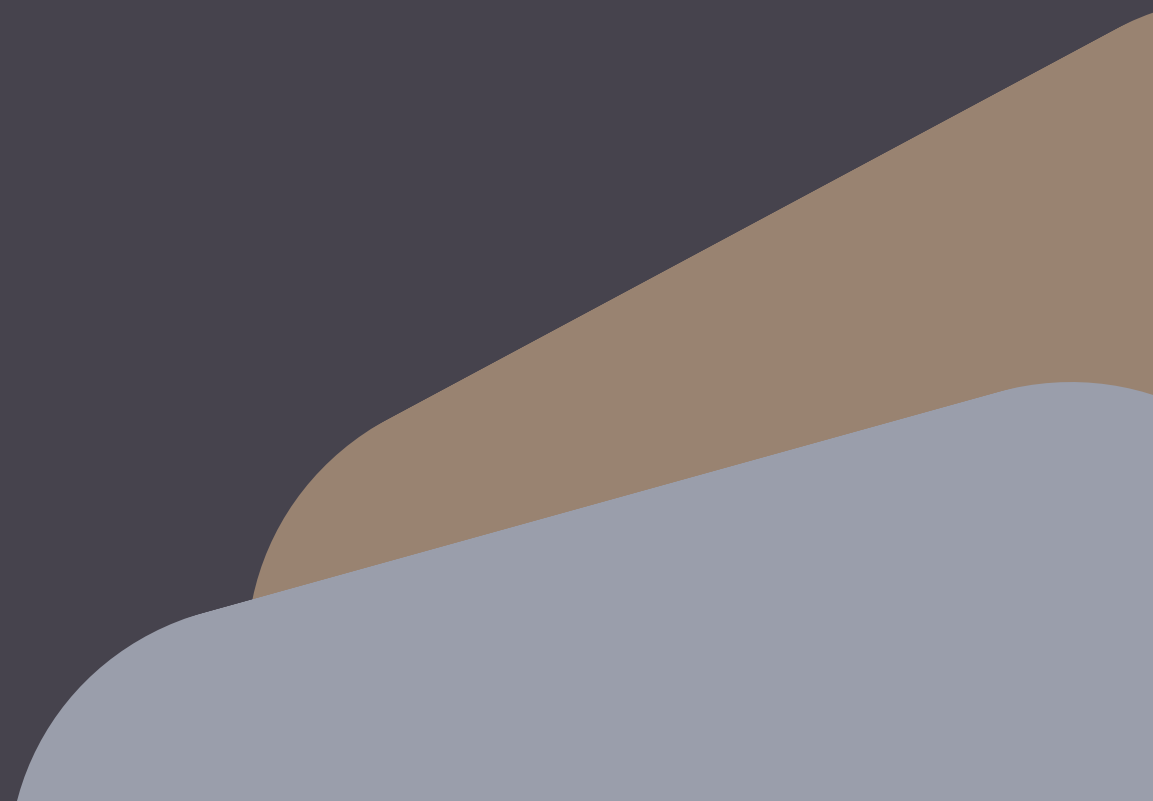
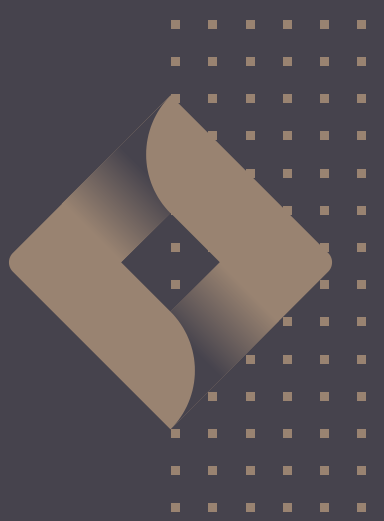


Programação de Soluções Computacionais

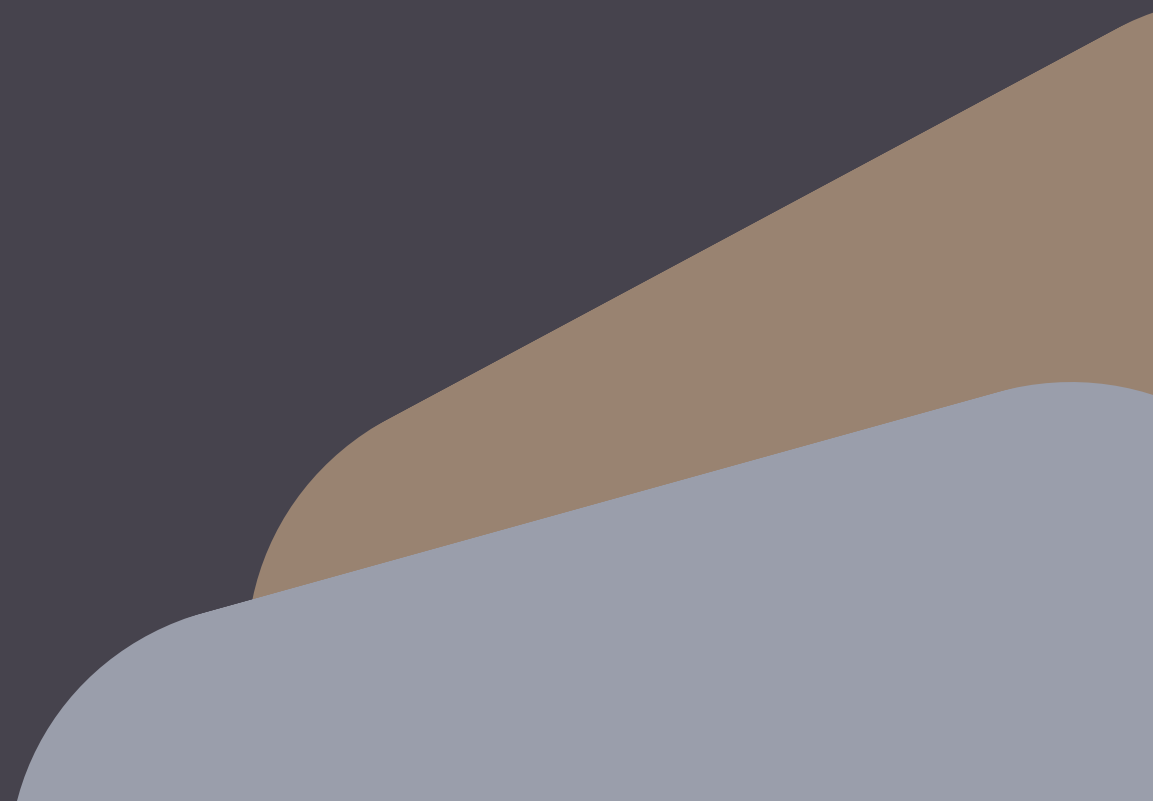
Software de Eventos





Nosso Time

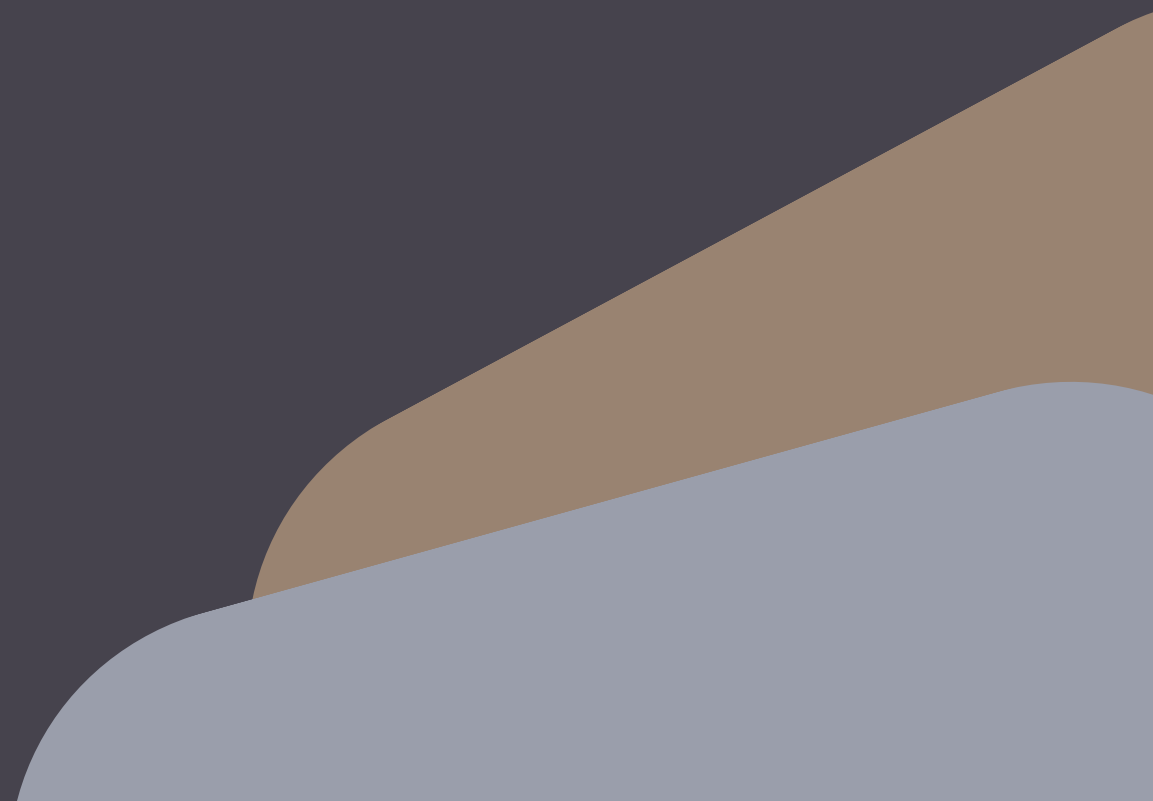

Vinicius Dias
Michel Cruz
Ricardo Souza





Introdução

O trabalho descreve a experiência de desenvolver um código de eventos em Python, empregando o paradigma de orientação a objetos e o banco de dados SQLite. O código permite criar e gerenciar eventos, abrangendo cadastros de usuários, empresas, locais, datas, horários e temas. Desenvolvido com o código estruturado em classes que incorporam princípios como encapsulamento, herança, polimorfismo e associação.



Desenvolvimento

O desenvolvimento do código de evento foi desafiador, demandando dedicação, pesquisa e colaboração do grupo. As principais dificuldades enfrentadas incluíram recuo incorreto, definição de variáveis de classe e uso indevido de expressões, que foram corrigidos com formatação automática, modificador "self" e revisão detalhada das expressões. A resolução desses desafios envolveu discussões em grupo, consulta a materiais de apoio e testes usando o depurador do Python.

Ao progredir no desenvolvimento, o grupo percebeu a necessidade de criar classes e métodos adicionais para aprimorar a funcionalidade do programa. Foram criadas classes para Usuário, Empresa, Local, Data, Horário e Tema, com métodos para cadastrar, listar, alterar e excluir elementos. Além disso, implementou-se o banco de dados SQLite, utilizando o módulo sqlite3 do Python e o DB Browser for SQLite. Foram criadas tabelas seguindo o modelo relacional, com campos e chaves primárias e estrangeiras, e comandos SQL foram empregados para manipulação de dados nas tabelas.



Conclusão

O desenvolvimento do código do sistema de eventos em Python foi uma experiência enriquecedora para o grupo, permitindo a aplicação de conceitos de programação orientada a objetos e banco de dados relacional. O resultado final foi um programa funcional que atende aos requisitos propostos, destacando-se pela eficiência na criação e gerenciamento de eventos. Embora reconheça a possibilidade de melhorias, como validação de dados, interface gráfica e relatórios, o grupo expressa a intenção de continuar aprimorando suas habilidades e explorar novas funcionalidades, demonstrando um compromisso contínuo com o estudo e crescimento na área de programação.

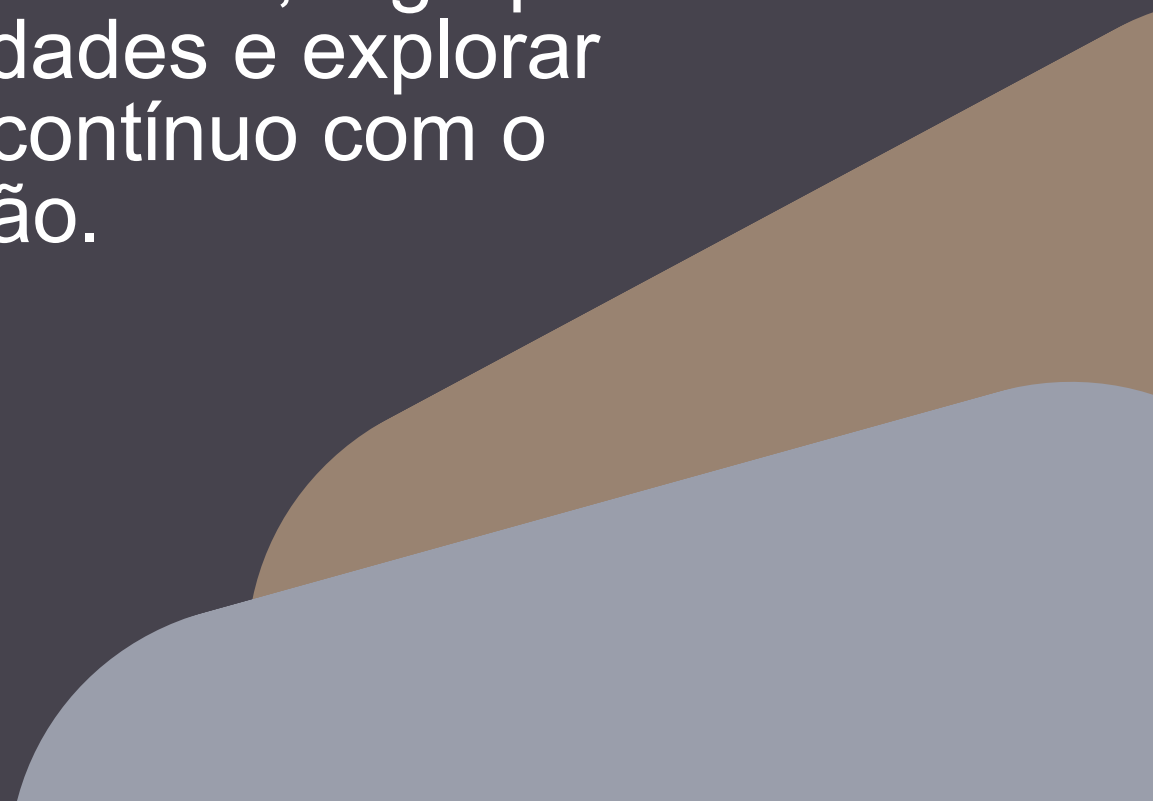

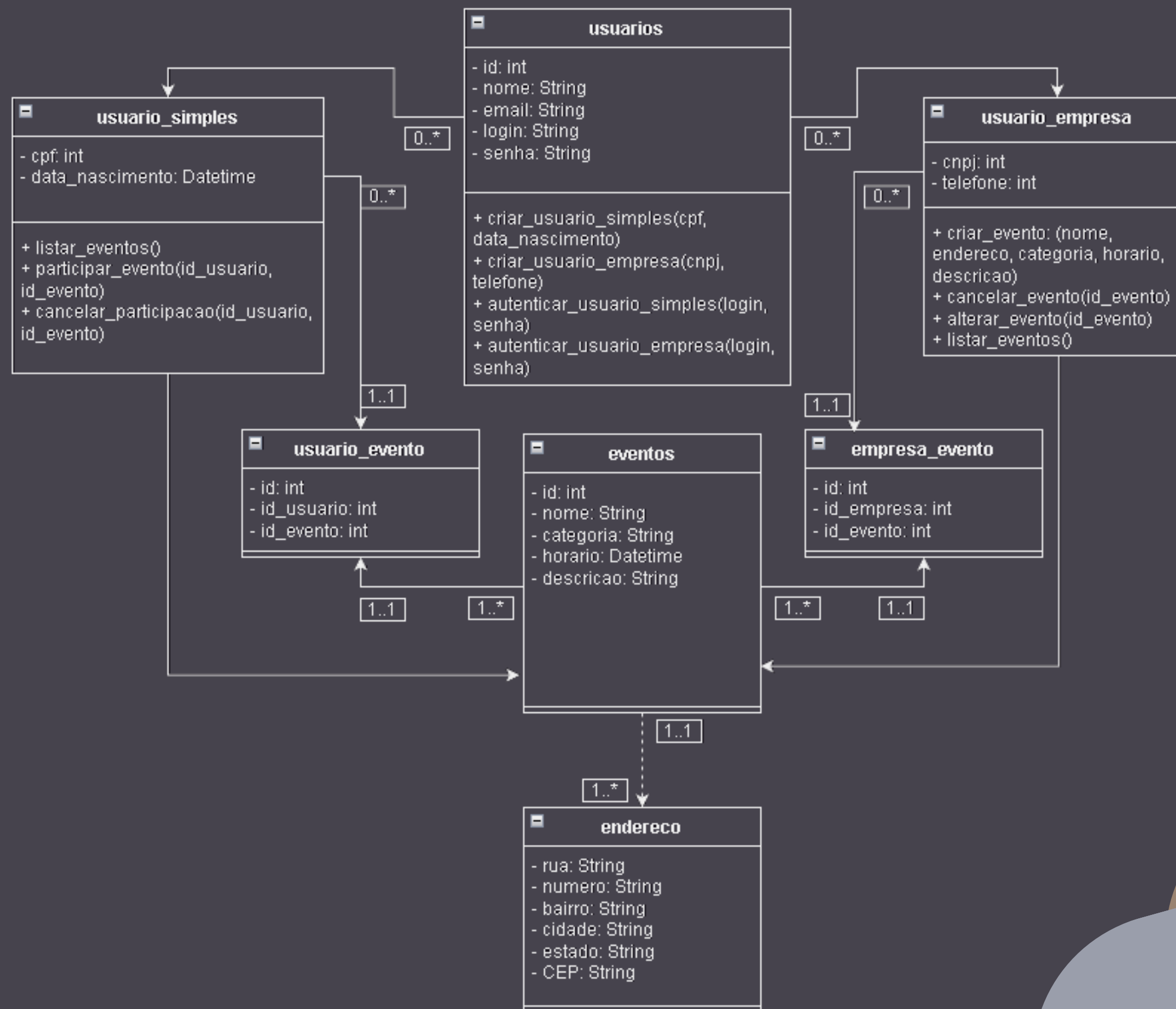


Diagrama de Classe





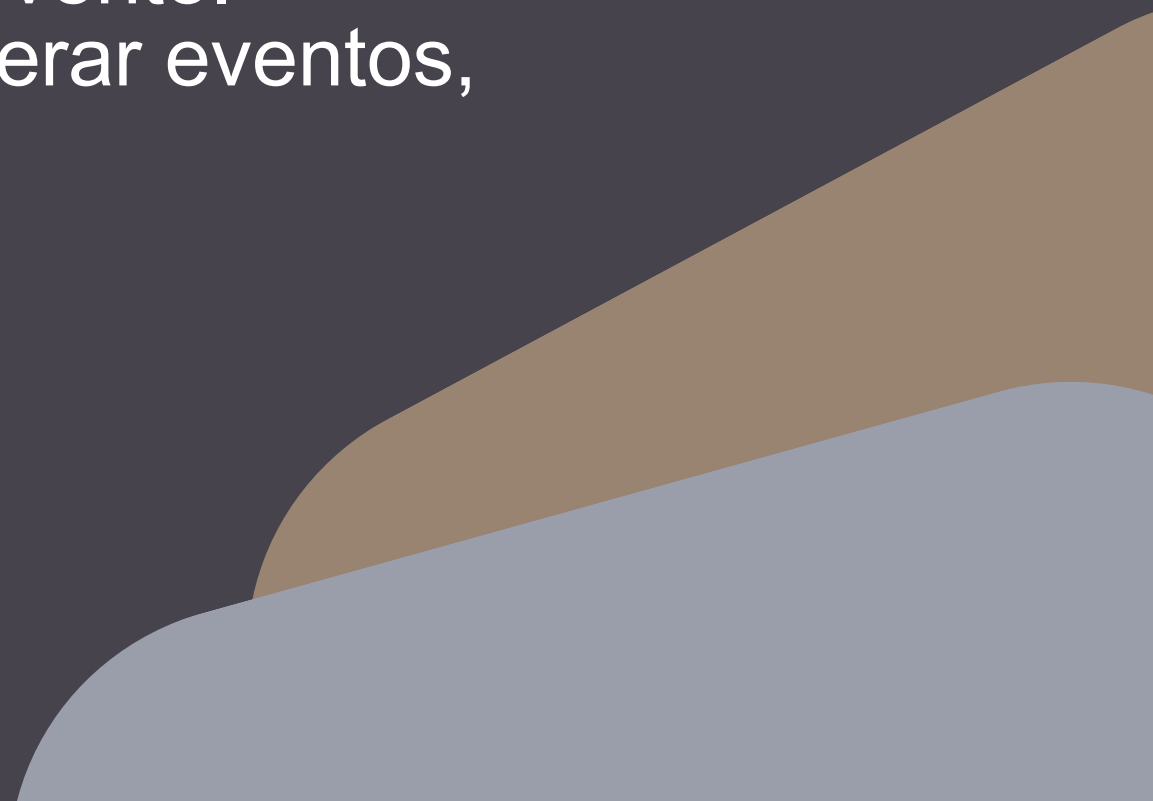

Requisitos Funcionais

RF01- O sistema deve permitir que os usuários se cadastrem fornecendo informações como nome, e-mail, idade, login, senha, no caso de empresa informar o CNPJ e telefone da empresa e os demais itens citados.

RF02 - Para execução do login, usuário deverá fornecer seu e-mail e senha.

RF03 - O usuário deve conseguir se desconectar, alterar cadastro da conta, cadastrar, participar, listar e cancelar participação em um evento.

RF04 - A empresa deve ser a única a poder cadastrar e alterar eventos, podendo também listar os clientes confirmados.





Requisitos Não Funcionais

RNF01 - Caso a operação esteja fora das opções ou informada incorretamente, aparecer mensagem de erro, caso certa, aparecer mensagem de sucesso.

RNF02 - O sistema deve conseguir suportar a demanda de usuários.

RNF03 - O sistema deve ser capaz de identificar o tipo do usuário.

RNF04 - O sistema deve ser fácil de usar e entender.

RNF05 - O sistema deve fornecer o menu diferente para o usuário e empresa.

RNF06 - O sistema deve ter diferença entre usuário e empresa e suas permissões.

RNF07 - O sistema deve prover meios de cadastrar, participar, listar, cancelar e alterar o evento.

RNF08 - O sistema deve possuir um banco de dados para armazenar as informações.

RNF09 - O sistema deverá possuir filtragem na busca por eventos.

