

Licenciatura em Engenharia Informática

UC: Programação Web

**Relatório do Trabalho**

Elaborado por :   
  
 Tiago Pires, 45573

Rúben Lopes, 51179

# Índice

1. Introdução

2. Enquadramento e Objetivos

3. Tecnologias e Ferramentas

4. Arquitetura do Sistema

5. Funcionalidades

6. Gestão de Utilizadores e Perfis

7. Segurança e Boas Práticas

8. Importação de Utilizadores

9. Conclusão e Futuras Expansões.

1. Introdução

Este relatório documenta o desenvolvimento de um sistema completo de gestão de energia renovável, projetado com base em tecnologias web modernas. A aplicação visa permitir a monitorização e gestão de painéis solares, simulação de produção energética, controlo de créditos e autenticação segura de utilizadores.

# 2. Enquadramento e Objetivos

O objetivo principal do projeto foi criar um sistema web funcional que reflita as necessidades reais de gestão de recursos energéticos. Foram definidos os seguintes objetivos específicos:

• Desenvolver funcionalidades distintas para diferentes papéis de utilizador (admin, técnico, gestor, cliente).

• Fornecer dashboards personalizados com estatísticas e histórico energético.

• Implementar sistema de registo e certificação de painéis solares.

• Otimizar o armazenamento e tratamento de dados com MongoDB.

# 3. Tecnologias e Ferramentas

A stack tecnológica escolhida segue as boas práticas da engenharia moderna de software. Foram utilizadas:

- Node.js e Express para backend, com organização modular por rotas, controladores e modelos.

- MongoDB como base de dados NoSQL, com manipulação via Mongoose.

- HTML5, CSS3 e JavaScript puro no frontend.

- Autenticação segura com JWT.

- Upload de ficheiros implementado com Multer.

- Ambiente de desenvolvimento com dotenv, nodemon.

# 4. Arquitetura do Sistema

A aplicação foi estruturada segundo o padrão MVC (Model-View-Controller), com separação clara de responsabilidades. Cada componente do sistema possui funcionalidades bem definidas. A estrutura inclui:

• `models/`: Representam os dados da aplicação (painéis, utilizadores, energia).

• `controllers/`: Contêm a lógica de negócio para cada endpoint.

• `routes/`: Responsáveis pelo mapeamento das URLs.

• `middleware/`: Verificações de autenticação, MFA e permissões de acesso.

• `public/`: Interface gráfica separada por perfis de utilizador.

• `services/`: Contêm utilitários externos como MFA e validadores.

# 5. Funcionalidades

A aplicação disponibiliza funcionalidades específicas consoante o tipo de utilizador autenticado:

• Registo/Login seguro com JWT.

• Configuração de autenticação multifator (2FA) via Google Authenticator.

• Dashboard informativo com indicadores de produção de energia.

• Registo e gestão de painéis solares por parte dos clientes.

• Validação e certificação de instalações por técnicos especializados.

• Consulta de créditos energéticos acumulados e histórico de produção.

• Criação, edição e remoção de contas pelo administrador.

• Upload de certificados com validação de formato e destino seguro.

# 6. Gestão de Utilizadores e Perfis

Foram implementadas funcionalidades específicas para cada tipo de utilizador:

• Admin: Acesso total ao sistema, gestão de contas e permissões.

• Técnico: Acesso ao dashboard técnico com validação de instalações.

• Gestor: Acesso a relatórios e análises de produção e consumo.

• Cliente: Registo de painéis e acompanhamento dos próprios dados energéticos.

# 7. Segurança e Boas Práticas

A segurança foi uma das prioridades do projeto, garantindo proteção contra acessos não autorizados. As principais medidas incluem:

• Hashing de passwords com bcryptjs.

• Autenticação baseada em tokens JWT.

• Middleware para validação de permissões por role.

• Armazenamento de segredos em `.env`.

• Upload seguro com validação de extensão e tipo de ficheiro.

# 8. Importação de Utilizadores

Foi criado um ficheiro `users.json` com utilizadores predefinidos para testes. Os dados incluem emails, funções e palavras-passe encriptadas.

O comando para importação é:

**mongoimport --uri="mongodb://localhost:27017/renewable\_energy" --collection=users --file=users.json --jsonArray**

# 9. Conclusão e Futuras Expansões

A aplicação demonstra um sistema realista e funcional de gestão de energia, com componentes robustos e escaláveis. A implementação modular permite futuras melhorias, como:

• Interface em React para maior interatividade.

• Integração com APIs externas de energia (IoT).

• Sistema de notificações automáticas por email.

• Mecanismos de partilha e comércio de créditos energéticos.

• Deploy em servidores cloud (AWS, Heroku, etc.).

O projeto serviu como uma excelente aplicação prática dos conhecimentos adquiridos em Programação Web.