

# Documento de requisitos - FitTrack

Versão 1.2

Alunos:

- Murilo Maciel Rodrigues
- Felipe Oliveira Raimundo
- Paulo Antônio Bastos de Farias

## Sumário

<b>Introdução.....</b>	<b>2</b>
Definições e Abreviações.....	2
<b>Usuários Identificados.....</b>	<b>2</b>
<b>Requisitos Funcionais.....</b>	<b>3</b>
Usuário.....	3
Aluno.....	3
Professor de Educação Física.....	3
Dono de Academia.....	4
Administrador.....	4
Equipe de Desenvolvimento/QA.....	5
<b>Requisitos Não-Funcionais.....</b>	<b>5</b>
Disponibilidade.....	5
Privacidade e segurança.....	6
Usabilidade.....	6
Suportabilidade.....	6
Interoperabilidade.....	7
Manutenibilidade.....	7
Desempenho.....	7
Implementação.....	7
Implantação.....	8
<b>Wireframes.....</b>	<b>8</b>
Tela de Login.....	8
Tela de Ajustes de Perfil.....	9
Tela de Cadastro de Ficha do Professor.....	9
Tela de Dashboard do Aluno.....	10
Tela de Gestão de Usuários.....	10
Tela de Dashboard do Dono de Academia.....	11

## Histórico de revisões

Data	Versão	Descrição
05/11/2025	1.0	Criação do documento
06/11/2025	1.1	Adição de requisitos funcionais
07/11/2025	1.2	Adição de requisitos não-funcionais
09/11/2025	1.3	Adição de wireframes
13/11/2025	1.4	Revisão dos requisitos
15/11/2025	1.5	Adição de mais wireframes

## Introdução

O objetivo deste documento é apresentar os requisitos de software do produto FitTrack.

## Definições e Abreviações

Definições de todos os termos, acrônimos e abreviações necessárias à adequada interpretação do Documento de Requisitos.

- RFU - Requisito Funcional de Usuário do Sistema
- RFA – Requisito Funcional de Aluno
- RFP – Requisito Funcional de Professor
- RFD – Requisito Funcional de Dono de academia
- RFAd – Requisito Funcional de Administrador do Sistema
- RFQA - Requisito Funcional da equipe de QA/Desenvolvimento
- RNF – Requisito Não-Funcional

## Usuários Identificados

- Usuário do Sistema
  - Usuário Comum
    1. Aluno
    2. Professor de Educação Física
    3. Dono de Academia
  - Administrador de Sistema

- Equipe de Desenvolvimento / QA
- Responsável pela Segurança

## Requisitos Funcionais

### Usuário

[RFU001] Como usuário, gostaria de realizar login e logout de forma segura, para proteger meus dados e minha conta.

[RFU002] Como usuário, gostaria de poder recuperar minha senha por meio do e-mail, para que eu possa acessar novamente o sistema em caso de esquecimento.

[RFU003] Como usuário, gostaria que o sistema oferecesse autenticação de dois fatores (2FA), para aumentar a segurança do meu acesso.

[RFU004]. Como usuário, gostaria de poder curtir e comentar publicações, para interagir com outros membros da comunidade.

[RFU005] Como usuário, gostaria de poder seguir outros usuários, para acompanhar suas atualizações e progressos.

[RFU006] Como usuário, gostaria de configurar lembretes personalizados, para gerenciar melhor minha rotina de treinos.

[RFU007] Como usuário, gostaria de realizar cadastro no sistema.

### Aluno

[RFA001] Como aluno, gostaria de visualizar e registrar meus treinos concluídos, informando cargas, repetições e observações, para acompanhar meu desempenho.

[RFA002] Como aluno, gostaria que cada treino concluído pudesse gerar uma publicação no meu mural, desde que eu permita, para compartilhar meu progresso com outros usuários.

[RFA003]. Como aluno, gostaria que o mural exibisse meu histórico de evolução, funcionando como uma linha do tempo fitness.

[RFA004] Como aluno, gostaria de comparar meus resultados atuais com períodos anteriores, para avaliar minha evolução física.

### Professor de Educação Física

[RFP001] Como professor, gostaria de poder criar fichas de treino personalizadas para cada aluno, para adaptar os exercícios às necessidades individuais.

[RFP002] Como professor, gostaria de editar e atualizar as fichas de treino existentes, para refletir a evolução do aluno.

[RFP003] Como professor, gostaria de exportar relatórios de progresso dos alunos, para realizar acompanhamentos personalizados e prestar feedbacks.

[RFP004] Como professor, gostaria de enviar mensagens de feedback aos alunos, para manter a comunicação e o acompanhamento personalizado.

## Dono de Academia

[RFD001] Como dono de academia, gostaria que o sistema permitisse a criação de grupos fechados de academias, para postagens e interações restritas apenas aos membros da academia.

[RFD002] Como dono de academia, gostaria de gerenciar o cadastro de alunos e professores, para manter o controle das pessoas vinculadas à academia.

[RFD003] Como dono de academia, gostaria de visualizar métricas gerais da academia, como número de alunos ativos e nível de engajamento, para acompanhar o desempenho do negócio.

## Administrador

[RFAd001] Como administrador de sistema, gostaria de realizar backup e restauração de dados, para garantir a disponibilidade das informações.

[RFAd002] Como administrador de sistema, gostaria de monitorar o status do servidor e desempenho do sistema, para prevenir falhas e quedas.

[RFAd003] Como administrador de sistema, gostaria de atualizar e implantar novas versões da aplicação, para manter o sistema seguro e estável.

[RFAd004] Como administrador, gostaria que o sistema armazenasse registros de desempenho e estatísticas personalizadas, para futuras análises e relatórios.

[RFAd005] Como administrador, gostaria que o sistema enviasse notificações automáticas de treino (ex: "Hoje é dia de treino de pernas"), para incentivar a constância do aluno.

[RFD006] Como administrador de sistema, gostaria de gerenciar permissões de usuários, para controlar o acesso a diferentes áreas do sistema.

## Equipe de Desenvolvimento/QA

[RFQA001] Como equipe de Desenvolvimento/QA, gostaria de ter acesso a um ambiente de testes isolado, para validar novas funcionalidades antes de liberar ao público.

[RFQA002] Como equipe de Desenvolvimento/QA, gostaria de registrar bugs e não conformidades diretamente no sistema, para rastrear correções de forma eficiente.

[RFQA003] Como equipe de Desenvolvimento/QA, gostaria de executar testes automatizados, para garantir que novas versões não quebrem funcionalidades existentes.

## Requisitos Não-Funcionais

### Disponibilidade

[RNF001] O sistema deverá operar de forma contínua. Durante pelo menos 98% do tempo, salvo em manutenção técnica e problemas com o servidor. O mesmo se aplica à sua disponibilidade e transparência para com assuntos legais e ordens jurídicas.

[RNF002] Quando funcionando, o sistema deve garantir a disponibilidade dos dados no horário que o usuário requisitar

[RNF003] O sistema deve funcionar parcialmente em modo offline, mantendo informações básicas em cache local, como treinos previamente baixados, e sincronizando os dados automaticamente quando a conexão com a internet for restabelecida.

[RNF004] O sistema deverá ser compatível com dispositivos móveis e computadores que atendam aos requisitos mínimos de processamento e memória estabelecidos na documentação técnica.

[RNF005] O sistema deverá ser compatível com a plataforma em que for implementado.

[RNF006] O sistema deve armazenar dados em seus servidores, garantindo disponibilidade contínua.

### Privacidade e segurança

[RNF007] O sistema deve contar com um mecanismo robusto de criptografia para proteger dados sensíveis dos usuários, como senhas, informações pessoais e credenciais de acesso.

[RNF008] O sistema deve ser desenvolvido de forma que todos os seus módulos, serviços, políticas de retenção e tratamento de dados estejam plenamente alinhados e em conformidade com a Lei Geral de Proteção de Dados (LGPD - Lei nº 13.709/2018) e demais legislações brasileiras aplicáveis.

[RNF009] O sistema deve registrar logs detalhados de acesso e atividade, abrangendo eventos como autenticações, modificações de dados e operações administrativas. Esses registros devem incluir informações como data, hora, usuário responsável e tipo de ação executada, permitindo auditoria, rastreabilidade e monitoramento de incidentes, sempre respeitando os princípios de privacidade e confidencialidade.

[RNF010] O sistema deve permitir controle de permissões e papéis de usuário, possibilitando a definição de diferentes níveis de acesso conforme funções, responsabilidades e necessidades operacionais.

[RNF011] O sistema deve realizar backups automáticos e periódicos dos dados, assegurando a integridade e a disponibilidade das informações armazenadas.

## Usabilidade

[RNF012] O sistema deve ser desenvolvido de forma que futuramente possa estar disponível nos idiomas Português, Inglês e Espanhol. Isso inclui a tradução de interfaces, mensagens de erro, instruções e conteúdos dinâmicos do sistema.

[RNF013] O sistema deve possuir uma interface intuitiva, acessível e responsiva, proporcionando uma experiência de uso fluida e agradável nos diferentes dispositivos. A interface deve ser projetada de forma a garantir usabilidade em diversos tamanhos de tela e resoluções.

[RNF014] O sistema deve oferecer acessibilidade para usuários com deficiências visuais, motoras e outras necessidades especiais, seguindo boas práticas e diretrizes de design inclusivo.

[RNF015] O sistema deve garantir uma navegação fluida e sem esforço para os usuários, minimizando a quantidade de etapas necessárias para realizar tarefas comuns. A interface deve ser consistente em todas as suas páginas e módulos, mantendo padrões de design.

## Suportabilidade

[RNF016] O sistema deve estar disponível para ser acessado por navegadores web modernos (Chrome, Firefox, Edge, Safari)

[RNF017] O sistema deve estar disponível para ser acessado via aplicativo mobile para celulares com sistema Android ou iOS.

[RNF018] O sistema deve ser desenvolvido de forma que não consuma muitos recursos na versão mobile.

## Interoperabilidade

[RNF019] O sistema deve ser desenvolvido de forma que possa utilizar APIs externas, como por exemplo: a API do ViaCEP para cadastro de endereços.

[RNF020] O sistema deve ser capaz de se integrar com plataformas de pagamento externas, como PayPal, Stripe, PagSeguro, ou MercadoPago, para realizar pagamentos de mensalidades.

## Manutenibilidade

[RNF021] O sistema deve ter código-fonte que segue padrões de nomenclatura e boas práticas recomendadas pela linguagem utilizada, facilitando leitura e manutenção, tanto no código, quanto comentários e commits no git.

[RNF022] O sistema deve ter documentação técnica, que deve ser mantida atualizada sempre que houver mudanças significativas na estrutura do sistema.

[RNF023] A estrutura de pastas do projeto deve seguir uma organização lógica e documentada para facilitar a localização de arquivos.

## Desempenho

[RNF024] O tempo de resposta para carregamento de páginas e funcionalidades principais (login, visualização de treinos e mural) não deve exceder 3 segundos em condições normais de rede.

[RNF025] O sistema deve ser capaz de suportar pelo menos 150 usuários acessando simultaneamente sem degradação perceptível de desempenho.

## Implementação

[RNF026] A autenticação deve usar padrões seguros, como por exemplo: o JWT, que garante segurança no login dos usuários. Dentre outros padrões.

[RNF027] A comunicação entre frontend e backend deve ser feita via API REST, garantindo integração e escalabilidade.

## Implantação

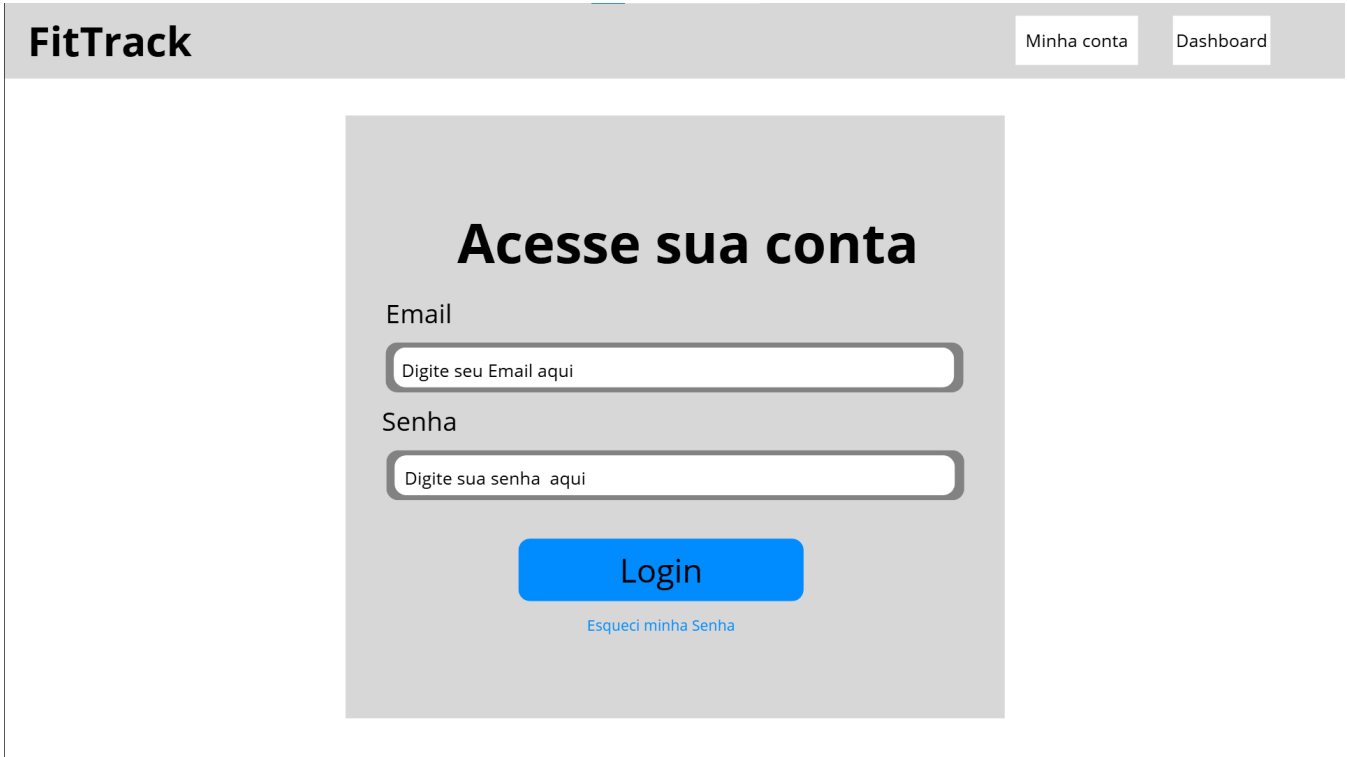
[RNF028] O sistema deve ser desenvolvido de forma que ele possa ser implantado em ambiente de nuvem, como AWS, Azure ou Google Cloud.

[RNF029] A atualização do sistema deve ocorrer sem necessidade de interromper totalmente o serviço.

[RNF030] O ambiente de produção deve possuir monitoramento contínuo, incluindo logs de erro e alertas de funcionamento.

# Wireframes

## Tela de Login



## Tela de Ajustes de Perfil





FitTrack

Minha conta

Dashboard

Alunos

João Silva

Felipe Oliveira

Paulo Antônio

Ficha de Treino do Aluno

Dia do Treino

Dia A / Segunda

Grupo Muscular

Peito

Selecionar Aluno

João

Exercício	Séries	Repetições	Carga (kg)	Comentário
Supino Reto	3	10	10	Halteres
Supino Declinado	3	12	20	Barra
Cross Over	3	12	20	Barra

Adicionar Exercício

FitTrack

Minha conta

Dashboard

Navegação

Minha Ficha

Meu Perfil

Mural

Minha Ficha

Dia do Treino

Dia A / Segunda

Exercício	Séries	Repetições	Carga
Supino Reto	3	10	10
Supino Declinado	3	12	20
Cross Over	3	12	20

Evolução

The line graph shows the progression of training volume across five exercises. The y-axis represents volume from 0 to 40. The x-axis lists the exercises: Supino Reto, Supino Declinado, Cross Over, Pullover, and Crucifixo. The volume starts at approximately 18 for Supino Reto, rises to 26 for Supino Declinado, dips slightly to 23 for Cross Over, then jumps to 35 for Pullover and remains at 35 for Crucifixo.

Exercício	Volume
Supino Reto	18
Supino Declinado	26
Cross Over	23
Pullover	35
Crucifixo	35

# Tela de Gestão de Usuários

FitTrack

Minha contaDashboard

Filtros

Tipo

Todos

Professores

Alunos

Dono de academia

Academia

Digite o nome...

Gestão de Usuários

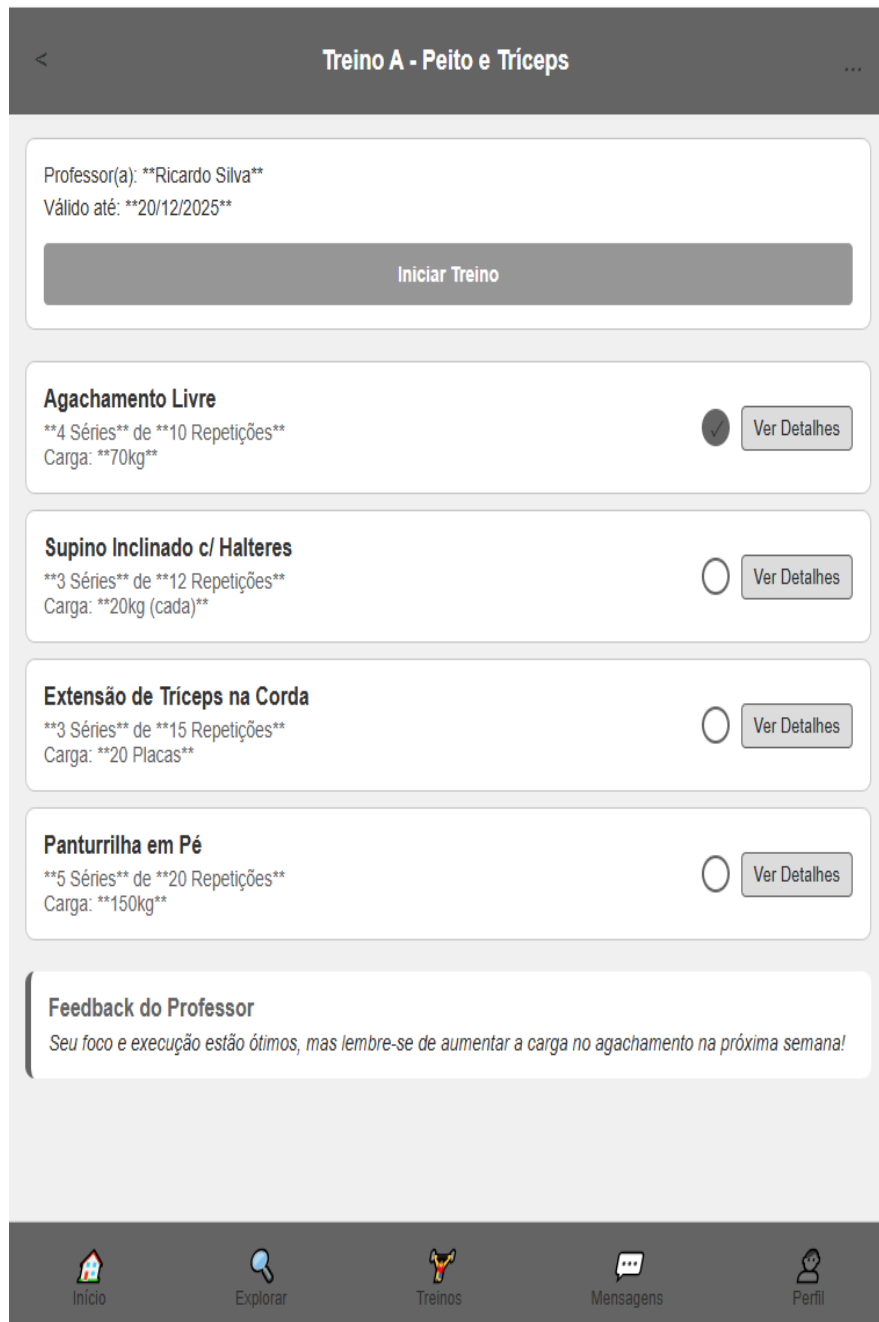
Nome	Email	Tipo	Status
João Silva	João@gmail.com	Aluno	Ativo
Carlos Eduardo	Carlos@gmail.com	Professor	Ativo
Roberto Gomes	Roberto@gmail.com	Dono de Academia	Ativo
Joana Silva	Joana@gmail.com	Aluno	Inativo
Valéria Oliveira	Valeria@gmail.com	Professor	Ativo
Lucas Maciel	Lucas@gmail.com	Dono	Ativo

Novo Usuário

Editar Usuário

Excluir Usuário

# Tela de treino do Aluno



Tela de rede social do Aluno



Usuario\_Exemplo1



Início



Explorar



Adicionar Post



Mensagens



Perfil

---

**Validado por:**

Felipe Oliveira Raimundo | Engenheiro de Software

[raimundo.felipe@academico.ifpb.edu.br](mailto:raimundo.felipe@academico.ifpb.edu.br)

IFPB - Av. Primeiro de Maio, 720, João Pessoa, Paraíba

Murilo Maciel Rodrigues | Engenheiro de Software

[murilo.maci@academico.ifpb.edu.br](mailto:murilo.maci@academico.ifpb.edu.br)

IFPB - Av. Primeiro de Maio, 720, João Pessoa, Paraíba

Paulo Antônio Bastos de Farias | Engenheiro de Software

[antonio.bastos@academico.ifpb.edu.br](mailto:antonio.bastos@academico.ifpb.edu.br)

IFPB - Av. Primeiro de Maio, 720, João Pessoa, Paraíba