uso da tecnologia: exemplo de aplicação, com código. Utilize diagrama UML para ilustrar o cenário.

3 Resultados esperados

Deverão ser entregues 3 artefatos:

1. Uma documentação em PDF.
2. Um vídeo tutorial.
3. O código fonte completo do projeto desenvolvido no Eclipse.

3.1 Documentação

O grupo deverá desenvolver um pequeno *hand-out* de até no máximo 6 páginas, que descreva o tópico estudado. É incentivada a utilização de recursos visuais, tais como exemplos, tabelas, figuras, diagramas UML, etc, para ilustrar o tema.

O *Hand-out* deverá ser escrito utilizando o modelo L^AT_EX de artigo da revista *Abakós*, do Instituto de Ciências Exatas e Informática da PUC Minas, disponibilizado no link: <https://www.overleaf.com/latex/templates/modelo-canonical-abakos-puc-minas/czwspnpvhdwb>. O texto deve ser claro e sucinto. Serão avaliados, além do conteúdo, a forma de apresentação, a correção no uso da língua portuguesa e a clareza na exposição dos argumentos.

3.2 Vídeo

Neste trabalho deverá ser produzido um vídeo, com até no máximo 15 minutos sobre o conteúdo estudado (ideal são 12 minutos). O objetivo principal do vídeo é apresentar a ideia geral do tema e como ele pode ser aplicado, utilizando exemplos. O vídeo deve ter um caráter mais motivacional que técnico.

A estrutura do vídeo deve ter foco no engajamento do espectador, e deve possuir a seguinte anatomia: nos primeiros 3 minutos toda a ideia geral já deve ter sido transmitida; em seguida devem entrar as explicações e exemplos. Até os 9 a 12 minutos, todo o conteúdo essencial deve ter sido dado e deve-se iniciar a conclusão.

3.3 Forma de entrega do trabalho

Como o SGA não permite a postagem de arquivos com mais de 3 Mb, o aluno deverá disponibilizar o *vídeo* e o Projeto Eclipse completo em algum mecanismo de compartilhamento de arquivos (4shared, CloudStore, Box, Dropbox, Google Drive, OneDrive, etc).

O *hand-out* deve conter o link compartilhado logo na página inicial do trabalho.

A documentação deverá ser entregue em um arquivo em formato pdf, cujo nome do arquivo seja a concatenação dos nomes dos integrantes do grupo.

Referências

Joshua Bloch. *Effective Java*. Addison-Wesley, Boston, MA, 3 edition, 2018. ISBN 978-0-13-468599-1. URL <https://www.safaribooksonline.com/library/view/effective-java-third/9780134686097/>.