# Sistema de Gestão de Eventos

**Documentação do Projeto**

**Disciplina**: Sistemas Distribuídos

**Professor**: João Ferreira

**Instituição: Uninassau - Graças**

Equipe:

- Diego Lima (01401412) - Documentador

- Gilvanelson Nascimento (01395387) - Scrum Master

- Alesson Calaça (01378540) - Desenvolvedor Front-End

- Luiz Pereira (01170935) - Desenvolvedor Bac

# Índice

1. Introdução

2. Ferramentas e Tecnologias Utilizadas

3. Descrição do Sistema

4. Estrutura Analítica do Projeto

5. Cronograma do Projeto

6. Resumo de Gestão de Tarefas

7. Casos de Uso

8. Protótipos

9. Lições Aprendidas

10. Conclusão

# 1. Introdução

O presente documento descreve o projeto "Sistema de Gestão de Eventos", desenvolvido pela equipe da disciplina de Sistemas Distribuídos. O objetivo do projeto é criar uma API escalável e eficiente para gerenciar eventos de diferentes naturezas, oferecendo uma solução prática e integrada para instituições como ONGs, empresas e organizações religiosas.

# 2. Ferramentas e Tecnologias Utilizadas

As principais ferramentas e tecnologias utilizadas no desenvolvimento deste projeto foram:

- \*\*Figma\*\*: Design da interface do usuário.

- \*\*Visual Studio e .Net Core\*\*: Desenvolvimento do back-end.

- \*\*Create React App\*\*: Desenvolvimento do front-end.

- \*\*MySQL\*\*: Banco de dados.

- \*\*AWS/CloudFlare\*\*: Hospedagem e infraestrutura.

# 3. Descrição do Sistema

A API Mega Eventos é uma solução para gerenciar eventos de forma centralizada. Entre as funcionalidades principais estão: visualização de eventos, cadastro, inscrições, e notificações personalizadas. Focada na experiência do usuário, a solução utiliza comunicação em tempo real e relatórios para facilitar a administração.

# 4. Estrutura Analítica do Projeto

Abaixo está a análise das principais tarefas do projeto, divididas por fases e recursos:

1. Elaboração do Diagrama de Casos de Uso (Planejamento - Analista de Sistemas)

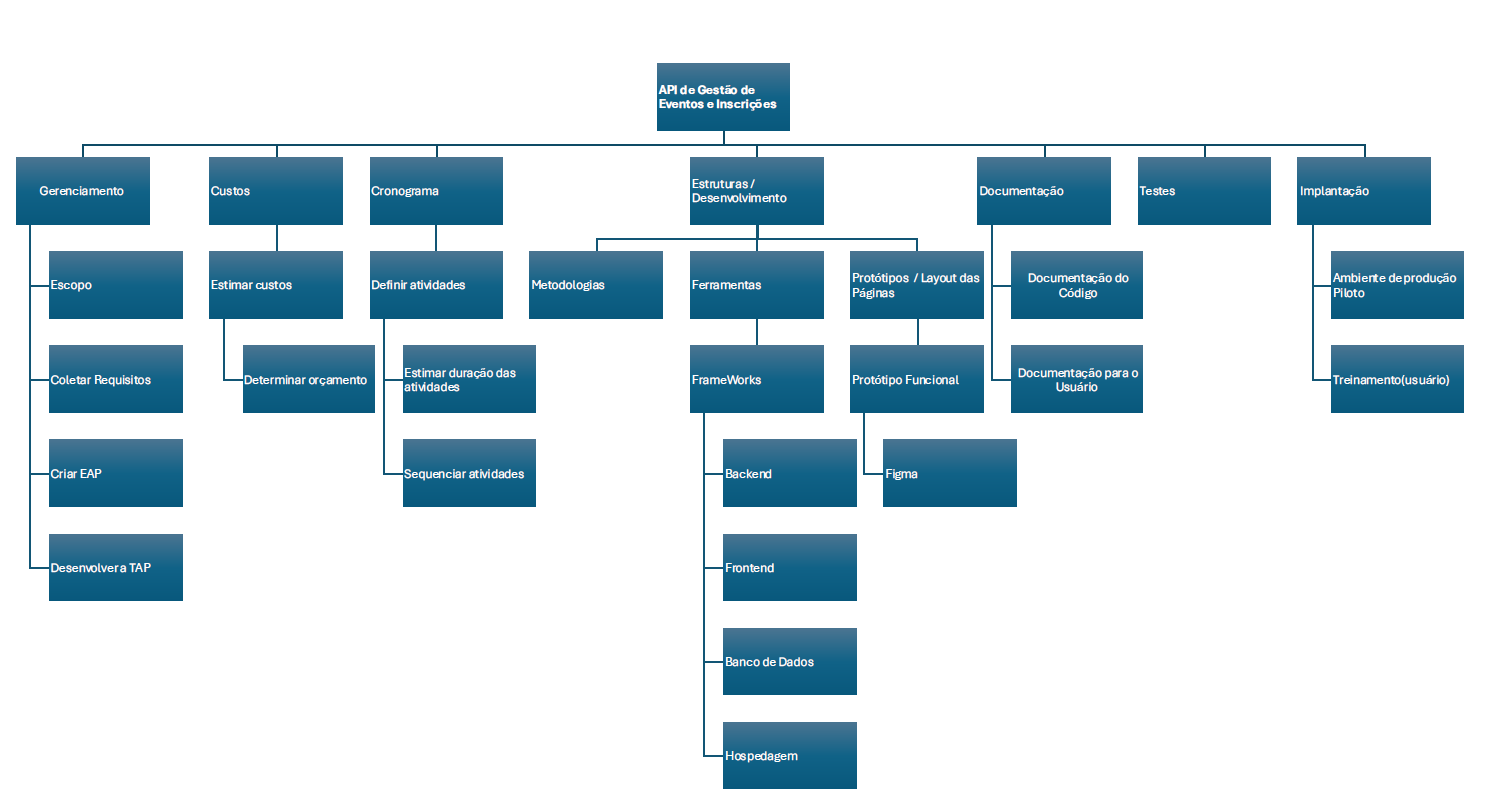
2. Design da Interface do Usuário (Design - Designer UX/UI)

3. Desenvolvimento do Back-End (Desenvolvimento - Desenvolvedor Back-End)

4. Desenvolvimento do Front-End (Desenvolvimento - Desenvolvedor Front-End)

5. Configuração do Banco de Dados (Implementação - DBA)

6. Hospedagem na AWS/CloudFlare (Implementação - Cloud Engineer)



# 5. Cronograma do Projeto

- Diagrama de Casos de Uso: Concluído em 15/09/2024

- Design da Interface: Em andamento, previsto para 30/09/2024

- Back-End: Em andamento, previsto para 01/11/2024

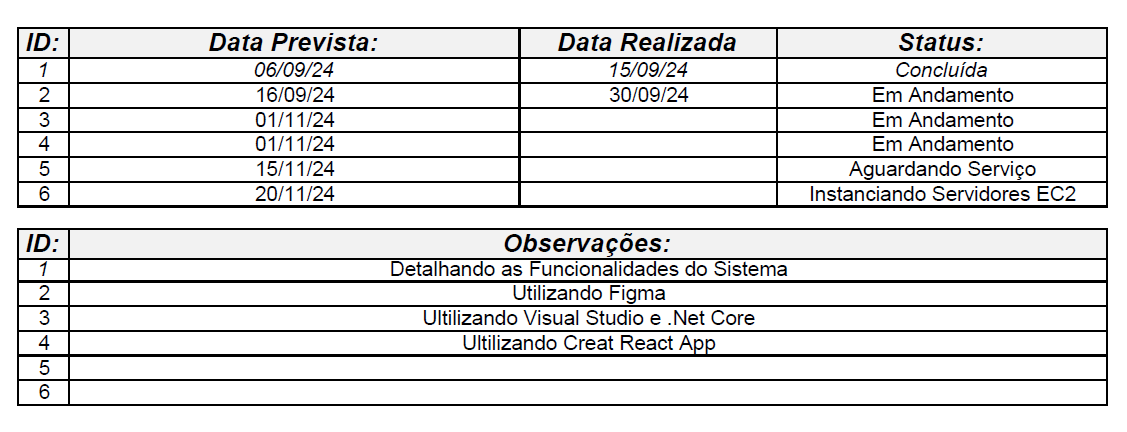
- Front-End: Em andamento, previsto para 01/11/2024

- Banco de Dados: Aguardando, previsto para 15/11/2024

- Hospedagem: Previsto para 20/11/2024

# 6.1 Análise de Custos

# 



# 6. Resumo de Gestão de Tarefas

O gerenciamento foi realizado via Trello para acompanhar o progresso e facilitar a comunicação. O repositório no GitHub permitiu colaboração e versionamento. Link: https://github.com/Diegodody/MegaEventApi

# 7. Casos de Uso

**UC722 - Visualizar Evento**  
**Ator Principal:** Usuário, Administrador  
**Objetivo:** Permitir a visualização dos detalhes de um evento específico.

### ****Pré-condições:****

* O evento deve estar cadastrado.
* O usuário/administrador deve estar autenticado (se necessário).

### ****Pós-condição:****

* Exibição de detalhes como nome, data, descrição e local do evento.



**UC721 - Cadastrar Evento**  
**Ator Principal:** Administrador  
**Objetivo:** Permitir o cadastro de novos eventos no sistema.

### ****Pré-condição:****

* O administrador deve estar autenticado.

### ****Pós-condição:****

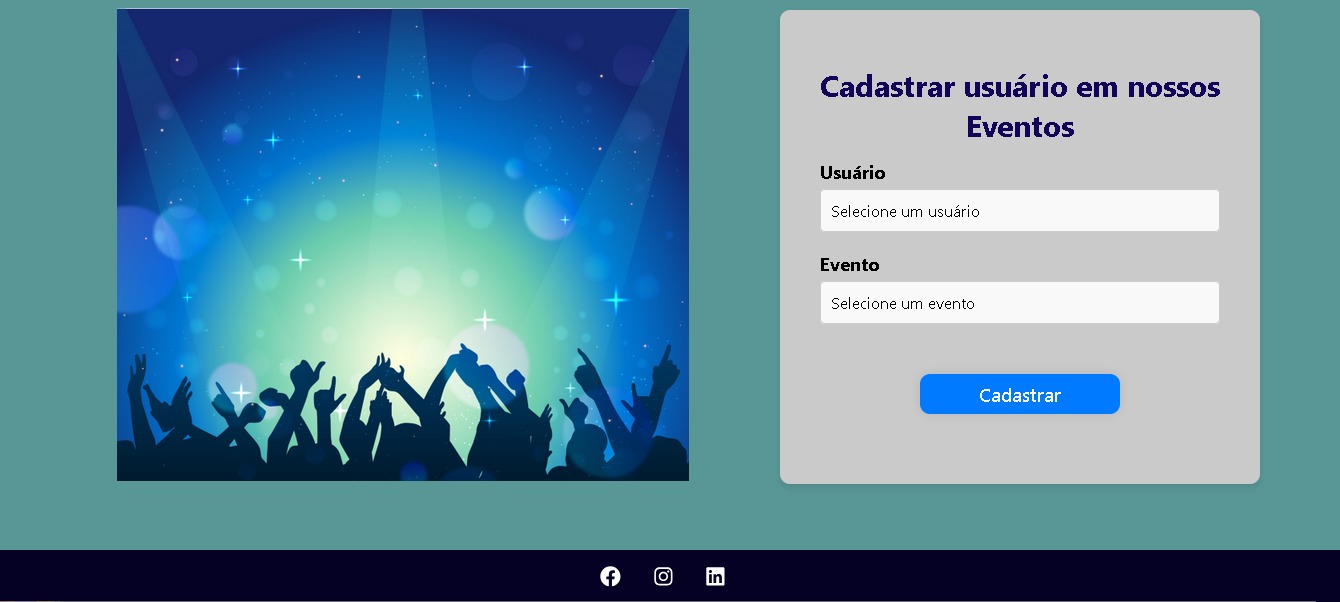
* O evento será salvo no banco de dados e exibido na lista de eventos.

# 

# 8. Protótipos

Os protótipos foram desenvolvidos no Figma, detalhando as interfaces do sistema e garantindo uma navegação intuitiva.









# 

# 9. Lições Aprendidas

Entre os aprendizados estão:

- A importância do planejamento inicial para reduzir retrabalho.

- Escolhas tecnológicas como React e C# garantiram eficiência e escalabilidade.

- Boas práticas de versionamento e CI/CD melhoraram a colaboração.

- Documentação clara alinhou expectativas.

# 10. Conclusão

Este projeto proporcionou uma experiência prática de desenvolvimento de sistemas distribuídos. O aprendizado acumulado reforça a importância de planejamento, colaboração e práticas técnicas sólidas para o sucesso de projetos futuros.