

## 1. Introdução

- **Nome do Projeto:** Miau

- **Breve Descrição do Projeto:**

O Miau é uma rede social dedicada a amantes de gatos, onde usuários podem criar perfis para si mesmos, enviar fotos de seus pets com informações como nome, peso, idade e descrição, visualizar feeds com scroll infinito, comentar em fotos (inclusive com sugestões geradas por IA), e explorar perfis de outros usuários com seus pets. O objetivo é conectar donos de gatos em uma comunidade divertida e interativa, resolvendo a falta de plataformas específicas para esse público. O público-alvo são donos de gatos que buscam compartilhar e celebrar seus pets.

- **Motivação para a Escolha do Projeto:**

A escolha do projeto nasceu da paixão por gatos e do desejo de criar uma forma divertida de compartilhar fotos de pets. Diferente das opções disponíveis hoje, o Miau propõe uma experiência intuitiva ao combinar envio de fotos com informações detalhadas dos gatos, como nome, peso e idade, e a integração de inteligência artificial para sugerir comentários. Esse diferencial, aliado a uma interface simples e funcional, torna o projeto um desafio técnico empolgante e uma oportunidade de oferecer algo novo e adorável para os amantes de gatos.

## 2. Tecnologias Utilizadas

- **Linguagens de Programação:**

- Front-end: JavaScript (React.js),
- Back-end: C# (.NET)

- **Frameworks e Bibliotecas:**

- Front-end: React.js (com hooks e bibliotecas como Axios para chamadas API), SCSS module para estilização
- Back-end: ASP.NET Core (API REST)
- Banco de Dados: Entity Framework Core
- IA: Ollama (para sugestões de comentários)

- **Banco de Dados:**

- SQLServer (com stored procedures, views e functions para lógica de banco)

- **Ferramentas de Desenvolvimento:**

- Visual Studio Code (editor para front-end)
- Visual Studio (IDE para back-end)
- Git e GitHub (controle de versão)
- Cloudinary (hospedagem de imagens)
- Docker (ainda uma dúvida, para rodar Ollama)

## 3. Requisitos do Sistema

- **Requisitos Funcionais:**

- O sistema deve permitir o cadastro e login de usuários com autenticação segura.

- O sistema deve possibilitar que donos de pets enviem fotos de seus gatos com informações (nome, peso, idade, descrição).
- O sistema deve exibir um feed de fotos com scroll infinito.
- O sistema deve permitir comentários em fotos, incluindo sugestões geradas por IA.
- O sistema deve possibilitar a visualização de perfis de outros usuários com as fotos de seus pets.
- O sistema deve permitir abrir detalhes de uma foto com suas informações e comentários.
- O sistema deve armazenar imagens em um serviço externo (Cloudinary).
- O sistema deve utilizar stored procedures, views e functions no SQL Server para gerenciar dados.
- 

#### 4. Histórias de Usuário

- "Como dono de pets, eu quero criar uma conta para mim para que eu possa postar fotos dos meus gatos e compartilhá-las com outros usuários."
- "Como dono de pets, eu quero enviar fotos dos meus gatos com nome, peso, idade e descrição para que outros possam conhecer mais sobre eles."
- "Como dono de pets, eu quero visualizar um feed de fotos de outros gatos com scroll infinito para que eu possa me inspirar e interagir."
- "Como dono de pets, eu quero comentar nas fotos dos gatos de outros usuários para que eu possa interagir com eles."
- "Como dono de pets, eu quero receber sugestões de comentários geradas por IA para que eu possa interagir mais rapidamente."
- "Como dono de pets, eu quero visualizar o perfil de outro usuário para que eu possa ver todas as fotos dos seus gatos."
- "Como dono de pets, eu quero abrir os detalhes de uma foto para que eu possa ver suas informações completas e comentários."

#### 5. Plano de Desenvolvimento

- **Roadmap com as Etapas de Desenvolvimento:**
  1. **Semana 1:** Configuração do ambiente (React.js com SCSS, .NET, SQL Server, Cloudinary, Ollama).
  2. **Semana 2:** Desenvolvimento do back-end (modelos, autenticação, endpoints para fotos e comentários, stored procedures/views/functions).
  3. **Semana 3:** Integração com Cloudinary para upload de imagens e com Ollama para sugestões de IA.
  4. **Semana 4:** Desenvolvimento do front-end (páginas de login/cadastro, feed com scroll infinito).
  5. **Semana 5:** Implementação de perfis de usuário e detalhes de fotos no front-end.
  6. **Semana 6:** Testes e ajustes finais (responsividade, bugs, desempenho).

## 6. Conclusão e Próximos Passos

- **Conclusão:**

O CatSphere será uma aplicação funcional que permite aos donos de pets compartilhar fotos de seus gatos de forma inovadora, com integração de IA e uma interface atraente. O uso de SQL Server com stored procedures, views e functions, junto com SCSS para estilização, agrega valor técnico ao projeto, tornando-o ideal para estudo, demonstração de habilidades full-stack e uma aplicação real.

- **Melhorias Futuras:**

- Adicionar notificações para novos comentários ou curtidas.
- Implementar um sistema de curtidas nas fotos.
- Permitir edição de fotos diretamente no app (usando ferramentas do Cloudinary).
- Hospedar o sistema em nuvem (ex.: Azure) para uso real.