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

Desenvolvimento de Sistemas de Software (Grupo 22)

Ano Lectivo de 2009/10

Universidade do Minho

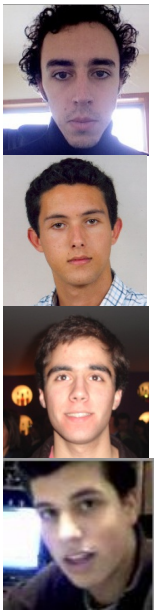
escola de engenharia

departamento de informática

Gestão de Condomínios

Desenvolvido por:

António Bouças (49330)
Paulo Festa (49419)
Tiago Mendes (51760)
André Magalhães (51821)



Novembro, 2009

Resumo

O projecto desenvolvido e descrito neste documento é parte integrante da cadeira de *Desenvolvimento de Sistemas de Software* do curso Licenciatura em Engenharia Informática (LEI) e visa modelar e concretizar um projecto.
Este projecto tem como principal objectivo desenvolver uma aplicação multi-utilizador que permita ao fornecedor proporcionar serviços, ao cliente requisita-los e ao administrador garantir a correcta gestão dos condomínios fechados.

Conteúdo

Conteúdo 1

Lista de Figuras 3

1 Introdução 6
- 1.1 Apresentação do Caso de Estudo . . . 6
- 1.2 Objectivos . . . 7

2 Modelo do Domínio 8
- 2.1 Glossário . . . 9
- 2.2 Decisões . . . 10
- 2.3 Diagrama do modelo do domínio . . . 11
- 2.4 Alterações na segunda etapa . . . 13

3 Use Cases 15
- 3.1 Use Cases multi-utilizador . . . 15
  - 3.1.1 Login e Logout . . . 15
  - 3.1.2 Pesquisar actividade . . . 15
  - 3.1.3 Consultar contratos . . . 16
- 3.2 Administrador . . . 16
  - 3.2.1 Gerir Utilizadores . . . 16
  - 3.2.2 Análise de Resultados . . . 16
  - 3.2.3 Alterar Comissão . . . 17
- 3.3 Fornecedor . . . 17
  - 3.3.1 Consultar Saldo . . . 17
  - 3.3.2 Editar Actividade . . . 17
  - 3.3.3 Anular Serviço . . . 17
  - 3.3.4 Anular Actividade . . . 17
  - 3.3.5 Inserir Serviço . . . 18
  - 3.3.6 Inserir Actividade . . . 18
- 3.4 Cliente . . . 19

3.4.1 Consulta Histórico 19
3.4.2 Cancelar Serviço 19
3.4.3 Cancelar Actividade 20
3.4.4 Requisitar Serviço 20
3.4.5 Requisitar Actividade 20
3.5 Alterações na segunda etapa 21

4 Diagramas de Sequência **22**
4.1 Decisões 22
4.2 Exemplo do processo de refinamento 23

5 Diagrama de Classes **26**
5.1 Decisões 26

6 Conclusão **29**

7 Anexos **31**
7.1 Descrições textuais dos Use Cases 31
7.2 Diagramas de Sequência 59
7.3 Métodos e Atributos do Diagrama de Classes 87

Lista de Figuras

2.1 Modelo do Domínio - Lista de Actividades 9
2.2 Modelo do Domínio - Relação entre Contrato,Cliente e Valor 10
2.3 Modelo do Domínio 12
2.4 Modelo do Domínio obtido na segunda etapa 14

3.1 Use Case "Inserir Actividade"(Main Flow) 18
3.2 Use Case "Inserir Actividade"(Alternative & Exception) 19
3.3 Use Cases 20

4.1 Diagrama de Sequência de Sistema para Consultar Histórico 24
4.2 Diagrama de Sequência já com Sub-Sistemas para Consultar Histórico . . . 24
4.3 Diagrama de Sequência de Implementação após sofrer todas as refinações . 25

5.1 Diagrama de Classes (os métodos e atributos foram omitidos) 28

7.1 Descrição textual - Adicionar Acesso 32
7.2 Descrição textual - Alterar Comissão 33
7.3 Descrição textual - Anular Actividade 34
7.4 Descrição textual - Anular Serviço 35
7.5 Descrição textual - Análise de Resultados 36
7.6 Descrição textual - Cancelar Actividade 37
7.7 Descrição textual - Cancelar Serviço 38
7.8 Descrição textual - Consultar Histórico 39
7.9 Descrição textual - Consulta de contratos em vigor 40
7.10 Descrição textual - Consulta de contratos finalizados 41
7.11 Descrição textual - Consultar Contratos 42
7.12 Descrição textual - Consultar Saldo 43
7.13 Descrição textual - Editar Actividade 44
7.14 Descrição textual - Gerir Utilizadores 45
7.15 Descrição textual - Inserir Actividade 46
7.16 Descrição textual - Inserir Serviço 47

7.17 Descrição textual - Log In 48
7.18 Descrição textual - Log Out 49
7.19 Descrição textual - Pesquisar por actividade 50
7.20 Descrição textual - Pesquisar por código 51
7.21 Descrição textual - Pesquisar por descrição 52
7.22 Descrição textual - Pesquisar por preço 53
7.23 Descrição textual - Remover Acesso 54
7.24 Descrição textual - Requisitar Actividade 55
7.25 Descrição textual - Requisitar Serviço 56
7.26 Descrição textual - Resultados Globais 57
7.27 Descrição textual - Resultados Parciais 58
7.28 Diagrama de Sequência - Adicionar acesso 60
7.29 Diagrama de Sequência - Alterar comissão 61
7.30 Diagrama de Sequência - Análise de Resultados 62
7.31 Diagrama de Sequência - Anular actividade 63
7.32 Diagrama de Sequência - Anular Serviço 64
7.33 Diagrama de Sequência - Cancelar Actividade 65
7.34 Diagrama de Sequência - Cancelar Serviço 66
7.35 Diagrama de Sequência - Consulta de contratos em vigor 67
7.36 Diagrama de Sequência - Consulta de contratos finalizados 68
7.37 Diagrama de Sequência - Consulta Histórico 69
7.38 Diagrama de Sequência - Consultar Contratos 70
7.39 Diagrama de Sequência - Consultar Saldo 71
7.40 Diagrama de Sequência - Editar Actividade 72
7.41 Diagrama de Sequência - Gerir Utilizadores 73
7.42 Diagrama de Sequência - Inserir Actividade 74
7.43 Diagrama de Sequência - Inserir Serviço 75
7.44 Diagrama de Sequência - Log In 76
7.45 Diagrama de Sequência - Log Out 77
7.46 Diagrama de Sequência - Pesquisar Actividade 78
7.47 Diagrama de Sequência - Pesquisar por código 79
7.48 Diagrama de Sequência - Pesquisar por descrição 80
7.49 Diagrama de Sequência - Pesquisar por preço 81
7.50 Diagrama de Sequência - Remover Acesso 82
7.51 Diagrama de Sequência - Requisitar Actividade 83
7.52 Diagrama de Sequência - Requisitar Serviço 84
7.53 Diagrama de Sequência - Resultados Globais 85
7.54 Diagrama de Sequência - Resultados Parciais 86
7.55 Métodos e Atributos das Classes do package User Interface 87

7.56 Métodos e Atributos das Classes do package GereComSaber 88
7.57 Métodos e Atributos das Classes do package Contratos 88
7.58 Métodos e Atributos das Classes do package Utilizadores 89
7.59 Métodos e Atributos das Classes do package Movimentos 89
7.60 Métodos e Atributos das Classes do package Formas de Pagamento 90
7.61 Métodos e Atributos das Classes do package Sessões 90

Capítulo 1

Introdução

O **UML** (Unified Modeling Language) é uma linguagem de modelação que nos permite visualizar um projecto com base em diagramas padronizados e com forte carga semântica. Projectos desenvolvidos segundo esta perspectiva requerem o levantamento de requisitos junto do cliente e conhecimento da área de negócio na qual o projecto se baseia, antes de modelar o sistema ou mesmo desenvolver a aplicação. A implementação apenas surgirá após a completa modelação do sistema.

É na fase de modelação que surge o **Modelo do Domínio** que não é mais que um glossário com o vocabulário fundamental do contexto do problema. Relacionando esses termos iremos definir classes (e as suas ligações) que serão, com grande probabilidade, entidades ou eventos do negócio. Surgem também os **Use Cases**, que são elementos que traduzem funcionalidades do sistema.

Com a descrição textual dos diversos Use Cases podemos então definir os Diagramas de Sequência de Sistema que posteriormente refinados com os sub-sistemas provenientes do Diagrama de Classes darão lugar aos Diagramas de Sequência de Implementação.

O Diagrama de Classes surge como uma evolução do Modelo do Domínio, partindo dos elementos presentes neste devemos decidir quais serão de facto classes e quais passarão a ser atributos. Podemos também incluir novas classes por motivos de implementação e facilidade.

1.1 Apresentação do Caso de Estudo

Este projecto tem como objectivo o desenvolvimento de uma aplicação informática eficiente que possibilite gerir serviços em condomínios fechados. Esses serviços são prestados por fornecedores e requisitados por clientes. A GereComSaber é uma empresa intermediária na relação cliente-fornecedor e que garante que todos os serviços são prestados e pagos.

Cada fornecedor irá disponibilizar uma lista de serviços, sendo que cada serviço tem a si associado um conjunto de actividades que podem ser contratadas individualmente ou colectivamente. O cliente poderá aceder a essas actividades e escolher as que pretende adquirir. No entanto, esta comunicação terá que se fazer através da aplicação visto que o cliente não possui qualquer contacto com os fornecedores. Por este motivo os pagamentos serão sempre realizados "à cabeça"à GereComSaber, ficando esta responsável por pagar aos fornecedores. Todas as actividades pagas pelo cliente incluem uma comissão, que será o lucro da empresa GereComSaber.

A requisição de novas actividades implica a celebração de novo contrato. Por esse motivo a aplicação terá que gerir os acréscimos no montante pago (mensalmente,trimestralmente,etc) pelo cliente. Por outro lado poderá existir cancelamento de contratos sem prejuízo para o cliente, o que desde de logo implica que haja estornos.

1.2 Objectivos

Na primeira fase construiu-se o **Modelo do Domínio**, bem como o diagrama de **Use Cases**, textualmente definidos.

Na segunda etapa, já é pretendido que apresentemos os **Diagramas de Sequência** criados a partir das descrições textuais dos Use Cases definidos na primeira etapa bem como o **Diagrama de Classes** convenientemente refinado.

Capítulo 2

Modelo do Domínio

O **Modelo do Domínio** tem um papel importante na fase inicial da modelação do projecto. Após a recolha de requisitos, por indicação do cliente e dados relativos a área onde se enquadra o problema, constrói-se um glossário com os termos mais relevantes para o domínio do problema que, normalmente, será analisado quer pelos responsáveis do projecto quer pelo cliente que encomenda o software.

Com este dicionário terminológico poderemos relacionar e organizar as entidades identificadas, assim como os eventos, de forma a obter ligações semânticas entre entidades.

De forma a facilitar a leitura das relações entre entidades (ou eventos) e visualizar o sistema, que através do seu trabalho em conjunto fornece um comportamento cooperativo, existem duas características importantes:

Multiplicidade Indica quantos objectos partilham na relação entre eventos/entidades. Da *Figura 2.1* poderemos retirar que uma (multiplicidade 1) contém uma ou mais (multiplicidade 1..*) actividade e que cada uma destas contém um preço.

Navegação Indica em que sentido pode ser lida e analisada a relação. Da *Figura 2.1* podemos concluir que uma actividade possui um preço mas não poderemos dizer que um preço possui uma actividade.

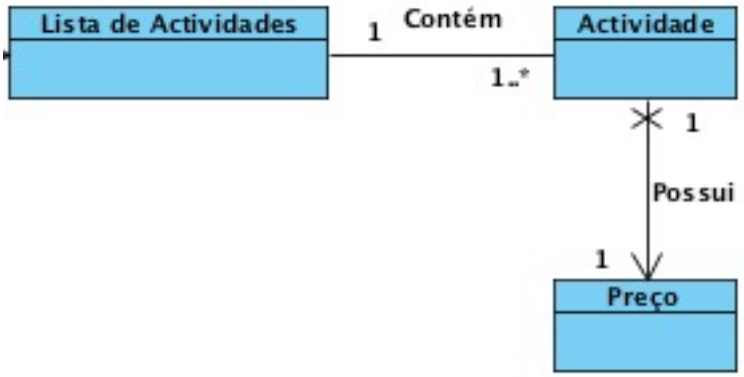


Figura 2.1: Modelo do Domínio - Lista de Actividades

2.1 Glossário

A identificação de eventos ou entidades não é um processo directo, tendo que passar por vários refinamentos numa fase inicial do projecto. Inicialmente foi fácil identificar os actores que interagem com o sistema assim como as entidades mais importantes: contrato, serviços, actividades, etc. Após alterações, algumas delas já depois de ter concebido os **Use Cases**, obteve-se o seguinte glossário:

Fornecedor O actor que presta serviços ao cliente, usando a GereComSaber como intermediário.

Serviço Serviço prestado por um fornecedor (Ex: Jardinagem) que inclui um conjunto de actividades.

Actividade Realização de um trabalho contratado pelo cliente (Ex: corte de relva).

Lista de Actividades Conjunto de actividades.

Preço Valor monetário associado a uma actividade e pago pelo cliente

Carteira de Fornecedores Conjunto de Fornecedores.

Contrato Documento onde consta as actividades escolhidas e o cliente que as requisitou. Cada contrato apenas pode conter um serviço (tendo este múltiplas actividades).

Contratos Caducados Contratos em que a data de término já foi ultrapassada.

Lista de Contratos Lista de contratos (caducados + em vigor).

Valor Valor especificado em cada contrato pelas actividades contratadas.

Forma de Pagamento Modo como será pago as actividades (mensal, trimestral ou semestralmente).

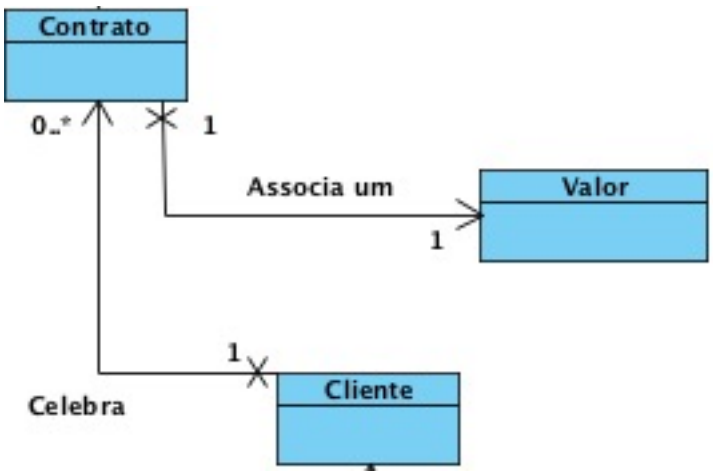


Figura 2.2: Modelo do Domínio - Relação entre Contrato,Cliente e Valor

Comissão Valor acrescentado ao preço das actividades para constar no contrato. Trata-se do lucro da GereComSaber

Cliente Actor que requisita actividades e celebra contratos.

Carteira de Cliente Conjunto de clientes.

2.2 Decisões

Sendo o modelo do domínio o primeiro contacto com o problema e com a área de negócio houve decisões que tiveram que ser tomadas. Algumas vieram-se a revelar erradas e foram corrigidas nos sucessivos refinamentos a que o modelo foi submetido. Outras prevaleceram até a fase actual, entre as quais:

- A carteira de cliente (ou fornecedor) existe mesmo que a GereComSaber não tenha qualquer cliente (ou fornecedor) associado. Daí a relação de 1 (Carteira) para * (fornecedores/clientes).
- A entidade **contrato** são os contratos em vigor, i.e., os que a data de termino é superior a data actual. É por esse motivo que a lista de contratos pode possuir (ou não) a entidade **contrato**.
- Considera-se que não sentido existe uma lista de actividades se não houver serviços.

Deve-se ainda realçar o facto de algumas relações terem sido ocultadas por uma questão de "implicações". A titulo de exemplo vejamos os pagamentos do cliente em relação ao contratos celebrados(*Figura 2.2*). Cada cliente celebra um ou mais contratos e cada contrato tem um valor associado. Por implicação cada cliente terá um ou mais valores associados.
Apesar desta conclusão, não vemos qualquer relação entre as entidades "valor"e "cliente".

2.3 Diagrama do modelo do domínio

Após esta primeira fase e tendo em consideração toda a informação a qual tivemos acesso (via cliente ou através do conhecimento adquirido na área de negócio) obtivemos o modelo presente na *figura 2.3*. Os use cases tiveram também um papel bastante importante, nomeadamente, na parte textual. Muitos dos substantivos usadas na descrição textual deram origem a entidades, restando depois apenas criar as relações existentes.

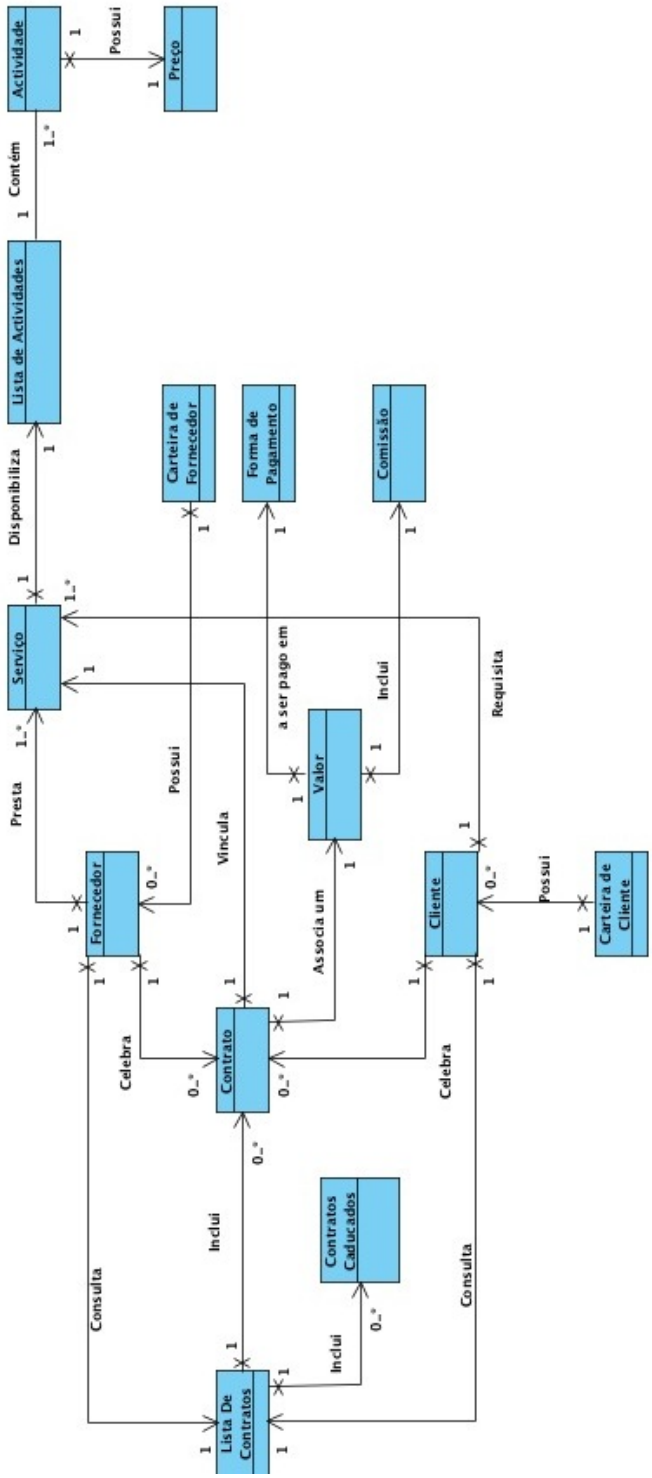


Figura 2.3: Modelo do Domínio

2.4 Alterações na segunda etapa

Para a segunda etapa, com o desenvolvimento dos Diagramas de Sequência e Diagrama de Classes surgiram naturalmente novos elementos que enriqueceram o Modelo relativamente ao que tínhamos realizado para a primeira etapa. Ao realizar os Diagramas de Sequência, mais concretamente ao determinar a divisão do Sistema monolítico em sub-Sistemas causou algumas alterações no Modelo de Domínio.

É fácil de perceber que existem elementos no Modelo de Domínio, os quais, não virão a ser classes por se tratarem de atributos de uma classe, no entanto, pensamos ser importante incluí-los pois estes são também parte relevante do problema bem como são referidos nas descrições textuais dos Use Cases.

O Modelo de Domínio ficou tal como visto na figura 2.4 e foi a partir deste que avançamos para o Diagrama de Classes.

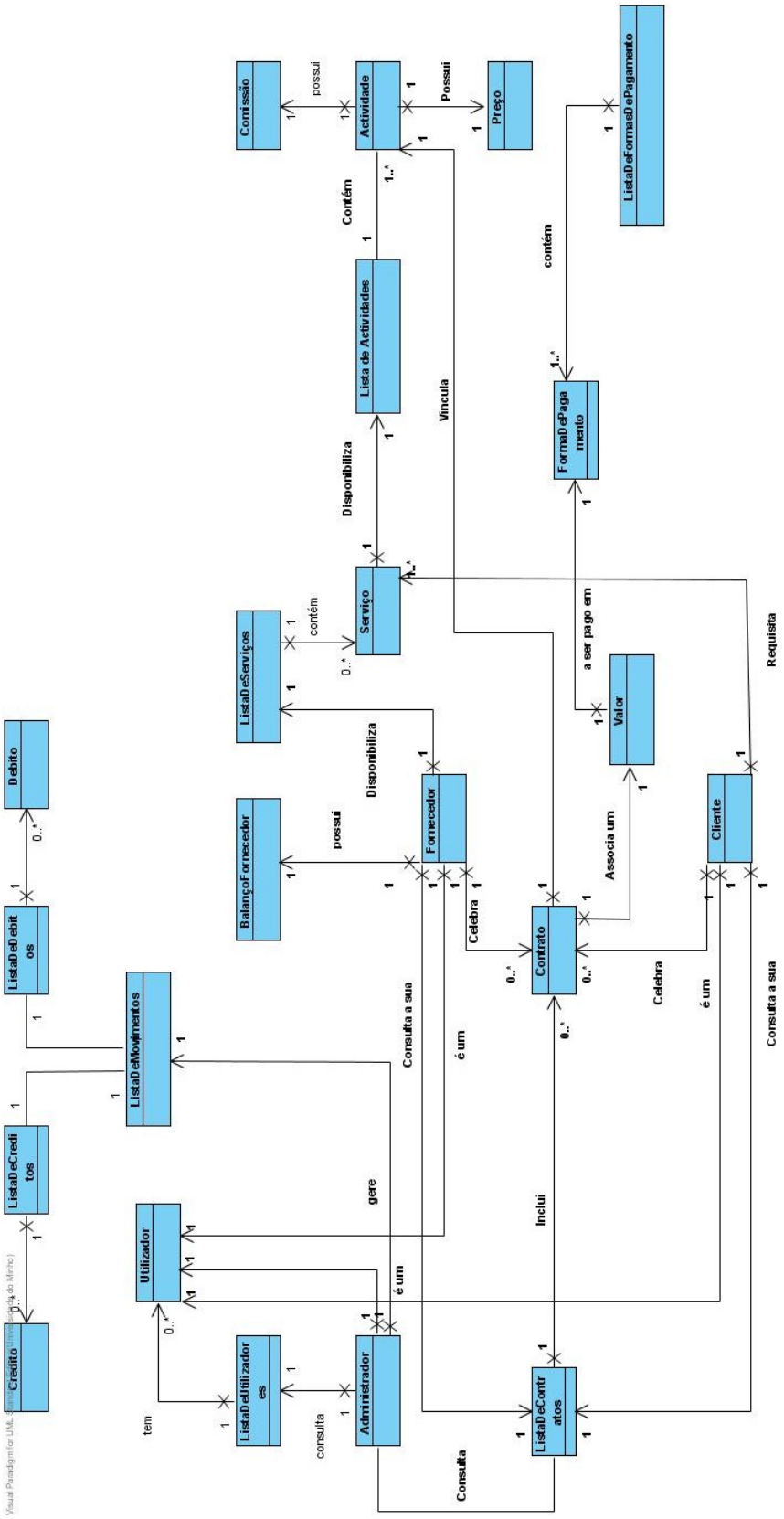


Figura 2.4: Modelo do Domínio obtido na segunda etapa

Capítulo 3

Use Cases

Os **Use Cases** identificam funcionalidades do nosso Sistema, que no seu completo conjunto, identificam todas as funcionalidades da GereComSaber bem como esta se deve comportar perante os diferentes tipos de utilizador. Servem também como forma de diálogo entre o Cliente e os responsáveis pelo desenvolvimento do software. Iremos de seguida falar do nosso diagrama de **Use Cases** subdividindo-os pelos actores que interagem com o Sistema.

3.1 Use Cases multi-utilizador

No nosso diagrama existem **Use Cases** que não são exclusivos de um só actor, estando disponíveis a todos.

3.1.1 Login e Logout

Entre eles surgem os de **Log In** e **Log Out**, que estão associados a todos os actores, estes são importantes para distinguir os diferentes privilégios e funcionalidades a que cada tipo de utilizador tem acesso, sendo que o **Log In** surge como pré-condição para todos os restantes **Use Cases**.

3.1.2 Pesquisar actividade

Todos os utilizadores têm a possibilidade de pesquisar por uma determinada actividade, este **Use Case** surge como uma generalização pois esta pesquisa pode ser efectuada de três formas diferentes:

Pesquisar por preço O utilizador introduz um intervalo de preço, ao qual será devolvida a lista de actividades cujo preço está compreendido nesse intervalo.

Pesquisar por código O utilizador introduz um código, sendo-lhe devolvida a actividade correspondente a esse código (se existir):

Pesquisar por descrição O utilizador introduz uma keyword ou um conjunto de keywords e é-lhe mostrado todas as actividades que na sua descrição esteja(m) a(s) keyword(s) fornecidas.

3.1.3 Consultar contratos

Outro **Use Case** comum a todos os utilizadores é o Consulta de Contratos, o qual surge também como uma generalização, pois damos a possibilidade de efectuar uma pesquisa por contratos em vigor ou por contratos finalizados. Esta separação surge pelo facto de em muitos casos ser muito mais eficaz apresentar apenas um dos casos. Vejamos, como exemplo, o caso em que o administrador após consultar os resultados parciais fica com dúvida em relação ao lucro alcançado. Neste caso é muito mais rápido para o administrador analisar uma lista com contratos finalizados (ou cancelados) do que analisar todos os contratos, visto que os contratos em vigor ainda não afectaram os resultados. O mesmo se passa se quiser saber qual o fornecedor que mais serviços presta no momento (apenas analisa os contratos em vigor). Para além disso é uma forma cómoda e funcional de guardar um histórico de actividades.

3.2 Administrador

Um administrador é qualquer pessoa que tenha privilégios para remover ou adicionar utilizadores e por outro lado, possa ter acesso a informações relativas ao funcionamento da GereComSaber, informações essas confidenciais estando ao alcance de um grupo muito restrito de pessoas como por exemplo as comissões.

3.2.1 Gerir Utilizadores

O administrador tem poder para decidir quem pode aceder ao sistema, podendo adicionar ou remover o acesso de um determinado utilizador. Por exemplo, ao adicionar um novo fornecedor à carteira de fornecedores, este passará a ter acesso ao sistema assim como um fornecedor que seja retirado da carteira de fornecedores deixará de ter acesso ao sistema. Por esta razão, temos aqui uma generalização que se decompõe em Adicionar Acesso e Remover Acesso.

3.2.2 Análise de Resultados

Para analisar os resultados da GereComSaber, o Administrador poderá querer ver os resultados de todo o período de vida da empresa ou apenas num determinado período, imaginemos que por exemplo, poderá querer ver apenas do mês anterior para elaborar um Balancete de Contabilidade. Como a Análise de Resultados, dá duas alternativas ao Administrador surge aqui novamente uma generalização.

3.2.3 Alterar Comissão

O administrador pode eventualmente querer alterar o valor da comissão de uma actividade, para tal, tem esta funcionalidade disponível que lhe permite alterar a comissão, isto vai implicar um aumento do preço para a contratação da actividade por parte do cliente, no entanto, a percentagem correspondente à comissão ganha pela GereComSaber será invisível ao cliente.

3.3 Fornecedor

Um Fornecedor que esteja ligado à GereComSaber disponibiliza uma lista de Serviços que presta, Serviços esses que possuem a si associados uma lista de Actividades. Por exemplo, o Serviço de Jardinagem fornecido pelo fornecedor X pode ter diversas actividades associadas como Cortar Relva, Aparar Cantos e Arranjar Buxos. Já o fornecedor Z, também possuidor do Serviço de Jardinagem já só fornece a opção de Cortar Relva.

3.3.1 Consultar Saldo

De forma a verificar o lucro que está a obter ao prestar os seus serviços aos clientes da GereComSaber e se está de facto a ser compensatório para ele a sua ligação com a GereComSaber, pode consultar o seu saldo que é dado desde que o Fornecedor está ligado à GereComSaber e também do ano corrente.

3.3.2 Editar Actividade

Pode haver necessidade por parte do Fornecedor em alterar um ou mais atributos de uma determinada actividade (exceptuando o código) e a aplicação concede esse privilégio ao Fornecedor.

3.3.3 Anular Serviço

A partir do momento em que um Fornecedor decidir deixar de prestar um determinado serviço tem a possibilidade de o fazer. A este **Use Case** está associado ao Anular Actividade com um estereótipo de include pois um serviço tem uma lista de actividades e se anulamos um serviço, então todas as actividades terão também de ser efectivamente anuladas portanto sempre que o Fornecedor anule um serviço, as actividades também terão sempre de ser anuladas.

3.3.4 Anular Actividade

Esta funcionalidade oferecida ao Fornecedor poderá ser chamada de duas formas distintas. Uma se o Fornecedor decidir explicitamente remover 1 ou mais actividades de um

		Actor Input	System Response
Flow of Events	1		Verifica se foi o fornecedor que activou directamente o Inserir Actividade.
	2	Insere o código do serviço onde quer anexar as novas actividades.	
	3		Certifica-se que o serviço existe.
	4		Disponibiliza os campos da nova actividade para o cliente preencher.
	5	Insere os dados relativos a nova actividade.	
	6	Confirma a inserção da actividade.	
	7		Cria e insere a nova actividade na base de dados.
	8		Questiona o fornecedor para saber se este pretende inserir mais actividades para o actual serviço.
	9	Informa que não quer inserir mais actividades.	
	10		Informa o fornecedor que o processo de inserção de actividade(s) foi concluído com sucesso.

Figura 3.1: Use Case "Inserir Actividade"(Main Flow)

determinado serviço ou então caso decida remover um serviço e aí serão removidas as actividades associadas a esse serviço.

3.3.5 Inserir Serviço

Está disponível para o Fornecedor a opção de inserir um novo serviço, o qual terá também de ter as suas respectivas actividades, esta relação é exprimida com um include porque Inserir Serviço chamará sempre o **Use Case** de Inserir Actividades.

3.3.6 Inserir Actividade

Esta funcionalidade tem uma particularidade quando comparada com as até aqui analisadas. Por um lado poderá ser executada pelo fornecedor de forma directa, i.e., sempre que há interesse de inserir uma nova actividade (num serviço já existente). Por outro lado existe a possibilidade de este Use Case ser executado por invocação do "Inserir actividade", i.e., sempre que o cliente insere um novo serviço e vai anexar actividades.
Na primeira situação não será necessário pedir o código do serviço dentro do "Inserir actividade"(visto que será inserido dentro do novo serviço) enquanto na segunda situação já é necessário. Esta questão de fluxo foi resolvida usando um caminho alternativo, no caso do Use Case "Inserir actividade"ter sido invocado por um outro Use Case, que nos faltar saltar o pedido de código de serviço.

Por outro lado o cliente poderá inserir múltiplas actividades para o mesmo serviço

		Actor Input	System Response
Alternative 1: Não foi o fornecedor que o invocou.	1		Salta para 4 (Main Flow).

		Actor Input	System Response
Alternative 3: O serviço não existe.	1		Informa o fornecedor que o código inserido não está associado a qualquer serviço.
	2		Volta para 2 (Main Flow).

		Actor Input	System Response
Exception 6: O fornecedor cancela.	1		Informa o fornecedor que o processo de inserção de actividade foi cancelado com sucesso.

		Actor Input	System Response
Exception 7: Não foi possível inserir a actividade.	1		Informa o fornecedor que o processo de inserção de actividade não foi concluído, indicando o motivo.

		Actor Input	System Response
Alternative 9: O cliente indica que quer inserir mais actividades.	1		Volta para 4 (Main Flow).

Figura 3.2: Use Case "Inserir Actividade"(Alternative & Exception)

sem que o Use Case tenha que ser invocado novamente. Este facto deve-se ao facto de o sistema questionar, no final de cada inserção, o cliente para saber se este pretende inserir mais actividades. Em caso afirmativo (e devido a um caminho alternativo) o use case voltará a ser invocado, excepção feita para o pedido de código do serviço, visto que será no mesmo anteriormente seleccionado. Estamos, de certa forma, na presença de um ciclo que só acaba quando o cliente acaba as inserções.
Na figura 3.1 é possível ver o fluxo, enquanto que na figura 3.2 visualiza-se os caminhos alternativos

3.4 Cliente

3.4.1 Consulta Histórico

O cliente tem acesso à visualização de todo o seu histórico na GereComSaber, ou seja, pode ver todos os contratos estabelecidos quer estejam em vigor ou já tenham expirado, quanto já gastou em serviços contratados, etc.

3.4.2 Cancelar Serviço

Se um cliente entender já não precisar das actividades contratadas relativas a um serviço, pode fazê-lo com esta funcionalidade. A relação de include com o **Use Case** de Cancelar Actividade está presente, pois, cancelando um serviço, todas as actividades subjacentes terão de ser canceladas também.

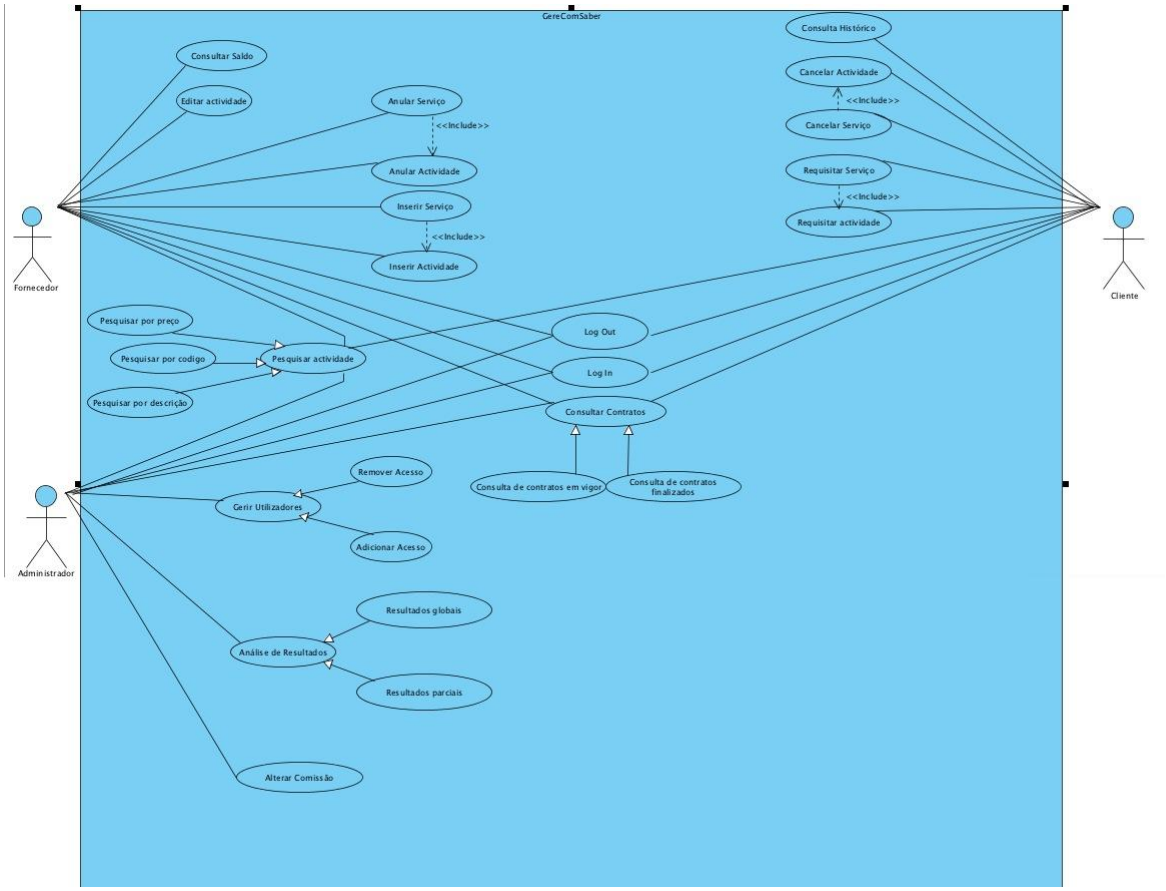


Figura 3.3: Use Cases

3.4.3 Cancelar Actividade

Para além deste **Use Case** poder ser invocado caso um cliente decida cancelar um serviço, outra alternativa seria decidir apenas remover uma ou mais actividades desde que já tenham sido contratadas e que o contrato ainda se mantenha em vigor.

3.4.4 Requisitar Serviço

O cliente pode, através da aplicação, requisitar um novo serviço e ao fazê-lo terá também de escolher as actividades dentro desse serviço que pretende contratar. Esta relação implica que haja uma relação de include com o Requisitar Actividade.

3.4.5 Requisitar Actividade

Para além da situação acima descrita em que o cliente pode chegar a este **Use Case**, o cliente pode também requisitar uma actividade que ainda não tenha contratado sendo em ambos os casos celebrado um novo contrato.

3.5 Alterações na segunda etapa

Não houveram alterações em relação ao diagrama de Use Cases, ou seja, não há funcionalidades novas nem foram retiradas funcionalidades tendo em conta o diagrama construido na primeira etapa.

Por motivos de desconhecimento da forma como se iria processar a entrega da fase 1 em que não seria entregue o .vpp, não incluimos todas as descrições textuais no nosso relatório dessa mesma etapa por forma a não sobrecarregá-lo pensando ser possível consultá-los no ficheiro .vpp correspondente. Portanto, para amenizar o erro causado, vamos desta vez inclui-los a todos na secção de anexos embora estes tivessem sido já definidos antes da deadline da primeira etapa.

Capítulo 4

Diagramas de Sequência

Os Diagramas de Sequência são diagramas que visam a modelação do comportamento de componentes de sistemas representando gráficamente as interações entre objectos através de mensagens que por eles são trocadas mostrando também a sequência temporal na qual é feito este diálogo entre objectos.

Estes diagramas são extraídos das definições textuais dos Use Cases, sendo que a cada Use Case corresponde um Diagrama de Sequência, ou seja, teremos um destes diagramas por cada funcionalidade. Utilizando o método apresentado nas aulas teóricas onde identificamos as transacções numa descrição textual, damos seguimento à passagem para um Diagrama de Sequência no qual tratamos não só do "Main Flow"mas também dos fluxos de Excepção e Alternativa através das Frames do UML disponíveis para o efeito como ref, loop, alt, etc.

A criação destes Diagramas sofre também de sucessivos refinamentos sendo usual partirmos de um Diagrama de Sequência de Sistema onde um actor apenas envia mensagens a uma classe System e nela se colocam todos os métodos necessários para que se tenha a funcionalidade pretendida. No entanto, este System trata-se de um sistema monolítico que é irreal daí que a próxima refinação seja dividir este sistema em sub-sistemas distintos e bem divididos que irão fornecer a funcionalidade pretendida, para a qual são criadas classes que estão usualmente representadas no Modelo de Domínio sendo que neste momento é também criada a ponte para que se possa avançar no Diagrama de Classes. Por último, devemos ainda trocar as mensagem entre os Objectos do Sistema por métodos concretos que farão parte das classes presetes no Diagrama de Classes.

4.1 Decisões

Nos diagramas de Sequência consideramos que o cliente apenas comunica com uma interface do tipo JFrame. É esta que depois terá a responsabilidade de reencaminhar os dados para as restantes classes.

Ao longo do desenvolvimento dos diagramas de sequencia fomos usando a estratégia demonstrada nas figuras 4.1, 4.2 e 4.3 . No entanto, com a experiência adquirida, fomos evoluindo ao ponto de conseguir partir de uma instância semelhante à da 4.2 sem passar pelo Diagrama de Sequência de Sistema (como o da figura 4.1), refinando-a posteriormente.

Usamos em alguns diagramas (muito poucos) o acesso à seguinte mensagem: "Invocado Pelo Utilizador X?", que devolve um boolean e X se refere a um determinado utilizador em concreto. Esta mensagem serve para, por exemplo na funcionalidade "Inserir Actividade", perceber se o pedido foi feito directamente pelo cliente na Interface (carregou no botão "Inserir Actividade") ou se foi através de um outro método (o cliente carregou no botão "Inserir Serviço"que por sua vez fez referência ao "Inserir Actividade") e como estas situações precisam de tratamento ligeiramente diferente há necessidade de as diferenciar para as implementar correctamente.

Todas as referências que ocorrem no diagrama são alusivas aos esteriótipos de "include"dos use cases. Não usamos mais nenhum tipo de referência para outros diagramas. Apesar de facilitar a construção dos diagramas e evitar a duplicação de dados tivemos alguma dificuldade para nesta fase envergar por essa via. Por esse motivo, decidimos fazer um diagrama completo para cada funcionalidade do nosso sistema pois assim fica mais explícito o fluxo completo, e vêmo-lo de uma só vez (sem ter de analisar referências para outros Diagramas de Sequência que estarão noutro local).

4.2 Exemplo do processo de refinamento

O número de refinações nunca é sempre o mesmo sendo este dependente do tipo de Diagrama de Sequência em questão, tendo a ver com a sua complexidade e mesmo com a nossa inspiração no momento que os desenvolvemos. No entanto, foi natural começar-se com um Diagrama de Sequência de Sistema como presente na figura 4.1, onde o Sistema é visto na globalidade e as mensagens são apenas trocadas entre este e o Actor.

Numa próxima refinação é nosso objectivo sub-dividir esse Sistema global em vários sub-sistemas que serão Classes do nosso problema e portanto estando presentes no Diagrama de Classes. Na figura 4.2 essa sub-divisão já foi efectuada tendo nós também já transformado as mensagem em métodos concretos das classes do Diagrama de Classes.

Na figura 4.3 temos o Diagrama de Sequência de Implementação na sua versão final onde refinamos ainda mais em relação ao Diagrama obtido em 4.2, de modo a obter a melhor aproximação à implementação da solução e corrigindo alguns erros ou situação menos concretas ou causadores de dúvida.

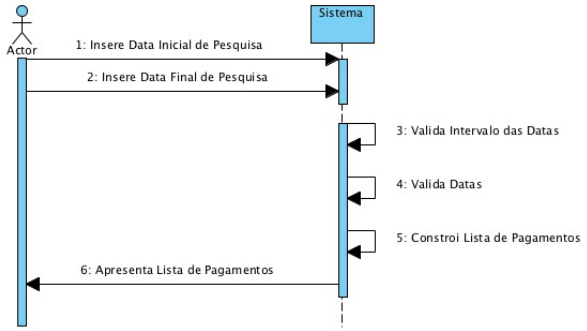


Figura 4.1: Diagrama de Sequência de Sistema para Consultar Histórico

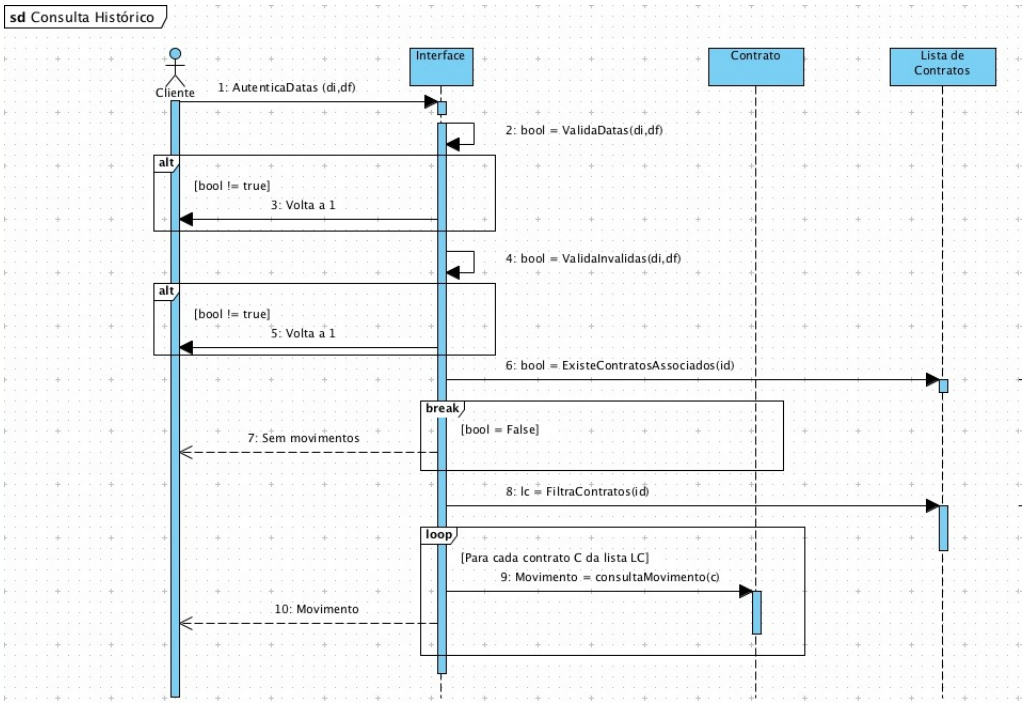


Figura 4.2: Diagrama de Sequência já com Sub-Sistemas para Consultar Histórico

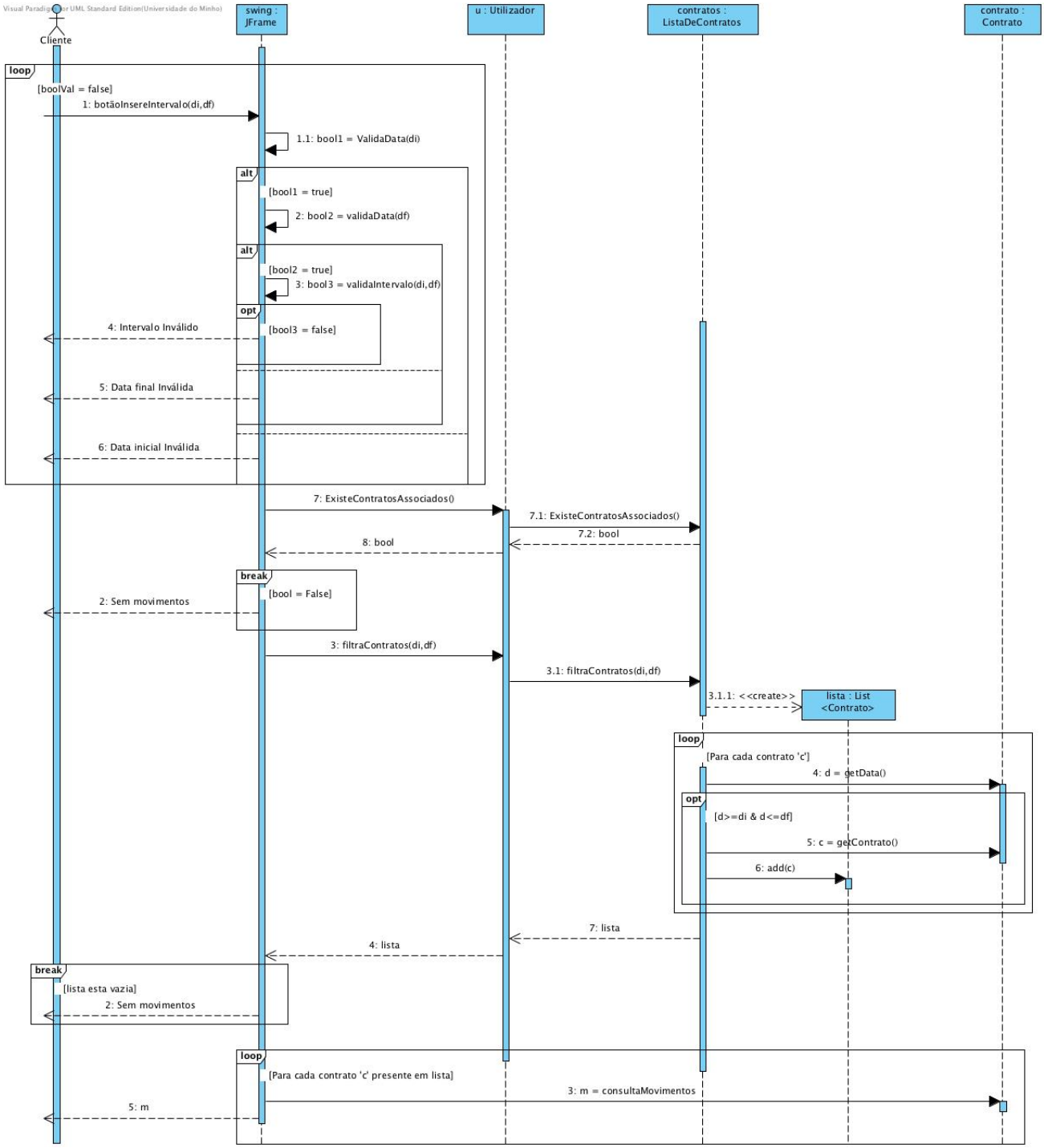


Figura 4.3: Diagrama de Sequência de Implementação após sofrer todas as refinações

Capítulo 5

Diagrama de Classes

O Diagrama de Classes surge como um primeiro diagrama que deixa de ter em conta a captura de requisitos e que tem como objectivo a concepção da Arquitectura bem como a sua implementação. Este Diagrama descreve a estrutura do sistema revelando as suas classes, os atributos e métodos dessas mesmas classes e as relações que as classes estabelecem entre si.

Num diagrama de Classes, cada classe é representada por um rectângulo contendo o nome da classe, os métodos disponíveis bem como os atributos e respectiva visibilidade (package, public, private ou protected). As linhas entre rectângulos indicam relações entre os objectos dessas classes (generalização, dependência, associação, agregação ou composição) as quais também são definidas com a presença de multiplicidade e indicação do seu papel (role).

O começo do desenvolvimento deste tipo de diagrama surge principalmente dos elementos presentes no Modelo do Domínio, no entanto, a refinação dos Diagramas de Sequência com Sub-Sistemas contribui também ela para o aparecimento de novas classes neste diagrama. Pelo facto de os Diagramas de Sequência também traduzirem a troca de mensagens entre Objectos, fica a partir daí fácil de perceber quais os métodos que cada classe irá precisar de ter para que se implementem as funcionalidades do Sistema.

5.1 Decisões

Apenas indicamos os métodos necessários para que se consiga desenvolver a aplicação. Deste forma podemos ter X atributos mas apenas ter um set (visto ser o único em que haja necessidade de mudança) ou um get (porque é o único que necessitamos de aceder).

Grande parte dos métodos irão invocar outros métodos não provocando qualquer alteração nos atributos da classe. Estes métodos surgem apenas numa prespectiva de propagação, ou seja, alcançar dados de outras classes nas quais haja dependência.

Assume-se que ao nível da classe JFrame teremos um objecto do tipo GereComSaber que nos dará acesso directo a:

- Lista de Movimentos: É a partir daqui que temos acesso a todos os movimentos da GereComSaber (sejam estes créditos ou débitos). Esta classe (e respectivos métodos) nunca são usados por nenhum utilizador que não seja o administrador.

- Lista de Utilizadores: Através daqui detectamos todos os utilizadores. Salienta-se tambem que quando fizermos login é aqui que iremos procurar o utilizador em concreto ficando com uma referência para objecto, permintindo depois usá-lo de forma directa, como por exemplo: utilizador.getCodigo();

- Lista de Contratos: Apesar de comum a todos os utilizadores todos os métodos precisam do código do utilizador filtrando apenas os contratos que lhe dizem respeito. Deste modo cada utilizador consegue apenas ver os contratos onde é visado.

- Lista de Formas de Pagamento: Contém as possíveis formas de pagamento.

- Lista de Sessões: Mantém uma lista de nomes de utilizador que estão com sessão iniciada na aplicação, assim como a hora em que efectuaram o login.

Decidimos colocar a Lista de Serviços (e respectivas Actividades) apenas acessível aos utilizadores que sejam Fornecedores, deste modo faremos: utilizador.getServiços() ou utilizador.getActividades(), obtendo os serviços que cada um fornece.

Houve também a preocupação em englobar as nossas Classes em packages de modo a conseguirmos especificar a arquitectura lógica do sistema bem como agrupar as Classes que partilhem um mesmo domínio.

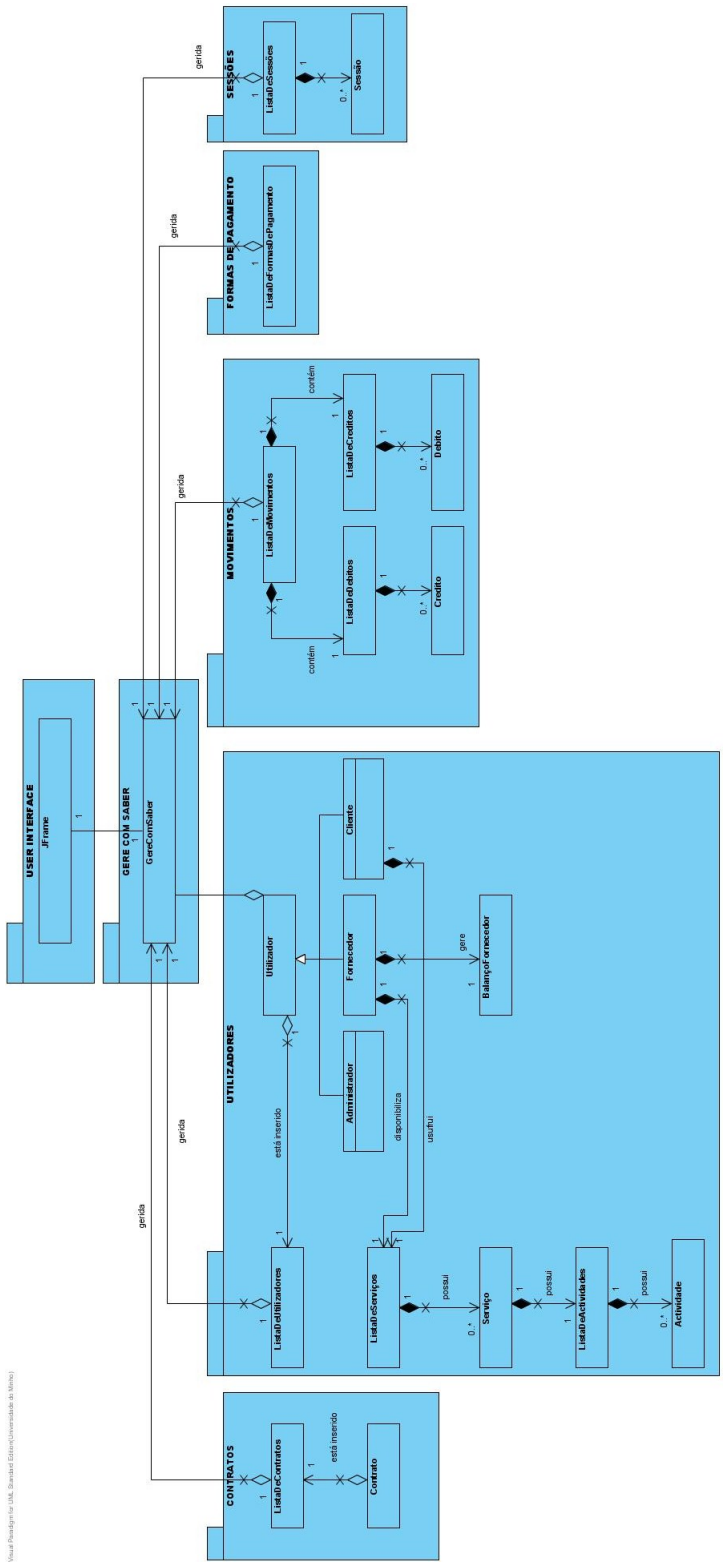


Figura 5.1: Diagrama de Classes (os métodos e atributos foram omitidos)

Capítulo 6

Conclusão

Para este projecto os objectivos são claros e foram, de uma forma geral para estas duas etapas, cumpridos. A única dificuldade passou pelo facto de nunca termos tipo, no seio do grupo, qualquer contacto com o UML ou com este tipo de modelação de sistemas. Nesse sentido as aulas práticas foram uma mais valia, apesar de não serem por si só suficientes havendo a necessidade de grande prática por conta própria.

Podemos, desde de já, tirar ilações em relação ao Modelo do Domínio e aos Use Cases. Em relação ao primeiro concluímos que o mesmo tem uma importância muito elevada no desenvolvimento de um projecto, coisa que não era de facto perceptível antes do inicio do mesmo. A construção de um dicionário de termos, onde muitos deles serão com grande probabilidade (e só agora começamos a tomar conhecimento disso) classes do nosso projecto informático, e as relações entre esse dicionário ajudaram a ganhar uma visão global sobre o sistema. Este último aspecto é relevante na altura da idealização e construção dos Use Cases.

Os Use Cases vêm-nos ajudar na visualização das funcionalidades do sistema e na indicação de quem são as pessoas (actores) que as podem usar. O principal obstáculo foi o facto de termos que garantir um fluxo coerente em Use Cases onde havia pelo menos um **include**, situação onde quer um determinado Use Case fosse invocado directamente pelo utilizador quer fosse invocado por um outro Use Case o fluxo teria que ser sequencialmente correcto. Excepção feita para este ponto o desenvolvimento desta etapa decorreu sem sobressaltos.

O desenvolvimento dos Diagramas de Sequência revelaram ser muito trabalhosos pelo facto de exigirem sucessivas refinações e dado possuirmos um número considerável de Use Cases acabando por tratar-se de uma tarefa muito "time-consuming". No entanto, deu-nos a entender os métodos que serão necessários para implementarmos as funcionalidades vindas dos Use Cases bem como as Classes às quais teremos de enviar as respectivas mensagens.

No que diz respeito ao Diagrama de Classes, surgiu da análise do Modelo do Domínio

e das decisões do que seria necessário ter em termos de Classes para a implementação do projecto bem como os sub-sistemas presentes nos Diagramas de Sequência. A construção e refinação deste conseguiu dar-nos uma referência daquilo a que se podemos chamar os "alicerces" da aplicação da GereComSaber.

Nesta segunda fase com o desenvolvimento dos Diagramas de Sequência bem como o Diagrama de Classes temos já uma ideia mais clara da implementação da aplicação requisitada do que tínhamos em relação à primeira fase. É agora fácil perceber o que devemos implementar bem como a ordem sequencial dos métodos para termos disponíveis as funcionalidades necessárias e presentes no diagrama de Use Cases.

Capítulo 7

Anexos

7.1 Descrições textuais dos Use Cases

Adicionar Acesso - Use Case

Use Case Details

Name:	Adicionar Acesso
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	Gerir Utilizadores
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 19:39:38
Brief Description	O administrador concede o acesso ao sistema a um cliente ou fornecedor.
Preconditions	O administrador efectuou o login e o Super Use Case foi invocado.
Post-conditions	Sucesso: O cliente ou fornecedor possui uma conta para aceder ao sistema Insucesso: Sem alteração do sistema.
Flow of Events	

	Actor Input	System Response
1		Disponibiliza os campos necessários, que devem ser preenchidos, para a criação de novo acesso (ou permissão).
2	Insere os dados para criação de novo acesso.	
3		Verifica dados inseridos.
4		Cria a conta para o utilizador pretendido.

Alternative 3: Dados inválidos.

	Actor Input	System Response
1		Informa que dados sao invalidos.
2		Voltar a 1 (Main Flow).

Figura 7.1: Descrição textual - Adicionar Acesso

Alterar Comissão - Use Case

Use Case Details

Name:	Alterar Comissão
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 16:16:10
Brief Description	O administrador altera a comissão que a GereComSaber recebe ao estabelecer contratos com os clientes.
Preconditions	O administrador efectuou o login.
Post-conditions	Sucesso: A comissão é alterada. Insucesso: A comissão não é alterada.

Flow of Events

	Actor Input	System Response
1	Insere código da actividade sobre a qual pretende alterar a comissão.	
2		Valida código da actividade.
3		Informa administrador da actual comissão da GereComSaber.
4		Pede o valor da nova comissão.
5	Insere a nova comissão (em percentagem).	
6	Confirma nova comissão.	
7		Valida valor da comissão.
8		A comissão é actualizada pelo sistema.

Alternative 2.1: Código inválido

	Actor Input	System Response
1		Informa administrador que o código é inválido.
2		Voltar a 1 (Main Flow).

Alternative 2.2: Código inexistente.

	Actor Input	System Response
1		Informa administrador que o código inserido não está associado a nenhuma actividade.
2		Voltar a 1 (Main Flow).

Exception 6: Administrador não confirma

	Actor Input	System Response
1		A comissão não é alterada e o processo é abortado.

Alternative 7: Comissão inválida.

	Actor Input	System Response
1		Informa administrador que a comissão introduzida não é valida.
2		Voltar a 5 (Main Flow).

Figura 7.2: Descrição textual - Alterar Comissão

Anular Actividade - Use Case

Use Case Details

Name:	Anular Actividade
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 18:17:23
Brief Description	O fornecedor pode anular uma, ou mais, actividade(s) sempre que deixe de disponibiliza-la(s).
Preconditions	O fornecedor já executou o login.
Post-conditions	Sucesso: Foram anuladas todas as actividades pretendidas. Insucesso: Não foram anuladas quaisquer actividades. Sem alteração do sistema.

Flow of Events

	Actor Input	System Response
1		Verifica se foi o fornecedor que activou directamente o Anular Actividade.
2		Apresenta a lista de actividades em vigor.
3	Selecciona a(s) actividade(s) que pretende anular.	
4	Confirma o pedido de anulação.	
5		Remove as actividades.
6		Notifica o fornecedor que a anulação de actividades foi efectuado com sucesso.

Alternative 1: Não foi o fornecedor que o invocou.

	Actor Input	System Response
1		Saltar para 4 (Main Flow).

Exception 2: Não existe serviços para anular.

	Actor Input	System Response
1		Informa o fornecedor da inexistência de actividades para anular.

Exception 4: O fornecedor não confirma.

	Actor Input	System Response
1		Informa o fornecedor que o pedido de anulação não foi executado.

Figura 7.3: Descrição textual - Anular Actividade

Anular Serviço - Use Case

Use Case Details

Name:	Anular Serviço
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 18:10:57
Brief Description	O fornecedor pode a qualquer momento anular qualquer serviço que possua.
Preconditions	O fornecedor já efectuou o login.
Post-conditions	Sucesso: O sistema anulou o serviço pretendido. Insucesso: Sem anulação de serviço(s). O sistema não se altera.

Flow of Events

	Actor Input	System Response
1		Apresenta a lista de serviços actualmente em vigor.
2	Selecciona o(s) serviço(s) que pretende anular.	
3	Confirma o pedido de anulação.	
4		Identifica quais as actividades associadas ao serviço seleccionado.
5		Include: Anular actividade.
6		Informa o fornecedor que a anulação foi processada com sucesso.

Exception 1: Não existe serviços para anular.

	Actor Input	System Response
1		Informa o fornecedor da inexistência de serviços para anular.

Exception 4: Fornecedor cancela pedido de anulação.

	Actor Input	System Response
1		Envia a confirmação da anulação do serviço ao fornecedor.

Figura 7.4: Descrição textual - Anular Serviço

Análise de Resultados - Use Case

Use Case Details

Name:	Análise de Resultados
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main	
Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 15:51:37
Brief Description	O administrador pretende ver os resultados financeiros da GereComSaber num período por ele inserido ou na totalidade do período de vida da GereComSaber.
Preconditions	O administrador efectuou login.
Post-conditions	Um dos sub-Use Cases é invocado.
Flow of Events	

	Actor Input	System Response
1	Insere qual a opção que pretende.	
2		Encaminha a escolha do administrador para o sub-Use Case respectivo.

Figura 7.5: Descrição textual - Análise de Resultados

Cancelar Actividade - Use Case

Use Case Details

Name:	Cancelar Actividade
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 10:59:41
Brief Description	Sempre que o cliente sente a necessidade de cancelar uma(s) actividade(s) de um (ou mais) serviço(s) poderá faze-lo. Neste caso será reembolsado num montante referente as actividades não prestadas.
Preconditions	O cliente já efectuou o login.
Post-conditions	Sucesso: Foram removidas todas as actividades pretendidas. Insucesso: Não foram removidas quaisquer actividades. Sem alteração do sistema.

Flow of Events

	Actor Input	System Response
1		Analisa se foi o cliente que activou directamente o cancelamento de actividades.
2		Apresenta a lista de actividades actualmente contratadas.
3	Selecciona a(s) actividade(s) que pretende cancelar.	
4		Calculo do estorno financeiro.
5		Indica ao cliente o montante que irá receber após o cancelamento.
6	Confirma o pedido de cancelamento.	
7		Cancela todas as actividades seleccionadas.
8		Procede ao reembolso (Calculado em 4).
9		Notifica o cliente que o cancelamento de actividades foi efectuado com sucesso.

Alternative 1: Não foi o cliente que invocou.

	Actor Input	System Response
1		Saltar para 4 (Main Flow).

Exception 2: Não existe serviços para cancelar.

	Actor Input	System Response
1		Informa o cliente da inexistência de actividades para cancelar.

Exception 6: O cliente anula o pedido de cancelamento.

	Actor Input	System Response
1		Informa o cliente que o pedido de cancelamento foi anulado.

Figura 7.6: Descrição textual - Cancelar Actividade

Cancelar Serviço - Use Case

Use Case Details

Name:	Cancelar Serviço
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main	
Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 11:38:20
Brief Description	O cliente poderá optar por cancelar um serviço a qualquer momento, tendo direito a um estorno financeiro relativo aos dias/meses em que não usufrui do serviço.
Preconditions	O cliente já executou o login.
Post-conditions	Sucesso: O cliente cancelou o serviço pretendido. Insucesso: Sem cancelamento de serviços. O sistema não é alterado.

Flow of Events

	Actor Input	System Response
1		Apresenta a lista de serviços actualmente contratados.
2	Selecciona o(s) serviço(s) que pretende cancelar.	
3		Identifica quais as actividades associadas ao(s) serviço(s) seleccionado(s).
4		include: Cancelar actividade.
5		Informa o cliente que o cancelamento foi processado com sucesso.

Exception 1: Não existe serviço(s) para cancelar.

	Actor Input	System Response
1		Informa o cliente da inexistência de serviços para cancelar.

Figura 7.7: Descrição textual - Cancelar Serviço

Consulta Histórico - Use Case

Use Case Details

Name:	Consulta Histórico
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main	
Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 11:19:34
Brief Description	Todos os serviços e respectivas actividades contratadas pelo cliente são pagas. Os pagamentos podem ser consultados fazendo uma consulta do histórico de pagamentos.
Preconditions	O cliente já executou o login.
Post-conditions	Sucesso: Será apresentada uma lista com a data do pagamento, montante e serviço/actividade a que diz respeito. Insucesso: Não é apresentada lista de pagamentos.

Flow of Events

	Actor Input	System Response
1	Selecciona a data inicial e final que pretende para a pesquisa.	
2		Valida ambas as datas.
3		Valida o intervalo inserido.
4		Testa se existe contratos.
5		Constrói uma lista com todos os pagamentos realizados no intervalo escolhido.
6		Apresenta a lista ao cliente.

Alternative 3: Data inicial é mais recente que data final.

	Actor Input	System Response
1		Informa o cliente que o intervalo é inválido.
2		Volta a 1 (Main Flow).

Alternative 2: Datas inválidas.

	Actor Input	System Response
1		Informa o cliente que as datas são inválidas.
2		Volta a 1 (Main Flow).

Exception 4: Sem Pagamentos.

	Actor Input	System Response
1		Não é apresentada qualquer lista e o cliente é informado que não existe pagamentos entre as referidas datas.

Figura 7.8: Descrição textual - Consultar Histórico

Consulta de contratos em vigor - Use Case

Use Case Details

Name:	Consulta de contratos em vigor
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	Consultar Contratos
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 18:21:28
Brief Description	O utilizador pesquisa todos os contratos que estejam em vigor.
Preconditions	O utilizador está autenticado no sistema, seja como fornecedor, cliente ou como administrador.
Post-conditions	Sucesso: O utilizador possui uma lista dos contratos em vigor. Insucesso: Não é apresentado qualquer resultado. Sem alterações do sistema.
Flow of Events	

	Actor Input	System Response
1		Pesquisa todos os contratos que estejam em vigor.
2		Filtra apenas os contratos onde o utilizador é visado.
3		E apresentada ao utilizador a lista dos contratos.

Exception 3: Não existem contratos em vigor.

	Actor Input	System Response
1		Informa o utilizador de que não existem contratos em vigor naquele momento.

Alternative 2: Não há filtragem pois trata-se do administrador.

	Actor Input	System Response
1		Salta para 3 (Main Flow)

Figura 7.9: Descrição textual - Consulta de contratos em vigor

Consulta de contratos finalizados - Use Case

Use Case Details

Name:	Consulta de contratos finalizados
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	Consultar Contratos
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 18:26:54
Brief Description	O utilizador pesquisa contratos acabados num determinado intervalo de tempo.
Preconditions	O utilizador está autenticado no sistema, seja como fornecedor ou como administrador.
Post-conditions	Sucesso: O utilizador possui uma lista dos contratos acabados no intervalo de tempo pretendido. Insucesso: Não é apresentado qualquer resultado. Sem alterações do istema.

Flow of Events

	Actor Input	System Response
1	Introduz o limite inferior e superior do intervalo de tempo.	
2		Valida a data introduzida.
3		Valida intervalo temporal.
4		São pesquisados todos os contratos que tenham finalizado entre as duas datas introduzidas.
5		Filtra apenas os contratos onde o utilizador é visado.
6		São apresentados todos os contratos findos que obedeçam ao critério de pesquisa.

Alternative 2.1: Foi introduzida uma data superior a data actual.

	Actor Input	System Response
1		Informa o utilizador de que a data inserida é uma data futura.
2		Volta a 1 (Main flow).

Alternative 2.2: Data inválida.

	Actor Input	System Response
1		Informado o utilizador de que a data é inválida.
2		Volta a 1 (Main flow).

Alternative 3: A segunda data introduzida é anterior à primeira.

	Actor Input	System Response
1		O sistema informa o utilizador de que a segunda data é anterior à primeira, criando um intervalo de tempo inválido.
2		Volta a 1 (Main flow).

Exception 6: Não existem contratos finalizados no intervalo de tempo pretendido.

	Actor Input	System Response
1		O utilizador é informado de que não existem contratos acabados entre as datas seleccionadas.

Alternative 5: Não há filtragem pois trata-se do administrador.

	Actor Input	System Response
1		Salta para 6 (Main Flow).

Figura 7.10: Descrição textual - Consulta de contratos finalizados

Consultar Contratos - Use Case

Use Case Details

Name:	Consultar Contratos
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main	
Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 17:33:36
Brief Description	O utilizador poderá pesquisar os contratos em vigor ou finalizados. Só serão visualizados os contratos directamente relacionados com o utilizador em causa (excepção feita para o administrador que visualiza todos os contratos).
Preconditions	O utilizador está autenticado no sistema, seja como fornecedor, cliente ou como administrador.
Post-conditions	Um dos sub-Use Cases é efectuado.
Flow of Events	

	Actor Input	System Response
1	O utilizador escolhe se pretende visualizar os contratos em vigor ou acabados.	
2		O sistema encaminha o utilizador para a funcionalidade pretendida.

Figura 7.11: Descrição textual - Consultar Contratos

Consultar Saldo - Use Case

Use Case Details

Name:	Consultar Saldo
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main	
Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 16:32:41
Brief Description	Todo o fornecedor tem um saldo a si associado que pode consultar, i.e., sempre que um pagamento é feito ele será reflectido no saldo do ano corrente e no saldo total da actividade do fornecedor.
Preconditions	O fornecedor já executou login.
Post-conditions	Será apresentado o saldo correspondente ao ano corrente e o saldo total do fornecedor (valores proveniente da GereComSaber).
Flow of Events	

	Actor Input	System Response
1		Calcula o saldo correspondente ao ano corrente.
2		Calcula o saldo total desde do inicio da actividade do fornecedor.
3		Apresenta os saldos ao fornecedor.

Figura 7.12: Descrição textual - Consultar Saldo

Editar actividade - Use Case

Use Case Details

Name:	Editar actividade
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 16:38:42
Brief Description	O fornecedor possui actividades a si associadas que contém diversas informações. Estas podem ser editadas, pelo fornecedor, a qualquer momento.
Preconditions	O fornecedor já executou o login.
Post-conditions	Sucesso: O sistema irá editar os dados que o fornecedor indicou. Insucesso: O sistema não se altera.

Flow of Events

	Actor Input	System Response
1	Insere o código da actividade.	
2		Valida o código da actividade.
3		Recolhe os dados associados a actividade.
4		Apresenta os dados ao fornecedor para este edita-los.
5	Edita os dados que pretende.	
6		Valida os novos dados.
7		Procede a alteração de dados na actividade.
8		Notifica que a alteração foi bem sucedida.

Alternative 2.1: Código inválido.

	Actor Input	System Response
1		Informa o fornecedor que o código que introduziu não é valido.
2		Volta a 1 (Main Flow).

Alternative 2.2: Código sem actividade associado.

	Actor Input	System Response
1		Informa o fornecedor que o código que introduziu não corresponde a qualquer actividade.
2		Volta a 1 (Main Flow).

Alternative 6: Existem dados inválidos.

	Actor Input	System Response
1		Informa o fornecedor da existência de dados inválidos.
2		Volta a 5 (Main Flow).

Figura 7.13: Descrição textual - Editar Actividade

Gerir Utilizadores - Use Case

Use Case Details

Name:	Gerir Utilizadores
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main	
Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 15:48:28
Brief Description	O administrador opta por criar conta ou remover uma já existente
Preconditions	O administrador já fez login
Post-conditions	Um dos sub-Use Cases é efectuado
Flow of Events	

	Actor Input	System Response
1	Especifica se pretende criar conta ou remover uma conta.	
2		Encaminha o administrador para a funcionalidade pretendida.

Figura 7.14: Descrição textual - Gerir Utilizadores

Inserir Actividade - Use Case

Use Case Details

Name:	Inserir Actividade
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 19:38:56
Brief Description	Sempre que surja(m) nova(s) actividade(s) para um serviço o fornecedor poderá inseri-la(s).
Preconditions	O fornecedor já efectuou login.
Post-conditions	Sucesso: Foram inseridas todas as actividades pretendidas. Insucesso: Não foram inseridas quaisquer actividades. Sem alteração do sistema.

Flow of Events

	Actor Input	System Response
1		Verifica se foi o fornecedor que activou directamente o Inserir Actividade.
2	Insere o código do serviço onde quer anexar as novas actividades.	
3		Certifica-se que o serviço existe.
4		Disponibiliza os campos da nova actividade para o cliente preencher.
5	Insere os dados relativos a nova actividade.	
6	Confirma a inserção da actividade.	
7		Cria e insere a nova actividade na base de dados.
8		Questiona o fornecedor para saber se este pretende inserir mais actividades para o actual serviço.
9	Informa que não quer inserir mais actividades.	
10		Informa o fornecedor que o processo de inserção de actividade(s) foi concluído com sucesso.

Alternative 1: Não foi o fornecedor que o invocou.

	Actor Input	System Response
1		Salta para 4 (Main Flow).

Alternative 3: O serviço não existe.

	Actor Input	System Response
1		Informa o fornecedor que o código inserido não está associado a qualquer serviço.
2		Volta para 2 (Main Flow).

Exception 6: O fornecedor cancela.

	Actor Input	System Response
1		Informa o fornecedor que o processo de inserção de actividade foi cancelado com sucesso.

Alternative 9: O cliente indica que quer inserir mais actividades.

	Actor Input	System Response
1		Volta para 4 (Main Flow).

Figura 7.15: Descrição textual - Inserir Actividade

Inserir Serviço - Use Case

Use Case Details

Name:	Inserir Serviço
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 19:30:02
Brief Description	Sempre que o fornecedor passe a disponibilizar um novo serviço poderá inserir o serviço no sistema.
Preconditions	O fornecedor já efectuou o login.
Post-conditions	Sucesso: O fornecedor inseriu um novo serviço. Insucesso: O fornecedor não inseriu um novo serviço. O sistema não é alterado.

Flow of Events

	Actor Input	System Response
1		Disponibiliza os campos referentes ao novo serviço, que devem ser preenchidos pelo fornecedor.
2	Insere os dados relativos ao novo serviço.	
3		Valida os dados.
4	O fornecedor confirma a inserção do serviço.	
5		Insere novo serviço no sistema
6		Include: Inserir Actividade.
7		Confirma se houve inserção de novas actividades.
8		Informa o fornecedor que o novo serviço, assim como a(s) nova(s) actividade(s), estão agora disponíveis.

Exception 4: Fornecedor cancela pedido de inserção.

	Actor Input	System Response
1		Notifica o fornecedor que a inserção de serviço foi cancelada.

Alternative 3: Dados invalidos

	Actor Input	System Response
1		Notifica o cliente que introduziu dados invalidos
2		Volta a 1 (main flow).

Figura 7.16: Descrição textual - Inserir Serviço

Log In - Use Case

Use Case Details

Name:	Log In
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 14:39:35
Brief Description	Todo o utilizador para entrar no sistema tem de ter permissao para tal, ou seja, tem de estar logged in.
Preconditions	
Post-conditions	Sucesso: O utilizador faz log in. Insucesso: O utilizador nao faz log in.
Flow of Events	

	Actor Input	System Response
1	Insere os dados (username e password).	
2		Valida os dados.
3		Da' permissoes de acesso ao utilizador.

Alternative 2: Dados invalidos ou inexistentes

	Actor Input	System Response
1		Informa o utilizador que os dados que introduziu sao invalidos ou inexistentes.
2		Saltar para 1 (Main Flow).

Figura 7.17: Descrição textual - Log In

Log Out - Use Case

Use Case Details

Name:	Log Out
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main	
Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 14:30:36
Brief Description	Todo o actor tem de fazer logout para sair da aplicaçao.
Preconditions	O actor tem de estar logged in.
Post-conditions	O actor tem de estar logged out.
Flow of Events	

	Actor Input	System Response
1		Retira as permissoes de acesso ao utilizador.

Figura 7.18: Descrição textual - Log Out

Pesquisar actividade - Use Case

Use Case Details

Name:	Pesquisar actividade
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 15:51:14
Brief Description	O utilizador efectua uma pesquisa sobre todas as actividades disponíveis, podendo fazê-lo por preço, código ou descrição.
Preconditions	O utilizador está autenticado no sistema, seja como fornecedor, como administrador ou cliente.
Post-conditions	Um dos sub-Use Cases é efectuado.
Flow of Events	

	Actor Input	System Response
1	Especifica qual o critério que pretende para efectuar a procura	
2		Encaminha o utilizador para a funcionalidade correspondente

Figura 7.19: Descrição textual - Pesquisar por actividade

Pesquisar por codigo - Use Case

Use Case Details

Name:	Pesquisar por codigo
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	Pesquisar actividade
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 16:01:05
Brief Description	O utilizador pesquisa uma actividade cujo código corresponda ao introduzido.
Preconditions	O utilizador está autenticado no sistema, seja como fornecedor, como administrador ou cliente.
Post-conditions	O utilizador possui a informação sobre a actividade com o código especificado se ela existir.

Flow of Events

	Actor Input	System Response
1	Introduz o código a pesquisar.	
2		Valida o código introduzido.
3		Verifica se existe alguma actividade com o código especificado.
4		São consultados todos os dados sobre a actividade.
5		Apresenta os dados da actividade ao utilizador.

Alternative 2: O código introduzido não é válido.

	Actor Input	System Response
1		Informa o utilizador de que o código introduzido não é válido.
2		Volta a 1 (Main flow).

Alternative 3: Não existe nenhuma actividade com o código introduzido.

	Actor Input	System Response
1		Informa o utilizador de que não existe nenhuma actividade identificada pelo código que introduziu.
2		Volta a 1 (Main flow).

Figura 7.20: Descrição textual - Pesquisar por código

Pesquisar por descrição - Use Case

Use Case Details

Name:	Pesquisar por descrição
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	Pesquisar actividade
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 16:35:58
Brief Description	O utilizador pesquisa uma actividade cuja descrição corresponda, parcial ou totalmente, com o texto introduzido.
Preconditions	O utilizador está autenticado no sistema, seja como fornecedor, como administrador ou cliente.
Post-conditions	Sucesso: O utilizador possui uma lista das actividades cuja descrição correspondam ao introduzido. Insucesso: Não é apresentada qualquer lista. Sem alteração do sistema.

Flow of Events

	Actor Input	System Response
1	O utilizador introduz o pedaço de texto a pesquisar	
2		O sistema valida o texto introduzido.
3	O utilizador selecciona se pretende resultados que contenham o texto introduzido ou que sejam exactamente iguais.	
4		O sistema recolhe opção seleccionada e, com essa informação. efectua a pesquisa de todas as actividades afectadas.
5		São apresentadas ao utilizador todas as actividades que obedeçam ao critério especificado.

Alternative 2:

	Actor Input	System Response
1		O sistema informa o utilizador de que deve introduzir algo no campo de pesquisa para esta poder ser efectuada ou retirar os caracteres inválidos.
2		Volta a 1 (Main flow).

O texto introduzido é vazio ou possui caracteres inválidos.

Alternative 5: Não existem actividades que satisfaçam os critérios.

	Actor Input	System Response
1		O utilizador é informado de que não existem actividades que satisfaçam a descrição introduzida.
2		Volta a 1 (Main flow)

Figura 7.21: Descrição textual - Pesquisar por descrição

Pesquisar por preço - Use Case

Use Case Details

Name:	Pesquisar por preço
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	Pesquisar actividade
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 15:53:51
Brief Description	O utilizador pesquisa uma actividade cujo preço esteja contido num intervalo especificado.
Preconditions	O utilizador está autenticado no sistema, seja como fornecedor, como administrador ou cliente.
Post-conditions	Sucesso: O utilizador possui uma lista de todas as actividades que se encaixem na faixa de preços pretendida. Insucesso: Não é apresentada qualquer lista. Sem alteração do sistema.
Flow of Events	

	Actor Input	System Response
1	Insere o valor mínimo e máximo para o preço.	
2		Sistema valida os preços introduzidos.
3		São pesquisadas todas as actividades cujo preço encaixem no intervalo introduzido.
4		E apresentada ao utilizador uma lista das actividades resultantes da pesquisa.

Alternative 2.1: Valor introduzido é negativo ou inválido.

	Actor Input	System Response
1		O utilizador é informado de que o valor introduzido não é válido.
2		Volta a 1 (Main flow).

Alternative 2.2: Valor máximo é menor do que o valor mínimo.

	Actor Input	System Response
1		O utilizador é informado de que o preço máximo que introduziu é superior ao preço mínimo e é informado que terá de introduzir novo par de valores.
2		Volta a 1 (Main flow).

Exception 4: Não existem actividades com preços situados no intervalo pretendido.

	Actor Input	System Response
1		O utilizador é informado de que não existem actividades que correspondam aos critérios de pesquisa.

Figura 7.22: Descrição textual - Pesquisar por preço

Remover Acesso - Use Case

Use Case Details

Name:	Remover Acesso
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	Gerir Utilizadores
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 15:22:48
Brief Description	O administrador remove o acesso ao sistema (permissão) a um cliente ou fornecedor.
Preconditions	Administrador já efectuou o login e o Super Use Case foi invocado.
Post-conditions	Sucesso: O utilizador escolhido perdeu o acesso ao sistema Insucesso: Não há alterações do sistema.

Flow of Events

	Actor Input	System Response
1	Escolhe o tipo de utilizador (fornecedor ou cliente) ao qual pretende remover permissões.	
2		Fornece a lista de utilizadores (mediante a escolha feita em 1) com acesso ao sistema.
3	Selecciona o(s) utilizadores que irão perder as permissões.	
4		O(s) utilizador(es) perde(m) o acesso ao sistema.
5		Informa o cliente que a remoção de utilizador(es) foi concluída com sucesso.

Exception 2: Nao existe utilizadores do tipo selecionado

	Actor Input	System Response
1		Avisa o administrador que não existe utilizadores do tipo seleccionado.

Figura 7.23: Descrição textual - Remover Acesso

Requisitar actividade - Use Case

Use Case Details

Name:	Requisitar actividade
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 12:48:18
Brief Description	O cliente poderá requisitar uma nova actividade de um dado serviço sempre que pretender.
Preconditions	O cliente já efectuou o login.
Post-conditions	Sucesso: O cliente requisita uma nova actividade. Um novo contrato é criado. Insucesso: Não existe nova actividade requisitada. O sistema não se altera.

Flow of Events

	Actor Input	System Response
1		Analisa se foi o cliente que activou directamente a requisição de uma nova actividade.
2		Identifica as actividades existentes (e não contratadas) nos serviços que o cliente possui.
3		Apresenta as actividades identificadas.
4	Selecciona a(s) nova(s) actividade(s) que pretende requisitar.	
5		Calcula o montante todo a ser pago pelas novas actividades.
6		Apresenta o valor a pagar ao cliente.
7		Apresenta a possibilidade de pagamento.
8	Selecciona o tipo de pagamento.	
9		Apresenta um resumo do novo contrato (valor e data de pagamentos, serviço, actividade, etc).
10	Valida as condições do novo contrato.	
11		Procede a realização do novo contrato.
12		Adiciona o valor+comissao aos creditos da GereComSaber
13		Adiciona o valor aos debitos da GereComSaber.
14		Adiciona o valor aos creditos do fornecedor
15		Indica ao cliente que as novas actividades foram contratadas com sucesso.

Alternative 1: Não foi o cliente que activou.

	Actor Input	System Response
1		Saltar para 5 (Main Flow).

Exception 2: O cliente não tem mais actividades disponíveis para requisitar.

	Actor Input	System Response
1		Informa o cliente da inexistência de actividades para requisitar.

Alternative 10: O cliente rejeita as condições.

	Actor Input	System Response
1		Saltar para 4 (Main Flow).

Figura 7.24: Descrição textual - Requisitar Actividade

Requisitar Serviço - Use Case

Use Case Details

Name:	Requisitar Serviço
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 12:42:00
Brief Description	O cliente poderá requisitar um novo serviço sempre que pretenda.
Preconditions	O cliente já efectuou o login.
Post-conditions	Sucesso: O cliente requisitou um novo serviço. Insucesso: O cliente não requisitou novos serviços. O sistema não se altera.

Flow of Events

	Actor Input	System Response
1		Identifica os serviços que o cliente ainda não possui.
2		Apresenta esses serviços ao cliente.
3	Selecciona o(s) serviço(s) que pretende.	
4		Identifica e apresenta ao cliente as actividades associadas ao(s) serviço(s) escolhido(s), assim como os preços.
5	Selecciona as actividades que pretende contratar.	
6		Include: Requisitar actividade.

Exception 1: O cliente já possui todos os serviços.

	Actor Input	System Response
1		Informa o cliente que já possui todos os serviços disponiveis.

Figura 7.25: Descrição textual - Requisitar Serviço

Resultados globais - Use Case

Use Case Details

Name:	Resultados globais
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main	
Super Use Case	Análise de Resultados
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 15:57:46
Brief Description	O administrador prentende consultar os resultados financeiros de todo o período de vida da GereComSaber e o sistema devolve-lhe a lista com esses resultados
Preconditions	O administrador já efectuou login e o Super Use Case foi invocado.
Post-conditions	Sucesso: Apresentação da lista de créditos e débitos financeiros. Insucesso: Não é apresentada qualquer lista.
Flow of Events	

	Actor Input	System Response
1		Identifica movimentos financeiros (débitos e créditos).
2		Agrupa os resultados e mostra-os ao utilizador.

Exception 2: Não existe resultados.		Actor Input	System Response
	1		Informa o cliente que não existem movimentos.

Figura 7.26: Descrição textual - Resultados Globais

Resultados parciais - Use Case

Use Case Details

Name:	Resultados parciais
Rank:	Unspecified
Justification:	
Leaf:	false
Root:	false
Stereotype:	UseCase
Abstract:	false
Documentation:	

Flow of Events

Use Case Descriptions

Main

Super Use Case	Análise de Resultados
Author	Eduardo, Paulo, Tiago, André
Date	24/Nov/2009 16:04:22
Brief Description	O Administrador pretende ver os resultados financeiros da GereComSaber relativos a um período de tempo por ele definido.
Preconditions	O administrador já efectuou login e o Super Use Case foi invocado.
Post-conditions	Sucesso: Apresentação da lista de créditos e débitos financeiros. Insucesso: Não é apresentada qualquer lista.

Flow of Events

	Actor Input	System Response
1	Insere data de inicio e de fim de período que pretende.	
2		Valida datas.
3		Valida período.
4		Identifica movimentos financeiros (débitos ou créditos) no período indicado.
5		Agrupa os resultados e mostra-os ao utilizador.

Alternative 2: Data inválida.

	Actor Input	System Response
1		Informa o administrador que a data inserida é inválidas.
2		Voltar a 1 (Main Flow).

Alternative 3: Intervalo inválido.

	Actor Input	System Response
1		Informa o administrador que a data inicial é mais recente que a data final.
2		Voltar a 1 (Main Flow).

Exception 5: Não existe resultados.

	Actor Input	System Response
1		Informa o cliente que não existem movimentos.

Figura 7.27: Descrição textual - Resultados Parciais

7.2 Diagramas de Sequência

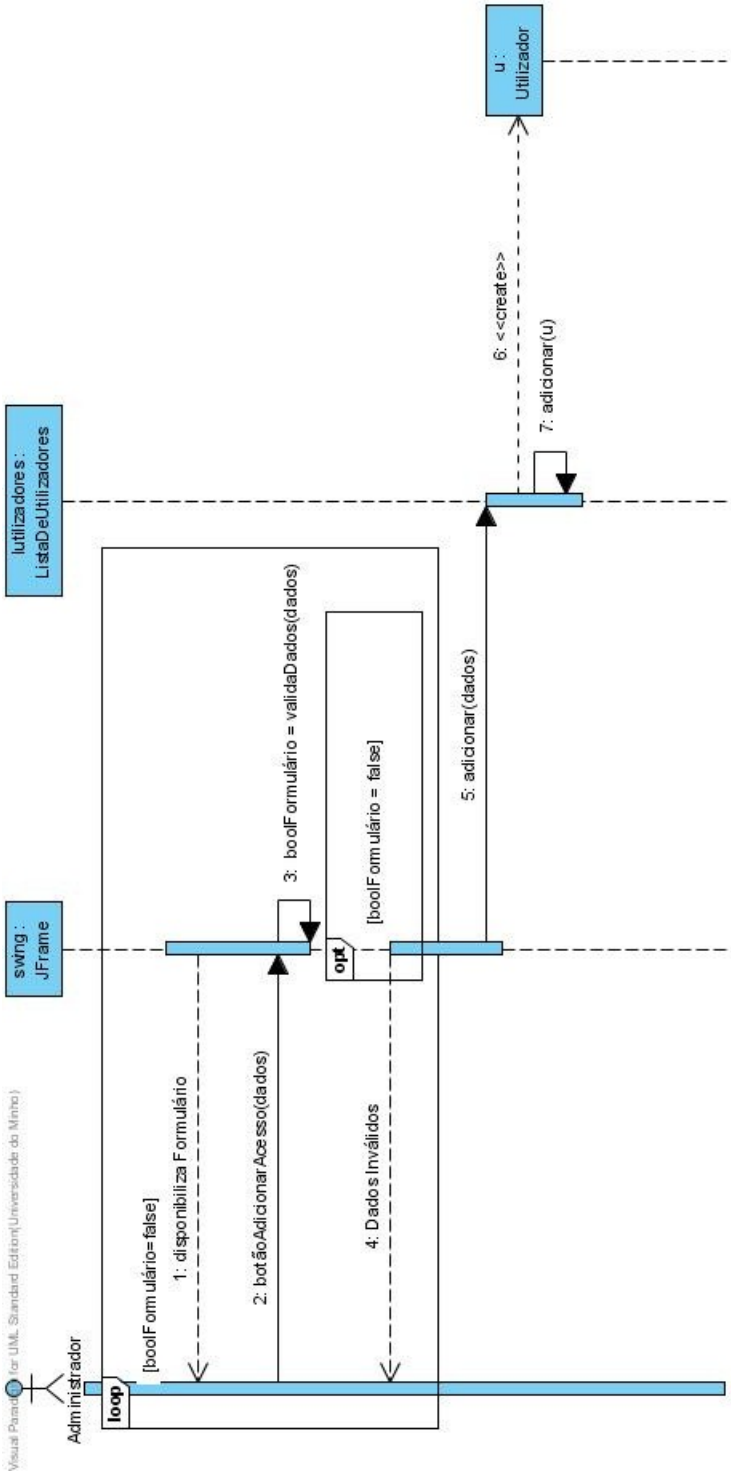


Figura 7.28: Diagrama de Sequência - Adicionar acesso

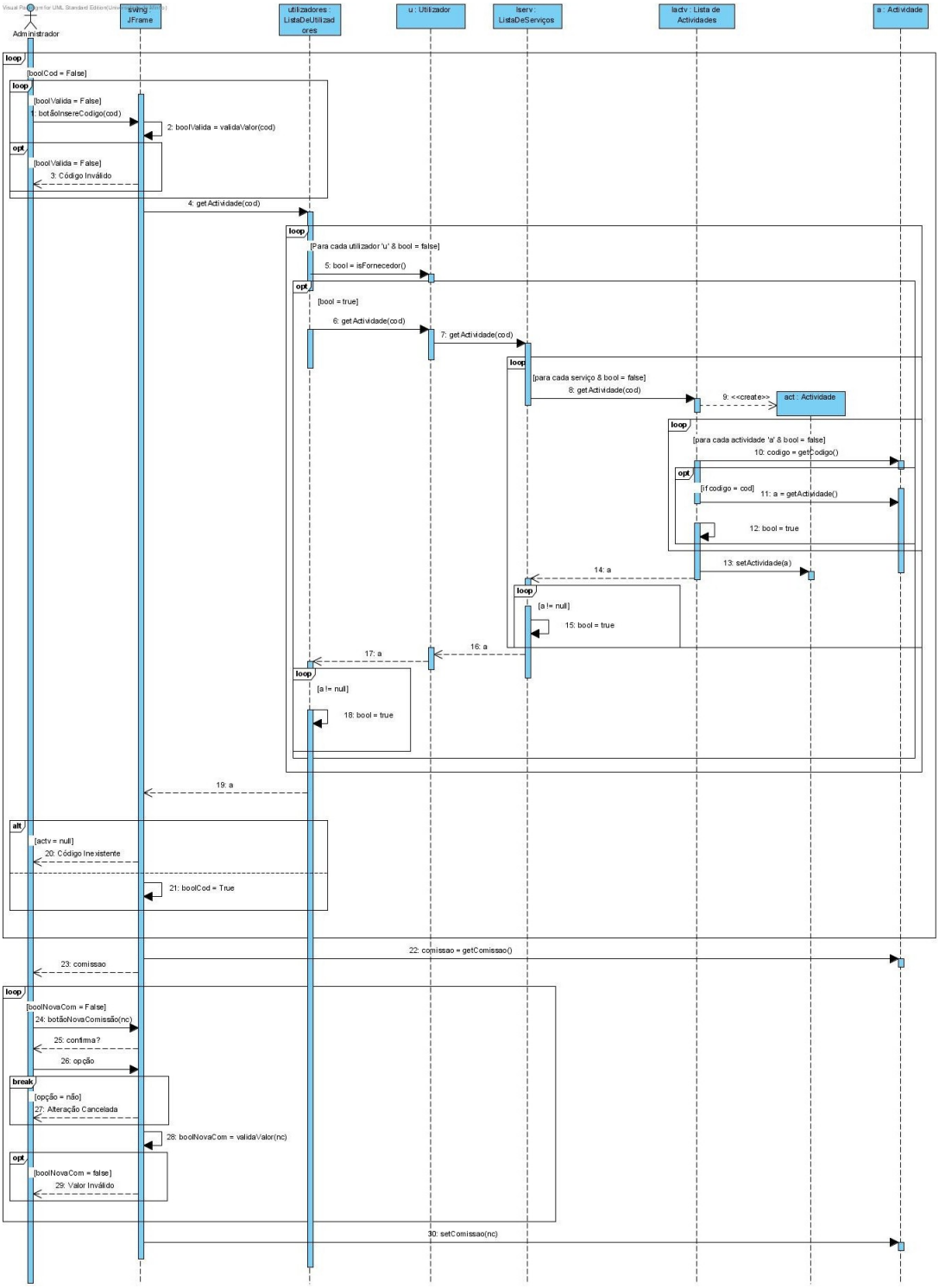


Figura 7.29: Diagrama de Sequência - Alterar comissão

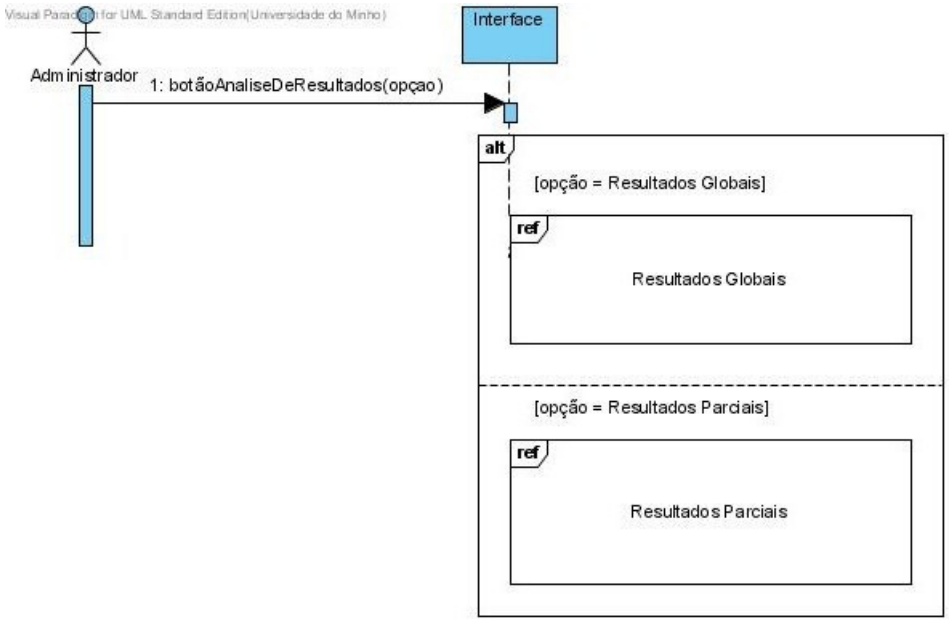


Figura 7.30: Diagrama de Sequência - Análise de Resultados

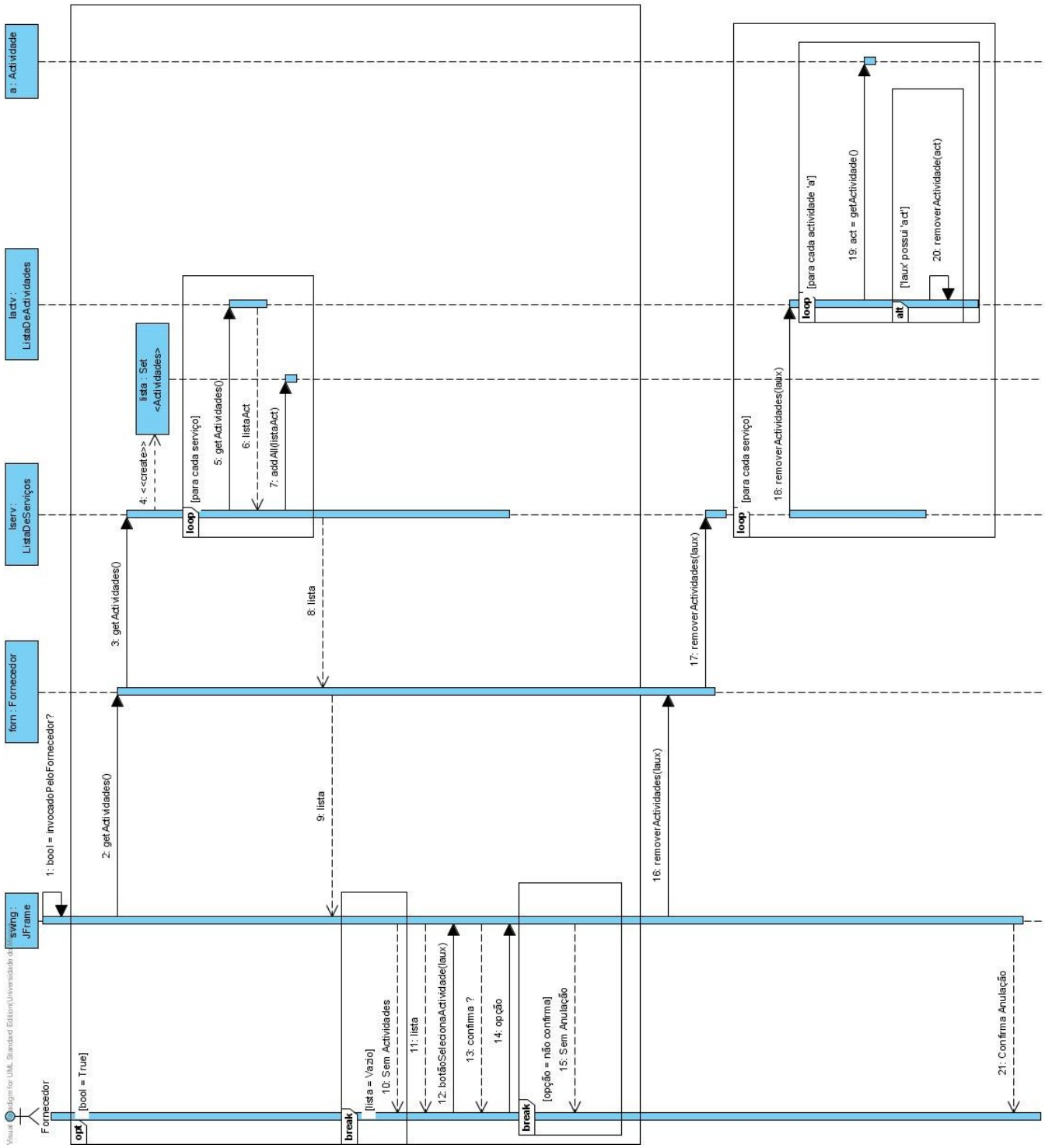


Figura 7.31: Diagrama de Sequência - Anular actividade

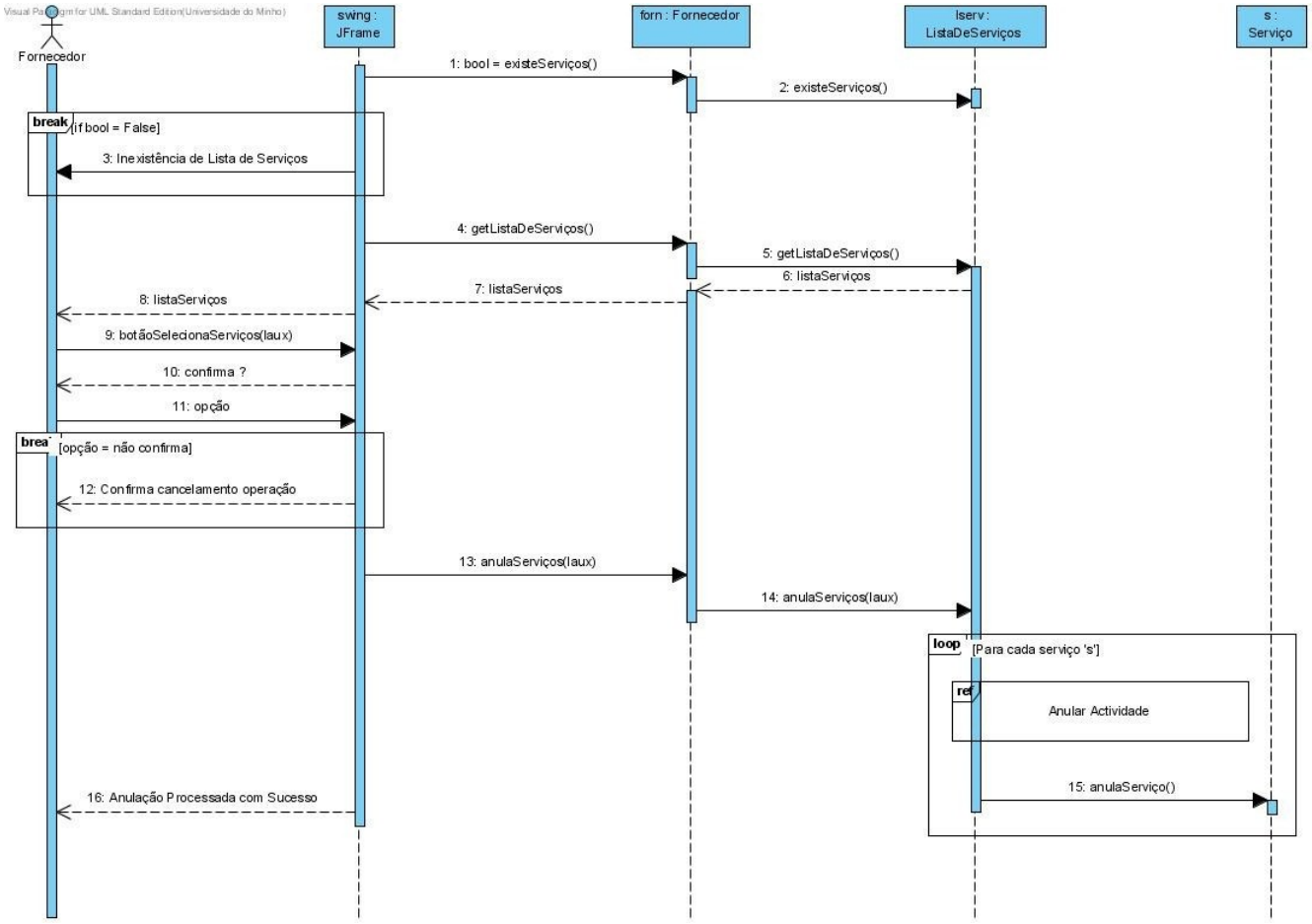


Figura 7.32: Diagrama de Sequência - Anular Serviço

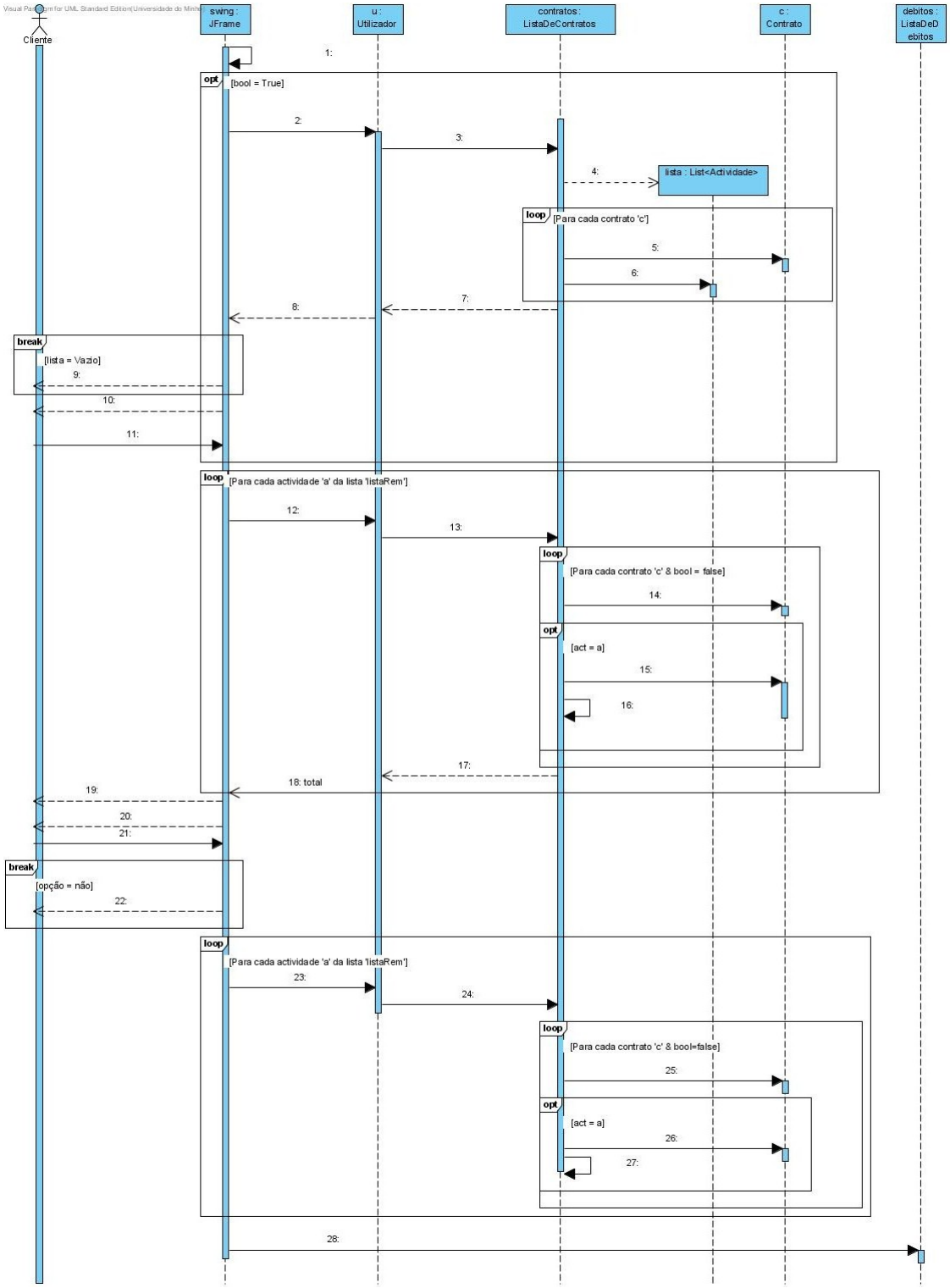


Figura 7.33: Diagrama de Sequência - Cancelar Actividade

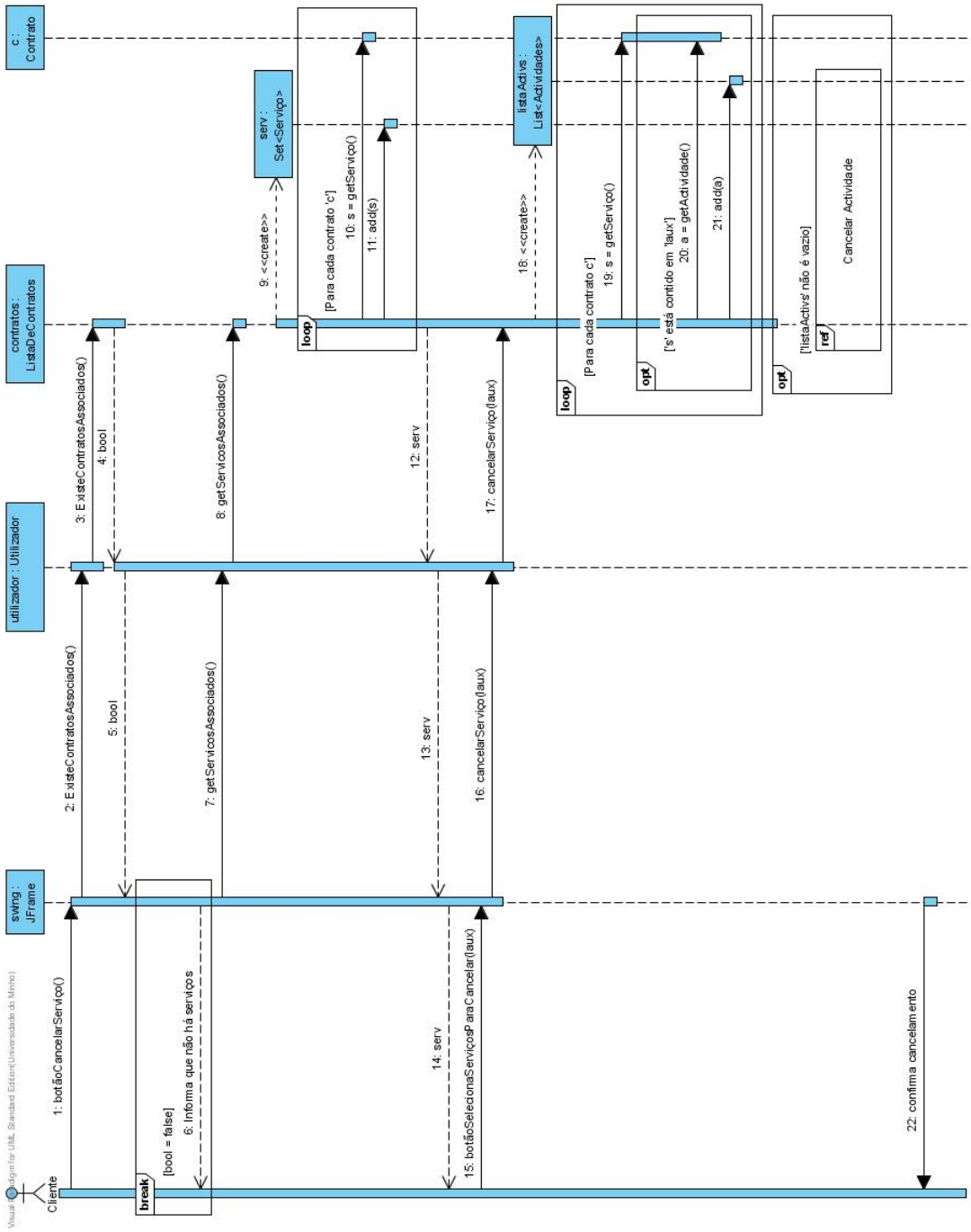


Figura 7.34: Diagrama de Sequência - Cancelar Serviço

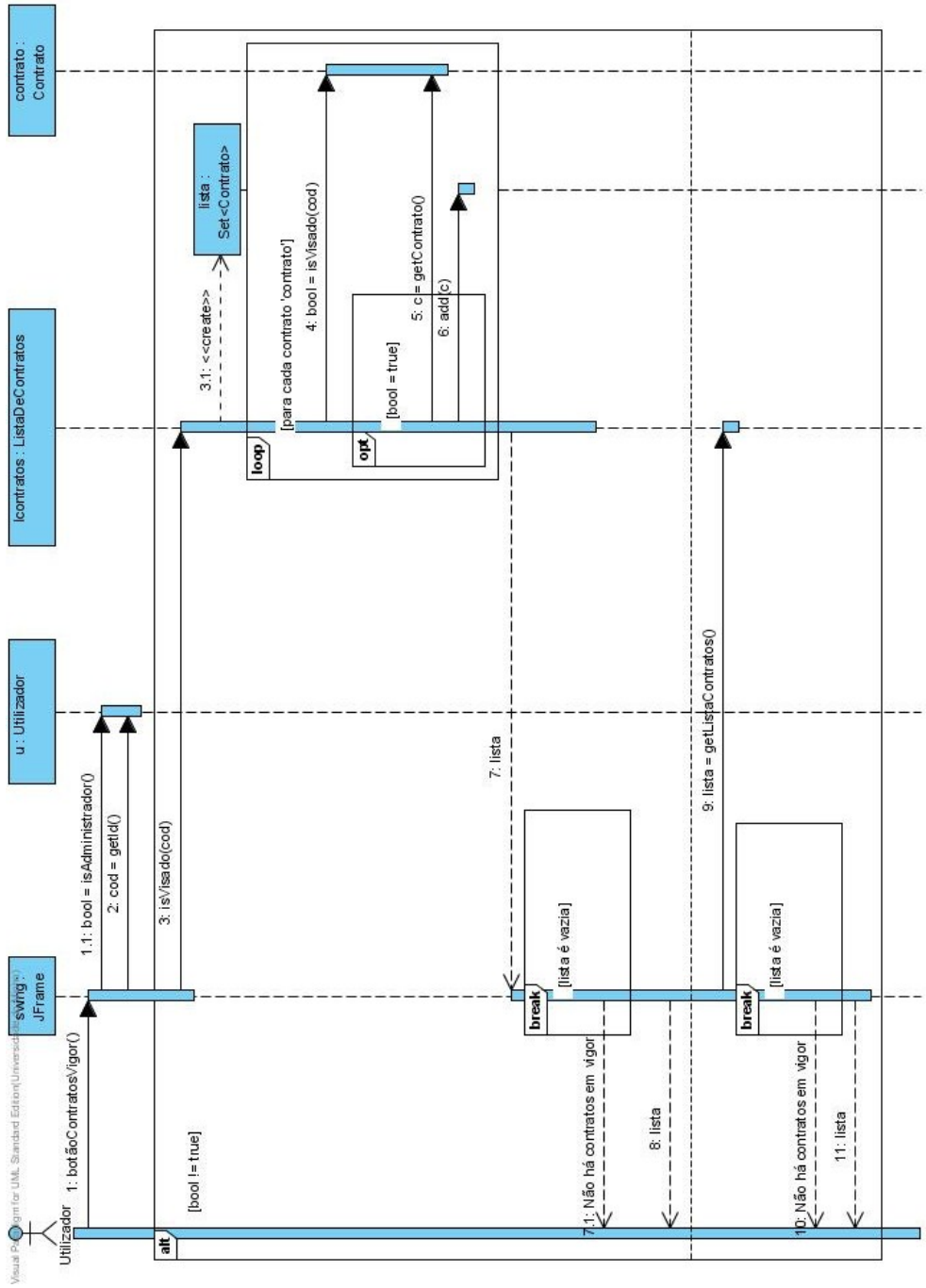


Figura 7.35: Diagrama de Sequência - Consulta de contratos em vigor

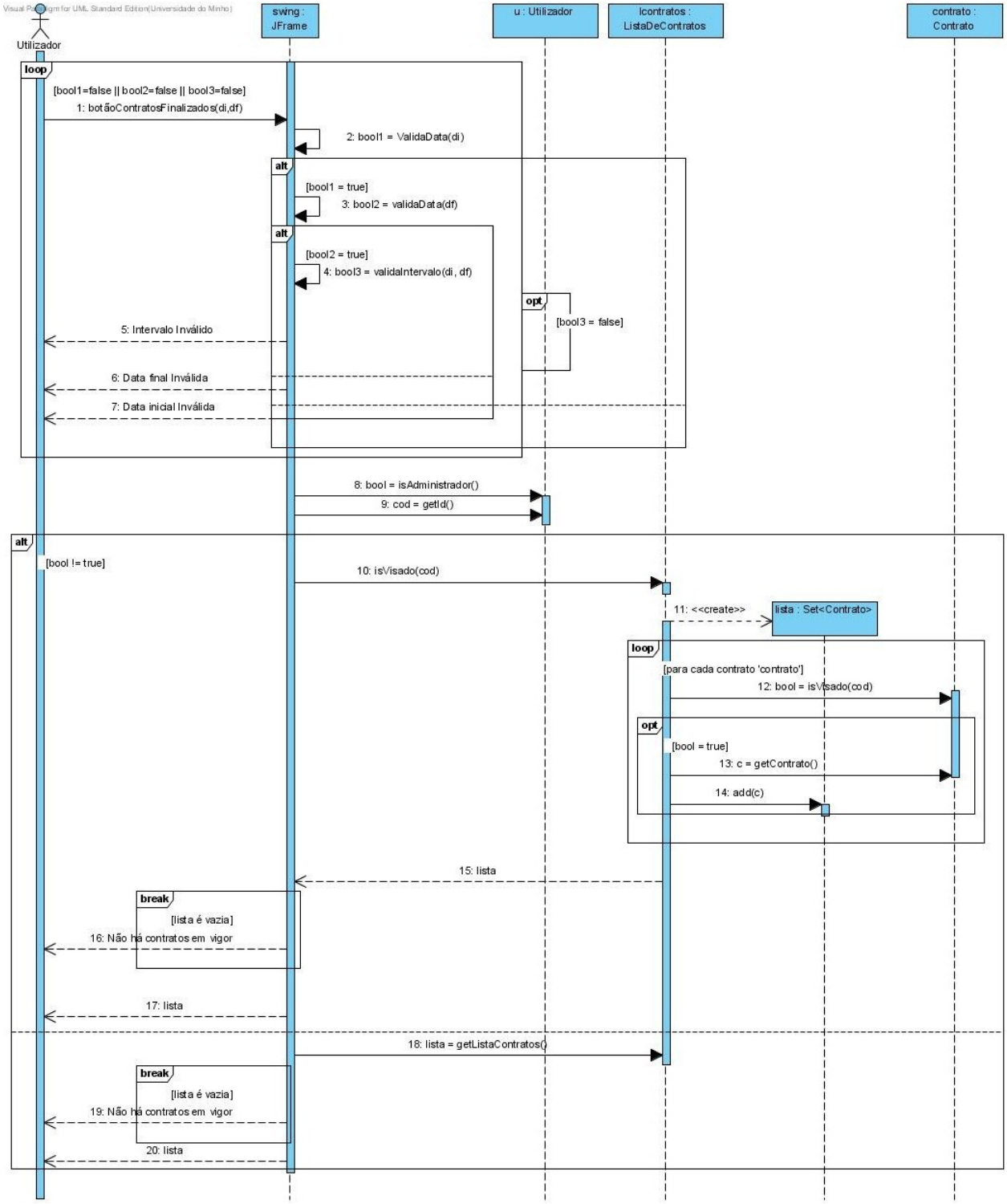


Figura 7.36: Diagrama de Sequência - Consulta de contratos finalizados

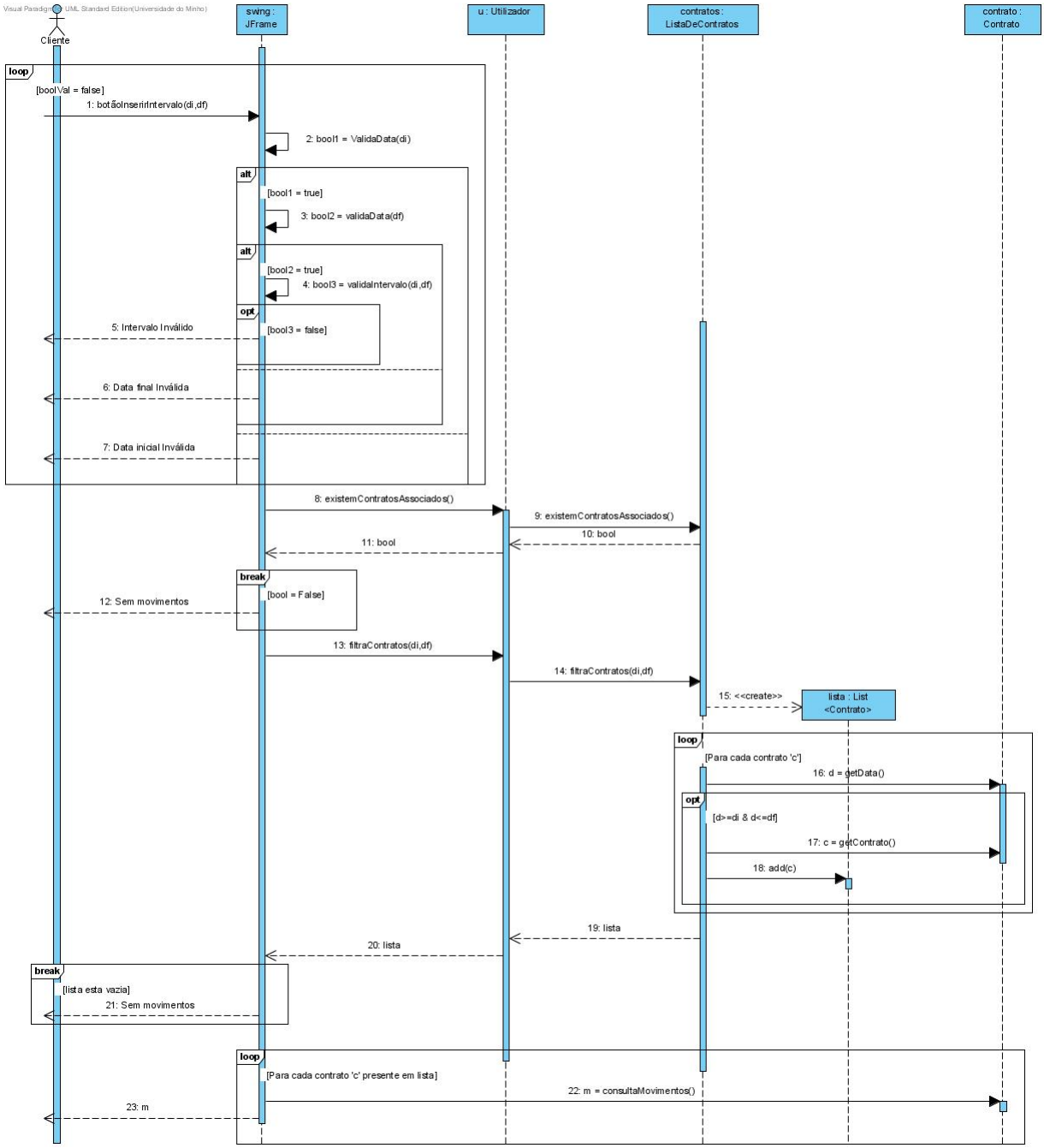


Figura 7.37: Diagrama de Sequência - Consulta Histórico

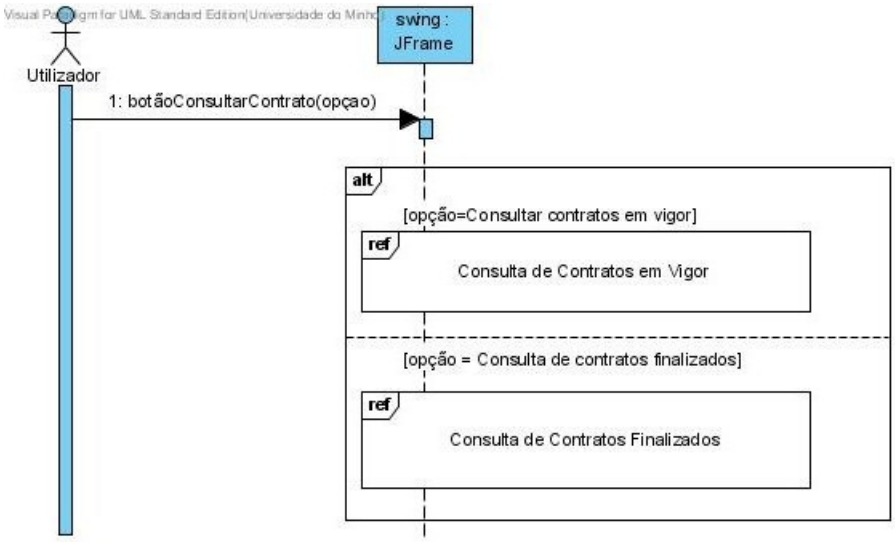


Figura 7.38: Diagrama de Sequência - Consultar Contratos

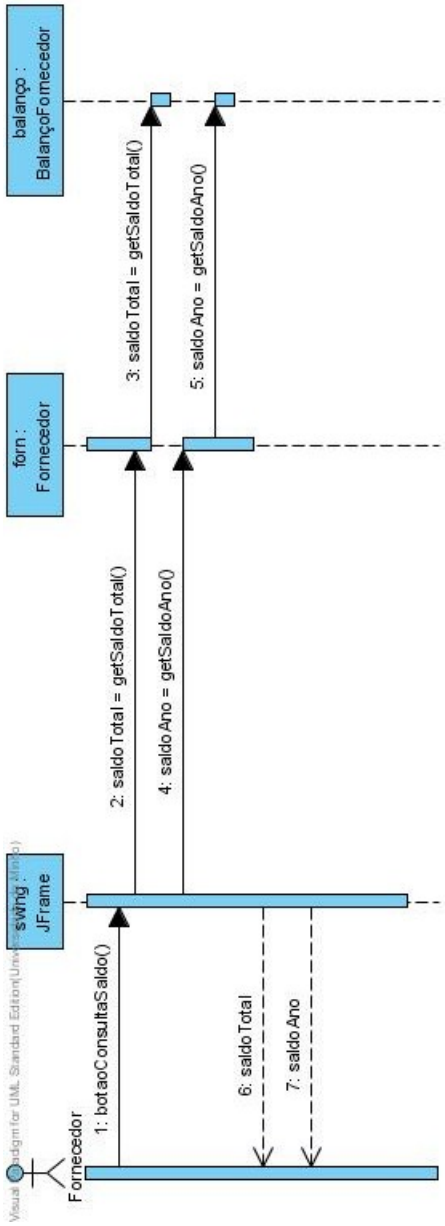


Figura 7.39: Diagrama de Sequência - Consultar Saldo

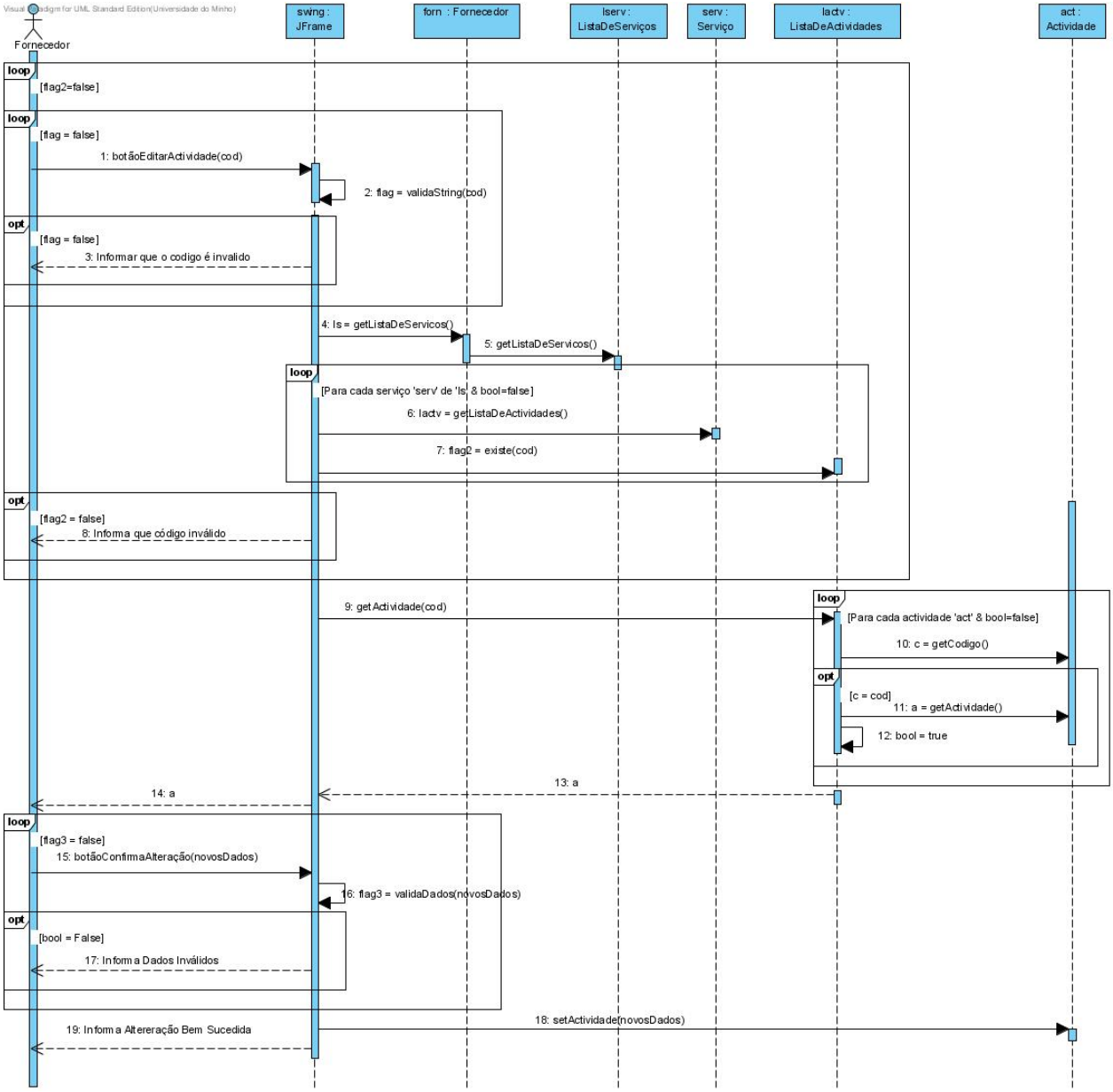


Figura 7.40: Diagrama de Sequência - Editar Actividade

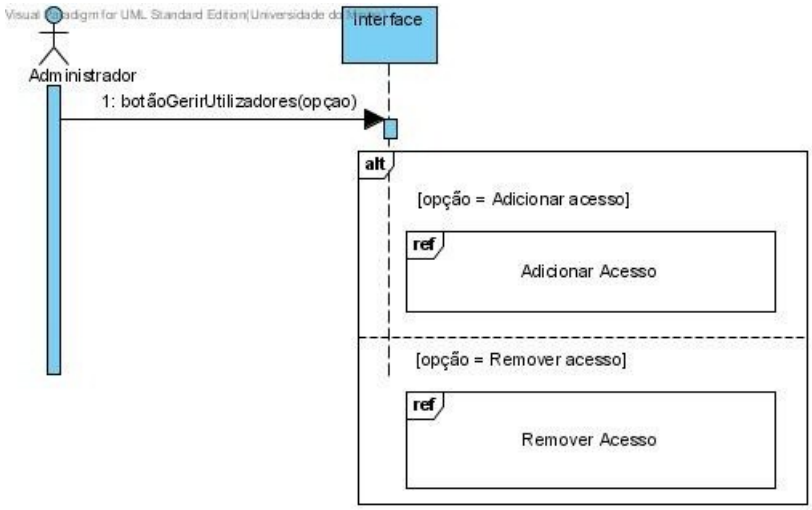


Figura 7.41: Diagrama de Sequência - Gerir Utilizadores

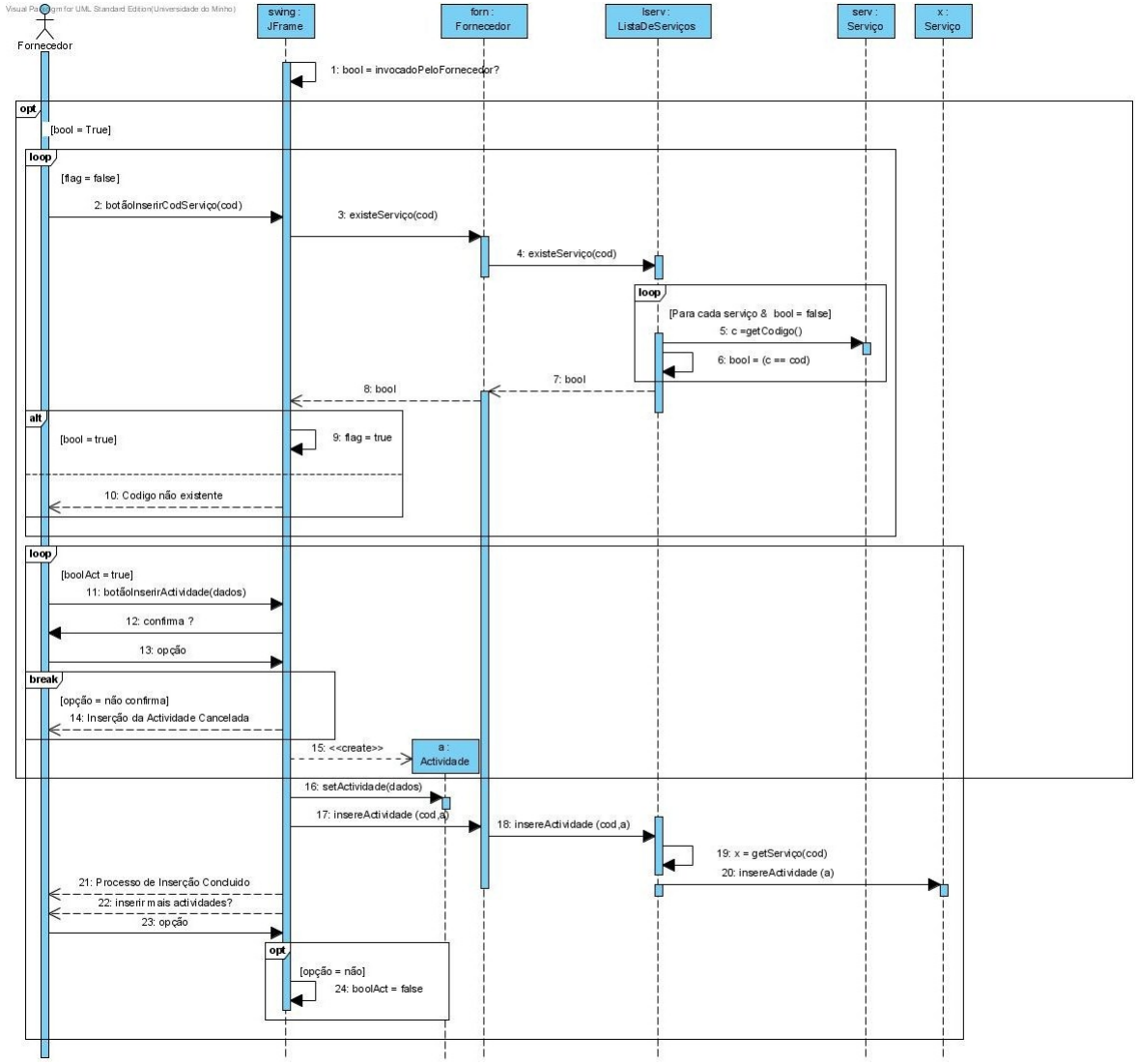


Figura 7.42: Diagrama de Sequência - Inserir Actividade

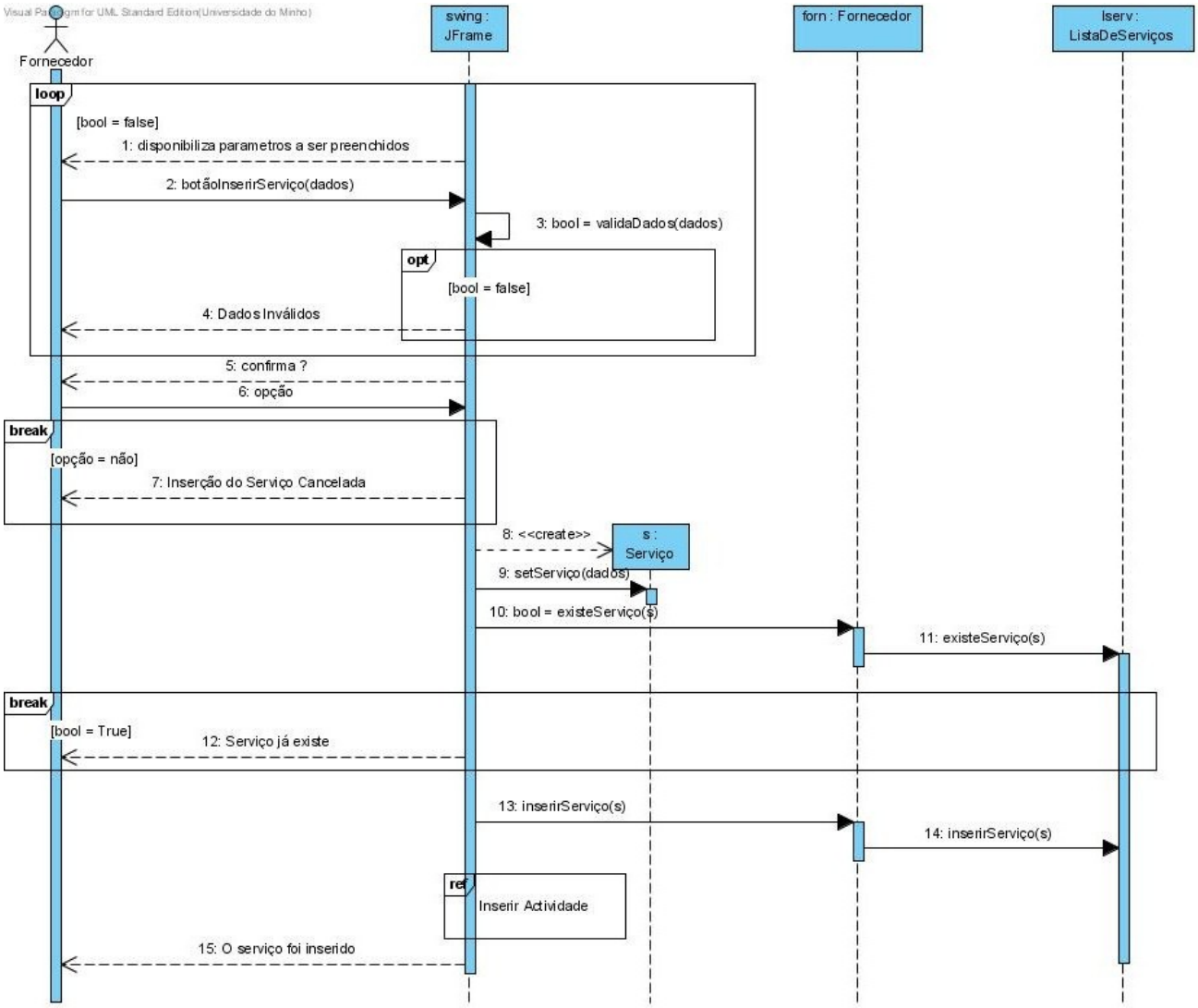


Figura 7.43: Diagrama de Sequência - Inserir Serviço

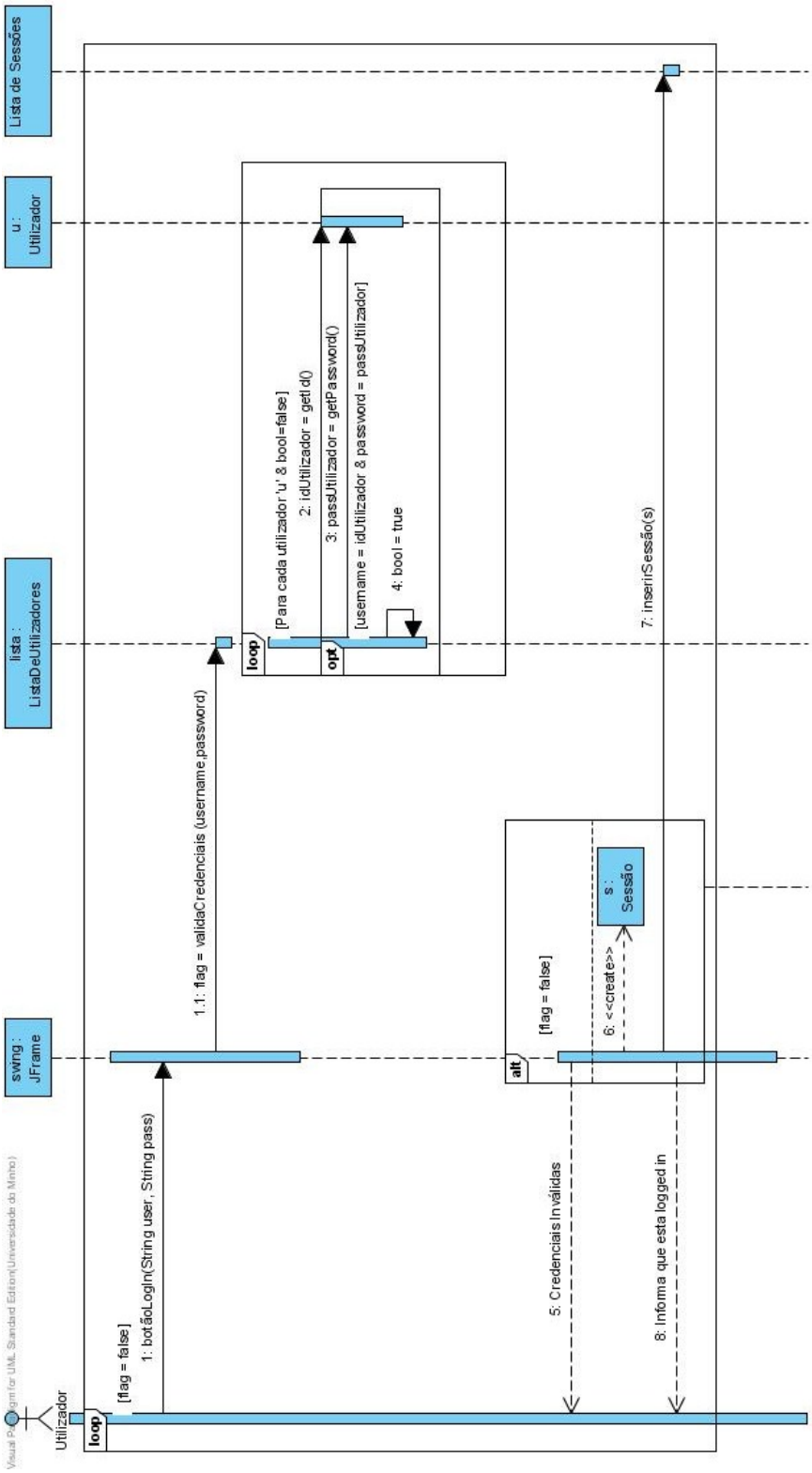


Figura 7.44: Diagrama de Sequência - Log In

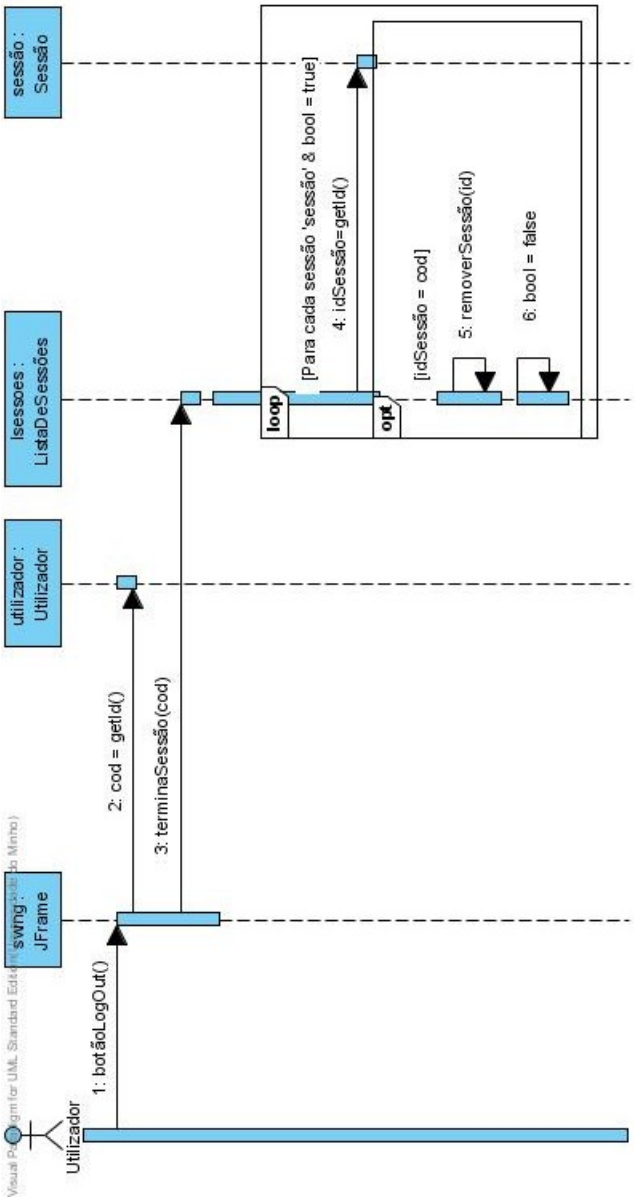


Figura 7.45: Diagrama de Sequência - Log Out

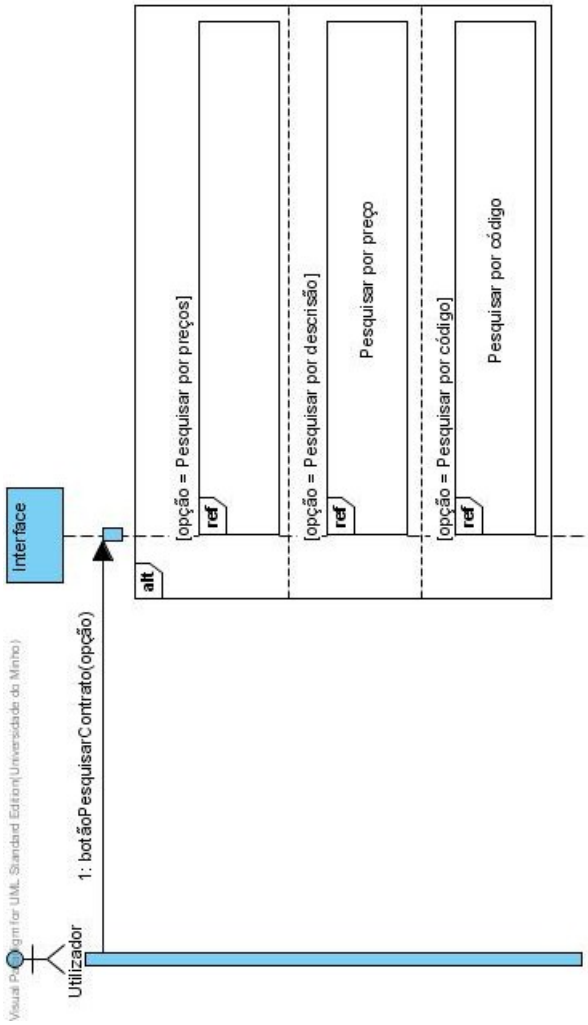


Figura 7.46: Diagrama de Sequência - Pesquisar Actividade

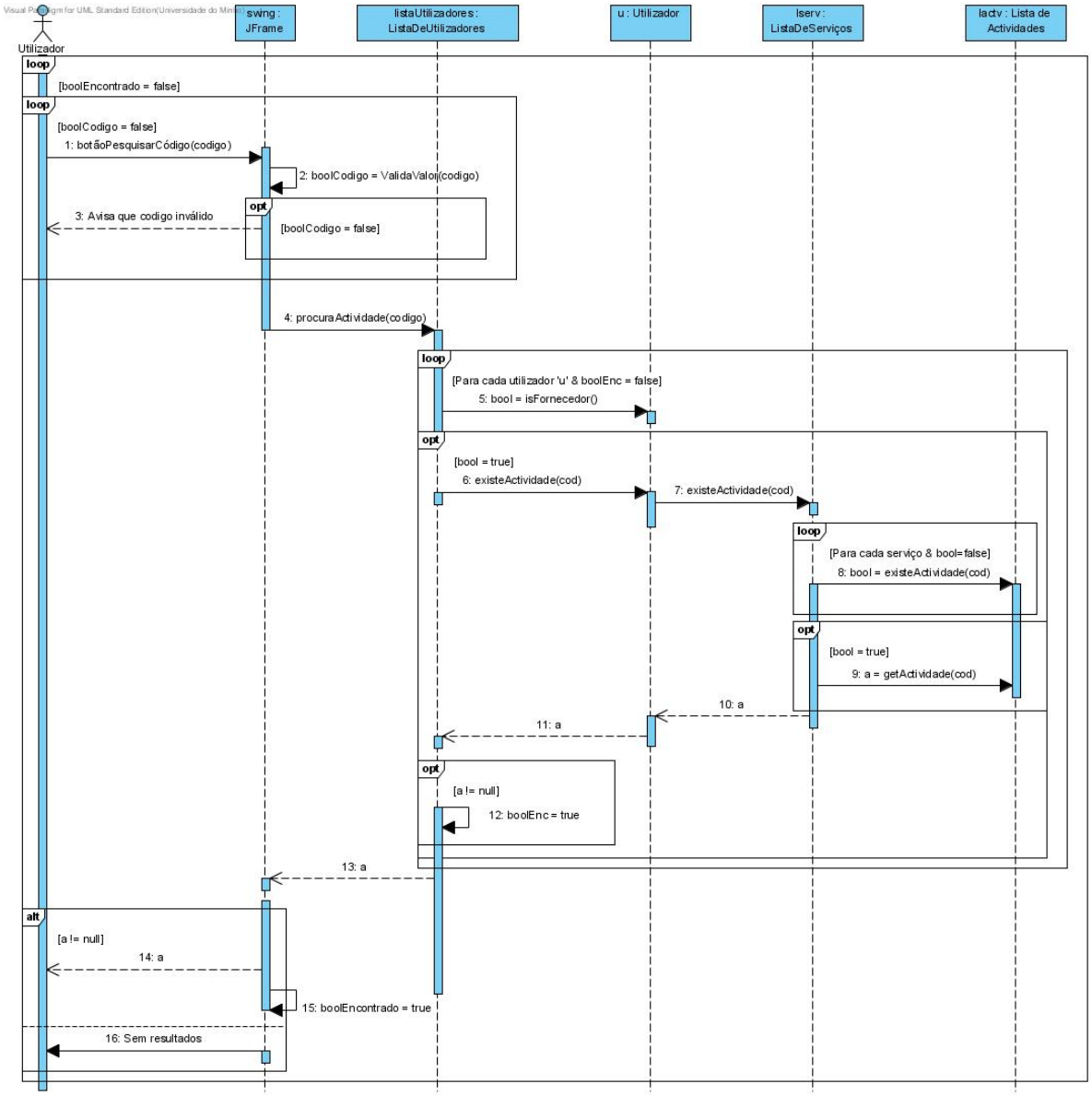


Figura 7.47: Diagrama de Sequência - Pesquisar por código

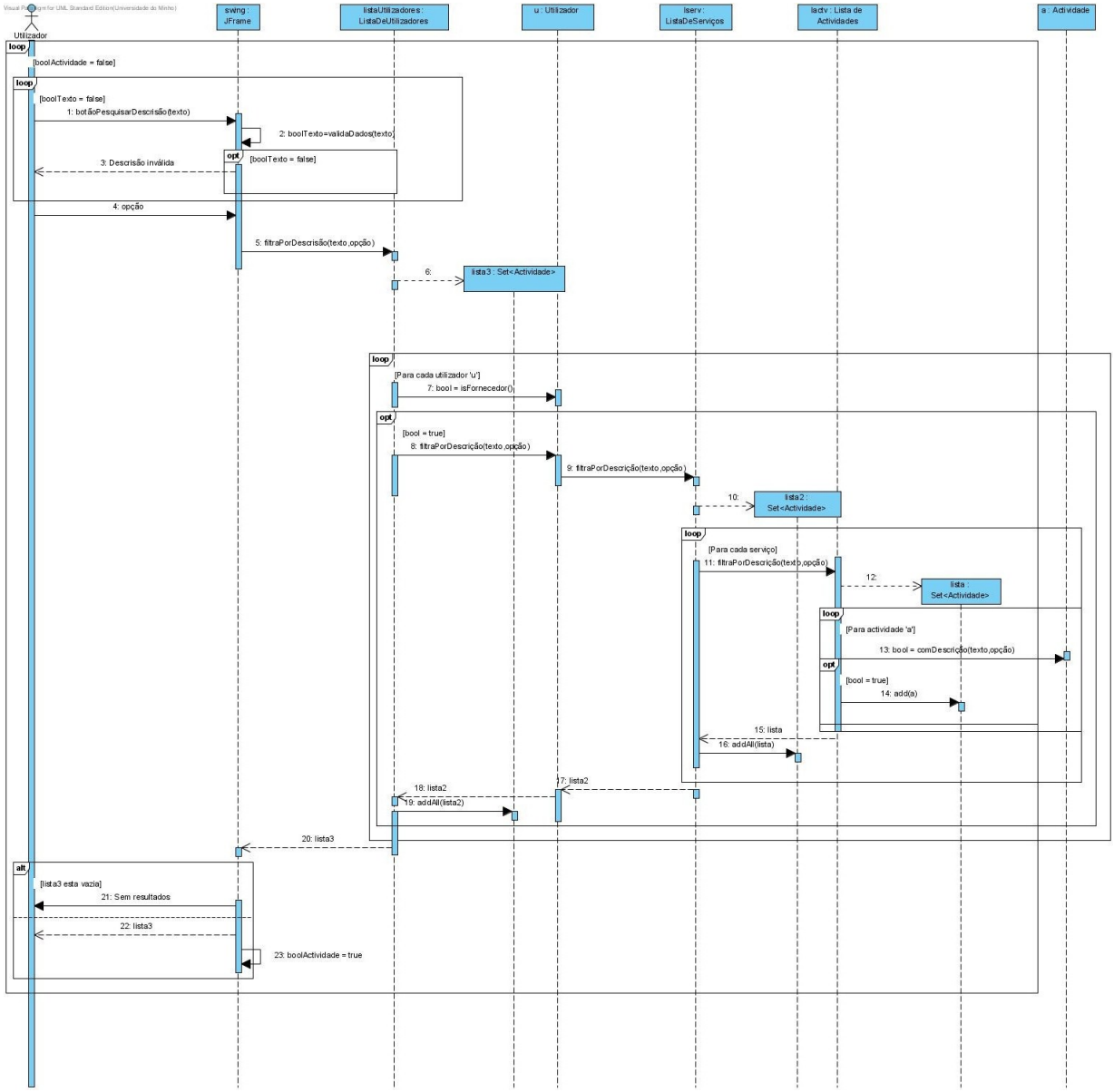


Figura 7.48: Diagrama de Sequência - Pesquisar por descrição

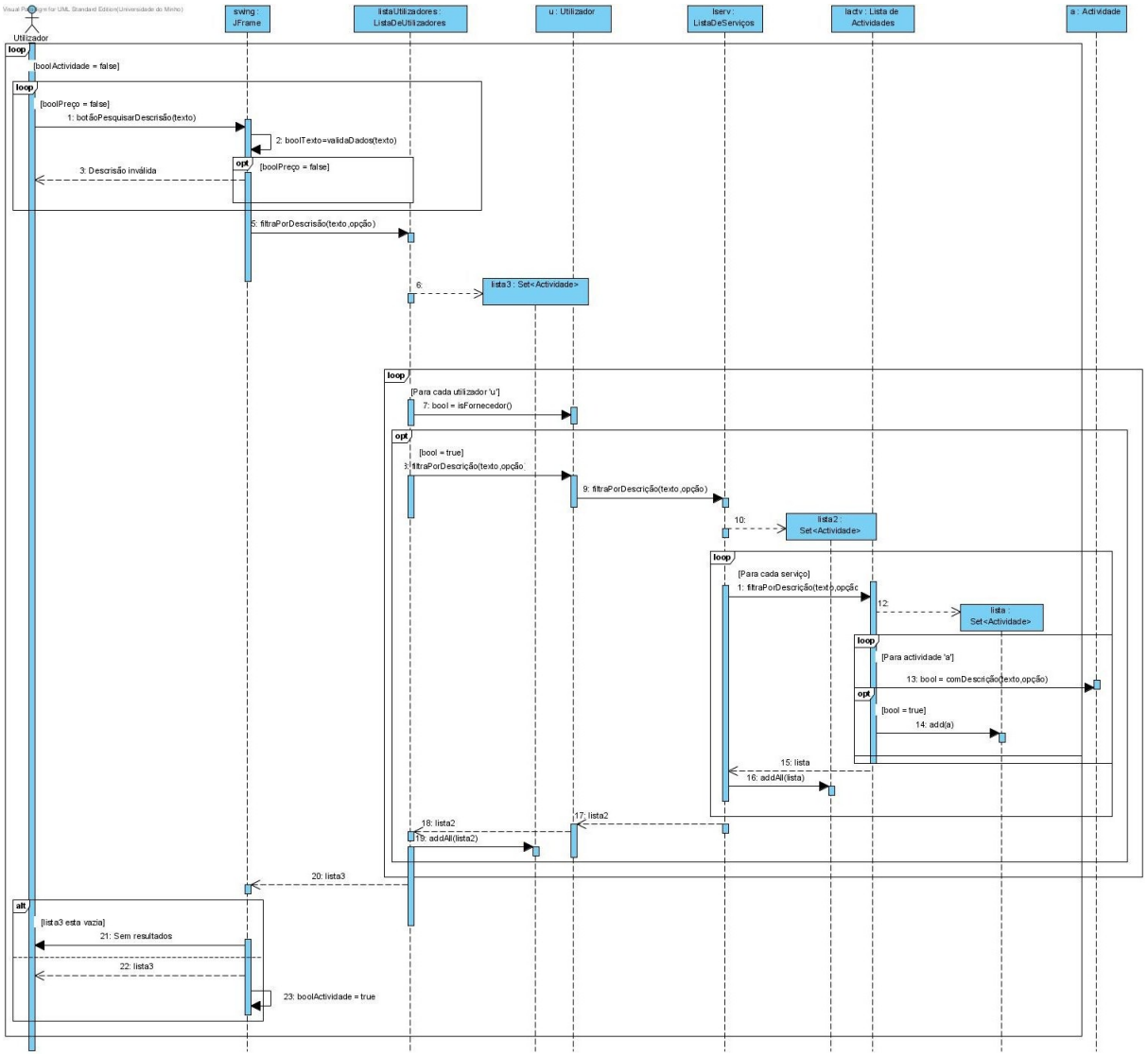


Figura 7.49: Diagrama de Sequência - Pesquisar por preço

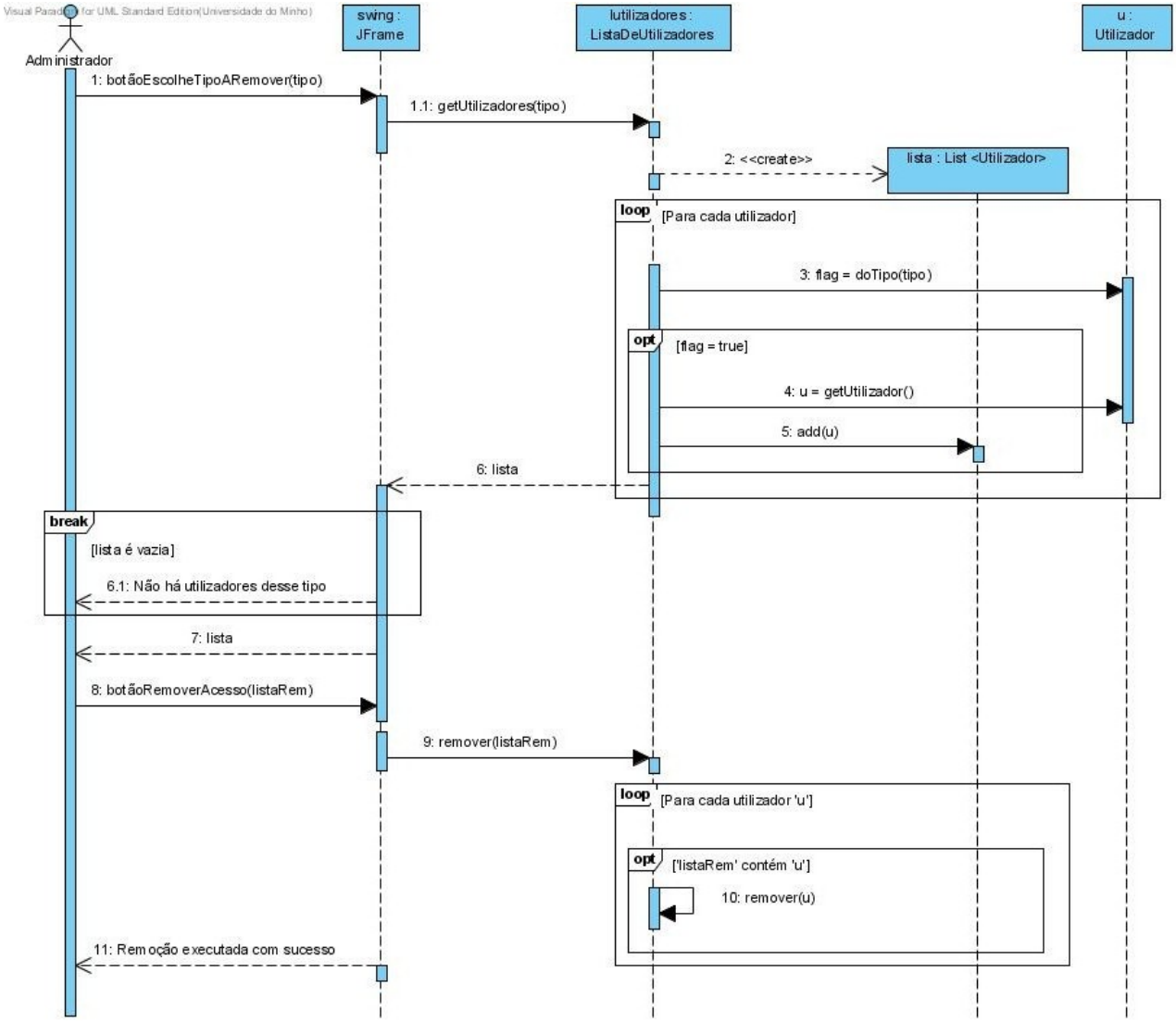


Figura 7.50: Diagrama de Sequência - Remover Acesso

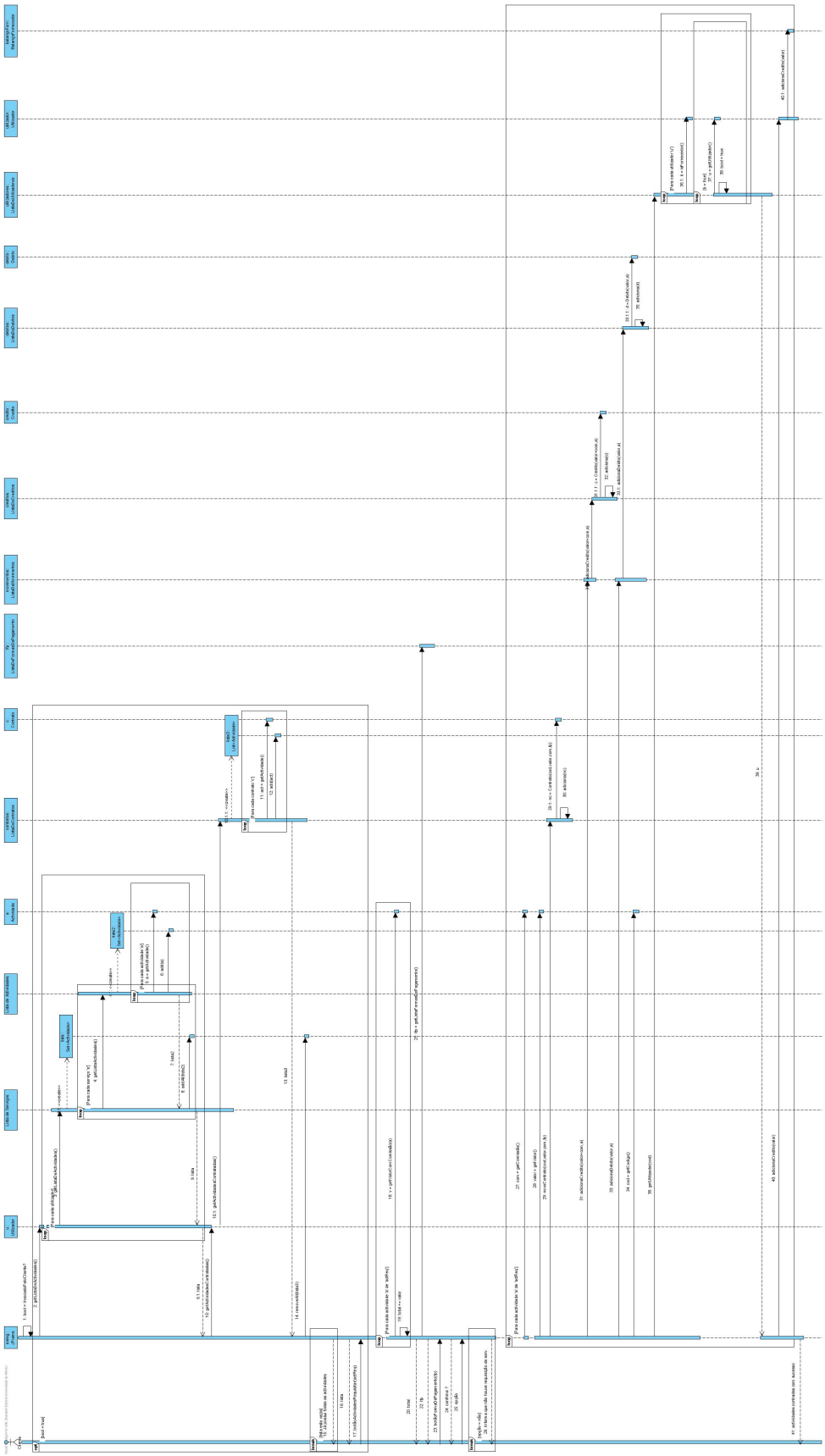


Figura 7.51: Diagrama de Sequência - Requisitar Actividade

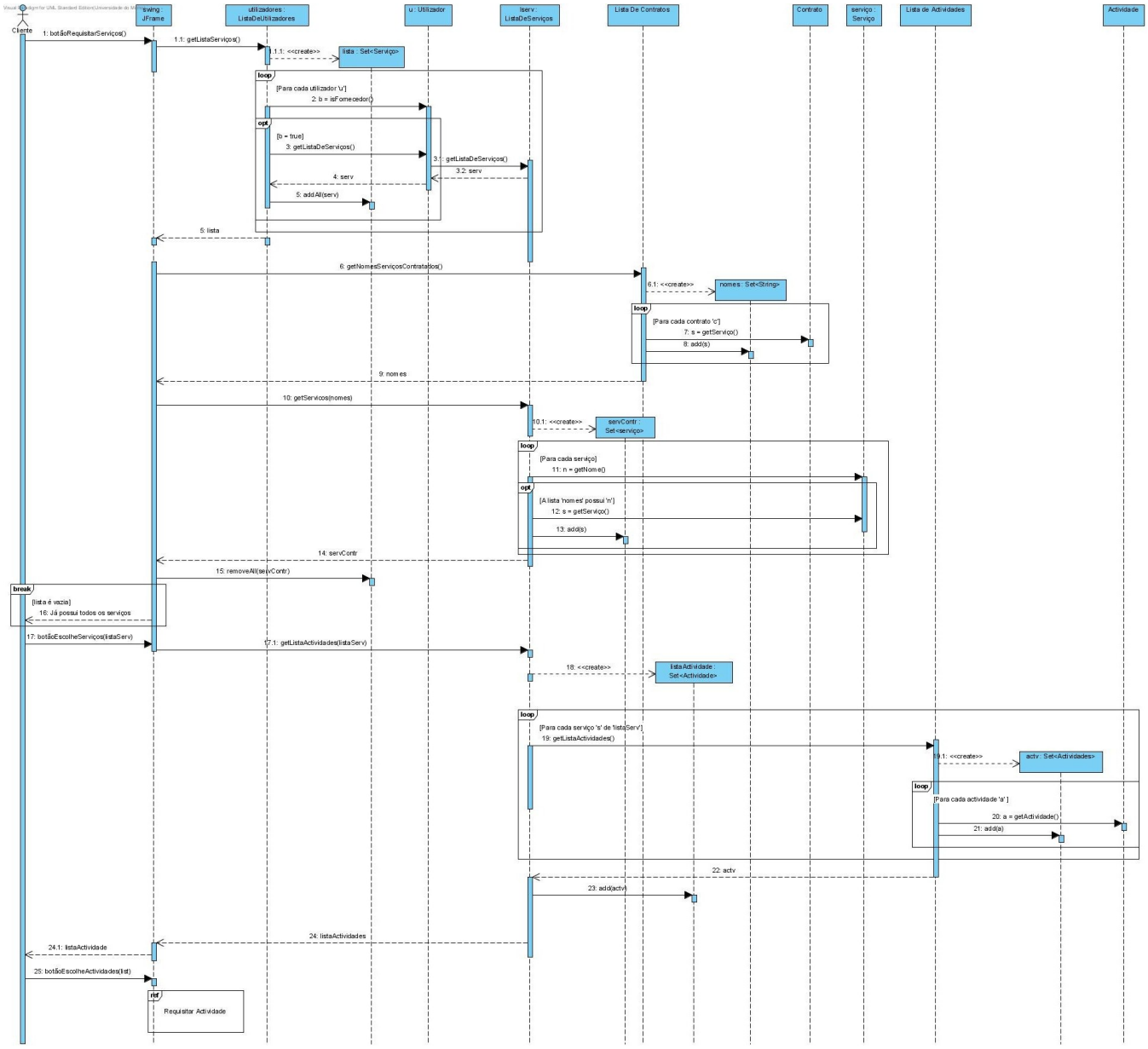


Figura 7.52: Diagrama de Sequência - Requisitar Serviço

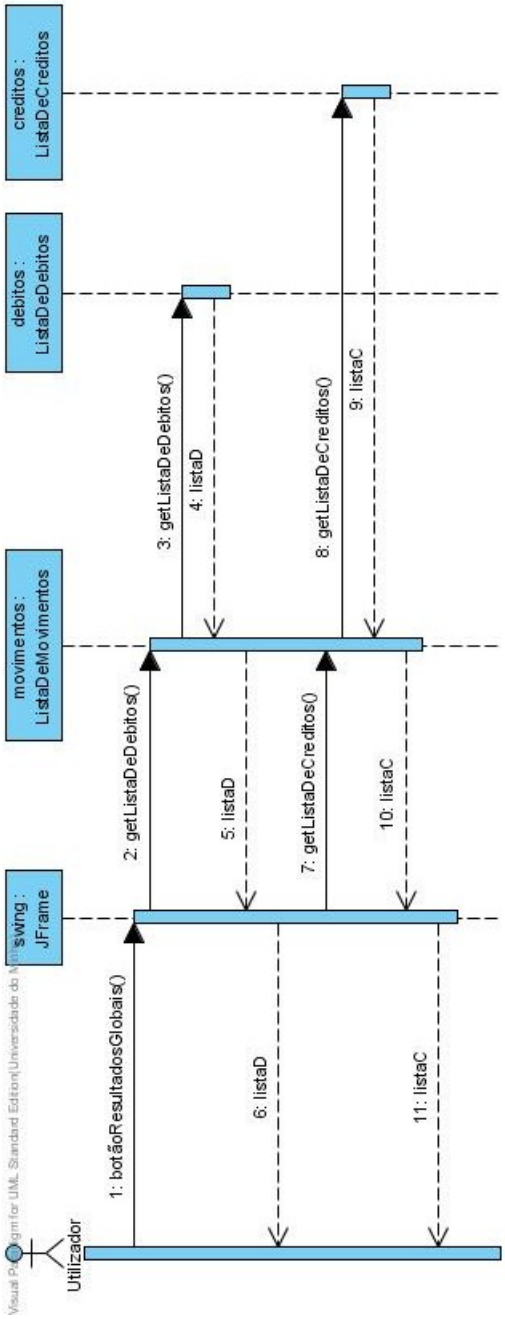


Figura 7.53: Diagrama de Sequência - Resultados Globais

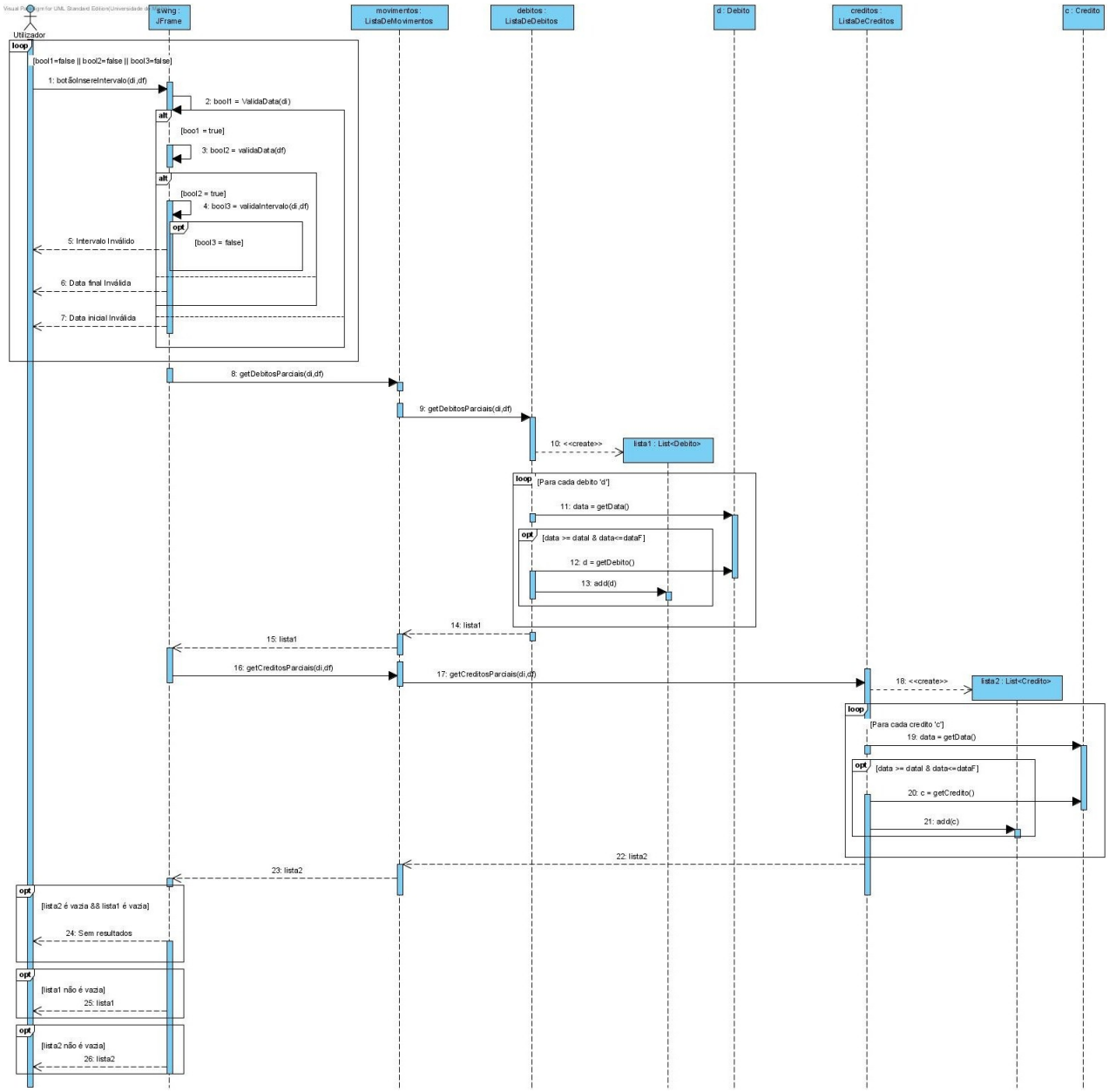


Figura 7.54: Diagrama de Sequência - Resultados Parciais

7.3 Métodos e Atributos do Diagrama de Classes

Figura 7.55: Métodos e Atributos das Classes do package User Interface

Figura 7.56: Métodos e Atributos das Classes do package GereComSaber

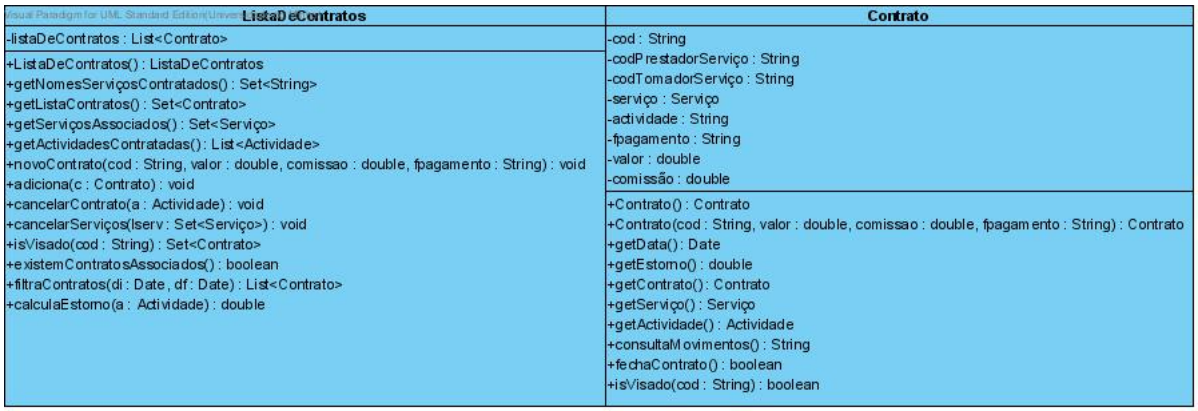


Figura 7.57: Métodos e Atributos das Classes do package Contratos

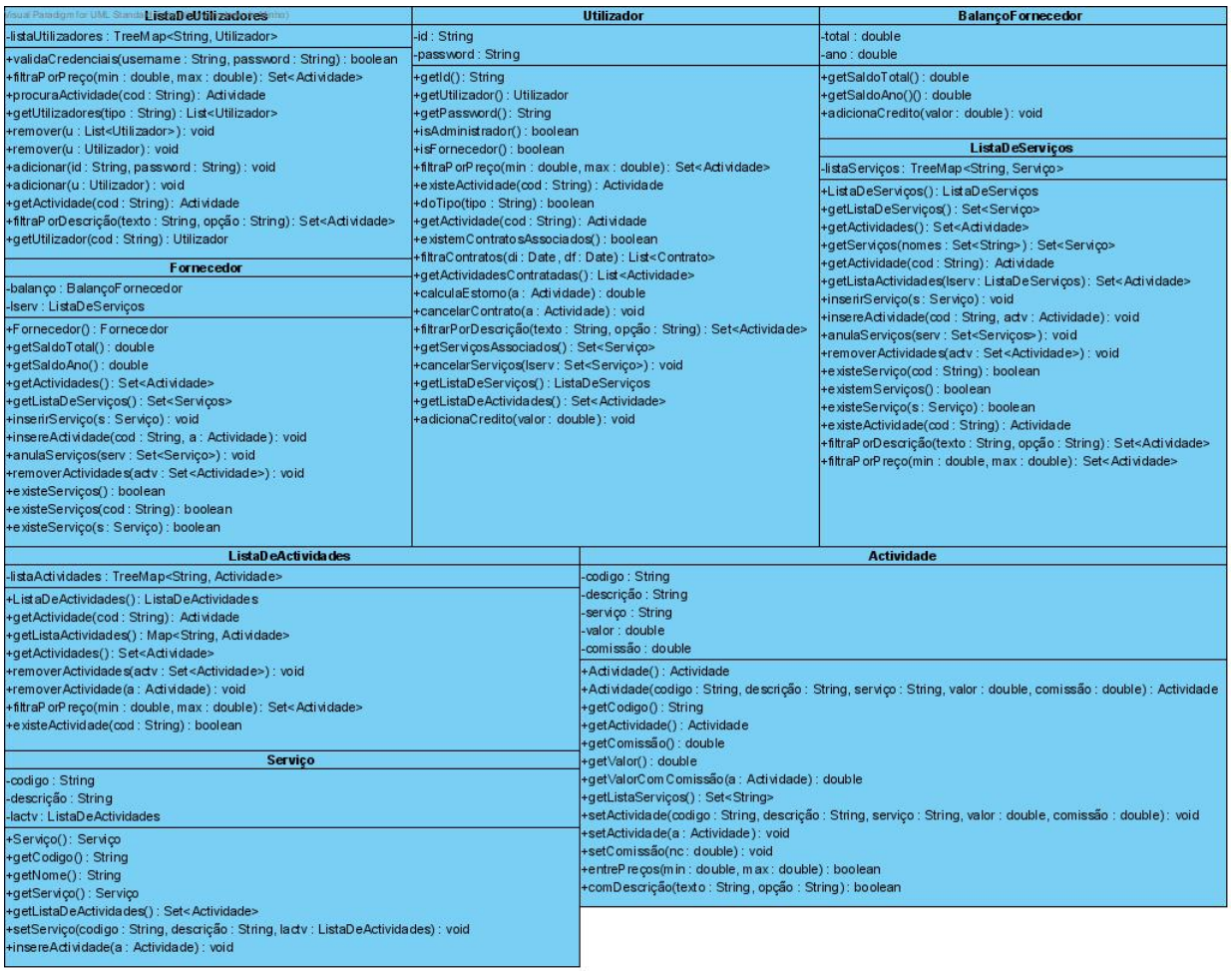


Figura 7.58: Métodos e Atributos das Classes do package Utilizadores

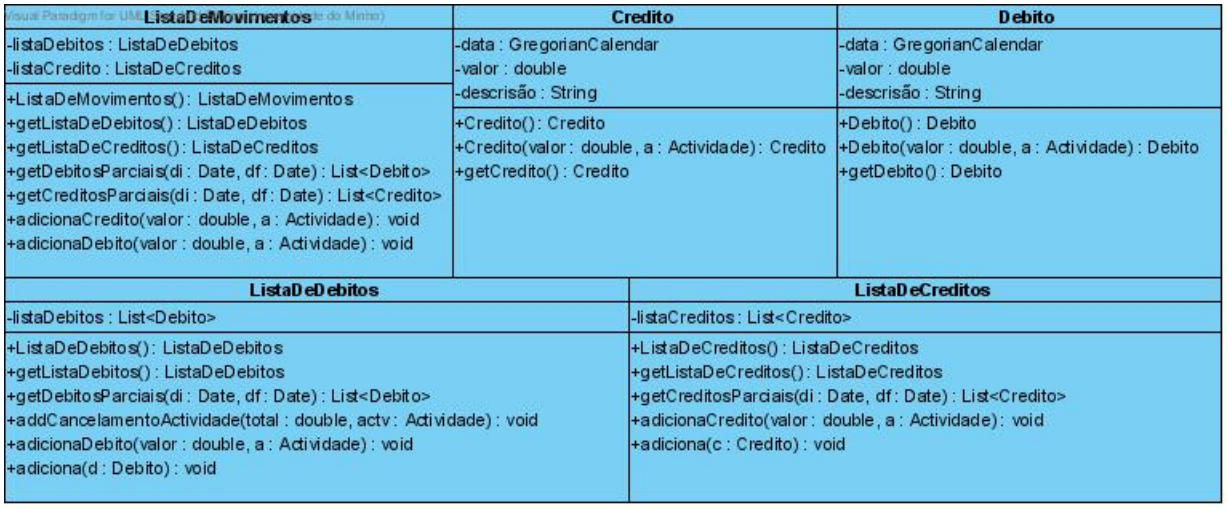


Figura 7.59: Métodos e Atributos das Classes do package Movimentos

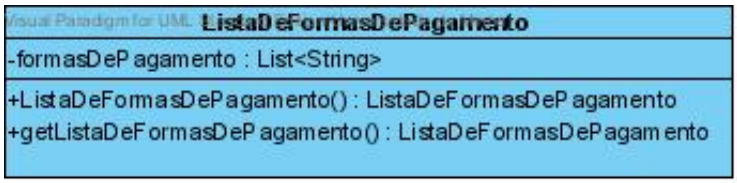


Figura 7.60: Métodos e Atributos das Classes do package Formas de Pagamento

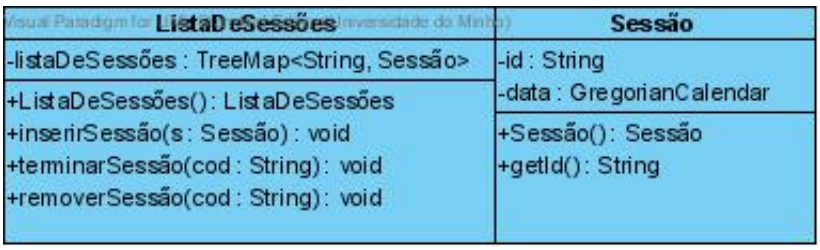


Figura 7.61: Métodos e Atributos das Classes do package Sessões