

Relatório do Projeto  
Laboratório de Informática

G36 - Época Especial

Julho de 2025

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>2</b>
1.1	Contextualização . . . . .	2
<b>2</b>	<b>Objetivos do Projeto</b>	<b>3</b>
<b>3</b>	<b>Análise e Estrutura do Sistema</b>	<b>4</b>
3.1	Estruturas de Dados . . . . .	4
3.2	Fluxo de Funcionamento . . . . .	4
<b>4</b>	<b>Descrição das Funcionalidades</b>	<b>5</b>
4.1	Carregamento de Dados . . . . .	5
4.2	Listagem e Consulta . . . . .	5
4.3	Cálculo de Estatísticas . . . . .	5
<b>5</b>	<b>Principais Funções e Algoritmos</b>	<b>6</b>
5.1	Gestão de Listas Ligadas . . . . .	6
5.2	Validação de Dados . . . . .	6
5.3	Exemplo de Utilização . . . . .	6
<b>6</b>	<b>Resultados e Testes</b>	<b>7</b>
6.1	Dificuldades Encontradas . . . . .	7
<b>7</b>	<b>Conclusão</b>	<b>8</b>
7.1	Trabalho Futuro . . . . .	8
<b>8</b>	<b>Anexos</b>	<b>9</b>
8.1	Estruturas de Dados . . . . .	9
8.2	Documentação Automática . . . . .	9

# Capítulo 1

## Introdução

O presente relatório descreve, de forma detalhada, o desenvolvimento do projeto "Sistema de Gestão do Espaço Social" realizado no âmbito do Laboratório de Informática. O objetivo principal deste sistema é gerir de forma eficiente as refeições servidas a utentes, funcionários e gerir toda a informação associada a ementas, escolhas e controlo de custos.

### 1.1 Contextualização

A gestão de refeições em instituições sociais exige rigor, eficiência e flexibilidade. O sistema desenvolvido permite automatizar tarefas como o registo de funcionários, definição de ementas semanais, recolha de escolhas dos utentes e análise estatística do consumo alimentar.

## Capítulo 2

# Objetivos do Projeto

- Permitir o carregamento e gestão de dados de funcionários.
- Gerir ementas semanais, com vários tipos de prato.
- Recolher e analisar as escolhas alimentares dos utentes.
- Listar refeições por dia e por utente.
- Calcular médias de calorias e apresentar resumos semanais.
- Gerar relatórios detalhados e garantir a integridade dos dados.

# Capítulo 3

## Análise e Estrutura do Sistema

O sistema foi desenvolvido em linguagem C, com uma arquitetura modular, separando as funcionalidades em diferentes ficheiros e módulos:

- **main.c**: Implementa o menu principal e a lógica de interação com o utilizador.
- **funcoes.c**: Contém as funções auxiliares e principais para manipulação de dados.
- **funcionarios.h**: Estruturas e protótipos para gestão de funcionários.
- **estruturas.h**: Estruturas de dados globais e protótipos de funções.

### 3.1 Estruturas de Dados

O sistema utiliza listas ligadas para armazenar funcionários, ementas e escolhas, permitindo flexibilidade na gestão dinâmica dos dados. As principais estruturas são:

- **Funcionario**: Número, nome, NIF, telefone.
- **Ementa**: Dia, data, pratos (carne, peixe, dieta, vegetariano) e calorias.
- **Escolha**: Dia, número do funcionário, tipo de prato escolhido.

### 3.2 Fluxo de Funcionamento

O utilizador interage com o sistema através de um menu textual, podendo carregar dados, consultar informações, gerar relatórios e atualizar valores. O ciclo principal do programa garante que todas as operações críticas só são permitidas após o carregamento dos dados essenciais.

# Capítulo 4

## Descrição das Funcionalidades

### 4.1 Carregamento de Dados

- **Funcionários:** Lidos de ficheiro, validados e armazenados em lista ligada.
- **Ementas:** Carregadas semanalmente, permitindo atualização dinâmica.
- **Escolhas:** Importadas de ficheiro, associando cada escolha a um funcionário e a um tipo de prato.

### 4.2 Listagem e Consulta

- Listagem de refeições por dia da semana.
- Consulta de refeições de um utente por intervalo de datas ou semana completa.
- Resumo semanal de consumo por utente.
- Geração de tabelas detalhadas para análise alimentar.

### 4.3 Cálculo de Estatísticas

- Cálculo de médias de calorias consumidas num dado período.
- Controlo do número máximo de refeições diárias.
- Atualização e consulta do valor da refeição.

# Capítulo 5

## Principais Funções e Algoritmos

### 5.1 Gestão de Listas Ligadas

Todas as entidades (funcionários, ementas, escolhas) são geridas por listas ligadas, permitindo inserção, remoção e pesquisa eficiente.

### 5.2 Validação de Dados

São implementadas rotinas para validação de datas, limites diários de refeições e integridade dos dados lidos dos ficheiros.

### 5.3 Exemplo de Utilização

```
=== Sistema de Gestao do Espaco Social ===
```

- a. Carregar dados dos funcionarios
- b. Carregar ementa semanal
- c. Carregar escolhas dos utentes
- 2. Listar refeicoes requeridas por dia
- 3. Resumo semanal de consumo por utente
- 4. Consultar refeicoes de um utente
- 5. Calcular medias de calorias
- 6. Gerar tabela semanal detalhada de um utente
- 0. Sair

- a) Carregar dados dos funcionários
- b) Carregar ementa semanal
- c) Carregar escolhas dos utentes

...

digite a opção pretendida...

# Capítulo 6

## Resultados e Testes

Foram realizados vários testes de carregamento de dados, consulta e geração de relatórios. O sistema demonstrou robustez na validação de dados e flexibilidade na análise estatística.

### 6.1 Dificuldades Encontradas

- Garantir a integridade dos dados ao ler ficheiros com diferentes formatos.
- Gerir corretamente a memória dinâmica, evitando fugas de memória.
- Implementar validação rigorosa de datas e limites diários.
- Otimizar a navegação e pesquisa nas listas ligadas.



# Capítulo 7

## Conclusão

O projeto permitiu consolidar conhecimentos de programação em C, gestão de memória, estruturas de dados dinâmicas e documentação técnica. O sistema desenvolvido cumpre todos os requisitos propostos, sendo facilmente extensível e adaptável a novas funcionalidades.

### 7.1 Trabalho Futuro

- Implementação de interface gráfica.
- Integração com base de dados relacional.
- Exportação de relatórios em formatos PDF/Excel.
- Otimização de algoritmos para grandes volumes de dados.

# Capítulo 8

## Anexos

### 8.1 Estruturas de Dados

```
// Exemplo de estrutura Funcionario
typedef struct {
    int numero;
    char nome[MAX_NOME];
    int nif;
    int telefone;
} Funcionario;
```

### 8.2 Documentação Automática

A documentação técnica detalhada do código-fonte foi gerada automaticamente com Doxygen e encontra-se disponível na pasta `doc`.