Politécnico do Porto
Instituto Superior de Engenharia do Porto

Academy +

Desenvolvimento da Interface Interativa da aplicação U=RIsolve Academy

Pedro Miguel Meneses Cruz

LEEC
Licenciatura em Engenharia Eletrotécnica e de Computadores

Departamento de Engenharia Eletrotécnica
Instituto Superior de Engenharia do Porto

Setembro, 2025

Este relatório satisfaz, parcialmente, os requisitos que constam da Ficha de Unidade Curricular de Projeto/Estágio, do 3º ano, da Licenciatura em Engenharia Eletrotécnica e de Computadores.

Candidato: Pedro Miguel Meneses Cruz, Nº 1141265, `1141265@isep.ipp.pt`

Orientação Científica: André Rocha, `anr@isep.ipp.pt`

Coorientação Científica: João Ferreira, `jfa@isep.ipp.pt`

ISEP INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO

Departamento de Engenharia Eletrotécnica
Instituto Superior de Engenharia do Porto
Rua Dr. António Bernardino de Almeida, 431, 4200–072 Porto

Setembro, 2025

Agradecimentos

Gostaria de agradecer às pessoas que me ajudaram e apoiaram na realização deste projeto.

Em primeiro lugar, quero agradecer ao meu orientador, André Rocha, pela oportunidade de desenvolver este projeto e pela possibilidade de crescimento dos meus conhecimentos que me proporcionou. Agradeço também pelas reuniões de preparação e enquadramento do projeto e pelo apoio na elaboração do relatório.

Gostaria ainda de agradecer ao meu coorientador, João Ferreira, pelo seu apoio e acompanhamento na fase de desenvolvimento. A sua disponibilidade e o seu conhecimento técnico da plataforma foram fundamentais, não só nos primeiros passos, mas também ao longo de todo o desenrolar do projeto.

Por último, estou grato à minha família pelo apoio emocional prestado durante este período. Pelo suporte e compreensão da minha ansiedade e frustração.

Resumo

Este relatório apresenta o trabalho desenvolvido na criação de uma nova versão da interface de utilizador da plataforma educativa *U=RIsolve Academy*, sendo o objetivo principal promover uma experiência de utilização mais envolvente, moderna e funcional. Para tal, foi desenvolvido um conjunto de novas funcionalidades, das quais se destacam: a gestão de perfil do utilizador, um modo de treino melhorado (que permite a seleção de tópicos e oferece um sistema de *feedback*), bem como a inclusão de elementos de gamificação, tais como conquistas, pontos e tabelas classificativas.

Este projeto contempla o desenvolvimento de novos elementos visuais e também definir uma estrutura de metadados associada aos utilizadores e aos diferentes componentes da aplicação. A aplicação adopta uma arquitetura cliente-servidor, com base no padrão *Model-View-Controller* (MVC), com recurso às tecnologias: *JavaScript* (JS) (*front-end* e *back-end*), *Handlebars*, *Cascading Style Sheets* (CSS) e *MySQL*.

A introdução das novas funcionalidades tem o intuito de tornar o uso da plataforma mais interessante, sobretudo para estudantes. Onde estas novidades podem ajudar a ter um acompanhamento mais contínuo às disciplinas e permitem que cada aluno concentre o estudo nas áreas em que sente mais dificuldades.

Palavras-Chave: *U=RIsolve Academy*, gamificação, plataforma educativa, aplicação, *Node.js*, arquitetura MVC.

Abstract

This report show the work developed in the making of a new user interface version for the educatinal platform U=RIsolve Acadamy, with the main objective to promote a user expirience more envolving, modern and functional. For that, a set of new functionalities were developed, of which stand out: user profile management, an improved practice mode (that allows the selection of topics and offers a feedback system), aswell the addition of gamification elements, such as conquests, points and leaderboards.

To put these features into practice, it was necessary develop new visual elements and define a metadata struture associated to the users and the multiple aplication componentes. The aplication adopted a client-server architecture, based on pattern *Model-View-Controller* (MVC), using the following technologies: *JavaScript* (JS) (front-end and back-end), Handlebars, *Cascading Style Sheets* (CSS) and MySQL.

The introduction of new features aims to make using the platform more interesting, especially for students. Where these new features can help provide more continuous monitoring of subjects and allow each student to focus their studies on the areas in which they experience the most difficulty.

Keywords: *U=RIsolve Academy*, gamification, educational platform, application, *Node.js*, architecture MVC.

Índice

Lista de Figuras **vii**

Lista de Tabelas **ix**

Lista de Acrónimos **xi**

1 Introdução **1**
1.1 Contextualização 1
1.2 Descrição do Projeto 2
1.3 Calendarização 2
1.4 Organização do Relatório 3

2 Enquadramento Tecnológico **5**
2.1 Estado da Arte 5
2.2 Gamificação 8
2.3 Arquitetura Cliente-Servidor e Padrão MVC 8

3 Tecnologias de Desenvolvimento **9**
3.1 Tecnologias Utilizadas 9
3.2 Linguagens e bibliotecas de programação 9
3.3 Ferramentas de Apoio e Desenvolvimento 10
3.3.1 Ferramentas de organização 10
3.3.2 Ambiente de desenvolvimento integrado 11
3.3.3 Docker 11
3.3.4 Ferramentas de AI 11

4 Implementação **13**
4.1 Arquitetura da Aplicação 13
4.2 Estrutura Funcional e Geral 14
4.3 Implementação das Funcionalidades 15
4.3.1 Perfil do utilizador 16
4.3.2 Modo de Treino Melhorado 21
4.3.3 Elementos de Gamificação 28
4.4 Metadados 30

5 Conclusões **35**
5.1 Reflexão Final 35
5.2 Competências Adquiridas 36
5.3 Trabalho Futuro 37

Referências **38**

Lista de Figuras

1.1 Cronograma do projeto 2

2.1 Diagrama do padrão MVC [10] 8

4.1 Arquitetura da Aplicação. 14
4.2 Diagrama de árvore da aplicação. 15
4.3 Interface de Registo. 16
4.4 Interface de confirmação do registo de conta. 17
4.5 Interface de *Login*. 17
4.6 Interface de Recuperação de Palavra-passe. 18
4.7 Interface de Gestão de Conta. 19
4.8 Menu de Perfil. 19
4.9 Página de Estatísticas do utilizador. 20
4.10 Interface de Rastreio de Tópico. 20
4.11 Interface de prestação Semanal por Subtópico. 21
4.12 Fluxograma do modo de treino. 22
4.13 Interface de Treino. 23
4.14 Fluxograma de preparação da sessão de questionário. 23
4.15 Fluxograma da sessão de questionário. 24
4.16 Interface de Questionário. 25
4.17 Indicação de resposta selecionada. 26
4.18 Diagrama de submissão de resposta. 26
4.19 Indicação de resposta correta. 27
4.20 Indicação de resposta errada. 27
4.21 Apresentação da mensagem de *feedback*. 28
4.22 Página de Sessão de Questionário Completo. 28
4.23 Nível e Barra de Experiência do utilizador. 29
4.24 Tabela Classificativa. 30
4.25 Diagrama relacional. 32
4.26 Diagrama relacional do Modo de Treino. 33

Lista de Tabelas

2.1 Comparação entre Khan Academy, Duolingo e Mimo. 7

4.1 Resumo de metadados das tabelas relacionadas com modo de treino. 32

Lista de Acrónimos

CSS	*Cascading Style Sheets*
HTML	*HyperText Markup Language*
HTTP	*Hypertext Transfer Protocol*
IDE	*Integrated Development Environment*
JS	*JavaScript*
JSON	*JavaScript Object Notation*
LEEC	Licenciatura em Engenharia Eletrotécnica e de Computadores
MVC	*Model-View-Controller*
SASS	*Syntactically Awesome Stylesheet*
SQL	*Structured Query Language*
VS Code	*Visual Studio Code*
XP	*Experience Points*

Capítulo 1

Introdução

O trabalho retratado no presente relatório foi realizado no âmbito curricular de Projeto/Estágio da Licenciatura em Engenharia Eletrotécnica e de Computadores (LEEC) do ISEP.

No relatório é descrito o desenvolvimento de funcionalidades adicionais para a plataforma *U=RIsolve Acadamy* [1], com o intuito de promover uma experiência mais moderna, envolvente e funcional.

1.1 Contextualização

Este projeto dá continuidade ao desenvolvimento da plataforma *U=RIsolve Academy*, criada com o propósito de apoiar estudantes de eletrotecnia com ferramentas fundamentais, como o osciloscópio virtual e o editor de circuitos, que são essenciais para a aprendizagem na área de engenharia eletrotécnica.

Nos tempos de hoje, em que a tecnologia está mais presente no dia a dia e em constante e rápida evolução, o trabalho realizado visa oferecer uma experiência mais envolvente, com novas funcionalidades e moderna.

Não se pode deixar de realçar as dificuldades de adaptação que os estudantes enfrentam no início do curso superior [2], e as plataformas educativas podem ajudar, proporcionando recursos e ferramentas para uma aprendizagem mais personalizada e iterativa, facilitando o acompanhamento das unidades curriculares [3].

1.2 Descrição do Projeto

Este projeto tem como objetivo a criação de uma nova versão de interface de utilizador da plataforma *U=RIsolve Academy*. A nova versão do software tem como objetivo melhorar a experiência do utilizador, promovendo um maior envolvimento dos estudantes e motivando-os para a realização de atividades ligadas ao estudo da análise de circuitos elétricos. A proposta pretende facilitar o rastreio da prestação individual, melhorar o modo de treino e introduzir elementos de gamificação, procurando motivar o uso mais regular com a criação de metas extras.

Objetivos

O projeto tem os seguintes objetivos para a implementação das novas funcionalidades:

- **Perfil do utilizador** – desenvolver uma área para gestão da conta e estatísticas do utilizador;
- **Modo de treino melhorado** – permitir a seleção de tópicos, melhorar a interface para questões e incorporar sistemas de feedback e ajuda;
- **Elementos de gamificação** – introduzir sistemas de motivação como conquistas, sistema de pontos, barras de progresso e tabelas de liderança.

1.3 Calendarização

Para garantir uma boa gestão do tempo e o cumprimento dos objetivos, o projeto foi dividido nas seguintes fases: pré-estudo, estudo do estado da arte, implementação e redação do relatório. A Figura 1.1 apresenta o cronograma das principais etapas do projeto.

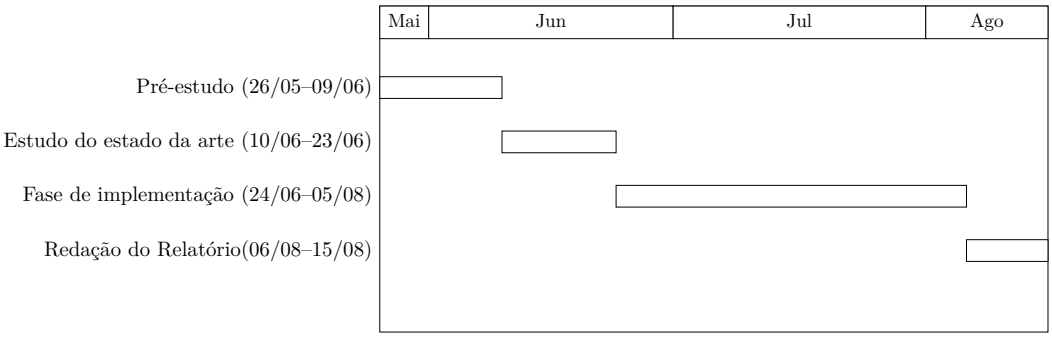

Figura 1.1: Cronograma do projeto

1.4 Organização do Relatório

Este capítulo apresenta a organização deste relatório, com breves descrições do conteúdo de cada capítulo. O presente relatório está organizado da seguinte forma:

- **Capítulo 2 – Enquadramento Tecnológico:** Analisa o estado da arte, descreve as principais tecnologias utilizadas e apresenta a arquitetura da aplicação.
- **Capítulo 3 – Tecnologias de Desenvolvimento:** Detalha as linguagens, bibliotecas, ferramentas e metodologias empregues no desenvolvimento, bem como o ambiente de trabalho utilizado.
- **Capítulo 4 – Implementação:** Descreve as principais funcionalidades desenvolvidas, a estrutura de dados e a interface da aplicação.
- **Capítulo 5 – Conclusões:** Apresenta uma reflexão final sobre o trabalho realizado, as competências adquiridas e sugestões para melhorias e projetos futuros.

Capítulo 2

Enquadramento Tecnológico

2.1 Estado da Arte

O uso de plataformas educativas com elementos de gamificação tem atraído atenção devido ao seu potencial de aumentar a motivação dos estudantes e favorecer aprendizagens mais significativas. Elementos como pontuações, desafios e recompensas são apontados como estratégias eficazes para promover o envolvimento dos alunos ao longo do processo de aprendizagem [4]. Neste capítulo, pretende-se apresentar algumas tendências relacionadas com o uso da gamificação em plataformas educacionais e comparar as diferentes abordagens.

Plataformas Educativas

Khan Academy

A Khan Academy [5] é uma plataforma de aprendizagem focada em matemática, do pré-escolar ao 12.º ano, e ciência física, química e biologia. Apresenta ao utilizador a escolha dos tópicos de aprendizagem, organizados por áreas temáticas (como pré-escolar, geometria, fundamentos de álgebra, etc.) ou por ano de escolaridade.

Cada tópico está dividido por subtemas e é apresentada uma breve introdução, incluindo informações sobre os requisitos prévios necessários para sua compreensão e estudo. Os subtemas estão organizados em duas áreas, "Aprender" e "Praticar":

- Na secção "Aprender" é feito o ensino do subtema através de vídeos e descrições detalhadas;

- Na secção "Praticar" são propostos ao utilizador exercícios interativos para testar e consolidar os conhecimentos adquiridos.

A Khan Academy também incorpora elementos de gamificação para incentivar o progresso e motivar os utilizadores. À medida que os utilizadores resolvem exercícios, vão ganhando medalhas, conquistas, pontos de energia e níveis de conta que funcionam como recompensas simbólicas pelo seu desempenho.

O acompanhamento do progresso é feito através de uma tabela de atividade do utilizador que apresenta dados por subtema, incluindo data, nível de proficiência, evolução, número de respostas acertadas por número total de exercícios realizados e tempo investido em cada temática.

Duolingo

O Duolingo [6] é uma plataforma de aprendizagem linguística, com cursos adaptados à língua nativa de cada utilizador. Antes da criação da conta, é realizada uma avaliação diagnóstica para aferir o conhecimento prévio da língua que o utilizador deseja aprender ou aprimorar. Com base nos resultados, é determinada a secção onde o utilizador irá iniciar a sua aprendizagem.

O Duolingo incorpora fortemente elementos de gamificação na sua interação com o utilizador. As secções de aprendizagem são apresentadas como uma progressão linear de objetivos (níveis), subdivididas por temáticas (natureza, saúde, relacionamentos, entre outras).

Cada exercício é acompanhado por uma barra de progresso, que segue a execução total da atividade. No final, o utilizador recebe pontos de experiência, *Experience Points* (XP), funcionando como recompensa pelo seu desempenho.

Como forma de penalização, o utilizador tem um limite de respostas erradas, representado por vidas. Cada utilizador tem, no máximo, cinco vidas, sendo recarregada uma a cada cinco horas.

A plataforma apresenta também um sistema de missões diárias, que recompensam com XP, e o conceito de *streaks* (sequências de dias de utilização), o que incentiva o uso contínuo e diário da plataforma. Existe ainda um sistema de liga, que adiciona uma componente competitiva entre utilizadores, promovendo o envolvimento dos utilizadores com a plataforma.

Mimo

A Mimo [7] é uma plataforma de aprendizagem na área da programação e do desenvolvimento web. A plataforma apresenta ao utilizador diferentes "Caminhos", divididos em dois tópicos:

- **Carreira**: Desenvolvimento *Full-Stack*, *Python*, *Back-End* e *Front-End*;

- **Linguagem**: *JavaScript*, *TypeScript*, *React*, *Cascading Style Sheets* (CSS), *Structured Query Language* (SQL), *HyperText Markup Language* (HTML) e *Python*.

A interação com o utilizador baseia-se em elementos de gamificação. Cada "Caminho" está dividido em temas sequenciais, sendo que cada tema apresenta uma progressão linear de subtemas (níveis).

Os subtemas incluem uma breve descrição do objetivo de cada lição, seguida de exercícios para consolidar a temática abordada. O progresso é realizado à medida que o utilizador responde corretamente aos exercícios. Respostas corretas em sequência conferem um bónus de pontos de experiência XP no final do subtema. Em contrapartida, uma resposta errada resulta na penalização de um "coração". O utilizador tem, no máximo, cinco "corações", sendo que cada um é recarregado ao fim de uma hora e trinta minutos.

O utilizador é incentivado a interagir com a plataforma diariamente através de *streaks* (sequência de uso diário) e de um ranking semanal. A utilização contínua é recompensada com XP extra e unidade monetária virtual, que pode ser trocada por benefícios adicionais, como "corações" extras ou ganho duplo de XP.

Comparação das Plataformas Educativas

Após a análise individual das características de cada plataforma, a Tabela 2.1 apresenta uma comparação sumária das plataformas, bem como a aplicabilidade dos elementos de gamificação de cada uma no presente projeto.

Tabela 2.1: Comparação entre Khan Academy, Duolingo e Mimo.

Parâmetro	**Khan Academy**	**Duolingo**	**Mimo**
Tipo de Conteúdo Principal	Matemática e Ciências	Línguas	Programação e Web Dev
Elementos de Gamificação	Medalhas, pontos, níveis	XP, vidas, missões, streaks, ligas	XP, corações, streaks, ranking semanal, moeda virtual
Estrutura de Progressão	Tópicos e Subtemas	Secções e Níveis	Caminhos, Temas e Subtemas
Aplicabilidade no Projeto	Alta — a estrutura de progressão semelhante aos objetivos e a inexistência de penalidades enquadra-se no ambiente da plataforma.	Aplicável — o sistema de vidas pode gerar frustração; no entanto, os elementos de progressão linear, XP, streaks e ligas podem ser motivadores extra.	Aplicável — o sistema de corações pode não ser o mais adequado; porém, o uso de moeda virtual pode recompensar e motivar a utilização da plataforma.

2.2 Gamificação

Atualmente, é possível tornar os processos educativos mais atrativos e envolventes recorrendo a conceitos encontrados em jogos. A gamificação tem o objetivo de motivar o utilizador apelando ao seu sentido social, explorador ou conquistador [8].

Os elementos mais comuns de gamificação são pontos, níveis, medalhas, tabelas classificativas, desafios e *feedback* imediato. Estes elementos pretendem criar um ambiente interativo, competitivo e estimulante para quem interage com as plataformas que os executam.

Assim sendo, a gamificação pode contribuir significativamente no contexto educativo, aumentando o envolvimento e a motivação dos estudantes.

2.3 Arquitetura Cliente-Servidor e Padrão MVC

A aplicação segue o modelo cliente-servidor, onde o cliente (navegador) comunica com o servidor por meio de pedidos *Hypertext Transfer Protocol* (HTTP). O padrão *Model-View-Controller* (MVC) foi o escolhido previamente para a organização da aplicação. Este padrão permite a separação entre a lógica do servidor (*back-end*), controlo e manipulação dos dados e a apresentação(*front-end*), facilitando a manutenção e escalabilidade da aplicação. O padrão está dividido em 3 camadas, *model*, *view* e *controller*. Na camada *model* encontra-se a lógica de dados, a forma como os dados são criados, guardados e manipulados. A camada *view* é responsável pela interface do utilizador, e a camada *controller* é intermediária entre as camadas *model* e *view* [9]. A Figura 2.1 ilustra o padrão MVC [10].

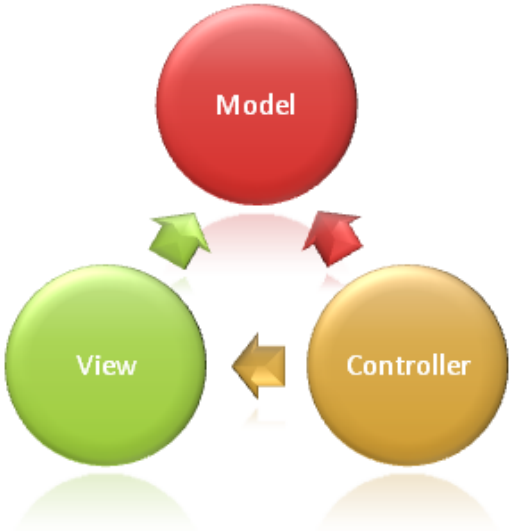

Figura 2.1: Diagrama do padrão MVC [10]

Capítulo 3

Tecnologias de Desenvolvimento

3.1 Tecnologias Utilizadas

As tecnologias base utilizadas neste projeto foram previamente definidas no âmbito do projeto original *U=RIsolve Academy*. Assim, a escolha de *Node.js*, *MySQL*, *Handlebars*, CSS e *Docker* resulta da continuidade e integração com a arquitetura já existente. O presente trabalho focou-se na adição de novas funcionalidades na plataforma, respeitando as tecnologias e padrões previamente estabelecidos.

3.2 Linguagens e bibliotecas de programação

No desenvolvimento deste projeto foram utilizadas as linguagens *JavaScript* (JS), *HyperText Markup Language* (HTML), *Cascading Style Sheets* (CSS) e *Structured Query Language* (SQL), usadas no desenvolvimento de aplicações web. Estas linguagens são um padrão no desenvolvimento web e na gestão de bases de dados, e são a base na qual o trabalho que originou este projeto foi desenvolvido e, como tal, são o seu ponto de partida.

A plataforma *U=RIsolve Academy* foi desenvolvida em *Node.js*, que permite a execução de JS no lado do servidor, possibilitando a criação de aplicações web dinâmicas.

A gestão e criação da base de dados do projeto foram executadas recorrendo ao sistema de gestão de bases de dados *MySQL*, baseado na linguagem SQL.

Para a estruturação e criação dos estilos da interface da aplicação, foram utilizados o *Handlebars*, uma linguagem de *templating* que gera ficheiros HTML com os dados a apresentar ao utilizador, e o *Syntactically Awesome Stylesheet* (SASS), uma extensão de CSS que, através de funcionalidades extras, permitiu o uso de variáveis. HTML facilitou a organização e estruturação dos elementos na página e, por sua vez, o CSS permitiu a personalização estética desses mesmos elementos.

Para além das linguagens de programação referidas anteriormente, foram também utilizadas bibliotecas e *frameworks* de suporte. Estas facilitaram o desenvolvimento da aplicação, acrescentando funcionalidades. Entre as principais bibliotecas utilizadas, destacam-se:

- express - Responsável por gerir as rotas, sessões e autenticação;
- bcrypt - Método de encriptação usado para proteger palavras-passe;
- i18next - Permite que o texto da aplicação seja facilmente traduzido em várias linguagens;
- *JavaScript Object Notation* (JSON) - Utilizado para guardar e ler as traduções e fazer a troca de dados entre o *front end* e o *back end*.

3.3 Ferramentas de Apoio e Desenvolvimento

Esta secção apresenta os principais recursos de organização, comunicação e desenvolvimento utilizados na realização deste projeto. São descritas as funcionalidades usadas de cada ferramenta, bem como o seu enquadramento de utilização.

3.3.1 Ferramentas de organização

Neste projeto, foram utilizadas ferramentas de gestão de tarefas e controlo de versões de *software*.

Para a gestão de tarefas foi utilizado o *Trello* [11]. Esta plataforma permite a organização de ideias, conceitos ou projetos, que agiliza a gestão e a comunicação entre várias equipas. Possibilitou organizar e partilhar ideias, planear e rastrear tarefas.

O controlo de versões do software recorreu-se ao *GitHub* [12]. Foi adotado a convenção de versionamento semântico (vX.Y.Z), no qual o primeiro algarismo significa versão de desenvolvimento, o segundo algarismo a adição ou alteração significativa de funcionalidades, e o último algarismo alterações menores e correções de erros.

A aplicação funcionou como repositório do projeto e ofereceu um conjunto de ferramentas que facilitam o trabalho em equipa. Permitiu acompanhar, analisar e atualizar sem modificar ou perder o código-fonte, ferramentas imprescindíveis em projetos de grandes dimensões e em equipa.

3.3.2 Ambiente de desenvolvimento integrado

Para *Integrated Development Environment* (IDE) foi utilizado o *Visual Studio Code* (VS Code), uma plataforma de desenvolvimento robusta que suporta as várias línguas do projeto, e com acesso a várias extensões que ajudaram a simplificar o processo de desenvolvimento. A interface é apelativa, inclui navegador de documentos e permite trabalhar com múltiplos ficheiros em simultâneo.

3.3.3 Docker

O *Docker* é uma plataforma que permite o desenvolvimento de aplicações em ambientes virtuais, chamados contentores. Estes contentores estão isolados do sistema em que o *Docker* é executado, permitindo o desenvolvimento isolado de aplicações. Desta forma, é possível ter várias equipas a trabalhar em diferentes áreas da aplicação, sem conflitos e separado da versão implementada.

No projeto, o *Docker* foi utilizado para manter o ambiente da aplicação isolado do resto do sistema e pela conveniência de ter as dependências necessárias para correr a aplicação num contentor.

3.3.4 Ferramentas de AI

Durante o desenvolvimento do projeto, foi utilizada a ferramenta *GitHub Copilot*, através da extensão do VS Code.

O *Github Copilot* apoiou na simplificação do código com sugestões de soluções e auxiliou na resolução de problemas. Para além de simplificar o processo de escrita, o *GitHub Copilot* disponibiliza a funcionalidade de escrita preditiva, acelerando tarefas monótonas, como adicionar valores-chave a ficheiros JSON.

Capítulo 4

Implementação

4.1 Arquitetura da Aplicação

A arquitetura do sistema segue o modelo de aplicação *web* de camadas, baseado no padrão MVC, garantindo a separação de responsabilidades e facilidade de manutenção. A Figura 4.1 ilustra os principais agentes do sistema, *front end*, *back end* e base de dados.

Entre *front end* e *back end*, existe a troca de dados em formato JSON através de pedidos HTTP, como respostas a questões, progresso do utilizador, autenticação e pedidos de dados para renderização de páginas.

A comunicação entre *back end* e a base de dados na troca de dados estruturados via SQL, destacando-se inserção e leitura de utilizadores, tópicos, perguntas, progresso e estatísticas.

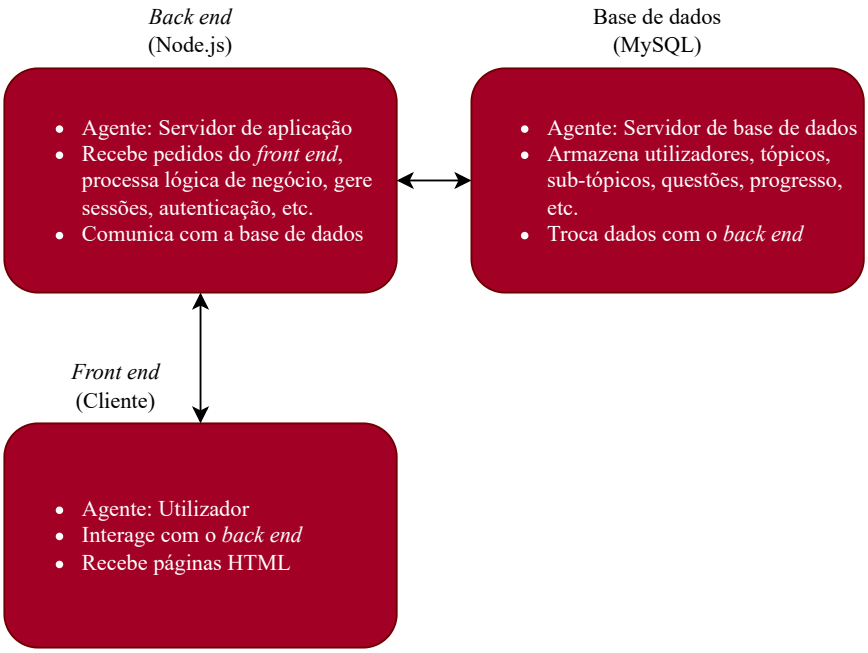

Figura 4.1: Arquitetura da Aplicação.

4.2 Estrutura Funcional e Geral

A aplicação está organizada em três blocos funcionais principais:

1. Perfil de utilizador – gere o registo, autenticação, perfil e progressão do utilizador;
2. Modo de treino – apresenta tópicos, permite sessões de questionário e atualiza o progresso;
3. Gamificação – recompensa e motiva a utilização.

Esta divisão permite uma melhor compreensão das principais funcionalidades e como interagem para oferecer uma experiência completa ao utilizador.

Após a identificação dos principais blocos funcionais, importa compreender a estrutura técnica modular da aplicação. O *back end* contém o código executado pelo servidor, desempenhando funções como responder a pedidos HTTP, renderização das *views* e comunicar com a base de dados.

O *front end* contém toda a lógica que ocorre no lado do cliente, com dois recursos principais, *views* e *public*. As *views* contêm os ficheiros *Handelbars* utilizados para renderizar HTML, já *public* contém os ficheiros JS, código que corre no lado do cliente, as imagens e sons usados na aplicação e definição dos estilos. Na Figura 4.2 está representado o diagrama de árvore da estrutura da aplicação.

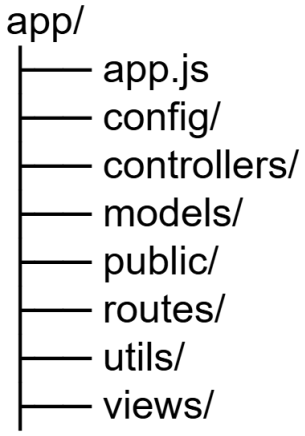

Figura 4.2: Diagrama de árvore da aplicação.

O ficheiro `app.js` é responsável pela configuração inicial e arranque do servidor. As configurações de ligação à base de dados estão definidas na pasta `config/`.

A lógica de *back end* e de processamento dos pedidos dos utilizadores é executada pelos controladores, na pasta `controllers/`. Os modelos na pasta `models/` definem a estrutura dos dados e contêm a lógica de interação com a base de dados.

Os recursos utilizados no *front end*, como CSS, imagens, ficheiros JS e sons, estão na pasta `public`.

As rotas de aplicação, que determinam o encaminhamento dos pedidos HTTP, estão organizadas na pasta `routes/`.

As ferramentas utilitárias encontram-se na pasta `utils`, enquanto a pasta `middleware/` contém o módulo de autenticação de pedidos. As internacionalizações da interface estão na pasta `locales/`, com os ficheiros de tradução de diferentes idiomas.

Por fim, a pasta `views/` contém os ficheiros *Handlebars* responsáveis pela renderização das páginas e elementos visuais apresentados ao utilizador, organizados em `layouts/`, `pages/` e `parcials/`.

4.3 Implementação das Funcionalidades

Nesta secção é abordada a implementação de cada bloco funcional e a integração de elementos de gamificação da aplicação.

4.3.1 Perfil do utilizador

O perfil do utilizador é uma componente central da plataforma, permite a gestão da conta, o acesso às funcionalidades e o acompanhamento do progresso e desempenho individualizado. A aplicação disponibiliza mecanismos de autenticação, edição de dados pessoais e visualização de estatísticas, organizados em três áreas principais: autenticação, gestão de conta e estatísticas.

1. **Registo e Autenticação**

   Para aceder às novas funcionalidades, é necessário criar uma conta de utilizador, fornecendo dados básicos de identificação e autenticação. Para a página de registo, foi utilizado um design simples, com elementos visuais reconhecíveis e mantendo a identidade da plataforma. A Figura 4.3 apresenta o aspeto da página de registo.

Figura 4.3: Interface de Registo.

Após completar o Registo, é apresentada a página de confirmação, Figura 4.4, e o utilizador pode aceder à plataforma através da página de *Login*, inserindo as suas credenciais.

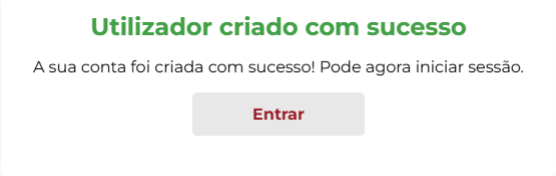

Figura 4.4: Interface de confirmação do registo de conta.

Para assegurar uma experiência consistente, é mantido os elementos visuais da página de Registo, como mostra a Figura 4.5.

Figura 4.5: Interface de *Login*.

Caso o utilizador se esqueça da palavra-passe, pode recorrer à página de recuperação de conta através do link "Esqueceu-se da palavra-passe" na página de *Login* (Figura 4.5). É possível alterar os dados de autenticação de conta, ao indicar dados de identificação. A Figura 4.6 apresenta a página de recuperação de palavra-passe, bem como os dados necessários para a sua recuperação.

Figura 4.6: Interface de Recuperação de Palavra-passe.

Segurança

As palavras-passe dos utilizadores são protegidas através de *hashing*, sendo transformadas usando o módulo *bcrypt* antes de serem armazenadas na base de dados. A aplicação utiliza o *framework express* para a gestão de sessões, que mantêm o cliente autenticado e protegem as rotas restritas apenas para utilizadores. Após um período de uma hora de inatividade, a sessão é terminada automaticamente. Os dados dos formulários são autenticados no *back end* e as palavras-passe nunca são armazenadas no lado do cliente, garantindo a segurança da informação dos utilizadores.

2. **Gestão de Conta** O utilizador tem acesso à página de gestão, representada na Figura 4.7, onde pode alterar as suas informações, como e-mail e o nome da conta.

Figura 4.7: Interface de Gestão de Conta.

O acesso à página está disponível no menu de perfil, clicando no avatar situado no topo direito da página, como demonstra a Figura 4.8. Este menu também permite aceder ao menu de configurações da interface da aplicação especificas a cada utilizador, bem como ao botão que permite terminar a sessão.

Figura 4.8: Menu de Perfil.

3. **Página de estatísticas**

   Cada utilizador pode acompanhar a atividade do modo de treino na página de estatísticas. A página está dividida em 3 contentores: performance geral, gráfico radial, e uma área para mostrar as conquistas do utilizador, como mostra a Figura 4.9. No contentor de performance geral, é possível visualizar a pontuação semanal, o gráfico de desempenho geral, representado por relâmpagos (0-5), e a tabela de desempenho por tópico, onde é possível ver o progresso em cada tópico, o número de subtópicos completos e o desempenho por cada tópico.

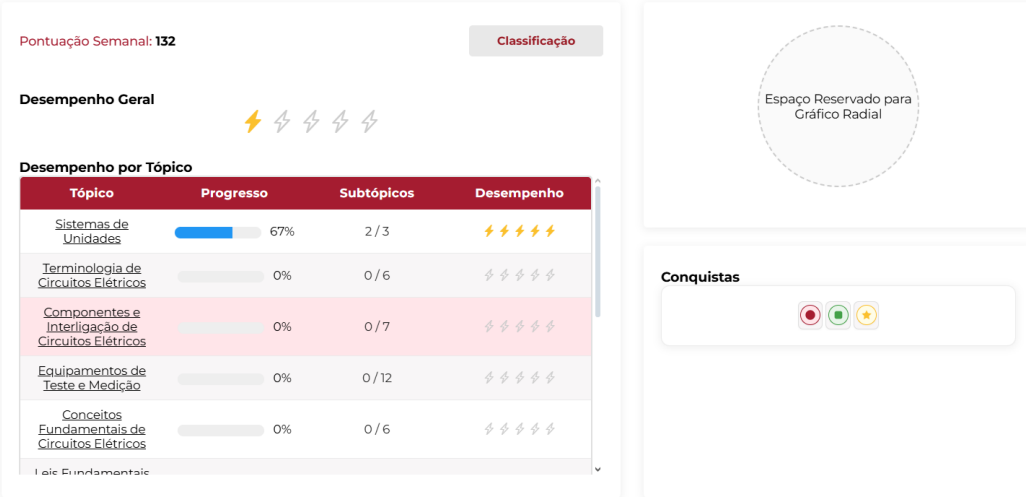

Figura 4.9: Página de Estatísticas do utilizador.

Ao carregar no título de um tópico, na tabela, o utilizador será redirecionado para a página de rastreio do tópico, onde é apresentado o desempenho de cada subtópico pertencente ao tópico escolhido, demonstrado na Figura 4.10. .

Subtópico	Tentativas Totais	Corretas Totais	Pontos Totais	Desempenho
Unidades SI	27	11	54	
Múltiplos e submúltiplos	25	12	50	
Algarismos significativos	30	12	48	

Figura 4.10: Interface de Rastreio de Tópico.

É possível ver a prestação semanal de cada subtópico ao carregar no título, como é apresentado na Figura 4.11

Subtópico	Tentativas Totais	Corretas Totais	Pontos Totais	Desempenho
Unidades SI	27	11	54	

Semana	Tentativas	Corretas	Pontos	Desempenho
28/07/2025	6	3	24	
04/08/2025	9	4	18	
11/08/2025	12	4	12	

Múltiplos e submúltiplos	25	12	50	
Algarismos significativos	30	12	48	

Figura 4.11: Interface de prestação Semanal por Subtópico.

4.3.2 Modo de Treino Melhorado

A Figura 4.12 apresenta o fluxograma do funcionamento do bloco do modo de treino. O processo inicia-se com a consulta à base de dados para carregar a lista de tópicos e subtópicos, seguido-se da sua apresentação ao utilizador. O utilizador pode explorar os diferentes tópicos. Assim que desejar, pode iniciar uma sessão de questionário sobre o tópico ou subtópico selecionado.

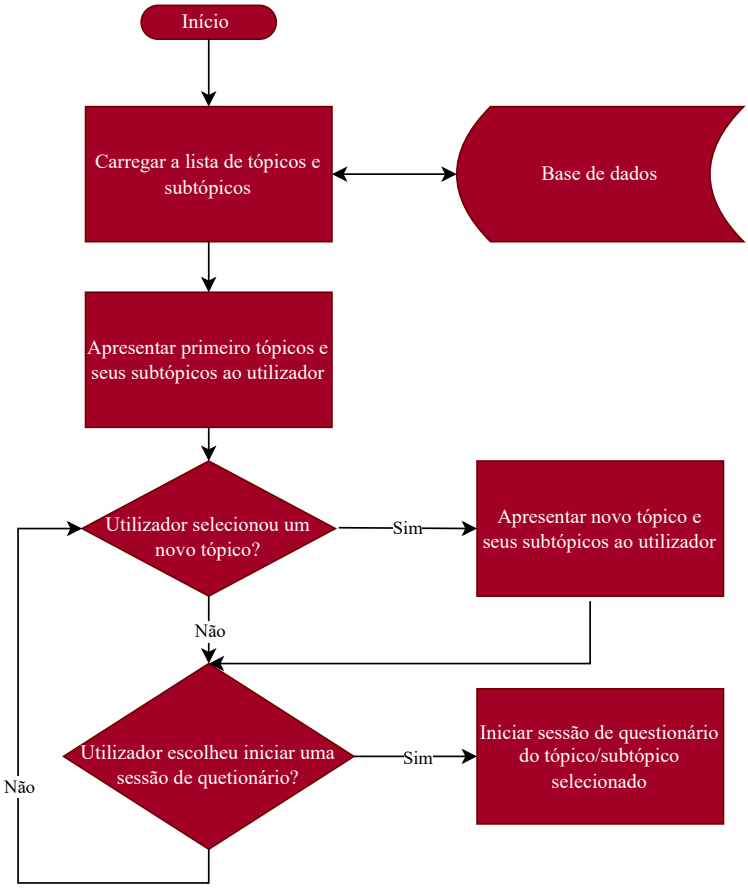

Figura 4.12: Fluxograma do modo de treino.

O modo de treino apresenta ao utilizador a lista de tópicos disponíveis para treino. Na Figura 4.13 podemos observar que ao escolher um tópico, é apresentada uma breve introdução e a lista de subtópicos, cada um com a sua descrição e uma imagem opcional. Nesta página, o utilizador pode iniciar a sessão de questionário, carregando no botão de "Iniciar" do subtópico pretendido, ou no botão "Iniciar Tópico" se pretender sessão com questões de todos os subtópicos.

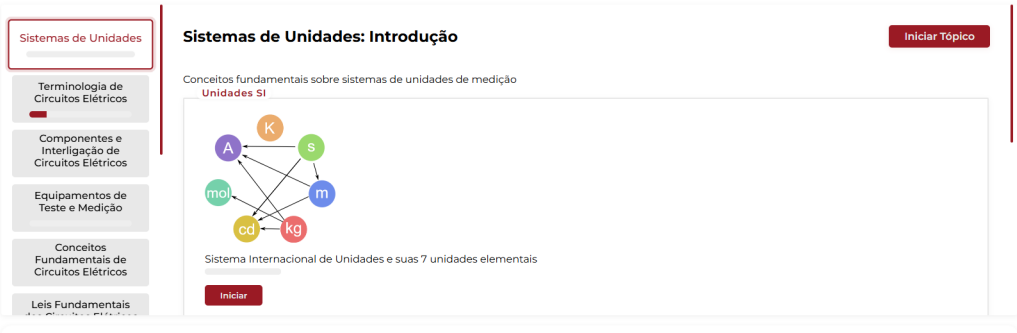

Figura 4.13: Interface de Treino.

Sessão de questionário

Antes de apresentar a sessão de questionário ao utilizador, o servidor prepara as questões a serem usadas. A Figura 4.14 mostra o fluxograma simplificado de preparação da sessão de questionário.

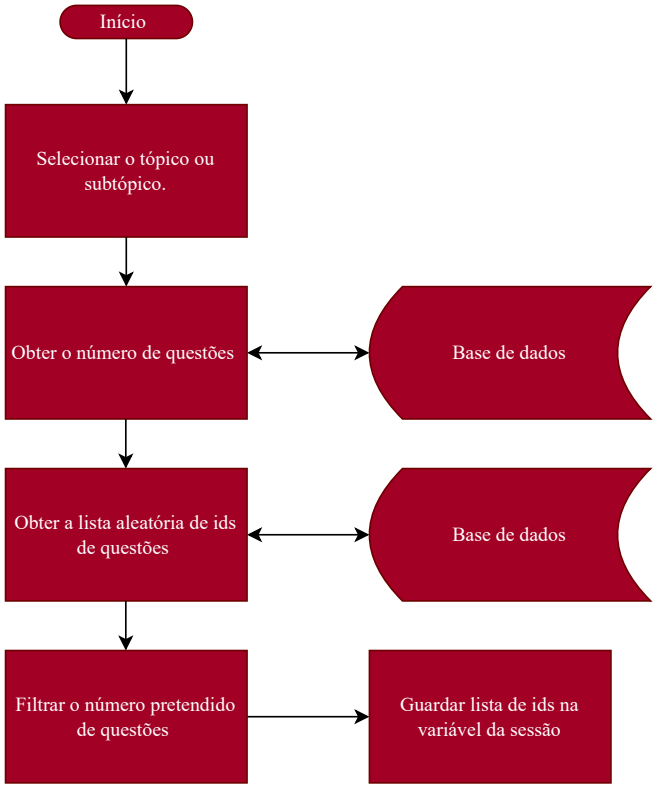

Figura 4.14: Fluxograma de preparação da sessão de questionário.

Quando se iniciar uma sessão de questionário, é obtido o número de questões através de uma chamada à base de dados. No caso de ser selecionado um tópico, o número de perguntas é igual ao número de subtópicos. No outro caso, se for selecionado um subtópico, esse número é previamente definido durante o registo de cada subtópico na base de dados. Em seguida, é obtida uma lista aleatória de identificadores (id) das questões, filtrando o número pretendido e, por fim, é guardada na variável para posteriormente ser usada para obter os dados das questões.

Após a preparação inicial, a aplicação corre a rotina da sessão de questionário, ilustrada na Figura 4.15.

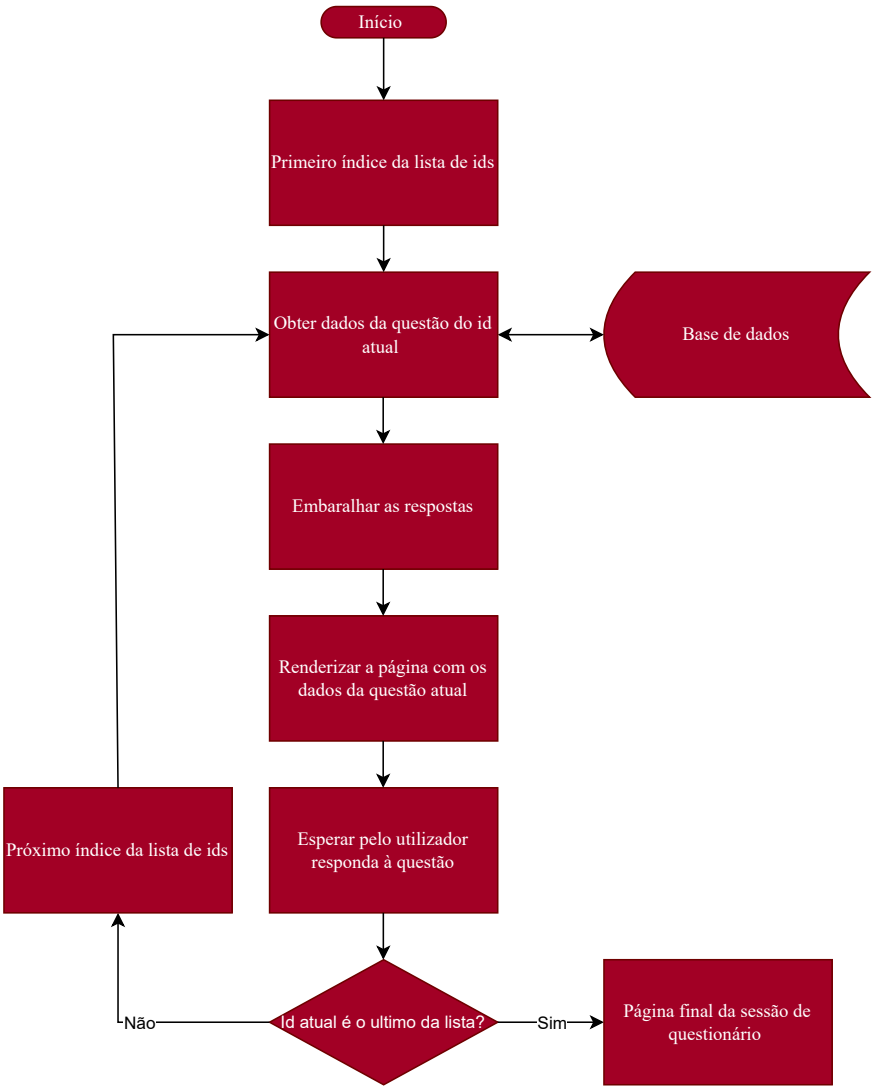

Figura 4.15: Fluxograma da sessão de questionário.

São obtidos e preparados os dados da primeira questão e a página da sessão de

questionário é renderizada e apresentada ao utilizador. Após o utilizador responder corretamente à questão, é verificada a existência de mais questões na sessão. Caso se verifique a existência da próxima questão, os dados da nova questão são tratados e apresentados, e assim sucessivamente. Quando todas as questões da sessão forem respondidas, é apresentada a página final com os resultados obtidos.

Ao utilizador é apresentada a interface de questionário, como demonstra a Figura 4.16. Com dois contentores principais, o contentor com a questão e a imagem associada, e o contentor das respostas. É possível ver também uma barra no topo da página, esta mostra o progresso da sessão de questionário, um indicativo do número de questões presentes na sessão.

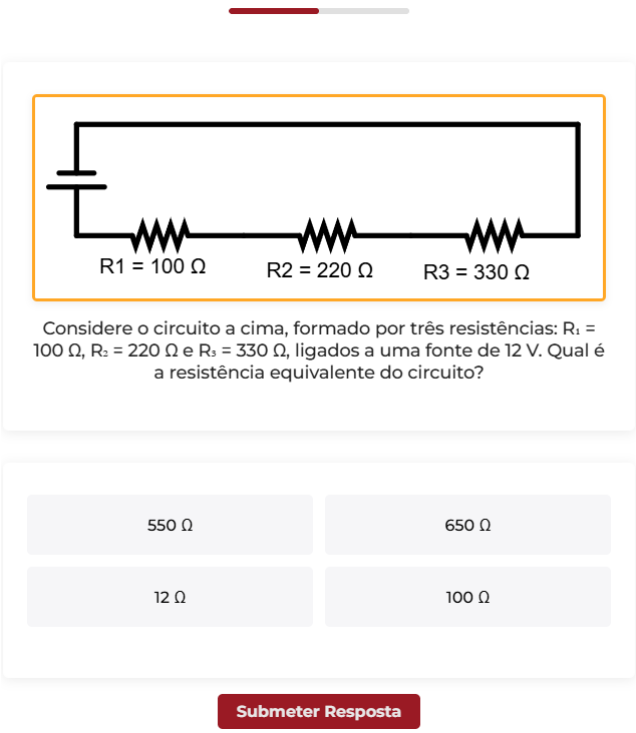

Figura 4.16: Interface de Questionário.

É permitida a seleção da resposta antes da submissão, como representado na Figura 4.17. Assim que o utilizador esteja confortável com a sua resposta, pode fazê-lo carregando no botão “Submeter Resposta”.

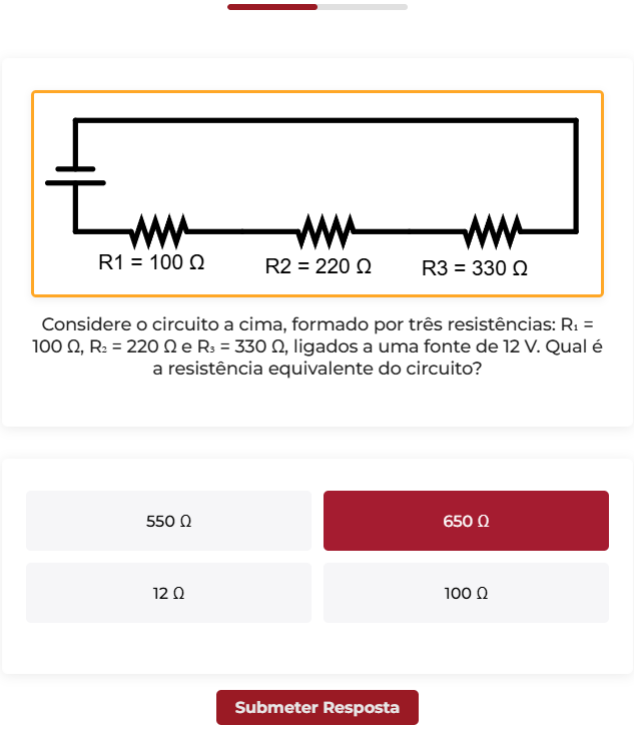

Figura 4.17: Indicação de resposta selecionada.

Após submeter a resposta, a escolha é enviada para o *back end*, onde é comparada com a resposta correta e responde ao *front end* com o resultado, demonstrado na Figura 4.18.

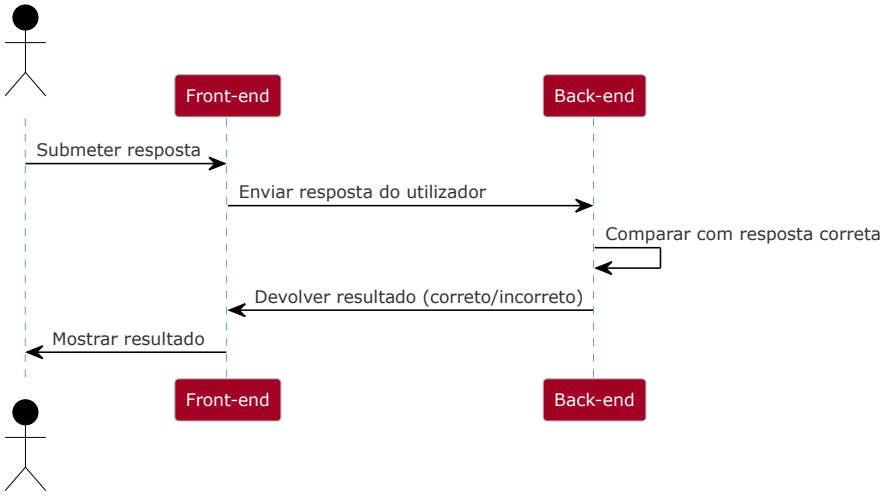

Figura 4.18: Diagrama de submissão de resposta.

Se a resposta foi a correta, a resposta selecionada muda de estilo e para a cor

verde, como a Figura 4.19 demonstra, e é tocado o som de resposta correta e, após alguns segundos, a sessão avança para a próxima questão ou redireciona para a página final da sessão, caso tenha sido a última questão.

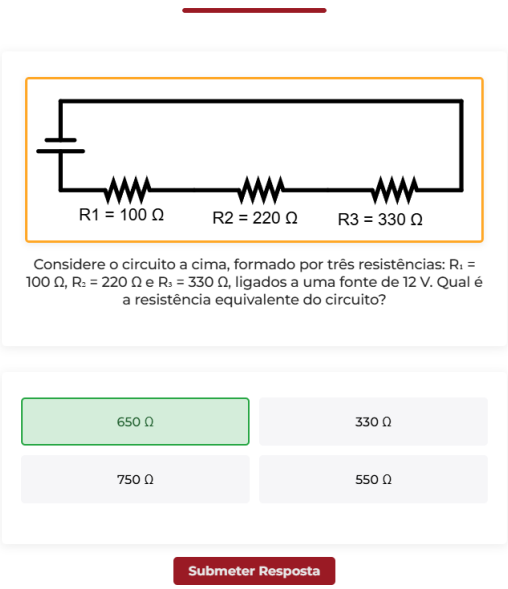

Figura 4.19: Indicação de resposta correta.

Por outro lado, se a resposta for incorreta, a seleção muda de estilo e para a cor vermelho vivo durante alguns segundos, como demonstrado na Figura 4.20, e o ícone "i" aparece no canto inferior direito do contentor da questão.

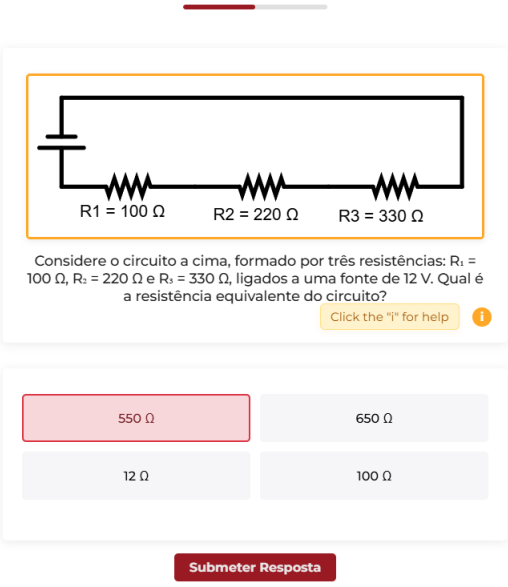

Figura 4.20: Indicação de resposta errada.

Ao carregar no "i", é possível ver uma mensagem de *feedback*,como apresentado na Figura 4.21, esta mensagem tem o intuito de ajudar o utilizador, caso pretenda, com informação sobre a temática da questão.

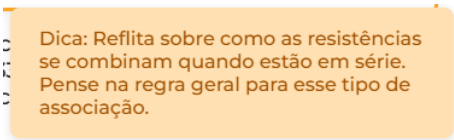

Figura 4.21: Apresentação da mensagem de *feedback*.

A Figura 4.22 mostra a página apresentada após concluir todas as questões da sessão. A página apresenta a mensagem de parabenização por completar a sessão, o gráfico de performance representado por relâmpagos (0-5) e a mensagem que varia conforme a performance obtida. Também mostra a pontuação semanal, o valor da pontuação obtida na sessão a verde e a barra de experiência com o novo valor realçado a verde.

Figura 4.22: Página de Sessão de Questionário Completo.

4.3.3 Elementos de Gamificação

Nível de Utilizador

O nível de utilizador é usado para quantificar a atividade no modo de treino. Tem por base os pontos conquistados no modo de treino. É um sistema linear que requer 100 pontos para subir de nível. A Figura 4.23 mostra como o nível é representado na aplicação, bem como a barra de experiência que indica o progresso do utilizador no nível atual, no topo direito da página.

Figura 4.23: Nível e Barra de Experiência do utilizador.

Modo de Treino

Na página do modo de treino, cada tópico e subtópico têm associado uma barra de progresso, como se pode verificar na Figura 4.13. Cada subtópico tem um número de questões respondidas pelo utilizador necessárias para completar o mesmo. À medida que o utilizador completa cada subtópico, a barra de progresso do tópico é atualizada para refletir esse avanço.

O utilizador conquista pontos ao responder a questões na sessão de questionário. No final, na página de sessão completa, é apresentado o número de pontos conquistados na sessão, bem como a performance do mesmo representada por relâmpagos (0-5). A pontuação a cada questão depende do valor de dificuldade da mesma, do peso do subtópico e do número de tentativas, segundo as equações $Pontos = PontosBase \times Peso \times Penalidade$ e $Penalidade = 1 - (Tentativas - 1) \times 0,25$, assumindo sempre que cada questão tem quatro opções e a penalidade nunca é negativa. Desta forma, o utilizador é penalizado por cada resposta errada, sem desmotivar novas tentativas. A performance é calculada tendo em conta o número de tentativas necessárias para o utilizador responder corretamente às questões, usando a equação $Performance = 5 \times (1 - ((Tentativas - Corretas)/Corretas) \times 0,25)$,no caso de *Corretas* ser nulo, o valor de *Performance* é também nulo. Desse modo, o cálculo premeia a eficiência do utilizador nas suas tentativas, sem penalizar em demasia as tentativas erradas.

Tabela classificativa

A tabela classificativa é um elemento motivador capaz de estimular os utilizadores mais competitivos. Na Figura 4.24 é possível ver que a tabela é composta por 4 colunas, posição classificativa, avatar e nome de utilizador, nível e pontuação semanal. O utilizador pode escolher entre nível e pontuação semanal para ordenar a tabela, mostrando apenas os 10 melhores utilizadores. A posição do utilizador é indicada no topo da página e, no caso de o utilizador estar na tabela, a linha correspondente é realçada.

Classificação

A sua posição: 3

Posição	Utilizador	Nível	Pontos Semanais
1	Luís	4	104
2	Rita	2	93
3	Pedro	1	25
4	João	1	0

Figura 4.24: Tabela Classificativa.

4.4 Metadados

Os metadados desempenham um papel fundamental na plataforma *U=RIsolve Academy*, permitindo o registo e, por consequência, o acompanhamento do progresso dos utilizadores. Neste projeto, os metadados referem-se aos dados associados a cada utilizador, tópico, subtópico ou questão, tais como o nome de utilizador, descrições do tópico, peso do subtópico, o número de tentativas de resposta e conquistas.

Organização na Base de Dados

Os metadados da plataforma estão organizados em várias tabelas da base de dados, permitindo o registo do perfil, do progresso, das conquistas e desempenho do utilizador.

As principais tabelas associadas são:

1. *users*;
2. *user_badges*;
3. *user_subtopic_progress*;
4. *weekly_training*.

Principais colunas de cada tabela:

- ***users*** - Armazena dados do perfil de cada utilizador.
  - id - identificador do registo de utilizador.
  - *username* - nome de utilizador.
  - e-mail - endereço de correio electrónico do utilizador.

  - *exp* - valor total ganho pelo utilizador no modo de treino.
  - *date_joined* - data do registo da conta.

- ***user_badges*** - Armazena as conquistas atribuídas ao utilizador.
  - id - identificador do registo da conquista.
  - *user_id* - identificador do utilizador ao qual o registo pertence.
  - *name* - nome da conquista.
  - svg - ícone da conquista.
  - *date_awarded* - data de atribuição da conquista.

- ***user_subtopic_progress*** - Armazena o progresso do utilizador em cada subtópico.
  - id - identificador do registo de progresso no subtópico.
  - *user_id* - identificador do utilizador ao qual o registo pertence.
  - *subtopic_id* - identificador do subtópico no qual existe progresso.
  - *progress* - valor do progresso do subtópico registado.

- ***weekly_training*** - Armazena o registo do treino semanal do utilizador.
  - *user_id* - identificador do utilizador ao qual o registo pertence.
  - *question_id* - identificador da questão respondida.
  - *week_start* - registo da data da segunda feira da semana em que o registo foi efectuado.
  - *tries* - registo do número de tentativas de resposta.
  - *completed* - registo do número de tentativas corretas.
  - *score* - registo da pontuação total obtida.

Cada tabela armazena diferentes tipos de metadados, que se relacionam entre si através de *Foreing Keys*, garantindo a consistência dos dados.

As relações entre as tabelas são estabelecidas principalmente através do *user_id*, que funciona como *Foreing Keys*, ligando as conquistas, o progresso por subtópico e o registo do treino semanal ao respetivo utilizador. Por exemplo, a tabela *user_subtopic_progress* está associada à tabela *users* através do *user_id* e com a tabela *subtopics* pelo *subtopic_id*.

A Figura 4.25 apresenta um diagrama simplificado das relações entre as principais tabelas.

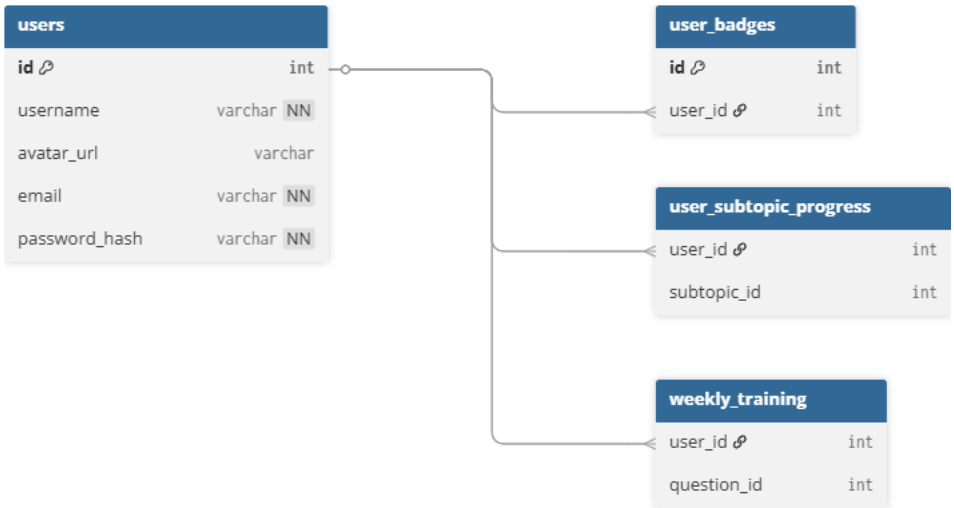

Figura 4.25: Diagrama relacional.

A relação *one-to-many* entre a tabela *users* com as restantes tabelas permite garantir a integridade dos registos, não sendo possível a presença de registos relativos utilizadores inexistentes. Esta relação possibilita maior eficiência na consulta seletiva dos dados através do comando *JOIN* do SQL.

Além das tabelas diretamente relacionadas com o utilizador, a base de dados da aplicação inclui as tabelas *topics*, *subtopics* e *questions* associadas ao modo de treino da plataforma. A tabela *subtopics* está relacionada com a tabela *topics* através do *topic_id*, permitindo organizar os subtópicos dentro de cada tópico. Por sua vez, a tabela *questions* está ligada à tabela *subtopics* pelo *subtopic_id*, possibilitando a associação de cada questão ao respetivo subtópico. A Tabela 4.1 apresenta a síntese das colunas principais de cada tabela.

Tabela 4.1: Resumo de metadados das tabelas relacionadas com modo de treino.

Tabela	**Metadados**
topics	id, name, description
subtopics	id, topic_id, title, weight
questions	id, subtopic_id, question_text, difficulty, correct_answer

Estas relações são fundamentais para contextualizar o progresso do utilizador em função dos conteúdos praticados. Por exemplo, quando o utilizador responde a uma questão, a aplicação regista os metadados do progresso e desempenho, mas também associa esta informação ao tópico e subtópico através das ligações entre as tabelas. Assim, é possível acompanhar o desempenho por utilizador em cada tópico, subtópico e questão da plataforma. A Figura 4.26 está representado o diagrama relacional destas tabelas.

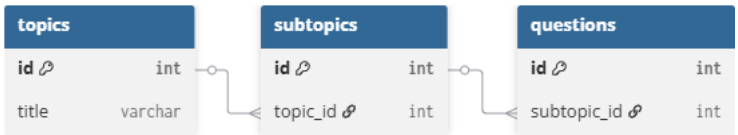

Figura 4.26: Diagrama relacional do Modo de Treino.

Exemplo Prático de Utilização dos Metadados

Considere-se o seguinte cenário: o utilizador realiza uma sessão de questionário do modo de treino. Ao responder a uma questão do subtópico *Método da resistência equivalente*, pertencente ao tópico *Métodos de Análise de Circuitos Elétricos*, a aplicação regista automaticamente vários metadados na base de dados. Por exemplo, na tabela *user_subtopic_progress* é atualizado automaticamente, na coluna *progress*, se o utilizador completou o subtópico, na tabela *weekly_training* é registado as tentativas, na coluna *tries*, e os pontos, na coluna *score*, conquistados para a questão respondida na semana, na tabela *users* é adicionado à experiência os pontos conquistados, e, por fim, caso o utilizador tenha atingido uma meta ao responder à questão, é adicionado o *badge* da conquista na tabela *user_badges*.

Capítulo 5

Conclusões

5.1 Reflexão Final

Ao longo do desenvolvimento deste projeto, todos os objetivos definidos para o projeto foram cumpridos. Foram implementadas com sucesso as novas funcionalidades, nomeadamente, o perfil de utilizador, modo de treino melhorado e a incorporação de elementos de gamificação.

A aplicação permite o registo de utilizadores e possibilita o rastreio individual de atividade do modo de treino, garantindo assim que cada utilizador tem o conhecimento do seu progresso.

O modo de treino proporciona uma seleção mais abrangente e específica de temáticas. Também assegura sessões de questionário com características diferentes entre subtópicos.

A integração dos seguintes elementos contribui para uma experiência mais envolvente e motivadora para os utilizadores:

- Tabela classificativa — permite aos utilizadores comparar o seu desempenho com o dos restantes.
- Nível e experiência — proporcionam um sentido de progressão individual.
- Barras de progresso dos tópicos e subtópicos — facilitam o acompanhamento do progresso e incentivam a conclusão de objetivos.

- Efeitos sonoros na sessão de questionário — aumentam a imersão e tornam a interação mais dinâmica.

5.2 Competências Adquiridas

Este foi o projeto mais complexo em que participei e, durante o desenvolvimento, foram vários os desafios que encontrei. Os principais obstáculos foram erros inesperados nas alterações do código em outras partes da aplicação.

Também foi desafiador adaptar-me ao modelo organizacional da aplicação. No entanto, à medida que o trabalho foi avançando, senti-me cada vez mais confortável com o modelo.

Sinto que este projeto ajudou-me a evoluir as minhas competências técnicas e organizacionais de código e de gestão do projeto. Além disso, obrigou-me a sair da zona de conforto e a aprender novas tecnologias, tais como sistemas de gestão de bases de dados e o modelo padrão *Model-View-Controller* (MVC), essenciais no desenvolvimento de aplicações web.

Ao longo deste projeto, senti que evoluí bastante a nível técnico e organizacional. Trabalhei bastante com *JavaScript* (JS), *HyperText Markup Language* (HTML), *Cascading Style Sheets* (CSS), *Structured Query Language* (SQL) e com *Handlebars* para a renderização das páginas web no *front end*, e consegui ganhar experiência prática com Node.js e Express no desenvolvimento do *back end*. Tive a oportunidade de aprender a gerir bases de dados com *MySQL* e implementar o padrão MVC.

Para além das competências técnicas, o projeto contribuiu para o meu desenvolvimento organizacional, obrigando-me a gerir o meu tempo eficazmente para cumprir com os prazos estabelecidos. As ferramentas *GitHub* e *Trello* foram fundamentais neste processo, facilitaram a organização de tarefas, no acompanhamento da progressão do projeto e na gestão do código.

O *Github Copilot* revelou-se especialmente útil para esclarecer dúvidas relativas ao código e às tecnologias que não domino completamente, bem como para auxiliar na geração de dados de teste associados a operações de escrita e leitura na base de dados.

Foi uma ferramenta essencial que tornou o processo de desenvolvimento mais rápido, com tudo, é necessário sentido crítico na utilização, pois nem todas as sugestões eram as mais adequadas e, por vezes, erradas. Assim, o acompanhamento atento das recomendações do *Copilot* é indispensável para garantir a fiabilidade do resultado final.

5.3 Trabalho Futuro

No seguimento deste projeto, sobressaem as seguintes propostas de melhorias e futuros projetos:

- Melhor rastreio da atividade e monitorização em tempo real da aplicação que permita a deteção de erros e acompanhar o funcionamento.
- Adicionar ferramentas de administração onde é possível adicionar e remover conteúdos e moderação geral da aplicação.
- Incluir um "fórum"que permita interação entre utilizadores positiva, como colocar dúvidas sobre questões do modo de treino ou sobre as temáticas da plataforma.
- Adicionar mais elementos de gamificação, como prémios e desafios diários/semanais.
- Adicionar funcionalidades de validação de conta através da verificação de e-mail.
- Adicionar o gráfico radial à pagina de estatísticas.
- Adicionar elementos customizáveis, como avatares.

Acredito que a plataforma *U=Risolve Academy +* tem potencial para se tornar uma aplicação relevante e interessante no universo académico e didático.

Referências

[1] J. P. C. Ferreira, "Halogen: on the foundations for nlp interactions in electrical engineering education," Master's thesis, Faculdade de Engenharia da Universidade do Porto, Porto, 2025. [Citado na página 1]

[2] A. M. Araújo, A. A. Santos, A. P. Noronha, C. Zanon, J. A. Ferreira, J. R. Casanova, and L. S. Almeida, "Dificuldades antecipadas de adaptação ao ensino superior: Um estudo com alunos do primeiro ano," *Revista de Estudios e Investigación en Psicología y Educación*, vol. 3, no. 2, pp. 102–111, 2016. [Citado na página 1]

[3] B. Gavaia, J. Martins, P. Rodrigues, A. Barbot, and J. B. Lopes, "Gamificação e microcredenciais no ensino superior: estratégias para potenciar a motivação em contextos de baixo interesse," *PRATICA - Revista Multimédia de Investigação em Inovação Pedagógica e Práticas de e-Learning*, vol. 8, no. 3, p. 40–54, 2025. [Citado na página 1]

[4] T. A. M. Pedroso, "Gamificação para a melhoria da experiência de aprendizagem dos alunos: caso estudo learning scorecard," dissertação de mestrado, ISCTE – Instituto Universitário de Lisboa, 2018. [Citado na página 5]

[5] "Khan acadamy." Acedido no dia 10 de agosto de 2025. [Citado na página 5]

[6] "Duolingo." Acedido no dia 10 de agosto de 2025. [Citado na página 6]

[7] "Mimo." Acedido no dia 10 de agosto de 2025. [Citado na página 6]

[8] M. Kanazawa, "Which gamification elements to use for your future gamification project?." Acedido no dia 10 de agosto de 2025. [Citado na página 8]

[9] M. Fowler, *Patterns of Enterprise Application Architecture*. Boston: Addison-Wesley, 2002. [Citado na página 8]

[10] M. Learn, "Asp.net mvc overview," 2025. [Citado nas páginas vii e 8]

[11] "Trello," 2025. Acedido no dia 15 de agosto de 2025. [Citado na página 10]

[12] "Github," 2025. Acedido no dia 15 de agosto de 2025. [Citado na página 10]