

# DESENVOLVIMENTO DE SOFTWARE PARA A WEB 1

Profs. Alan D. B. Valejo & Delano M. Beder (UFSCar)

---

## Atividade A-2: Sistema para locação de bicicletas

O sistema deve possuir um cadastro de clientes, com os seguintes dados: e-mail, senha, CPF, nome, telefone, sexo e data de nascimento.

O sistema deve possuir um cadastro de locadoras, com os seguintes dados: e-mail, senha, CNPJ, nome e cidade.

O sistema deve possuir um cadastro de locações, com os seguintes dados: cliente, locadora e dia/horário da locação. Assume-se que a duração da locação é de 1 hora e sempre inicia-se em “hora cheia” (13h 00min etc)

O sistema deve atender aos seguintes requisitos:

- R1: CRUD<sup>1</sup> de clientes (requer login de administrador)
- R2: CRUD de locadoras (requer login de administrador)
- R3: Listagem de todas as locadoras em uma única página (não requer login)
- R4: Listagem de todas as locadoras por cidade (não requer login). Preferencialmente a cidade deveria ser escolhida através de uma lista.
- R5: Locação de uma bicicleta em uma locadora (requer login do cliente via e-mail + senha). Depois de fazer login, o cliente pode cadastrar uma locação. Para isso, deve escolher uma locadora (escolhendo a partir de uma lista), uma data, e deve ser gravado a locação na base de dados.
- R6: Listagem de todas as locações de um cliente (requer login do cliente via e-mail + senha). Depois de fazer login, o cliente pode visualizar todas as suas locações gravadas.
- R7: O sistema não deve permitir o cadastro de locações de um mesmo cliente ou de uma mesma locadora em um mesmo dia/horário.
- R8: Listagem de todas as locações de uma locadora (requer login da locadora via e-mail + senha). Depois de fazer login, a locadora pode visualizar todas as suas locações gravadas.
- R9: O sistema deve ser internacionalizado em pelo menos dois idiomas: português + outro de sua escolha.
- R10: O sistema deve validar (tamanho, formato, etc) todas as informações (campos nos formulários) cadastradas e/ou editadas.

O sistema deve tratar todos os erros possíveis (cadastros duplicados, problemas técnicos, etc) mostrando uma página de erros amigável ao usuário e registrando o erro no console.

**\*\*Arquitetura: \*\* Modelo-Visão-Controlador**

### **Tecnologias**

- Spring MVC, Spring Data JPA, Spring Security & Thymeleaf (Lado Servidor)
- Javascript & CSS (Lado Cliente)

### **Ambiente de Desenvolvimento**

- A compilação e o *deployment* deve ser obrigatoriamente ser realizado via *maven*.
- Os arquivos fonte do sistema devem estar hospedados obrigatoriamente em um repositório (preferencialmente github).

---

1. CRUD: **Create, Read, Update & Delete.**↵