

Motivação

Em um cenário global cada vez mais marcado pela desinformação, contas falsas e desrespeito às leis nacionais por grandes empresas de tecnologia, como a Meta e a X (antigo Twitter), torna-se urgente repensar a forma como plataformas de mensagens gerenciam a identidade de seus usuários. Recentemente, o Brasil e outros países chegaram a bloquear temporariamente esses serviços por não cumprirem determinações legais. Diante desse contexto, este trabalho propõe o desenvolvimento de um aplicativo de mensagens que utilize o CPF, ou equivalente nacional, como identificador único de cada usuário. Essa abordagem visa garantir que cada pessoa tenha apenas uma conta, reduzindo a criação de perfis falsos e promovendo maior segurança, autenticidade e rastreabilidade nas interações virtuais.

Objetivo

Criar um sistema para gerenciar um aplicativo de mensagens instantâneas que garanta que cada pessoa tenha apenas uma conta de usuário.

Instruções

1. O cliente realizou algumas exigências, sendo elas:
 - 1.1. Permitir o cadastro de usuários e administradores do sistema.
 - 1.2. Permitir o cadastro de grupos de conversa.
 - 1.3. Permitir a inclusão de usuários em grupos.
 - 1.4. Permitir o envio de mensagens entre usuários.
 - 1.5. Permitir o envio de mensagens em grupos.
 - 1.6. Permitir o envio de denúncia nos grupos.
 - 1.7. Cada usuário pode gerenciar seus próprios dados, assim como os dados dos grupos criados por ele, adicionando e/ou excluindo usuários ao grupo.
 - 1.8. Cada usuário pode gerenciar suas mensagens, alterando seu conteúdo e/ou excluindo. No caso de uma alteração, a data deve ser adicionada à mensagem. Já em uma exclusão, a mensagem sumirá apenas da sua caixa de mensagens.
 - 1.9. Um usuário pode enviar uma mesma mensagem instantaneamente para quantos grupos ele desejar, desde que pertença a eles.
 - 1.10. Um usuário pode enviar mensagens para si mesmo.
 - 1.11. Permita que o administrador possa ter uma conta de usuário comum com seus dados.
 - 1.12. Permita que o administrador possa desativar contas e grupos.

O detalhamento das especificações supracitadas não limita o BD a ser criado. Portanto, fique à vontade para efetuar quaisquer modificações que se façam necessárias.

2. O sistema também deve ser capaz de responder as questões a seguir:
 - 2.1. Qual é a média de mensagens que cada usuário recebe?
 - 2.2. Quais usuários tem mais amigos?
 - 2.3. Indicar a quantidade de usuários em cada grupo.
 - 2.4. Listar os usuários com mais de 3 denúncias.

- 2.5. Indicar os motivos mais comuns pelos quais as pessoas são denunciadas.
 - 2.6. Indicar os usuários que estão em mais grupos (ordenar por quantidade de grupos e nome).
 - 2.7. Qual usuário está há mais tempo sem enviar mensagens?
 - 2.8. Excluir automaticamente do sistema mensagens que ambos os usuários removeram de sua caixa de mensagens.
 - 2.9. Indicar quantas mensagens um determinado usuário enviou ou recebeu durante um determinado período.
 3. O trabalho deverá ser feito em Linguagem C (ou linguagens especificadas no item 6.4.), utilizando o Sistema Gerenciador de Banco de Dados (SGBD) MySQL. Para fazer conexão externa (como cliente) com o MySQL utilizando o Ubuntu, é preciso instalar o pacote `libmysqlclient-dev`.
 4. Para compilar o código em Linguagem C é preciso fazer o link com as bibliotecas do MySQL (veja como na documentação do SGBD).
 5. O trabalho é individual.
 6. Pontuação (0 - 7 ptos):
 - 6.1. Apresentação do diagrama DER (ou UML) associado ao banco de dados criado.
 - 6.2. Identação, comentários e domínio do código fonte.
 - 6.3. Apresentação do trabalho via vídeo assíncrono¹ de até 25 minutos, mostrando partes principais do código fonte e, além disso, o programa sendo compilado e executado com os devidos testes (conforme instruções do itens 1 e 2). Durante a apresentação deverá ser gravado a tela do computador, a voz e imagem (webcam) do(a) aluno(a). Caso a gravação do vídeo não seja viável, a apresentação será oral (dia a definir) utilizando uma máquina do laboratório² ou notebook próprio.
 - 6.4. Desenvolver interface gráfica, utilizando outra linguagem de programação (C++ / C# / Java / Python). Essa tarefa é extra, ou seja, em caso de execução desse item, o aluno poderá atingir até 7 pontos no total. Caso contrário, a pontuação máxima passa a ser 6 pontos.
 7. O trabalho deverá ser entregue até às 23h59m do dia 6 de julho de 2025 (domingo), e apresentado conforme especificado no item 6.3.
 8. **Não será permitida a entrega com atraso.**
 9. É imprescindível que os arquivos a serem entregues tenham nome e sobrenome do(a) aluno(a). Utilize a seguinte formatação:
 - **nome-sobrenome-src.zip**: código fonte.
 - **nome-sobrenome-diagrama.png**: diagrama DER (ou UML).
 - **nome-sobrenome-apresentacao.mp4**: arquivo MP4 (ou link compartilhado) contendo a apresentação do trabalho. Este arquivo também deve ser inserido na pasta da equipe do MS Teams, para visualização de todos.
-

¹Sugerido utilizar o MS Teams para este propósito. Basta agendar uma reunião, compartilhar a tela, habilitar vídeo e microfone, e iniciar a gravação. Ao final da reunião a gravação será concluída e estará pronta para compartilhamento.

²**Caso escolha esta opção teste antes da apresentação**, para garantir que seu programa esteja funcionando como esperado. Não será permitido instalar programas e/ou pacotes específicos no dia da apresentação.