

# Relatório - Prova 1

Aluno: Gabriel Cerqueira Araujo de Carvalho

Disciplina: ASI - Análise dos Sistemas de Informação

## Relacionamento de funcionalidades

### 1. INTRODUÇÃO

Este relatório tem como objetivo relacionar as principais funcionalidades da aplicação Rede Social W com os diferentes modelos e diagramas desenvolvidos ao longo da disciplina. São considerados, para cada funcionalidade, os requisitos funcionais correspondentes, os casos de uso, os diagramas de sequência, o diagrama de classes, o modelo entidade-relacionamento (MER) e o protótipo de alto nível. Dessa forma, busca-se verificar a coerência entre as diversas visões do sistema, desde a especificação dos requisitos até a modelagem de dados, da lógica de negócio e da interface com o usuário.

### 2. REQUISITOS FUNCIONAIS DA REDE SOCIAL W

Requisitos funcionais são descrições das funcionalidades que o sistema deve obrigatoriamente oferecer ao usuário, expressas em termos de serviços, ações ou comportamentos observáveis. No contexto da Rede Social W, foram definidos os seguintes requisitos funcionais principais:

ID	Descrição do Requisito Funcional
RF01	Realizar login: o sistema deve permitir que o usuário informe e-mail e senha para autenticação.
RF02	Registrar usuário: o sistema deve permitir o cadastro de um novo usuário com nome, e-mail e senha.

ID	Descrição do Requisito Funcional
RF03	Postar mensagem: o sistema deve permitir que um usuário autenticado publique uma nova mensagem de texto.
RF04	Curtir mensagem: o sistema deve permitir que um usuário autenticado curta mensagens existentes na timeline.
RF05	Comentar mensagem: o sistema deve permitir que um usuário autenticado registre comentários em uma mensagem.
RF06	Exibir timeline: o sistema deve apresentar uma lista de mensagens publicadas, com suas informações básicas de autor, conteúdo, curtidas e comentários.

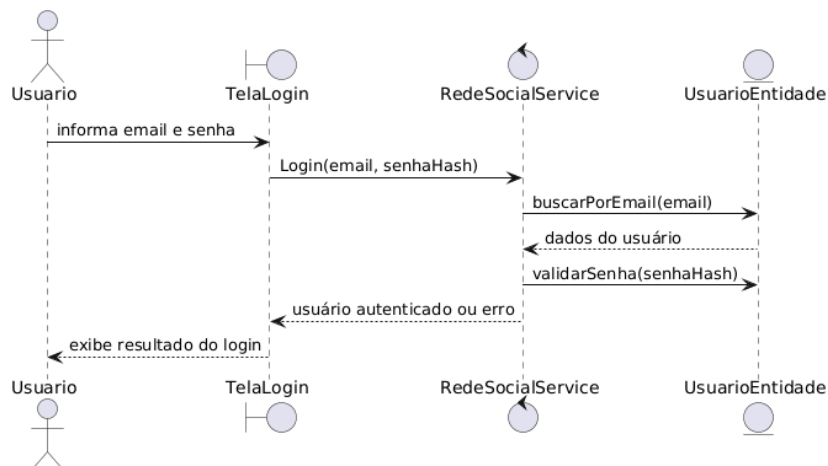
Esses requisitos servem de base para a construção dos casos de uso e dos demais modelos elaborados para o sistema.

### 3. RELACIONAMENTO ENTRE FUNCIONALIDADES E DIAGRAMAS

#### 3.1. LOGIN (RF01)

##### 3.1.1. MODELOS COMPORTAMENTAIS E ESTRUTURAIS

A funcionalidade de login está associada ao requisito funcional RF01 - Realizar login e é descrita pelo Caso de Uso UC01 – Login.



No Diagrama de Sequência de Login são representadas as interações entre o ator Usuario, a interface TelaLogin e a classe de controle RedeSocialService, que expõe o método Login(email, senhaHash). A lógica de autenticação é suportada pela classe Usuario, representada no diagrama de classes, que contém os atributos de identificação e o método Autenticar, responsável por VALIDAR A SENHA DO USUÁRIO.

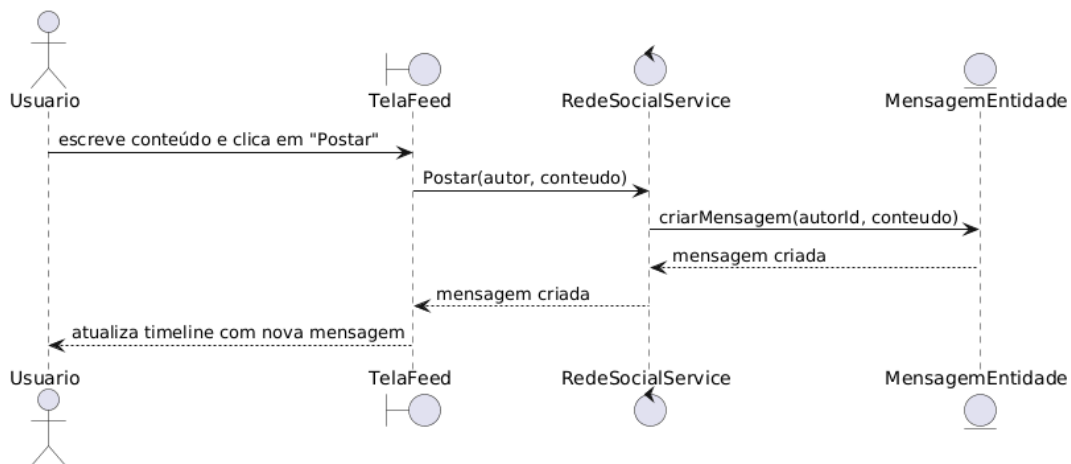
### 3.1.2. PERSISTÊNCIA E INTERFACE

No modelo entidade-relacionamento (MER), essa funcionalidade está relacionada à entidade USUARIO, que armazena os dados necessários para a autenticação, como id\_usuario, nome, email e senha\_hash. No protótipo de alto nível, o login é representado pela Tela de Login, na qual o usuário informa e-mail e senha e recebe o retorno de sucesso ou erro, em conformidade com o fluxo especificado no diagrama de sequência.

## 3.2. POSTAR MENSAGEM (RF03)

### 3.2.1. MODELOS COMPORTAMENTAIS E ESTRUTURAIS

A funcionalidade de postar mensagem está ligada ao requisito funcional RF03 - Postar mensagem e ao Caso de Uso UC03 - Postar mensagem.



No Diagrama de Sequência de Postar Mensagem, o ator Usuario interage com a TelaFeed, informando o conteúdo da mensagem e acionando a operação de postagem. A TelaFeed encaminha a requisição à classe RedeSocialService, que executa o método Postar(autor, conteúdo) e cria uma nova instância da classe Mensagem.

No diagrama de classes, a classe Mensagem reúne os atributos Id, Conteúdo e CriadoEm, armazenando as informações da publicação, enquanto a classe RedeSocialService coordena a criação das mensagens.

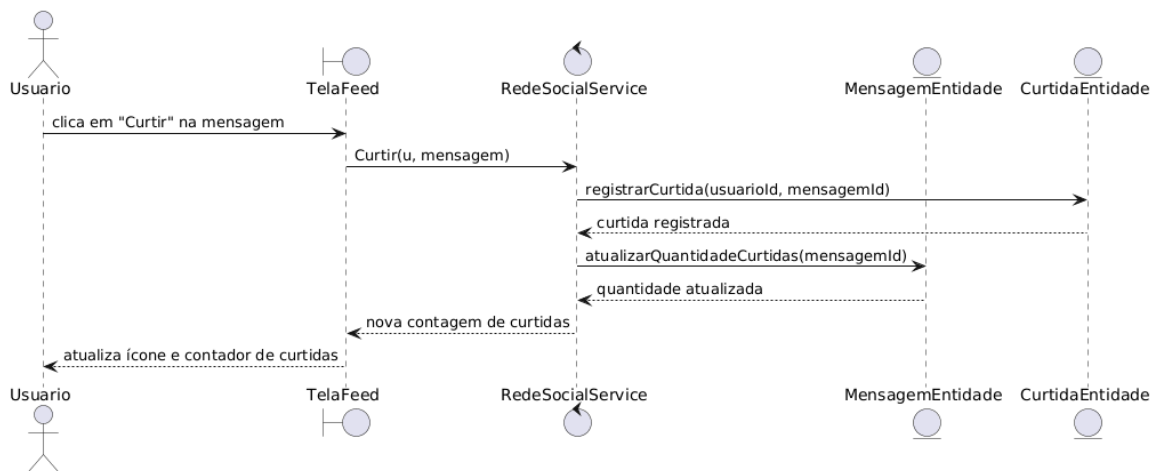
### 3.2.2. PERSISTÊNCIA E INTERFACE

No MER, essa funcionalidade corresponde à entidade MENSAGEM, que possui uma chave estrangeira para USUARIO, indicando qual usuário publicou cada mensagem, além dos campos conteúdo e criado\_em. No protótipo de alto nível, a operação de postagem é realizada na Tela de Feed, onde o usuário visualiza um campo de texto para digitar o conteúdo e um botão para confirmar a publicação, refletindo o comportamento descrito no diagrama de sequência.

### 3.3. CURTIR MENSAGEM (RF04)

#### 3.3.1. MODELOS COMPORTAMENTAIS E ESTRUTURAIS

A funcionalidade de curtir mensagem está relacionada ao requisito funcional RF04 - Curtir mensagem e ao Caso de Uso UC04 - Curtir mensagem.



O Diagrama de Sequência de Curtir Mensagem mostra o fluxo em que o ator Usuario, a partir da TelaFeed, seleciona a opção de curtir uma mensagem específica. A interface envia a solicitação para a classe RedeSocialService por meio do método Curtir(u, mensagem). Em

seguida, o serviço registra a curtida e atualiza a quantidade de curtidas associadas à mensagem antes de retornar a nova contagem para a interface.

No diagrama de classes, essa funcionalidade utiliza a classe Mensagem, que contém operações para gerenciar curtidas, em conjunto com a classe de serviço RedeSocialService.

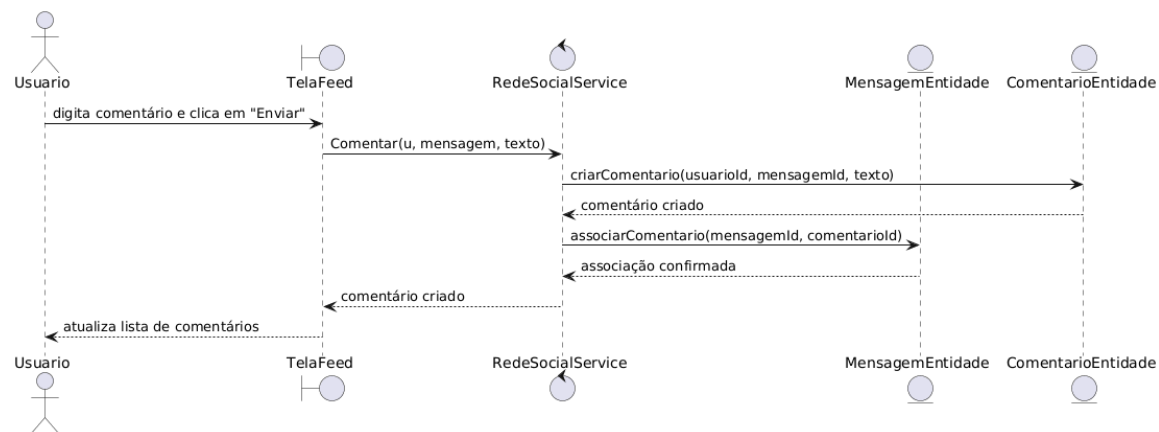
### 3.3.2. PERSISTÊNCIA E INTERFACE

No modelo entidade-relacionamento, o registro da curtida é representado pela entidade CURTIDA, que referencia USUARIO e MENSAGEM, indicando qual usuário curtiu qual mensagem, além do campo criado\_em para registrar a data e hora da ação. No protótipo de alto nível, a funcionalidade é visualizada pela presença de um ícone de curtida e de um contador na Tela de Feed, que é atualizado após a execução do fluxo modelado no diagrama de sequência.

## 3.4. COMENTAR MENSAGEM (RF05)

### 3.4.1. MODELOS COMPORTAMENTAIS E ESTRUTURAIS

A funcionalidade de comentar mensagem está associada ao requisito funcional RF05 - Comentar mensagem e ao Caso de Uso UC05 - Comentar mensagem.



No Diagrama de Sequência de Comentar Mensagem, o ator Usuario digita o texto do comentário na TelaFeed e aciona o envio. A interface encaminha a requisição para a classe RedeSocialService por meio do método Comentar(u, mensagem, texto), que cria um novo comentário e o associa à mensagem correspondente.

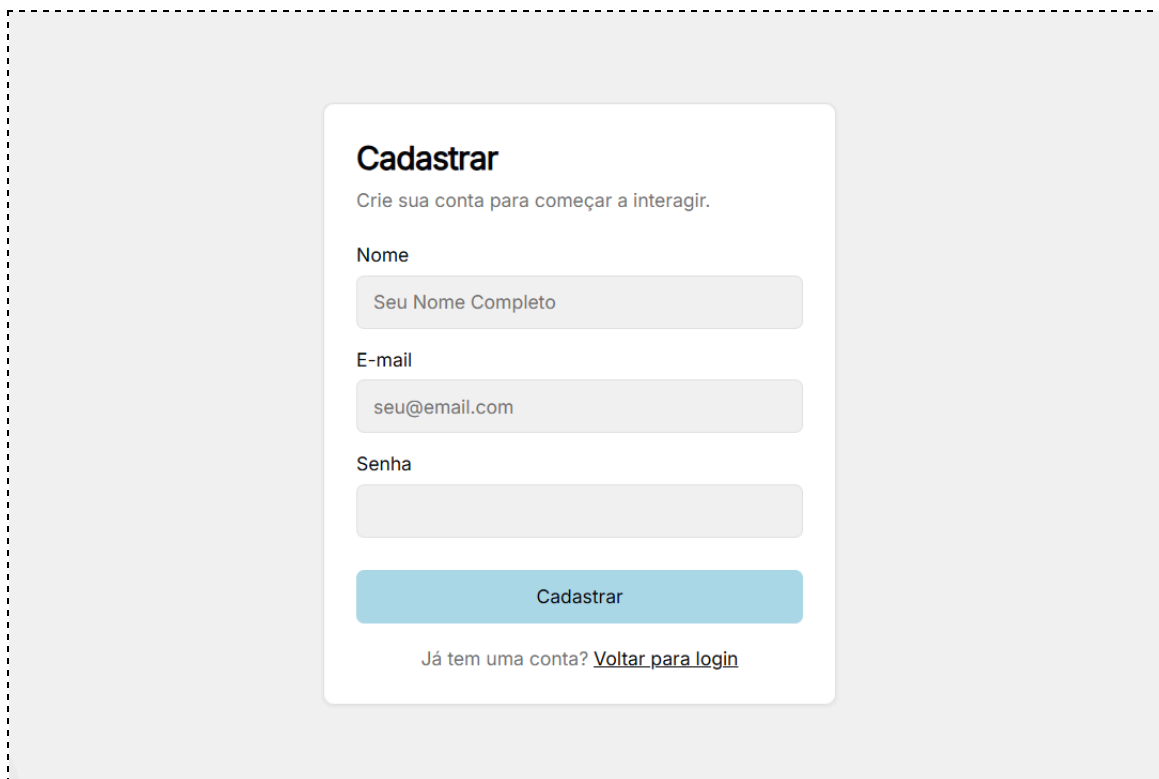
No diagrama de classes, essa funcionalidade envolve as classes Mensagem e Comentario, sendo que a primeira agrega uma coleção de comentários e oferece operações para adicioná-los, enquanto a segunda representa cada comentário individual, com seus atributos Id, Texto e CriadoEm.

### 3.4.2. PERSISTÊNCIA E INTERFACE

No MER, o comportamento é refletido pelas entidades COMENTARIO, MENSAGEM e USUARIO: cada registro de COMENTARIO referencia o usuário autor (id\_usuario) e a mensagem à qual ele pertence (id\_mensagem), além do texto e da data de criação. No protótipo de alto nível, a funcionalidade é representada pela área de comentários da Tela de Feed, onde são exibidos os comentários existentes e onde o usuário pode incluir novos comentários, em conformidade com o fluxo descrito no diagrama de sequência.

## 3.5. REGISTRO DE USUÁRIO (RF02) E EXIBIÇÃO DA TIMELINE (RF06)

### 3.5.1. PROTÓTIPO DA TELA DE CADASTRO



O protótipo da tela de cadastro é apresentado dentro de um retângulo tracejado. No centro, há um formulário branco com o título "Cadastrar" em negrito. Abaixo do título, há uma instrução: "Crie sua conta para começar a interagir." O formulário contém três campos de entrada: "Nome" com o placeholder "Seu Nome Completo", "E-mail" com o placeholder "seu@email.com", e "Senha" com um campo vazio. Abaixo dos campos, há um botão azul com o texto "Cadastrar". Na base do formulário, há um link: "Já tem uma conta? [Voltar para login](#)".

A Tela de Cadastro de usuário contém os campos "Nome", "E-mail" e "Senha", além do botão "Cadastrar" e do link "Voltar para login". Essa tela implementa o requisito funcional RF02 - Registrar usuário, permitindo a criação de uma nova conta na aplicação, em conformidade com o caso de uso de registro de usuário e com a entidade USUARIO do modelo entidade-relacionamento.

### 3.5.2. PROTÓTIPO DA TELA DE FEED (TIMELINE)


Rede Social W

Olá, Gabriel Cerqueira

O que você está pensando?

Postar

Larissa Santos

Primeira mensagem na Rede Social W! Muito animada para começar.  


Curtir (15)

Comentar

Diego Silva

Bem-vinda, Lari!

Marcelo Honda


Que legal ter você aqui!

Escreva um comentário...

Enviar

Postar

Larissa Santos

Primeira mensagem na Rede Social W! Muito animada para começar.  


Curtir (15)

Comentar

Diego Silva

Bem-vinda, Lari!

Marcelo Honda

Que legal ter você aqui!

Escreva um comentário...

Enviar

Diego Silva

Alguém tem uma boa recomendação de livro de ficção científica?

Curtir (8)

Comentar

Escreva um comentário...

Enviar

A Tela de Feed (timeline) apresenta a área principal de interação do usuário com a Rede Social W. Esta tela é fundamental para o requisito funcional RF06 - Exibir timeline, apresentando cartões de mensagens com informações do autor, conteúdo, curtidas e comentários.

Ela também integra as funcionalidades de:

- **Postar Mensagem (RF03):** Área para postagem com o campo "O que você está pensando?" e o botão "Postar".
- **Curtir Mensagem (RF04):** Botão "Curtir" e contador de curtidas.
- **Comentar Mensagem (RF05):** Botão "Comentar" e área para inserção e visualização de comentários.

A interface da Tela de Feed está coerente com os diagramas de sequência de Postar Mensagem, Curtir Mensagem e Comentar Mensagem, e reflete as estruturas das classes Mensagem e Comentario e das entidades MENSAGEM, COMENTARIO e CURTIDA.

#### 4. CONCLUSÃO

A partir da análise realizada, observa-se que as funcionalidades de login, registro, postagem, curtida e comentário da Rede Social W estão representadas de forma coerente nos diferentes modelos desenvolvidos. Os requisitos funcionais servem de base para a definição dos casos de uso, que, por sua vez, são detalhados nos diagramas de sequência. O diagrama de classes e o modelo entidade-relacionamento complementam essa visão, descrevendo as estruturas de dados e as relações entre usuários, mensagens, curtidas e comentários. Por fim, o protótipo de alto nível traduz essas funcionalidades para a perspectiva da interface com o usuário. Essa integração entre requisitos, modelos e protótipo contribui para uma compreensão mais completa e consistente do sistema, facilitando tanto a comunicação entre os envolvidos no projeto quanto a futura implementação e evolução da aplicação.