



Relatório Projeto P00

REALIZADO POR:

FRANCISCO QUERIDO-2021221158 | MIGUEL BRAGA-2021221519

DOCENTE: NUNO PIMENTA



1 - Objetivo do projeto:

O projeto final da cadeira de POO tem como objetivo pôr em prática as competências desenvolvidas ao longo do semestre. Para a realização deste trabalho foi necessário implementar soluções relativamente ao funcionamento e manipulação de classes e relações entre as mesmas para poder trabalhar as várias funcionalidades implementadas, tratamento de ficheiros de forma a poder ler e guardar informação, assim como interfaces gráficas para criar um menu interativo para o utilizador.

2 - Estrutura do projeto:

O projeto consiste na gestão da StarThrive, uma empresa responsável pela gestão de várias outras, mais pequenas, empresas.

Estas outras empresas são caracterizadas pela sua categoria, nome, distrito e localização (coordenadas GPS). A categoria das empresas pode ser restauração (café, pastelaria ou restaurante) ou mercearia (frutaria ou mercado).

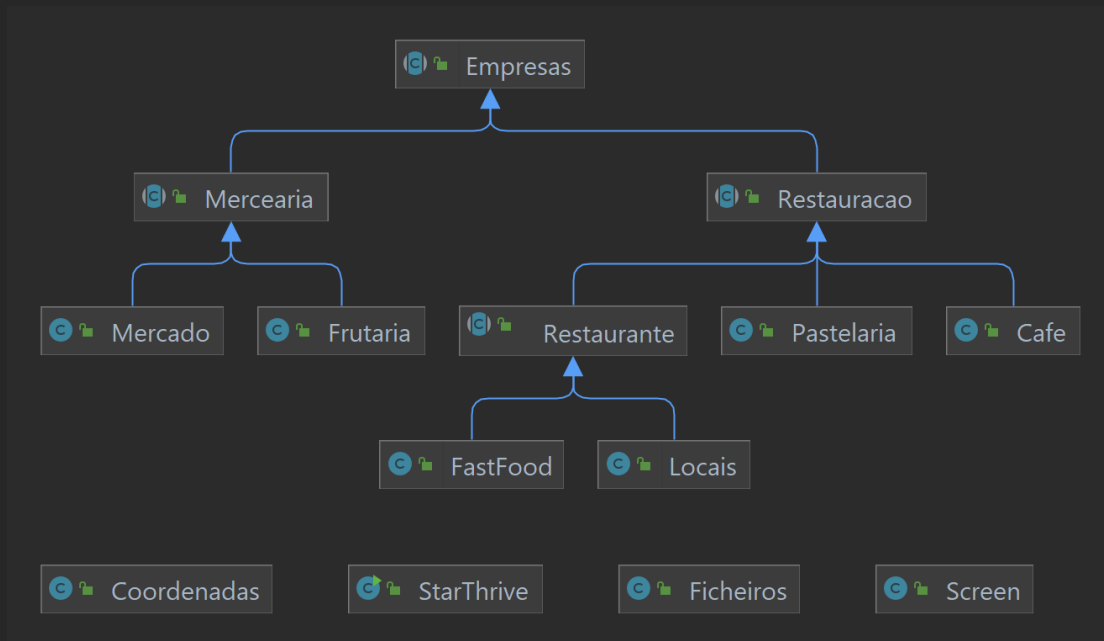
As empresas da restauração têm como grande despesa o custo dos empregados de mesa e o custo médio do salário anual dos empregados. Os cafés são também caracterizados pelo número médio de clientes diário, número médio de cafés que vendem por dia e o valor médio de faturação anual por café vendido por dia. As pastelarias são também definidas pelo número médio de clientes diário, número de bolos que vendem por dia e o valor médio de faturação anual por bolo que vendido por dia. Por último, os restaurantes podem ser locais ou “fast-food”. Qualquer restaurante é caracterizado pelo número médio de clientes diário e pelo número de dias de funcionamento por ano. Um restaurante local tem de ter ainda em consideração o número de mesas interiores, número de mesas de esplanada, custo de licença anual por mesa de esplanada e pelo valor médio de faturação de cada mesa por dia, enquanto que um restaurante de “fast-food” é caracterizado ainda pelo número de mesas interiores, valor médio da faturação de cada mesa por dia, pelo número médio de clientes “drive-thru” e pelo valor médio diário de faturação de cada um destes clientes.

No caso da empresa ser categorizada como mercearia, esta pode ser considerada uma frutaria, caracterizada adicionalmente pelo número de produtos, valor médio de faturação anual e pelo custo anual de limpeza do estabelecimento ou, alternativamente, ser considerada um mercado. No caso de ser um mercado, o estabelecimento é também definido pelo tipo de mercado (mini / super / híper), pela área dos seus corredores (m^2), pelo valor de faturação média por m^2 e também pelo custo anual de limpeza do local.



Como é possível observar, o projeto trabalha vários tipos (e subtipos) de empresas com diversos dados. De forma a podermos trabalhar informação é necessário separar todas as empresas que não tenham dados semelhantes em diferentes classes.

Na figura abaixo é possível observar o diagrama de classes do nosso projeto:





3 – Classes:

Em Java, as classes podem se relacionar entre si através do Polimorfismo e da Herança. Para poder trabalhar os vários tipos de informação que cada empresa apresenta foram criadas diversas classes que comunicam entre si através destas relações. As classes apresentam os seus atributos, construtores e métodos, que serão explicados individualmente abaixo. De forma a não repetir a mesma informação ao longo destas cadeias de classes, optámos por apenas expor e descrever os métodos de cada classe onde são realizadas alterações a nível de variáveis importantes para o funcionamento do programa e métodos abstratos. Por outras palavras, Construtores, “Getters”, “Setters” e outros métodos que apenas definam variáveis de outras classes na classe em questão foram omitidos.



3.1 – Empresas:

A superclasse “Empresas” é uma classe abstrata e pode ser considerada a “base” de qualquer empresa gerida pela StarThrive. Isto, pois esta classe possui características (atributos e métodos) comuns às suas subclasses mas que se vão obter de forma diferente dependendo da subclasse que se inserem.

Atributos:

- Nome da Empresa (nome);
- Distrito da Empresa (distrito);
- Coordenadas GPS (gps); -> Objeto da Classe Coordenadas

Métodos:

- Despesa Anual (despesaAnual);
- Receita Anual (receitaAnual);
- Lucro (lucro): Receita Anual – Despesa Anual;
- Capacidade (capacidade): return -1 -> Caso o valor desta variável se mantenha -1, então a empresa em questão é da categoria “Mercearia”, se for da categoria “Restauracao” então a sua capacidade será um valor positivo definido nesta classe;

NOTA: Métodos Despesa Anual e Receita Anual não têm código (são apenas declarados) pois são implementados de forma diferente dependendo da categoria da empresa.



3.1.1 – Restauração:

A classe “Restauracao” é uma subclasse abstrata de “Empresas” que, como é lógico, preserva os atributos e métodos da classe “Empresas” e implementa outros atributos e métodos. Desta classe derivam as classes “Restaurante”, “Pastelaria” e “Café”, ou seja, os 3 tipos de empresas da categoria restauração.

Atributos:

- Média de Salário Anual dos Empregados (custoSalMedAnual);
- Número de Empregados (nrEmpMesa);
- Número Médio de Clientes por Dia (nrMedClientesDia);

Métodos:

- Despesa Anual (despesaAnual): Números de Empregados * Média de Salário Anual dos Empregados;
- Receita Anual (receitaAnual);
- Categoria (getCateg): “Restauração”
- Capacidade (capacidade): return Número Médio de Clientes por Dia

NOTA: Método Receita Anual não tem código (é apenas declarado) pois é implementado de forma diferente dependendo da subcategoria da empresa.



3.1.1.1 – Café

A classe “Cafe” é subclasse da classe “Restauracao”, logo conserva os atributos e métodos da sua superclasse e adiciona os seus próprios:

Atributos:

- Número de Cafés vendidos por Dia (nrCafeVendDia);
- Média de Faturação por Café vendido por Dia (medFatCafeVendDia);

Métodos:

- Receita Anual (receitaAnual): Média de Faturação por Café vendido por Dia * Número de Cafés vendidos por Dia;



3.1.1.2 – Pastelaria

A classe “Pastelaria” é subclasse da classe “Restauracao”, logo conserva os atributos e métodos da sua superclasse e adiciona os seus próprios:

Atributos:

- Número de Bolos vendidos por Dia(nrMedBolosDia);
- Média de Faturação por Bolo vendido por Dia (medFatBoloVendDia);

Métodos:

- Receita Anual (receitaAnual): Média de Faturação por Bolo vendido por Dia * Número de Bolos vendidos por Dia;



3.1.1.3 – Restaurante:

A classe Restaurante é uma subclasse abstrata de “Restauracao”, que preserva os seus atributos e métodos e implementa outros novos de acordo com as necessidades das empresas desta subcategoria. Desta classe derivam as classes “Locais” e “Fastfood”, ou seja, os 2 tipos de restaurantes.

Atributos:

- Número Médio de Clientes por Dia (nrMedClientesDia);
- Número de Dias de Funcionamento num Ano (nrDiasFuncAno);

Métodos:

- Receita Anual (receitaAnual): return Número de Dias de Funcionamento por Ano, para passar esta variável para o método Receita Anual das suas subclasses



3.1.1.3.1 – Locais:

A classe “Locais” é subclasse de “Restaurante”, ou seja, preserva as suas variáveis e métodos, e acrescenta ainda outros novos.

Atributos:

- Número de Mesas Interiores (nrMesasInt);
- Número de Mesas Esplanada (nrMesasEsp);
- Custo Licença Anual por Mesa Esplanada (custoLicAnMesaEsp);
- Valor Médio de Faturação por Mesa por Dia (valMedFatMesaDia);

Métodos:

- Despesa Anual (despesaAnual): valor da Despesa Anual da Superclasse + Número de Mesas Esplanada * Custo Licença Anual por Mesa Esplanada;

- Receita Anual (receitaAnual): (Número de Mesas Esplanada + Número de Mesas Interiores) * Valor Médio de Faturação por Mesa por Dia * Número de Dias de Funcionamento por Ano (valor guardado no próprio método na Superclasse)



3.1.1.3.2 – Fast-Food:

A classe “Fastfood” é subclasse de “Restaurante”, ou seja, preserva as suas variáveis e métodos, e implementa mais de acordo com as necessidades deste tipo de estabelecimento.

Atributos:

- Número de Mesas Interiores (nrMesasInt);
- Número de Clientes Drive-Thru (nrClientesDriveThru);
- Valor Médio de Faturação por Mesa por Dia (valMedFatMesaDia);
- Valor Médio de Faturação por Cliente Drive-Thru (ValMedClienteDriveThru);

Métodos:

Receita Anual (receitaAnual): (Número de Mesas Interiores * Valor Médio de Faturação por Mesa por Dia + Número de Clientes Drive-Thru * Valor Médio de Faturação por Cliente Drive-Thru) * Número de Dias de Funcionamento por Ano (valor guardado no próprio método na Superclasse);



3.1.2 – Mercearia:

A classe “Mercearia” é uma subclasse abstrata de “Empresas” que, como é lógico, preserva os atributos e métodos da classe “Empresas” e implementa outros atributos e métodos. Desta classe derivam as classes “Frutaria” e “Mercado”, ou seja, os 2 tipos de empresas da categoria mercearia.

Atributos:

- Custo Limpeza por Ano (custoLimpAno);

Métodos:

- Receita Anual (receitaAnual);
- Despesa Anual (despesaAnual): Custo Limpeza por Ano;
- Categoria (getCateg): “Mercearia”;

NOTA: Método Receita Anual não tem código (é apenas declarado) pois é implementado de forma diferente dependendo da subcategoria da empresa.



3.1.2.1 – Frutaria:

A classe “Frutaria” é subclasse da classe “Mercearia”, logo conserva os atributos e métodos da sua superclasse e adiciona os seus próprios:

Atributos:

- Número de Produtos (nrProd);
- Valor Médio de Faturação Anual por Produto (medFatProdAno);

Métodos:

- Receita Anual (receitaAnual): Número de Produtos * Valor Médio de Faturação Anual por Produto;



3.1.2.2 – Mercado:

A classe “Mercado” é derivada da classe “Mercearia”, ou seja, preserva as suas variáveis e métodos, e implementa mais de acordo com as necessidades deste tipo de estabelecimento.

Atributos:

- Tipo (tipo);
- Área Corredores [m²] (areaCorredores);
- Valor Médio da Faturação Anual por m² por Ano (medFatMAno);

Métodos:

- Receita Anual (receitaAnual): Área dos Corredores * Valor Médio da Faturação Anual por m² por Ano;



4 – Funcionamento do Programa:

Ao correr o programa este tenta ler, primeiramente, um ficheiro de objetos com as informações das empresas. Se este ficheiro não for encontrado ou não for possível lê-lo, é lido, alternativamente, um ficheiro de texto com as informações sobre os diferentes tipos de empresas.

Se o programa conseguir obter a informação guardada, este abre o menu gráfico. A partir deste menu é possível realizar várias operações, listar várias características das empresas, nome, tipo, distrito, despesa anual, receita anual e por fim, se é lucrável, assim como apresentar para cada tipo de empresa, a empresa com maior receita anual (nome e valor), a empresa com menor despesa anual (nome e valor) e ainda a empresa com maior lucro anual (nome e valor do lucro). Outra funcionalidade da nossa aplicação é a apresentação das 2 empresas de restauração com maior capacidade de clientes por dia.

O programa permite ainda ao utilizador a manipulação de informações da base de dados, criar, editar e apagar empresas da mesma e ainda listar as empresas e indicadores desejados pelo utilizador.

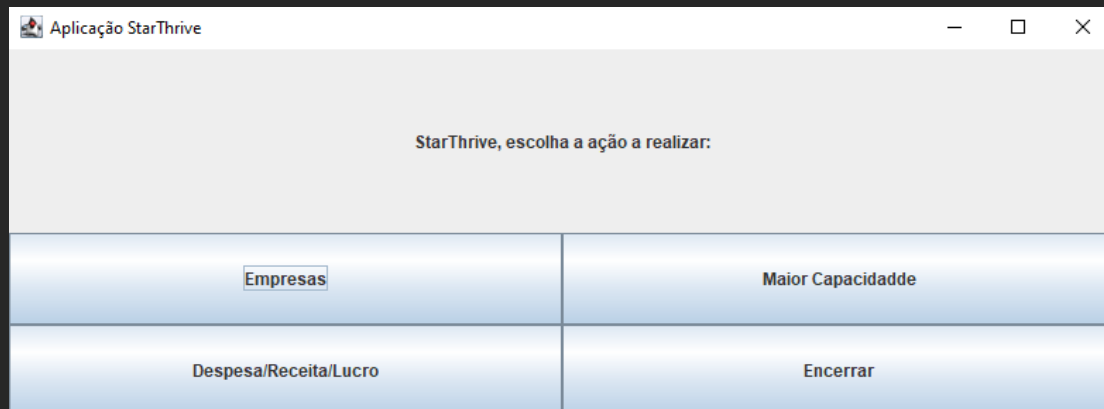
Ao editar a base de dados o programa automaticamente altera o ficheiro de objetos de acordo com as mudanças efetuadas.

4 – Manual do Utilizador:

O utilizador interage com o programa através da sua interface gráfica. O Manual do Utilizador explica e esclarece quaisquer dúvidas que o utilizador pode ter ao utilizar a aplicação.

4.1 – Menu Principal:

O menu do nosso projeto tem a seguinte aparência:



Como pode observar, o menu apresenta 4 janelas diferentes:

- Empresas
- Despesa/Receita/Lucro
- Maior Capacidade
- Encerrar

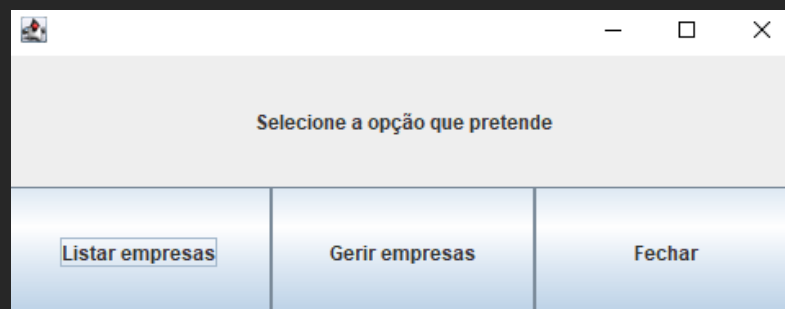
Os 3 primeiros menus levam o utilizador para os “Menus Funcionalidades” do aplicativo, sendo que a última janela simplesmente encerra o programa.

4.2 – Menus Funcionalidades:

4.2.1 – Empresas:

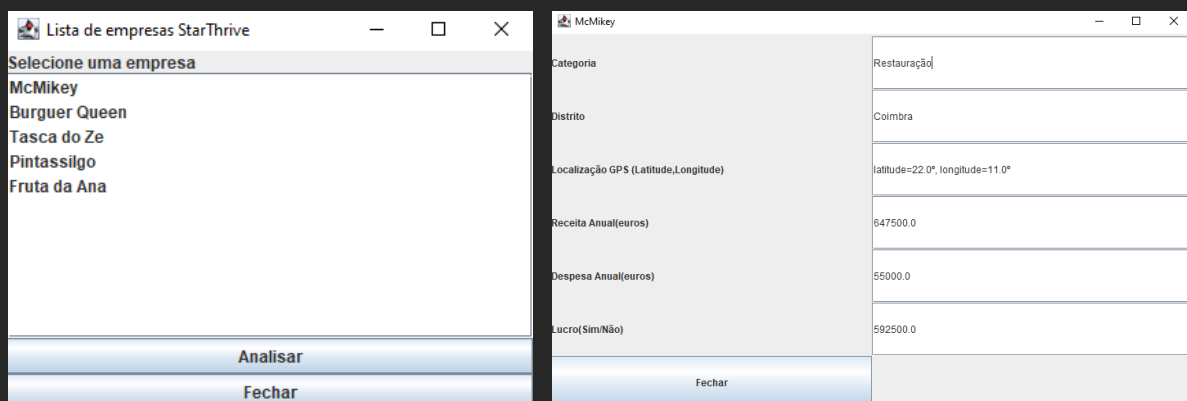
Menu Empresas tem 3 janelas diferentes:

- Listar Empresas
- Gerir Empresas
- Fechar



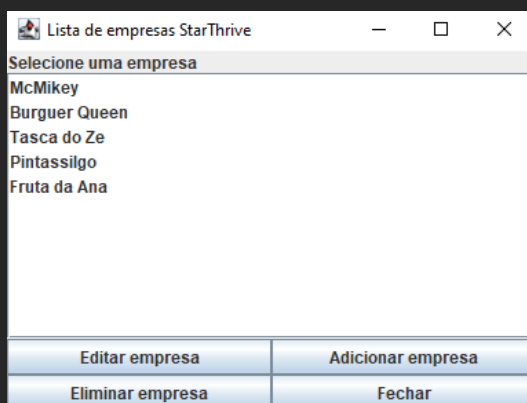
4.2.1.1 – Listar Empresas:

Ao clicar em “Listar Empresas” abre o menu da esquerda com as várias empresas, de seguida, ao selecionar uma empresa e clicar “Analisar”, o programa abre o menu da direita onde poderá conferir informações sobre a empresa selecionada.



4.2.1.2 – Gerir Empresas:

Ao clicar em “Gerir Empresas” abre um menu com as empresas, de seguida, deverá seleccionar a opção pretendida (caso queira Editar ou Eliminar uma empresa deverá seleccioná-la primeiro)



4.2.2 – Despesa/Receita/Lucro:

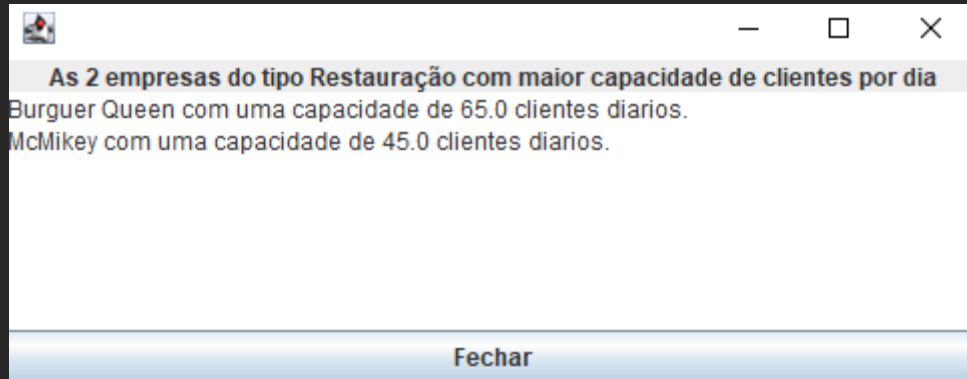
Se seleccionar esta opção, irá abrir um menu com informações sobre as empresas de cada categoria com:

- Maior Receita
- Menor Despesa
- Maior Lucro



4.2.3 – Maior Capacidade:

Ao escolher esta janela, o programa abrirá esta janela onde estarão expostas, para consulta, as duas empresas do tipo Restauração com maior capacidade de clientes por dia.



5 – Conclusão:

Resumidamente, este projeto explorou os nossos conhecimentos da cadeira de POO a um nível que as aulas não permitem e desafiou-nos e incentivou-nos a pesquisar soluções para os nossos problemas.

Deste modo, é possível afirmar que os conceitos desta cadeira estão agora mais bem consolidados.