

DESENVOLVIMENTO DE SOFTWARE PARA A WEB 1

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

Atividade AA-2: Sistema para agendamento de consultas médicas

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

O sistema deve possuir um cadastro de médicos, com os seguintes dados: e-mail, senha, CRM, nome e especialidade.

O sistema deve possuir um cadastro de consultas, com os seguintes dados: paciente, médico e data/hora da consulta. Assume-se que a duração da consulta é de 30 minutos e sempre inicia-se em “hora cheia” (14h 00min etc) ou “hora meia” (14h 30min etc).

O sistema deve atender aos seguintes requisitos:

- R1: CRUD¹ de médicos (requer login de administrador)
- R2: CRUD de pacientes (requer login de administrador)
- R3: Listagem de todos os médicos em uma única página (não requer login)
- R4: Listagem de todos os médicos por especialidade (não requer login). Preferencialmente a especialidade deveria ser escolhida através de uma lista.
- R5: Agendamento de consulta com um médico (requer login do paciente via email + senha). Depois de fazer login, o paciente pode cadastrar uma consulta. Para isso, deve escolher um médico (escolhendo a partir de uma lista), uma data/horário, e deve ser gravado a consulta na base de dados.
- R6: Listagem de todas as consultas de um paciente (requer login do paciente via e-mail + senha). Depois de fazer login, o paciente pode visualizar todas as suas consultas gravadas.
- R7: O sistema não deve permitir o cadastro de consultas de um mesmo médico ou de um mesmo paciente em uma mesma data/horário.
- R8: Listagem de todas as consultas de um médico (requer login do médico via e-mail + senha). Depois de fazer login, o médico pode visualizar todas as suas consultas 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, em arquivo ou na base de dados.

****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: **C**reate, **R**ead, **U**date & **D**elete.↵