

Programação Avançada em Java

Trabalho Prático 2

Introdução

Neste trabalho os alunos deverão desenvolver um jogo “do galo” (ou “da velha” ou “3 em Linha”) web com base em JSFs.

O jogo é disputado entre dois jogadores, cada qual com peças distintas (por exe: “x” e “o”). Joga-se numa posição qualquer de uma posição 3 por 3. Ganha o primeiro jogador a ter três peças das suas peças em linha vertical, horizontal ou diagonal. A Figura 1 apresenta um exemplo.

Data de Entrega: 16 de Fevereiro de 2020

Objetivo: Aquisição de conhecimentos e competências de programação web com JavaServer Faces (JSFs). O aluno deve tirar partido das funcionalidades aprendidas nas aulas, tais como Facelets e Expression Language.

Restrições e Entrega: A utilização de *maven* e *git* é obrigatória. A entrega do trabalho deverá ser feita enviando o link do repositório *git* para os docentes.

Não deve ser utilizado código de terceiros (e.g. componentes ou templates disponíveis na internet) devendo o trabalho ser desenvolvido integralmente pelos elementos do grupo.



Figura 1. Exemplo de tabuleiro do jogo, com vitória das peças “o”, na diagonal.
Os alunos têm liberdade para decidir o aspeto do tabuleiro e das peças.

Parte 1 – Funcionalidades do Jogo “do Galo”

A aplicação a desenvolver deverá permitir a duas pessoas jogarem alternadamente, no mesmo browser, colocando as peças respetivas.

Antes do jogo iniciar, deve ser possível aos dois jogadores definirem os seus nomes, sendo associados a um de dois tipos de peças do jogo (diferentes cores, diferentes imagens). Ao iniciar o jogo, o servidor deve sortear qual o jogador que coloca a primeira peça. A vez de jogar deve alternar automaticamente entre os jogadores.

Em cada momento, a página deve indicar qual é o jogador a fazer a próxima jogada. Para efetuar cada jogada deve bastar clicar com o rato na posição em que pretende que a peça seja colocada. Essa informação deve chegar ao servidor, que irá determinar em que posição a peça irá ficar e gerar um tabuleiro atualizado.

Após cada jogada, a aplicação deve analisar o tabuleiro para descobrir se há um vencedor. Caso haja, deve terminar o jogo, destacando as 3 peças que formam a combinação vencedora.

Toda a lógica do jogo deve ser processada e armazenada no **servidor**.

Parte 2 – Funcionalidades Adicionais

Por cada jogo terminado deve ser armazenado no servidor o vencedor (nome), a duração total (temporal), e o estado final do tabuleiro, podendo assim o utilizador visualizar um **histórico** de jogos e resultados. Esta informação deve ser apresentada de forma a que o utilizador a possa analisar facilmente (por exemplo, numa tabela). Será valorizada a usabilidade da(s) página(s) web. Cada **utilizador**, identificado pelo **nome** inserido no início do jogo, deve poder **consultar o histórico** dos seus jogos.

Deve existir outra página para **consultar estatísticas globais** como: quantas vezes ganha cada uma das cores disponíveis e quantas vezes ocorre cada tipo de vitória: vertical (|), horizontal (–), diagonal para a frente (/) ou diagonal para trás (\). Estas estatísticas deverão ser globais a toda a aplicação, ou seja, deverá ser partilhada por todos os utilizadores do jogo, mesmo que em browsers diferentes.

São valorizadas soluções que guardem esta informação de forma persistente, i.e. entre diferentes execuções do servidor, permitindo guardar e carregar as estatísticas globais e histórico dos jogos de cada jogador.