

MC322 – Programação Orientada a Objetos

Laboratório 08 – 2s2023

Prof. Dr. Bruno Cafeo (Professor)

Fillipe dos Santos Silva

Jesamin Zevallos Quispe

Vinicius Leme Soares

Discord: <https://discord.gg/mHcUFRDGgQ>

1 Tratamento de Exceções

A biblioteca universitária é um espaço dinâmico que atende a uma variedade de necessidades dos alunos e do corpo docente. Ela oferece serviços que vão desde o empréstimo de livros, recursos multimídia, até a reserva de salas e equipamentos. Para garantir a eficiência e flexibilidade desses serviços, é fundamental que a estrutura de dados por trás desses sistemas seja altamente reutilizável.

2 Contexto

No sistema de gerenciamento da biblioteca universitária multimídia, diversos erros podem ocorrer devido a operações inválidas, dados incorretos inseridos pelos usuários ou falhas na conexão com a base de dados. Para garantir uma boa experiência ao usuário e manter a integridade dos dados, o tratamento de exceções é fundamental.

3 Objetivo

Implementar o tratamento de exceções adequado para as seguintes situações:

3.1 Empréstimo de Item

1. Se um membro tentar emprestar um número de itens maior do que seu limite permitido.
2. Se o item solicitado para empréstimo não estiver disponível no momento.
3. Se um membro com histórico de multas pendentes tentar realizar um empréstimo.

3.2 Devolução de Item

1. Se um membro tentar devolver um item que ele não emprestou.
2. Se o item a ser devolvido estiver danificado e precisar de manutenção.

3.3 Reserva de Sala de Estudo

1. Se um membro tentar reservar uma sala que já está reservada no horário solicitado.
2. Se a capacidade máxima de pessoas para a sala for excedida.

3.4 Acesso ao Sistema

1. Se um funcionário (atendente, gerente ou administrador) tentar acessar uma funcionalidade para a qual não tem permissão.
2. Se um membro ou funcionário tentar fazer login com credenciais inválidas.

3.5 Cadastro de Itens Multimídia

1. Se tentar cadastrar um item com um ISBN ou identificador já existente na base de dados.
2. Se os dados fornecidos para o item (como formato de arquivo para e-books ou duração para CDs e DVDs) forem inválidos.

4 Instruções

1. Para cada situação acima, defina uma exceção personalizada. Por exemplo:
`ExcecaoLimiteEmprestimoExcedido`, `ExcecaoItemNaoDisponivel`, etc.
2. Implemente blocos `try-catch` nas classes e métodos apropriados para tratar cada uma das situações.
3. Nos blocos `catch`, forneça uma mensagem informativa para o usuário, indicando o problema encontrado.
4. Onde apropriado, após capturar uma exceção, o sistema deve permitir ao usuário corrigir a ação ou fornecer dados corretos.
5. Para situações mais críticas, como danos aos itens, considere a possibilidade de lançar a exceção até um nível mais alto no sistema, onde possa ser tratada adequadamente (por exemplo, registrando o dano e enviando o item para manutenção).
6. Implemente outras exceções caso julgue necessário ou pertinente.

5 Avaliação

- Entrega realizada dentro do prazo estipulado.
- Execução do código.
- Qualidade do código desenvolvido (saída dos dados na tela, tabulação, comentários).
- Desenvolvimento correto das exceções.
- **Entregar no diretório files do repositório o diagrama de classes UML completo e atualizado em relação ao laboratório anterior.**

6 Entrega

- Você não precisa criar outro repositório ou outro projeto, a ideia é avançar sobre o repositório que você já tem.
- A entrega do Laboratório é realizada exclusivamente via **Github**. Para a submissão no Github, gere um release (tag) com a identificação do laboratório no formato <lab08-RA>. Por exemplo, para o aluno com RA 123456, a tag será: lab08-123456.
- Evite criar *releases* enquanto não tiver certeza que seu código está funcionando como esperado.
- Utilize os horários de laboratório e atendimentos para tirar eventuais dúvidas de submissão e também relacionadas ao desenvolvimento do laboratório.
- Data da entrega: **01/10/2023** até às **23h59**.