**Requisitos Funcionais:**

**Cadastro e Gerenciamento de Usuários**

* + **Registro de Usuários:** Permitir que discentes e docentes se registrem no sistema fornecendo informações básicas (nome, e-mail, senha, etc.).
  + **Perfil do Usuário:** Oferecer uma página de perfil onde os usuários possam visualizar e editar suas informações pessoais.

**Gerenciamento de Eventos**

* + **Criação e Gestão de Eventos:** Permitir que administradores do sistema (como professores ou organizadores) criem, editem e excluam eventos esportivos.
  + **Visualização de Eventos:** Exibir uma lista ou calendário com todos os eventos programados, permitindo que usuários visualizem detalhes e se inscrevam nos eventos.

**Iniciações Esportivas**

* + **Informações sobre Iniciações Esportivas:** Oferecer uma seção com informações sobre programas de iniciação esportiva, incluindo descrições, horários e requisitos para participação.
  + **Inscrição em Iniciações:** Permitir que usuários se inscrevam em programas de iniciação esportiva diretamente pelo sistema.

**Agendamento e Reserva de Quadras**

* + **Visualização da Disponibilidade:** Permitir que usuários visualizem a disponibilidade das quadras esportivas em um calendário.
  + **Reserva de Quadras:** Facilitar a reserva de horários para uso das quadras, com confirmações automáticas e cancelamentos.

**Conteúdo Educativo e Notícias**

* + **Artigos e Notícias:** Publicar artigos e notícias relacionadas a esportes.

**Interação e Comunidade**

* + **Fóruns e Discussões:** Implementar seções de fóruns ou discussões para que os usuários possam trocar informações, dicas e experiências sobre esportes.
  + **Comentários e Avaliações:** Permitir que usuários comentem e avaliem **Gerenciamento Administrativo**
  + **Painel Administrativo:** Fornecer um painel para administradores do sistema para gerenciar usuários, eventos, reservas e conteúdos.

**Notificações de Eventos:**

* Enviar notificações por e-mail ou dentro da plataforma para lembrar os usuários sobre eventos futuros, inscrições confirmadas e alterações nos eventos.
* Alertas de Reservas: Informar os usuários sobre a confirmação ou cancelamento de reservas de quadras, e alertas de horários de uso programados.

**Acessibilidade e Usabilidade**

* + **Design Responsivo:** Garantir que o sistema seja acessível em dispositivos móveis e desktops, com um design responsivo e amigável.
  + **Facilidade de Navegação:** Proporcionar uma navegação intuitiva e fácil para todos os usuários, com menus e botões bem definidos.

**Requisitos Não Funcionais:**

**Desempenho**

* + **Tempo de Resposta:** O sistema deve carregar páginas e responder a interações do usuário em menos de 2 segundos para manter uma experiência fluida. (É possível usar lazy loading e compressão de imagens)
  + **Capacidade de Carga:** Suportar até [A DEFINIR] usuários simultâneos sem degradação perceptível do desempenho.

### **Segurança**

* + **Criptografia:** Todos os dados sensíveis, como senhas, devem ser armazenados usando criptografia forte (por exemplo, bcrypt).
  + **Autenticação e Autorização:** Implementar autenticação segura para proteger as contas dos usuários e garantir que apenas usuários autorizados tenham acesso a funções específicas.
  + **Proteção contra Ataques:** Implementar medidas de segurança contra ataques comuns, como injeção de SQL, XSS (Cross-Site Scripting), e CSRF (Cross-Site Request Forgery).
  + **Backup de Dados:** O sistema deve realizar backups regulares dos dados para evitar perda em caso de falha.

### **Usabilidade**

* + **Intuitividade:** A interface do usuário deve ser fácil de entender e usar, mesmo para pessoas com pouca experiência em tecnologia.
  + **Acessibilidade:** O sistema deve estar em conformidade com os padrões de acessibilidade (por exemplo, WCAG 2.1) para garantir que seja utilizável por pessoas com deficiências.

### **Compatibilidade**

* + **Cross-Browser:** O sistema deve ser compatível com os principais navegadores modernos (Chrome, Firefox, Edge, Safari).
  + **Multi-Dispositivo:** O sistema deve ser acessível e ter boa performance em diferentes tipos de dispositivos, incluindo desktops, laptops, tablets, e smartphones.

### **Confiabilidade**

* + **Resiliência:** O sistema deve ser capaz de se recuperar automaticamente de falhas menores e retomar a operação sem intervenção manual.

### **Escalabilidade**

* + **Escalabilidade Horizontal:** A arquitetura do sistema deve suportar a adição de mais servidores para lidar com o aumento de carga sem a necessidade de modificações significativas.
  + **Escalabilidade Vertical**: A arquitetura do sistema deve permitir a adição de mais recursos (como CPU, memória e armazenamento) a um único servidor para melhorar o desempenho sem alterar a estrutura existente.

### **Manutenibilidade**

* + **Documentação:** O código deve ser bem documentado para facilitar a manutenção e futuras atualizações.
  + **Modularidade:** O sistema deve ser construído com uma arquitetura modular para facilitar a manutenção, atualização e extensão de funcionalidades.

### **Portabilidade**

* + **Ambiente de Desenvolvimento e Produção:** O sistema deve ser facilmente migrável entre ambientes de desenvolvimento, teste e produção.

### **Regulamentação**

* + **Conformidade com LGPD/GDPR:** O sistema deve cumprir todas as regulamentações de proteção de dados relevantes, como a Lei Geral de Proteção de Dados (LGPD) no Brasil.

### **Localização**

* + **Formatação Local:** O sistema deve adaptar formatação de data e hora de acordo com a localidade do usuário.

**Segurança e Privacidade**

* + **Proteção de Dados:** Garantir que os dados dos usuários sejam armazenados e transmitidos de forma segura, conforme as normas de proteção de dados pessoais.
  + **Controle de Acesso:** Assegurar que diferentes tipos de usuários (discentes, docentes, administradores) tenham acesso apenas às funcionalidades apropriadas.