



UNIVERSIDADE DA BEIRA INTERIOR

PROGRAMAÇÃO NO SERVIDOR

Projeto de Gestão de Montagem de Drones

Fábio Maio 48286
Rafael Lourenço 48115
Simão Fraga 47831

Professores:
Carlos Cunha
Nuno Carapito

Junho 2024

Contents

1	Introdução	2
2	Sprint 1: Criação da documentação da API em OpenAPI	2
2.1	Objetivo	2
2.2	Passos	2
3	Sprint 2: Implementação da API para Autenticação e Registo de Novos Utilizadores	3
3.1	Objetivo	3
3.2	Passos	3
4	Sprint 3: Implementação da API para Registo de Peças	3
4.1	Objetivo	3
4.2	Passos	3
5	Sprint 4: Implementação da API para Registo de Montagem de Drones	4
5.1	Objetivo	4
5.2	Passos	4
6	Detalhes Técnicos	4
6.1	Tecnologias Utilizadas	4
6.2	Estrutura do Projeto	5
7	Conclusão	6

1 Introdução

O objetivo deste projeto foi desenvolver uma aplicação web para gerir o processo de montagem de drones. A aplicação permite aos utilizadores registarem-se, autenticarem-se, e gerirem informações relacionadas com a montagem de drones, incluindo peças necessárias e detalhes de montagens. A aplicação foi desenvolvida utilizando Node.js, Express.js, Mongoose, MongoDB, e JWT para autenticação.

2 Sprint 1: Criação da documentação da API em OpenAPI

2.1 Objetivo

Criar um ficheiro de documentação para estruturar todos os endpoints que serão criados futuramente.

2.2 Passos

1. Escolher um estruturador para criar os endpoints, no nosso caso foi o Swagger.
2. Criar os endpoints para a futura API.

3 Sprint 2: Implementação da API para Autenticação e Registo de Novos Utilizadores

3.1 Objetivo

Implementar a API para autenticação e registo de novos utilizadores, permitindo que os utilizadores possam registar-se e fazer login na aplicação.

3.2 Passos

1. Configuração inicial do projeto com Node.js e Express.js.
2. Configuração do MongoDB e Mongoose para o armazenamento de dados.
3. Implementação do esquema de utilizador (`user.js`) incluindo validações e encriptação de senhas.
4. Implementação dos endpoints para registo e login de utilizadores (`userRoutes.js`).
5. Testes dos endpoints utilizando Insomnia.

4 Sprint 3: Implementação da API para Registo de Peças

4.1 Objetivo

Implementar a API para gestão de peças necessárias para a montagem de drones, permitindo adicionar, listar, atualizar e apagar peças.

4.2 Passos

1. Criação do esquema de peças (`part.js`) incluindo validações.
2. Implementação dos endpoints para adicionar, listar, atualizar e apagar peças (`partRoutes.js`).
3. Testes dos endpoints utilizando Insomnia.

5 Sprint 4: Implementação da API para Registro de Montagem de Drones

5.1 Objetivo

Implementar a API para gestão de montagens de drones, incluindo a associação de peças e a data de montagem, e a consulta de montagens por utilizador.

5.2 Passos

1. Criação do esquema de montagens de drones (`droneAssembly.js`) incluindo a referência a peças e utilizadores.
2. Implementação dos endpoints para adicionar, listar, atualizar e apagar montagens de drones (`droneAssemblyRoutes.js`).
3. Testes dos endpoints utilizando Insomnia.

6 Detalhes Técnicos

6.1 Tecnologias Utilizadas

- **Swagger:** Conjunto de ferramentas para desenvolvedores de API.
- **Node.js:** Ambiente de execução para o JavaScript no lado do servidor.
- **Express.js:** Framework para o desenvolvimento de aplicações web.
- **Mongoose:** Biblioteca para modelagem de dados no MongoDB.
- **MongoDB:** Banco de dados NoSQL utilizado para armazenamento de dados.
- **JWT (JSON Web Tokens):** Utilizado para autenticação segura.

6.2 Estrutura do Projeto

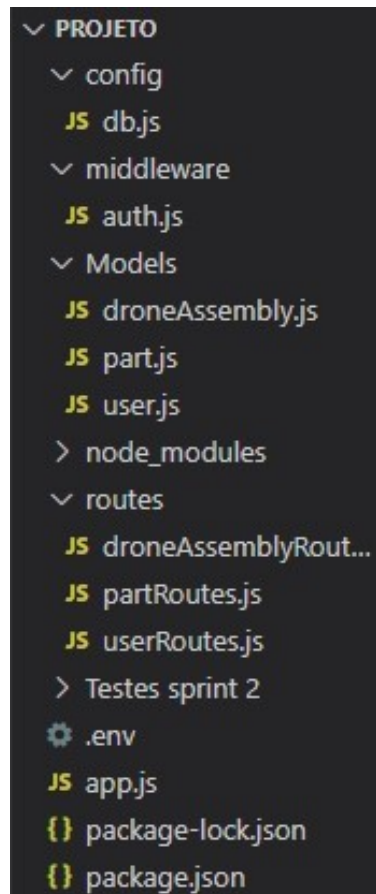


Figure 1: Organização das pastas do nosso projeto

7 Conclusão

Durante o desenvolvimento deste projeto, tivemos a oportunidade de aprender e aplicar diversos conceitos fundamentais no desenvolvimento de aplicações web utilizando Node.js, Express.js, MongoDB e Mongoose. A criação de uma aplicação para a gestão do processo de montagem de drones envolveu a implementação de funcionalidades críticas, como autenticação de utilizadores, gestão de peças e montagens de drones, e a documentação dos endpoints da API.

Trabalhar com sprints permitiu-nos focar em funcionalidades específicas e garantir que cada parte do sistema fosse completamente desenvolvida e testada antes de avançar para a próxima etapa. Esta abordagem incremental e iterativa ajudou a identificar e corrigir problemas mais cedo no processo de desenvolvimento, melhorando a qualidade geral do sistema.

A utilização de ferramentas como Insomnia para testar os endpoints da API foi crucial para garantir que cada funcionalidade estivesse a funcionar corretamente. A escrita de esquemas detalhados em Mongoose ajudou-nos a garantir a integridade dos dados e a aplicar validações importantes.

Em resumo, o desenvolvimento deste projeto não só resultou numa aplicação funcional para a gestão de montagem de drones, mas também nos proporcionou uma valiosa experiência prática com APIs, esquemas de base de dados e desenvolvimento ágil com sprints. Este conhecimento será fundamental para futuros projetos e para o desenvolvimento contínuo de competências em desenvolvimento de software.