

# Projeto



---

APLICAÇÃO WINDOWS FORMS – C#



# índice

<b>9</b>	REQUISITOS NÃO FUNCIONAIS USUARIO
<b>10</b>	REQUISITOS NÃO FUNCIONAIS SISTEMA
<b>11</b>	REQUISITOS FUNCIONAIS SISTEMA
<b>12</b>	REQUISITOS FUNCIONAIS SISTEMA
<b>13</b>	REQUISITOS FUNCIONAIS SISTEMA
<b>14</b>	DIAGRAMA DE CLASSES

<b>20</b>	DIAGRAMA DE SEQUÊNCIA DO RF03
<b>21</b>	DESCRIÇÃO DOS REQUISITOS DO SISTEMA RF04
<b>22</b>	DIAGRAMA DE SEQUÊNCIA DO RF04
<b>23</b>	DESCRIÇÃO DOS REQUISITOS DO SISTEMA RF05
<b>24</b>	DIAGRAMA DE SEQUÊNCIA DO RF05

<b>29</b>	FINALIZAÇÃO
-----------	-------------

<b>03</b>	QUEM SOMOS
<b>04</b>	OBJETIVOS
<b>05</b>	TECNOLOGIA UTILIZADA
<b>06</b>	DOCUMENTO REQUISITOS
<b>07</b>	REQUISITOS FUNCIONAIS USUARIO
<b>08</b>	REQUISITOS FUNCIONAIS USUARIO

<b>15</b>	DESCRIÇÃO DOS REQUISITOS DO SISTEMA RF01
<b>16</b>	DIAGRAMA DE SEQUÊNCIA DO RF01
<b>17</b>	DESCRIÇÃO DOS REQUISITOS DO SISTEMA RF02
<b>18</b>	DIAGRAMA DE SEQUÊNCIA DO RF02
<b>19</b>	DESCRIÇÃO DOS REQUISITOS DO SISTEMA RF03

<b>25</b>	DESCRIÇÃO DOS REQUISITOS DO SISTEMA RF06
<b>26</b>	DIAGRAMA DE SEQUÊNCIA DO RF06
<b>27</b>	MODELAGEM DO BANCO DE DADOS
<b>28</b>	MODELAGEM DO BANCO DE DADOS



# Quem Somos

SOMOS UM GRUPO DE ESTUDANTES DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS DO INSTITUTO FEDERAL DE CIÊNCIAS E TECNOLOGIA DE BARRETOS, COMPROMETIDOS EM APLICAR NOSSOS CONHECIMENTOS E HABILIDADES PARA DESENVOLVER SOLUÇÕES INOVADORAS E FUNCIONAIS.

NOSSO OBJETIVO É APRESENTAR UMA DOCUMENTAÇÃO DETALHADA DE UM PROJETO VOLTADO PARA A CRIAÇÃO DE UMA APLICAÇÃO DE CHAT UTILIZANDO WINDOWS FORMS E C#

AO EMBARCAR NESTE PROJETO, ESTAMOS COMPROMETIDOS EM CRIAR UMA APLICAÇÃO DE CHAT QUE SEJA TRANSPARENTE, FÁCIL DE USAR, INTUITIVA E COM UM DESIGN ATRATIVO. DESEJAMOS PROPORCIONAR AOS USUÁRIOS UMA EXPERIÊNCIA DE COMUNICAÇÃO DINÂMICA E ENVOLVENTE, CAPAZ DE FACILITAR A TROCA DE MENSAGENS EM TEMPO REAL.

---

## PROGRAMADORES

RENAN IAGO CHAVES

AMERSON SAMUEL OLIVEIRA MACIEL

JONATHAN RODRIGUES BERNARDES

---

## ANALISTAS

ELIEZER ALMEIDA DE ARAUJO

PEDRO HENRIQUE SANTOS MORAIS

# Objetivos

ESTES SÃO OS PRINCIPAIS  
OBJETIVOS DESTE PROJETO

## OBJETIVO 1

O PROJETO EM QUESTÃO TEM COMO OBJETIVO DESENVOLVER UMA APLICAÇÃO DE CHAT UTILIZANDO O WINDOWS FORMS DO VISUAL STUDIO E A LINGUAGEM DE PROGRAMAÇÃO C#. A PROPOSTA PRINCIPAL É CRIAR UMA PLATAFORMA DE COMUNICAÇÃO QUE SEJA TRANSPARENTE, FÁCIL DE USAR, INTUITIVA E COM UM DESIGN ATRATIVO.

## OBJETIVO 2

OFERECER UMA INTERFACE AMIGÁVEL E INTUITIVA, PERMITINDO QUE OS USUÁRIOS POSSAM INTERAGIR DE FORMA SIMPLES E DIRETA, SEM COMPLICAÇÕES DESNECESSÁRIAS. ATRAVÉS DA UTILIZAÇÃO DO WINDOWS FORMS, SERÁ POSSÍVEL CRIAR UMA APLICAÇÃO COM ELEMENTOS VISUAIS ATRAENTES E DE FÁCIL COMPREENSÃO, PROPORCIONANDO UMA EXPERIÊNCIA AGRADÁVEL AOS USUÁRIOS.

## OBJETIVO 3

ALÉM DISSO, A TRANSPARÊNCIA É OUTRO ASPECTO FUNDAMENTAL A SER CONSIDERADO NESSE PROJETO. BUSCA-SE DESENVOLVER UMA APLICAÇÃO CONFIÁVEL E SEGURA, ONDE A PRIVACIDADE DOS USUÁRIOS SEJA PRESERVADA. SERÃO ADOTADAS MEDIDAS PARA GARANTIR A PROTEÇÃO DOS DADOS E A AUTENTICAÇÃO ADEQUADA DOS USUÁRIOS, DE MODO A OFERECER UM AMBIENTE SEGURO PARA A TROCA DE MENSAGENS.

# Tecnologias

---

## TECNOLOGIAS UTILIZADAS NA ELABORAÇÃO DO PROJETO

### Visual Studio(IDE)

O VISUAL STUDIO É UMA PODEROSA FERRAMENTA DE DESENVOLVIMENTO INTEGRADO (IDE) FORNECIDA PELA MICROSOFT. ELE OFERECE RECURSOS AVANÇADOS PARA EDIÇÃO DE CÓDIGO, DEPURAÇÃO, TESTES E IMPLANTAÇÃO. COM SUPORTE A VÁRIAS LINGUAGENS DE PROGRAMAÇÃO, INCLUINDO C#, É UMA ESCOLHA POPULAR PARA O DESENVOLVIMENTO DE APLICATIVOS WINDOWS FORMS.

### PicsArt

O PICSART É UM APLICATIVO DE EDIÇÃO DE IMAGENS AMPLAMENTE UTILIZADO. ELE OFERECE RECURSOS AVANÇADOS DE EDIÇÃO, COMO FILTROS, EFEITOS, FERRAMENTAS DE RECORTE, SOBREPOSIÇÕES, ENTRE OUTROS. O USO DO PICSART PODE SER RELEVANTE PARA ESTE PROJETO CASO SEJA NECESSÁRIO MANIPULAR OU PERSONALIZAR IMAGENS, COMO A IMAGEM DE PERFIL DOS USUÁRIOS NO APLICATIVO DE CHAT.

### Diagrams.net

O DIAGRAMS.NET, ANTERIORMENTE CONHECIDO COMO DRAW.IO, É UMA FERRAMENTA ONLINE PARA CRIAÇÃO DE DIAGRAMAS E FLUXOGRAMAS. ELA FORNECE UMA AMPLA GAMA DE RECURSOS PARA CRIAR DIAGRAMAS VISUAIS, INCLUINDO DIAGRAMAS DE CLASSES, FLUXOGRAMAS, DIAGRAMAS DE SEQUÊNCIA, ENTRE OUTROS. PODE SER ÚTIL NO PROJETO PARA CRIAR DIAGRAMAS DE CLASSE OU OUTROS TIPOS DE DIAGRAMAS QUE AUXILIEM NA COMPREENSÃO E VISUALIZAÇÃO DA ESTRUTURA DO APLICATIVO DE CHAT.

### SQL Server 2022

O SQL SERVER 2022 É A VERSÃO MAIS RECENTE DO SISTEMA DE GERENCIAMENTO DE BANCO DE DADOS DA MICROSOFT. ELE FOI PROJETADO PARA OFERECER RECURSOS AVANÇADOS DE ARMAZENAMENTO, PROCESSAMENTO E ANÁLISE DE DADOS, AJUDANDO AS EMPRESAS A LIDAR COM GRANDES VOLUMES DE INFORMAÇÕES DE MANEIRA EFICIENTE.

### SQL Server Management Studio (SSMS)

O SQL SERVER MANAGEMENT STUDIO (SSMS) É UMA FERRAMENTA DE GERENCIAMENTO E DESENVOLVIMENTO PARA O SQL SERVER, FORNECIDA PELA MICROSOFT. ELE OFERECE UMA INTERFACE GRÁFICA INTUITIVA E PODEROSA PARA ADMINISTRAR E MANIPULAR BANCOS DE DADOS DO SQL SERVER.

# Documento de Requisitos

---

## Documentação de requisitos

Na documentação de requisitos, foram levantadas questões sobre como elaborar o projeto para criar uma linha a se seguir do desenvolvimento, e se necessário fazer alterações que possam facilitar o entendimento de criação do projeto.

Alguns requisitos levantados para a elaboração deste projeto foram:

### OBJETIVO DO SISTEMA DE COMUNICAÇÃO

Voltado apenas ao diálogo entre duas pessoas, com apenas mensagens de texto e uso contínuo diário, com um nível de segurança dotado de um login e uma senha única para cada tipo de usuário, com autenticação de email e senha para evitar confusão entre os usuários que tenham nomes idênticos ou parecido.

O meio de comunicação utilizado pelo sistema será de rede local e banco de dados para armazenar as informações de cada usuário e cada conversa que criarem entre si.

Para melhor experiência de usuário com o sistema, uma possibilidade de alterar o tema de fundo da aplicação também será implementado, junto também de ser possível realizar alterações no perfil do usuário, sendo essas o nome de usuário, senha e imagem de perfil.

Necessário também implementação de um sistema que possa adicionar usuários diferentes a uma lista de amigos, posteriormente sendo possível a comunicação entre si.

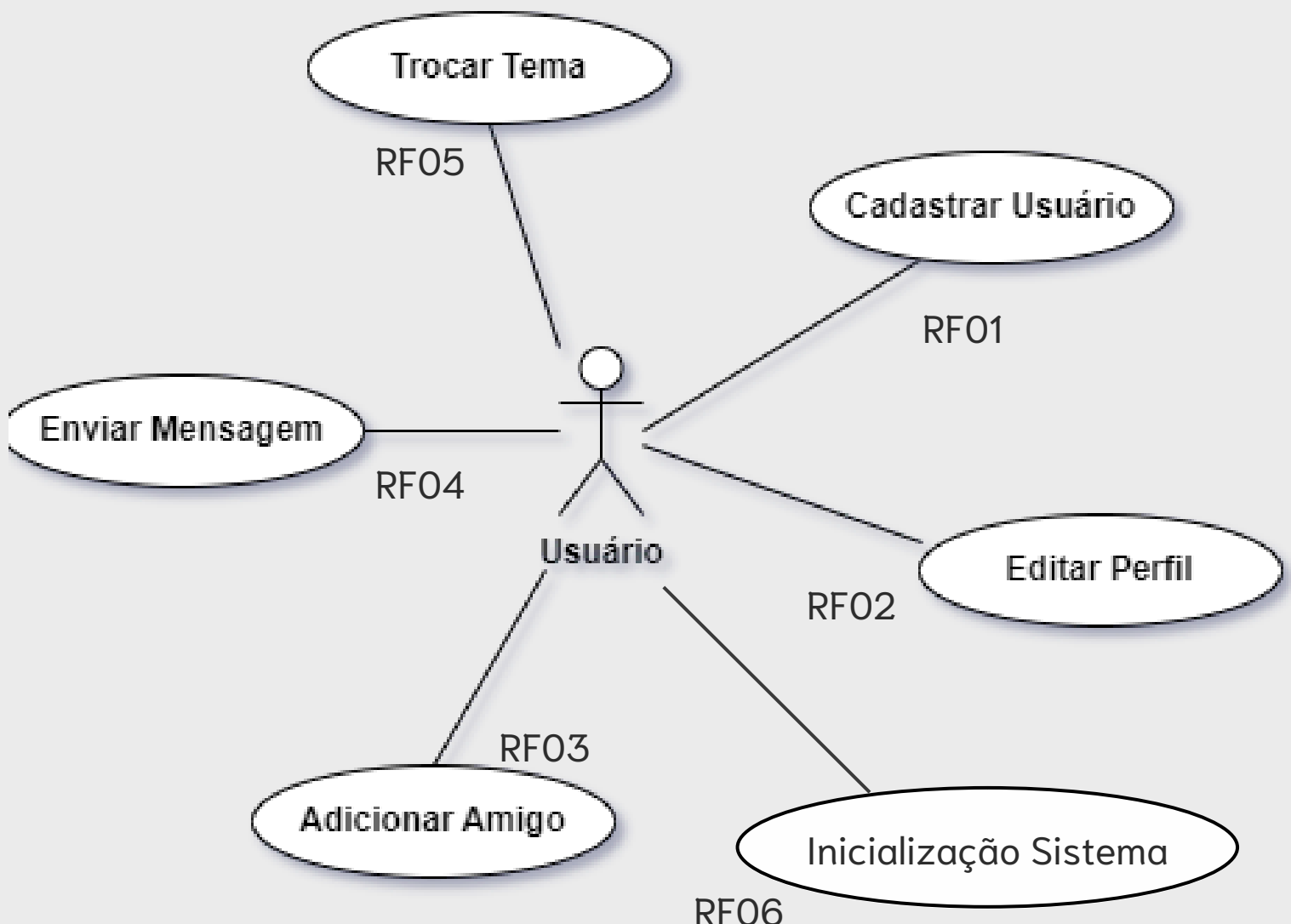
---

# Requisitos Funcionais de Usuário

Requisitos funcionais de usuário

As funcionalidades que o usuário poderá executar no sistema serão descritas nessa sessão. Na figura abaixo, será exibido um diagrama de casos de uso do sistema desenvolvido, mostrando todas as possibilidades de utilização que o usuário poderá exercer.

Figura 1 – Diagrama de Casos de Uso



# Requisitos Funcionais de Usuário

---

## Requisitos funcionais de usuário

O Usuário, é o ator responsável por todas as operações no sistema, a seguir, é descrito os requisitos funcionais de usuário.

### **Requisito RF01 – Cadastrar Usuário**

O sistema permite que o usuário possa se cadastrar como um novo usuário no sistema.

### **Requisito RF02 – Editar Perfil**

O sistema permite ao usuário cadastrado, editar seus dados, como Nome, Senha e Foto de Perfil.

### **Requisito RF03 – Adicionar Amigo**

O sistema permite ao usuário cadastrado, adicionar outros usuários cadastrados no sistema, na sua lista de amigos.

### **Requisito RF04 – Enviar Mensagem**

O sistema permite ao usuário cadastrado, enviar mensagens aos usuários que estejam na sua lista de amigos.

### **Requisito RF05 – Trocar Tema**

O sistema permite para qualquer usuário, cadastrado ou não, que altere o tema do sistema, ou seja, as cores de fundo do programa.

### **Requisito RF06 – Inicialização Sistema**

O sistema é inicializado para poder realizar todas as tarefas descritas acima.



# Requisitos Não Funcionais de Usuário

---

## Requisitos não funcionais de usuário

### Requisito RNFU01 – Responsividade

O sistema deve responder rapidamente às interações do usuário, garantindo um tempo de resposta mínimo entre o envio de uma ação (como enviar uma mensagem) e a exibição do resultado correspondente (como a mensagem aparecer na conversa). O chat deve ser ágil e evitar atrasos perceptíveis que possam prejudicar a experiência do usuário.

### Requisito RNFU03 – Disponibilidade

O sistema deve funcionar e estar disponível para uso via conexão LAN, ou seja, conexão local que seja dentro da mesma rede.

### Requisito RNFU04 – Facilidade de uso

O sistema deve ser intuitivo e de fácil utilização, permitindo que os usuários, mesmo sem experiência prévia, consigam realizar as principais tarefas do chat de forma simples e sem complicações. Isso inclui uma interface amigável, instruções claras e um fluxo de interação natural.

# Requisitos Não Funcionais do Sistema

---

## Requisitos não funcionais do Sistema

Descrevem os requisitos não funcionais do sistema, como a utilização, desempenho, segurança, escalabilidade de uso entre outras funcionalidades não funcionais.

### Requisito RNFO1 – Utilização

O sistema proposto foi desenvolvido para sistemas operacionais Windows 10, em linguagem de programação C#, e de forma a aproveitar a interação desta linguagem com o sistema operacional utilizado. Foi utilizado o *.NET framework 7.0.0* para o desenvolvimento do sistema, sendo assim necessário que o usuário possua instalado em seu computador essa versão do *framework*

### Requisito RNFO2 – Desempenho

O sistema deve ser capaz de processar e exibir as mensagens de forma rápida e eficiente, garantindo uma experiência de chat fluida, mesmo quando houver um grande volume de mensagens ou usuários simultâneos.

### Requisito RNFO3 – Segurança

O sistema deve garantir a segurança das contas de todos os usuários cadastrados no sistema, utilizando criptografia para proteger as senhas no banco de dados.

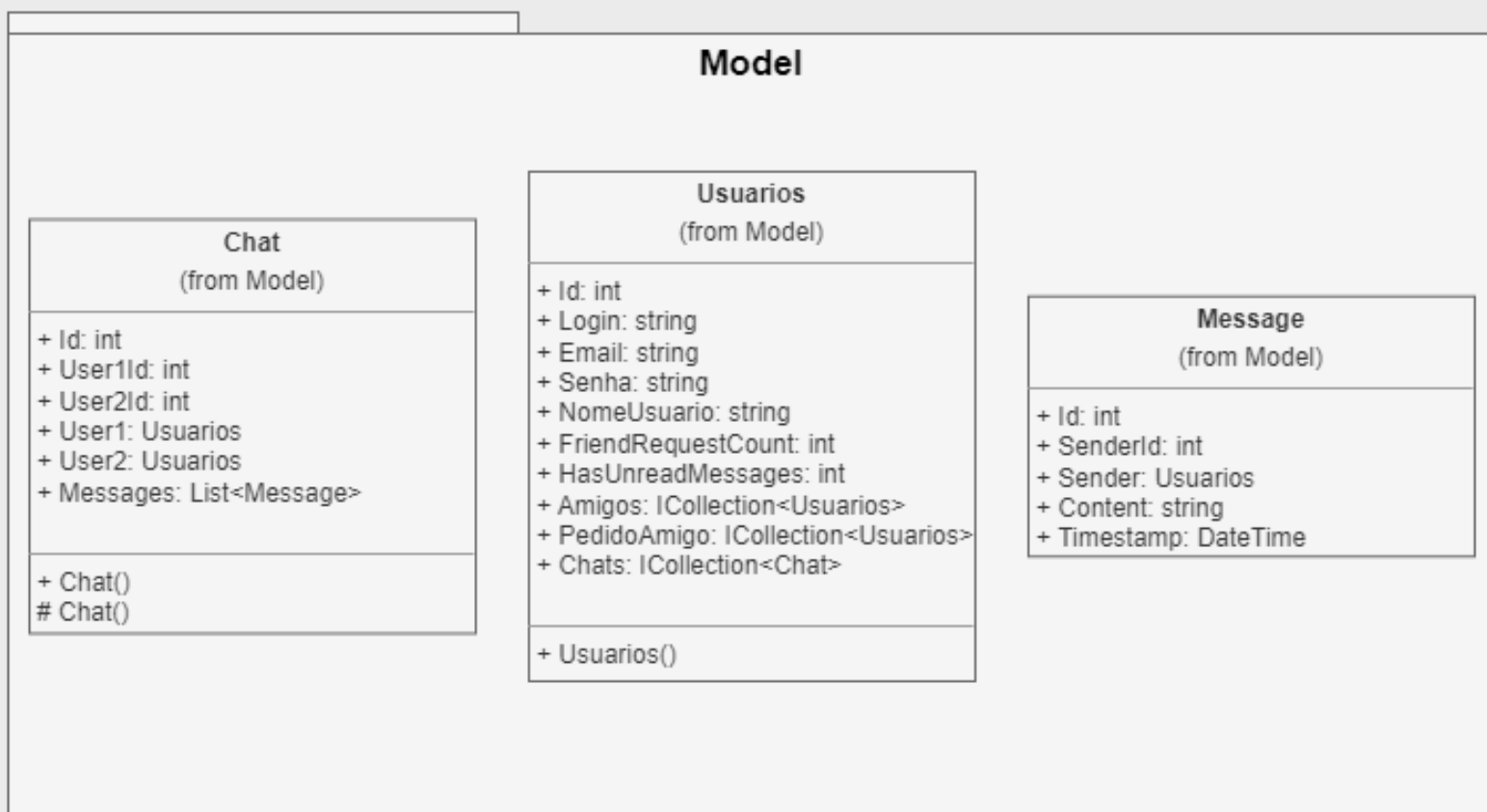
### Requisito RNFO4 – Escalabilidade

O sistema deve ser capaz de lidar com o crescimento no número de usuários e volume de mensagens sem comprometer o desempenho. Deve ser possível adicionar recursos adicionais, como servidores ou capacidade de armazenamento, para atender às demandas crescentes do sistema.

# Requisitos Funcionais do Sistema

## Funcionalidades do sistema

Aqui serão descritas as funcionalidades do sistema de forma completa para auxílio no desenvolvimento do sistema. Nessa sessão da documentação, irá ser distribuído em camadas as classes do sistema de forma a facilitar o entendimento do sistema como um todo.



A imagem acima é uma representação da pasta Model que irá comportar as classes que serão desenvolvidas no projeto, a fim de colaborar para com que a aplicação atinja os objetivos abordados.

# Requisitos Funcionais do Sistema

Funcionalidades do sistema

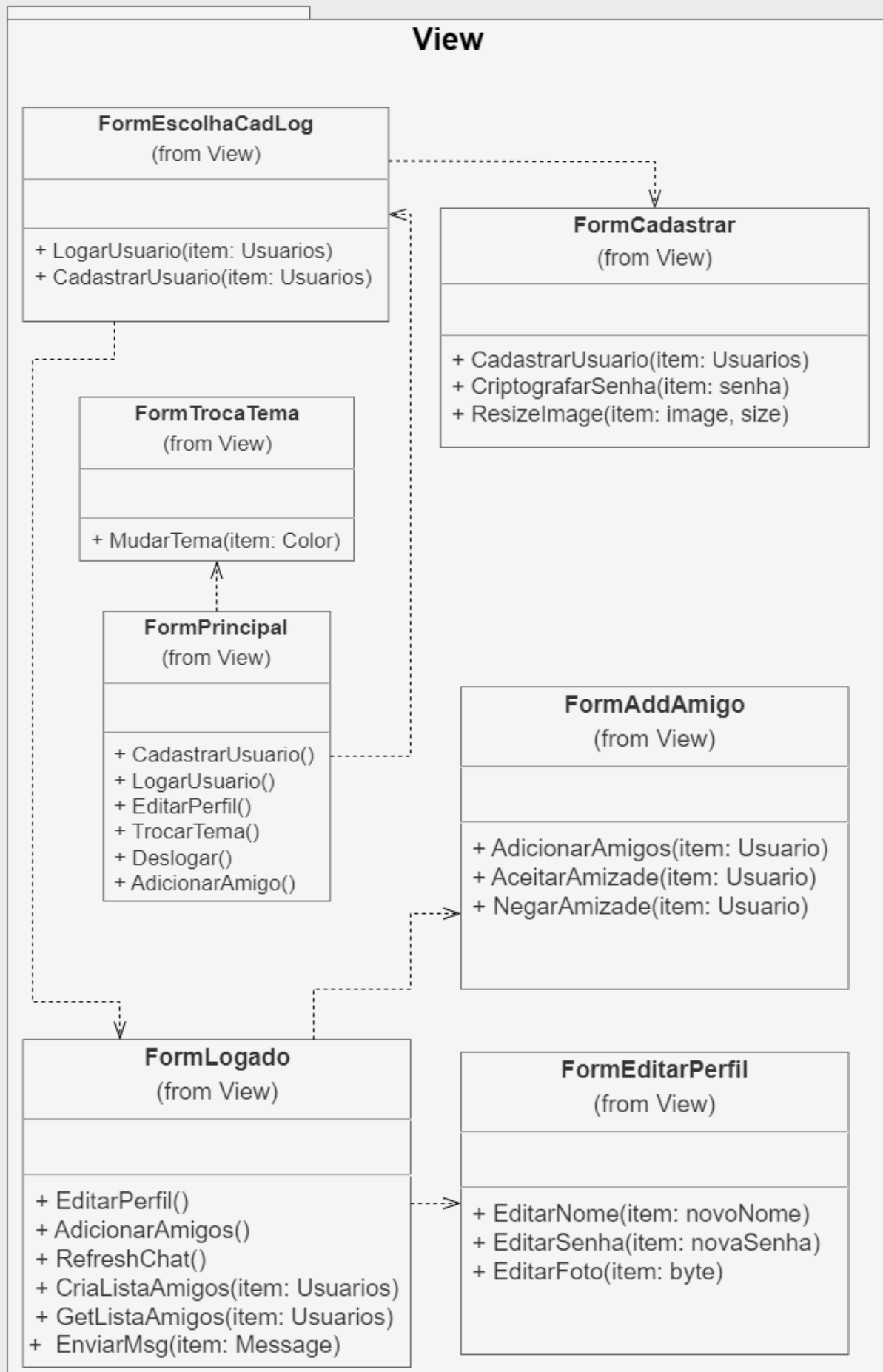
## Control

### UsuarioController (from Control)

- SalvarUsuario(item: Usuarios): void
- AtualizarNome(item: login, novoNome): void
- AtualizarSenha(item: login, novaSenha): void
- SalvarImagem(item: login, imageData): void
- GetImagem(item: login): byte
- EnviarPedidoAmizade(item: senderEmail, recipientEmail): void
- GetFriendList(item: userId): List<Usuarios>
- SaveChat(item: chat): void
- GetChat(item: user1, user2): Chat

# Requisitos Funcionais do Sistema

Funcionalidades do sistema



# Diagrama de Classes

Diagrama de Classes do projeto ChatOn

Usuarios
+ Id: int + Login: string + Senha: string + Email: string + NomeUsuario: string + FriendRequestCount: int + HasUnreadMessages: int + Amigos: ICollection<Usuarios> + PedidoAmigo: ICollection<Usuarios> + Chats: ICollection<Chat>
+ Usuarios()

Message
+ Id: int + SenderId: int + Sender: Usuarios + Content: string + Timestamp: DateTime

Chat
+ Id: int + User1Id: int + User2Id: int + User1: Usuarios + User2: Usuarios + Messages: List<Message>
+ Chat(user1, user2)

As imagens ao lado representam a classe Usuarios do programa, que irá conter as classes Message e Chat para cooperarem entre si para criarem um chat específico entre 2 usuários distintos e mensagens armazenadas para cada usuário que a enviou, para quando a funcionalidade estiver de acordo, o banco de dados conseguir recuperar o chat por completo das mensagens que um usuário enviou para o outro na conversa existente entre ambos.

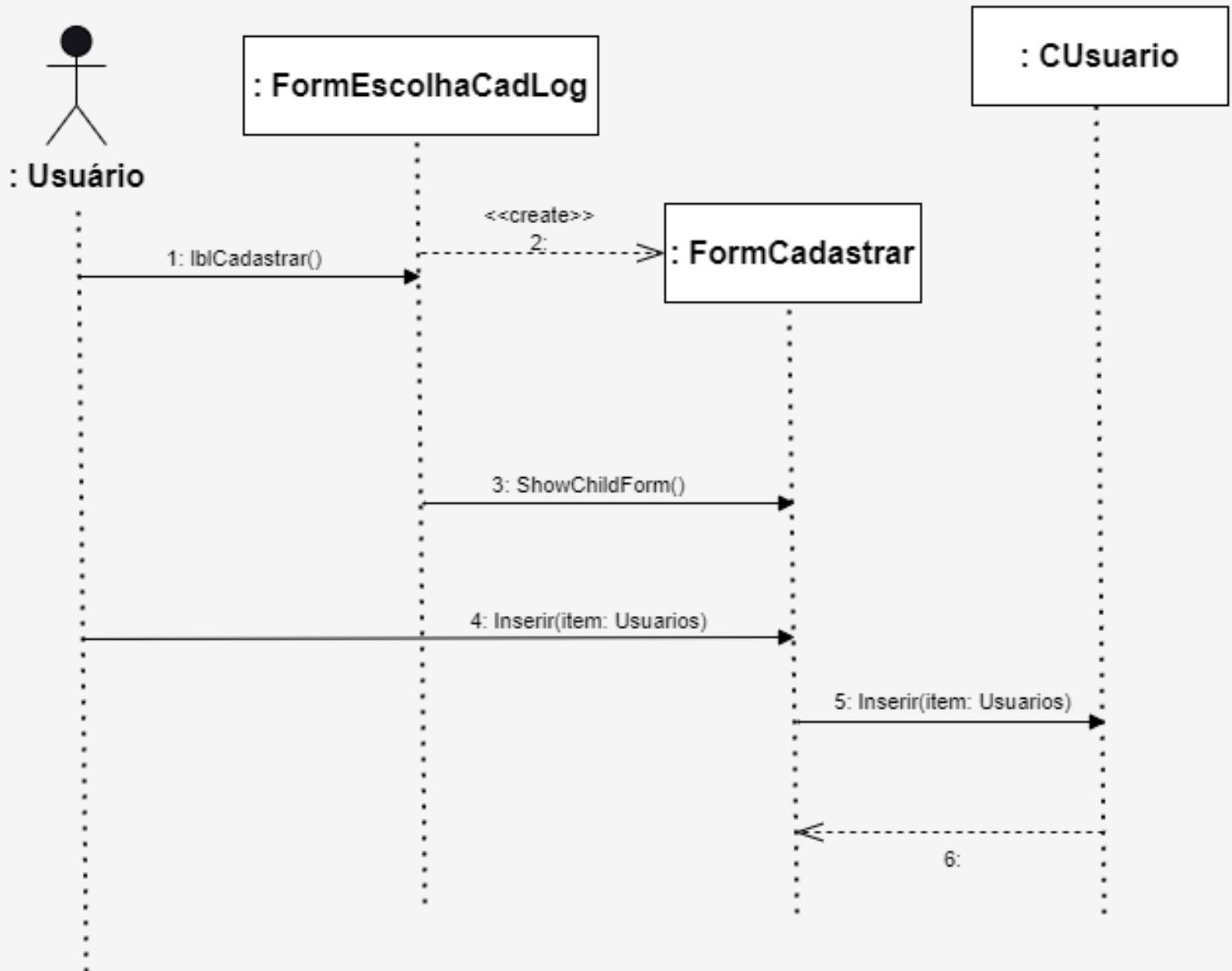
# Descrição dos Requisitos do Sistema

## Requisito RF01 – Cadastrar Usuário

<b>Função</b>	Cadastra um novo usuário no Banco de Dados.
<b>Descrição</b>	Insere na tabela Usuario do banco de dados um novo usuário contendo as informações fornecidas pelo usuário do sistema.
<b>Entradas</b>	Variável Login do tipo string; Variável Email do tipo string; Variável Senha do tipo string.
<b>Origem</b>	Usuário.
<b>Destino</b>	Cadastrar novo usuário no banco de dados.
<b>Ação</b>	O usuário indica que deseja se cadastrar no sistema, selecionando uma opção no formulário principal do sistema, um formulário será exibido com campos de cadastro e ao clicar no botão "Cadastrar" se caso todos os campos forem completos corretamente, realiza a chamada do evento Click para realizar a inserção do novo usuário no sistema por meio da função SalvarUsuario.
<b>Pré-Condição</b>	O sistema deve estar em execução e o banco de dados online.
<b>Pós-Condição</b>	O sistema deve avisar ao usuário se o cadastro foi realizado com sucesso ou se ocorreu algum erro

# Diagrama de Sequência do RF01

## Interaction IntCadastrarUsuario





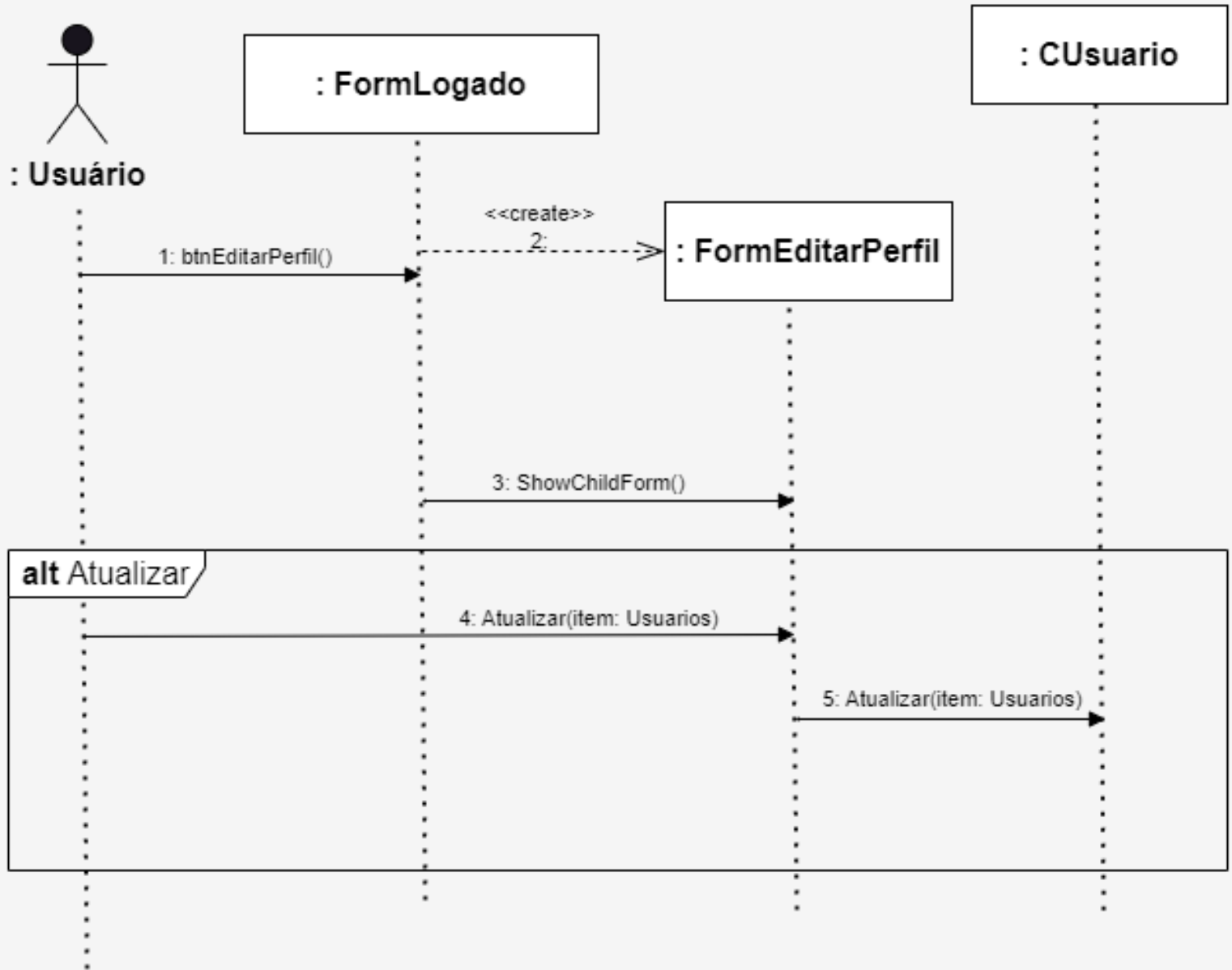
# Descrição dos Requisitos do Sistema

## Requisito RF02 – Editar Perfil

<b>Função</b>	Editar nome, senha e imagem de perfil do usuário.
<b>Descrição</b>	Atualiza na tabela Usuario do banco de dados os dados referentes à opção que o usuário determinar, como Nome, Senha ou Imagem de Perfil.
<b>Entradas</b>	Variável novoNome do tipo string; Variável novaSenha do tipo string; Variável ImgPerfil do tipo Byte.
<b>Origem</b>	Usuário.
<b>Destino</b>	Atualizar dados no banco de dados referente ao usuário logado.
<b>Ação</b>	O usuário indica o que ele pretende atualizar preenchendo os dados nos campos vazios que representam o nome, senha ou no campo de imagem, posteriormente clicando em um botão que representa o campo preenchido para atualizar no banco os novos dados preenchidos pelo mesmo.
<b>Pré-Condição</b>	O sistema deve estar em execução e o banco de dados online.
<b>Pós-Condição</b>	O sistema deve avisar ao usuário se os dados foram atualizados com sucesso ou se ocorreu algum erro

# Diagrama de Sequência do RF02

## Interaction IntEditarPerfil



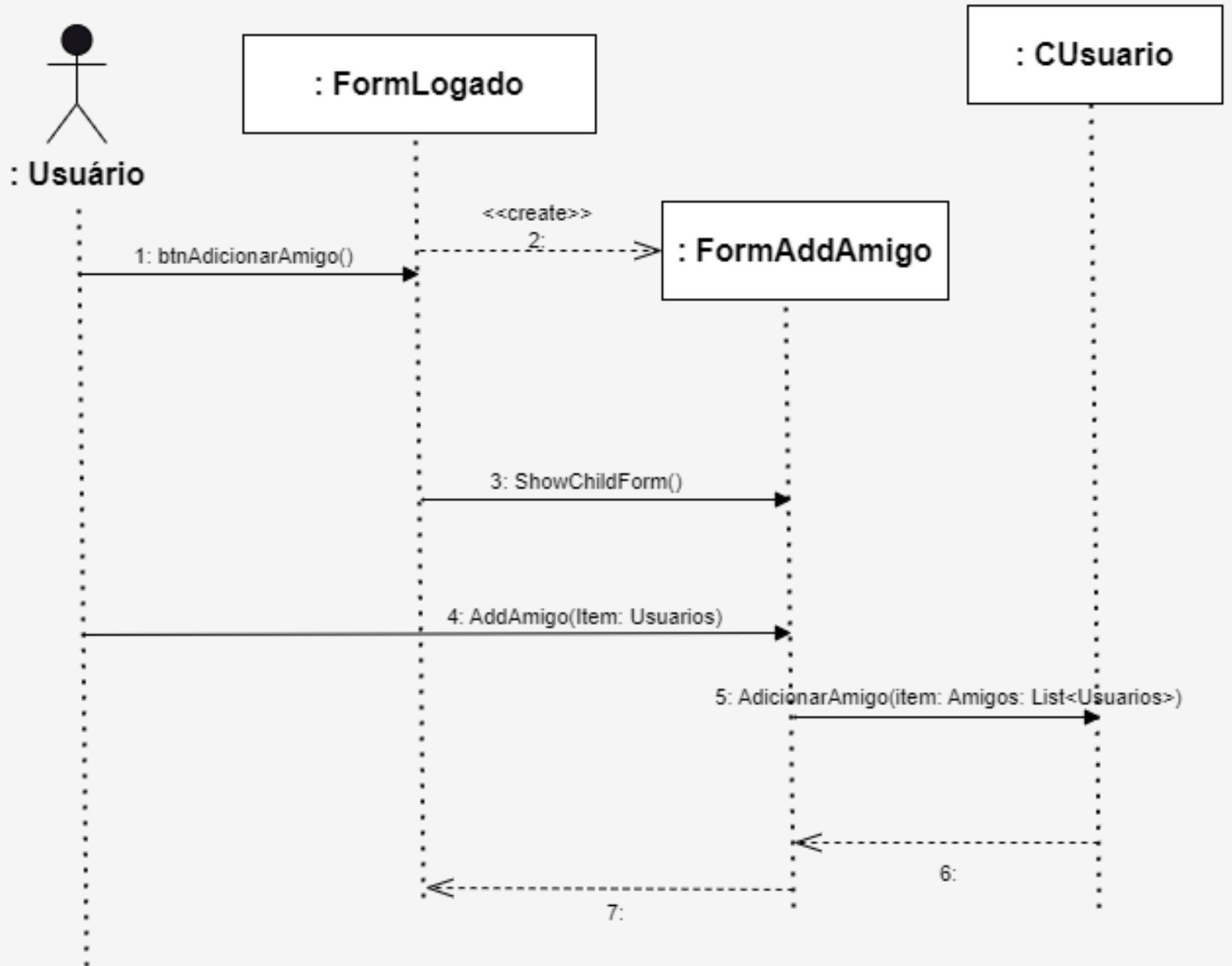
# Descrição dos Requisitos do Sistema

## Requisito RF03 – Adicionar Amigo

<b>Função</b>	Adicionar um usuário na lista de amigos do usuário logado.
<b>Descrição</b>	Adiciona na tabela UsuariosUsuarios do banco de dados o ID do usuarioAmigo a ser adicionado e o pedido de amizade do usuário que o enviou, e quando o outro usuário aceitar o pedido, será adicionado o ID do usuarioAmigo que enviou o pedido de amizade e o usuário que recebeu o pedido, assim automaticamente serão adicionados na lista de amigos de cada um cada usuário que enviou o pedido de amizade.
<b>Entradas</b>	Variável email do tipo string.
<b>Origem</b>	Usuário.
<b>Destino</b>	Adicionar no banco de dados na tabela UsuariosUsuarios os dados referentes ao ID de quem enviou o pedido de amizade e o ID do alvo do pedido de amizade.
<b>Ação</b>	O usuário logado seleciona o email do usuário que ele pretende adicionar na sua lista de amigos e clica no botão Enviar pedido de amizade, assim enviando a solicitação e aguardando para ser aceita ou cancelada.
<b>Pré-Condição</b>	O sistema deve estar em execução e o banco de dados online.
<b>Pós-Condição</b>	O sistema deve avisar ao usuário se o pedido de amizade foi enviado com sucesso ou se ocorreu algum erro.

# Diagrama de Sequência do RF03

## Interaction IntAddAmigo



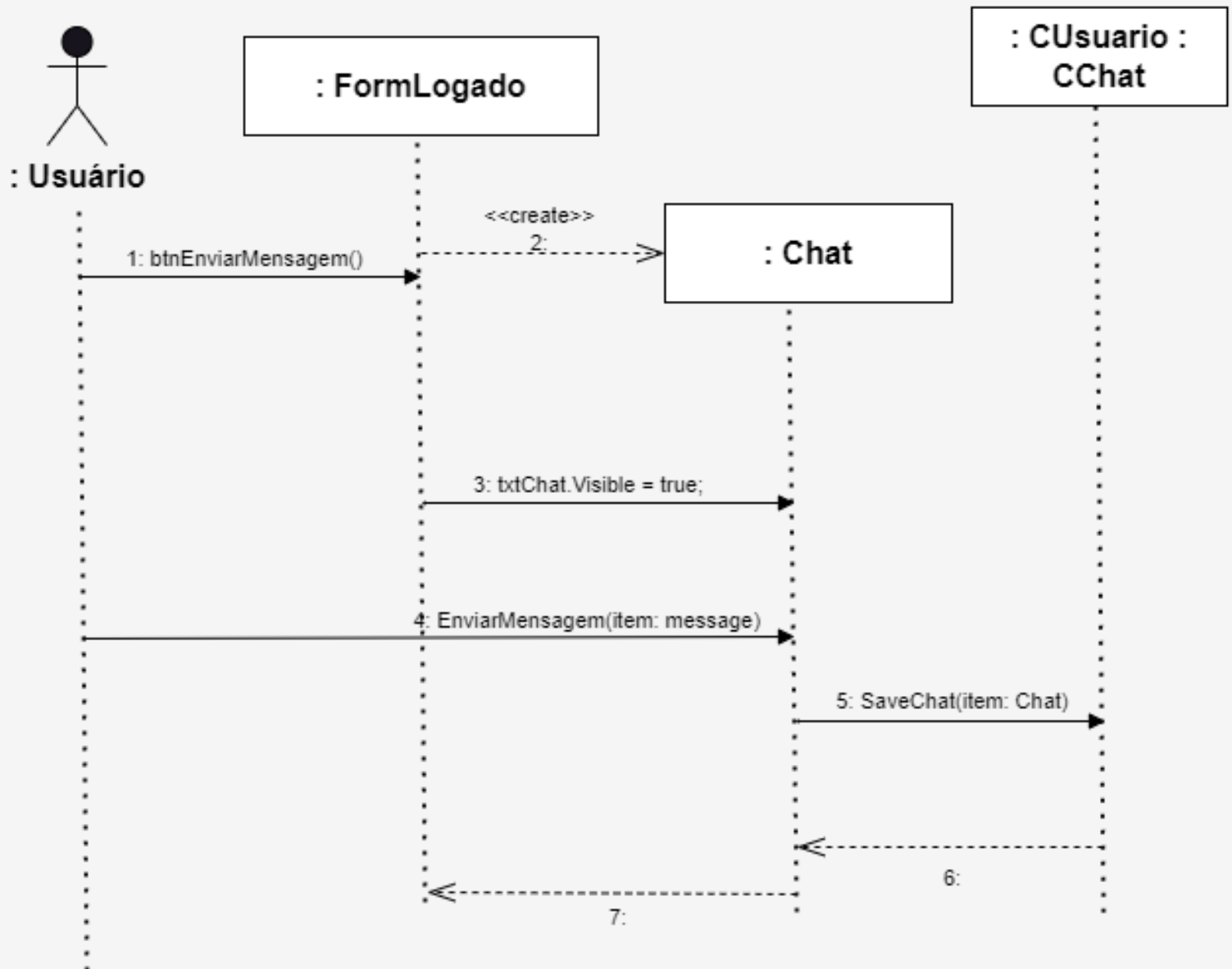
# Descrição dos Requisitos do Sistema

## Requisito RF04 – Enviar Mensagem

<b>Função</b>	Enviar mensagem para um usuário que esteja na lista de amigos do usuário logado.
<b>Descrição</b>	Adiciona uma nova conversa na tabela Chat do banco de dados com o ID dos dois usuários que estão conversando, gerando um ID próprio para o chat e adicionando o chat criado na tabela Chat do banco de dados, e assim ambos os usuários terão um chat com uma ID específica deles e todas as mensagens enviadas por cada usuário vai ter: um ID próprio, ID de quem enviou a mensagem, conteúdo da mensagem, o horário que a mensagem foi enviada e o ID do chat que as mensagens estão sendo enviadas.
<b>Entradas</b>	Variável content do tipo string.
<b>Origem</b>	Usuário.
<b>Destino</b>	Adicionar no banco de dados na tabela Chat um novo chat com uma ID única do chat entre dois usuários e suas respectivas IDs, criando na tabela Message uma nova conversa entre ambos os usuários.
<b>Ação</b>	O usuário logado seleciona o usuário que está na lista de amigos e posteriormente digita uma mensagem e clica no botão enviar, e a mensagem será enviada no chat de ambos os usuários podendo ocorrer uma interação mútua.
<b>Pré-Condição</b>	O sistema deve estar em execução e o banco de dados online.
<b>Pós-Condição</b>	O sistema deve mostrar a mensagem enviada no chat de ambos os usuários corretamente.

# Diagrama de Sequência do RF04

## Interaction EnviarMensagem

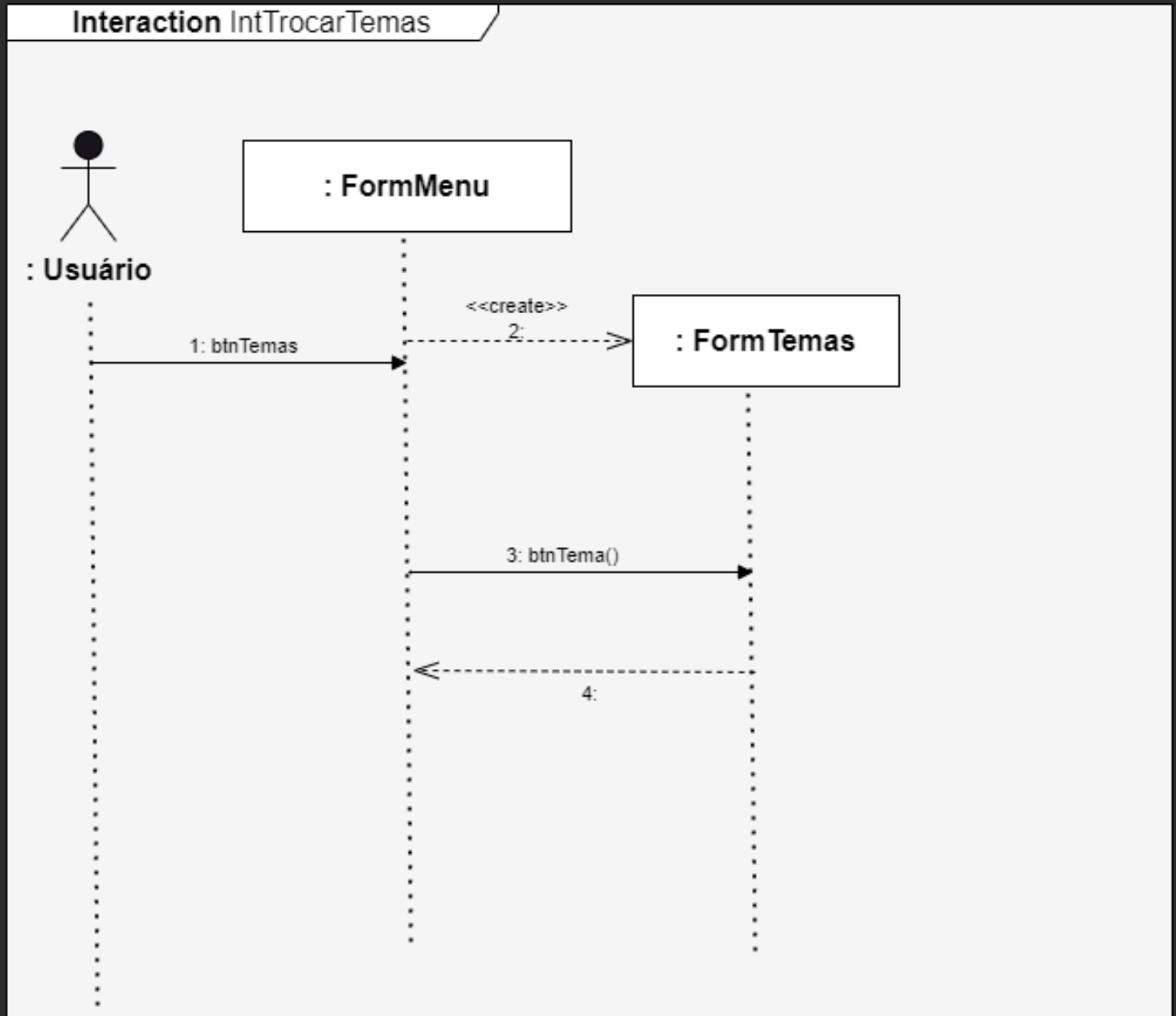


# Descrição dos Requisitos do Sistema

## Requisito RF05 – Trocar Temas

<b>Função</b>	Trocar o tema de fundo do programa para cores diferentes, para agradar mais visualmente os gostos do usuário.
<b>Descrição</b>	Muda o tema de fundo do programa em todas as abas e formulários que ele abra, mantendo a cor escolhida, podendo trocá-la a qualquer momento.
<b>Entradas</b>	Variável cor1 do tipo Color; Variável cor2 do tipo Color.
<b>Origem</b>	Usuário.
<b>Destino</b>	–
<b>Ação</b>	O usuário logado ou não, seleciona qualquer um dos 6 temas disponíveis para trocar as cores do tema de fundo do programa por meio de botões.
<b>Pré-Condição</b>	O sistema deve estar em execução.
<b>Pós-Condição</b>	O sistema deve trocar as cores de imediato corretamente.

# Diagrama de Sequência do RF05





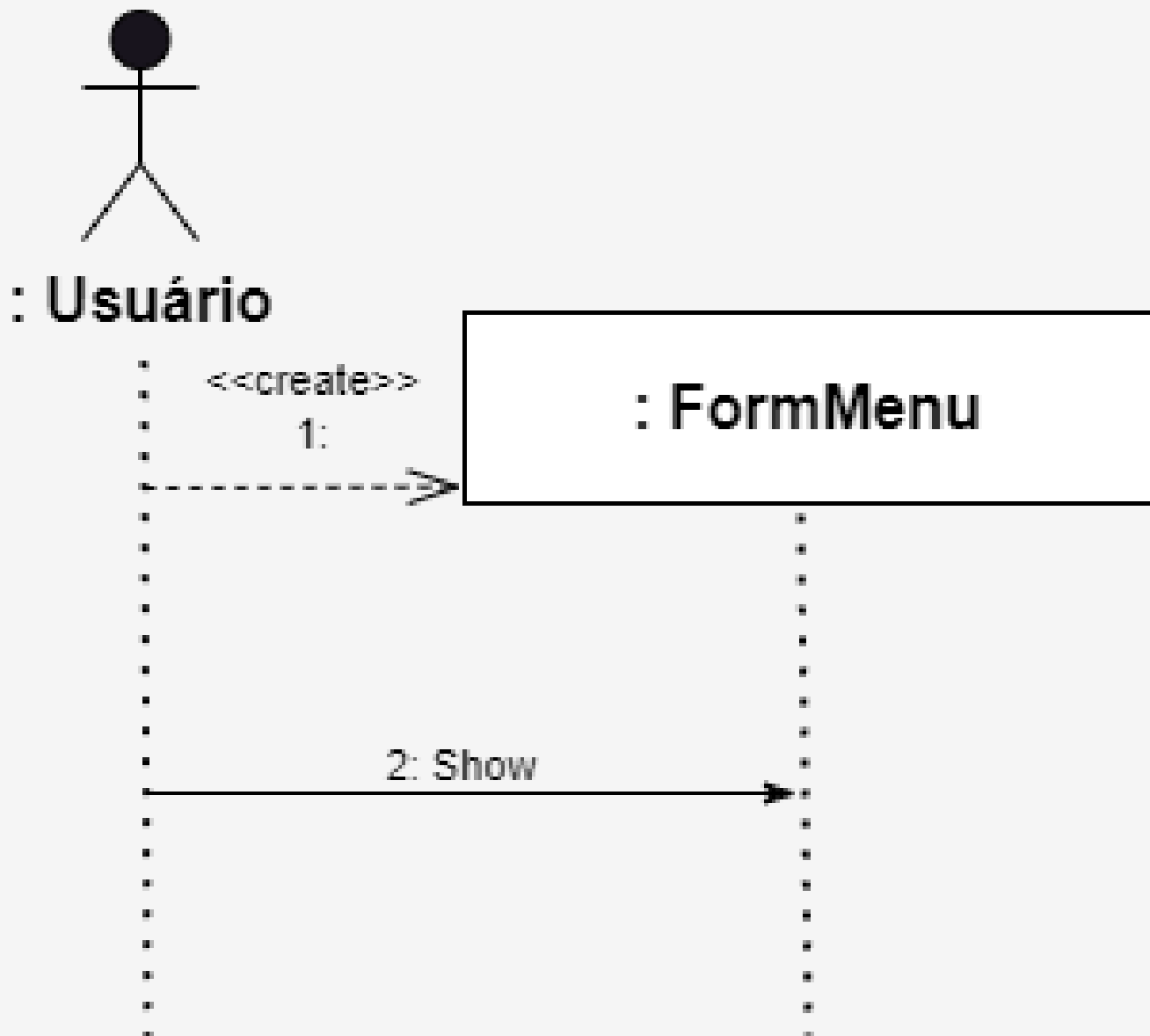
# Descrição dos Requisitos do Sistema

## Requisito RF06 – Iniciar Sistema

<b>Função</b>	Inicia o sistema.
<b>Descrição</b>	Realiza a inicialização do sistema mostrando o formulário principal, que por meio dele será possível acessar os demais formulários.
<b>Entradas</b>	Nenhuma.
<b>Origem</b>	–
<b>Destino</b>	–
<b>Ação</b>	Ao iniciar o sistema, um novo objeto do tipo FormMenu deve ser criado e exibido. Ele deve possuir um menu para permitir que o usuário possa trocar de tema e realizar seu login ou cadastrar-se no sistema. As demais opções só serão liberadas caso o usuário esteja logado no sistema.
<b>Pré-Condição</b>	Que o computador do usuário esteja ligado e que tenha os frameworks necessários para seu funcionamento.
<b>Pós-Condição</b>	O sistema deve inicializar normalmente ou retornar algum erro dizendo qual componente falta no computador do usuário para ser inicializado.

# Diagrama de Sequência do RF06

## Interaction Iniciar Sistema



# Modelagem do Banco de Dados

Modelagem do Banco de Dados

A MODELAGEM DO BANCO DE DADOS FOI DIVIDIDA EM 4 CLASSES, DENTRE ELAS SÃO:

## USUÁRIOS

A CLASSE USUÁRIOS É A CLASSE PRINCIPAL QUE IRÁ ARMAZENAR O LOGIN, EMAIL, NOME DE USUÁRIO, SENHA, IMAGEM DE PERFIL, LISTA DE AMIGOS, PEDIDOS DE AMIZADE E CHAT PARA CADA AMIGO. AQUI É DEFINIDO A JUNÇÃO DAS 3 PRINCIPAIS CLASSES PARA FAZER COM QUE O PROGRAMA DE FATO FUNCIONE COMO FORA PLANEJADO E DOCUMENTADO.

## CHAT

A CLASSE CHAT É A CLASSE QUE CONTÉM O SEU PRÓPRIO ID E O ID DOS DOIS USUÁRIOS QUE ESTÃO CONVERSANDO ENTRE SI, ALÉM DE ARMAZENAR UMA LISTA DE MENSAGENS DE CADA USUÁRIO DENTRO DELA MESMA, PARA FACILITAR A LEITURA E RECUPERAÇÃO DA CONVERSA QUANDO UM USUÁRIO MIGRA DE UMA CONVERSA PARA OUTRA.

## MESSAGE

A CLASSE MESSAGE É UMA CLASSE QUE POSSUI UM ID PRÓPRIO, O USUÁRIO QUE ENVIOU A DETERMINADA MENSAGEM, UMA STRING COM O CONTEÚDO DA MENSAGEM E UM DATETIME TIMESTAMP PARA PEGAR O HORÁRIO COM A QUAL A MENSAGEM FOI ENVIADA.

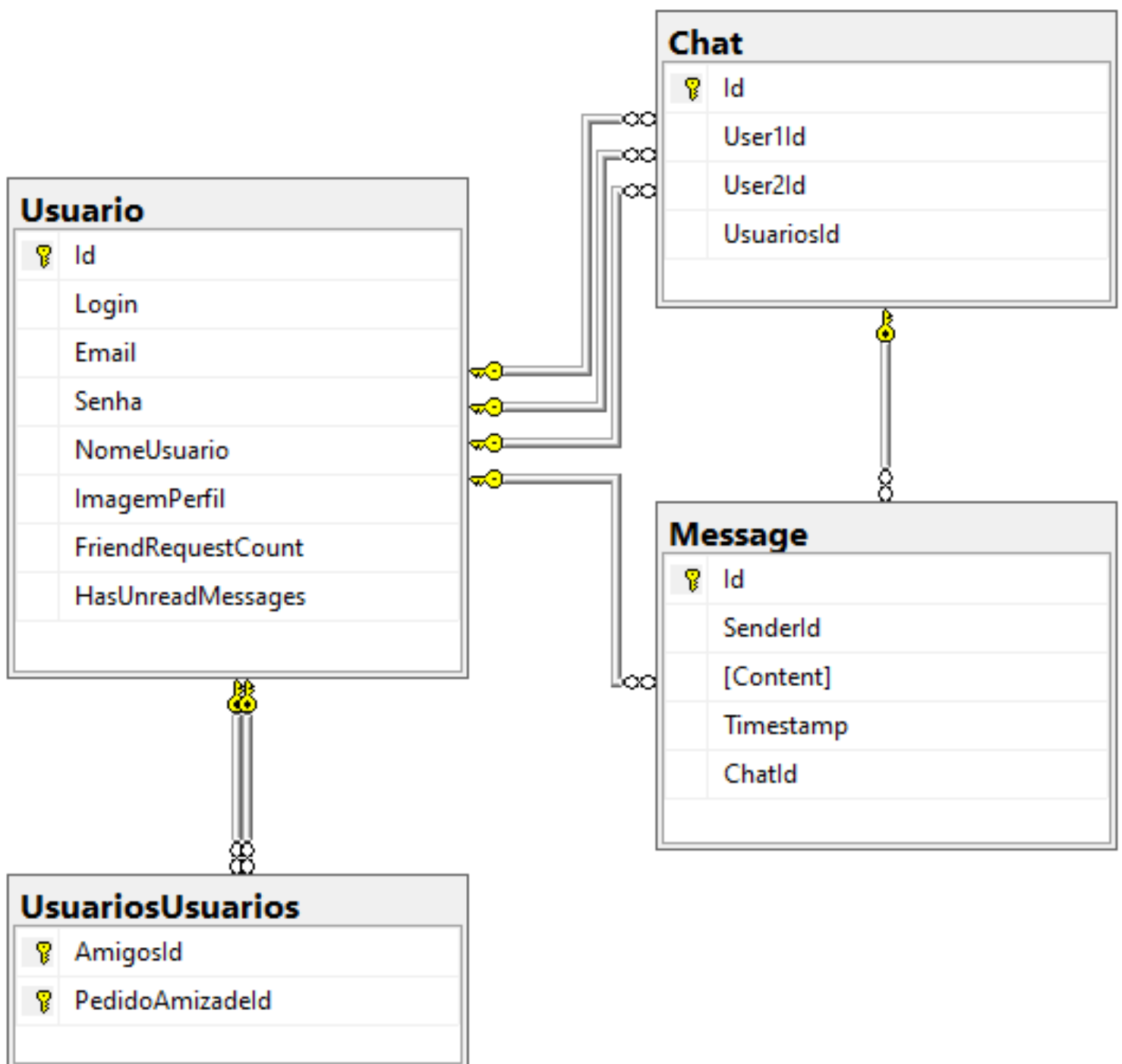
## TEMASETTINGS

A CLASSE TEMASETTINGS É UMA CLASSE QUE PEGA DUAS CORES, UMA COR PARA O MENU DO PROGRAMA E OUTRA COR PARA O FUNDO DO PROGRAMA, PARA PODER SER POSSÍVEL OCORRER A MODIFICAÇÃO DOS TEMAS NO PROGRAMA, MUDANDO PARA TEMA ESCURO, TEMA VERDE, TEMA AZUL ENTRE OUTROS, DANDO UMA DIVERSIDADE A MAIS NA INTERFACE DO PROGRAMA PARA O USUÁRIO.

# Modelagem do Banco de Dados

Modelagem do Banco de Dados

Nessa seção é descrito como o armazenamento do sistema é feito no banco de dados utilizando SQL Server 2022





ChatOn

---

TIAGO ALEXANDRE DOCUSSE

Professor

### Alunos

---

AMERSON SAMUEL OLIVEIRA MACIEL – Programador

ELIEZER ALMEIDA DE ARAUJO – Analista

JONATHAN RODRIGUES BERNARDES – Programador

PEDRO HENRIQUE SANTOS MORAIS – Analista

RENAN IAGO CHAVES – Programador