



Residência em Software

Residência em Tecnologia da Informação e Comunicação

Exercitando Classes e Objetos em Java

Professor:

Alvaro Degas Coelho



INSTITUIÇÃO EXECUTORA



COORDENADORA



APOIO





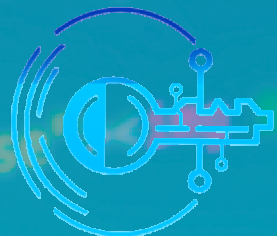
Residência
em Software

Classe
Atributo1
Atributo2
Metodo1()
Metodo2()



Residência
em Software

Usuario
Nome
Email
DataNascimento
ListaPostagens
Getters e Setters
Usuario(Nome, Email)
NovaPostagem(postagem)
ListaPostagens()



Residência
em Software

Rede Social

InstaDegas



INSTITUIÇÃO EXECUTORA



COORDENADORA



APOIO



A classe Usuario

- Um usuario deve possuir os atributos
 - Nome
 - Email
 - Senha
 - ListaPostagens
- E os métodos
 - Getters e setters mais
 - novaPostagem(postagem)
 - listaPostagens()

Usuario
Nome Email Senha ListaPostagens
Getters e Setters Usuario(Nome, Email) NovaPostagem(postagem) listaPostagens()

Implemente a classe Usuario

Usuario
Nome Email Senha ListaPostagens
Getters e Setters Usuario(Nome, Email) NovaPostagem(postagem) ListaPostagens()

Amizades

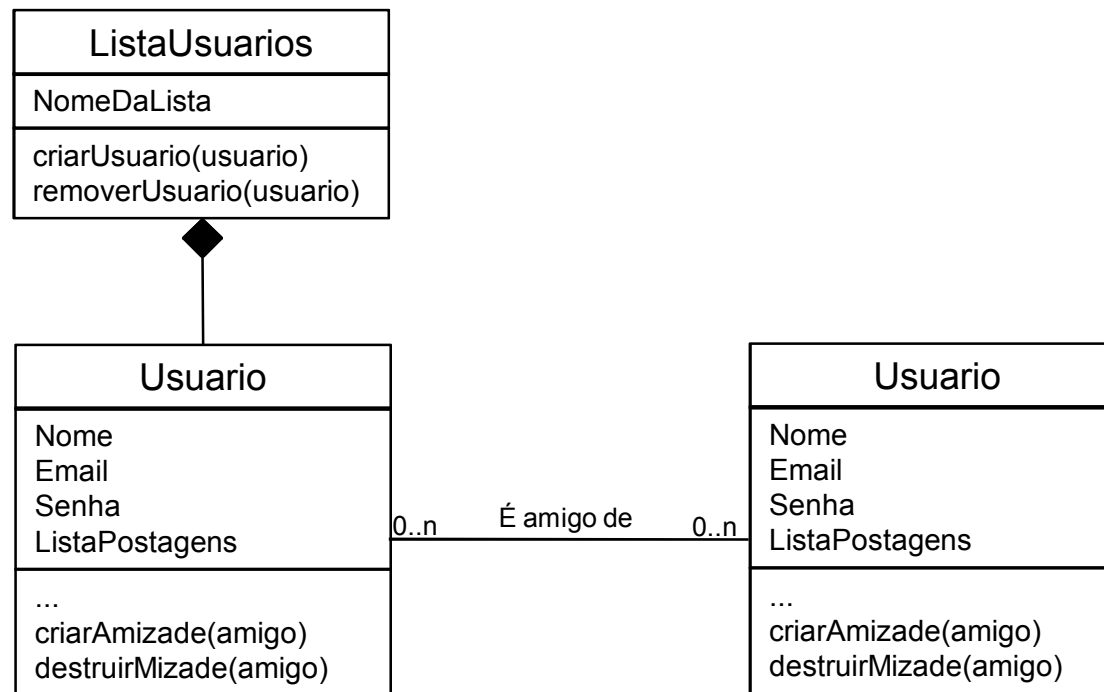
- Usuários são amigos de Usuários
 - Uma amizade é uma relação N-N entre usuários
- Como implementar esta relação?



Métodos novos

- **criarAmizade(Usuario amigo)**
 - Cria uma relação de amizade entre este usuário e seu novo amigo.
 - Observações
 - Precisa criar a amizade inversa entre o amigo e este usuário
 - Use “this” pra identificar este objeto (Google!!!)
 - Não pode criar uma amizade que já exista.
 - Use exceptions
- **destruirAmizade(Usuario amigo)**
 - Exclui uma amizade entre este usuário e seu amigo
 - Se a amizade não existir? Exception!

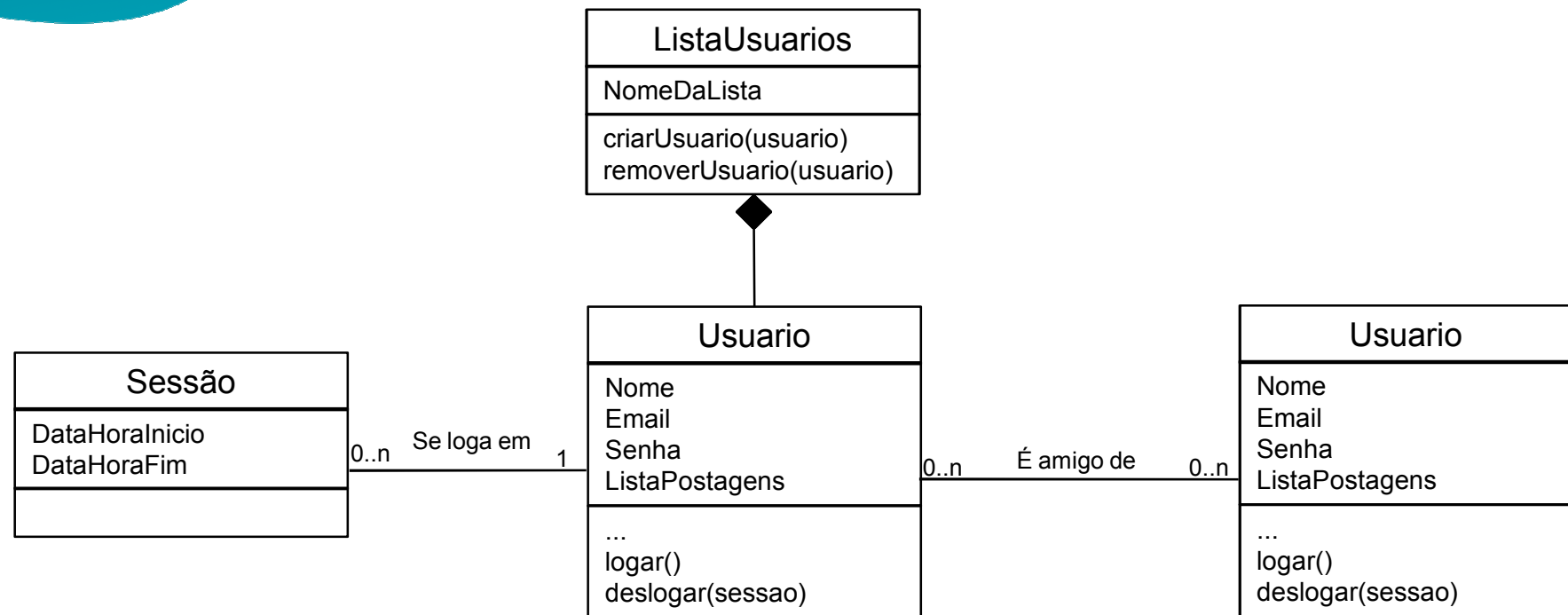
Coleção de Usuários



A coleção de usuários

- São todos os usuários desta rede social
- Implementa uma composição com usuários
 - Como faz?
- Implementa o método `criarUsuario(usuario)` que acrescenta usuario na sua lista
- Implementa o método `removerUsuario(usuario)` que remove o usuário da lista
- Com as devidas exceptions

Sessões



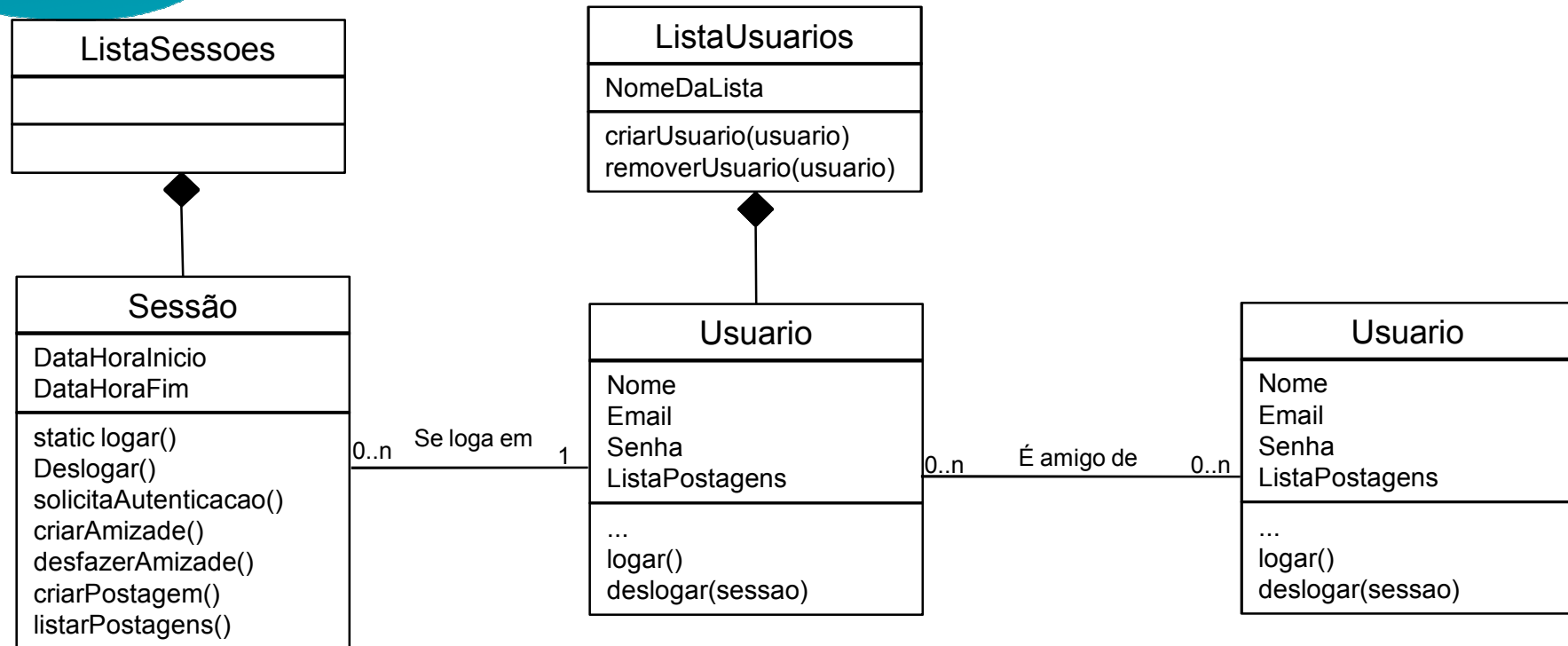
Sessões

- Quando um usuário começa a utilizar a rede social, é criada uma sessão()
- Há uma relação de 1 pra n entre usuário e sessões
 - como implementa? Google!
- O usuário possui um método estático `logar(usuario)` que cria uma nova sessão
 - A data e a hora de início da sessão é buscada do sistema. Como? Google!
 - O usuário dono da sessão é passado como parâmetro
- O usuário possui um método `deslogar(sessao)` que encerra uma sessão
 - A data e a hora de fim da sessão é buscada do sistema. Como? Google!

Mais sobre a classe Sessão

- Esta classe possui um método estático autenticar(lista de usuários)
 - Solicita o nome de um usuário e sua senha. Se o usuário e a senha coincidirem, instancia uma nova sessão para aquele usuário (use exceptions) – método login()
 - Esta nova sessão fica disponível na lista de sessões
- Esta classe possui um método criarAmizade(nomeDoAmigo)
 - De posse do usuário que está na sessão executa seu método criarAmizade()
 - Solicita o nome do amigo e passa como parâmetro
- Esta classe possui um método desfazerAmizade(nomeDoExAmigo)
 - Análoga
 - Solicita o nome do ex-amigo e passa como parâmetro
- Esta classe possui um método incluirPostagem()
 - Solicita uma postagem e a inclui na lista de postagens do usuário da sessão
- Esta classe possui um método listarPostagens()
 - Lista todas as postagens do usuário da sessão

Sessões



Crie a interface do usuário

- Uma tela inicial
 - Menu com
 - 1. NovoUsuario
 - Solicita os dados de um novo usuário
 - Insere na lista (use exceptions)
 - 2. Remove Usuario
 - Solicita usuário e, depois a senha. Confirma que deseja incluir. Exclui todas as amizades daquele usuário. Exclui todas as sessões daquele usuário. Exclui o usuário da lista
 - 3. Logar
 - Solicita usuário e, depois a senha. Cria uma sessão e a insere na lista. Vai pro menu de sessão
 - 0. Sair
 - Sai do programa

Menu de Sessão

- Um menu com opções
 - 1. Listar postagens
 - Listar todas as postagens do usuário
 - 2. CriarPostagem
 - Executa o método incluirPostagem() da sessão
 - 3. CriarAmizade (string nomeUsuario)
 - Localiza o usuário (novo amigo) na lista. Executa criarAmizade(amigo)
 - Use exceptions
 - 4. DesfazerAmizade (string nomeUsuario)
 - Localiza o usuário (ex amigo) na lista. Executa destruirAmizade(amigo)
 - Use exceptions
 - 0. Menu anterior
 - Executa deslogar()
 - Volta para o menu principal