

Programação para Internet I

Trabalho Final - Individual

Objetivo: avaliar o conhecimento do aluno, em relação aos conhecimentos adquiridos na unidade curricular, a partir da criação de um sistema de autenticação simples.

Critérios de avaliação:

- **Prazo:** solicitar o *pull request* até a data limite. **(1,5)**
- **Completeness:** atendimento dos requisitos solicitados nas instruções. **(1,5)**
- **Código:** organização; indentação; espaços desnecessários. **(0,5)**
- **Arquivos:** nomeados corretamente; nas pastas corretas; de acordo com as instruções. **(0,5)**
- **HTML:** utilização correta dos elementos HTML, inclusive dos elementos semânticos do HTML 5. **(0,5)**
- **Estilos:** aplicação correta dos estilos CSS; quantidade de estilos aplicados. **(1,0)**
- **Banco de dados:** script de criação do banco executa corretamente. **(0,5)**
- **Versionamento:** solicitar o *pull request* apenas com os arquivos que realmente devem ser adicionados, modificados ou excluídos. Fazer apenas um *pull request* para entrega do trabalho. **(0,5)**
- **PHP:**
 - execução sem erros ou avisos. **(0,5)**
 - validação das entradas do usuário. **(0,5)**
 - conexão e utilização do banco de dados. **(0,5)**
 - utilização de sessão. **(0,5)**
 - redirecionamentos entre as páginas. **(0,5)**
 - exibição de mensagens aos usuários. **(0,5)**
 - armazenamento criptografado da senha. **(0,5)**

Instruções:

Cada aluno deve desenvolver um sistema de autenticação simples, de acordo com as instruções abaixo:

- atualize sua cópia local do repositório da turma;
- crie uma pasta no diretório raiz do projeto com seu primeiro nome (todas minúsculas). Alunos com homônimos na turma: incluam a primeira letra do primeiro sobrenome. Ex: daniel f
- mova os arquivos criados por você (dos exercícios anteriores) para esta pasta recém criada.
- crie os seguintes arquivos dentro da pasta recém-criada:
 - **banco-de-dados.sql:** deve conter o *script* para criação de um banco de dados. O banco de dados deve ter o mesmo nome da pasta recém criada e deve conter apenas uma tabela, que armazenará o nome e a senha dos usuários. Crie também neste *script* um usuário do MySQL que será configurado na aplicação para acessar o banco.
 - **index.php:** deve conter campos de formulário para o usuário informar o nome de usuário e senha, e um botão para submeter os dados para a página login.php. Crie também um link para a página novo-usuario.php, na qual será possível cadastrar um novo usuário.
 - **login.php:** deve conter o código PHP para verificar se o usuário está cadastrado na base de dados e se a senha confere. Se ok, criar a sessão, redirecionar o usuário para uma das suas páginas de conteúdo e exibir a mensagem: “Seja bem vindo, xxx”, onde xxx é o nome do usuário que acaba de logar. Senão, retornar para index.php e exibir uma mensagem de erro.
 - **novo-usuario.php:** deve conter campos de formulário para o usuário informar o nome de usuário, a senha, para confirmar a senha e um botão para submeter os dados para a página cadastrar-usuario.php.
 - **cadastro-usuario.php:** deve conter o código PHP para cadastrar um novo usuário na base de dados. Se o usuário informou um nome válido e as senhas forem idênticas, cadastre o usuário no banco, recarregue a mesma página e informe que o novo usuário foi cadastrado com sucesso. Senão, recarregue a mesma página e emita uma mensagem de erro.
 - **logout.php:** deve possuir o código PHP para encerrar a sessão do usuário e redirecionar para index.php.
- nas páginas de conteúdo, inserir um link para efetuar logout que redirecione para logout.php. Pode ser utilizado um menu superior que apareça em todas as páginas de conteúdo para inserção deste link de logout.
- crie os links que forem necessários para permitir a navegação no sistema.
- devido à movimentação dos arquivos, confira se todas as ligações com outros arquivos estão corretas.
- execute os *commits* necessários e o *pull request*.