**Projeto: Agenda de Eventos com Painel de Administração**

**Funcionalidade geral:**

Um sistema onde os usuários podem ver eventos públicos (como workshops, shows, palestras, etc.) e os administradores podem cadastrar, editar e excluir eventos por um painel protegido.

**Frontend (React):**

* Página inicial com **lista de eventos futuros** (buscados da API).
* Filtros por **data**, **tipo de evento**, ou **localização**.
* Página de **detalhes do evento** (com informações completas).
* Área de **login** para administrador.
* Painel de administrador com:
  + Lista de eventos.
  + Formulário para **criar/editar/excluir** eventos.

**Backend (Node.js + Express):**

* API RESTful com rotas:
  + GET /events – lista eventos.
  + GET /events/:id – detalhe de um evento.
  + POST /events – criar evento (autenticado).
  + PUT /events/:id – editar evento (autenticado).
  + DELETE /events/:id – deletar evento (autenticado).
* **JWT Auth** para proteger rotas do painel admin.
* Uso de **MongoDB** para persistência dos dados.

**Extras opcionais (para expandir o projeto):**

* Upload de imagem para cada evento (usando multer).
* Integração com mapa (Google Maps API) para mostrar local dos eventos.
* Responsividade para mobile.
* Deploy com **Vercel (frontend)** e **Render/Heroku (backend)**.

**Funcionalidades Mínimas**

**Autenticação Admin (JWT):**

* Rota de login no backend que retorna token.
* Painel admin protegido no frontend via PrivateRoute.

**Eventos:**

* Listagem pública de eventos.
* Detalhes do evento com descrição, data, local.
* CRUD completo no painel do admin.

**Ordem de Desenvolvimento**

**Etapa 1: Finalizar Backend com segurança e organização**

Antes de integrar com o frontend, o backend precisa estar sólido.

**Tarefas:**

1. **Autenticação com JWT** (já feito!)
2. **Adicionar roles (admin, usuário)**
3. **Middleware de autorização (ex: só admin pode criar/editar eventos)**
4. **Testar todas as rotas com Postman (inclusão, edição, exclusão, listagem)**

**Etapa 2: Frontend - Login e controle de sessão**

Depois de garantir que o backend está 100%, vamos conectar com React.

**Tarefas:**

1. **Tela de Login em React**
2. **Armazenar token (localStorage)**
3. **Tela protegida: painel de administração (só acessa com token)**
4. **Tela pública: listar eventos (não precisa de token)**

**Etapa 3: Painel de Admin no React**

Aqui você cria o "CMS" (admin) para seu sistema de eventos.

**Tarefas:**

1. **Listar eventos**
2. **Criar novo evento**
3. **Editar evento**
4. **Deletar evento**
5. **Adicionar upload de imagem (opcional)**

**Etapa 4: Recursos extras e refinamentos**

Quando tudo estiver funcionando, você pode incrementar com:

* Filtros e busca por data, título ou local
* Interface visual para o calendário
* Upload de imagem (com Cloudinary ou Multer)
* Responsividade e design bonito

**Então, agora seguimos com:**

**Etapa 1: Adicionar roles e autorização no backend**

Quer que a gente implemente isso agora? Te passo o código completo de:

* Modelo de User com campo role
* Middleware para permitir apenas admin
* Exemplo de uso nas rotas