Saúde e Bem-estar - Base de Dados

Trabalho realizado por:

Tiago Martins Nº82295

André Santos Nº82296

Gonçalo Marques Nº

Enzo Correia Nº 82299

Índice

[Introdução 4](#_Toc199182350)

[Divisão de tarefas 4](#_Toc199182351)

[Interface gráfica 5](#_Toc199182352)

[Estrutura da User interface 5](#_Toc199182353)

[Base de dados 8](#_Toc199182354)

[Estrutura da Base de dados 8](#_Toc199182355)

[Funcionalidades implementadas 9](#_Toc199182356)

[Integração com a Base de Dados 9](#_Toc199182357)

[Gráfico 11](#_Toc199182358)

[Testes e resultados 11](#_Toc199182359)

[Exportação e Importação 12](#_Toc199182360)

[Testes realizados 12](#_Toc199182361)

[Conclusão 13](#_Toc199182362)

Índice de Figuras

[Figura 1 5](#_Toc199180158)

[Figura 2 5](#_Toc199180159)

[Figura 3 5](#_Toc199180160)

[Figura 4 5](#_Toc199180161)

[Figura 5 5](#_Toc199180162)

[Figura 6 6](#_Toc199180163)

[Figura 7 7](#_Toc199180164)

[Figura 8 7](#_Toc199180165)

[Figura 9 7](#_Toc199180166)

[Figura 10 7](#_Toc199180167)

# Introdução

Este projeto foi desenvolvido com o objetivo de permitir aos utilizadores registarem os seus hábitos diários relacionados com a prática de exercício físico e a ingestão calórica, promovendo, dessa forma, o seu bem-estar e saúde. A aplicação foi implementada em Python, com uma interface gráfica desenvolvida através da biblioteca CustomTkinter, proporcionando uma experiência de utilização moderna e intuitiva. Para enriquecer a visualização dos dados, foi integrada a biblioteca Matplotlib, permitindo a geração de gráficos informativos e apelativos. Todos os dados são armazenados e organizados de forma estruturada numa base de dados MySQL, garantindo persistência, segurança e fácil acesso à informação.

# Divisão de tarefas

Interface Gráfica – André Santos

Base de Dados – Gonçalo Marques

Gráfico – Tiago Martins

Exportação e Importação – Enzo Correia

# Interface gráfica

## Estrutura da User interface

A user interface esta dívida em 4 partes a página inicial as definições, a tela de importar / exportar a base de dados e a parte de inserir atividades

A página inicial reage a faceto e um utilizador estar ligado ou não.

|  |  |
| --- | --- |
| Uma imagem com captura de ecrã, texto, software, Software de multimédia  Os conteúdos gerados por IA poderão estar incorretos.  Figura | Uma imagem com texto, captura de ecrã, software, diagrama  Os conteúdos gerados por IA poderão estar incorretos.  Figura |

Caso o utilizador não esteja logado ou não tenha dados associados fica como na imagem 1 após login fica como na imagem 2 podendo os dois botões dos cantos superiores estarem em outras cores

Nesta página estão presentes dois botões um em cada canto onde o da direita leva para as definições e o da direita abre a página das definições e o da esquerda abre um menu tipo dropdown como na imagem 3 onde cada um dos botões tem uma função diferente

|  |  |  |
| --- | --- | --- |
| Uma imagem com texto, captura de ecrã, Tipo de letra, número  Os conteúdos gerados por IA poderão estar incorretos.  Figura | Uma imagem com texto, captura de ecrã, Tipo de letra, software  Os conteúdos gerados por IA poderão estar incorretos.  Figura | Uma imagem com texto, captura de ecrã, Tipo de letra  Os conteúdos gerados por IA poderão estar incorretos.  Figura |

Adicionar registo abre um pop up igual ao da Imagem 4 onde tem um botão para pôr a data do dia e outro para guardar.

Ver registos abre um pop up igual ao da Imagem 5 onde mostra todos os registos de todas as contas.

O botão gráfico serve para atualizar o gráfico.

E o importar/exportar abre a página correspondente.

A página das defensões esta dividida entre duas partes as configurações e a parte do login como na imagem 6.

|  |
| --- |
| Uma imagem com texto, captura de ecrã, software, Software de multimédia  Os conteúdos gerados por IA poderão estar incorretos.  Figura |

No lado esquerdo esta as configurações começado pela primeira altera a cor de fundo de um tema escuro para um tema claro, o segundo abre um pop up onde e possível escolher a cor pretendida para os botões como na imagem 8 e depois aplica a cor e define quando passar o rato por cima ficar uma cor mais escura , o botão de alterar a cor de texto muda entre preto e branco a cor do texto em seguida temos um botão e um sider onde da para mudar o volume e testar com o botão, depois temos o ultimo botão onde vai guardar as escolhas ao utilizador que tenha feito o login.

|  |  |  |
| --- | --- | --- |
| Uma imagem com captura de ecrã, texto, software, Software de multimédia  Os conteúdos gerados por IA poderão estar incorretos.Uma imagem com texto, captura de ecrã, software, Ícone de computador  Os conteúdos gerados por IA poderão estar incorretos.  Figura | Uma imagem com texto, captura de ecrã, design gráfico, Gráficos  Os conteúdos gerados por IA poderão estar incorretos.  Figura | Uma imagem com texto, captura de ecrã, Tipo de letra, Azul elétrico  Os conteúdos gerados por IA poderão estar incorretos.  Uma imagem com texto, captura de ecrã, Tipo de letra, Azul elétrico  Os conteúdos gerados por IA poderão estar incorretos.  Figura |

No outro lado da página temos o login onde tem um processo de verificação onde diz se o utilizador não existe ou se já existe diz também se a palavra-passe esta certa, caso esteja da uma mensagem de bem-vindo.

No último ecrã e para exportar e importar a base de dados que ira ser aprofundado noutra parte do relatório esta página tem 3 botões onde tem o importar exportar e o voltar como na imagem 10

|  |
| --- |
| Uma imagem com texto, captura de ecrã, software, Software de multimédia  Os conteúdos gerados por IA poderão estar incorretos.  Figura |

# Base de dados

Testes e resultados na parte gráfica

|  |  |
| --- | --- |
| Testes | Resultados |
| Funcionalidades dos botões | 100% completo |
| Aplicação de mudança de  cor dos botões | 97% completo erro a aplicar mudas  No ecrã de importação/exportação |
| Mudança de cor de fundo | 100% completo |
| Mudança de volume do som | 100% completo |
| Login | 100% completo |
| Aplicado de gráfico | 100% completo |
| Inserir dados | 100% completo |
| Mostra dados | 100% completo |
| Mudar em ecrãs | 100% completo |
| Pop up dos lembretes | 100% completo |
| Visibilidade dos botões  e letras | 100% completo dependendo da configuração do usuário |

## Estrutura da Base de dados

**Utilizadores**

Guarda os dados de login dos utilizadores.

Campo Tipo Descrição

id INT Identificador único

nome VARCHAR(100) Nome do utilizador (único)

senha VARCHAR(100) Palavra-passe

**Registos**

Regista os dados diários inseridos por cada utilizador.

Campo Tipo Descrição

id INT Identificador do registo

data DATE Data do registo

exercicio INT Minutos de exercício nesse dia

calorias INT Calorias ingeridas nesse dia

user\_id INT ID do utilizador que criou o registo.

A coluna user\_id liga cada registo ao utilizador correspondente, garantindo que os dados são personalizados.

**Preferências**

Foi planeada uma tabela de preferências visuais (como cor dos botões ou fundo) associadas a cada utilizador, mas ainda em desenvolvimento.

Campo Tipo Descrição

user\_id INT Ligação ao utilizador

cor\_fundo VARCHAR Cor de fundo escolhida

cor\_btn VARCHAR Cor dos botões escolhida

## Funcionalidades implementadas

* Login de utilizador e criação de nova conta.
* Registo de atividade: o utilizador pode adicionar:
* Data do registo (validada).
* Minutos de exercício (entre 0 e 1440).
* Calorias ingeridas (entre 0 e 10.000).
* Listagem dos registos com dados apenas do utilizador autenticado.
* Gráfico de distribuição (pizza) dos dados agregados por utilizador.
* O gráfico é automaticamente atualizado após cada novo registo.
* Interface gráfica moderna feita com customtkinter, que permite:
* Alternar entre páginas (principal, definições)
* Aplicar temas visuais.
* Acesso aos dados de forma intuitiva.

## Integração com a Base de Dados

A ligação à base de dados é feita com a biblioteca mysql.connector. Cada função que interage com a base de dados utiliza o método obter\_conexao() definido num módulo externo (db.py). Todas as operações são feitas com boas práticas de segurança e consistência como:

* Uso de with para garantir fecho da conexão
* Uso de placeholders %s para evitar SQL injection
* Conversão e validação de dados antes de inserir

Exemplo de inserção de registo com associação ao utilizador:

cursor.execute("""

INSERT INTO registos (data, exercicio, calorias, user\_id)

VALUES (%s, %s, %s, %s)

""", (data, exercicio, calorias, user\_id))

# Gráfico

A aplicação foi desenvolvida usando a biblioteca matplotlib com o objetivo de apresentar graficamente os dados do utilizador de forma clara e visual. Foi usado um pie chart que representa a distribuição entre o total de minutos de exercício físico e o total de calorias registadas por cada utilizador.

Este gráfico ocorre na função mostrar\_grafico\_do\_user(), que é executada após um utilizador efetue o login. A função obtém os dados necessários a partir da base de dados, somando o total de minutos de exercício e o total de calorias e, em seguida, utiliza a matplotlib para criar uma visualização desses dados.

Os principais componentes da matplotlib utilizados são:

* Figure e add\_subplot: utilizados para criar e configurar a área do gráfico.
* ax.pie(...): responsável por desenhar o pie chart, com as fatias representando os valores de exercício e calorias. Também são definidos os rótulos, as cores das fatias e a apresentação automática de percentagens.
* FigureCanvasTkAgg: permite que o gráfico seja renderizado diretamente na interface gráfica Tkinter, possibilitando a sua visualização dentro do frame reservado ao gráfico.

## Testes e resultados

Ao longo do desenvolvimento da aplicação, foram realizados diversos testes para garantir que o gráfico apresentava corretamente os dados do utilizador. Inicialmente, os testes focaram-se na verificação dos valores representados nas fatias do gráfico, assegurando que os totais de exercício e calorias eram corretamente somados e visualizados.

Durante esse processo, foi também conduzida uma pesquisa para identificar o tipo de gráfico mais adequado à natureza dos dados. Após analisar diferentes alternativas de visualização, concluiu-se que o gráfico de pizza era a melhor opção para representar a distribuição proporcional entre os dois indicadores (exercício e calorias), de forma clara e intuitiva.

Esta escolha teve como objetivo principal proporcionar ao utilizador uma leitura rápida e visual do seu desempenho, reforçando a usabilidade e eficácia da interface gráfica.

# Exportação e Importação

A função serializar(obj) serve para converter datas em texto, porque os ficheiros JSON não conseguem guardar datas diretamente. Então, sempre que aparece uma data, esta função transforma-a num formato que o JSON consegue guardar sem dar erro.

A função exportar\_dados(user\_id=None) é usada para guardar os dados da base de dados em ficheiros .csv e .json. Se for indicado um utilizador (com o user\_id), ela vai buscar só os registos dessa pessoa. Se não for passado nenhum ID, exporta todos os registos. Depois cria dois ficheiros com os dados: um CSV e um JSON. Os nomes dos ficheiros mudam consoante o que foi exportado. No fim, aparece uma mensagem a dizer que correu tudo bem. Se acontecer algum erro, mostra uma mensagem a dizer o que falhou. A ligação à base de dados é sempre fechada, para evitar problemas.

A função importar\_dados(user\_id=None) faz o contrário: lê dados de um ficheiro .csv ou. json e mete-os na base de dados. Primeiro, o utilizador escolhe o ficheiro no computador. Depois o programa lê linha a linha e guarda os dados na tabela registos. Se já existir um registo com o mesmo ID, ele atualiza em vez de duplicar. Se for passado um user\_id, os dados vão todos para esse utilizador. Se não, vai buscar o ID ao próprio ficheiro. Se faltar uma data, ele mete a data de hoje por defeito. Quando acabar, aparece uma mensagem a dizer que foi importado com sucesso. E se foi para um utilizador em específico, o programa atualiza o gráfico com os dados dessa pessoa.

Já a função mostrar\_pagina\_import\_export() serve só para mostrar a página da aplicação onde o utilizador pode importar ou exportar dados. Primeiro esconde as outras páginas do ecrã, e depois mostra esta, onde vão aparecer os botões para importar e exportar.

## Testes realizados

Os testes foram realizados da seguinte forma: com a base de dados aberta, procedeu-se à exportação dos ficheiros nos formatos JSON e CSV através da interface gráfica da aplicação. Após a exportação inicial, foram realizadas alterações nos dados da base de dados, tendo-se efetuado uma nova exportação para verificar se as modificações tinham sido corretamente refletidas nos ficheiros gerados, o que foi confirmado com sucesso.

Para testar a funcionalidade de importação, utilizaram-se os ficheiros previamente exportados, nos quais foram efetuadas alterações manuais. Esses ficheiros foram então importados para a aplicação, e confirmou-se que as alterações foram corretamente inseridas na base de dados.

# Conclusão

O projeto encontra-se totalmente funcional e cumpre os objetivos definidos inicialmente. Os dados são devidamente organizados por utilizador e armazenados de forma segura, garantindo a integridade e confidencialidade das informações. A integração entre a interface gráfica e a base de dados demonstra-se estável e eficiente, proporcionando uma experiência de utilização fluida e responsiva. A aplicação permite, assim, o registo, visualização e análise de dados de forma intuitiva, contribuindo para uma gestão eficaz da informação por parte dos utilizadores.