



Método mandarMensagem() é utilizado para gerenciar problemas de usuários, oferecendo suporte por bate-papo

Usuario
Classe Usuário possui cpf, nome, telefone, nível de fluência, lista de conversas, lista de conquistas, uma lista de cursos e histórico de lições.
O usuário pode dizer o cpf, nome, telefone, senha e seu nível de fluência.
O Usuário consegue ver todos os seus atributos, mas não consegue modificar as listas diretamente.
Usuário pode realizar exercício ou mandar mensagens para outros usuários e contatar suporte(administradores), além de poder ver a sua conversa, que contem essas mensagens.
Os métodos que adicionam ou removem um curso/conquista servem apenas para o administrador usa-los para solucionar problemas

Conversa
Classe Conversa possui id, lista de participantes e lista de mensagens.
A conversa pode ser inicializada com a lista de participantes (2 máximo).
A conversa tem um método que retorna a lista de mensagens que ela possui e um método que configura mensagem por meio de seu objeto, acessando seus métodos

Mensagem
Classe Mensagem possui id, conteúdo, data de postagem, data de edição, remetente e destinatário.
O objeto mensagem pode ser criado com um destinatário específico.
A mensagem pode ser enviada com um conteúdo, pode ser editada ou excluída.

Lição
Classe Licao possui id, curso_id(FK para curso a qual pertence), titulo, descricao, dificuldade, pontuação que inicia zerada e lista de exercícios.

Licao pode ser inicializada com titulo, descricao, dificuldade e uma lista de exercicios

Possui os getters para mostrar os dados sobre a lição

Setter de pontuação é utilizado para definir a pontuação da lição, utilizada depois de marcar a opção correta. (Exercicio possui id da licao, podendo puxar seu método)

Administrador
Classe administrador possui cpf, nome, email e senha.
O administrador pode ser inicializado com os 4 atributos a serem definidos.
O administrador pode gerenciar cursos ou usuários, os 2 métodos retornam o objeto referente a classe, possibilitando utilizar métodos da classe gerenciada, para ler, criar, atualizar ou remover. (Mais detalhes são ditos em cada classe afetada)
O administrador também pode gerenciar problemas técnicos, conversando com o usuário e dando suporte por bate-papo.

Curso
Classe curso possui id, cpf a quem detém o curso (usuario_cpf [FK]), id da conquista referente ao curso (conquista_id [FK]), nome, nível do curso, requisitos a serem cumpridos para terminar o curso, uma lista de lições e um conteúdo explicativo do curso.

O inicalizador do curso aceita nome, requisitos e conteúdo para serem criados, além do nome da conquista e sua descrição a ser definida.

Os getters permitem que os dados sejam visualizados, os setters(isso inclui adicionar e remover requisitos) permitem que o administrador gerencie o curso.

Conquista
Classe Conquista possui id, nome, descrição, classificação e usuario_cpf(FK para usuário que desbloqueou a conquista). Uma conquista é criada junto com curso, quando desbloqueada, um usuário é associado a ela.

A conquista é criada com nome e descrição.
Os getters permitem visualizar os dados da conquista
o setter para classificação permite que a cada curso associado a conquista promova a classificação do usuário, conforme cada curso é concluído.

Exercicio
Classe Exercicio possui id, licao_id(FK da licao a qual pertence), tipo de exercicio, descricao, vetor de alternativas, resposta do usuario e resposta correta do exercicio.

O exercicio pode ser criado com tipo de exercicio, descricao, vetor de alternativas e resposta correta do exercicio.

os getters permitem que os dados do exercicio seja visualizado
metodo marcarResposta permite que seja marcada a resposta do exercicio
metodo verResposta permite ver a resposta correta