

Universidade do Minho
Conselho de Cursos de Engenharia
Licenciatura em Engenharia Informática

Desenvolvimento de Sistemas de Software

Ano Lectivo de 2009/10



escola de engenharia



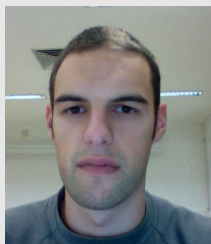
departamento de
informática

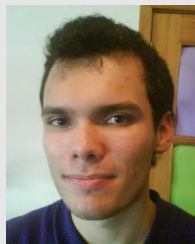
GereComSaber

Cláudio Novais, Henrique Carvalho, Nuno Coelho, Pedro Carvalho, Tiago Dias

Dezembro, 2009

Constituição do Grupo 20

Pedro Carvalho 40514		Tiago Dias 43204	
-------------------------	---	---------------------	---

Henrique Carvalho 49326		Cláudio Novais 49350	
----------------------------	--	-------------------------	--

Nuno Coelho 49412			
----------------------	---	--	--

Projecto

GereComSaber - Gestão de Condomínios

Observações

Avaliação

Resumo

Cada vez mais a quantidade, qualidade e rapidez dos serviços prestados por uma determinada empresa são fundamentais para que se tenha sucesso nas vendas e se consiga ganhar vantagem relativamente à concorrência numa sociedade actual, cada vez mais consumista.

Pelo evoluir das tecnologias e pela crescente oferta de softwares de gestão devido à conjectura actual, torna-se necessário fazer algo inovador que corresponda na totalidade ao que o comprador deste software pretenda. Sendo assim, desenvolveu-se uma aplicação específica para gestão de condomínios capaz de controlar todas as transacções de informação de uma forma dinâmica e com o menor uso de recursos possível.

Conteúdo

Resumo	i
Conteúdo	iii
Lista de Figuras	iv
1 Introdução	1
2 Análise de Requisitos	2
3 Levantamento de Requisitos	4
3.1 Objectivos	4
3.2 Decisões pertinentes	5
4 Desenho do sistema	7
4.1 Modelo de domínio	7
4.2 Diagramas de Use Cases	8
4.2.1 Diagrama de sistema	8
4.3 Diagramas de Sequência	14
4.3.1 Sistema	14
4.3.2 Gestão de Clientes e Gestão de funcionários	14
4.3.3 Gestão de Fornecedores	15
4.3.4 Gere Contracto Cliente	15
4.3.5 Consulta Estatísticas	17
4.4 Diagrama de Classes	18
5 Conclusões e Trabalho Futuro	20
Bibliografia	21

Referências WWW	22
A Anexos	23

Lista de Figuras

4.1	Modelo de domínio	8
4.2	Diagrama de Sistema	9
4.3	Diagrama de gestão de Funcionários	10
4.4	Diagrama de gestão de Fornecedores	11
4.5	Diagrama de gestão de Serviços do Cliente	12
4.6	Diagrama de Consulta de Estatísticas	13
4.7	Diagrama de Classes	19

1 Introdução

No âmbito da disciplina de Desenvolvimento de Sistemas de *Software* foi-nos proposta a realização de um projecto de software relativo a uma empresa de gestão de condomínios, **GereComSaber**, que fosse capaz de assegurar os diversos trabalhos a efectuar em casa dos condóminos.

Sendo assim, a equipa responsável pela realização do projecto decidiu que o problema a resolver seria baseado software de gestão que permitisse celebrar contratos, adicionando serviços que serão prestados aos condóminos através de terceiros, bem como a gestão da facturação individual de cada cliente.

De facto, e com o surgir de novas exigências por parte da sociedade moderna, torna-se cada vez mais necessário criar um sistema capaz de automatizar todo o processo de gestão de serviços em condomínios fechados, garantindo um acesso rápido e eficaz à informação, por forma a facilitar todo o processo de prestação de serviços, além de permitir uma melhor gestão e um melhor processo de cálculo do valor a pagar por cada contrato. Deverá então, ser disponibilizada aos utilizadores deste sistema, uma plataforma simples e de fácil percepção, que permita o acesso rápido à informação, possuindo igualmente um vasto leque de funcionalidades de utilidade comprovada.

Propusemo-nos então realizar um projecto de software que permita ao utilizador aceder facilmente a esta informação, simplificando todo o processo da celebração de contratos e posterior acerto de contas.

Este relatório de acompanhamento está organizado conforme o processo de desenvolvimento de um projecto segundo o Modelo em Cascata. Assim, começa-se por abordar a Análise de Requisitos onde é feita uma primeira análise do problema proposto e selecciona-se os objectivos a cumprir. De seguida, especifica-se o Desenho do Sistema, onde é detalhado o funcionamento do *software* a criar, utilizando para isso a ferramenta de modelação **Visual Paradigm 7.1** proposta pelo docente da disciplina, mostrando as necessidades que devem ser desenvolvidas de forma a facilitar, no futuro, o trabalho dos programadores.

2 Análise de Requisitos

Ter uma boa análise de requisitos é quase obrigatório para poder esclarecer todos os objectivos a cumprir e evitar surpresas logo aquando do desenho do sistema. Assim, de forma a cumprir essa necessidade, foi feita uma pesquisa aprofundada sobre o tema deste projecto e determinaram-se os seguintes requisitos que um programa deste tipo tem em geral:

- Suporte para vários fornecedores;
- Cada fornecedor tem diferentes serviços;
- Cada serviço tem diversas ofertas;
- Pode haver fornecedores com serviços iguais;
- Calcular acréscimos e estornos;
- O estorno pode ser entregue ao cliente sobre a forma de dinheiro ou crédito;
- Cada cliente tem um histórico de compras;
- Sistema deve possibilitar vários tipos de pagamento;
- O *software* deve ter suporte para prestações e controlo de inflação e juros;
- Cliente pode subscrever várias actividades (com custos monetários individuais), de diferentes empresas ou da mesma;
- Existem comissões diferentes por cada serviço;
- Cliente pode adicionar ou retirar serviços quando quiser;
- Emissão de notificações de pagamento;
- Guardar em base de dados o valor a facturar mensalmente;
- Calcular o valor a facturar;

- Cliente pode cancelar o contrato;
- Controlo de impostos;
- Impressão de Factura aquando do negócio e aquando dos pagamentos;
- Possibilidade de impressão de segunda via de factura;
- Amostragem de estatísticas sobre a empresa: melhores fornecedores ao nível económico, serviços mais lucrativos, serviços mais requisitados;
- Possibilidade de descontos/promoções;
- Funcionalidades de automatização de inserção de serviços e ofertas dos fornecedores (*parsers*, por exemplo);
- Funcionalidades de exportação de dados, nomeadamente serviços pedidos pelos clientes a fornecedores, ou toda a facturação para as finanças;
- Interface adequada a diversos tipos de utilizador, funcionários, administradores, fornecedores e clientes;
- Autenticação dos utilizadores;
- Distinção entre os utilizadores, através da autenticação, restringindo os menos importantes.

3 Levantamento de Requisitos

3.1 Objectivos

Depois do grupo de trabalho ter uma ideia do que pode e deve fazer, consoante o tempo e importância de cada requisito, foi decidido que o projecto deverá seguir os objectivos seguintes:

- Desenvolver um sistema fiável e robusto que possibilite a gestão dos diferentes serviços;
- Sistema modular, desenvolvido segundo um modelo iterativo, permitindo assim actualizações fáceis;
- Controlo de utilizadores, com 2 níveis de permissões (administrador e funcionário);
- Suporte para vários fornecedores;
- Cada fornecedor suporta vários serviços;
- Cada serviço contém várias ofertas;
- Cada fornecedor tem uma comissão em concreto;
- Cliente pode efectuar o pagamento de 3 maneiras distintas (pagamento anual, semestral ou trimestral);
- Cliente pode adicionar ou retirar serviços quando quiser (sistema calcula acréscimos/estornos automaticamente);
- Consulta de estatísticas incluirá ver lucro total e lucro que determinado fornecedor dá, num determinado intervalo de tempo;
- Desenvolver um sistema que guarda a informação mensal relativa aos serviços contratados por cada cliente de modo a proceder à facturação de forma simples e automática;

- Facturação automática sem necessidade de funcionário ter de verificar a data de facturação para cada cliente;
- Cálculo de acréscimos/extornos a efectuar devido a mudanças nos serviços no período de facturação anterior.

3.2 Decisões pertinentes

Durante a realização da lista de objectivos, deparámo-nos com alguns problemas que exigiram decisões pertinentes. De seguida apresenta-se a lista dessas decisões:

- Nem o cliente nem o fornecedor interagem directamente com o sistema;
- A gestão de comissões está contida em cada fornecedor, por isso cada fornecedor tem uma comissão própria;
- Existem apenas 3 modos de pagamento: Anual, Semestral e trimestral;
- Por questões de simplificação, quando o cliente cria um contrato este será sempre aceite, isto é, o cliente não pode pedir um orçamento ou fazer uma simulação;
- Cada cliente tem um tipo de pagamento e não cada factura tem um tipo de pagamento;
- Após a primeira prestação, os pagamentos seguintes serão efectuados no início do mês correspondente;
- No caso de alteração para novos serviços, estes serão cobrados na próxima prestação. Essa prestação inclui o valor dos dias anteriores que não foram pagos e inclui o valor dos serviços até à próxima prestação;
- O administrador é o único que consegue alterar comissões, gerir funcionários e fornecedores;
- O Administrador é o único que pode ver as estatísticas;
- O Cliente quando cancela um contrato recebe logo o extorno, não tendo assim que esperar pela próxima factura para o receber, como acontece quando cancela serviços mas não o contrato;
- Cada Cliente pode ter apenas um contrato, sendo este passível de alteração;
- Ao criar o contrato é apenas subscrito um serviço, caso o cliente queira adicionar mais deverá fazê-lo a parte, como alteração;
- Adicionar/Remover serviços a um contrato funciona apenas à unidade;

- A facturação é diária, ou seja, se utilizou um determinado serviço durante um período, será contabilizado cada dia em termos de facturação;
- É assumido que os clientes pagam sempre as facturas no prazo e que nunca ficam a dever dinheiro à GereComSaber.

4 Desenho do sistema

Este capítulo consiste na descrição pormenorizada daquilo que foi realizado em cada uma das diferentes partes aquando da sua especificação no Visual Paradigm. Será feita uma breve análise de cada diagrama, tentando explicitar sucintamente as decisões mais importantes que foram tomadas para a realização do trabalho final.

4.1 Modelo de domínio

O Modelo de domínio é uma representação estrutural dos vários elementos que irão compor o futuro sistema permitindo aos programadores ter uma visão conceptual preliminar deste. Assim, e através deste modelo, será facilitado o trabalho de descrever o comportamento do sistema pelos *Use Cases*. Posto isto, e após a análise extensiva do enunciado do projecto, o grupo de trabalho chegou ao desenho da figura 4.1 para o seu Modelo de domínio.

Como se pode verificar pelos relacionamentos das várias entidades, é possível percorrer um caminho começando sempre pela entidade GereComSaber. Esta entidade está presente no modelo do domínio principalmente com esse objectivo, possibilitar um caminho para uma melhor compreensão do futuro sistema.

Apesar do administrador ter todos os “poderes” de gestão, neste modelo não lhe adicionamos uma relação de criação de contratos, no entanto, também poderá fazê-lo. Para além dele, o funcionário é a única identidade que pode fazer contratos, que são um relacionamento entre um cliente e várias ofertas.

Como na realidade, legalmente só se pode facturar aquando do pagamento do cliente, as facturas só são geradas quando é feito um pagamento.

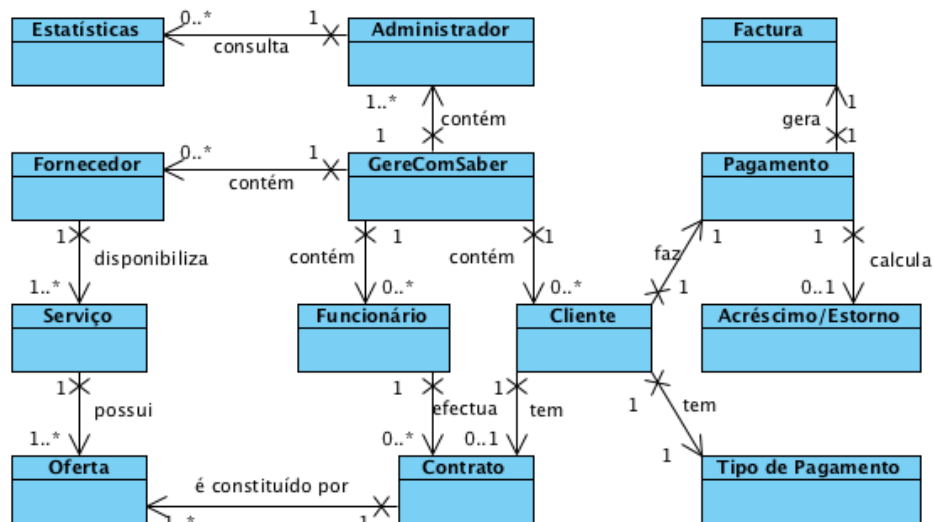


Figura 4.1: Modelo de domínio

4.2 Diagramas de Use Cases

Tal como foi dito anteriormente, os Diagramas de *Use Cases*, servem para descrever o comportamento que os vários intervenientes têm no sistema. Tendo por base o Modelo de Domínio anteriormente especificado, chegou-se a um Diagrama de Sistema que engloba vários sub-diagramas, actores e *Use Cases*, que são explicados a seguir.

4.2.1 Diagrama de sistema

Este diagrama (Figura 4.2) apresenta uma vista geral do sistema a desenvolver. Inclui todos os *Use-Cases* necessários para a resolução do problema proposto, desde a autenticação, gestão de Clientes/Funcionários/Fornecedores até à Gestão de Contratos do Cliente e a consulta de Estatísticas. De seguida mostramos de forma mais detalhada cada um destes componentes, tentando ser sempre o mais explícito e sucinto possíveis.

Diagrama de Gestão de Clientes e Gestão de Funcionários

Com a análise efectuada anteriormente ao problema chegou-se à conclusão de que o sistema teria de manter um registo da informação respeitante a Clientes e Funcionários. Sendo assim, foram os dois divididos nos seus respectivos sub-diagramas de *Use-Cases* tendo estas definidas as opções de adicionar/editar/remover os Clientes/Funcionários como se pode verificar pela figura 4.3.

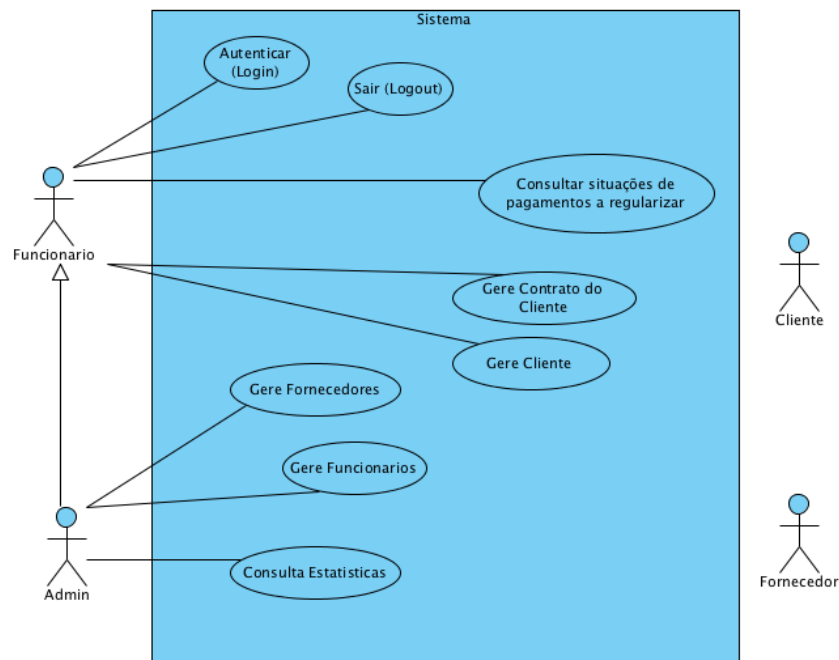


Figura 4.2: Diagrama de Sistema

Diagrama de gestão de fornecedores

Junta-se neste diagrama (Figura 4.4) as opções básicas de adicionar/editar/remover Fornecedor com as opções de criar/editar/remover Serviços uma vez que é óbvio o seu relacionamento.

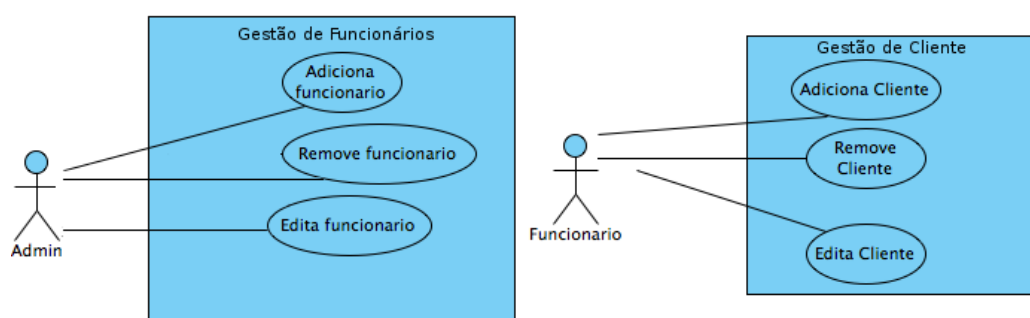


Figura 4.3: Diagrama de gestão de Funcionários

Diagrama de gestão de serviços de cliente

Devido à complexidade deste diagrama (Figura 4.5), a explicação dos vários *Use Cases* será feita através de uma lista:

Adiciona Serviço Antes de mais, o cliente deverá existir no sistema. Depois de introduzir o código do cliente, o funcionário escolhe os serviços a subscrever que serão disponibilizados numa tabela. O sistema actualiza automaticamente o acréscimo que o cliente deverá pagar;

Cria contrato Este *Use Case* começa com a introdução do código do cliente (que já deverá existir no sistema). Depois de verificado se o cliente ainda não possui contrato, o funcionário escolhe os serviços a subscrever que serão disponibilizados numa tabela. O sistema actualiza automaticamente o acréscimo que o cliente deverá pagar.

Calcula diferença factura anterior Este *Use Case* é activado quando se torna necessário calcular o valor de uma factura a pagar pelo Cliente, quer porque chegou um período de pagamento ou então porque o cliente decidiu subscrever novos serviços ou então cancelar algum dos serviços que tinha contratados. Sendo assim, e tendo por base o valor calculado na ultima factura, é utilizado este Use-Case que recalcula o valor a cobrar.

Use-Case que recalcula o valor a cobrar ou a estornar de forma a acertar as contas com o cliente caso seja necessário.

Calcula Valor Factura Para criar uma notificação de pagamento torna-se necessário primeiramente calcular o valor a cobrar. Sendo assim e tendo como base os serviços que o cliente tem contratados no momento mais o valor dos acertos que eventualmente possam existir (créditos/extornos devido à facturação anterior), é calculado o valor que será apresentado na notificação de pagamento e que deverá ser pago pelo Cliente.

Creditar Extorno Quando a notificação de um pagamento apresenta um valor negativo, é necessário fazer um extorno ao cliente, por isso este use case tem como função a

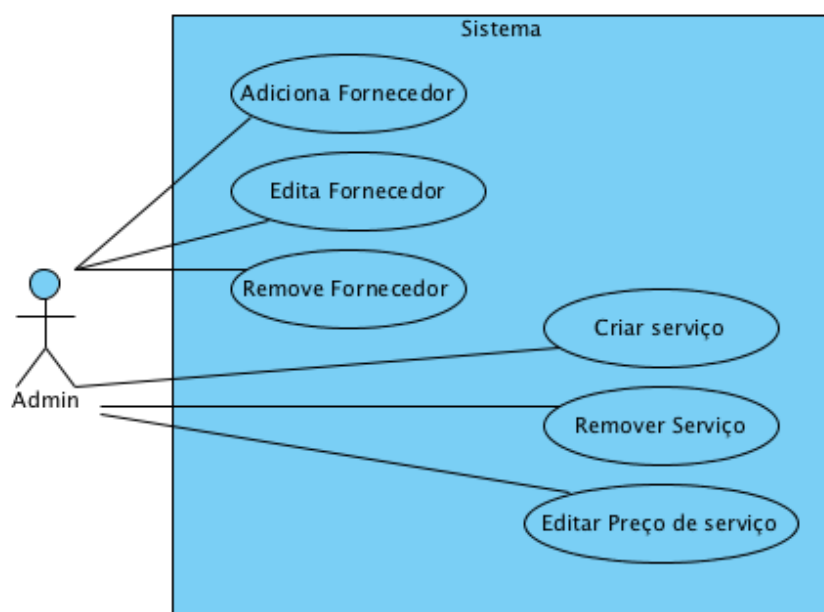


Figura 4.4: Diagrama de gestão de Fornecedores

apresentação de uma mensagem ao funcionário de que terá de fazer um extorno a um determinado cliente com um determinado valor.

Notifica Pagamento Quando é necessário apresentar ao cliente uma notificação de pagamento ao cliente, quer por ele ter efectuado um contrato com a empresa, quer por se encontrar no prazo de pagamento de uma parte do serviço, ou por se encontrar no fim do contrato e ser necessário fazer ajustes de pagamentos, vai criar essa notificação com o valor a pagar.

Calcular Pagamento Mensal Neste use case , através dos serviços contratados de um determinado cliente, vai guardar ou fazer um update na base de dados do valor dos serviços para cada mês que serão facturados, desde o mês em que se encontra até ao final do contrato.

Confirma Pagamento O Funcionário terá que, após um pagamento do cliente, fazer uma confirmação manual desse mesmo pagamento. Será imprimido um talão após a confirmação, esta acção é representada pelo use case "Imprime talão".

Emite Segunda Via de Factura O cliente pode a qualquer momento requisitar a segunda via da factura, por esse motivo o funcionário terá ao seu dispor esta simples funcionalidade, a segunda via da factura à semelhança do "Confirma Pagamento", tem como consequência a impressão de um talão.

Consulta Serviços Disponíveis Cada empresa sub-contratada pela GereComSaber contém a sua lista de serviços, este use case, descreve portanto uma forma de aceder a

essa lista.

Consulta Serviços subscritos pelo Cliente Um cliente pode subscrever a um vasto número de serviços, cada cliente tem portanto um ou mais serviços a si associados. A qualquer momento o actor Funcionário poderá consultar esta lista de serviços contratados pelo cliente.

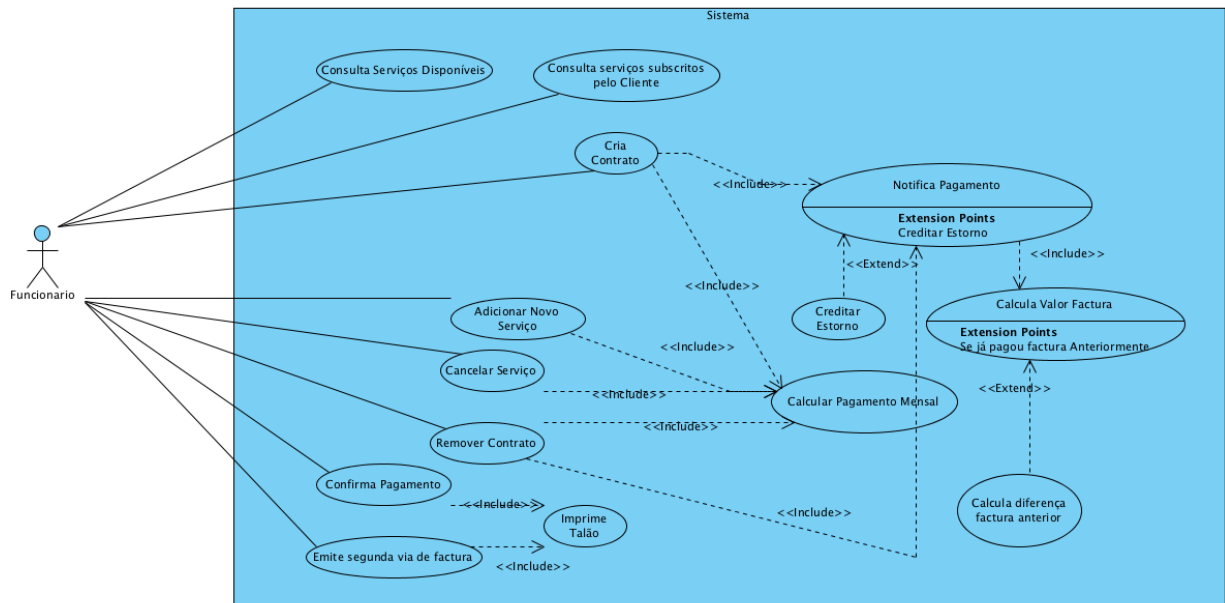


Figura 4.5: Diagrama de gestão de Serviços do Cliente

Diagrama de Consulta de Estadísticas

As estatísticas são sempre uma parte muito importante num sistema de *software* pois fornecem uma imagem clara ao seu utilizador da forma como está a correr o seu negócio. Uma vez que pode conter informações confidenciais, o seu acesso é restrito aos administradores do sistema.

Cancelar serviço Um cliente poderá remover um serviço que tenha adquirido previamente. O funcionário deverá introduzir o código de cliente (pois cada cliente possui apenas um contrato e é o suficiente para o identificar). Seguidamente o sistema apresentará uma lista dos serviços subscritos e o serviço a remover deverá ser escolhido. O use case Calcular Pagamento Mensal é então chamado para calcular o ajuste dos pagamentos e possíveis estornos.

Remover contrato Caso o cliente queira cancelar o contrato, primeiramente o funcionário deverá identificá-lo através do código de cliente. Com base nesse código o sistema

sabe qual o contrato a cancelar e remove todos os serviços subscritos. É invocado o use case Calcular Pagamento Mensal para calcular o ajuste dos pagamentos. Seguidamente o sistema cancela o contrato e o cliente é notificado de possíveis ajustes monetários.

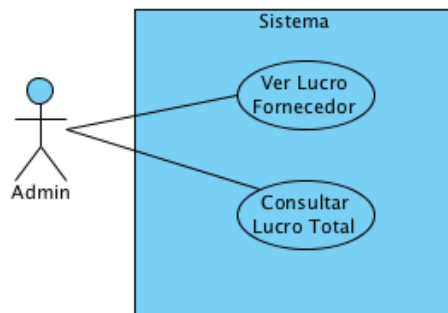


Figura 4.6: Diagrama de Consulta de Estatísticas

Modificações

Os use cases Subscreve Serviço e Remover Serviço Contratado, foram incorporados nos use cases que lhes faziam include. Esta decisão deve-se à sua baixa complexidade (a qual só se tornou perceptível após serem feitos os respectivos diagramas de sequência).

Use case Editar Dados pessoais foi removido devido a redundância de funcionalidade.

4.3 Diagramas de Sequência

Seguindo a ordem de desenvolvimento iniciada neste projecto, a fase seguinte é à modelação dos *Use Cases* e dos seus detalhes é a criação dos diagramas de Sequência. Estes diagramas representam a sequência dos processos, mais especificamente de mensagens passadas entre os sub-sistemas, que devem ser executados em cada *Use Case*. Esta representação deve ser simples e lógica de forma a possibilitar uma rápida compreensão da forma como os grupos de objectos colaboram para um determinado objectivo ao longo do tempo.

4.3.1 Sistema

Autenticar (login) O login (conjunto usuário e palavra-chave) é inserido. Com esse nome de usuário a classe *GerirFuncionário* vai buscar o funcionário correspondente existente em sistema e verifica se a palavra-chave é a mesma. Caso o usuário tenha introduzido mal os dados ou não exista no sistema, é apresentada uma mensagem de erro e é solicitada a introdução de novos dados.

4.3.2 Gestão de Clientes e Gestão de funcionários

Para os sub-*Use Case* definidos anteriormente, o grupo de trabalho procedeu à sua especificação criando os respectivos diagramas de sequência. Tal como foi explicado nos *use cases*, nesta secção juntamos as explicações das duas gestões uma vez que estas são muito similares apenas diferenciando-se em algumas variáveis. Tendo por base os detalhes criados na primeira fase, foi fácil criar um primeiro desenho temporal das interações entre os actores e o sistema. Numa fase posterior estes foram aprofundados de forma a mostrar também todas as interações com os diferentes sub-sistemas.

Adicionar Cliente/Funcionário Estes Diagramas de Sequência começam por criar o objecto que será posteriormente preenchido. Após a inserção dos dados por parte do actor, o sistema verifica-os e, no caso de não haver dados repetidos (BI e contribuinte), o objecto criado é preenchido com os novos dados e posteriormente enviado para a camada de dados.

Editar Cliente/Funcionário Os diagramas de Sequência para editar os dados, funcionam de uma forma um pouco diferente dos anteriores, uma vez que começam por consultar um objecto já criado. Ou seja, inicialmente o actor insere o código respectivo e é verificada a sua existência. Em caso afirmativo, actor poderá digitar as novas alterações e, se válidas, o sistema actualiza os dados da mesma forma que os dados são adicionados no diagrama explicado anteriormente.

Remover Cliente/Funcionário Tal como o diagrama anterior, inicialmente é solicitada a inserção do código referente ao Cliente/Funcionário pretendido. O sistema valida a sua existência e em caso afirmativo questiona o actor se tem a certeza dessa acção. Em caso afirmativo, o sistema irá alterar o estado dessa entidade para inactiva, fazendo com que esta não seja mais apresentada na interface.

4.3.3 Gestão de Fornecedores

Adicionar Fornecedor Inicialmente é criada uma entidade que será posteriormente preenchida pelos dados que o actor insere. Esses dados antes de serem inseridos são verificados e em caso de sucesso o novo fornecedor é adicionado à camada de dados que posteriormente grava-o.

Editar Fornecedor Temporalmente, este diagrama começa pela inserção do código do fornecedor que é verificado quanto à sua existência. Em caso afirmativo, é solicitado ao actor a inserção dos novos dados que, após validação, serão actualizados.

Remover Fornecedor Da mesma forma que o diagrama anterior, o actor começa por inserir o código do fornecedor, o sistema valida a sua existência e pede confirmação. Em caso afirmativo, o estado do fornecedor será alterado para inactivo.

Criar Serviço Após a inserção do código do fornecedor e de ter validada a sua existência, existirá um ciclo que questionará o actor por novos serviços. Para cada serviço, o actor insere o nome que é verificado de forma a evitar dados duplicados. No caso de ser um nome válido, é criado o serviço que terá relacionadas ofertas que são criadas no ciclo logo a seguir. Esse ciclo, tal como o que o contém, também pergunta repetidamente ao actor novas ofertas. Por cada oferta são solicitados os dados referentes a esta, nomeadamente, preço, nome e observações.

Editar Preço de Serviço Inicialmente, o actor insere o código do fornecedor que, se validado, apresenta os vários serviços que o fornecedor disponibiliza. Após a selecção do serviço pretendido é apresentada uma lista com as respectivas ofertas. O actor selecciona a oferta que pretende editar, altera o seu valor e confirma a alteração. Em caso afirmativo o sistema actualizará os dados referentes à oferta modificada.

Remover Serviço Depois de o actor inserir o código do fornecedor e após a sua validação, o actor selecciona um dos serviços disponibilizados. O sistema pede confirmação e, em caso afirmativo, altera o estado do serviço para inactivo.

4.3.4 Gere Contracto Cliente

Consulta Serviços Disponíveis É invocado por um funcionário quando pretende listar os serviços de todos os fornecedores no sistema. É portanto uma acção simples cujo

única excepção se verifica no caso de não haver serviços inseridos no sistema.

Consultar Serviços subscritos por cliente O nome é por si só indicativo da sua utilidade. Um funcionário pretende listar os serviços contratados por determinado cliente. O primeiro passo será então verificar se esse cliente existe, se de facto existe, então irá buscar a lista de serviços, havendo ainda o caso de a lista estar vazia, ou seja, o cliente não ter serviços subscritos, se assim for é mostrada uma mensagem de erro ao utilizador.

Confirma Pagamento Quando um cliente efectua um pagamento o funcionário tem por sua responsabilidade notificar o sistema desse acontecimento. O "confirma pagamento" começa por encontrar o cliente, se este não existe é gerado um erro, e o sistema voltará a pedir de novo a identificação do cliente, caso contrário é feita uma pesquisa por todos os pagamentos não efectuados e devolvida uma lista destes. O funcionário selecciona um desses pagamentos e procede a marcar-lo como pago, o sistema cria então uma factura guardando-a e imprimindo de seguida.

Emite Segunda Via Factura Caso haja necessidade o cliente está no seu direito de evocar a segunda via da factura. Para isso o funcionário introduzirá a identificação do cliente, confirmando o sistema a existência deste, e no caso de não existência é emitido um aviso e o sistema pedirá ao funcionário nova identificação do utilizador. No caso de existir cliente, o sistema devolverá uma lista de facturas para o funcionário escolher, após escolhida a factura em questão, o sistema emitirá a segunda via da factura imprimindo-a. Se a lista de facturas se encontrar vazia é emitida uma mensagem de aviso.

Cancelar Serviço É introduzido o código de cliente. Caso este exista no sistema (a verificação é feita pela classe GerirCliente) é carregado o objecto 'c' correspondente ao cliente em questão. É pedida a lista de ofertas subscritas por esse cliente e o actor deverá escolher a oferta a remover. É então removida do contrato do cliente e é invocado o cálculo do pagamento mensal para regularizar a situação financeira do cliente. Também é re-calculado o lucro da empresa. No final é apresentada uma mensagem de sucesso.

Adicionar Novo Serviço É introduzido o código de cliente. Caso este exista no sistema (a verificação é feita pela classe GerirCliente) é carregado o objecto 'c' correspondente ao cliente em questão. É pedida a lista de ofertas disponíveis e o actor deverá escolher a oferta a adicionar. É então adicionada ao contrato do cliente e é invocado o cálculo do pagamento mensal para regularizar a situação financeira do cliente. Também é re-calculado o lucro da empresa. No final é apresentada uma mensagem de sucesso.

Cria Contrato É introduzido o código de cliente. Caso este exista no sistema (a verificação é feita pela classe GerirCliente) é carregado o objecto 'c' correspondente ao cliente em questão. O cliente não poderá possuir contrato e este é criado (objecto 'co'). É pedida a lista de ofertas disponíveis e o actor deverá escolher a oferta a adicionar. É

então adicionada ao contrato do cliente e é invocado o cálculo do pagamento mensal para regularizar a situação financeira do cliente. Também é re-calculado o lucro da empresa. Seguidamente o sistema apresenta os tipos de pagamento disponíveis e o escolhido é guardado no objecto 'c' (o cliente). O cliente é então notificado do pagamento que terá de efectuar. No final é apresentada uma mensagem de sucesso.

Remover Contrato É introduzido o código de cliente. Caso este exista no sistema (a verificação é feita pela classe GerirCliente) é carregado o objecto 'c' correspondente ao cliente em questão. A classe GerirContrato remove todas as ofertas do contrato do cliente (objecto co) e é invocado o cálculo do pagamento mensal para regularizar a situação financeira do cliente. Também é re-calculado o lucro da empresa. O contrato é removido do sistema e é enviada uma notificação de pagamento (que neste caso será uma devolução) ao cliente. No final é apresentada uma mensagem de sucesso.

Notifica Pagamento A Notificação de Pagamento tem como a criação de um objecto denominado de "CartaNotificação" que, além de gerar a notificação que será entregue ao cliente que contém a informação do valor que o cliente terá de pagar (ou receber no caso de o valor a ser extornado ser maior ao valor que terá de pagar), também servirá para mais tarde quando o cliente proceder ao pagamento dos serviços ser feita a criação da factura. Para a criação desta notificação, recorre-se ao CalculoValorFactura, que tem como objectivo fazer a soma dos valores dos serviços que terão de ser facturados. No caso de já ter sido feito uma facturação anterior, uma outra classe irá calcular a diferença do valor que foi facturado e o valor dos serviços que foram realmente prestados. Na sua parte final, este diagrama de sequência apresenta uma mensagem de informação ao funcionário no caso de ter de ser realizado um estorno, e guarda o objecto "CartaNotificação" que como foi referido será útil para a criação da factura. Por fim é impressa a notificação, e no caso de não ser possível é apresentada uma mensagem de erro.

Calcular Pagamento Mensal Este diagrama de sequência tem como função a criação de objectos do tipo "Pagamento", que servirão para o cálculo do valor a ser facturado. Para tal o sistema irá criar uma lista com os serviços aderidos pelo cliente de forma a somar o seu valor e para todos os meses desde o actual até ao mês final do contrato irá guardar a informação com o valor a ser cobrado. Este cálculo do pagamento mensal será processado sempre que houver uma alteração na lista de serviços aderidos pelo cliente, e a informação guardada é um excelente meio para se proceder posteriormente ao cálculo de creditações ou estornos.

4.3.5 Consulta Estatísticas

Ver lucro fornecedor As acções de estatística só poderão ser realizadas por um utilizador com privilégios para tal, neste caso, o administrador. "Ver Lucro Fornecedor" pretende fazer disponível o lucro total obtido através de um dado fornecedor. Sendo então a

primeira acção neste diagrama de sequência a de verificar se o fornecedor existe, caso exista então é pedido ao utilizador um intervalo de tempo, uma data de início e outra de fim. O intervalo de tempo é devidamente confirmado pelo sistema, para que não se verifique por exemplo o caso de a data de fim ser anterior à data de início. Depois da validação da data o sistema devolverá o lucro obtido por o fornecedor, naquele espaço de tempo, e o lucro é então apresentado ao utilizador.

Consultar Lucro Total Semelhante a "Ver Lucro Fornecedor" com a diferença que pretende calcular o lucro total proveniente de todos os fornecedores no sistema. Mais uma vez é pedido um intervalo de tempo e este é validado, sendo posteriormente pedido ao sistema o cálculo do total do lucro.

4.4 Diagrama de Classes

O diagrama de classes como modelo absoluto, produto da junção dos vários sub-diagramas de classes permite ter uma visão global do sistema muito importante na fase de implementação.

O desenvolvimento dos sub-diagramas de classe são parte de grande relevo no desenvolvimento dos diagramas de sequência. Os diagramas de sequência são inicialmente constituídos apenas por acções conceptuais, depois de concebidos, inicia-se a modelação das classes que serão constituídas pelos vários métodos, que por sua vez, serão a programação das acções dos diagramas conceptuais.

A interligação das classes torna-se clara durante o refinamento dos diagramas de sequência. Assim, no final, resultam vários sub-diagramas de classes que depois de agrupados geram um diagrama final de classes complexo, como se pode ver na figura [4.7](#).

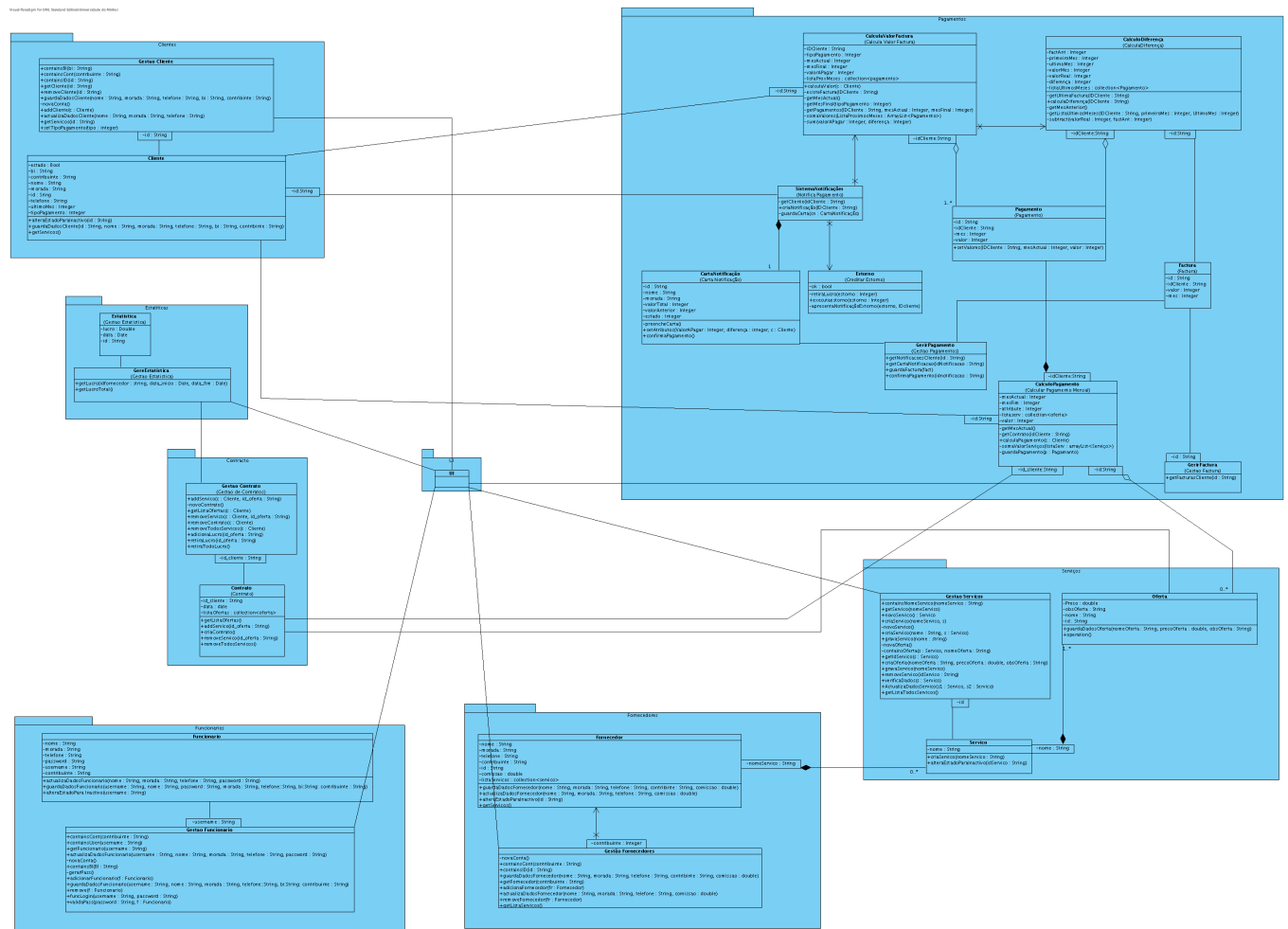


Figura 4.7: Diagrama de Classes

5 Conclusões e Trabalho Futuro

Este documento, em particular, tem como objectivo apresentar a fundamentação, análise e especificação para o projecto a realizar. Pretende-se aqui indicar e caracterizar de um ponto de vista mais geral a aplicação a ser desenvolvida, expondo as ideias e expectativas do cliente, bem como as soluções encontradas pela equipa.

Terminada a primeira fase de análise de requisitos e especificação permanecerá uma ideia semi-concreta da aplicação como um todo. Debatidas todas as funcionalidades e, após esclarecidas e contestadas todas as ambiguidades com o cliente pretende-se, irá se avançar para a próxima fase de especificação.

Os diagramas de *use case* foram usados como fundamento para a concepção de diagramas de sequência e classe, que englobam um nível de detalhe pormenorizado da interacção dos processos num espaço temporal, e a organização geral dos objectos assim como a relação entre eles. Os diagramas de sequência e classe por estarem a um nível bastante próximo da programação, antecederão a fase mais prática, de implementação da aplicação.

Bibliografia

G. Booch, J. Rumbaugh, I. Jacobson. *The Unified Modeling Language User Guide*, Addison-Wesley, 1998, ISBN 0-201-57168-4

Cockburn, Alistair. *Writing Effective Use Cases*, Addison-Wesley, 2000, ISBN 978-0201702255

Referências WWW

- 01 [**Condomínio fechado**](#)(visitado em Novembro de 2009)
Site de uma empresa de condomínios fechados
- 02 [**Homepage do Visual Paradigm**](#)(visitado em Novembro de 2009)
Site do programa de modelação utilizado neste projecto
- 03 [**Blog do Marco Mendes - Modelagem de Domínio**](#)(visitado em Novembro de 2009)
Artigo do “Blog do Marco Mendes” que explica o que é o modelo de domínio,
- 04 [**UML Use Case Diagrams: Tips and FAQ**](#)(visitado em Novembro de 2009)
Página que tem as perguntas mais frequentes sobre o UML. Excelente para tirar dúvidas.

A Anexos

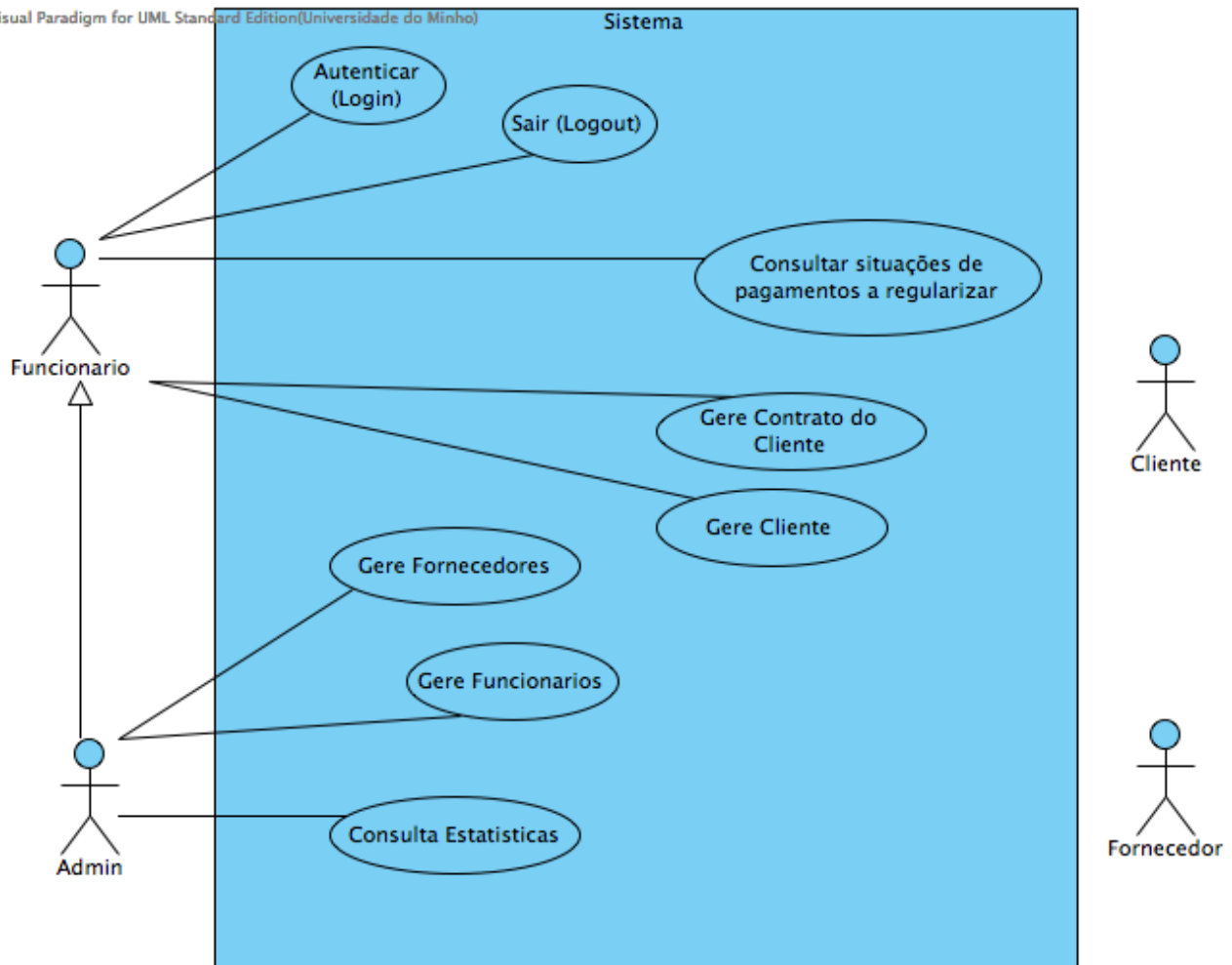
Table of Contents

Sistema	2
Gere Contrato do Cliente	5
Gere Fornecedores	16
Gere Funcionarios	23
Gere Dados Cliente	27
Consulta Estatísticas	31
Remover Contrato	34
Cancelar Serviço	35
Adicionar Novo Serviço2	36
Consulta Serviços Disponíveis - Main	37
Consulta serviços subscritos pelo Cliente - Main	38
Confirma Pagamento - Main	39
Emite segunda via de factura - Main	40
Consulta Serviços Disponíveis	41
Consulta serviços subscritos pelo Cliente	42
Confirma Pagamento	43
Emite segunda via de factura	44
Autenticar (Login)2	45
Cria Contrato2	46
Autenticar (Login)	47
Calcula Valor Factura	48
Calcula diferença factura anterior	49
Calcular Pagamento Mensal	50
Creditar Estorno	51
Notifica Pagamento	52
Adicionar Novo Serviço	53
Cria Contrato	54
Adicionar Cliente	55
Edita Cliente	56
Remove Cliente	57
Adiciona funcionario	58
Remove funcionario	59
Edita funcionario	60
Adiciona Fornecedor	61
Edita Fornecedor	62
Remove Fornecedor	63
Criar serviço	64
Remover Serviço	65
Editar Preço de serviço	66
Ver Lucro Fornecedor	67
Ver Lucro Fornecedor - Consultar lucro de fornecedor	68
Consultar Lucro Total - Consultar Lucro Total	69
Consultar Lucro Total	70

Use Case Diagram

Sistema

Visual Paradigm for UML Standard Edition(Universidade do Minho)



Use Case - Autenticar (Login)

Use Case Descriptions

Main	
Super Use Case	
Author	tiago
Date	Nov 21, 2009 5:54:30 PM
Brief Description	O funcionario pretende autenticar-se no sistema
Preconditions	Os dados do funcionario existem no sistema
Post-conditions	O funcionario esta autenticado

Flow of Events		Actor Input	System Response
	1	O funcionario introduz os dados no sistema	
	2		O sistema valida os dados do funcionario
	3		O sistema apresenta uma mensagem de sucesso
Alternative flow of events 2a: os dados estao errados		Actor Input	System Response
	1		O sistema apresenta uma mensagem de aviso
	2		voltar ao passo 1

Use Case - Gere Cliente

Use Case - Gere Contrato do Cliente

Use Case - Gere Fornecedores

Use Case - Gere Funcionarios

Use Case - Consulta Estatisticas

Use Case - Consultar situações de pagamentos a regularizar

Use Case Descriptions

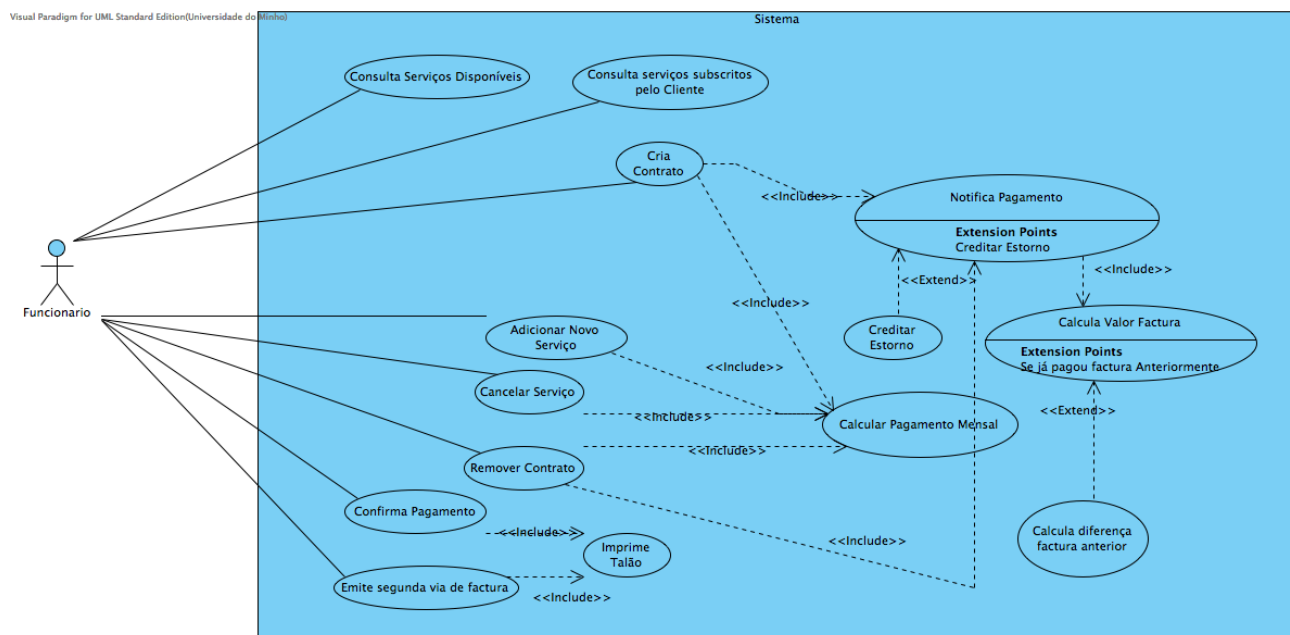
Main			
Super Use Case			
Author	tiago		
Date	Nov 30, 2009 11:19:12 PM		
Brief Description	O funcionario pretende verificar as situacoes de pagamentos a regularizar		
Preconditions	O funcionario esta autenticado		
Post-conditions			
Flow of Events		Actor Input	System Response
	1	O funcionario pede a lista das situacoes a regularizar	
	2		O sistema apresenta a lista

Use Case - Sair (Logout)

Use Case Descriptions

Sair			
Super Use Case			
Author	claudio, nuno		
Date	20/Nov/2009 17:10:00		
Brief Description	Sair da conta actual		
Preconditions	Funcionário autenticado		
Post-conditions	Utilizador faz logout		
Flow of Events		Actor Input	System Response
	1	Funcionário pede para sair do sistema	
	2		Sistema remove funcionário da sessão actual
	3		Sistema mensagem de sucesso

Gere Contrato do Cliente



Use Case - Confirma Pagamento

Use Case Descriptions

Main		
Author	Pedro Carvalho	
Date	Nov 22, 2009 9:15:56 PM	
Brief Description	Um funcionário pretende confirmar um pagamento de um cliente	
Preconditions	O funcionário está autenticado e o cliente existe	
Post-conditions	Pagamento é confirmado	
Flow of Events		

Alternative Flow of Events 2a Cliente não existe		Actor Input	System Response
	1		Sistema mostra mensagem de erro.
	2		Voltar ao passo 1.

Use Case - Emite segunda via de factura

Use Case Descriptions

Main			
Author	Pedro Carvalho		
Date	Nov 22, 2009 9:53:13 PM		
Brief Description	Emite a segunda via da factura para um cliente		
Preconditions	Cliente existe		
Post-conditions	A segunda via é emitida		
Flow of Events		Actor Input	System Response
	1	Funcionário insere o código do cliente.	
	2		Sistema valida código do cliente.
	3		Sistema devolve uma lista de facturas.
	4	Funcionário seleciona a factura em questão.	
	5	Funcionário seleciona emitir segunda via.	
	6		<<include>> Imprime Talão
Alternative Flow of Events 2a Cliente não existe		Actor Input	System Response
	1		Sistema mostra mensagem de erro.
	2		Voltar ao passo 1.
Exception 3a Não existem facturas		Actor Input	System Response
	1		Sistema mostra mensagem de erro.

Use Case - Adicionar Novo Serviço

Use Case Descriptions

Main	
Super Use Case	
Author	tiago
Date	Nov 21, 2009 6:08:48 PM
Brief Description	O funcionario pretende adicionar servicos ao cliente
Preconditions	O funcionario esta autenticado, o cliente existe no sistema, existem servicos no sistema
Post-conditions	E adicionado um novo servico ao cliente

Flow of Events		Actor Input	System Response
	1	O funcionario introduz o codigo do cliente	
	2		O sistema valida o codigo do cliente
	3		O sistema apresenta os servicos disponiveis
	4	O funcionario escolhe o servico	
	5		O servico e adicionado ao contrato do cliente
	6		<include> Calcular Pagamento
	7		Os dados sao actualizados no sistema
Alternative flow of events 2a: codigo de cliente errado		Actor Input	System Response
	1		O sistema apresenta uma mensagem de aviso
Exception: 3 nao existem servicos disponiveis		Actor Input	System Response
	1		O sistema apresenta uma mensagem de erro

Use Case - Calcular Pagamento Mensal

Use Case Descriptions

Calcular Pagamento	
Super Use Case	
Author	henrique
Date	Nov 22, 2009 10:47:47 PM
Brief Description	Calcula o valor que o cliente terá que pagar
Preconditions	Houve uma mudança nos serviços aderidos do cliente
Post-conditions	Valor do montante mensal a pagar actualizado

Flow of Events		Actor Input	System Response
	1	O sistema de calculo de pagamento pede o mes corrente	
	2		O sistema diz qual é o mes corrente
	3	O sistema de calculo de pagamento pede qual o ultimo mes do contrato	
	4		O sistema diz qual é o ultimo mes correspondene ao contrato
	5	O sistema de calculo de pagamento pede a lista de serviços e respectivo valor que o cliente tem contratado	
	6		o sistema oferece a lista de serviços a que o cliente aderiu
	7	sistema de calculo de pagamento soma o valor dos serviços	
	8	sistema de calculo de pagamento guarda o valor que será cobrado ao cliente para todos os meses desde o presente até ao final do contrato	
	9		O sistema guarda os dados

Use Case - Cria Contrato

Use Case Descriptions

Main			
Super Use Case			
Author	tiago		
Date	Nov 21, 2009 6:08:16 PM		
Brief Description	O funcionario pretende criar um novo contrato para o cliente		
Preconditions	O funcionario esta autenticado e o cliente existe		
Post-conditions	O contrato e criado		
Flow of Events		Actor Input	System Response
	1	O funcionario introduz o codigo do cliente	
	2		O sistema valida o codigo do cliente
	3		O sistema verifica se o cliente ja tem contrato
	4		O sistema apresenta os servicos disponiveis
	5	O funcionario escolhe o servico	
	6		O servico e adicionado ao contrato do cliente
	7		<include> Calcular Pagamento

	8		O sistema apresenta os tipos de pagamentos
	9	O funcionario escolhe o tipo de pagamento pretendido	
	10		<include> Notificação Pagamento
	11		Os dados sao actualizados no sistema
Alternative flow of events 2a: codigo de cliente errado		Actor Input	System Response
	1		O sistema apresenta uma mensagem de aviso
	2		voltar ao passo 1
Exception: 3 o cliente ja tem contrato		Actor Input	System Response
	1		O sistema apresenta uma mensagem de erro
Exception: 4 nao existem servicos disponiveis		Actor Input	System Response
	1		O sistema apresenta uma mensagem de erro

Use Case - Imprime Talão

Use Case Descriptions

Main			
Super Use Case			
Author	Pedro Carvalho		
Date	Nov 22, 2009 10:34:48 PM		
Brief Description	O sistema Imprime um Talão.		
Preconditions			
Post-conditions	Sucesso: O sistema imprime o talão. Insucesso: O sistema não imprime o talão.		
Flow of Events		Actor Input	System Response
	1		Sistema Imprime Talão.

Use Case - Creditar Estorno

Use Case Descriptions

Creditar Extorno			
Super Use Case			
Author	henrique		
Date	Nov 22, 2009 10:25:21 PM		
Brief Description	Creditação ao cliente de um valor resultante do pagamento de serviços posteriormente cancelados		
Preconditions	A notificação de pagamento apresenta um valor negativo		
Post-conditions	É devolvido o montante ao cliente		
Flow of Events		Actor Input	System Response
	1	O sistema de notificacao apresenta o valor a ser extornado	
	2		O sistema retira esse valor ao lucro da empresa
	3		O sistema apresenta uma notificacao ao funcionario

Use Case - Notifica Pagamento

Use Case Descriptions

Notifica pagamento			
Super Use Case			
Author	henrique		
Date	Nov 23, 2009 5:06:13 PM		
Brief Description	Cria uma notificação com o valor e data de pagamento para o cliente		
Preconditions			
Post-conditions	É impressa a notificação de pagamento para ser enviada ao Cliente		
Flow of Events		Actor Input	System Response
	1		<include> Calcula Valor Fatura
	2	O sistema de notificações pede os dados do cliente	
	3		O Sistema dá os dados do Cliente
	4	O sistema de notificações cria a carta de notificação de pagamento	
	5	O sistema pede para imprimir a carta de notificação	
	6		O sistema imprime a carta de notificação

Exception: 6 Impossibilidade de imprimir		Actor Input	System Response
	1		O sistema guarda a notificação a ser impressa
	2		O sistema avisa o funcionário do erro ocorrido
	3		
Alternative Flow of events 4a - Extension Point: pagamento negativo		Actor Input	System Response
	1		Extended by: Creditar extorno
	2		voltar o passo 5

Use Case - Calcula Valor Factura

Use Case Descriptions

Calcula Valor Factura			
Super Use Case			
Author	henrique		
Date	Nov 23, 2009 3:52:00 PM		
Brief Description	Calcula o valor que o cliente terá que pagar		
Preconditions			
Post-conditions	É enviada uma notificação de pagamento ao cliente		
Flow of Events		Actor Input	System Response
	1		O Sistema verifica se é a primeira notificação a ser criada
	2	o sistema de notificações pede o valor dos meses correspondentes à proxima factura	
	3		O sistema dá a lista dos valores dos meses da proxima factura
	4	o sistema de notificações faz a soma do valor dos meses dados	
	5	o sistema de notificações faz a soma de todos os valores a cobrar	
Alternative Flow of Events 1a - Extension Point: Cliente já pagou uma factura anteriormente		Actor Input	System Response
	1		Extended by: Calcula diferença factura anterior
	2		Voltar ao passo 2

Use Case - Calcula diferença factura anterior

Use Case Descriptions

Calcula Diferença Factura Anterior

Super Use Case			
Author	henrique		
Date	Nov 23, 2009 4:29:14 PM		
Brief Description	Calcula o valor dos ultimos meses que não foi cobrado devido a mudanças no plano de serviços		
Preconditions	Cliente ja pagou pelo menos uma factura anterior		
Post-conditions	O sistema de notificações recebe o valor para proceder à emissão de uma nova factura		
Flow of Events		Actor Input	System Response
	1	O sistema de notificações pede o valor da ultima factura	
	2		Sistema dá o valor da ultima factura
	3	O sistema de notificações pede o valor dos meses correspondentes à ultima factura	
	4		Sistema dá o valor dos meses correspondentes à ultima factura
	5	O sistema de notificações soma o valor dos ultimos meses	
	6	O sistema de notificações subtrai o valor da ultima factura à soma do valor dos meses correspondente a essa factura	

Use Case - Remover Contrato

Use Case Descriptions

Main	
Super Use Case	
Author	tiago
Date	Nov 25, 2009 4:20:04 PM
Brief Description	Cancela o contrato com o cliente
Preconditions	O funcionario esta autenticado, o contrato existe
Post-conditions	O contrato e cancelado

Flow of Events		Actor Input	System Response
	1	O funcionario introduz o codigo do cliente	
	2		O sistema valida o codigo
	3		Os servicos sao todos removidos do contrato
	4		<include> Calcular Pagamento Mensal
	5		O sistema cancela o contrato
	6		<include> Notifica Pagamento
	7		O sistema apresenta uma mensagem de sucesso
Alternative flow of events 2a o codigo esta errado		Actor Input	System Response
	1		O sistema apresenta uma mensagem de aviso
	2		voltar ao passo 1

Use Case - Cancelar Serviço

Use Case Descriptions

Main			
Super Use Case			
Author	tiago		
Date	Nov 25, 2009 4:10:43 PM		
Brief Description	Cancela um serviço anteriormente contratado		
Preconditions	O funcionario esta autenticado, o cliente tem pelo menos um servico contratado		
Post-conditions	O servico foi removido do contrato		
Flow of Events		Actor Input	System Response
	1	O funcionário introduz o código do cliente	
	2		O sistema valida o codigo do cliente
	3		O sistema apresenta a lista de servicos contratados
	4	O funcionario escolhe o servico	
	5		O servico e removido do contrato
	6		<include> Calcular Pagamento Mensal
	7		O sistema apresenta mensagem de sucesso

Alternative flow of events 2a o código do cliente está errado		Actor Input	System Response
	1		O sistema apresenta uma mensagem de aviso
	2		voltar ao passo 1

Use Case - Consulta serviços subscritos pelo Cliente

Use Case Descriptions

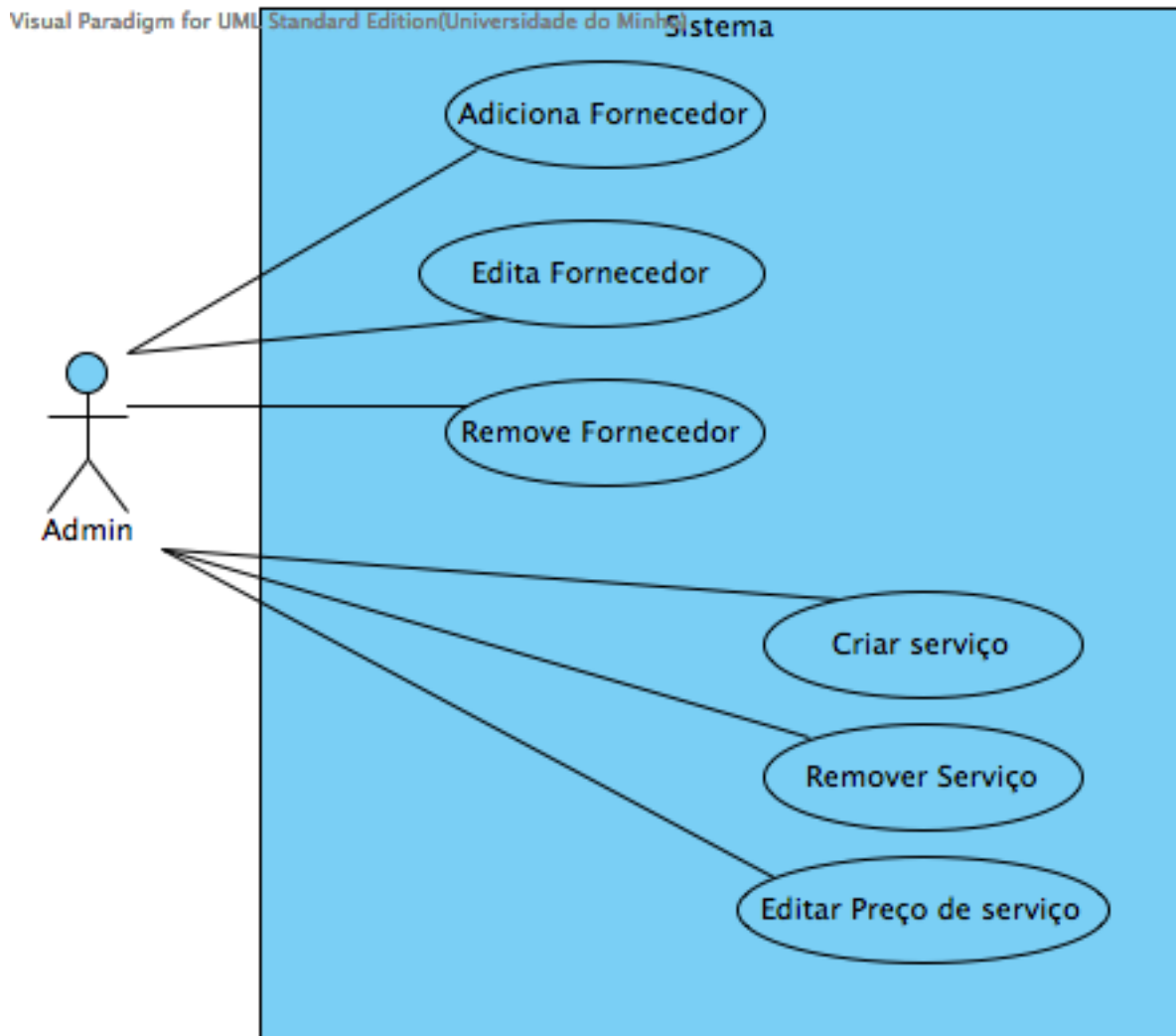
Main			
Author	Pedro Carvalho		
Date	Nov 22, 2009 10:45:16 PM		
Brief Description	Consultar lista de serviços que um cliente adquiriu		
Preconditions	Funcionário autenticado e cliente existe.		
Post-conditions	Sistema mostra lista de serviços do cliente.		
Flow of Events		Actor Input	System Response
	1	Funcionário introduz código do cliente.	
	2		Sistema confirma código do cliente.
	3		Sistema devolve lista de serviços do cliente.
Alternate Flow of Events 2a Cliente não existe		Actor Input	System Response
	1		Sistema mostra uma mensagem de erro.
	2		Voltar ao passo 1.
Alternate Flow of Events 3a Cliente não tem serviços		Actor Input	System Response
	1		Sistema mostra mensagem no lugar da lista.

Use Case - Consulta Serviços Disponíveis

Use Case Descriptions

Main			
Author	Pedro Carvalho		
Date	Nov 22, 2009 10:40:19 PM		
Brief Description	Consultar Lista de serviços de empresas contidas no sistema.		
Preconditions	Funcionário autenticado		
Post-conditions	Sistema mostra lista		
Flow of Events		Actor Input	System Response
	1	Funcionário requisita a lista de serviços no sistema.	
	2		Sistema mostra lista de serviços.
Alternate Flow of Events 2a Lista de Serviços Vazia		Actor Input	System Response
	1		Sistema mostra mensagem no lugar da lista.

Gere Fornecedores



Use Case - Adiciona Fornecedor

Use Case Descriptions

Adicionar Fornecedor	
Super Use Case	
Author	claudio, nuno
Date	20/Nov/2009 14:42:55
Brief Description	Criar uma nova conta de fornecedor
Preconditions	Administrador autenticado
Post-conditions	Sucesso: Nova conta de fornecedor no sistema Insucesso: Sistema não é alterado

Normal Flow of Events		Actor Input	System Response
	1	Administrador insere dados do fornecedor	
	2		O Sistema verifica repetição de dados (contribuinte)
	3		O Sistema cria conta de novo fornecedor
	4		O Sistema envia mensagem de sucesso
Alternativa Flow of Events 2a- dados inseridos ja constam no sistema		Actor Input	System Response
	1		
	2		O Sistema avisa repetição de dados
	3		Volta ao passo 1

Use Case - Criar serviço

Use Case Descriptions

Criar Serviço			
Super Use Case			
Author	claudio nuno		
Date	20/Nov/2009 18:21:00		
Brief Description	Permite criar serviço de um fornecedor		
Preconditions	Administrador autenticado e fornecedor existente		
Post-conditions	Sucesso: Serviço criado e associado a um fornecedor Insucesso: Sistema nao se altera		
Normal Flow of Events		Actor Input	System Response
	1	Insere codigo de fornecedor	
	2		Sistema valida codigo do fornecedor
	3	Insere dados do serviço	
	4		Sistema valida dados
	5		Sistema cria serviço
	6		Sistema actualiza ficha de fornecedor
	7		Sistema envia mensagem de sucesso
Alternativa Flow of Events 2a - Codigo invalido		Actor Input	System Response
	1		
	2		Sistema envia mensagem de codigo invalido
	3		Volta ao passo 1

Alternativa Flow of Events 4a - Dados Invalidos		Actor Input	System Response
	1		
	2		
	3		
	4		Sistema envia mensagem de dados invalidos
	5		Volta ao passo 3

Use Case - Remover Serviço

Use Case Descriptions

Remove serviço			
Super Use Case			
Author	claudio, nuno		
Date	20/Nov/2009 23:08:40		
Brief Description	Remove um serviço prestado por um fornecedor		
Preconditions	Administrador autenticado		
Post-conditions	Sucesso: Serviço "removido" do sistema Insucesso: Sistema não se altera		
Normal Flow of Events		Actor Input	System Response
	1	Inserir código Fornecedor	
	2		Sistema valida código
	3		Sistema apresenta lista de serviços disponibilizados pelo fornecedor
	4	Administrador selecciona serviço	
	5		Sistema pede confirmação para a "remoção"
	6	Administrador confirma remoção	
	7		Sistema altera estado do serviço para "inactivo"
	8		Sistema apresenta mensagem de sucesso
Alternativa Flow of Events 2a - Código inválido		Actor Input	System Response
	1		
	2		Sistema envia mensagem de código inválido
	3		Volta ao passo 1

Alternativa Flow of Events 6a- Administrador não confirma		Actor Input	System Response
	1		
	2		
	3		
	4		
	5		
	6		
	7		Volta ao passo 1

Use Case - Editar Preço de serviço

Use Case Descriptions

Editar Preço de serviço			
Super Use Case			
Author	claudio nuno		
Date	20/Nov/2009 23:09:21		
Brief Description	Permite editar a tabela de preços de um serviço de um fornecedor		
Preconditions	Administrador autenticado		
Post-conditions	Sucesso:Tabela de Preços do Serviço editado Insucesso: Sistema nao se altera		
Normal Flow of Events		Actor Input	System Response
	1	Administrador insere código do fornecedor	
	2		Sistema valida código do fornecedor
	3		Sistema apresenta lista de serviços
	4	Administrador selecciona serviço a editar	
	5		Sistema apresenta tabela de preços em vigor
	6	Administrador edita preços da tabela	
	7		Sistema valida dados
	8		Sistema pede confirmação
	9	Administrador confirma	
	10		Sistema actualiza tabela de preços do serviço
	11		Sistema apresenta mensagem de sucesso

Alternativa Flow of Events 2a- Código do fornecedor inválido		Actor Input	System Response
	1		
	2		Sistema envia mensagem de código inválido
	3		Volta ao passo 1
Alternativa Flow of Events 7a- Preços inválidos		Actor Input	System Response
	1		
	2		
	3		
	4		
	5		
	6		
	7		Sistema envia mensagem de preços inválidos
	8		Volta ao passo 6
Alternativa Flow of events 10a- Administrador não confirma edição do preço		Actor Input	System Response
	1		
	2		
	3		
	4		
	5		
	6		
	7		
	8		
	9		
	10		Sistema volta ao passo 1

Use Case - Remove Fornecedor

Use Case Descriptions

Remover Fornecedor	
Super Use Case	
Author	claudio, nuno
Date	20/Nov/2009 15:31:41
Brief Description	Remove conta de fornecedor
Preconditions	Administrador autenticado
Post-conditions	Sucesso: O sistema modifica a conta para inactiva Insucesso: Sistema não é alterado

Normal Flow of Events		Actor Input	System Response
	1	Administrador insere código de fornecedor	
	2		Sistema valida código de fornecedor
	3		Sistema apresenta dados do fornecedor e solicita confirmação
	4	Administrador confirma remoção	
	5		Sistema modifica propriedade de conta para inactiva
	6		Sistema envia mensagem de "remoção"
Alternativa Flow of Events 2a - Codigo invalido		Actor Input	System Response
	1		
	2		Sistema avisa codigo invalido
	3		Volta ao passo 1
Alternativa Flow of Events 5a - Administrador nao confirmou remoção		Actor Input	System Response
	1		
	2		
	3		
	4		
	5		Volta ao passo 1

Use Case - Edita Fornecedor

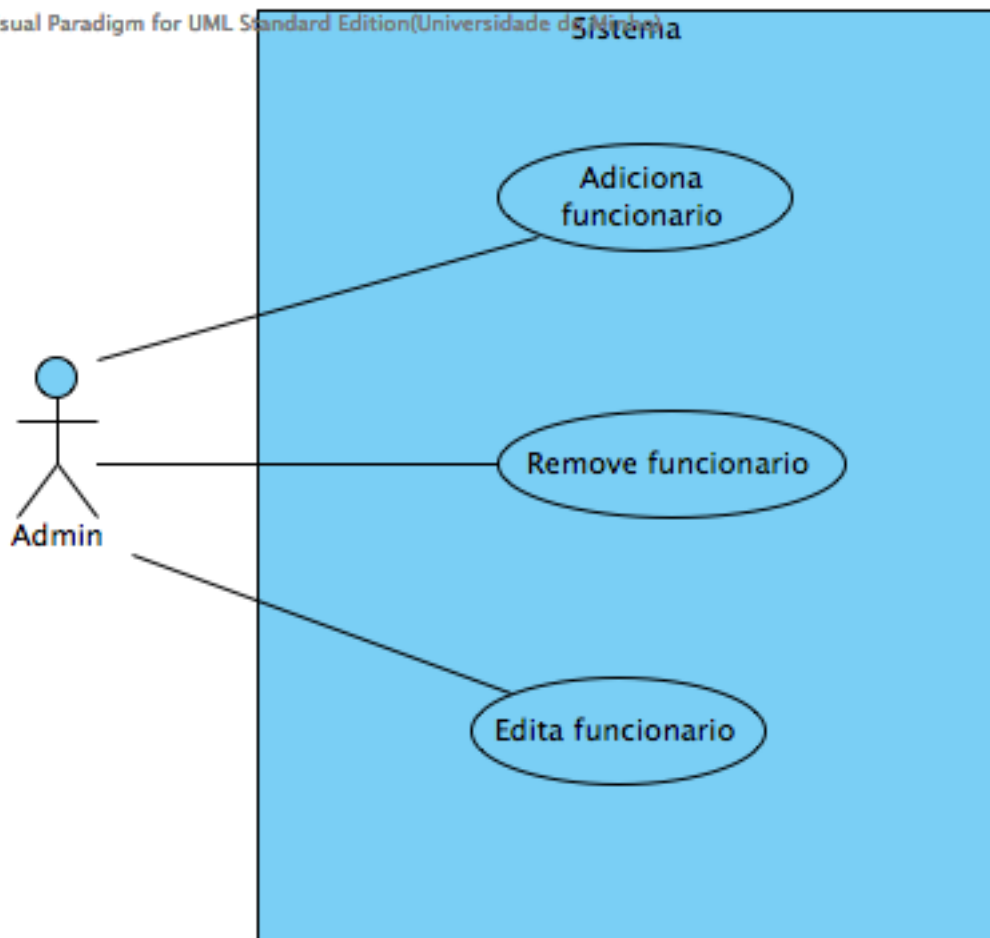
Use Case Descriptions

Editar Dados do fornecedor	
Super Use Case	
Author	claudio, nuno
Date	20/Nov/2009 16:12:13
Brief Description	Editar dados da conta de um fornecedor
Preconditions	Administrador autenticado
Post-conditions	Sucesso: O sistema modifca dados da conta do fornecedor Insucesso: Sistema não é alterado

Normal Flow of Events		Actor Input	System Response
	1	Administrador insere código do fornecedor	
	2		Sistema valida código do fornecedor
	3		Sistema apresenta os dados do fornecedor
	4	Administrador altera dados	
	5		Sistema pede confirmação
	6	Administrador confirma alteração de dados	
	7		Sistema actualiza conta
	8		Sistema envia mensagem de confirmação
Alternativa Flow of Events 2a - código inválido		Actor Input	System Response
	1		
	2		Sistema avisa código inválido
	3		Volta ao passo 1
Alternativa Flow of Events 8a - Administrador não confirmou alteração		Actor Input	System Response
	1		
	2		
	3		
	4		
	5		
	6		
	7		
	8		Volta ao passo 3

Gere Funcionarios

Visual Paradigm for UML Standard Edition(Universidade de Minho)



Use Case - Adiciona funcionario

Use Case Descriptions

Adicionar funcionário	
Super Use Case	
Author	claudio, nuno
Date	20/Nov/2009 14:33:25
Brief Description	Criar uma nova conta de funcionário
Preconditions	Administrador autenticado
Post-conditions	Sucesso: Nova conta de funcionário no sistema Insucesso: Sistema não é alterado

Normal Flow of Events		Actor Input	System Response
	1	Administrador insere dados do funcionário	
	2		O Sistema verifica repetição de dados (nome de utilizador, BI e contribuinte)
	3		O Sistema cria password aleatoria
	4		O Sistema cria conta de novo funcionário
	5		O Sistema envia mensagem de sucesso com nome de utilizador e password
Alternativa Flow of Events 2a- dados inseridos ja constam no sistema		Actor Input	System Response
	1		
	2		O Sistema avisa repetição de dados
	3		Volta ao passo 1

Use Case - Remove funcionario

Use Case Descriptions

Remover funcionário		
Super Use Case		
Author	claudio, nuno	
Date	20/Nov/2009 15:24:48	
Brief Description	Remove conta de funcionário	
Preconditions	Administrador autenticado	
Post-conditions	Sucesso: O sistema modifica a conta para inactiva Insucesso: Sistema não é alterado	
Normal Flow of Events		Actor Input
	1	Administrador insere username de funcionário
	2	
	3	
	4	Administrador confirma remoção
	5	
	6	
		System Response
		Sistema valida username de funcionário
		Sistema apresenta dados do funcionário e solicita confirmação
		Sistema modifica propriedade de conta para inactiva
		Sistema envia mensagem de "remoção"

Alternativa Flow of Events 2a - Username invalido		Actor Input	System Response
	1		
	2		Sistema avisa username invalido
	3		Volta ao passo 1
Alternativa Flow of Events 5a - Administrador nao confirmou remoção		Actor Input	System Response
	1		
	2		
	3		
	4		
	5		Volta ao passo 1

Use Case - Edita funcionario

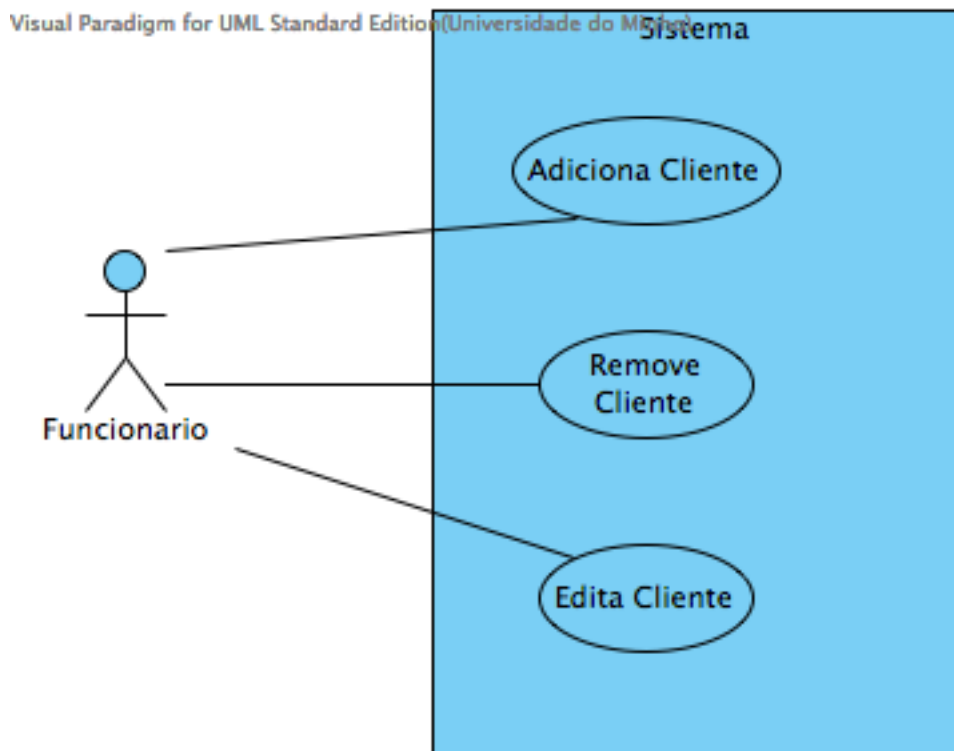
Use Case Descriptions

Editar dados do funcionário			
Super Use Case			
Author	claudio, nuno		
Date	20/Nov/2009 15:52:59		
Brief Description	Editar dados da conta de um funcionario		
Preconditions	Administrador autenticado		
Post-conditions	Sucesso: O sistema modifca dados da conta do funcionário Insucesso: Sistema não é alterado		
Normal Flow of Events		Actor Input	System Response
	1	Administrador insere código do funcionário	
	2		Sistema valida código do funcionário
	3		Sistema apresenta os dados do funcionário
	4	Administrador altera dados	
	5		Sistema valida dados (passwords,nome de utilizador, BI e contribuinte)
	6		Sistema pede confirmação
	7	Administrador confirma alteração de dados	
	8		Sistema actualiza conta
	9		Sistema envia mensagem de confirmação

Alternativa Flow of Events 2a - código inválido		Actor Input	System Response
	1		
	2		Sistema avisa código inválido
	3		Volta ao passo 1
Alternativa Flow of Events 5a - Dados inválidos		Actor Input	System Response
	1		
	2		
	3		
	4		
	5		Sistema avisa que há dados inválidos
	6		Volta ao passo 3
Alternativa Flow of Events 8a - Administrador não confirmou alteração		Actor Input	System Response
	1		
	2		
	3		
	4		
	5		
	6		
	7		
	8		Volta ao passo 3

Use Case Diagram

Gere Dados Cliente



Use Case - Adiciona Cliente

Use Case Descriptions

Adicionar Cliente			
Super Use Case			
Author	Claudio, Nuno		
Date	20/Nov/2009 14:13:19		
Brief Description	Criar uma nova conta de cliente		
Preconditions	Funcionário autenticado		
Post-conditions	Sucesso: Nova conta de cliente no sistema Insucesso: Sistema não é alterado		
Normal Flow of Events		Actor Input	System Response
	1	Funcionário insere dados do cliente	
	2		O Sistema verifica repetição de dados (BI e contribuinte)
	3		O Sistema cria conta de novo cliente
	4		O Sistema envia mensagem de sucesso com o código de cliente atribuído

Alternativa Flow of Events 2a- dados inseridos ja constam no sistema		Actor Input	System Response
	1		
	2		O Sistema avisa repetição de dados
	3		Volta ao passo 1

Use Case - Remove Cliente

Use Case Descriptions

Remover Cliente			
Super Use Case			
Author	claudio, nuno		
Date	20/Nov/2009 14:54:21		
Brief Description	Remove conta de cliente		
Preconditions	Funcionário autenticado		
Post-conditions	Sucesso: O sistema modifica a conta para inactiva Insucesso: Sistema não é alterado		
Normal Flow of Events		Actor Input	System Response
	1	Funcionário insere código de cliente	
	2		Sistema valida código de cliente
	3		Sistema apresenta dados do cliente e solicita confirmação
	4	Funcionário confirma remoção	
	5		Sistema modifica propriedade de conta para inactiva
	6		Sistema envia mensagem de "remoção"
Alternativa Flow of Events 2a - Codigo invalido		Actor Input	System Response
	1		
	2		Sistema avisa codigo invalido
	3		Volta ao passo 1
Alternativa Flow of Events 5a - Funcionario nao confirmou remoção		Actor Input	System Response
	1		
	2		
	3		
	4		
	5		Volta ao passo 1

Use Case - Edita Cliente

Use Case Descriptions

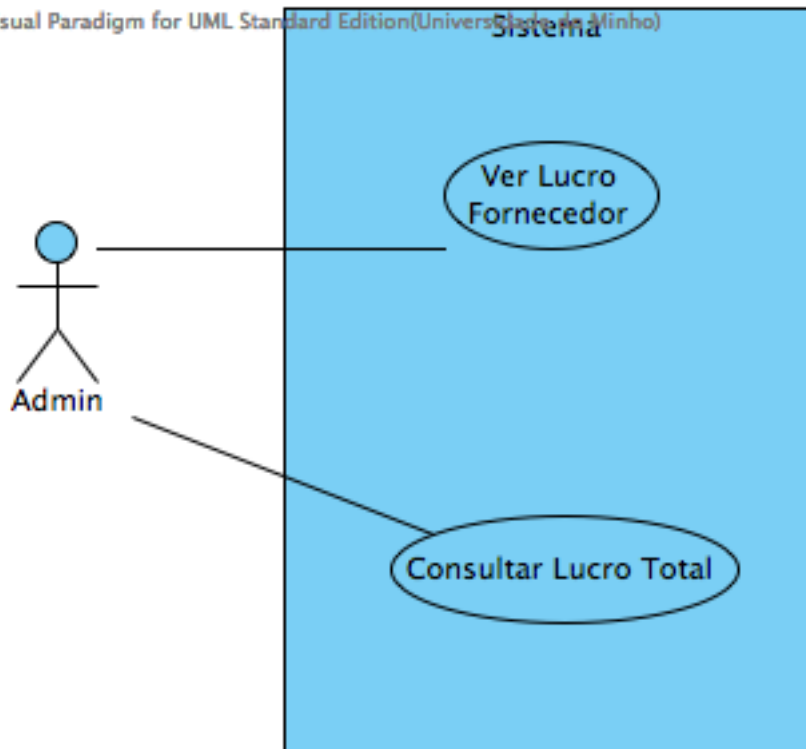
Editar Cliente			
Super Use Case			
Author	claudio, nuno		
Date	20/Nov/2009 16:15:04		
Brief Description	Editar dados da conta de um funcionario		
Preconditions	Administrador autenticado		
Post-conditions	Sucesso: O sistema modifca dados da conta do funcionário Insucesso: Sistema não é alterado		
Normal Flow of Events		Actor Input	System Response
	1	Administrador insere código do Cliente	
	2		Sistema valida código do cliente
	3		Sistema apresenta os dados do cliente
	4	Administrador altera dados	
	5		Sistema valida dados
	6		Sistema pede confirmação
	7	Administrador confirma alteração de dados	
	8		Sistema actualiza conta
	9		Sistema envia mensagem de confirmação
Alternativa Flow of Events 2a - codigo invalido		Actor Input	System Response
	1		
	2		Sistema avisa codigo invalido
	3		Volta ao passo 1
Alternativa Flow of Events 5a - Dados invalidos		Actor Input	System Response
	1		
	2		
	3		
	4		
	5		Sistema avisa que há dados invalidos
	6		Volta ao passo 3

Alternativa Flow of Events 8a - Funcionario nao confirmou alteração		Actor Input	System Response
	1		
	2		
	3		
	4		
	5		
	6		
	7		
	8		Volta ao passo 3

Use Case Diagram

Consulta Estatísticas

Visual Paradigm for UML Standard Edition(University of Minho)



Use Case - Ver Lucro Fornecedor

Use Case Descriptions

Consultar lucro de fornecedor	
Super Use Case	
Author	claudio, nuno
Date	20/Nov/2009 16:44:32
Brief Description	Permite saber qual o lucro que determinado fornecedor está a dar à GereComSaber, num determinado intervalo de tempo
Preconditions	Administrador autenticado
Post-conditions	true

Normal Flow of Events		Actor Input	System Response
	1	Administrador insere código do fornecedor	
	2		Sistema valida código do fornecedor
	3	Administrador escolhe o intervalo de tempo	
	4		Sistema valida intervalo de tempo
	5		Sistema calcula lucro do respectivo fornecedor
	6		Sistema apresenta resultado
Alternativa Flow of Events 2a - código inválido		Actor Input	System Response
	1		
	2		Sistema avisa código inválido
	3		Volta ao passo 1
Alternativa Flow of Events 4a - o intervalo de tempo inválido		Actor Input	System Response
	1		Sistema avisa intervalo de tempo inválido
	2		Volta ao passo 3

Use Case - Consultar Lucro Total

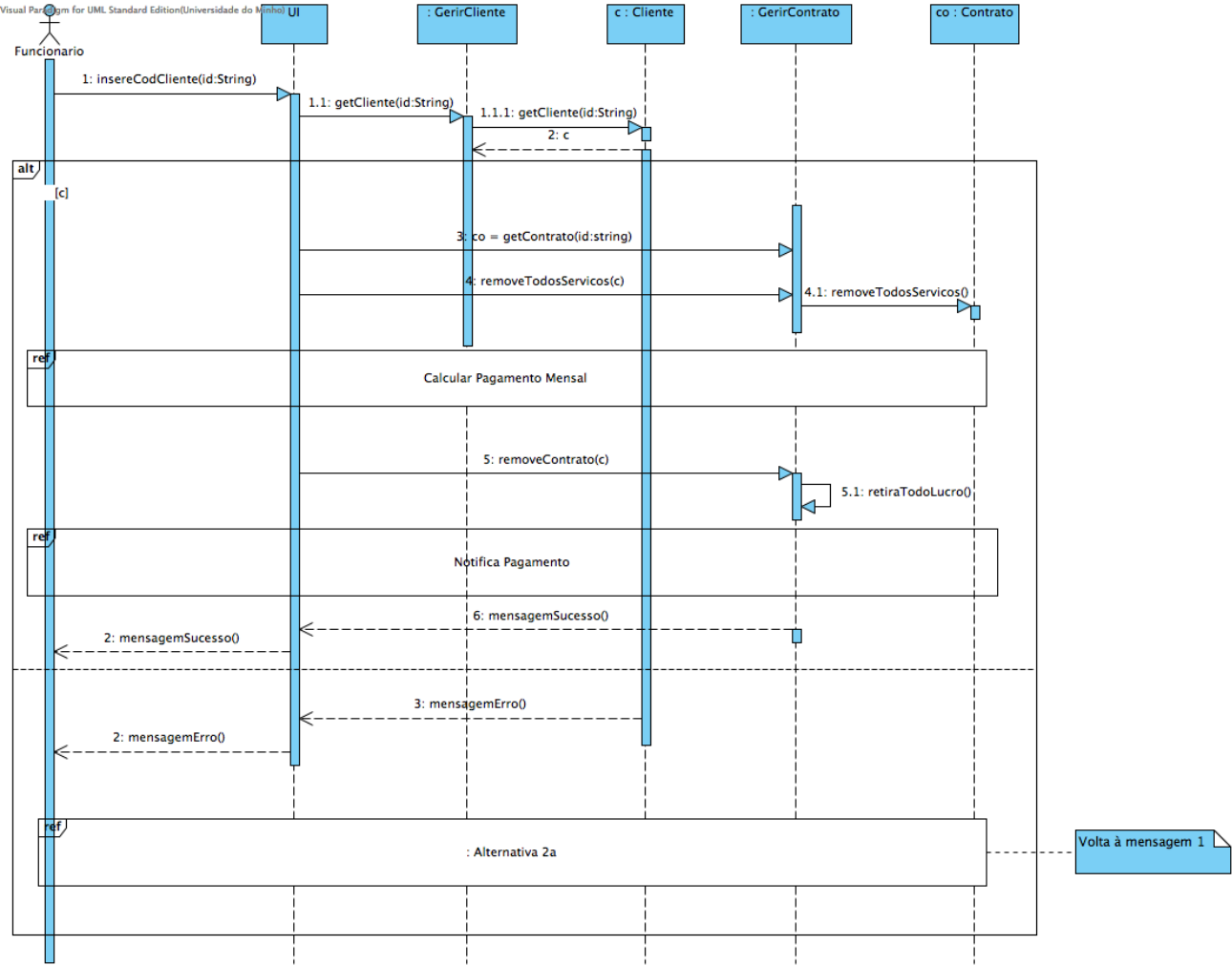
Use Case Descriptions

Consultar Lucro Total			
Super Use Case			
Author	claudio, nuno		
Date	20/Nov/2009 16:41:02		
Brief Description	Permite consultar o lucro que a GereComSaber num determinado intervalo de tempo		
Preconditions	Administrador autenticado		
Post-conditions	true		
Normal Flow of Events		Actor Input	System Response
	1	Administrador escolhe o intervalo de tempo	
	2		Sistema valida o intervalo de tempo
	3	Administrador pede cálculo do lucro total	
	4		Sistema Calcula valor
	5		Sistema apresenta resultado

Alternative Flow of Events 2a - intervalo de tempo invalido		Actor Input	System Response
	1		Sistema avisa intervalo de tempo invalido
	2		Volta ao passo 1

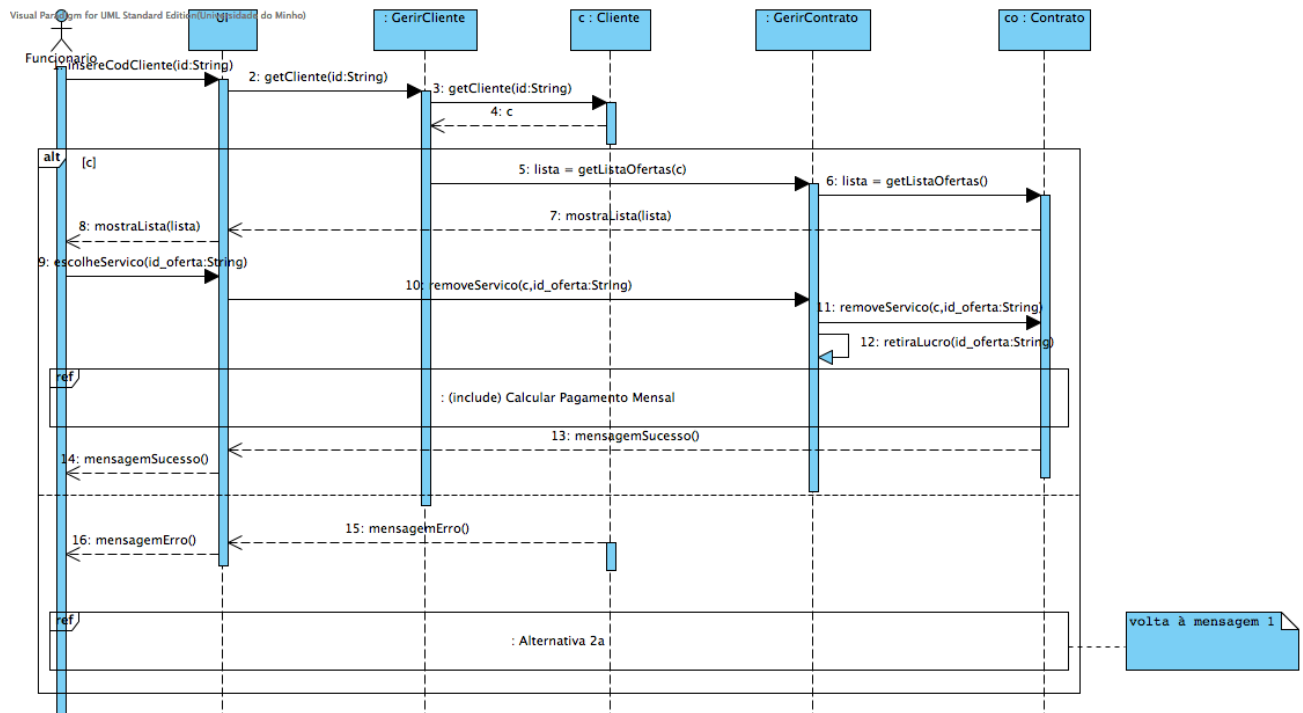
Sequence Diagram

Remover Contrato



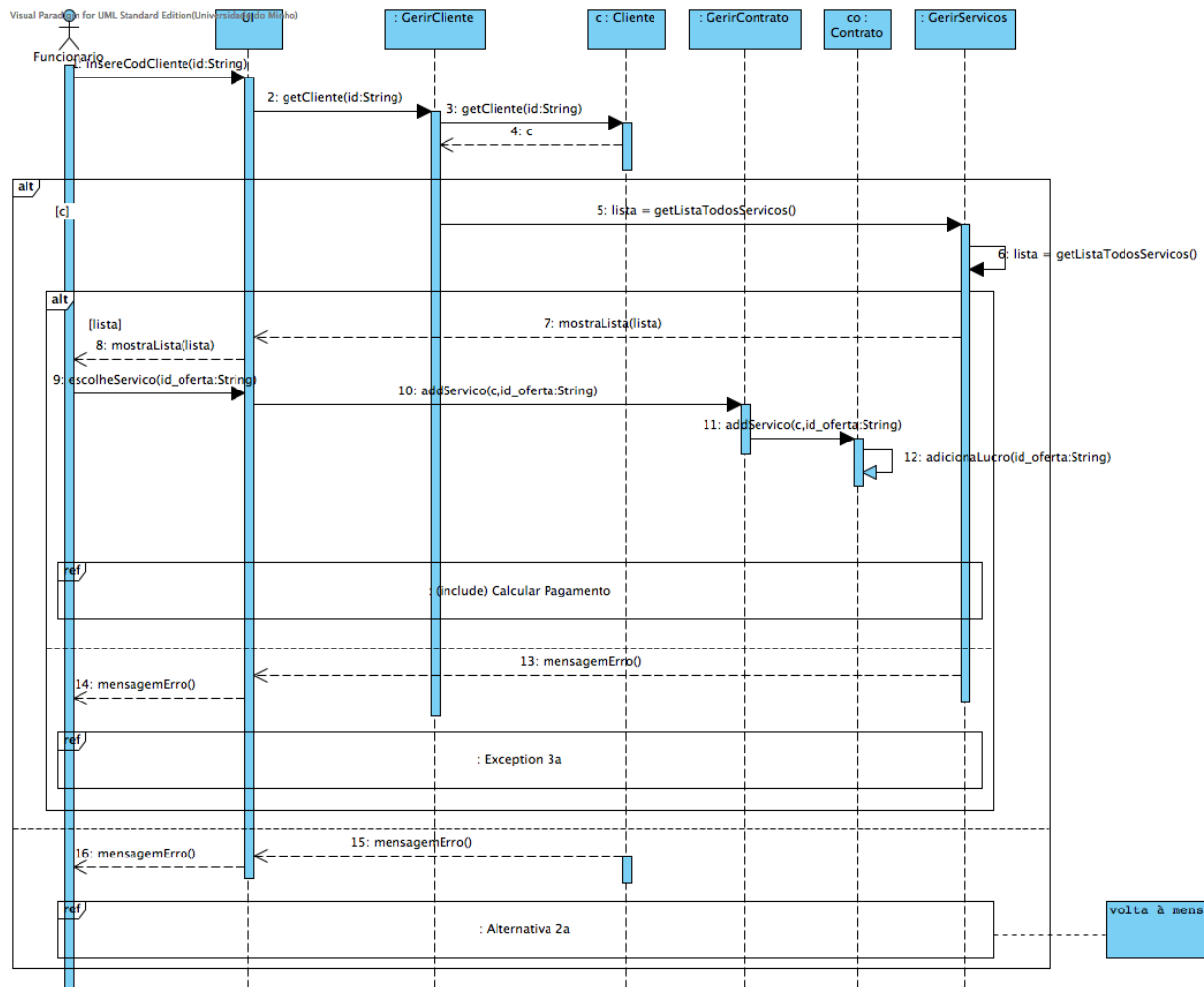
Sequence Diagram

Cancelar Serviço



Sequence Diagram

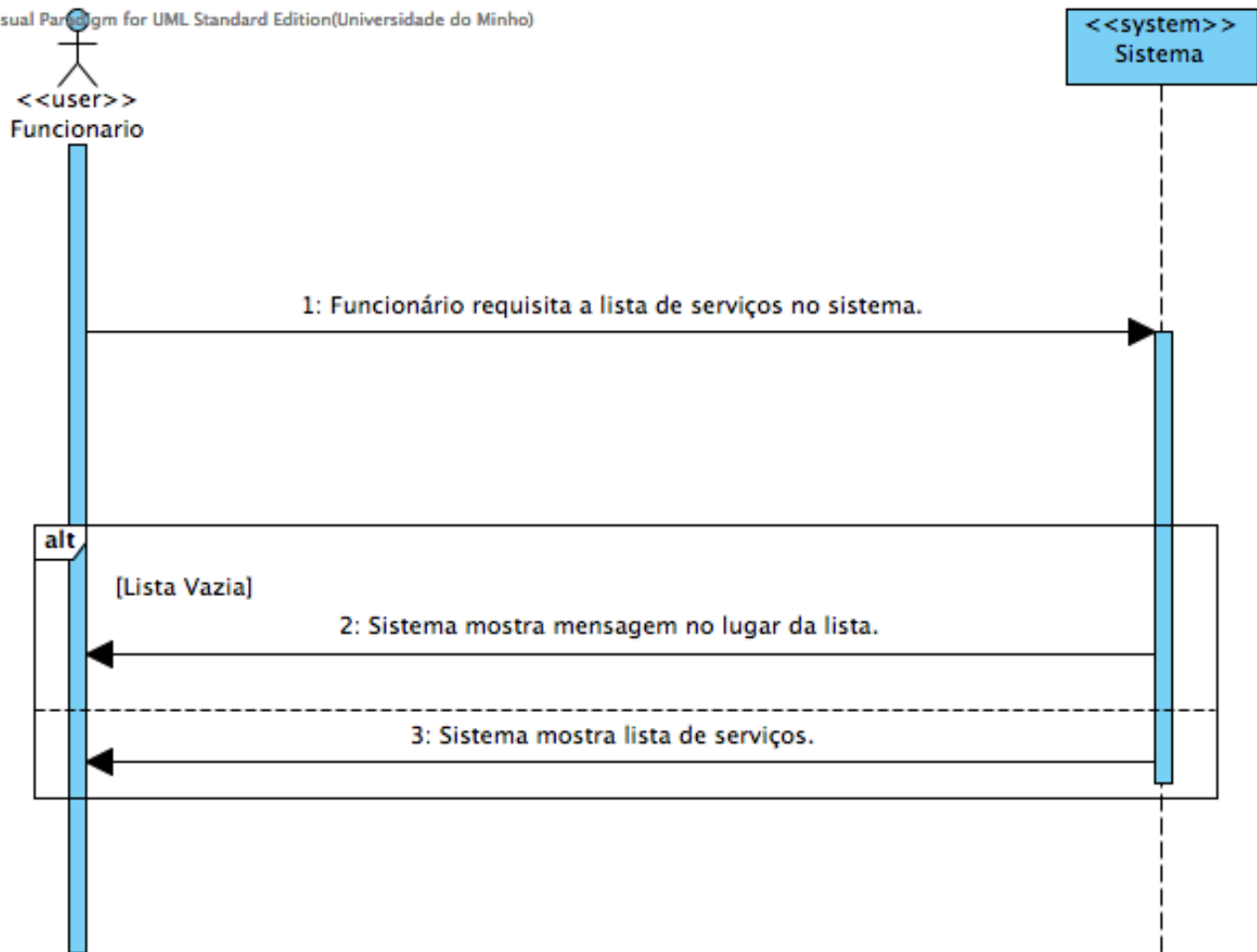
Adicionar Novo Serviço2



Sequence Diagram

Consulta Serviços Disponíveis - Main

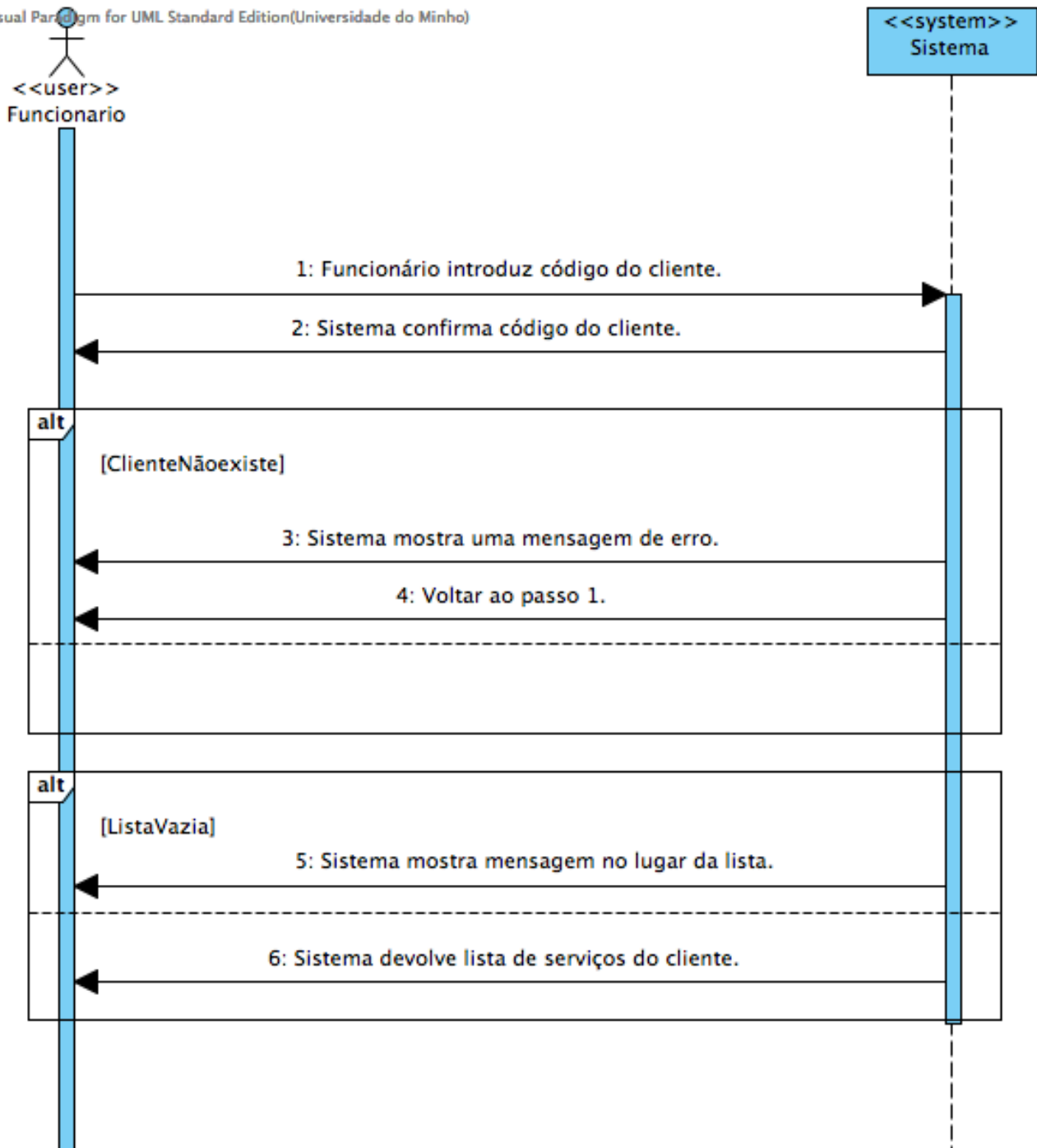
Visual Paradigm for UML Standard Edition(Universidade do Minho)



Consulta serviços subscritos pelo Cliente

- Main

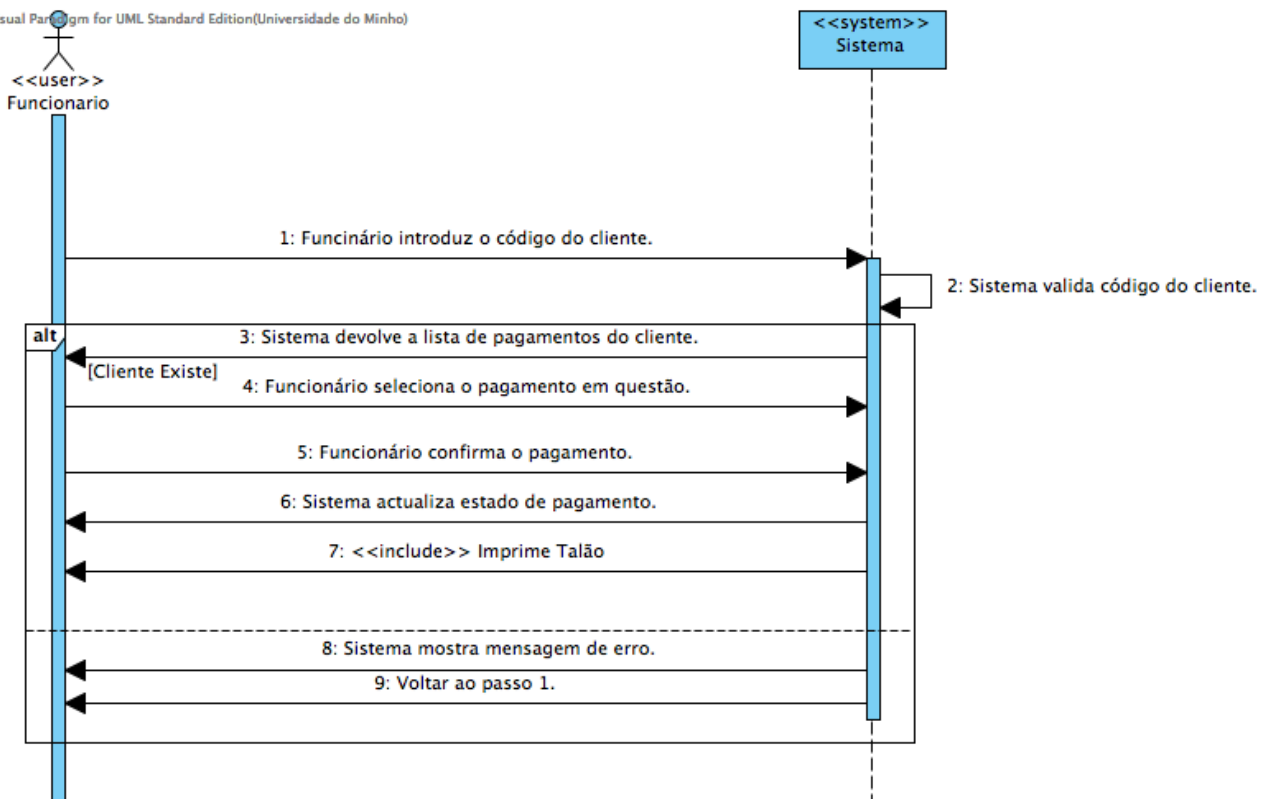
Visual Paradigm for UML Standard Edition(Universidade do Minho)



Sequence Diagram

Confirma Pagamento - Main

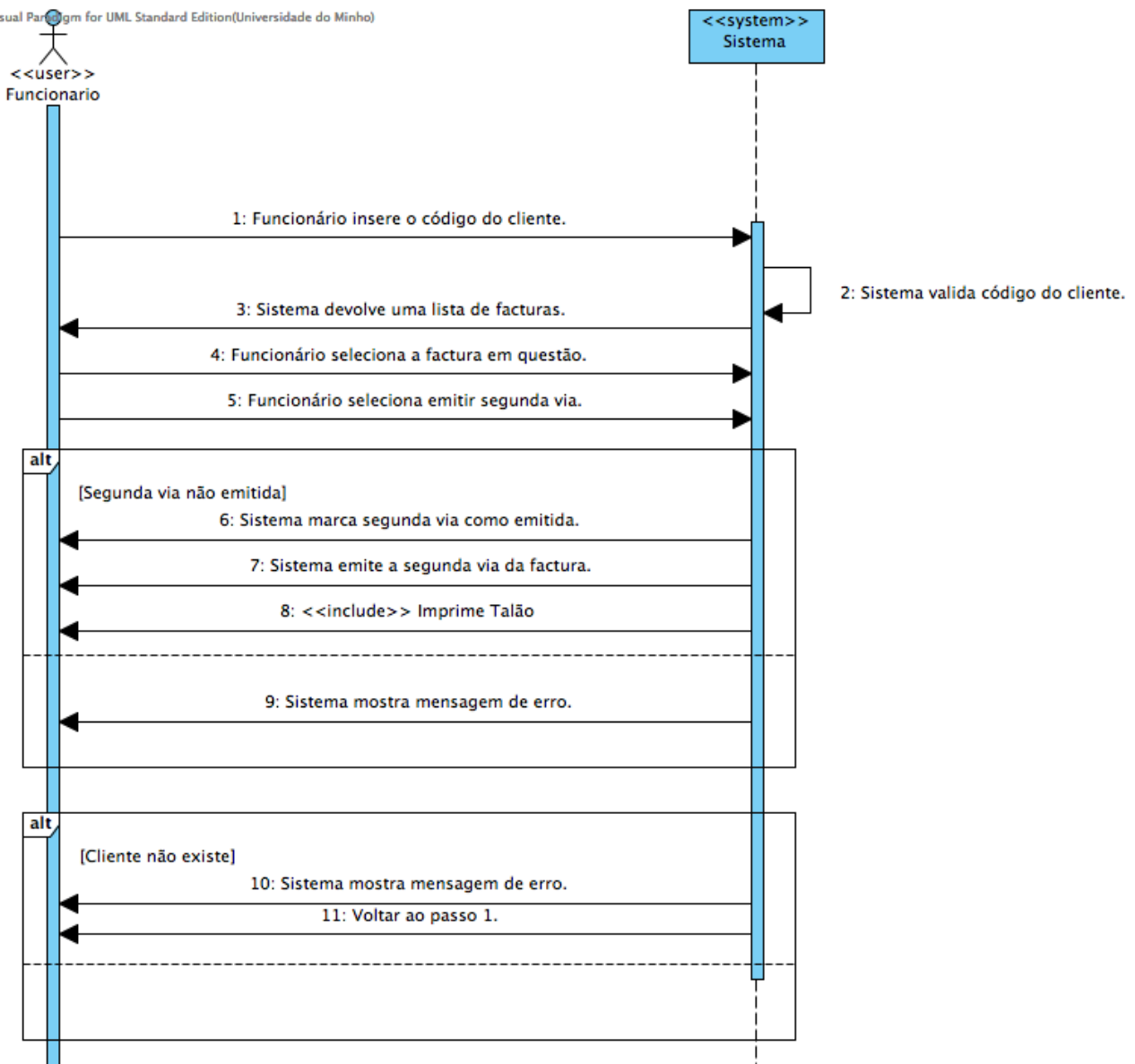
Visual Paradigm for UML Standard Edition (Universidade do Minho)



Sequence Diagram

Emite segunda via de factura - Main

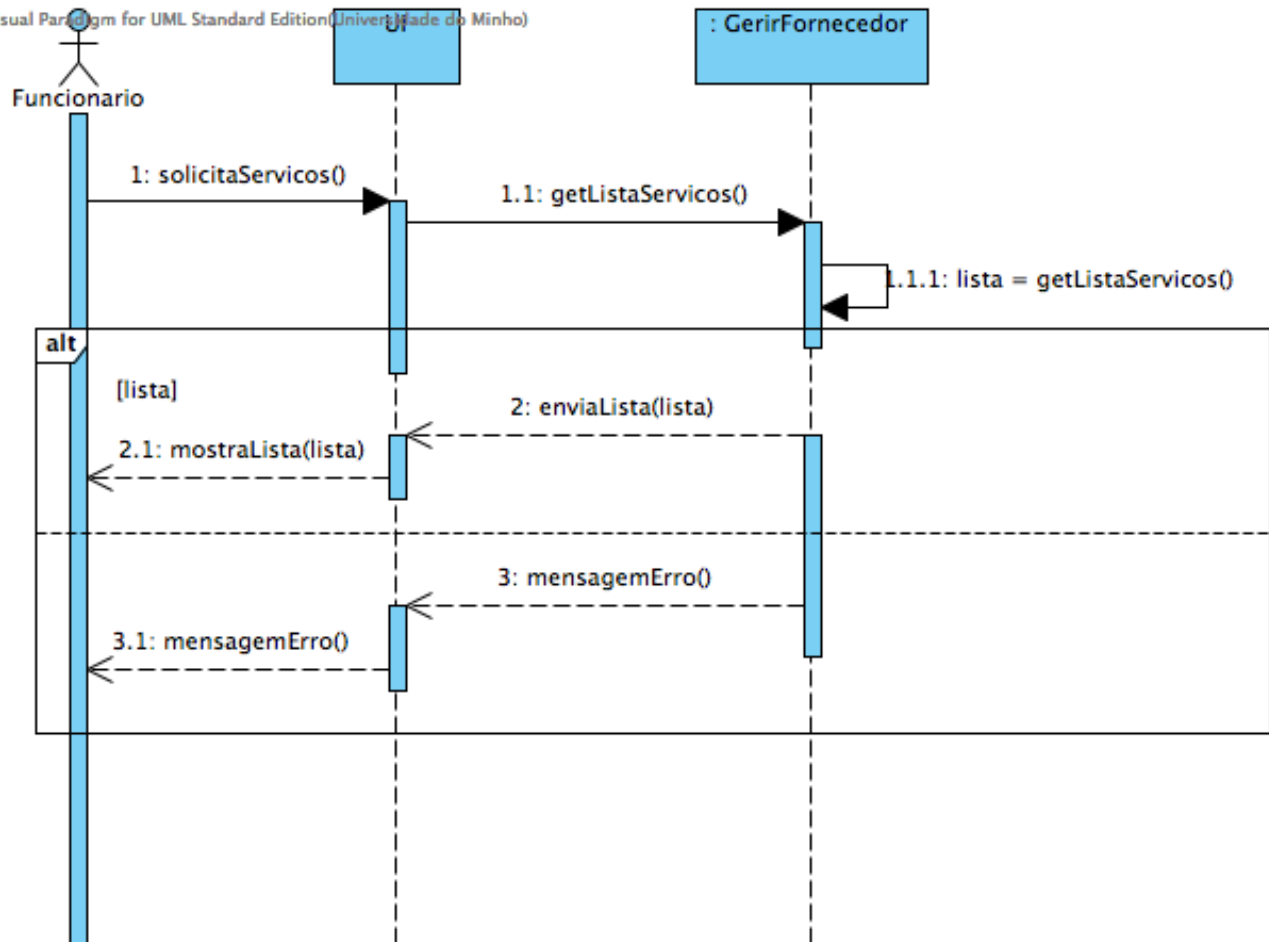
Visual Paradigm for UML Standard Edition(Universidade do Minho)



Sequence Diagram

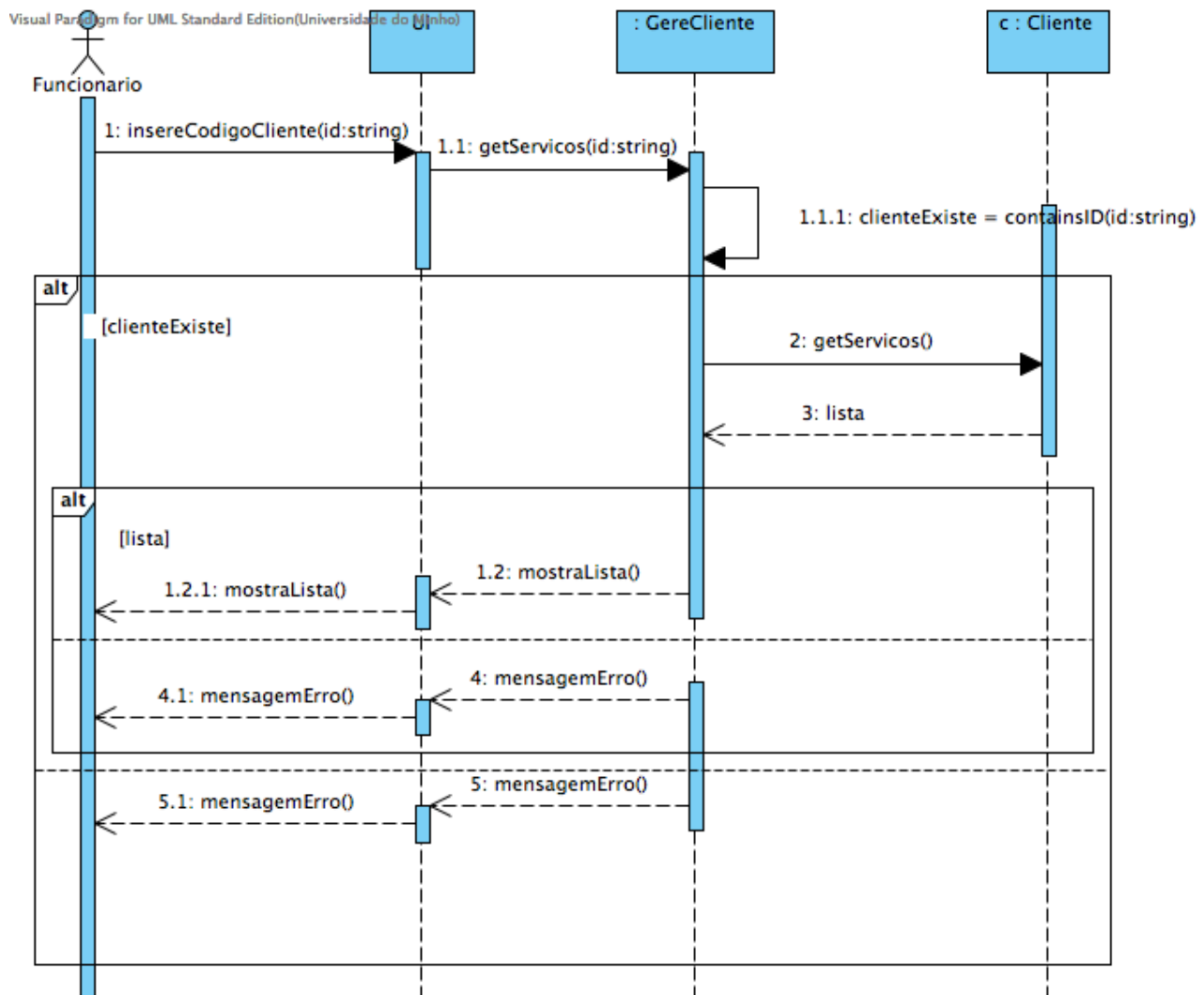
Consulta Serviços Disponíveis

Visual Paradigm for UML Standard Edition (Universidade do Minho)

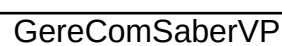


Sequence Diagram

Consulta serviços subscritos pelo Cliente



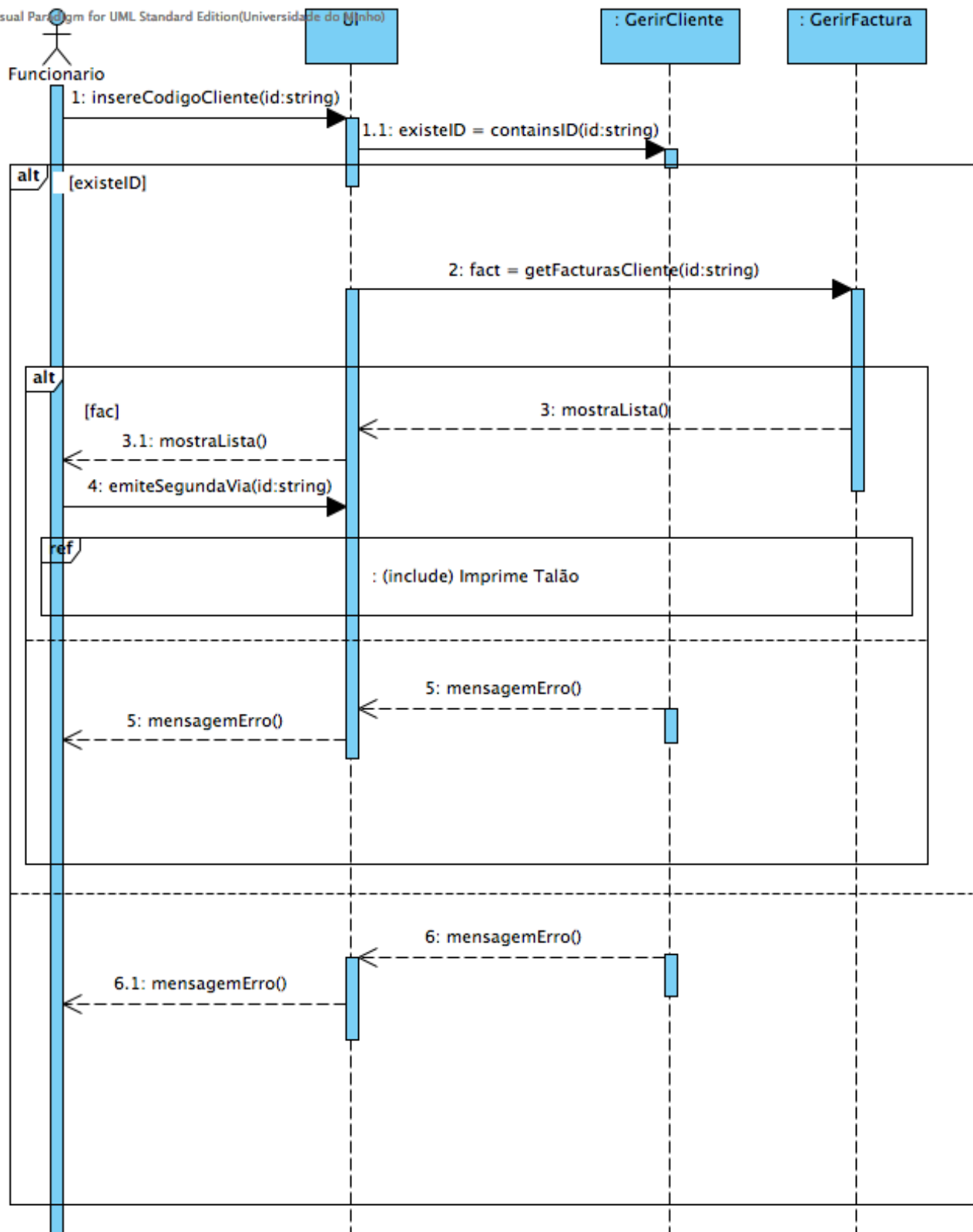
Confirma Pagamento



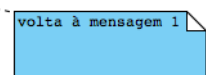
Sequence Diagram

Emite segunda via de factura

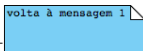
Visual Paradigm for UML Standard Edition(Universidade do Minho)



Autenticar (Login)2



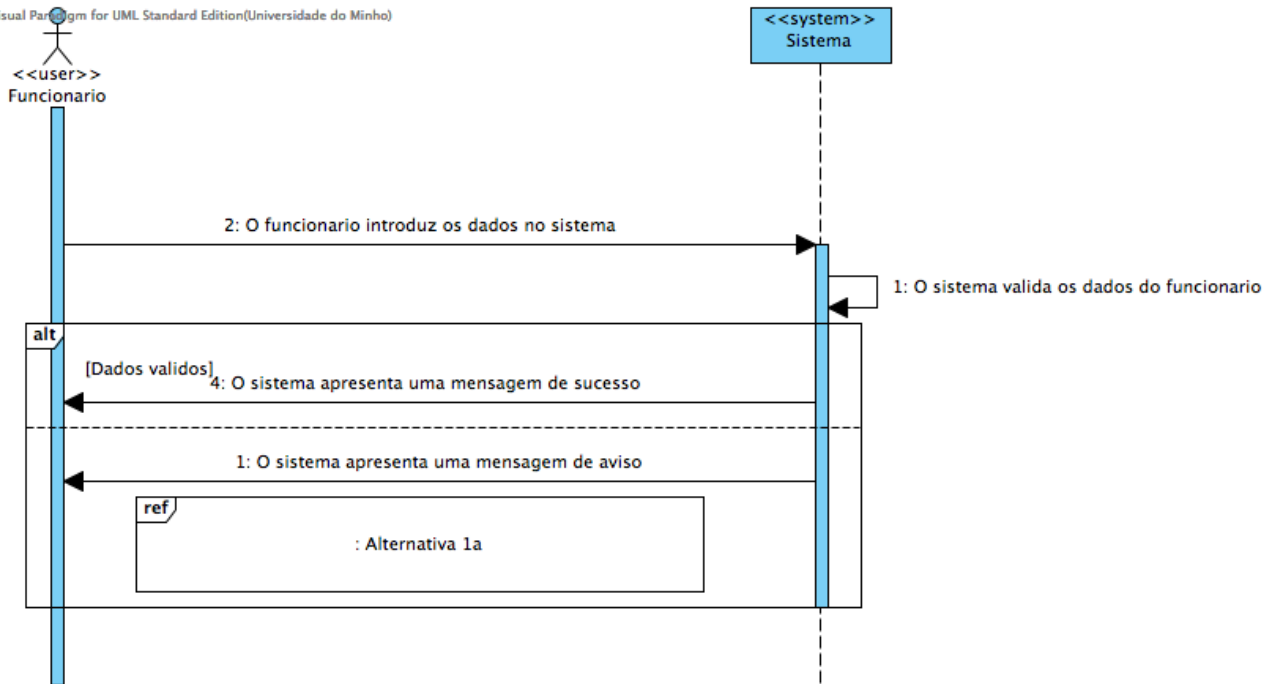
Visual Paradigm for UML Standard Edition(Universidade do Minho)



Sequence Diagram

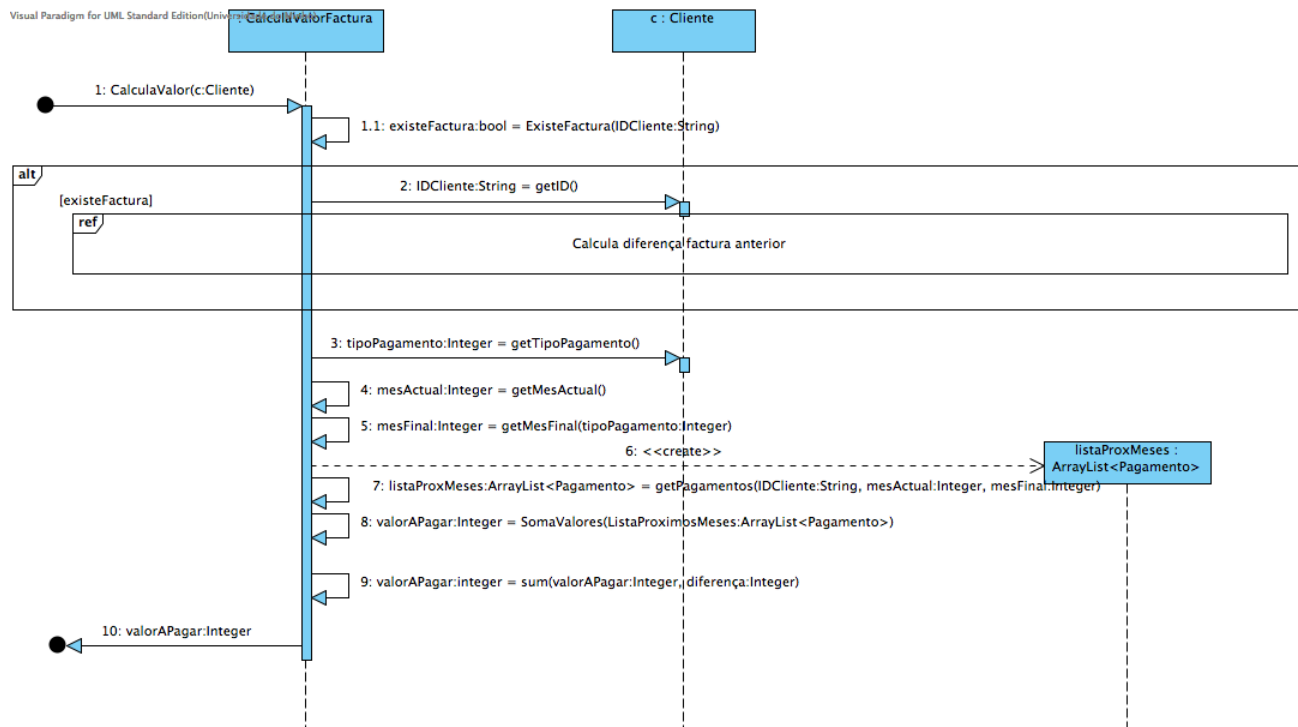
Autenticar (Login)

Visual Paradigm for UML Standard Edition(Universidade do Minho)



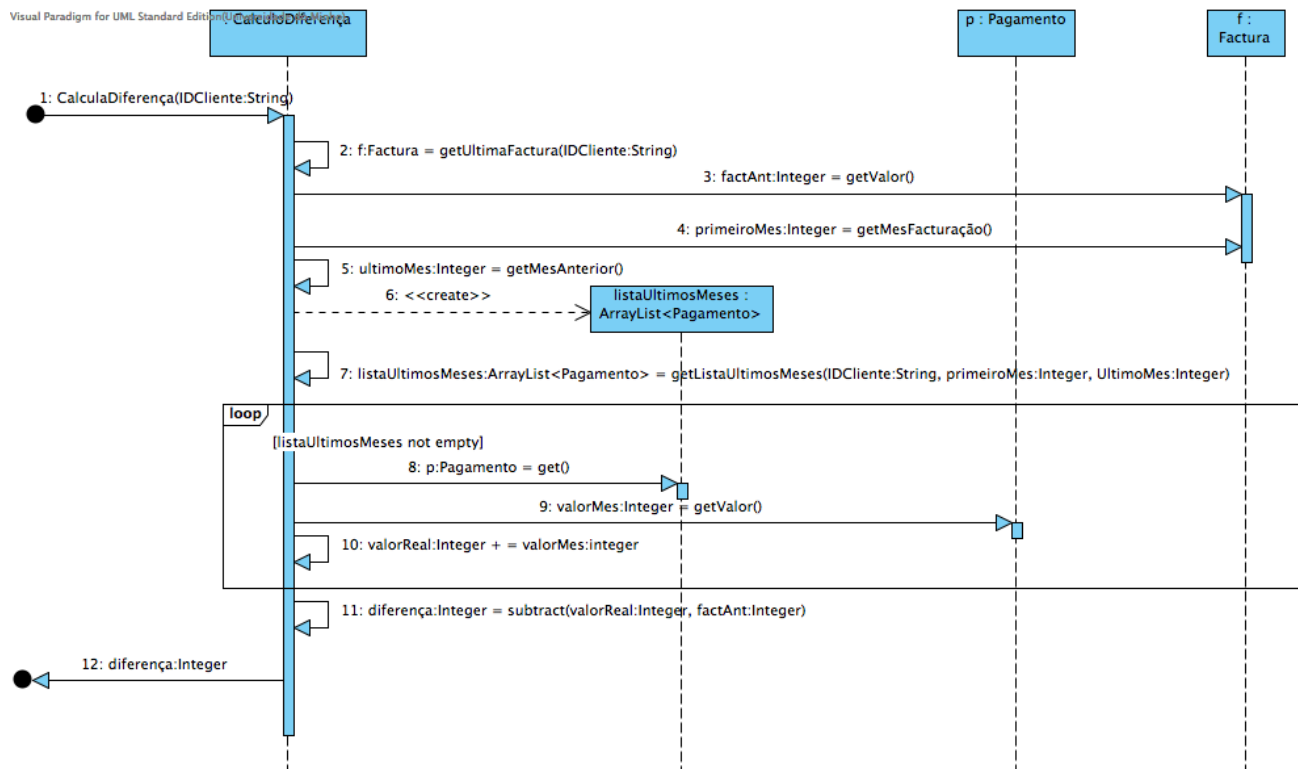
Sequence Diagram

Calcula Valor Factura



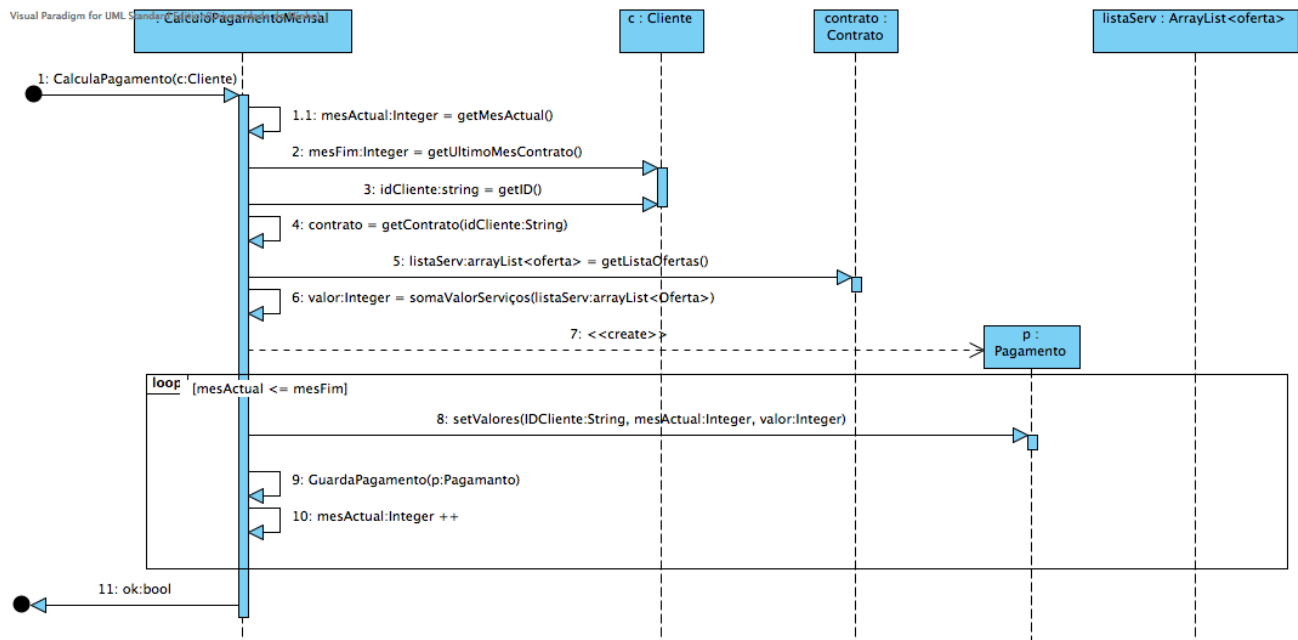
Sequence Diagram

Calcula diferença factura anterior



Sequence Diagram

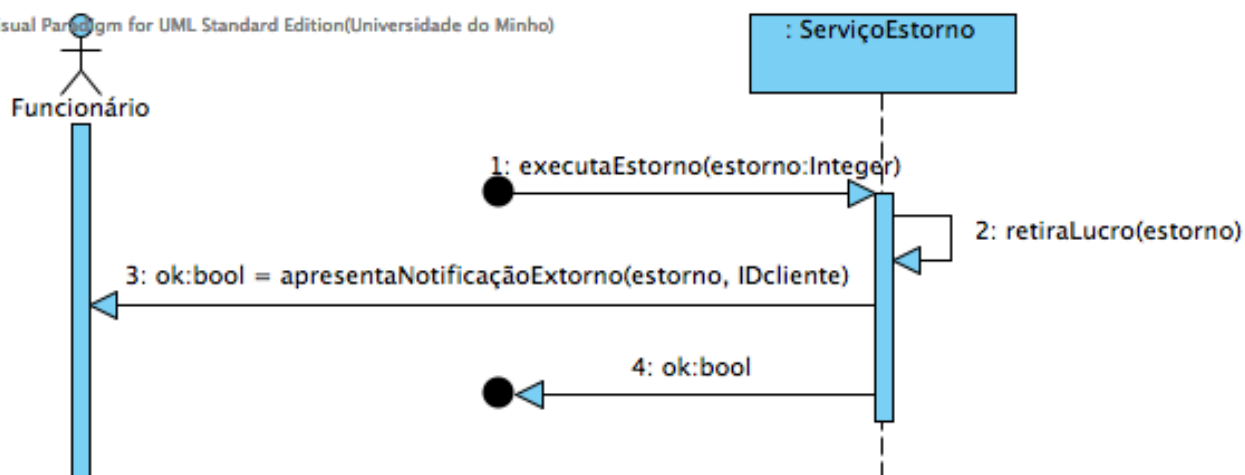
Calcular Pagamento Mensal



Sequence Diagram

Creditar Estorno

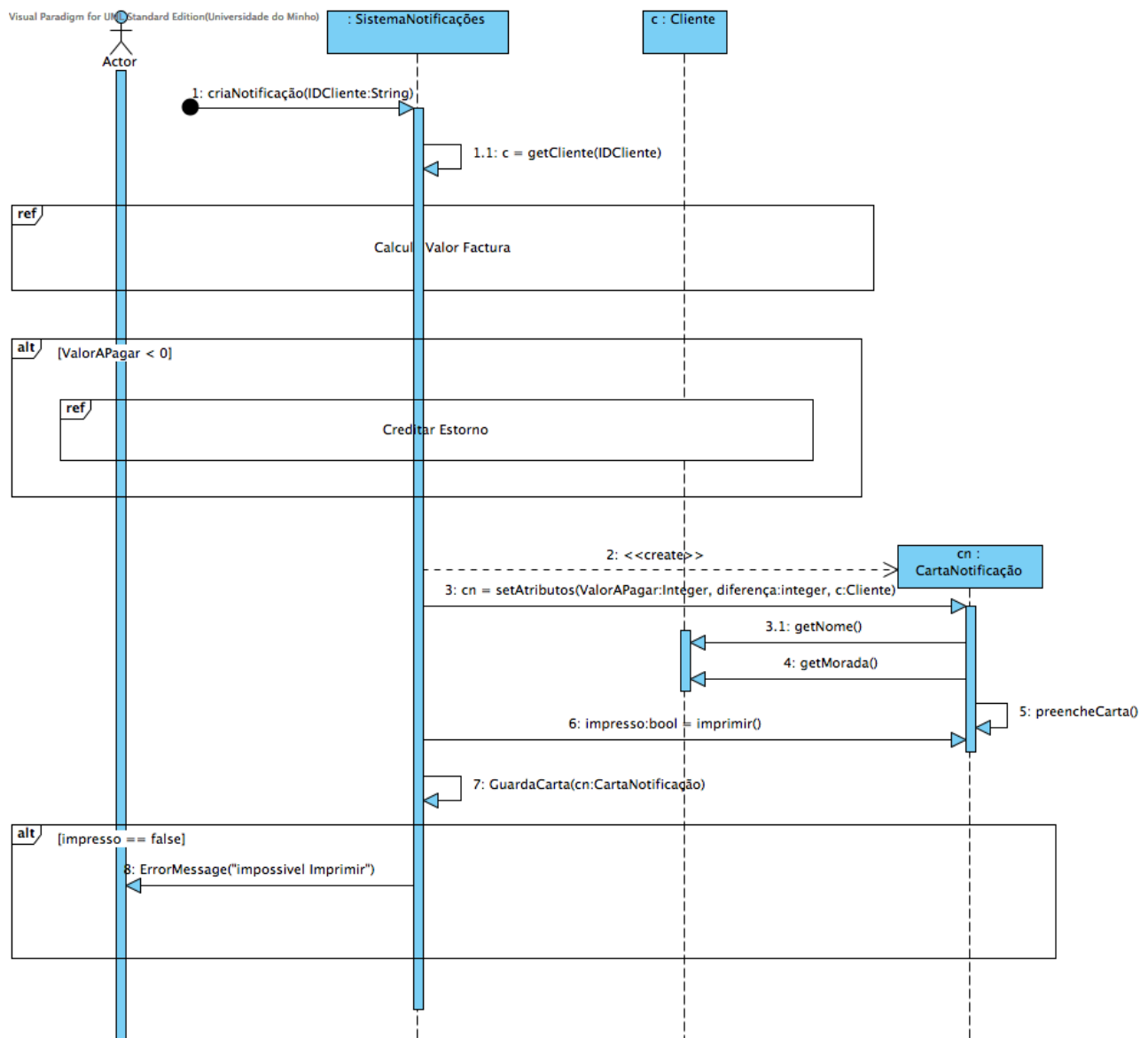
Visual Paradigm for UML Standard Edition(Universidade do Minho)



Sequence Diagram

Notifica Pagamento

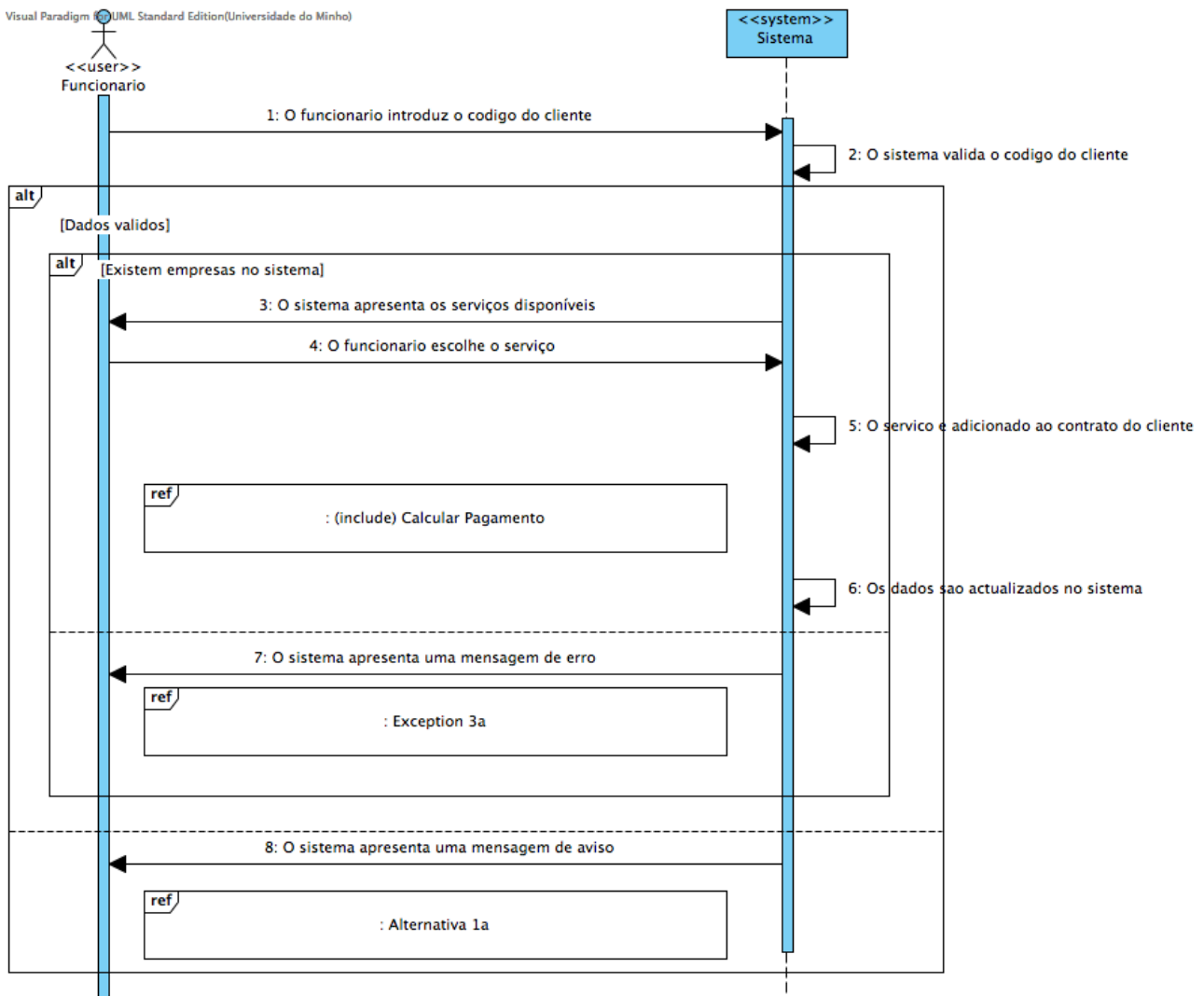
Visual Paradigm for UML Standard Edition (Universidade do Minho)



Sequence Diagram

Adicionar Novo Serviço

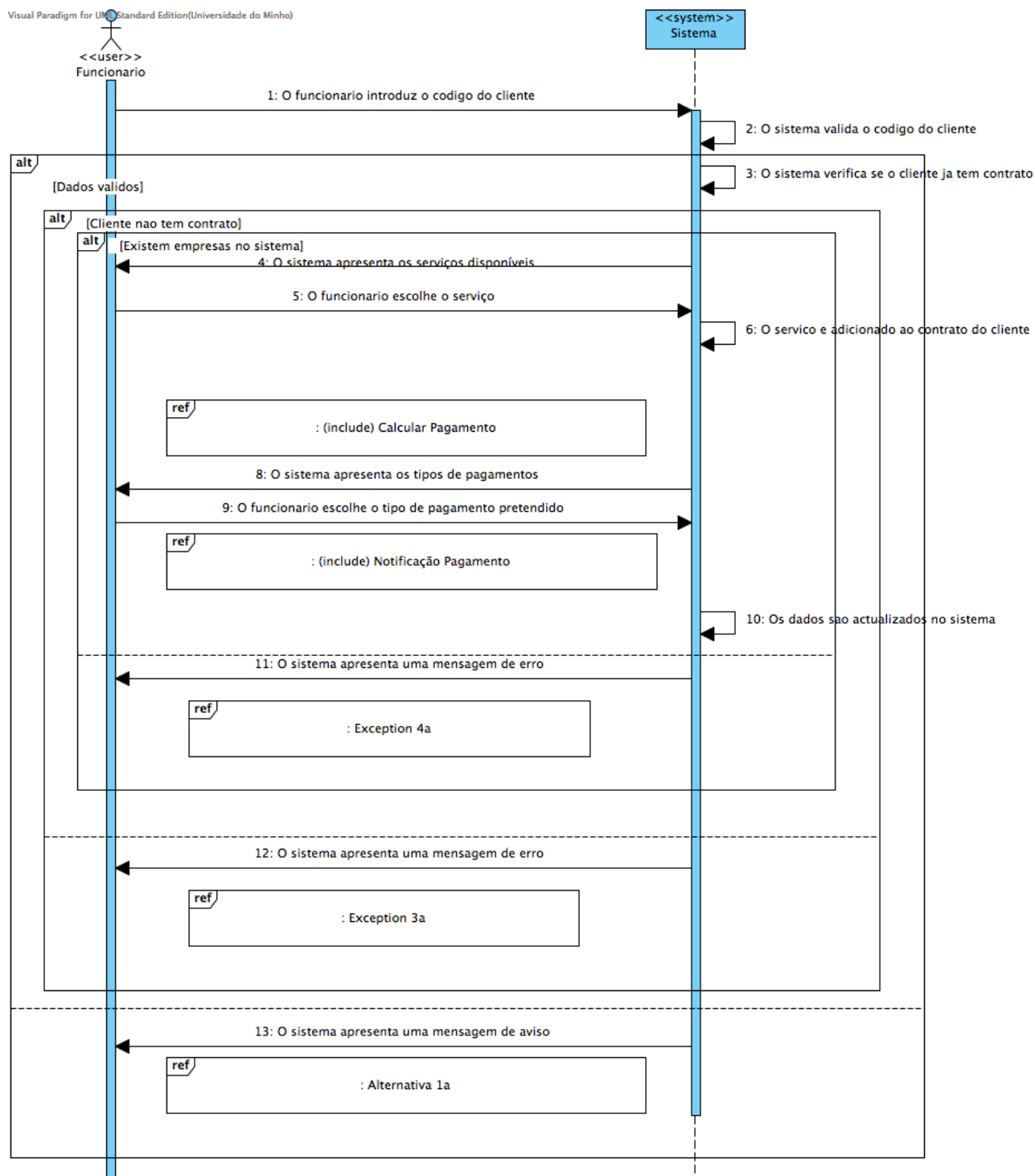
Visual Paradigm UML Standard Edition (Universidade do Minho)



Sequence Diagram

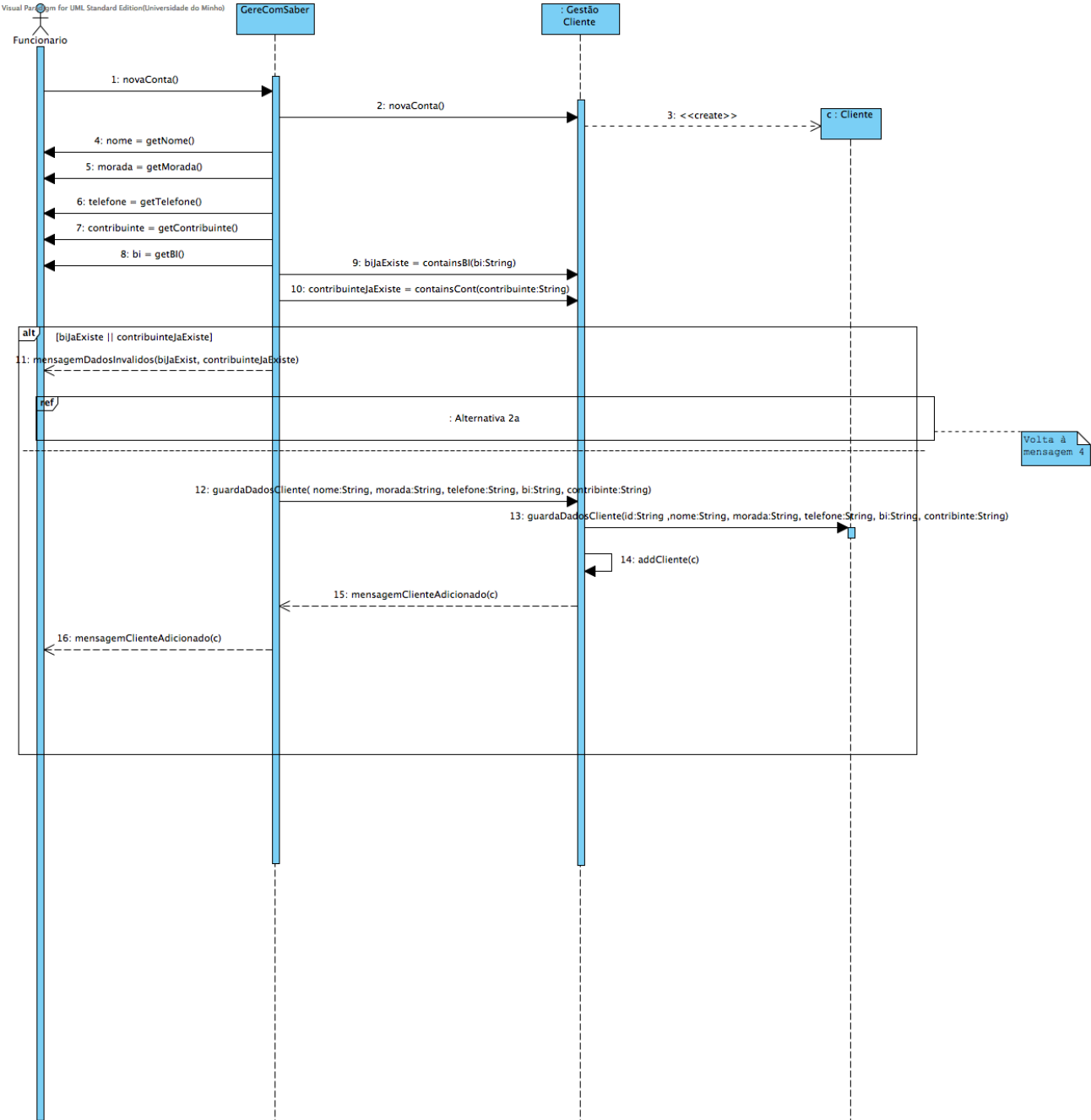
Cria Contrato

Visual Paradigm for UML Standard Edition (Universidade do Minho)



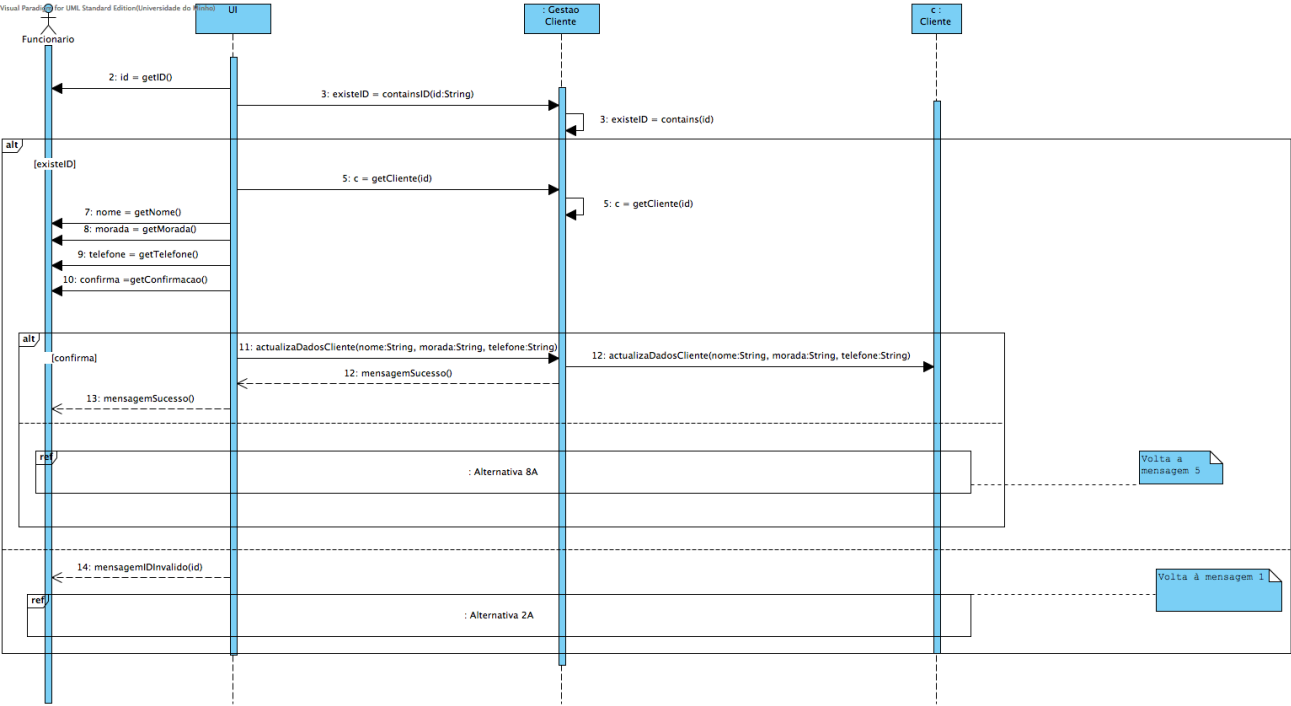
Sequence Diagram

Adicionar Cliente



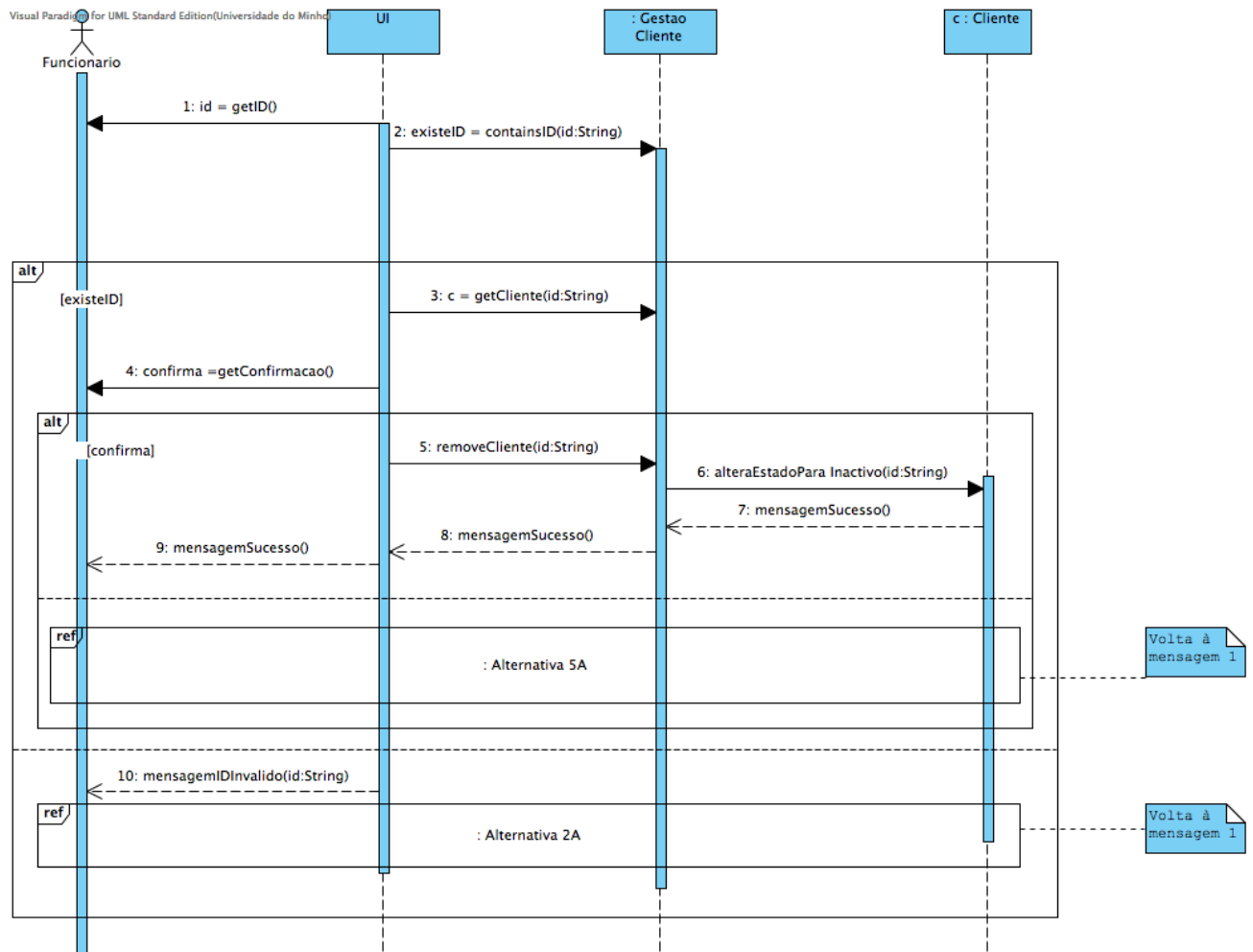
Sequence Diagram

Edita Cliente



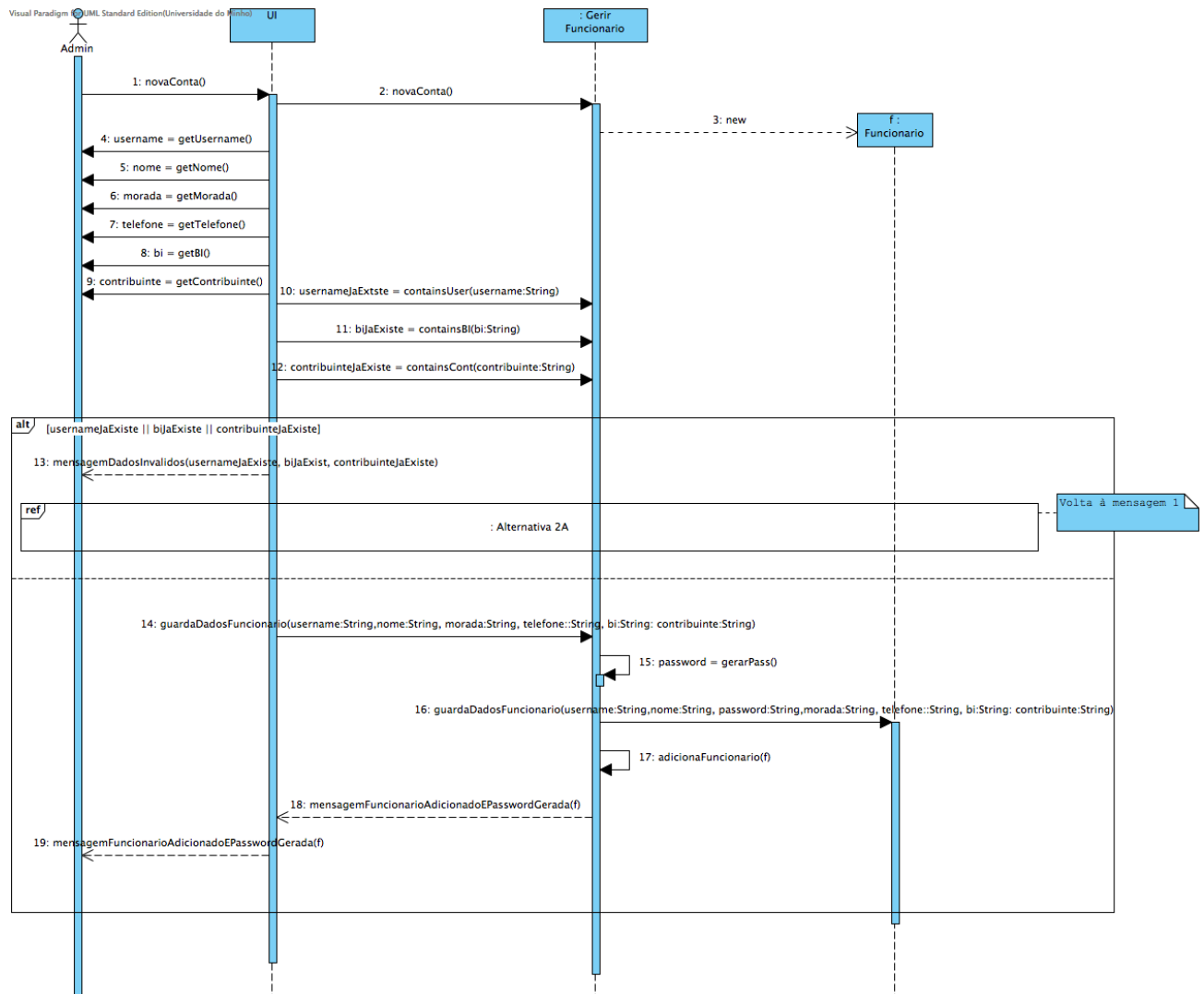
Sequence Diagram

Remove Cliente



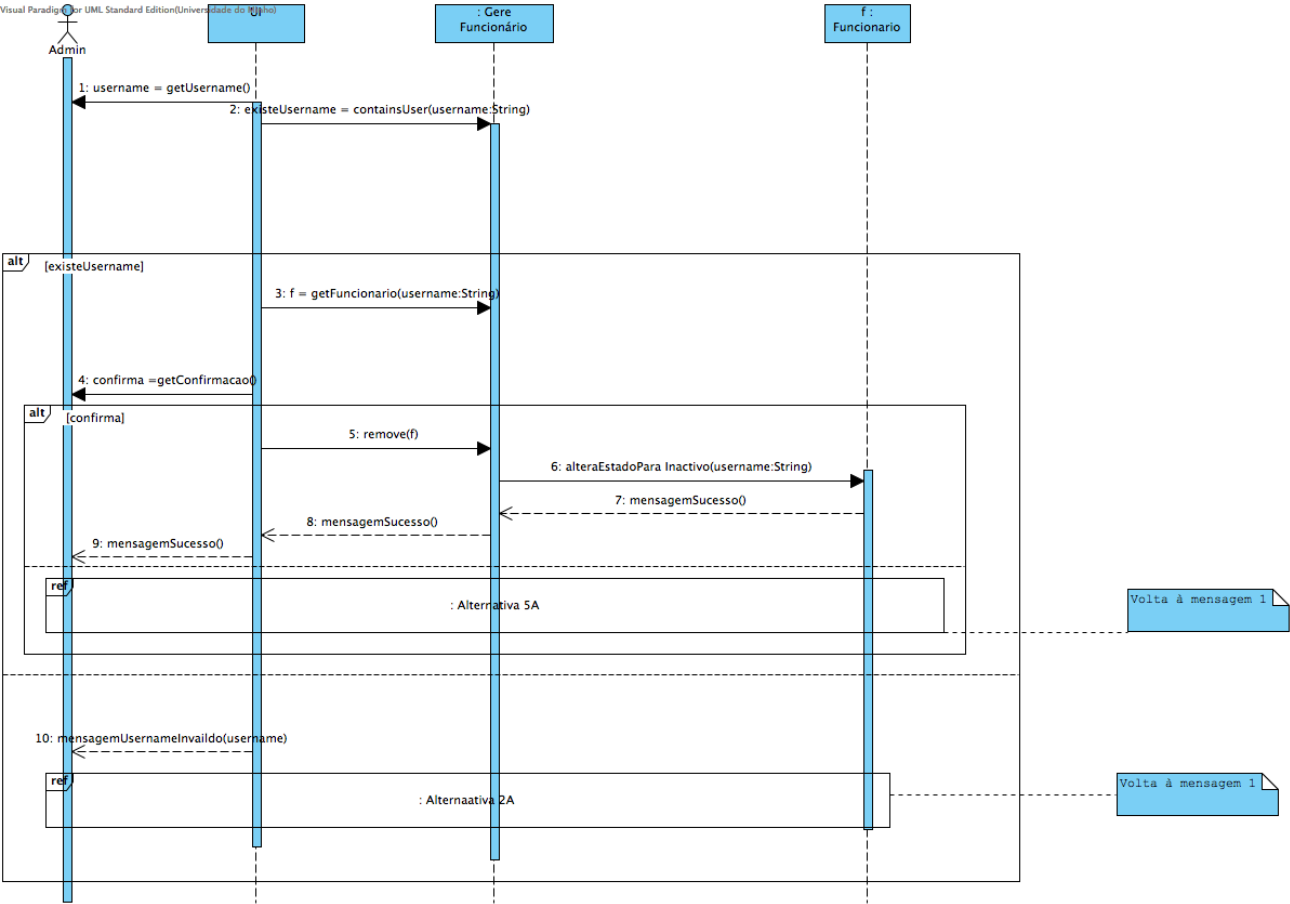
Sequence Diagram

Adiciona funcionario



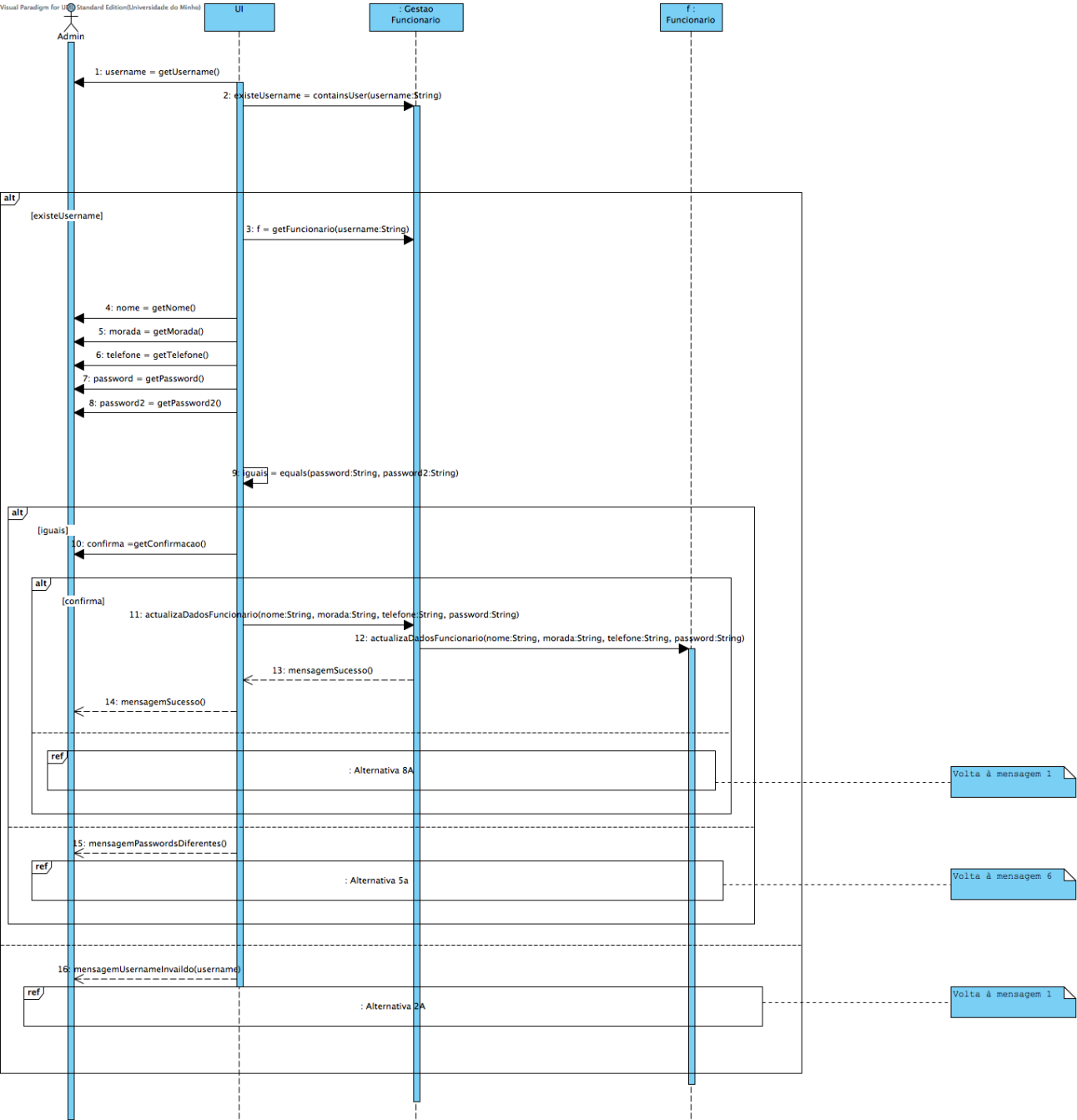
Sequence Diagram

Remove funcionario



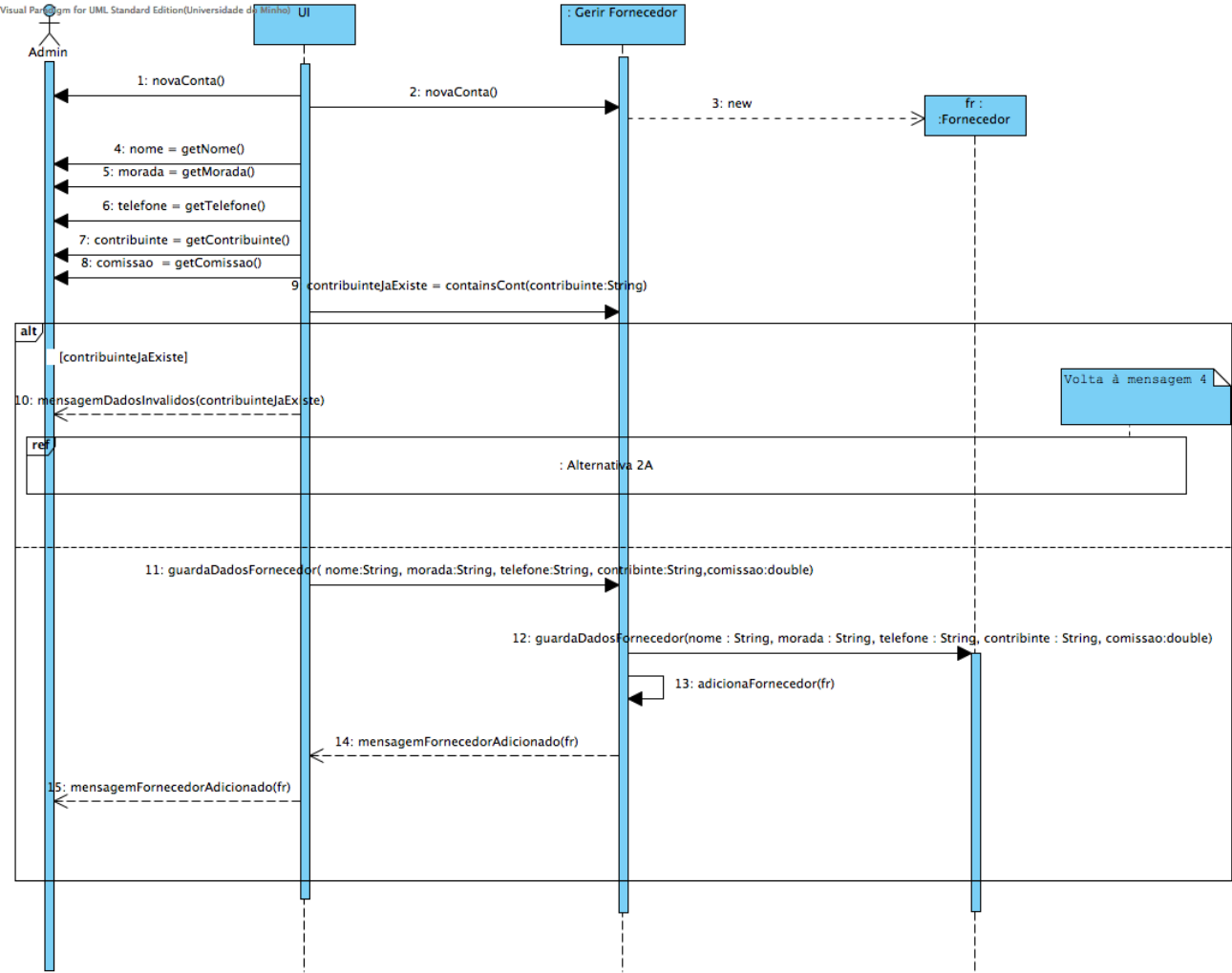
Sequence Diagram

Edita funcionario



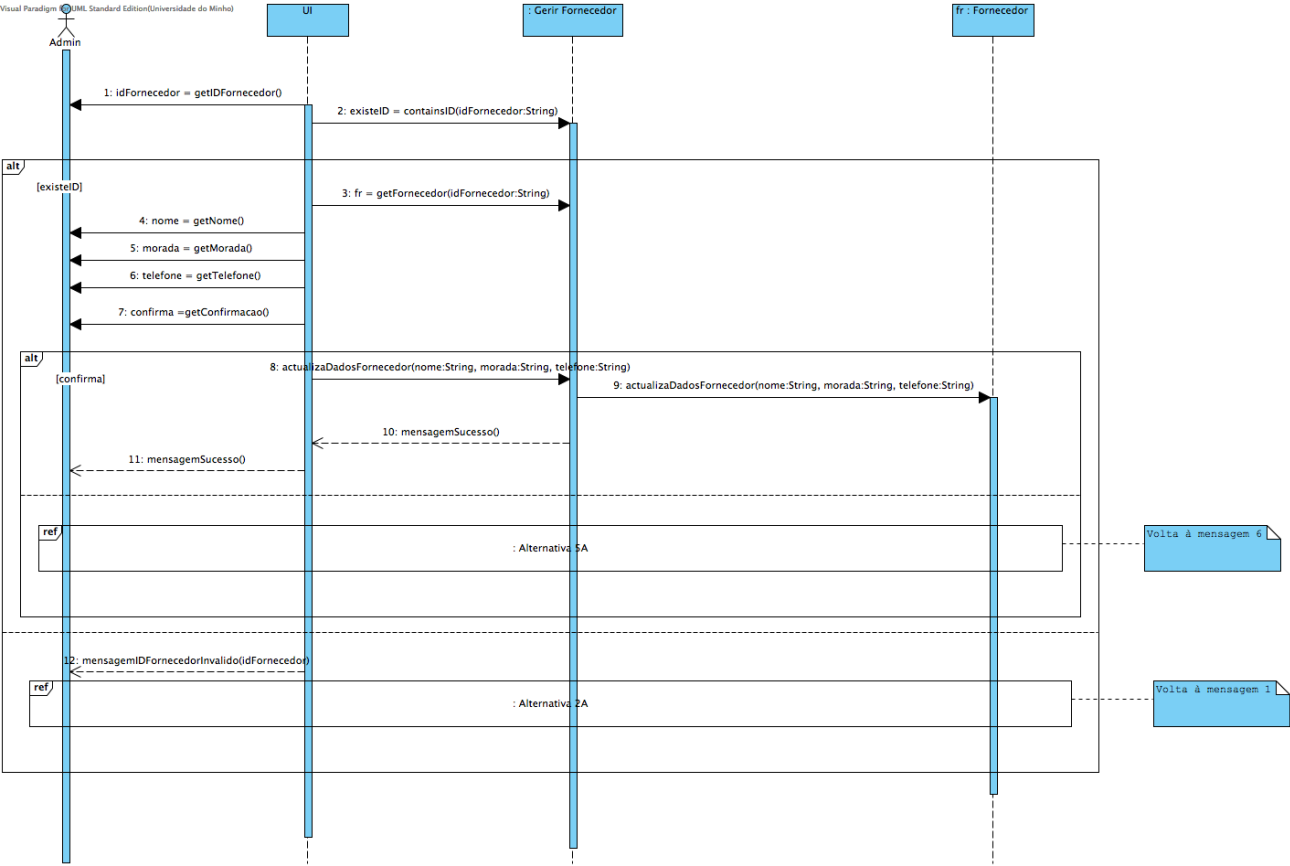
Sequence Diagram

Adiciona Fornecedor



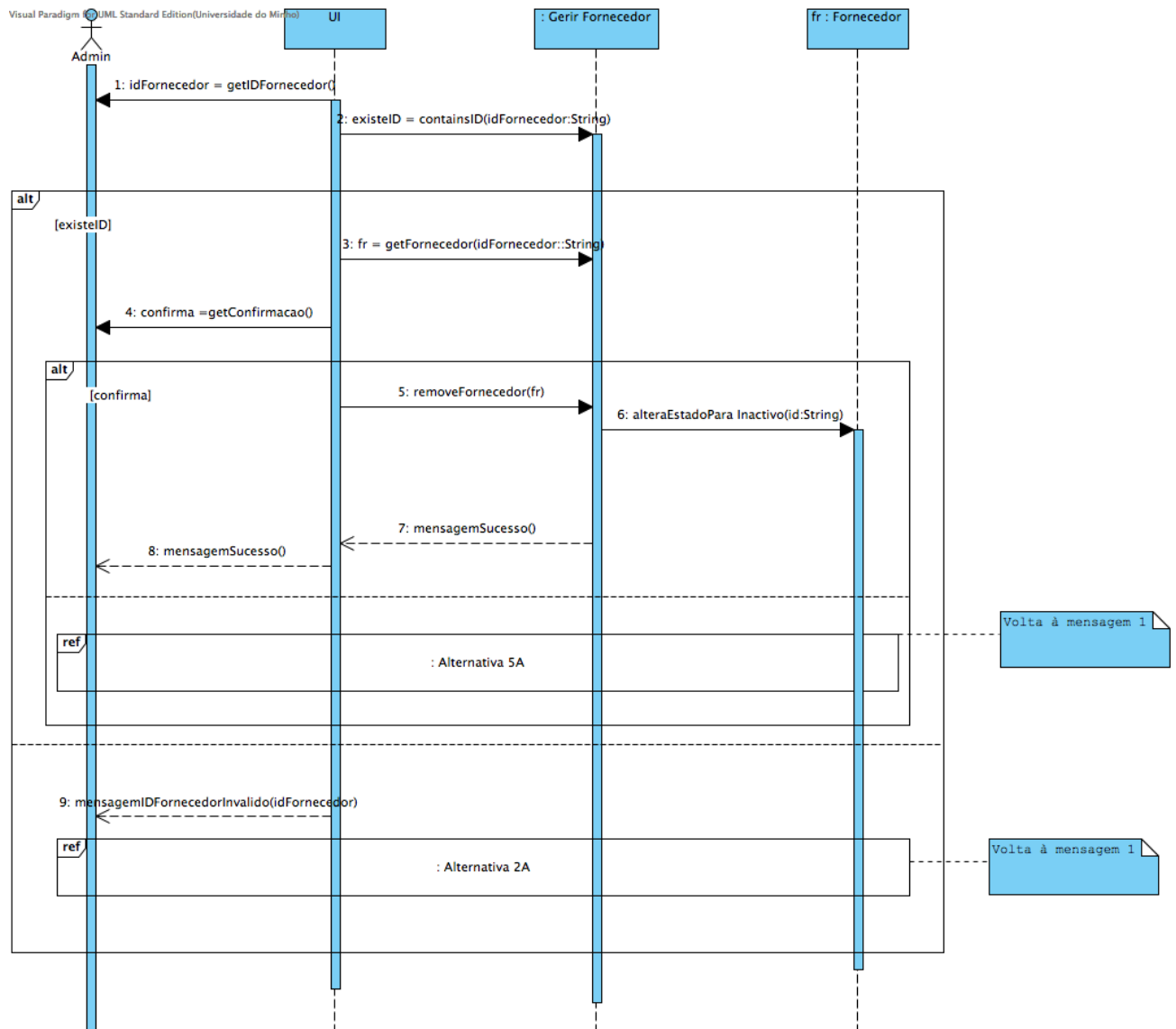
Sequence Diagram

Edita Fornecedor



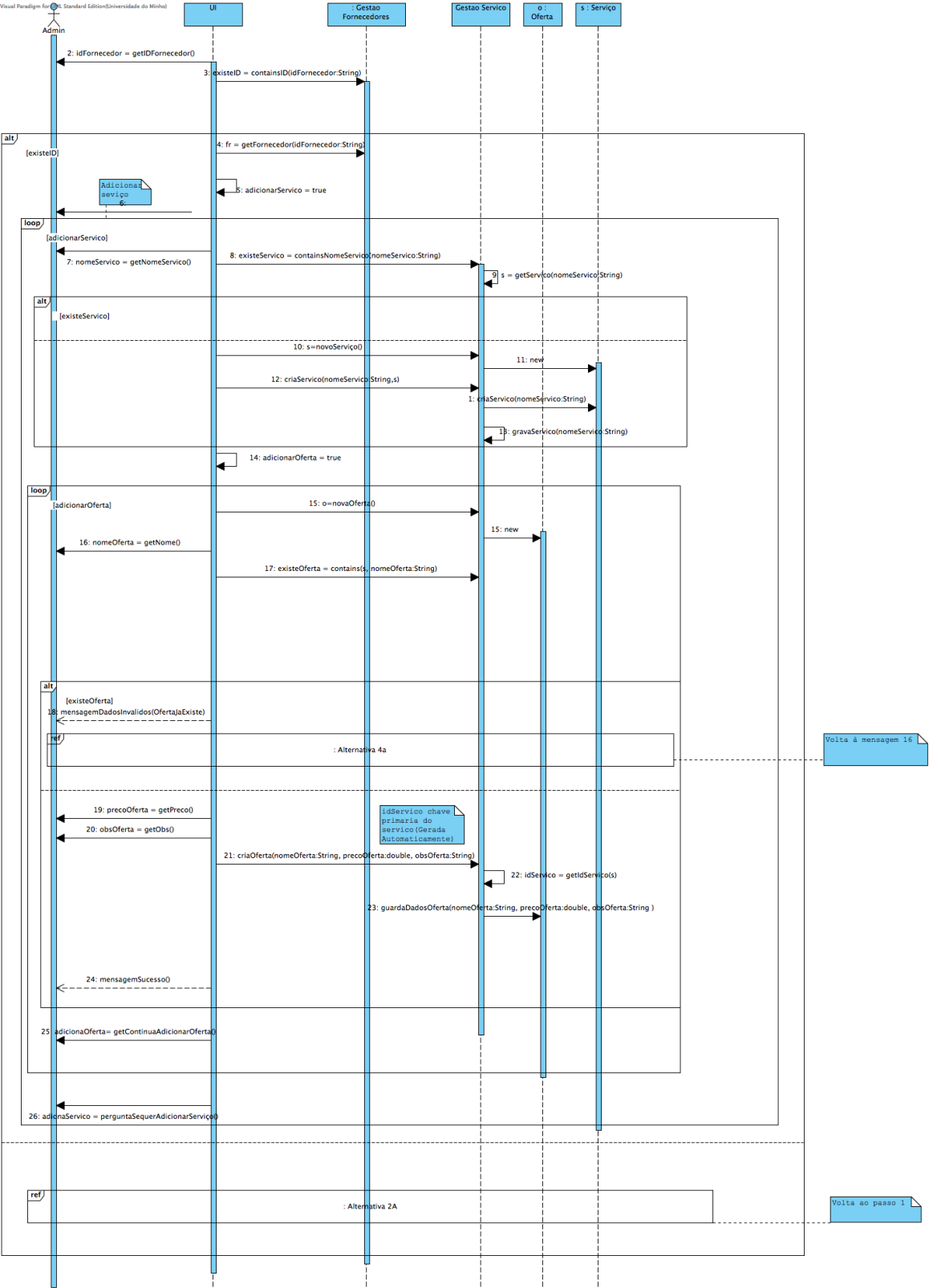
Sequence Diagram

Remove Fornecedor



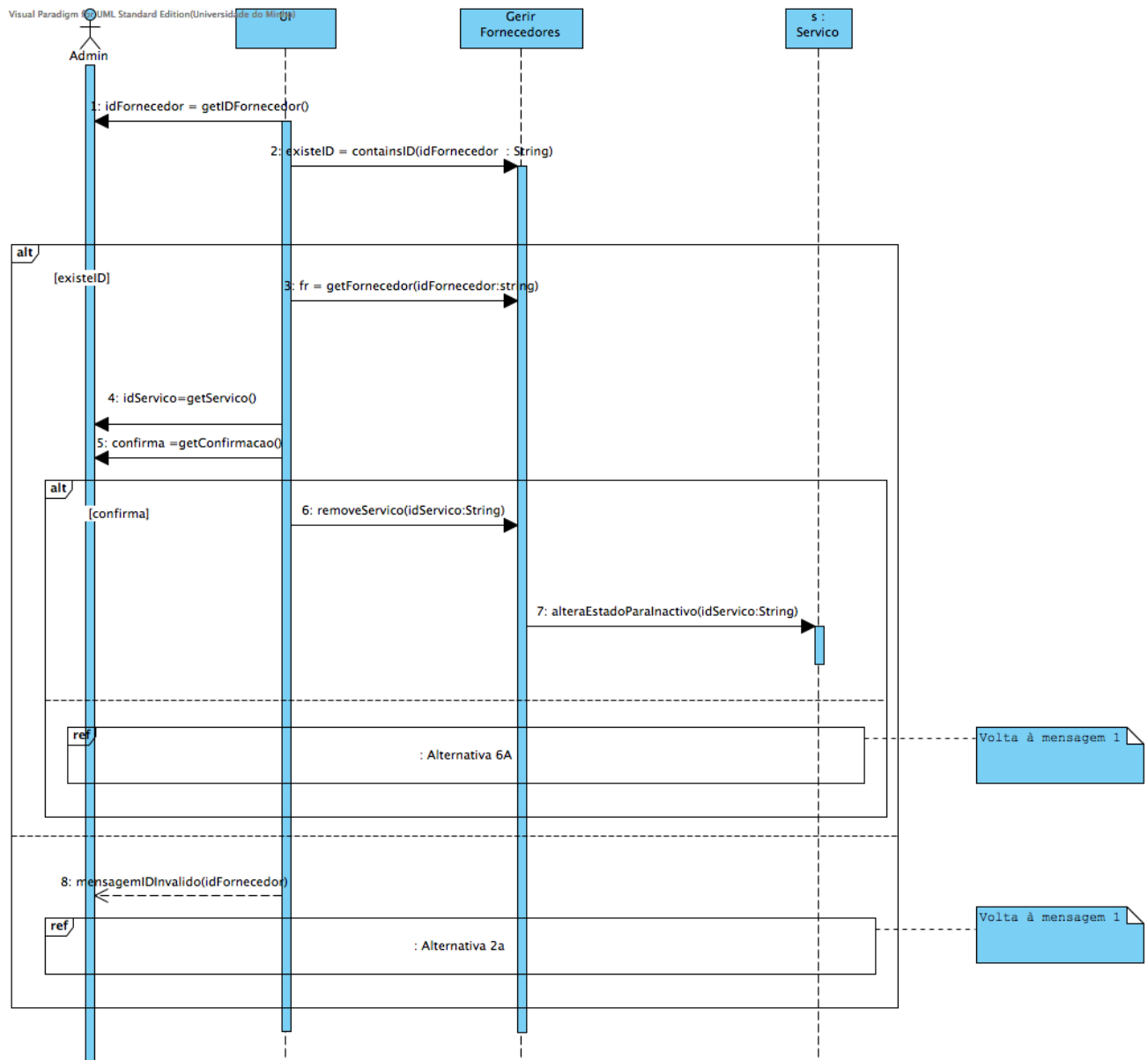
Sequence Diagram

Criar serviço

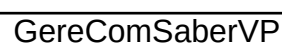


Sequence Diagram

Remover Serviço



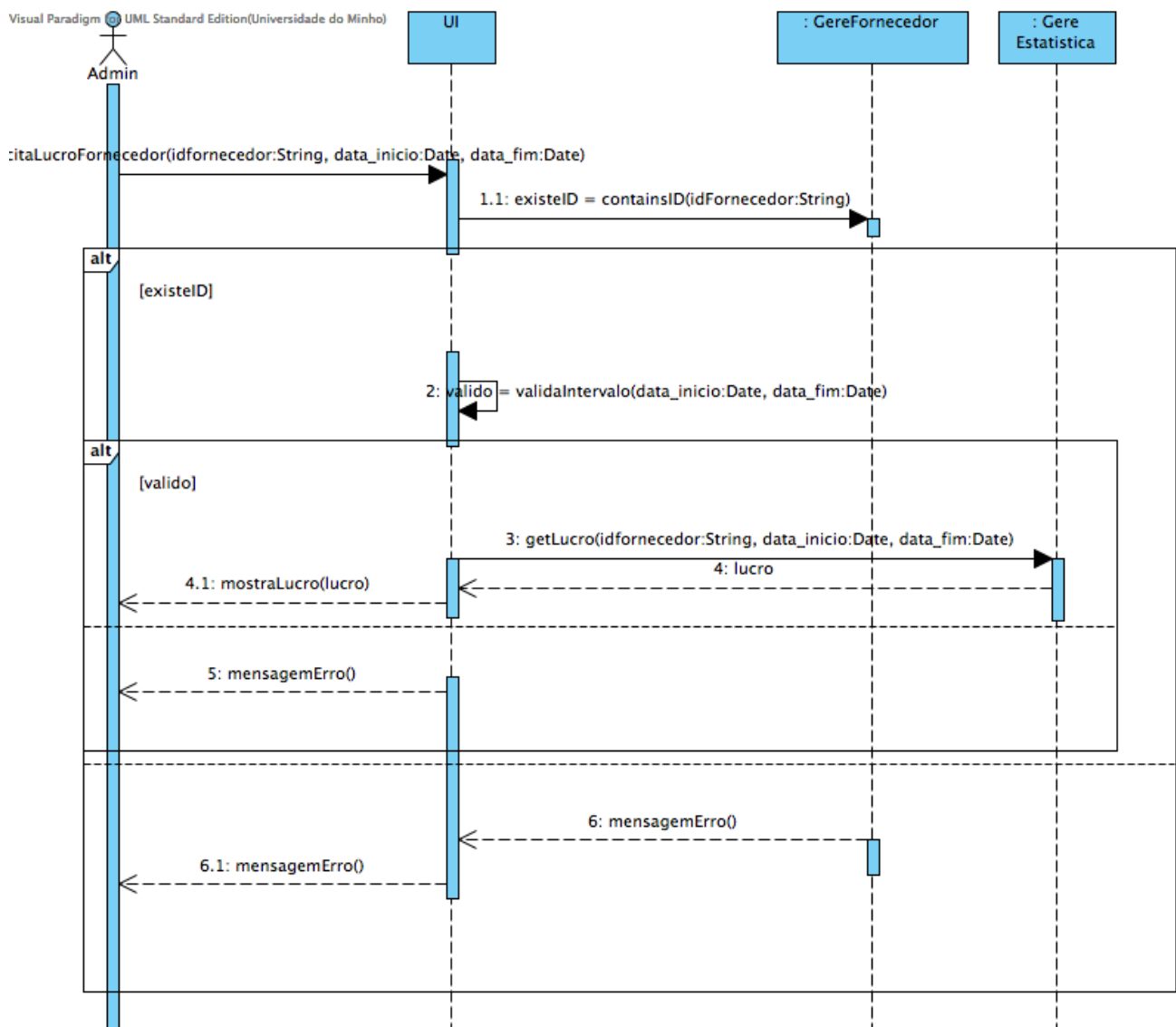
Editar Preço de serviço



Sequence Diagram

Ver Lucro Fornecedor

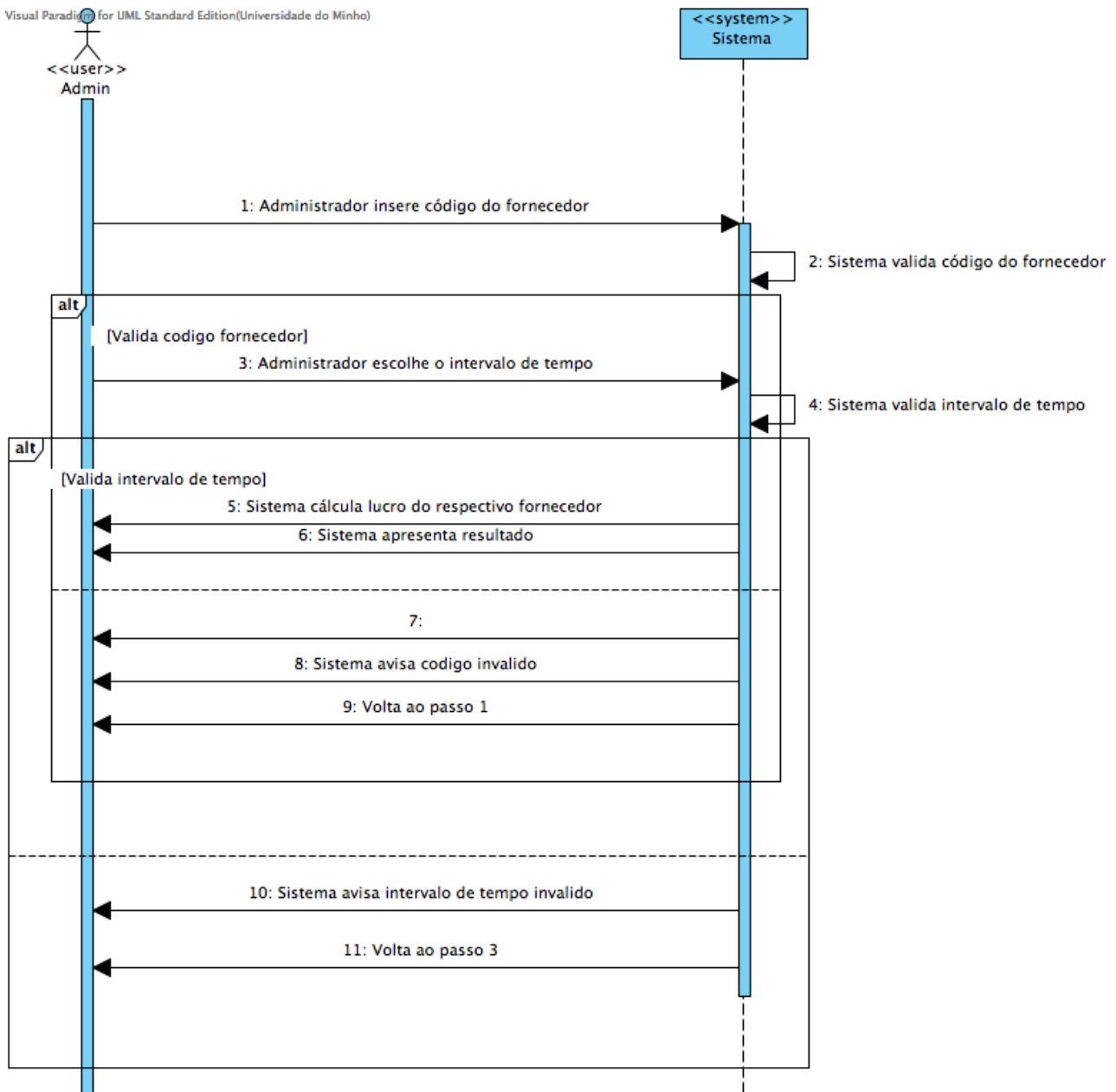
Visual Paradigm UML Standard Edition(Universidade do Minho)



Sequence Diagram

Ver Lucro Fornecedor - Consultar lucro de fornecedor

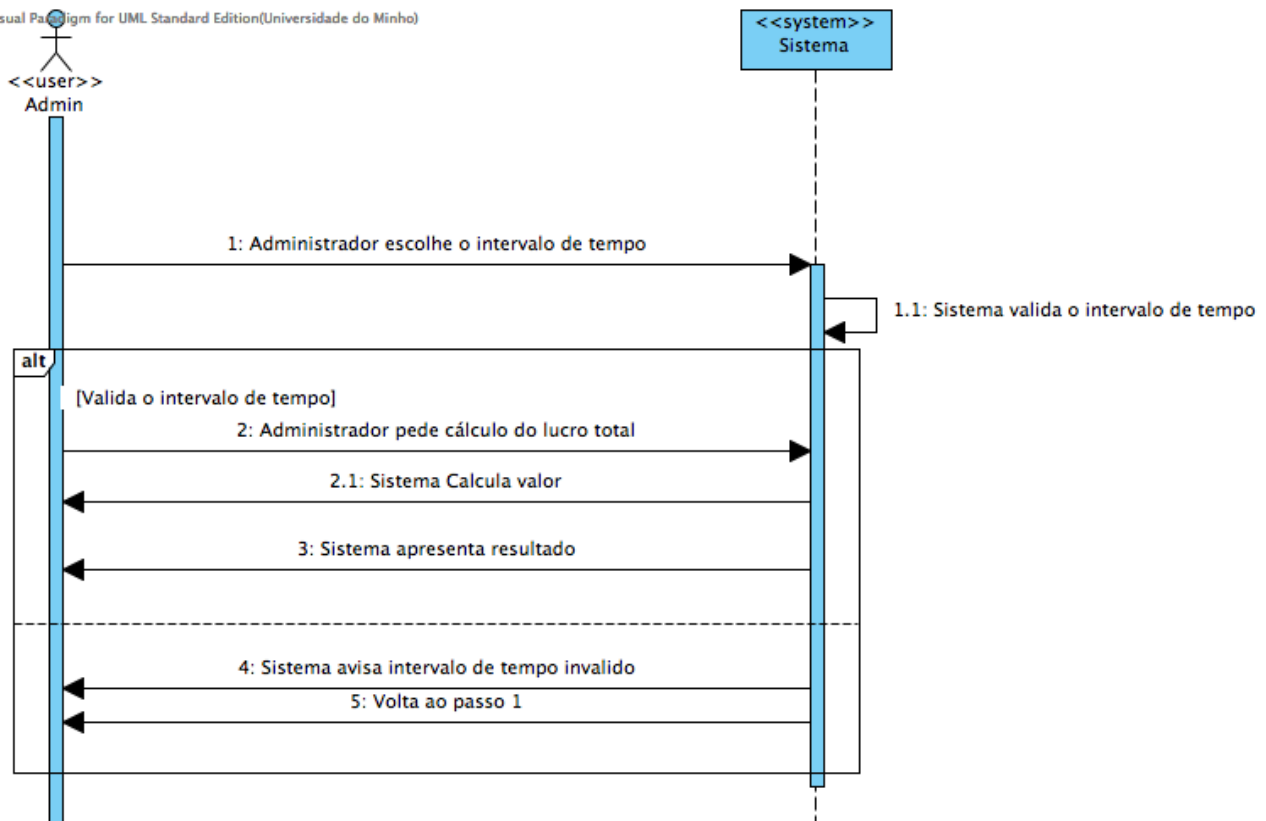
Visual Paradigm for UML Standard Edition(Universidade do Minho)



Sequence Diagram

Consultar Lucro Total - Consultar Lucro Total

Visual Paradigm for UML Standard Edition(Universidade do Minho)



Sequence Diagram

Consultar Lucro Total

Visual Paradigm for UML Standard Edition (Universidade do Minho)

