



AULA PRÁTICA 05

DISCIPLINA: Banco de Dados

CURSO: Ciência da Computação e Engenharia de Software

SEMESTRE: 3º

TURNO: Matutino

DOCENTE RESPONSÁVEL: Moises Silva de Sousa

E-MAIL: moises.sousa@idp.edu.br

CARGA HORÁRIA: 2h

ANO E SEMESTRE DE REFERÊNCIA: 2025/1

OBJETIVO

Essa atividade tem como objetivo reforçar o uso do Git para controle de versão e colaboração em equipe. Durante o processo, os grupos irão praticar a criação de repositórios, o versionamento de código, a revisão por pares e a implementação de melhorias sugeridas por outra equipe.

CONTEXTO

Cenário

Os grupos da aula de Banco de Dados são startups que desenvolveram seus próprios **Sistemas de Gestão de Estacionamento**. Cada uma seguiu abordagens diferentes para implementar funcionalidades como controle de entrada e saída de veículos, pagamento, gerenciamento de vagas e geração de relatórios.

Agora, essas startups foram contratadas por um grande investidor que deseja **padronizar e aprimorar os sistemas**. Para isso, foi organizada uma **auditoria cruzada**, onde cada empresa revisará o código de outra e proporá melhorias.

O objetivo é garantir que os sistemas sejam **eficientes, seguros e escaláveis**, aplicando boas práticas de desenvolvimento e versionamento no **Git**.

Missão das Equipes

Cada equipe tem **duas responsabilidades principais**:

1. **Revisar o código de outra equipe** para identificar **possíveis melhorias** e sugerir ajustes.
2. **Aplicar as sugestões recebidas** e documentar as mudanças no histórico do repositório.

A revisão deve considerar:



- Organização e clareza do código.
- Performance e eficiência.
- Possíveis falhas ou inconsistências.
- Segurança e boas práticas de desenvolvimento.

CRONOLOGIA DA AULA

Passo 1: Escolher um líder para a equipe

Cada grupo deve eleger um líder. O líder será responsável por:

- Criar o repositório no GitHub.
- Adicionar os membros da equipe ao repositório como colaboradores.
- Gerenciar o fluxo de commits e alterações.

Os demais membros do grupo devem trabalhar em conjunto para garantir que todas as implementações estejam organizadas e bem documentadas.

Passo 2: Criar o repositório e subir as implementações existentes

O líder do grupo deve criar um repositório no GitHub e adicionar os arquivos já implementados até o momento.

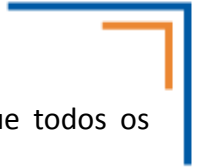
- Todos os arquivos do projeto devem ser adicionados ao repositório.
- O código deve estar organizado e funcionando corretamente.
- O grupo deve garantir que todas as dependências necessárias estejam disponíveis para facilitar o entendimento de outro grupo.

Os comandos necessários para esta etapa estão detalhados no arquivo comandos_git.pdf.

Passo 3: Sorteio dos grupos e adição do líder do outro grupo

O professor realizará um sorteio para definir quais grupos irão revisar o código de quais equipes.

- Após o sorteio, cada grupo receberá um projeto diferente para revisar.
- O líder do grupo deve adicionar o líder do outro grupo como colaborador no repositório para que ele tenha acesso ao código.



- O professor pode auxiliar na organização dessa distribuição, garantindo que todos os grupos participem da troca de códigos.

Passo 4: Propor mudanças no código

Cada grupo analisará o código do outro grupo e identificará possíveis melhorias.

- A equipe deve focar em aspectos como legibilidade, eficiência, boas práticas e organização do código.
- Se houver bugs ou trechos confusos, o grupo deve apontá-los e sugerir correções.
- A equipe deve sugerir melhorias com base em boas práticas de desenvolvimento, como refatoração de código, otimização de consultas (se houver banco de dados) e melhorias na estrutura do projeto.

O professor pode definir um tempo específico para essa análise e garantir que as sugestões sejam bem fundamentadas.

Passo 5: Solicitar mudanças via README.md

As sugestões devem ser registradas no arquivo README.md do repositório.

- O grupo deve listar as sugestões de forma clara e organizada, utilizando a linguagem Markdown para formatação.
- As mudanças propostas podem incluir trechos de código ou explicações sobre como melhorar a implementação.
- O objetivo é fornecer feedback construtivo e facilitar a implementação das mudanças pelo grupo original.

Os comandos necessários para editar e subir o README.md estão disponíveis no arquivo comandos_git.pdf.

Passo 6: Implementar as mudanças e subir as alterações

O grupo original deve analisar as sugestões recebidas e trabalhar nas alterações recomendadas.

- Caso o grupo discorde de alguma sugestão, pode justificar no próprio README.md ou abrir uma discussão com o grupo revisor.

Passo 7: Permitir que o professor veja o histórico de versões



O professor terá acesso ao repositório e poderá acompanhar todas as alterações feitas, verificando o histórico de commits.

- Ele poderá conferir quais mudanças foram sugeridas e quais foram implementadas.
- O professor pode usar o histórico do Git para avaliar a participação de cada grupo e fornecer feedback adicional.

Os comandos necessários para visualizar o histórico de versões estão disponíveis no arquivo **comandos_git.pdf**.

Conclusão e Entrega

Os trabalhos serão submetidos via GitHub e deve conter:

- Acesso do professor ao repositório do grupo;
- Diagrama de Entidade-Relacionamento atualizado;
- Código SQL atualizado.