



ESCOLA  
SUPERIOR  
DE TECNOLOGIA  
E GESTÃO

# Projeto de Laboratório de Programação

Licenciatura em Engenharia Informática

Licenciatura em Segurança Informática em Redes de  
Computadores

2020/2021

**Grupo 244**

8200302, Hélder Branco

## 1. Introdução

Mannager 0.5

Este projeto/programa serve para facilitar o calculo salarial de uma dada empresa, organizar os dados de todos os trabalhadores e guardar os mesmos, facilitando vários processos essenciais a qualquer empresa.

Este projeto divide-se em 3 grandes partes, tratamento de dados de cada colaborador, informação sobre os dias de trabalho(jornadas) dos mesmos e a realização automática de relatórios importantes para a empresa e que auxiliem o calculo salarial, suporte do programa de avaliação continua.

## 2. Funcionalidades requeridas

As funcionalidades requeridas no enunciado deste projeto são:

- Tratamento de dados dos colaboradores

Criei uma estrutura genérica para o funcionário, onde defino o código como um número inteiro, o nome como um array de caracteres que pode ter no máximo 51 caracteres, a quantidade de dias trabalhados como um inteiro, o estado como inteiro e os projetos em que está envolvido determinado funcionário, no máximo 5 projetos.

- o Criar/Adicionar;

Para criar/Adicionar um colaborador criei uma função que faz com que seja possível inserir um funcionário e todos os seus dados. Começo por pedir o nome do mesmo e com a função procurarFunc, que posteriormente explicarei, procuro se existe algum funcionário com o mesmo nome se não de seguida peço o código, marco o seu estado como 1-ativo. De retorno incremento um ao contador para poder controlar o tamanho e número de funcionários. Se este nome já existir digo ao utilizador que já existe esse funcionário.

- o Editar a informação

Tal como no inserir começo por procurar o funcionário, só que agora espero que ele já exista e se isto for verdade edito o nome, o código do mesmo e os projetos onde trabalha. Se o funcionário não existir digo ao utilizador que este não existe.

- o Remover (apenas deve marcar como removido e não apagar a informação referente ao funcionário). Para remover o funcionário apenas mudo o seu estado de 1-ativo para 0-removido.

### ➤ Jornadas/Registo de trabalho

Existem 3 tipos de jornada, falta, meia jornada e jornada completa. Dentro das jornadas, cada dia deverá estar associado a um projeto no qual o funcionário esteja a trabalhar/associado.

Para registar esta informação começo por procurar o funcionário e se este existir pergunto o dia, mês e ano. De seguida pergunto qual o projeto e o tipo de jornada, 0-falta, 1-meia jornada e 2-jornada completa.

Ainda somo o total de cada tipo de jornada, mas estes ficam-me com valores sem sentido no contexto do objetivo do projeto.

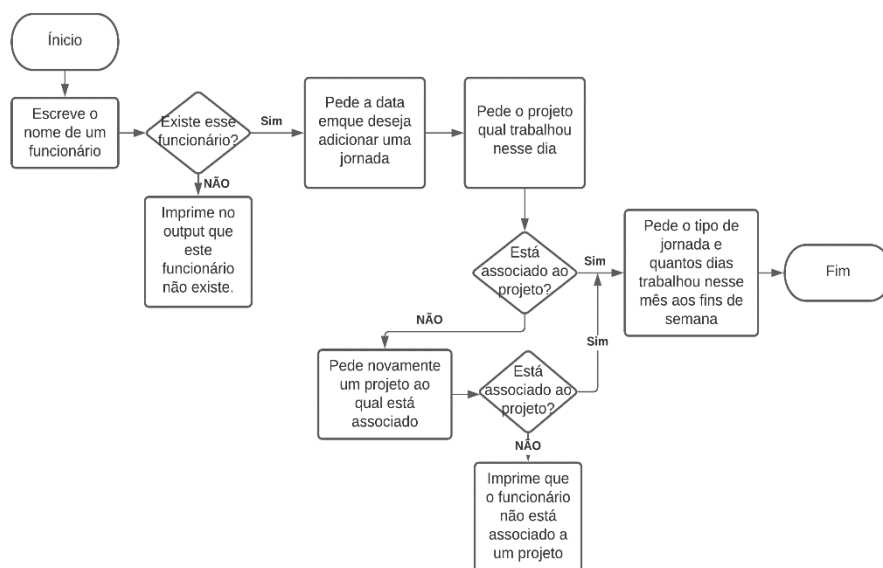


Figura 1- Fluxograma do registo de jornadas

Durante o desenvolvimento do projeto não pus os fins de semana a funcionar corretamente. Cada vez que peço uma nova jornada mesmo que esta seja de um mês pedido anteriormente já e do mesmo funcionário, volto a pedir, assim o registo pode mudar.

#### ➤ Processamento de relatórios/Exportação

Deve ser possível com toda a informação dos funcionários e também com informações das jornadas criar relatórios que sejam pertinentes para a empresa e ajudem esta a melhorar e otimizar os seus gastos, recursos humanos e logística da mesma.

Toda a informação anteriormente referida deve ser guardada em ficheiros de texto, tanto os relatórios como os dados de cada funcionário e das suas jornadas do mês escolhido pelo utilizador.

No ficheiro .txt escrevo o código e o nome de cada funcionário de seguida as diversas jornadas que ele teve em determinado mês e estão a associadas a um projeto.

#### ➤ Persistência de dados

Através dos ficheiros criados anteriormente o programa deve guardar a informação neles e quando encerrar o programa estes devem persistir nos ficheiros e quando abrimos de novo o projeto os dados devem ser carregados para o mesmo sem perder qualquer informação.

Sempre que necessário deve ser utilizado memória dinâmica de modo a facilitar o armazenamento da informação.

Os dados dos funcionários são persistentes mas para a informação das jornadas não tive tempo para fazer o mesmo então cada vez que fecho o programa perco toda a informação que inseri sobre as jornadas.

### **3. Funcionalidades propostas**

Uma das funcionalidades que me eu propus a fazer foi uma listagem dos funcionários no output do programa para tornar mais fácil a utilização do programa. Esta é uma função bastante simples, mas que no final dá sempre alguma ajuda ao utilizador.

A funcionalidade apenas passa por imprimir o nome, o código e o estado de todos os funcionários que estejam guardados nos ficheiros.

Outra das funcionalidade foi a função procurarFuncionarios, esta era necessária ao programa para todo ele funcionar, então não sei bem se posso considerar uma funcionalidade extra.

Mas passa por pedir nome e procurar na lista se existe outro igual na lista de funcionários.

#### **4. Estrutura analítica do projeto**

Após a leitura atenta do enunciado do trabalho comecei por fazer toda uma estrutura para o funcionário, a qual contém o seu nome, código, estado, a quantidade de dias trabalhados pelo mesmo e os projetos ao qual está associado. De seguida criei uma estrutura para o geral dos funcionários, na qual tenho um contador para estes e um tamanho que é aumentado quando necessário pelo programa.

Após as estruturas fiz todas as funções necessárias para preencher e modificar os dados dos funcionários, como por "inserirFuncionarios" ou "editarFuncionarios".

Quando esta parte do programa estava toda a funcionar e sem erros passei para a parte de ficheiros, precisava de guardar os dados dos funcionários, então criei funções que permitem guardar os funcionários num "ficheiro.text" e carregar os mesmo de volta para o programa quando encerrar/fechar o mesmo, abrir e os dados ainda persistirem.

Passei para as jornadas e para as jornadas tal como em outras partes do programa criei 2 duas funções uma que permite preencher as jornadas de cada funcionário e uma para imprimir em um ficheiro.txt a informação das jornadas (nome do funcionário, código, data e o tipo de jornada feita nesse dia por esse) e o projeto a qual está associado. Esta foi uma parte complicada e com diversos "bug's" que acabei por resolver, mas que me levaram bastante tempo, algumas funcionalidades das mesmas não foram desenvolvidas, como por exemplo a "contagem" dos fins de semanas. Por vezes não conseguia guardar nenhuma informação no ficheiro outras vezes guardava apenas partes da informação ou guardava informações erradas. Esta parte

deveria também ser persistente, mas devido ao tempo apertado para a entrega não o consegui.

Nos últimos dias antes da entrega do projeto dediquei-me a resolver erros do programa e trabalhar sobre a documentação do projeto e no relatório.

## **5. Funcionalidades implementadas**

As funcionalidades que implementei foi a listagem de todos os funcionários no output do programa e procurar funcionários.

### ➤ Listar

A listagem é simples, mas dá sempre uma ajuda ao utilizador caso este se esqueça de alguma informação de um funcionário. Esta passa por imprimir toda a informação referente aos funcionários no output.

### ➤ Procurar

O procurar é uma função que de certa maneira não era exigida, mas era quase obrigatório ter para que todo o programa funcionasse, sem esta era impossível, a meu ver, desenvolver o programa. Esta procura na lista dos funcionários se determinado funcionário existe se sim retorna um valor, se não diz ao utilizador que não existe esse funcionário.

## **6. Conclusão**

O trabalho foi em grande parte concluído com sucesso, apesar das dificuldades, com exceção de algumas funcionalidades. A principal razão foi o tempo apertado devido às diversas avaliações, o que fez com que não fizesse todas as funcionalidades.

Estou satisfeito com o projeto, sei que com mais tempo poderia fazer algo melhor, mas com todas as ferramentas que tive a mão está bastante bem elaborado.

Com o projeto, sem duvida algum aprendi bastante e aprofundei de uma maneira absurda os meus conhecimentos sobre programação.

