



Escopo do Projeto Integrador (Backend) - Desafio 3

Grupo:

Camila Lima
João Victor Pereira Andrade
Karina Camargo Araujo
Luan Lazarine
Pamela Oliveira Silva
Vitor Gabriel Teles de Lima

Data:

06/05/2025

Projeto-Verão - Sistema de Gerenciamento de Treinos!

A aplicação foi idealizada para pessoas que desejam organizar seus treinos, acompanhar sua evolução e manter uma vida saudável de forma prática. O objetivo é centralizar todas as informações sobre rotina de exercícios, histórico de treinos, medidas corporais e objetivos, promovendo disciplina, motivação e constância.

Descrição Geral

O Projeto-Verão é um aplicativo de Gerenciamento de Treinos e tem como propósito oferecer uma solução prática e eficiente para pessoas que desejam organizar sua rotina de treinos, acompanhar a evolução física e manter hábitos saudáveis. A plataforma permite o registro, visualização, atualização e remoção de treinos, dados físicos de forma simples e estruturada.

A aplicação foi idealizada especialmente para usuários individuais, personal trainers e academias de pequeno a médio porte, visando proporcionar uma gestão mais personalizada e eficiente da vida fitness. O objetivo é facilitar o controle da rotina de exercícios, melhorar a

adesão aos planos de treino e promover mais engajamento com a saúde física e mental, por meio de um sistema intuitivo, seguro e adaptável às necessidades de cada usuário.

Público-alvo:

O público-alvo deste sistema são pessoas que praticam atividades físicas regularmente, iniciantes na academia, personal trainers e qualquer indivíduo que deseja melhorar sua qualidade de vida por meio do controle da sua rotina fitness.

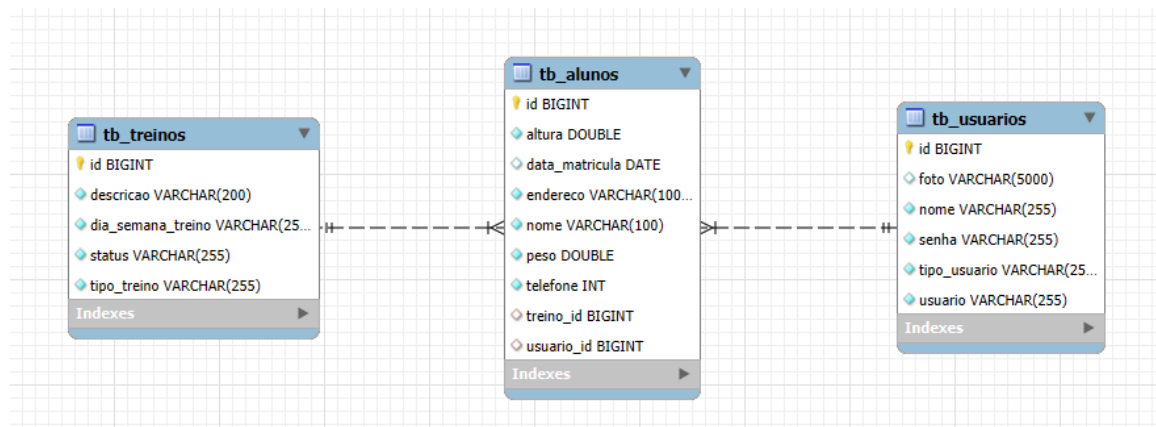
Problema que o projeto visa resolver:

O Projeto-Verão foi desenvolvido para solucionar diversos desafios enfrentados por pessoas que desejam manter uma rotina saudável e organizada no que diz respeito à prática de exercícios físicos, nutrição e bem-estar. Dentre as principais soluções oferecidas, destacam-se:

- Centralização e organização de dados pessoais de treino: substitui anotações manuais, planilhas ou aplicativos genéricos por um sistema completo e estruturado, com fácil acesso ao histórico de treinos, metas e registros físicos.
- Agilidade na rotina fitness: facilita a visualização, edição e acompanhamento de treinos, tornando mais prática a rotina de quem treina por conta própria ou com acompanhamento profissional.
- Segurança e confidencialidade: protege os dados pessoais e físicos dos usuários, com acesso restrito e seguro, respeitando a privacidade dos registros de saúde e progresso.
- Facilidade de atualização e acompanhamento da evolução: permite ajustes rápidos em planos de treino, metas e medidas corporais, promovendo um acompanhamento constante e adaptável.
- Suporte à evolução e performance: fornece recursos para que usuários e profissionais acompanhem resultados e adaptem os planos com base na performance individual, promovendo ganhos consistentes e personalizados.

Dessa forma, o Projeto-Verão moderniza e otimiza a experiência de gerenciamento da vida fitness, promovendo mais autonomia, motivação e eficiência no alcance dos objetivos de saúde e bem-estar.

Entidade e Atributos da Entidade



Banco de dados: db_app_fitness_backend

Entidade Principal: Aluno (tb_alunos)

Atributos:

- **id**: Identificador único do aluno
- **nome**: Nome completo do aluno
- **endereco**: Endereço do aluno
- **telefone**: Telefone para contato
- **data_matricula**: Data da matrícula do aluno
- **altura**: Altura do aluno
- **peso**: Peso do aluno
- **treino_id**: Referência ao treino associado ao aluno (relacionamento com tb_treinos)
- **usuario_id**: Referência ao usuário associado ao aluno (relacionamento com tb_usuarios)

Entidade: Usuário (tb_usuarios)

Atributos:

- **id**: Identificador único do usuário
- **nome**: Nome do usuário do sistema
- **usuario**: Nome de login do usuário
- **senha**: Senha de acesso
- **foto**: Foto do usuário
- **tipo_usuario**: Tipo de usuário (ex: administrador, instrutor, aluno)

Entidade: Treino (tb_treinos)

Atributos:

- **id:** Identificador único do treino
- **descricao:** Descrição do treino
- **tipo_treino:** Tipo do treino (ex: hipertrofia, resistência, emagrecimento)
- **dia_semana_treino:** Dias da semana em que o treino ocorre
- **status:** Status do treino (ativo, inativo, concluído)

Funcionalidades Principais (CRUD)

O sistema contará com funcionalidades básicas de CRUD (Create, Read, Update, Delete) para as entidades Usuários, Alunos e Treinos, permitindo o gerenciamento completo das informações cadastradas.

Entidade Usuário:

- **Criar Usuário:** Cadastrar um novo usuário com dados como nome, login, senha, tipo de usuário e foto.
- **Autenticar:** Validar o acesso do usuário por meio de autenticação com login e senha.
- **Listar Usuários:** Exibir todos os usuários cadastrados, com filtros por nome ou ID.
- **Atualizar Usuário:** Editar os dados de um usuário existente.
- **Excluir Usuário:** Remover um usuário do sistema.

Entidade Aluno:

- **Criar Aluno:** Cadastrar um novo aluno com informações pessoais e vinculação a treino e usuário.
- **Listar Alunos:** Exibir todos os alunos cadastrados, com filtros por nome, ID ou data de matrícula.
- **Atualizar Aluno:** Editar dados de um aluno já registrado.
- **Excluir Aluno:** Remover um aluno da base de dados.

Entidade Treino:

- **Criar Treino:** Cadastrar um novo plano de treino com descrição, tipo, dias da semana e status.
- **Listar Treinos:** Visualizar todos os treinos cadastrados com filtros por tipo ou status.
- **Atualizar Treino:** Modificar as informações de um treino existente.
- **Excluir Treino:** Remover um treino do banco de dados.

Essas funcionalidades serão acessadas por meio de endpoints RESTful, testadas com ferramentas como Insomnia, garantindo a integridade e funcionamento correto da API.

Tecnologias Utilizadas (Banco de Dados e Backend)

O projeto foi desenvolvido em **Java**, utilizando o framework **Spring Boot**, que facilita a criação de aplicações web com uma configuração simples e organizada. Ele permite uma estrutura baseada em camadas (Controller, Service e Repository) e integração fácil com o banco de dados, além do gerenciamento de dependências com o **Maven**.

O banco de dados utilizado será o **MySQL**, um sistema relacional amplamente usado, confiável e compatível com o ecossistema Java.

Para testar as requisições HTTP da aplicação, como criação, edição e exclusão de dados, será utilizada a ferramenta **Insomnia**, que permite simular e validar chamadas à API de forma prática e eficiente.

Além disso, estamos utilizando o **Trello** para o **gerenciamento de tarefas**, organização das entregas e acompanhamento do progresso do projeto em equipe.

Resumo das tecnologias:

- **Linguagem:** Java
- **Framework:** Spring Boot
- **Banco de Dados:** MySQL
- **Ferramenta de Teste:** Insomnia
- **Gerenciador de Dependências:** Maven
- **JPA/Hibernate:** Para mapeamento objeto-relacional (ORM)
- **Gerenciamento de Tarefas:** Trello