

# Projetos 01

Tópicos de Informática para Automação

Mário Antunes

27 de Outubro de 2025

## Projetos

Formem grupos de dois ou três alunos (excepcionalmente, os projetos podem ser feitos individualmente) e selecionem um dos seguintes projetos. Todos os projetos serão alojados no GitHub, usando o [GitHub Classroom](#).

O repositório deve conter todos os scripts e ficheiros de configuração relevantes, bem como um README.md com instruções sobre como implementar o projeto. Além disso, o repositório deve ainda conter um PDF com o relatório do projecto. Este projeto tem a duração de três semanas. Têm até ao final desta semana para notificar o vosso professor sobre os membros do grupo e o tópico escolhido.

Para terem acesso ao GitHub Classroom, precisam de uma conta GitHub e de fazer parte da organização detiuaveiro. Não se esqueçam de contactar o vosso professor com qualquer questão. Mais instruções serão adicionadas.

### 1. Site Estático de Alto Desempenho com Cache

- **Descrição:** Implemente um serviço web de alto desempenho usando Docker Compose. Esta configuração deve incluir dois serviços: um servidor web (como **Caddy** ou **Apache httpd**) e uma cache de proxy reverso (como **Squid**). O conteúdo do site estático (uma página complexa com vários estilos e imagens) deve ser servido a partir de um **volume** montado no contentor do servidor web. A cache deve ser configurada para se posicionar à frente do servidor web, e apenas a porta da cache deve ser exposta.
- **Tópicos Principais:** Docker Compose (multi-serviço), Caddy/httpd, Squid, volumes, redes de contentores.

### 2. O Solver “Na Minha Máquina Funciona”: Um Dev Container

- **Descrição:** Crie um Dockerfile para uma linguagem de programação específica (p. ex., Python, C++, ou Node.js). Este Dockerfile deve instalar o compilador/interpretador e todas as bibliotecas necessárias. O projeto usará Docker Compose e um **volume** para montar uma pasta de código local, permitindo-lhe compilar/executar o seu código *de dentro* do contentor, garantindo um ambiente de compilação (build) reproduzível.
- **Tópicos Principais:** Dockerfile, volumes, Docker Compose, gestão de pacotes (apt).

### 3. Backup Automatizado para o Nextcloud

- **Descrição:** Escreva um **script Bash** que cria um backup comprimido .tar.gz de um diretório especificado. O script deve então mover este arquivo para uma pasta local que está a ser monitorizada pelo **Nextcloud Desktop Client**. O objetivo é criar um sistema de backup totalmente automatizado onde os ficheiros locais são arquivados e depois sincronizados automaticamente para um servidor Nextcloud remoto.
- **Tópicos Principais:** Scripting Bash (tar, date), cron, cliente Nextcloud.

## 4. Site de Anúncios da Turma com WordPress

- **Descrição:** Implemente uma instalação completa do WordPress usando Docker Compose. Isto requer a orquestração dos contentores `wordpress` e `mysql` (ou MariaDB). Deve usar **volumes** para persistência. O objetivo é configurar o site como um simples feed de anúncios para esta turma, criando pelo menos dois posts e personalizando o tema.
- **Tópicos Principais:** Docker Compose (multi-serviço), WordPress, redes de contentores, volumes, variáveis de ambiente.

## 5. Confronto de Desempenho: VM vs. Contentor

- **Descrição:** Implemente um servidor web NGINX simples de duas formas: 1) dentro de uma **Debian VM** completa (usando VirtualBox/QEMU) e 2) dentro de um **contentor Docker**. Irá então escrever um relatório a comparar o tempo de arranque, o uso de RAM em inatividade e o espaço em disco ocupado por ambos os métodos.
- **Tópicos Principais:** Virtualização (configuração de VM), Contentores (Docker), ferramentas de monitorização de sistema (`top`, `df`, `time`).

## 6. Implementação da Wiki da Turma

- **Descrição:** Use o Docker Compose para implementar uma wiki totalmente funcional (p. ex., `dokuwiki/dokuwiki` ou `linuxserver/bookstack`) para servir como base de conhecimento para esta turma. O foco está em ler corretamente a documentação da imagem, gerir dados persistentes com **volumes**, e configurar o serviço usando variáveis de ambiente. Deve preencher a wiki com pelo menos cinco páginas de conteúdo dos materiais da aula.
- **Tópicos Principais:** Docker Compose, volumes, gestão de imagens de terceiros, variáveis de ambiente.