model Mensagem Usuario + emailDestinatario:type = + nome:type = defaultValue defaultValue + email:type + emailRemetente:type + senha:type + textoMensagem:type + idMensagem:type

repository

UsuarioRepository

+ fun insert(usuario: Usuario, callback: (Boolean) -> Unit, callbackMessage: (String) -> Unit)

- + fun login(usuario: Usuario, callback: (Boolean) -> Unit)
- + fun verificaSessao(callback: (Boolean) -> Unit)
- fun logoff()
- +fun findAll(callback: (List<Usuario>) -> Unit)

MensagemRepository

- + fun insert(mensagem: Mensagem, callback: (Boolean) -> Unit)
- + fun findMensagem(emailDestinatario: String, callback: (List<Mensagem>) -> Unit)

CadastroViewModel: ViewModel()

inserted: LiveData<Boolean> insertedMessage: LiveData<String>

-fun insert(nome: String, email: String, senha: String)

ChatViewModel(emailDestinatario: String): ViewModel()

inserted: LiveData<Boolean>

mensagens: LiveData<List<Mensagem>>

-fun insert(emailDestinatario: String, textoMensagem: String)

-fun findMensagem(emailDestinatario: String)

MainViewModel:ViewModel()

LoginViewModel : ViewModel()

logged: LiveData<Boolean>

-fun login(email: String, senha: String) -fun verificaLogin()

val users: LiveData<List<Usuario>>

fun logout() private fun loadUsers()

adapter

ContatosAdapter(val listener: ContatoClickListener?): RecyclerView.Adapter<ContatosAdapter.ViewHolder>()

+override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): ViewHolder -override fun onBindViewHolder(holder: ContatosAdapter.ViewHolder, position: Int) +override fun getItemCount(): Int

-fun submitDataset(dataset: List<Usuario>)

MensagemAdapter(val emailDestinatario:String) :RecyclerView.Adapter<RecyclerView.ViewHolder>()

+override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): RecyclerView.ViewHolder

+override fun getItemCount(): Int

-override fun onBindViewHolder(holder: RecyclerView.ViewHolder, position: Int) +override fun getItemViewType(position: Int): Int

-fun submitDataset(dataset: List<Mensagem>)

«interface»

listeners

fun clickContatoItem(position: Int)

ContatoClickListener

MainActivity: AppCompatActivity(),ContatoClickListener

-override fun onCreate(savedInstanceState: Bundle?)

+override fun onCreateOptionsMenu(menu: Menu?): Boolean

+override fun onOptionsItemSelected(item: MenuItem):Boolean

-override fun clickContatoItem(position: Int)

-private fun setupObservers()

-private fun setupRecyclerView()

-private fun configuraToolbar()

-private fun pesquisar() -private fun sair()

activity

CadastroActivity: AppCompatActivity()

-override fun onCreate(savedInstanceState: Bundle?) -private fun setupListeners()

-private fun setupObservers()

-private fun validaCadastro()

-private fun cadastrarUsuario(strNome: String, strEmali: String, strSenha: String)

ChatActivity: AppCompatActivity()

-override fun onCreate(savedInstanceState: Bundle?)

+override fun onOptionsItemSelected(item: MenuItem): Boolean

-private fun configuraReciclerView()

-private fun setupListeners()

-private fun enviarMensagem() -private fun openBundle()

-private fun configuraToolbar()

LoginActivity : AppCompatActivity()

override fun onCreate(savedInstanceState: Bundle?)

private fun setupObservers()

private fun abrirTelaPrincipal()

private fun setupListeners() private fun validarLogin()

private fun logarUsuario(strEmail: String, strSenha: String)

private fun abrirCadastro()

factory

ChatViewModelFactory(private val emailDestinatario: String): ViewModelProvider.Factory

override fun <T : ViewModel> create(modelClass: Class<T>): T