



Este documento constitui o primeiro enunciado do Projeto Prático 2 de Laboratório de Programação Modular. Novos requisitos e restrições surgirão conforme o projeto evolua. Assim, fiquem atentos aos artefatos e seus prazos de entrega.

#### **Título do projeto:** Controle de Aluguel de Equipamentos

A **Xulambs Heavy Rental** (XHR) é uma conhecida empresa fornecedora de uma variedade de equipamentos para construção civil, tais como escavadeiras, betoneiras, andaimes etc. Ela está precisando modernizar seu sistema de controle de aluguéis, pois neste momento tudo é feito manualmente em planilhas isoladas.

Seu grupo de trabalho foi convocado para criar esta solução, que permitirá à XHR gerenciar eficientemente o aluguel de seus diversos equipamentos. Ao entrevistar os interessados, vocês descobriram os fatos seguintes:

- Um equipamento é identificado por um código numérico único e possui uma descrição ("nome") de pelo menos 5 caracteres.
- Os equipamentos são alugados para clientes previamente cadastrados.
- Um cliente é cadastrado com um identificador único e o nome, que pode ser real (p. ex., "Ana da Silva") ou fantasia (p. ex., "Terreirão do Galo").
- Cada equipamento tem um preço por dia e pode ser alugado por um número ilimitado de dias corridos.
- Cada aluguel de equipamento é registrado no sistema obrigatoriamente com data de início e fim.
- Um equipamento só pode ser alugado se estiver livre em todo aquele período.
- O valor total do aluguel é calculado multiplicando o número de dias de aluguel pelo valor diário do equipamento.

#### **O sistema deve ser capaz de:**

1. Registrar novos aluguéis, incluindo todas as informações necessárias.
2. Calcular e mostrar o valor total do aluguel para cada registro.
3. Manter um registro dos aluguéis passados, incluindo informações sobre o cliente, o equipamento e as datas de início e término.
4. Permitir a consulta rápida dos aluguéis atuais e passados de um cliente específico.
5. Gerar um relatório mensal que liste todos os aluguéis do mês, incluindo o faturamento total.

#### **Artefatos esperados (revisão de 30/08):**

- Arquivo readme.md corretamente preenchido no repositório do grupo no GitHub Classroom;
- Diagrama de classes contendo atributos, métodos e relacionamentos enviado ao repositório;
- Implementação e teste das classes "Equipamento", "Cliente" e "Aluguel";

#### **Atenção para:**

- Códigos entregues devem ser aderentes às classes especificadas no diagrama de classes;
- Cada classe de negócio deve estar acompanhada de sua classe de teste unitário (JUnit);
- Todos os códigos entregues devem estar devidamente documentados (Javadoc);

#### **Cronograma de entregas:**

##### **Aula de 30/08:**

- Arquivo readme.md corretamente preenchido no repositório do grupo no GitHub Classroom;
- Diagrama de classes contendo atributos, métodos e relacionamentos enviado ao repositório;

##### **Aula de 06/09:**

- Implementação e teste das classes "Equipamento", "Cliente" e "Aluguel";

##### **Aulas seguintes: a definir.**

**Instruções e observações:**

- Os grupos serão obrigatoriamente compostos por 6 alunos. Grupos com menos alunos podem receber alunos que estão sem grupos no momento; também pode haver fusão ou remanejamento de grupos para que a configuração com 6 integrantes seja alcançada.
- O projeto deve estar hospedado obrigatoriamente no repositório correspondente do GitHub Classroom;
- A execução do projeto segue, em linhas gerais, a metodologia ágil Scrum para desenvolvimento. Assim, haverá reuniões semanais nas aulas para acompanhamento da evolução do projeto;
- Lembre-se: no Scrum é importante demonstrar evolução a cada semana, e não somente a entrega;
- Sugere-se utilizar o tempo das aulas para avançar na execução do projeto;
- Os requisitos do projeto cobrem conteúdos que serão ministrados ao longo das semanas das matérias de *Programação Modular* e *Laboratório de Programação Modular*;
- Os professores auxiliarão na distribuição das tarefas, porém espera-se que o time de desenvolvimento faça uma gestão interna das pendências;
- A execução do projeto é em grupo, porém as notas são atribuídas individualmente, levando-se em consideração o comprometimento e o cumprimento de tarefas de cada aluno;
- Dado o calendário estrito do semestre, na data de finalização do projeto será feita a última inspeção no repositório do GitHub Classroom. Modificações ou inclusões posteriores só serão levadas em consideração se previamente acordadas com o professor em condições especiais.

**Critérios de pontuação:**

- Diagrama de classes atualizado e aderente: 3 pontos;
- Requisitos corretamente implementados (classes e testes): 14 pontos;
- Documentação de código: 3 pontos;
- Atraso nos artefatos: **desconto** de 1 a 4 pontos por semana.

**A nota final** se dará pela soma acima, multiplicada por um peso entre 0 e 1 relativo ao acompanhamento da participação no projeto. Outra vez, lembre-se: não é só a entrega do produto finalizado que importa, é todo o processo de sua construção e as entregas parciais para o cliente.