

Empresa de Táxis

Algoritmos e Estruturas de Dados

2016/2017 – 1º Semestre

20 de Novembro de 2017

MIEIC

Prof. Ana Paula da Cunha Rocha

Ana Santos up200700742@fe.up.pt

Francisca Cerquinho up201505791@fe.up.pt

Mariana Silva up201506197@fe.up.pt

Índice

Introdução.....	2
Descrição sucinta do tema do trabalho	3
Descrição da solução implementada	4
Fases de implementação	4
Divisão estrutural	4
Descrição detalhada do funcionamento da aplicação	6
Principais funcionalidades	7
Diagrama UML do modelo de dados	9
Lista de Casos de Utilização para a aplicação	12
Relato das principais dificuldades encontradas no desenvolvimento do trabalho	13
Indicação do esforço dedicado por cada elemento do grupo	14
Conclusão.....	15
Bibliografia	16

Introdução

No âmbito da unidade curricular Algoritmos e Estrutura de Dados foi-nos proposto a realização de um projeto com base na linguagem de programação C++, que assenta num paradigma algorítmico e procedimental, pondo à prova, desta forma, os nossos conhecimentos relativamente às operações básicas CRUD (*Create, Read, Update, Delete*). O tema escolhido foi a implementação de um sistema informático para a gestão de serviços de uma empresa de táxis. Embora este enunciado se revista de uma aparente simplicidade, é na realidade desafiador das nossas capacidades e muito útil no dia-a-dia de uma empresa de táxis.

Efetivamente, ao longo deste relatório iremos abordar o modo de elaboração desta aplicação, identificando as estruturas utilizadas e suas respetivas relações. Teremos em conta, também, o papel deste projeto do ponto de vista de utilidade na vida real. No final, iremos abordar as principais dificuldades encontradas no desenvolvimento do trabalho, bem como a indicação do esforço dedicado por cada uma de nós.

Desta forma, procuramos responder ao que nos é exigido, de forma sucinta e explícita, sendo nossa intenção fazer com que o presente relatório sirva de guia e suporte para os interessados na aplicação.

Descrição sucinta do tema do trabalho

Esta aplicação baseia-se na gestão dos serviços prestados por uma corrente empresa de Táxis, permitindo a manipulação de todas as funcionalidades que, usualmente, uma central de Táxis deve possuir, designadamente na administração de clientes, serviços e rotas. Existirão clientes de dois tipos, com e sem registo. Dentro dos registados, há os clientes empresariais e particulares e os não registados, os ocasionais. Cada cliente possui um conjunto de propriedades identificadoras quer pessoais, quer relativas ao seu histórico de serviços e ao seu número de pontos pelo valor das viagens realizadas. Os serviços têm associado um custo que vai depender da duração da viagem e da distância percorrida e o seu pagamento pode ser efetuado de quatro formas diferentes, nomeadamente, por cartão multibanco e numerário, permitidas a todos os tipos de clientes, por cartão de crédito com acréscimo de 5% do preço do serviço ou ao fim do mês com acréscimo de 2%, restritas apenas a clientes registados. Esta aplicação desenvolve também promoções aos seus clientes registados, sendo acumulados pontos, se pertencerem a clientes isolados (não associados a companhias), ou valores determinados pelo pagamento efetuado, se forem clientes empresariais.

Descrição da solução implementada

Numa primeira fase, após a escolha do tema, tentamos compreender aquilo que nos era exigido, isto é, quais os conteúdos que deviam reger o nosso projeto, seguido de uma fase de pesquisa, onde nos tentamos informar ao máximo quais as funcionalidades fulcrais que numa aplicação deste género não poderiam faltar. Depois destas duas fases, esquematizamos o problema e dividimo-lo em estruturas adequadas. Recorremos, também, ao diagrama UML que para além de constituir um guia ao desenvolvimento do projeto, serve para o utilizador perceber a forma como foi desenhada a solução.

Fases de implementação

1. Compreender o que é pedido;
2. Pesquisa;
3. Esquematização da aplicação de acordo com os requisitos e divisão em estruturas;
4. Desenho do diagrama UML;
5. Implementação do código.

Divisão estrutural

Resolvemos dividir o trabalho em três classes principais (*CentralTaxis*, *Service* e *Customer*), em duas classes auxiliares (*Route*, *Date* e *Voucher*) e em duas classes derivadas da classe *Customer* (*CompanyCustomer* e *PrivateCustomer*), tornando o código mais modularizado.

A classe *CentralTaxis* foi criada para gerir todas as outras, permitindo o acesso a vetores e ficheiros correspondentes às classes *Service*, *Customer*, *Route* e *Voucher*, ler, guardar, inserir e remover dos ficheiros relativos aos serviços, clientes e rotas percorridas e, também, mostra os descontos disponíveis. Esta classe funciona como um sistema para administrar todas as outras.

A classe *Service* diz respeito ao modo de processamento dos serviços da empresa de Táxis, sendo detentora da informação sobre o cliente que efetua o serviço, a respetiva data, o custo e o trajeto percorrido. Dentro desta classe foi criada uma estrutura do tipo enumerador utilizada para restringir os possíveis tipos de pagamento desta empresa (numerário, cartão multibanco, cartão de crédito ou ao fim do mês).

A classe *Customer* agrega a informação relativa a um cliente, nomeadamente o seu NIF, nome, morada, número de telefone e o número de pontos acumulados. Dentro desta classe, foram criadas duas classes para exceções, nomeadamente *InvalidNifException* e *InvalidPhoneNumberException* que verificam se o nif e o número de telefone inseridos pelo utilizador são válidos. As suas classes derivadas, *CompanyCustomer*, diz respeito aos clientes pertencentes a empresas e a *PrivateCustomer*, a clientes isolados.

A classe *Route* detém as características do percurso, a origem, destino, distância percorrida e duração da viagem. No interior desta classe, encontram-se duas classes para exceções, *InvalidDistanceException* e *InvalidExpectedTimeException*, que apuram se os valores da distância e do tempo esperado são válidos.

A classe *Date* refere-se à manipulação das datas dos serviços, possibilitando a comparação de datas entre os serviços, obter o ano, mês e dia, verificar se são ou não válidas, nomeadamente se o número de dias do mês ou o número de meses do ano são coerentes ou se o ano inserido é bissexto.

Estes dados são guardados em quatro ficheiros diferentes (*services.txt*, *customers.txt*, *routes.txt* e *vouchers.txt*). O ficheiro relativo aos serviços é constituído pelo NIF dos clientes, a origem, o destino, a data, duração e distância do trajeto percorrido e o tipo de pagamento efetuado. O ficheiro com informação dos clientes é formado pela indicação do seu tipo, NIF, nome, morada, contacto telefónico e, no caso dos *PrivateCustomer*, o seu número de pontos e no que no caso dos *CompanyCustomer*, o seu montante acumulado. O ficheiros das rotas inclui todos os percursos já efetuados pela empresa, com referência à origem, destino, distância em quilómetros e tempo esperado. Por último, o ficheiro dos vouchers é apenas referente

às empresas clientes, uma vez que os seus descontos, ao contrário dos restantes, têm um tempo limite de um mês. Este ficheiro é formado pelo NIF, data e a percentagem de desconto a ser aplicado.

Consideramos pertinente a utilização de *maps*, nomeadamente na associação do voucher ao respetivo cliente (*map<int, Voucher*>*), para tornar a pesquisa a partir do NIF do cliente ainda mais eficiente.

Descrição detalhada do funcionamento da aplicação

Ao abrir a aplicação, surge um menu principal onde é pedido para escolher uma de cinco opções (*"Customer management"*, *"Services management"*, *"Routes management"*, *"Show discounts availables"*, *"Exit program"*). Dentro de cada uma das opções são abertos submenus que permitem ao utilizador escolher de uma forma simples e rápida as opções básicas desta aplicação.

Ao escolher a primeira opção, abre um submenu, com outras seis opções, onde as primeiras três são para mostrar informação acerca de todos os clientes, de apenas um ou por tipo com um submenu associado para possibilitar a pesquisa através do nif ou do seu nome. A quarta opção é a edição das propriedades de um cliente (nome, morada ou número de telefone) e as últimas duas é a possibilidade de remoção, através unicamente do nif, e inserção de um cliente da lista de clientes do ficheiro. Para inserir, primeiramente, é necessário escolher o tipo de cliente (Company Customer ou Private Customer) e dar informações acerca do nif, nome, morada e contacto telefónico.

Se for a segunda opção a selecionada, surge um submenu onde é dada a possibilidade de ver todos os serviços efetuados, ver os serviços apenas de um dia, entre duas datas escolhidas pelo utilizador, mostrar os serviços de apenas um cliente e inserir um novo serviço. Nesta última é questionado ao cliente se é ou não registado. Caso seja registado é lido o seu nif, origem, destino, duração, distância e tipo de pagamento (*"Cash"*, *"ATM"*, *"Credit"* ou *"End of month"*). Se não for registado, o

cliente pode proceder à inserção do novo serviço e registar-se ou continuar sem registo e se for esse o caso só é permitido efetuar o pagamento por cartão multibanco ou em numerário.

Na terceira opção, será permitido ao utilizador ver todas as rotas disponíveis, remover e inserir rotas. Nesta opção é pedido para colocar o nome da origem, destino, distância e duração da viagem.

A última opção deste menu principal consiste na visualização de todos os descontos disponíveis e na respetiva data de validade.

Principais funcionalidades

Sumarizando, a aplicação permite a inserção e remoção de um cliente, de um serviço ou de uma rota e estas últimas são adicionadas automaticamente caso o serviço prestado efetue uma rota diferente daquelas que fazem parte do ficheiro de rotas. Neste sistema, é possível visualizar a informação quer dos clientes (de forma individual, através do NIF ou nome, e/ou coletiva), quer dos serviços, quer das rotas por ordem alfabética, usando algoritmos de ordenação, a lista dos serviços efetuados num dia ou entre dias escolhidos pelo utilizador. É dada ainda a possibilidade de edição de um cliente à escolha, através da pesquisa sequencial pelo NIF ou pelo seu nome, sendo possível a mudança do seu contacto telefónico, morada e nome.

Quando é adicionado um novo serviço, o utilizador é questionado se é um cliente registado e de forma a confirmá-lo foi criada uma função (*verifyNifAlreadyExist(unsigned int, vector<Customer*>)*) que efetua um algoritmo de pesquisa binária. Logo a seguir, é pedida informação acerca do trajeto percorrido e do tipo de pagamento. Decidiu-se que por cada quilómetro percorrido é cobrado um euro e que, para além do acréscimo de 5% caso o tipo de pagamento escolhido fosse por cartão de crédito ou de 2% se for ao fim do mês, é ainda adicionada uma taxa de cinquenta cêntimos por cada minuto extra percorrido relativamente ao tempo esperado e é atribuído um ponto por cada dez euros pagos pelo cliente.

Quando é calculado o custo do serviço (a soma do número de quilómetros multiplicado pelo preço por cada quilómetro com o tempo extra multiplicado por cinquenta cêntimos), é verificado, no caso de ser um cliente empresarial, se possui vouchers disponíveis e se está dentro do prazo de validade estipulado (um mês), no caso dos restantes clientes não existem vouchers, é apenas verificado se o número de pontos acumulado é suficiente para emitir desconto e é apenas descontado se o cliente desejar, podendo optar por acumular pontos para obter o desconto máximo. Ainda relativo aos clientes empresariais, a atribuição de vouchers é feita com base no montante acumulado dos últimos serviços, se estes possuírem um valor acumulado inferior a duzentos e cinquenta euros não têm direito a vale, se for compreendido entre duzentos e cinquenta e quinhentos euros é atribuído um desconto de cinco por cento, entre quinhentos e setecentos e cinquenta euros, dez por cento, entre setecentos e cinquenta e mil euros, quinze por cento e se ultrapassar os mil euros, tem no máximo vinte por cento de desconto. Caso esse tipo de clientes tiverem já um voucher de quinze por cento e obterem outro de dez por cento, o máximo que podem descontar é vinte por cento, isto é, o montante é gerado cada vez que é emitido um vale, mas se entretanto o valor for suficiente para outro vale, este pode ser acumulado. No que diz respeito aos clientes privados, se tiverem menos de cinquenta pontos não têm desconto no serviço, entre cinquenta e cem é atribuído um desconto de dez por cento, entre cem e cento e cinquenta, de vinte por cento, entre cento e cinquenta e duzentos, de trinta por cento, entre duzentos e duzentos e cinquenta, de quarenta por cento e, por fim se tiverem um número de pontos superior a duzentos e cinquenta, têm no máximo cinquenta por cento de voucher. Quando utilizam os pontos acumulados, estes recomeçam novamente em zero.

Diagrama UML do modelo de dados



Figura 1 Classe Central Táxis



Figura 2 Classe Service

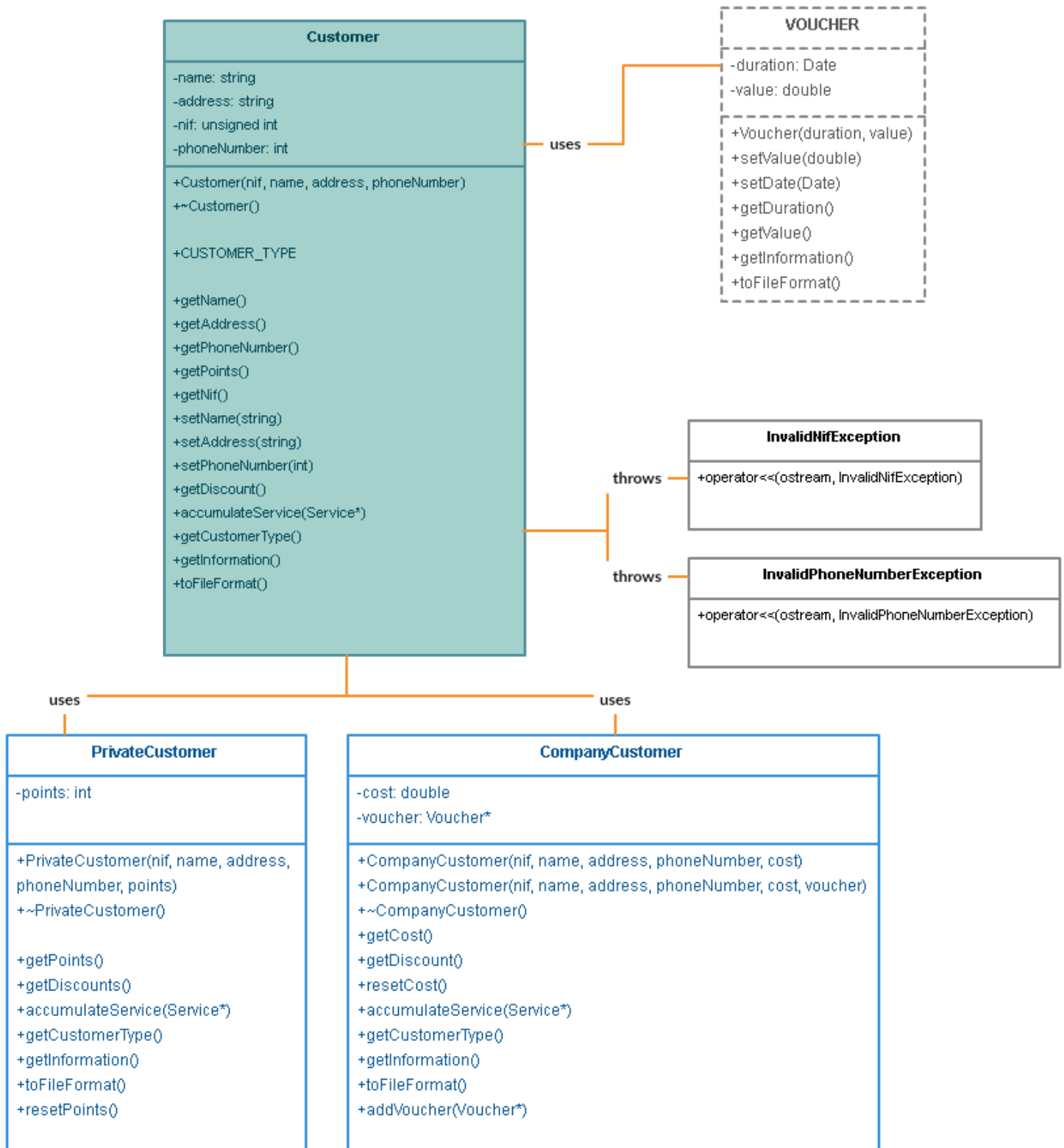


Figura 3 Classe Customer

Lista de Casos de Utilização para a aplicação

É notória a aplicabilidade que esta aplicação tem numa qualquer empresa de transportes. Além de guardar toda a informação necessária acerca dos seus clientes, serviços e rotas, processa os diversos tipos de pagamentos e descontos de uma forma eficaz e organizada, constituindo, desta forma, um sistema de grande utilidade. Torna, não só possível o transporte de uma entidade individual, como também a mobilidade de funcionários de uma empresa.

Caso este sistema informático tivesse associado um sistema GPS, conseguiríamos de uma forma automática ter acesso à localização do serviço e, desta forma, ao número de quilómetros efetuados, evitando a referência manual das características do percurso, o que o tornaria ainda mais viável e prático. Ainda assim, são inúmeras as circunstâncias de utilização no quotidiano para a aplicação.

Acreditamos que, para encarar o desafio da mobilidade nas grandes cidades, é preciso oferecer uma gama cada vez mais ampla de opções de mobilidade às pessoas e este sistema informático é mais um modo de locomoção na cidade.

Relato das principais dificuldades encontradas no desenvolvimento do trabalho

As funcionalidades planeadas foram, na totalidade, implementadas, embora os critérios onde sentimos mais dificuldade foram a concordância dos intervenientes relativamente à eficácia de determinadas funções, na verificação de entradas inválidas do utilizador e na atribuição dos vouchers de acordo com o tipo de cliente, atribuindo um prazo de validade de trinta dias para usufruir dos descontos no caso dos clientes empresariais, ao contrário dos clientes sem uma entidade associada que podem utilizar o voucher numa única utilização, tendo sido essa verificação do prazo de validade onde sentimos mais dificuldade para retornar o problema.

Indicação do esforço dedicado por cada elemento do grupo

Os elementos do grupo cooperaram entre si de forma a distribuir o trabalho uniformemente e para que houvesse partilha de sugestões e conhecimento entre os três. De uma forma organizada, planeamos a divisão de tarefas de modo a responder eficientemente aos objetivos iniciais. Todos os elementos participaram na construção do algoritmo, na elaboração do relatório e do diagrama UML, colaborando, assim em todas as fases de implementação do projeto.

Conclusão

A realização deste sistema informático para gestão de serviços de uma empresa de táxis, para além de nos ser muito útil do ponto de vista algorítmico e estrutural, são notórias as suas múltiplas vantagens e aplicações no quotidiano de uma entidade de transportes.

De facto, esta aplicação gere, de forma sistemática e funcional, todos os elementos fulcrais constituintes de uma empresa de táxis, tendo sido implementadas todas as funcionalidades básicas e necessárias de gestão de serviços, clientes, rotas, descontos e pagamentos.

Em suma, este trabalho teve uma grande influência no nosso percurso como alunos de Engenharia Informática e Computação, pois permitiu-nos não só solidificar os conhecimentos de conceitos lecionados, como classes, herança e polimorfismo, pesquisa e ordenação, como também saber como procurar e analisar determinados assuntos. Analisando o projeto realizado, consideramos que o objetivo foi cumprido, pois conseguimos desenvolver aquilo a que nos tínhamos proposto inicialmente.

Bibliografia

Cplusplus.com. Acedido em 2016. Disponível em <http://www.cplusplus.com/>.

Rodrigues, P., Pereira, P., Sousa, M. *Programação em C++ - Conceitos básicos e Algoritmos*. FCA.