

Digitalização e Análise Automática de Dados do Questionário Internacional de Atividade Física (IPAQ): API IPAQMove

Isabelly Araújo Temóteo
Centro Federal de Educação
Tecnológica de Minas Gerais –
CEFET-MG
isabellyatemoteo@gmail.com

Marcelo Teixeira de Sousa Filho
Centro Federal de Educação
Tecnológica de Minas Gerais –
CEFET-MG
marcelo.estudo.cefet@gmail.com

Matheus Amorim Amaral
Centro Federal de Educação
Tecnológica de Minas Gerais –
CEFET-MG
matheusamaral101204@gmail.com

Gabriela Rezende de Oliveira
Venturini
Centro Federal de Educação
Tecnológica de Minas Gerais –
CEFET-MG
doutoragabrielaVenturini@gmail.com

Luís Augusto Mattos Mendes
Centro Federal de Educação
Tecnológica de Minas Gerais –
CEFET-MG
luisaugusto@cefetmg.br

Abstract

This work presents IPAQMove, a REST API developed to digitalize and automate the processing of the International Physical Activity Questionnaire (IPAQ) proposed by the World Health Organization (WHO). This tool estimates the time spent on light, moderate, and vigorous physical activities during the week to calculate the Physical Activity Level (PAL). The manual calculation of the PAL makes large-scale studies time-consuming and prone to errors. IPAQMove addresses this issue by offering automated PAL classification, structured data storage, and tools for researchers to manage studies and analyze individual or aggregated results.

The API was implemented using TypeScript, Fastify, and MySQL, following Clean Architecture principles to ensure modularity and scalability. Endpoints are fully documented through Swagger. The results demonstrate that IPAQMove fills an important gap in current health technologies by providing a validated and standardized method for PAL assessment. Future work includes expanding security and authentication features.

Keywords

Nível de Atividade Física, Questionário Internacional de Atividade Física, API REST

1 Introdução

A Organização Mundial da Saúde (OMS) é composta por vários países cujo objetivo principal é fomentar o bem-estar físico, mental e social de toda população mundial. Além de atuar em questões como o controle de doenças epidêmicas e vacinação, também age monitorando o sedentarismo mundial.

Em 2023, segundo VANESSA [1], 47% dos brasileiros foram considerados fisicamente inativos, completamente sedentários. Entende-se que sedentário é a pessoa que gasta menos de 2.200 kcal por semana e, para ser medianamente saudável, no mínimo 150 minutos em exercícios moderados e aeróbicos por semana. Dá para se entender que não apenas exercícios intensos, mas, da mesma forma, a conclusão de tarefas domésticas têm gasto calórico, combatendo a falta de atividade física. Esse combate está intrinsecamente conectado à 4ª maior causa de óbitos mundialmente [1]. Além de diversos

benefícios para a saúde e tempo de vida, reduzir o sedentarismo colabora em questões socioeconômicas. Em vista disso, os gastos diretos atrelados com cuidados de saúde por conta de saúde à inatividade física chegam perto dos 4 bilhões de dólares anuais no Brasil. Desta forma, torna-se notório o investimento em políticas eficientes para incentivar a prática de atividades físicas [2].

Para ser feita a coleta de dados de grandes grupos populacionais sobre o Nível de Atividade Física (NAF), existe o requisito de instrumentos de precisão e fácil aplicação. Para esse problema, a OMS propôs o Questionário Internacional de Atividade Física (IPAQ). Este serve como um instrumento mundial para determinar o NAF dos indivíduos, estimando o tempo gasto em atividades físicas durante a semana [3]. Essa classificação NAF pode ser dividida entre muito ativo, ativo, irregularmente ativo A, irregularmente ativo B e sedentário.

De acordo com o Ministério da Saúde [4], a classificação do NAF divide-se em: (1) Muito ativo: atinge a recomendação de atividade vigorosa por pelo menos 5 dias na semana e pelo menos 30 minutos por sessão, ou a combinação de atividade vigorosa por pelo menos 3 dias na semana e pelo menos 20 minutos por sessão somada à atividade moderada ou leve por pelo menos 5 dias na semana e pelo menos 30 minutos por sessão; (2) Ativo: cumpre atividade vigorosa por pelo menos 3 dias na semana e pelo menos 20 minutos por sessão, ou atividade moderada ou leve por pelo menos 5 dias na semana e pelo menos 30 minutos por sessão, ou ainda qualquer atividade somada por pelo menos 5 dias na semana e pelo menos 150 minutos por semana; (3) Irregularmente ativo A: atinge pelo menos um dos critérios da recomendação quanto à frequência ou duração, sendo 5 dias na semana ou 150 minutos por semana; (4) Irregularmente ativo B: aquele que não atingiu nenhum dos critérios de recomendação descritos acima; e (5) Sedentário: não realiza nenhuma atividade por pelo menos 10 minutos contínuos na semana.

Aplicar o questionário IPAQ na população brasileira pode fazer com que o governo identifique áreas com maior sedentarismo e estude o planejamento urbano das mesmas. Assim, sendo possível criar políticas públicas voltadas à expansão e revitalização de ciclovias, iluminação pública e parques para atividades ao ar livre,

incluindo parques com aparelhos de ginástica e espaço para caminhadas acessíveis a pessoas idosas. É cabível também aumentar a carga horária de aulas de educação física e incentivar empresas a adotarem programas de saúde. Essas são apenas algumas possíveis mudanças que poderiam mudar bastante o índice de sedentarismo, que está igualmente atrelado à doenças crônicas não transmissíveis (DCNT). Segundo TONY e MAYKON [5], são exemplos das DCNT o aumento da pressão arterial, hiperglicemia, dislipidemias e câncer que são um problema tanto nacional quanto mundial. Reduzir a quantidade de pessoas com DCNT equivale a reduzir o número de hospitalizações e menor demanda por medicamentos. Além disso, possibilita impactar positivamente no aumento da qualidade de vida da população, tornando-a mais ativa e saudável [5]. Adicionalmente, a mudança na qualidade de vida impacta também nos hábitos dos cidadãos que podem reduzir o uso de veículos automotores, por exemplo. Aliado a qualidade de vida há a redução no uso de medicamentos o que impacta diretamente nos custos investidos pelo governo com medicamentos para a população. Portanto, economizar preventivamente com saúde possibilita, ao governo, investir em áreas distintas como educação e tecnologia.

Ademais, segundo RENATO [3], o questionário IPAQ é usado igualmente para estudos epidemiológicos por utilizar de dados sobre intensidade da atividade, duração, frequência e tipo de atividade. Assim, é possível estimar o valor do gasto calórico total e classificar as atividades realizadas por pelo menos 10 minutos como leves, moderadas e vigorosas [3]. É utilizado o MET-minutos/semana para medir quanta energia a pessoa gasta, sendo 1 MET, que significa "Equivalente Metabólico da Tarefa", equivalente a 3,5 ml/kg/min [3]. São considerados atividades leves as que gastam inferiores a 3 MET's, moderadas, entre 3,0 e 4,9 MET's e, vigorosas, 5 MET's ou superior.

O projeto IPAQmove visa facilitar a coleta automatizada dos dados e classificar o indivíduo de acordo com o NAF apresentado como no IPAQ físico [6]. Ao criar um login como usuário comum, é possível ter acesso ao questionário IPAQ no formato digital com o mesmo teor do questionário físico, proposto pela OMS. O IPAQmove contempla como funcionalidades o questionário em formato digital do IPAQ [6], gerenciamento de pesquisas pelos pesquisadores, acesso a *dashboard* com dados sumarizados, registro e login de usuários e cálculo de todo o processo para obter a classificação NAF. Essas são funcionalidades que serão detalhadas dessa forma, mais adiante quando abordarmos a solução.

O questionário IPAQ existe apenas como modelo físico, sendo necessário que o pesquisador, após aplicar o questionário, faça todo o trabalho de tabular e analisar os dados coletados manualmente para calcular o NAF dos indivíduos participantes de uma pesquisa. Em pesquisas que contemplam uma grande quantidade de indivíduos o esforço demandado é grande. O IPAQmove surge para facilitar esse processo, tendo o resultado dos MET's e a classificação do NAF dos indivíduos sendo efetuada de forma automatizada. Assim, o pesquisador consegue ver os resultados tanto individualmente quanto de toda a amostra de pessoas que participaram de uma pesquisa. Caso o usuário não seja um pesquisador, mesmo assim consegue saber sua classificação, basta apenas clicar no botão de finalizar o questionário.

O fluxo de execução do questionário IPAQ é dinâmico, o que possibilita que cada usuário, ao respondê-lo, possa percorrer um

caminho diferente. Ao manusear o IPAQ na versão manual [6], é possível perceber perguntas como "Atualmente você tem ocupação remunerada ou faz trabalho voluntário fora de sua casa?" e, caso a resposta for não, a seção 1 inteira é ignorada sendo abordada as questões da próxima seção. Cabe ressaltar que algumas perguntas na própria seção são de intensidades diferentes, o que pode acabar causando confusão na hora de obter o gasto calórico total. Além disso, existem tabelas para informar o tempo horas/minutos gastos ao longo de todos os dias da semana. Do questionário [6], o aplicador tem que pegar todas as 7 informações de cada um dos dias da semana para cada uma das 12 tabelas e somar todos os valores obtidos, além de usar a intensidade da atividade adequada. Para obter o gasto calórico é necessário somar o resultado do valor obtido pelas 3 intensidades identificadas. São muitas considerações para um único pesquisador prestar atenção e, considerando a aplicação do questionário em uma amostra de 30 pessoas, já gastaria horas afincadas sendo mais suscetível a falhas. Com a adoção do sistema IPAQmove, os resultados são obtidos, logo após a finalização do questionário, em poucos segundos, poupando muitas horas de trabalho além de evitar falhas de cálculo.

Para isso tudo ser possível, é crucial uma API (Interface de Programação de Aplicação) que conecte com o front-end e o back-end do sistema, tanto para o web quanto ao mobile. O software passou por mudanças estruturais visando o usuário administrador, indispensável ao sistema. Essa estrutura aborda os conceitos da Clean Architecture [7] para uma melhor construção, organização e manutenção.

Nesse contexto, o objetivo deste trabalho é projetar e implementar uma API REST para o sistema IPAQmove, visando padronizar o acesso aos dados do questionário IPAQ na versão digital, bem como oferecer ferramentas de armazenamento, consulta e processamento dos dados. Ela é importante por inserir uma camada a mais de segurança, já que a API controla as funções e o que consegue ser acessado sem exposição direta do banco de dados. Conta ainda com controle de autenticação e autorização com uso de tokens de forma a limitar funções da API que um tipo de usuário não poderia ter. Ademais, conta com a escalabilidade, por ser possível aplicar em sistemas web e mobile e a construção de novas funções não interfere nas que estão integradas, bem como permitir o front-end e o back-end evoluir de forma independente. Além disso, a padronização ajuda a entender como o sistema irá se comunicar.

Assim, se faz necessário o uso de uma arquitetura de API e construção das rotas e controladores levando aos *endpoints* principais, bem como outras pastas auxiliares como "config" para configuração do banco e ambiente, "middleware" para autenticação, logs e validação, "services" para regras de negócio e "types" para tipos TypeScript compartilhados como é abordado na seção 3. Segundo DAN [8], para o processo de API REST, um cliente de API envia uma chamada de API para o *endpoint* da API correspondente, no caso, são acessados usando métodos de solicitação HTTP, como GET, POST, PUT e DELETE como apresentado na seção 4.

O IPAQmove é uma solução multiplataforma, tanto web quanto mobile, sendo usada a mesma API em ambas as plataformas. Assim, é possível utilizá-la para registrar novas contas e efetuar o login, gerenciar os diferentes tipos de usuários (comum, pesquisador e administrador), gerenciar as pesquisas conseguindo criar, atualizar, buscar e deletar informações e, também, realizar os cálculos para o

NAF coletando as respostas de todas as perguntas e apresentando para o usuário como abordado na seção 4. Essas funções são a base para todo o sistema, sem elas nada terá a perspectiva de funcionar.

Para usuários de qualquer tipo, é necessário que tenha o consentimento para a coleta de dados estritamente necessários para o funcionamento. Essas questões serão tratadas utilizando do Termo de Consentimento Livre e Esclarecido (TCLE) ou o Termo de Assentimento Livre e Esclarecido (TALE) em caso de menor de idade, conforme a resolução 466 de 17 de dezembro de 2012. O TCLE tem cláusulas específicas à preservação, reuso e compartilhamento dos dados coletados em pesquisa. Ademais, o projeto consta também com a conformidade de diretrizes aplicáveis e com a Lei Geral de Proteção de Dados (LGPD). Sem esse consentimento, não será possível acessar o sistema. Sobre a coleta dos voluntários da pesquisa, serão variáveis sobre: nome completo, idade, sexo, massa corporal, estatura, se trabalha remunerado ou voluntário, quantidade de horas de trabalho e de sono, localidade e NAF calculado pelo próprio sistema com perguntas respondidas sobre frequência e duração de tempo junto da intensidade da atividade. Estes serão armazenados no próprio banco de dados do projeto.

A preservação dos dados a longo prazo obtidos nas pesquisas representam valor ao trabalho tendo em vista que dados como a distribuição de classificação do NAF por localidade, faixa etária ou gênero poderão ser utilizados como dados relevantes à pesquisadores e organizações. A escolha sobre quais dados são relevantes e com qualidade para serem mantidos estará em atendimento ao especificado na LGPD. Todos os dados necessários, que estejam bem documentados, completos e com potencial de contribuir com a pesquisa serão mantidos. Esses dados serão utilizados principalmente pelo administrador, que conseguirá, conforme localidade, data de realização do questionário IPAQ e outros filtros aplicáveis, fazer novas análises.

Para a metodologia, foi utilizado de uma adaptação do método de engenharia FEBRACE [9]. Tal adaptação consiste em 6 etapas e cabe à equipe realizar o levantamento de referências e sistemas correlatos, analisar os requisitos e atualizar a modelagem conforme a necessidade, desenvolver a solução, realizar testes, ajuste dos erros e disponibilizar o software no ambiente do Laboratório de Iniciação Científica e Extensão da Computação (LINCE).

O restante deste trabalho está organizado da seguinte forma: na Seção 2 são apresentados os Materiais e Métodos; na Seção 3 são apresentados os Resultados; na Seção 4 são apresentados as Discussões; e, por fim, na Seção 5 são apresentadas as Conclusões.

2 Materiais e Métodos

Este trabalho se caracteriza como um estudo de desenvolvimento tecnológico, cujo objetivo é projetar e implementar uma API REST para o sistema IPAQMove. A API visa padronizar o acesso aos dados do questionário IPAQ na versão digital, bem como oferecer ferramentas de armazenamento, consulta e processamento dos dados.

O processo de desenvolvimento seguiu o Método de Engenharia definido pela FEBRACE [9], adaptado por Soares [10] e Aguiar [11], sendo composto por seis etapas sistemáticas. A Etapa 1, de Levantamento de Referências e Sistemas Correlatos, consistiu na revisão de soluções existentes para a manipulação dos dados do IPAQ e outras métricas de saúde, estabelecendo a base teórica e técnica. Na sequência, a Etapa 2 focou na Análise e Modelagem, definindo os

requisitos funcionais e não funcionais, seguida pela elaboração do Diagrama Entidade-Relacionamento (DER). A fase de construção ocorreu na Etapa 3, o Desenvolvimento da API, com a implementação do software utilizando TypeScript¹ e Fastify², fundamentada na Clean Architecture [7] e com banco de dados MySQL³ hospedado no Laboratório de Iniciação Científica e Extensão da Computação (LINCE).

Posteriormente, a Etapa 4 envolveu o Planejamento e Execução de Testes, realizando a verificação sistemática de *endpoints* via extensão REST Client⁴ no VS Code⁵ para garantir segurança e consistência, incluindo a geração da documentação Swagger⁶. Com base nos resultados, realizou-se a Etapa 5, de Ajustes e Correções, aplicando manutenção corretiva e evolutiva. O ciclo encerra-se na Etapa 6, a Publicação, com a disponibilização final do produto de software para consumo. A Figura 1 apresenta a metodologia adotada.

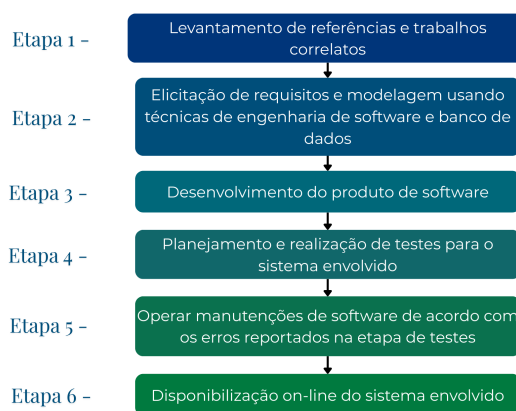


Figura 1: Etapas metodológicas do desenvolvimento.

Para viabilizar a implementação e documentação deste sistema, foi selecionado um conjunto específico de tecnologias e representações. Utilizou-se a linguagem TypeScript para garantir robustez através da tipagem, em conjunto com o framework Fastify [12] para criação de uma API de alto desempenho. A estrutura do projeto seguiu os preceitos da Clean Architecture [7] para facilitar a manutenção, enquanto os dados foram persistidos em MySQL. Para a validação e documentação, empregou-se o REST Client e o Swagger, respectivamente. Além do código fonte, o entendimento do sistema é complementado por representações visuais como o DER, o Modelo Lógico representado pelo Diagrama de Tabelas Relacionais (DTR) e a estrutura da arquitetura adotada.

3 Resultados

A construção da API resultou em uma solução documentada para manipulação do questionário IPAQ, para o sistema IPAQMove, relacionada a pesquisas que podem ser criadas por usuários do tipo

¹<https://www.typescriptlang.org/>

²<https://fastify.dev/>

³<https://www.mysql.com/>

⁴<https://marketplace.visualstudio.com/items?itemName=humao.rest-client>

⁵<https://code.visualstudio.com/>

⁶<https://swagger.io/>

pesquisador. A aplicação da Clean Architecture permitiu isolar regras de negócio, simplificar integrações e facilitar manutenções futuras [7]. A adoção de práticas consolidadas de engenharia de software contribuiu para um sistema mais consistente, modular e escalável [13]. A hospedagem no Laboratório de Iniciação Científica e Extensão da Computação (LINCE) possibilitou acesso centralizado e seguro aos dados.

Dentre os principais resultados obtidos, destaca-se a entrega de uma API REST funcional, operando com integração completa ao banco de dados MySQL e disponibilizando endpoints específicos para o gerenciamento de usuários, pesquisas, questionários, seções e respostas. O produto de software conta ainda com documentação automática gerada via Swagger, que inclui exemplos detalhados de entrada e saída, consolidando uma estrutura modular apta para expansão e alinhada aos princípios da Clean Architecture [7].

O desenvolvimento do produto de software foi conduzido mediante uma sequência de procedimentos técnicos que visaram a robustez e a manutenibilidade do sistema. Inicialmente, foi realizada a configuração do banco de dados MySQL em seu sistema de hospedagem, que incluiu a criação das tabelas, índices e chaves necessárias para o armazenamento dos dados do IPAQ. Simultaneamente, foi configurado o ambiente de desenvolvimento Node.js⁷ utilizando TypeScript para assegurar maior segurança de tipagem [14]. O processo seguiu com a criação da estrutura de pastas organizadas segundo os princípios da Clean Architecture [7], seguida pelo desenvolvimento dos endpoints REST necessários para usuários, pesquisas, questionários e respostas. Por fim, foi realizada a integração com o Swagger para a documentação automática das funcionalidades.

Os detalhes técnicos de implementação e as representações das soluções adotadas são descritos a seguir.

3.1 Estrutura da Arquitetura e Fluxo de Dados

A estrutura da API foi desenvolvida seguindo uma adaptação dos princípios da Clean Architecture, com o propósito de garantir o desacoplamento entre a lógica de negócio e a infraestrutura. O fluxo de dados de uma requisição segue a regra da dependência. Uma requisição HTTP percorre o fluxo sequencial detalhado na Figura 2.

Neste fluxo, a camada de *serviços* atua como os *Serviços de Aplicação*, realizando a lógica de negócio e as interações. Para o escopo do projeto, a lógica dos Casos de Uso foi consolidada nesta camada. A garantia do desacoplamento é dada pelo uso do *repositório*: já que a camada de *serviços* não acessa diretamente o banco de dados (MySQL), mas depende de uma *Interface de Repositório* (uma Abstração). Essa Interface é implementada na camada de Infraestrutura (implementação do banco de dados), que contém o código para a persistência.

Sendo assim, essa separação assegura que a lógica central de negócio seja testável e independente da tecnologia selecionada para o armazenamento, garantindo maior manutenibilidade futura. A manutenibilidade de software, conforme Pressman [15], refere-se à facilidade com que um programa pode ser corrigido, adaptado ou aprimorado após a sua entrega. Uma arquitetura bem desacoplada

contribui significativamente para essa característica, permitindo que alterações em uma camada tenham impacto mínimo em outras.

Além disso, em conformidade com a LGPD, a arquitetura emprega tokens JWT (*JSON Web Tokens*) para autenticação e criptografia de dados sensíveis, como senhas e palavras-chave. Variáveis sociodemográficas, essenciais para estudos epidemiológicos, são mantidas estruturadas no banco de dados após consentimento prévio. Contudo, a privacidade é garantida restringindo-se a visualização dessas informações exclusivamente a usuários com perfil de pesquisador ou administrador.

Camada de aplicação e domínio (Regra de negócio)

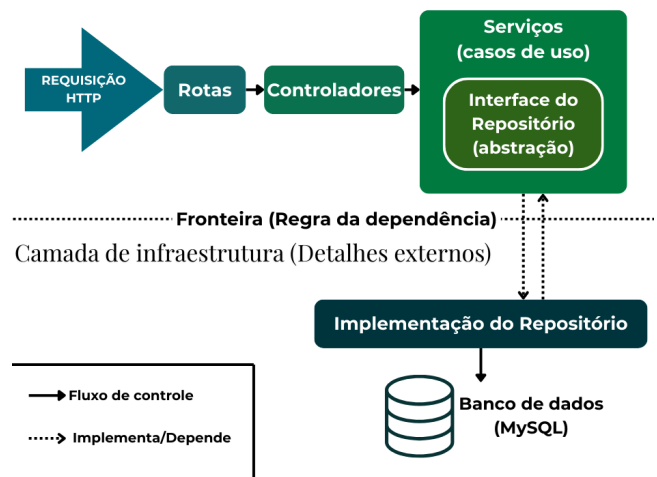


Figura 2: Fluxo de dados e camadas da arquitetura.

3.2 Tecnologias

A seleção das ferramentas tecnológicas priorizou desempenho e manutenibilidade. Para a persistência de dados, o Sistema Gerenciador de Banco de Dados (SGBD) adotado foi o MySQL. Sua escolha justifica-se por ser uma solução de código aberto amplamente consolidada e frequentemente utilizada no desenvolvimento de aplicativos web, garantindo confiabilidade no armazenamento relacional [16].

No desenvolvimento, optou-se pela linguagem TypeScript. A principal motivação para seu uso é a tipagem estática, que permite a detecção de erros em tempo de compilação e melhora a inteligibilidade do código, facilitando a aplicação dos princípios da Clean Architecture [17].

Para gerenciar as requisições HTTP, utilizou-se o framework Fastify. Diferente de outras opções do ecossistema Node.js, o Fastify foi escolhido especificamente por sua arquitetura focada em minimizar o consumo de recursos computacionais utilizados pelas operações internas da ferramenta, resultando em uma melhor performance, sendo capaz de processar um volume maior de requisições por segundo [18].

Por fim, para a validação e documentação, adotou-se o REST Client e o Swagger. O Swagger foi selecionado por permitir a descrição padronizada de APIs RESTful e oferecer uma interface interativa que facilita o consumo e o entendimento das funcionalidades [19].

⁷<https://nodejs.org/>

O REST Client foi empregado especificamente na Etapa 4 para a execução e verificação sistemática dos testes das funcionalidades garantindo a segurança e consistência do sistema [20].

3.3 Modelagem da Base de Dados

A fase de modelagem da base de dados do sistema resultou na elaboração de diagramas fundamentais para a estruturação do produto de software.

Diagrama Entidade-Relacionamento (DER) - A Figura 3 apresenta o DER desenvolvido para o sistema IPAQMove. Este diagrama

opera no nível conceitual, representando as entidades e seus relacionamentos de forma independente da implementação tecnológica. Segundo Silberschatz [21], o uso do modelo entidade-relacionamento é essencial para descrever a estrutura global dos dados e garantir que os requisitos semânticos do sistema sejam atendidos antes da implementação.

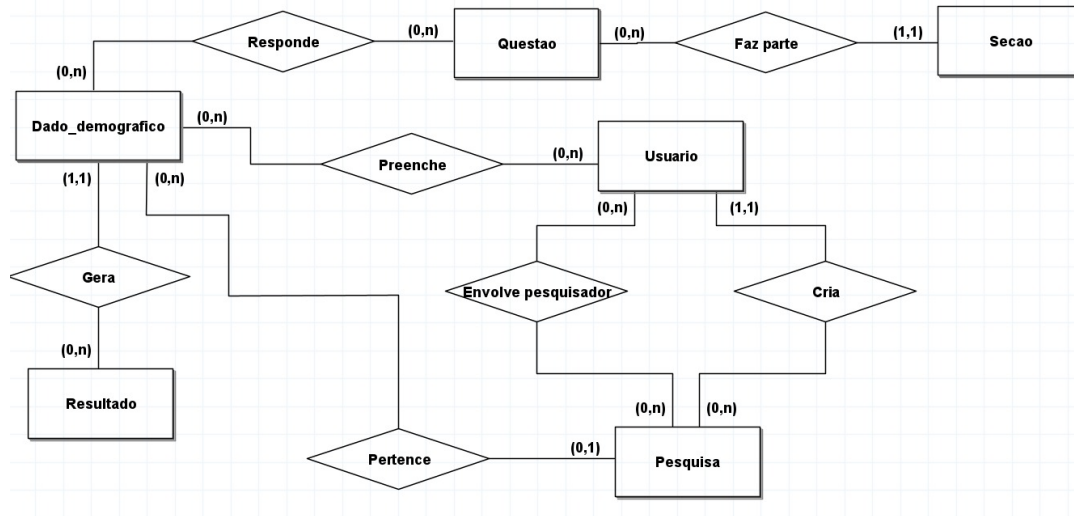


Figura 3: Diagrama Entidade-Relacionamento (DER) do sistema IPAQmove.

O modelo é composto por seis entidades principais que estruturam o domínio do problema. A entidade central Usuario representa os usuários comuns que irão responder as pesquisas, administradores ou pesquisadores. Sendo assim, possui um relacionamento de cardinalidade (1,n) com a entidade Pesquisa através da ação "Cria", indicando que um usuário é responsável por gerenciar múltiplos estudos.

A captura dos dados dos participantes é representada pela entidade Dado_demografico, que se relaciona com a Pesquisa (via "Pertence") e com o Usuario (via "Preenche"). A estrutura fixa do questionário IPAQ é modelada pelas entidades Secao e Questao, onde a relação "Faz parte" (1,1 para 0,n) define que uma seção agrupa diversas questões específicas.

O fluxo de respostas é mapeado pelo relacionamento "Responde" entre Dado_demografico e Questao. Por fim, a entidade Resultado é derivada deste conjunto de dados, sendo gerada (relacionamento "Gera") a partir do processamento das informações demográficas e das respostas submetidas, consolidando o cálculo do nível de atividade física.

Diagrama de Tabelas Relacionais (DTR) - A Figura 4 exibe o Diagrama de Tabelas Relacionais (DTR) apresentando os atributos de cada tabela. Esta representação detalha a estrutura relacional do banco de dados, especificando tabelas, colunas, tipos de dados e restrições. A elaboração do DTR é onde as abstrações conceituais

são convertidas em estruturas tabulares, estabelecendo chaves primárias e estrangeiras para assegurar a integridade referencial dos dados armazenados [22].

A entidade Usuario é responsável pelo gerenciamento de acesso e identificação, armazenando atributos como nome, email, senha, o perfil de acesso (tipo, status) e coordenadas geográficas (latitude, longitude, cidade, uf). Vinculada a esta, a tabela Pesquisa centraliza os metadados dos estudos criados, contendo descritores como nome_pesq, instituicao e palavra_chave, além do registro temporal definido em datahora_pesq.

Para a caracterização dos participantes, a tabela Dado_demografico registra dados antropométricos e sociais essenciais ao cálculo do algoritmo, incluindo peso, estatura, idade, sexo, bem como métricas de rotina como horas_trabalha_dia. A estrutura fixa do instrumento IPAQ é representada pelas tabelas Secao, que organiza o agrupamento (titulo, descricao), e Questao, que armazena o enunciado (texto_pergunta) e a referência à seção correspondente.

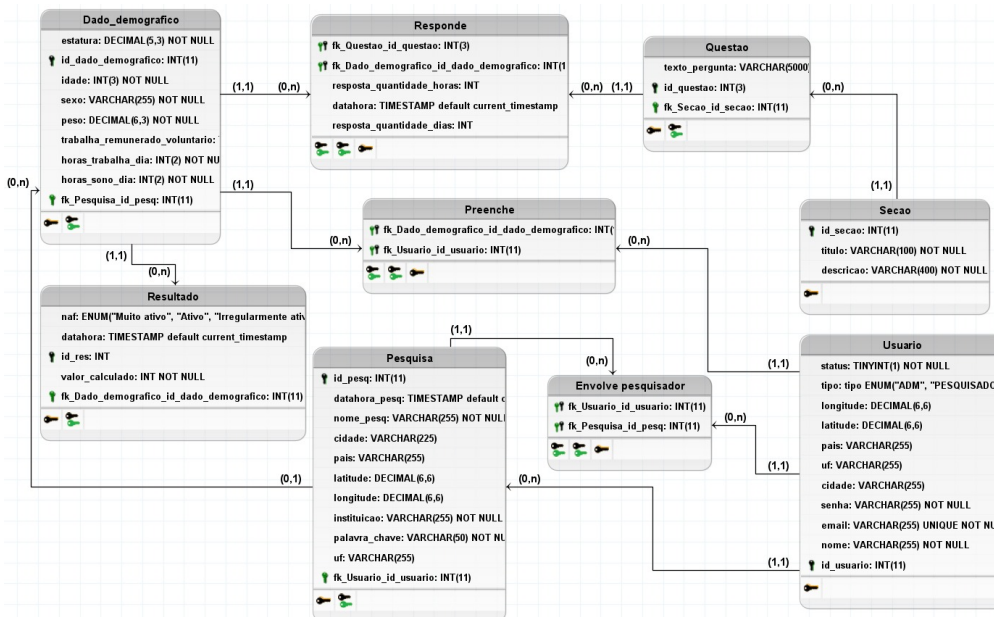


Figura 4: Diagrama de Tabelas Relacionais (DTR) do banco de dados.

Por fim, a interação do participante é persistida na entidade associativa Responde, que relaciona os dados demográficos às questões, armazenando os valores quantitativos informados (*resposta_quantidade_dias*, *resposta_quantidade_horas*) e a quantidade de tempo. O processamento final desses dados é consolidado na tabela Resultado, que armazena tanto o valor numérico obtido (*valor_calculado*) quanto a classificação categórica do Nível de Atividade Física na coluna *naf*.

3.4 Estrutura e Documentação da Solução

Esta subseção detalha a arquitetura interna do software e sua interface externa de documentação.

Arquitetura da API - A Arquitetura interna da API IPAQMove segue os princípios da Clean Architecture [7], organizando o projeto em camadas coesas e independentes. A representação da estruturação principal do sistema é exibida na Lista 1.

Lista 1: Estrutura do projeto IPAQMove

```
src/
config/ # Configuracoes MySQL e ambiente
controllers/ # Entrada e saida das requisicoes
middleware/ # Autenticacao, logs e validacao
routes/ # Mapeamento de rotas
services/ # Regras de negocio
types/ # Tipos TypeScript compartilhados
server.ts # Inicializao do servidor Fastify
```

Documentação Swagger - A documentação do sistema foi tratada como um componente essencial para que um dos softwares do ecossistema IPAQMove, seja o sistema web ou o aplicativo, consiga fazer solicitações e obter dados por meio de outro software também pertencente a esse ecossistema. Nesse caso, o software responsável por atender as solicitações e fornecer esses dados é a API IPAQMove. A integração com o Swagger foi realizada para gerar uma

interface visual e interativa da API, permitindo que os desenvolvedores testem e compreendam todos os *endpoints*, parâmetros e modelos de dados, sendo fundamental para a manutenção e integração contínua. A Figura 5 apresenta uma seção da documentação gerada automaticamente via Swagger. O uso de documentação padronizada favorece a clareza nas funcionalidades REST, conforme boas práticas modernas de APIs [23].

4 Discussão

O presente estudo cumpriu com sucesso o objetivo de projetar e implementar uma API REST para o sistema IPAQMove, padronizando o acesso aos dados do questionário IPAQ na versão digital e fornecendo ferramentas para o seu armazenamento, consulta e processamento automatizado. Segundo Robert C. Martin [7], a implementação da Clean Architecture em um sistema o torna escalável e eficiente, proporcionando uma maior eficácia quanto ao gerenciamento da coleta de dados do questionário IPAQ.

Uma constatação levantada durante a condução desta pesquisa foi a ausência, no ecossistema da tecnologia da saúde, de uma API padronizada de alto nível para consulta direta do Nível de Atividade Física (NAF) de um indivíduo. Esta brecha é compreensível, visto que os presentes métodos para obter os dados necessários para esta avaliação variam desde dispositivos *wearables* a questionários físicos, como o IPAQ. No entanto, sensores *wearables* que são capazes de realizar o cálculo do NAF tendem a ser muito caros e acabam inviabilizando pesquisas de grande escala. No Brasil, país de renda média, os estudos com acelerômetros enfrentam desafios significativos, e um alto custo, juntamente com a curta vida útil, falhas e perdas de monitores, o que pode comprometer a sustentabilidade e a continuidade da pesquisa [24]. Em contrapartida, questionários,



Figura 5: Interface Swagger da API IPAQMove.

como o Questionário Internacional de Atividade Física (*International Physical Activity Questionnaire* - IPAQ), permitem a coleta de grandes amostras populacionais [25].

Atualmente, o mercado de software aplicado à área da saúde é dividido em duas subcategorias principais de APIs: 1) plataformas *fitness* voltadas para eventos, como o Strava⁸, e 2) plataformas da saúde com o foco em entender separadamente cada métrica, como o Google Fit⁹ e Apple Fitness¹⁰. As APIs citadas no item 1, como

⁸<https://developers.strava.com/docs/getting-started/>

⁹<https://developers.google.com/awareness/overview?hl=pt>

¹⁰<https://developer.apple.com/health-fitness/>

a do Strava, são otimizadas para registro de atividades físicas intencionais, fornecendo métricas detalhadas sobre exercícios como corridas e caminhadas. Permitem rastrear e analisar o desempenho, focando em *insights* de inteligência artificial. Utilizam GPS e dados de desempenho, como ritmo (*pace*) e elevação, e as registram em um mapa detalhado para melhor visualização do usuário [26]. Já as APIs citadas no item 2, possuem o foco em capturar o comportamento físico do indivíduo por 24 horas, oferecendo dados brutos que podem servir para o cálculo do NAF. No entanto, estes dados coletados ainda exigem processamento e agregação para serem significativos, sendo necessário que o desenvolvedor utilize funções como *AggregateRequest* para realizar a conversão de dados brutos em narrativas de saúde significativas [27].

No âmbito acadêmico, esforços também foram direcionados para a transposição do IPAQ para meios digitais, visando mitigar os erros manuais de tabulação e o tempo despendido na aplicação presencial [28]. Estudos anteriores focaram majoritariamente no desenvolvimento de formulários web e aplicativos móveis autônomos, buscando validar a equivalência psicométrica entre o formato eletrônico e o impresso [29]. Entretanto, observa-se que tais soluções frequentemente operam como sistemas isolados ou "silos de dados", onde a lógica de cálculo do NAF e a base de dados permanecem acopladas à interface do usuário. Essa característica limita a reutilização do código e dificulta a interoperabilidade com outros sistemas de saúde ou prontuários eletrônicos, obrigando pesquisadores a exportarem dados manualmente para análises posteriores. A quebra desses silos de dados e a busca por interoperabilidade através de APIs tornou-se, inclusive, um dos maiores desafios atuais da informática em saúde [30].

Diferentemente dessas abordagens acadêmicas tradicionais, o presente trabalho não se limita apenas à digitalização do formulário, mas propõe uma mudança arquitetural ao expor as regras de negócio do IPAQ através de uma API RESTful. Enquanto a literatura existente tende a focar na ferramenta de coleta (o formulário final para o usuário), a API do IPAQMove foca na disponibilização do serviço de processamento, permitindo que o algoritmo de classificação do NAF seja consumido por aplicações terceiras de forma transparente e dissociada da interface. Essa abordagem preenche uma lacuna deixada por iniciativas acadêmicas anteriores, que, embora eficazes na validação do questionário digital, não priorizaram a integração sistêmica e a escalabilidade tecnológica propostas por esta pesquisa.

Portanto, apesar da disponibilidade de dados brutos nas APIs atuais (passos, distância, tempo, etc.), constata-se a ausência de ferramentas que processem essas informações para mensurar o NAF automaticamente. Neste contexto, a API desenvolvida para o IPAQMove preenche essa lacuna tecnológica, oferecendo um método validado internacionalmente que possibilita monitorar o comportamento sedentário. Sendo o sedentarismo uma questão central de saúde pública, esta abordagem torna-se essencial para permitir a análise de dados em escalas que seriam inviáveis apenas com sensores.

Contudo, apesar dos avanços propostos para a automação da ferramenta, convém destacar as limitações atuais deste estudo. Até o presente estágio de desenvolvimento, a avaliação do IPAQMove restringe-se à verificação técnica e aos testes funcionais em ambiente controlado. A solução ainda não foi aplicada em cenários

reais com usuários finais, o que impede, no momento, a mensuração empírica de métricas de desempenho sob demanda real e a quantificação exata da taxa de redução de erros em relação ao cálculo manual tradicional.

5 Considerações Finais

O presente trabalho cumpriu o objetivo de projetar e implementar uma API REST para o sistema IPAQMove. Até o estágio atual, foi possível apresentar os resultados preliminares referentes à modelagem, documentação e testes iniciais do software, confirmando a viabilidade técnica da solução proposta para a digitalização e mensuração do Nível de Atividade Física (NAF).

A validação da Clean Architecture nesta fase de desenvolvimento demonstrou ser fundamental, indicando que a estrutura do software possui a robustez necessária para garantir escalabilidade e manutenibilidade. Além disso, a pesquisa evidenciou uma lacuna nas tecnologias de saúde vigentes: enquanto ferramentas como Google Fit e Strava focam, respectivamente, em dados brutos de sensores ou métricas de desempenho esportivo, o IPAQmove se diferencia ao oferecer uma análise específica do comportamento sedentário baseada em protocolo validado.

Dessa forma, os resultados obtidos até o momento consolidam a base tecnológica necessária para que o IPAQmove atue como uma ferramenta eficaz em estudos epidemiológicos, viabilizando o monitoramento da saúde pública de forma automatizada e em larga escala.

Como trabalhos futuros, visando o aprimoramento contínuo e a validação extensiva da ferramenta, planeja-se a expansão das funcionalidades de segurança e dos mecanismos de autenticação da API. Além disso, prevê-se a condução de um estudo comparativo quantitativo que avalie o cálculo automatizado do NAF fornecido pelo sistema em relação ao cálculo manual tradicionalmente executado por pesquisadores, com o intuito de mensurar a precisão da ferramenta e a taxa de redução de erros matemáticos. Por fim, fará parte do escopo futuro a avaliação e documentação de métricas de desempenho da API em cenários reais de estresse. A coleta empírica de dados, focada no tempo de resposta e na capacidade de requisições processadas por segundo (RPS), servirá para justificar e atestar de forma robusta a escolha arquitetural do framework Fastify em detrimento de outras alternativas consolidadas no ecossistema Node.js.

Agradecimentos

Nossos agradecimentos ao Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG), à Diretoria de Graduação do CEFET-MG, ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), à Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) e ao Laboratório de Iniciação Científica e Extensão da Computação (LINCE) pelo apoio para a realização deste trabalho.

Referências

- [1] Vanessa Labanca et al. Educação em saúde no combate ao sedentarismo. *Pensar Acadêmico*, 21(5):1713–1728, October 2023.
- [2] Victória Paulo Menin. Caminhos para aumentar a prática de atividades físicas pela população. <https://agenciagov.etc.com.br/noticias/202408/caminhos-para-aumentar-a-pratica-de-atividades-fisicas-pela-populacao>, 2024. Acesso em: 14 de novembro 2025.
- [3] Renato Pardini et al. Validação do questionário internacional de nível de atividade física (ipaq - versão 6): estudo piloto em adultos jovens brasileiros. *Bras. Ciên. e Mov.*, 9(3):45–51, July 2001.
- [4] MINISTÉRIO DA SAÚDE. Classificação do nível de atividade física ipaq. <https://files.cercomp.ufg.br/weby/up/482/o/Classificacao-NivelAF-IPAQ2007.pdf?1435678568>, 2007. Acesso em: 14 de novembro 2025.
- [5] Tony Santos and Maykon Novais. Atividade física na atenção básica e doenças crônicas não transmissíveis: um breve olhar para o cenário atual no Brasil. *Revista Portal - Saúde e Sociedade*, 8(1):1–12, January 2025.
- [6] UDESC Universidade do Estado de Santa Catarina. Questionário internacional de atividade física. https://www.cfid.udesc.br/arquivos/id_submenu/1173/ipaq.pdf, 2023. Acesso em: 14 de novembro 2025.
- [7] Robert C. (Uncle Bob) Martin. *Clean Architecture: A Craftsman's Guide to Software Structure and Design*. Prentice Hall, 1 edition, 2017.
- [8] Dan Nosowitz and Michael Goodwin. O que é um endpoint de api? <https://www.ibm.com/br-pt/think/topics/api-endpoint>, 2025. Acesso em: 14 de novembro 2025.
- [9] FEBRACE Feira Brasileira de Ciências e Engenharia. Metodologia de engenharia. <https://febrace.org.br/participe/planeje-seu-projeto/requisitos/>, 2021. Acesso em: 14 de novembro 2025.
- [10] Guilherme Martins Soares, João Victor Domingos e Souza, Gabriella Castro Barbosa Costa Dalpra, Luis Augusto Mattos Mendes, and Luan Soares Oliveira. Sistema para verificação da não conformidade de produtos de software com a lei geral de proteção de dados pessoais (lgpd). In *Anais do XVI Computer on the Beach*, pages 478–482, 2025.
- [11] Bernardo Gomes de Aguiar, Eduardo Lima S. M. de Rezende, Luis Augusto Mattos Mendes, Inácio Matias Leonardo Lopes, and Alexandre M. Gama de Deus. Vintana: uma plataforma para publicação de acervos de rádios. In *Anais do XVI Computer on the Beach*, pages 689–694, 2025.
- [12] Fastify Team. Benchmarks: High performance node.js web framework. <https://fastify.dev/benchmarks/>, 2024. Acesso em: 21 de novembro 2025.
- [13] Ian Sommerville. *Software Engineering*. Pearson, 10 edition, 2021.
- [14] Dan Vanderkam. *Effective TypeScript: 62 Specific Ways to Improve Your TypeScript*. O'Reilly Media, Sebastopol, 2019. ISBN 978-1492053743.
- [15] Roger S. Pressman and Bruce R. Maxim. *Engenharia de Software: Uma Abordagem Profissional*. McGraw Hill Brasil, Porto Alegre, 8 edition, 2016.
- [16] Locaweb. Mysql: Guia completo de como utilizar o sistema. Locaweb Blog, 2024. URL <https://www.locaweb.com.br/blog/temas/codigo-aberto/mysql-guia-completo-de-como-utilizar-o-sistema/>. Acesso em: 11 out. 2025.
- [17] Microsoft. Typescript: Javascript with syntax for types. Documentação Oficial, 2024. URL <https://www.typescriptlang.org/>. Acesso em: 11 out. 2025.
- [18] Fastify. Fastify: Fast and low overhead web framework, for node.js. Documentação Oficial, 2024. URL <https://fastify.dev/>. Acesso em: 11 out. 2025.
- [19] SmartBear. Swagger: Api documentation & design tools for teams. SmartBear Software, 2024. URL <https://swagger.io/>. Acesso em: 11 out. 2025.
- [20] Huachao Mao. Rest client for visual studio code. <https://github.com/Huachao/vscode-restclient>, 2024. Versão 0.25.1. Acesso em: 21 de novembro 2025.
- [21] Abraham Silberschatz, Henry F. Korth, and S. Sudarshan. *Database System Concepts*. McGraw-Hill, 5 edition, 2006. ISBN 978-0-07-295886-7.
- [22] Carlos Alberto Heuser. *Projeto de Banco de Dados*. Bookman, 6 edition, 2009.
- [23] Brenda Jin, Saurabh Sahni, and Amir Shevat. *Designing Web APIs: Building APIs That Developers Love*. O'Reilly Media, Sebastopol, 2018. ISBN 978-1492026921. Capítulo sobre documentação e Developer Experience (DX).
- [24] Andrea Wendt et al. Lessons from 13 years of accelerometry measurements in five Brazilian cohorts: methodological aspects. *Cadernos de Saúde Pública*, 41(3):e00011724, 2025. doi: 10.1590/0102-3111XEN011724.
- [25] Andre Henriksen et al. Measuring physical activity using polar wrist-worn triaxial activity trackers: A systematic review. *International Journal of Exercise Science*, 13(4):438–454, May 2020.
- [26] Strava. Strava API. Strava, 2024. URL <https://developers.strava.com/docs/getting-started/>. Acesso em: 24 de novembro 2025.
- [27] Google. *Google API Awareness*. Google, 2024. URL <https://developers.google.com/awareness/overview?hl=pt>. Acesso em: 24 de novembro 2025.
- [28] I. R. Lemes et al. Concordância entre os formatos impresso e eletrônico do ipaq-l. *Revista Brasileira de Medicina do Esporte*, 16(1):51–54, 2010.
- [29] E. Ahlstrand et al. Measuring physical activity in a cardiac rehabilitation population using a smartphone-based questionnaire. *Journal of Medical Internet Research*, 15(3):e61, 2013.
- [30] N. Edulakanti. From data silos to smart integrations: A framework for enterprise-wide interoperability using apis, hl7, and json. *International Journal of Intelligent Systems and Applications in Engineering*, 12(1):245–255, 2024.