

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

João Fernandes, João Gonçalves, José Pereira, Ricardo Nogueira, Vasco Grilo

Dezembro, 2009

GereComSaber

Sistema de gestão de condomínios

GRUPO 8



João Pedro Afonso Fernandes 47098



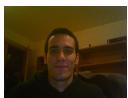
João de Melo Feio Pinheiro Gonçalves 49351



José Miguel Ribeiro Pinhão Pereira 47021



Ricardo Jorge da Silva Gama Nogueira 51851



Vasco André da Costa Grilo 49403

Dezembro, 2009

Resumo

Este relatório é referente à Fase 2 do projecto da Unidade Curricular Desenvolvimento de Sistemas de Software e o seu objectivo é a implementação de uma ferramenta capaz de auxiliar a gestão de serviços prestados a condomínios, e não só, seguindo normas de especificação ensinadas na cadeira. A linguagem intermédia usada foi o UML e a ferramenta utilizada foi o *Visual Paradigm*. Nesta Fase o objectivo foi desenhar a arquitectura do sistema e definir a interacção dos vários objectos que vão por em prática o mesmo. Para tal usaram-se os Diagramas de Classe para definir a arquitectura e, os Diagramas de Sequência para a interacção das classes.

Palavras-Chave: UML, Modelação, JAVA, Visual Paradigm.

Índice

Resumo	i
Índice	ii
1 Introdução	1
1.1 Motivação e Objectivos	1
1.2 Estrutura do Relatório	1
2 Requisitos	3
2.1 Levantamento de Requisitos	5
2.2 Modelo de Domínio	7
2.3 Use Cases	7
3 Arquitectura	9
3.1 Diagrama de Classes	10
3.2 Diagramas de Sequência	11
4 Alterações à Modelação Inicial	13
5 Conclusões e Trabalho Futuro	14
6 Anexos	15
Bibliografia	75

Índice de Figuras

2.1	Modelo de domínio	7
2.2	Menu Funcionário	8
3.1	Diagrama de classes	11
3.2	Diagrama de sequência para o registo de um novo Utilizador	12
6.1	Diagrama Menu Principal	16
6.2	Diagrama Menu Administrador	16
6.3	Descrição Use Case: Alterar Conta de Utilizador	17
6.4	Descrição Use Case: Registar novo Utilizador	18
6.5	Descrição Use Case: Remover Utilizador	18
6.6	Diagrama Menu Gerente	19
6.7	Descrição Use Case: Saltar para o dia X	19
6.8	Descrição Use Case: Confirmar Pagamento Extra de Cliente	20
6.9	Descrição Use Case: Confirmar Estorno devolvido a Cliente	20
6.10	Diagrama Menu Financeiro	21
6.11	Descrição Use Case: Calcular Margem da GereComSaber	21
6.12	Diagrama Gerir Serviços	22
6.13	Descrição Use Case: Inserir Novo Serviço a Fornecedor	23
6.14	Descrição Use Case: Alterar Serviços de Fornecedores	24
6.15	Descrição Use Case: Autenticação	24
6.16	Descrição Use Case: Remover Serviço a Fornecedor	25
6.17	Descrição Use Case: Inserir Nova Categoria de Serviço	26
6.18	Descrição Use Case: Remover Categoria de Serviço	27
6.19	Diagramas Gerir Clientes	28
6.20	Descrição Use Case: Activar Serviço a Cliente	29

6.21 Descrição Use Case: Consultar Lista de Clientes Activos	30
6.22 Descrição Use Case: Consultar Lista de Clientes Inactivos	30
6.23 Descrição Use Case: Remover Cliente	31
6.24 Diagrama Gerir Carteira de Fornecedores	31
6.25 Descrição Use Case: Inserir Novo Fornecedor	32
6.26 Descrição Use Case: Alterar Dados do Fornecedor	33
6.27 Descrição Use Case: Remover Fornecedor	34
6.28 Descrição Use Case: Consultar Lista de Fornecedores	34
6.29 Diagrama Menu Funcionario	35
6.30 Descrição Use Case: Registrar Novo Cliente	36
6.31 Descrição Use Case: Alterar Dados de Cliente	37
6.32 Descrição Use Case: Adicionar Serviço a Cliente	38
6.33 Descrição Use Case: Remover Serviço a um Cliente	39
6.34 Descrição Use Case: Extender Prazo de Duração de Serviço de um Cliente .	40
6.35 Descrição Use Case: Diminuir Tempo de Duração de Serviço de um Cliente .	41
6.36 Descrição Use Case: Consultar Lista de Serviços de um Cliente	42
6.37 Diagrama Menu Cliente	43
6.38 Descrição Use Case: Adicionar Serviço	44
6.39 Descrição Use Case: Cancelar Serviço	45
6.40 Descrição Use Case: Alterar Dados Pessoais	46
6.41 Descrição Use Case: Extender Tempo de Duração de Serviço	47
6.42 Descrição Use Case: Reduzir Tempo de Duração de Serviço	48
6.43 Descrição Use Case: Ver Estado Actual dos Serviços	49
6.44 Descrição Use Case: Consultar Lista de Serviços Disponíveis	49
6.45 Diagrama de sequência : existeFornecedor(nome)	50
6.46 Diagrama de sequência de Subsistema : Inserir Novo Fornecedor	50
6.47 Diagrama de sequência : inserirFornecedor(nome,contacto,NIB	50
6.48 Diagrama de sequência : gerarCodigoFornecedor()	51
6.49 Diagrama de sequência Subsistema : Remover Fornecedor	51
6.50 Diagrama de sequência : removerFornecedor(codigo)	51
6.51 Diagrama de sequência Subsistema : Alterar Dados Fornecedor	51
6.52 Diagrama de sequência : Alterar Dados Fornecedor	52
6.53 Diagrama de sequência : GerarCodigoContrato	52

6.54 Diagrama de sequência Subsistema : Adicionar Serviço	52
6.55 Diagrama de sequência : Adicionar Serviço	52
6.56 Diagrama de sequência Subsistema : Cancelar Serviço	53
6.57 Diagrama de sequência : cancelarServico(codCliente,codContrato)	53
6.58 Diagrama de sequência : getPrecoDia(contrato)	53
6.59 Diagrama de sequência Subsistema: Reduzir Tempo de Duração de Serviço .	54
6.60 Diagrama de sequência : reduzirTempoServico(codcliente,codcontrato,novadata)	54
6.61 Diagrama de sequência Subsistema : Extender Tempo de Duração de Serviço	54
6.62 Diagrama de sequência : extenderTempoServico(codcliente,codcontrato,novadata)	55
6.63 Diagrama de sequência : existeUtilizador(user)	56
6.64 Diagrama de sequência Subsistema : Registrar Novo Utilizador	56
6.65 Diagrama de sequência : registrarUtilizador(username,password)	56
6.66 Diagrama de sequência Subsistema : Alterar Conta de Utilizador	56
6.67 Diagrama de sequência : alterarUtilizador(username,novapassword)	57
6.68 Diagrama de sequência Subsistema : Remover Utilizador	57
6.69 Diagrama de sequência : removerUtilizador(username)	57
6.70 Diagrama de sequência : margemLucro()	58
6.71 Diagrama de sequência : saltarParaDia(data)	58
6.72 Diagrama de sequência : removerCliente(codCliente)	58
6.73 Diagrama de sequência : existeCliente(bi)	59
6.74 Diagrama de sequência : getCodigosClientesActivos()	60
6.75 Diagrama de sequência : getCodigosClientesInactivos()	60
6.76 Diagrama de sequência Subsistema : Autenticação	60
6.77 Diagrama de sequência : autenticar(String username,String password)	61
6.78 Diagrama de sequência Subsistema : Remover serviço a fornecedor	61
6.79 Diagrama de sequência : removeServico(codservico)	62
6.80 Diagrama de sequência : removeServicosTipoServico(codtipo)	63
6.81 Diagrama de sequência : removeServicoCategoria(codcat)	63
6.82 Diagrama de sequência : removeTiposServicoCategoria(codcat)	63
6.83 Diagrama de sequência Subsistema : Inserir nova categoria de servico	64
6.84 Diagrama de sequência : inserirCategoria(nome)	64
6.85 Diagrama de sequência : existeCategoria(nome)	65
6.86 Diagrama de sequência Subsistema : Remover Categoria	65

6.87 Diagrama de sequência : removerCategoria(codcategoria)	65
6.88 Diagrama de sequência : existeTipoServico(nome)	66
6.89 Diagrama de sequência : insereTipoServico(nome,codcat)	66
6.90 Diagrama de sequência : removeTipoServico(codtipo)	67
6.91 Diagrama de sequência : prestaSe rvico(codforn,codtipo)	67
6.92 Diagrama de sequência : temCategoria(codforn,codcat)	67
6.93 Diagrama de sequência : inserirCategoriaAFornecedor(codforn,codcat)	68
6.94 Diagrama de sequência Subsistema : Inserir Novo Servico A Fornecedor . . .	68
6.95 Diagrama de sequência : inserirServicoFornecedor(codforn,codtipo,codcat,desc,precoph) 68	
6.96 Diagrama de sequência Subsistema : Alterar Serviços	69
6.97 Diagrama de sequência : alteraServico(String codservico,String desc,float precoph)	69
6.98 Diagrama de sequência Subsistema : Registrar Novo Cliente	69
6.99 Diagrama de sequência : registrarNovoCliente(nome,contacto,email,idade,NIF,NIB,BI) 70	
6.100Diagrama de sequência : gerarCodigoTipoServico()	70
6.101Diagrama de sequência : gerarCodigoCategoria()	70
6.102Diagrama de sequência : gerarCodigoServico()	71
6.103Diagrama de sequência : getListaCategoriasArray()	72
6.104Diagrama de sequência : getListaClientesArray()	72
6.105Diagrama de sequência : getListaContratosArray()	72
6.106Diagrama de sequência : getListaFornecedoresArray()	72
6.107Diagrama de sequência : getListaTiposServicoArray()	73
6.108Diagrama de sequência : getListaTiposServicoArray()	73
6.109Diagrama de Classes	74

1

Introdução

1.1 Motivação e Objectivos

A motivação principal para este projecto foi a inclusão num ambiente minimamente realista de produção de software efectuada de forma correcta. Ou seja, passando primeiro por uma fase intensiva de especificação usando UML e, no final, traduzindo a especificação em JAVA.

Pretendeu-se desenvolver um sistema de software para uma gestão de condomínios, onde são prestados diversos tipos de serviços. O sistema desenvolvido, designado por Gere-ComSaber, efectua toda a gestão correspondente aos diversos serviços que fazem parte da oferta comercial, bem assim como registar e manter a informação dos clientes e a sua história de contractos com a empresa.

A proposta de modelação e respectivo desenvolvimento numa linguagem de programação orientada aos objectos, o JAVA, foi faseada no tempo. Desta forma, criou-se uma situação mais próxima da realidade que permitiu simular o comportamento de uma equipa de projecto numa situação real.

1.2 Estrutura do Relatório

Capítulo 1º - Introdução O leitor é introduzido ao objectivo do projecto, um sistema de gestão de condomínios.

Capítulo 2º - Requisitos É relatada a fase de levantamento de requisitos, iniciada pela especificação de um modelo de domínio, seguida da enumeração ilustrada dos casos de uso.

Capítulo 3º - Arquitectura Nesta secção é relatada a tomada de decisões no desenho da arquitectura de classes e interacção, assim como as iterações efectuadas.

Capítulo 4º - Alterações à Modelação Inicial Nesta secção faz-se uma pequena referência a Use Cases adicionados ao Modelo inicial para optimização do mesmo.

Capítulo 5º - Conclusão No capítulo final são tecidos alguns comentários relativos ao trabalho efectuado e motivação para trabalho futuro.

2

Requisitos

Produto exigido

“ Considere-se a área da actuação das empresas que fazem gestão do serviço em condomínios fechados. Nessas situações a empresa GereComSaber, fica encarregada de assegurar que os diversos trabalhos a efectuar em casa dos condóminos seja efectuado. Existem diversos tipos de serviço, a saber:

- Limpeza da casa;
- Cozinha;
- Jardinagem;
- Manutenção dos aparelhos eléctricos;
- Manutenção dos veículos automóveis;
- Etc.

A empresa GereComSaber, é apenas uma empresa intermediária, que não presta directamente os serviços. Para satisfazer as necessidades do cliente possui uma carteira de fornecedores que sub-contrata e que são os responsáveis pela realização dos serviços. É possível que para um determinado tipo de serviço exista mais do que um fornecedor disponível. Cabe à empresa GereComSaber escolher qual o fornecedor que indica ao cliente.

Apesar de as empresas fornecedoras poderem prestar o mesmo tipo de serviço, (ex: jardinagem) podem ter designações diferentes para os seus níveis de serviço.

Empresa 1:

- *Área relvada até 250 m²*
- *Área relvada até 500 m²*
- *Etc.*

Empresa 2:

- *Área relvada até 150 m²*
- *Área relvada até 400 m²*
- *Etc.*

As empresas fornecedoras para cada tipo de serviço, incluem várias actividades. Para o exemplo anterior, a Empresa 1, pode ter como actividades: “cortar relvado”, “tirar ervas daninhas”, “arranjar buxos”, etc.

Cada uma destas actividades tem um valor monetário distinto e o cliente pode escolher o serviço “à lista” que pretende. Cada empresa fornecedora de serviços paga uma comissão à GereComSaber. Estas empresas fornecedoras, tendo em conta a conjuntura actual, cobram o montante no início do período.

Ao longo do período (ano) o cliente pode ir acrescentando, ou retirando, serviços. Uma das grandes dificuldades que actualmente a GereComSaber enfrenta é no registo e análise destes movimentos de alterações. É também importante perceber que um acréscimo, ou diminuição de serviço, implica ter de efectuar acertos com os clientes:

- *Um acréscimo implica pagamento extra*
- *Um decréscimo implica um estorno*

A GereComSaber, pretende ter um sistema informático que permita efectuar a gestão da informação anterior apresentada.

Objectivos Macro:

- *Suportar vários fornecedores com ofertas diferentes;*
- *Gestão eficiente dos serviços contratados;*
- *Gestão das alterações ao serviço, ao longo do período;*

- *Previsão dos fluxos financeiros (acréscimos e estornos);*
- *Cálculo da margem da GereComSaber;*

”

2.1 Levantamento de Requisitos

A primeira fase de especificação de um projecto passa por “espremer” o que é requerido pelo cliente, de forma a sumarizar sucinta e directamente o que por ele foi exigido. Desta forma, procedemos a um levantamento de requisitos enumerados.

Requisitos do utilizador (funcionário e gerente)

- Adicionar Cliente (gerar código)
- Alterar dados pessoais do Cliente
- Aceder à conta de um Cliente (introduzindo o código do cliente)
- Adicionar serviços a um Cliente
- Cancelar serviços a um Cliente
- Estender o prazo de um serviço a um Cliente
- Reduzir o prazo de um serviço a um Cliente

Requisitos do utilizador (apenas gerente)

- Realizar todas as funcionalidades acessíveis ao funcionário
- Gerir toda a informação relativa aos tipos de serviço que a empresa pode fornecer (adicionar, remover, alterar)
- Gerir toda a informação relativa aos fornecedores sub-contratados
- Remover Cliente (marca-lo como inactivo)
- Consultar lista de clientes activos e inactivos
- Activar serviço a um cliente
- Calcular a margem de lucro da GereComSaber
- Alterar a data do sistema

Requisitos do utilizador (cliente)

- Contratar novo serviço
- Cancelar serviço
- Consultar ficha
- Consultar lista de serviços disponíveis
- Estender ou reduzir o prazo de um serviço contratado

Requisitos do utilizador (administrador)

- Adicionar utilizador de sistema
- Remover utilizador de sistema
- Alterar conta de um utilizador

Requisitos do Sistema

- Quatro tipos de utilizadores: funcionário, gerente, cliente e administrador de sistema
- Manter registo de todos os clientes, funcionários, fornecedores e serviços
- Possibilidade de alterar todos os registos
- Manter registo diferenciado de serviços activos, inactivos e pendentes
- Manter registo de serviços cancelados e extendidos
- Manter registo de todos os serviços e categorias de serviço
- Manter registo de estornos e pagamentos extra
- Actualizar a sua data de modo a prever os fluxos financeiros

2.2 Modelo de Domínio

Depois de uma fase de discussão com o cliente, os professores, foi esboçado um modelo de domínio. Nesta fase são analisadas e apontadas as entidades presentes no sistema, assim como a relação entre elas. O modelo pode ser visto na 2.1.

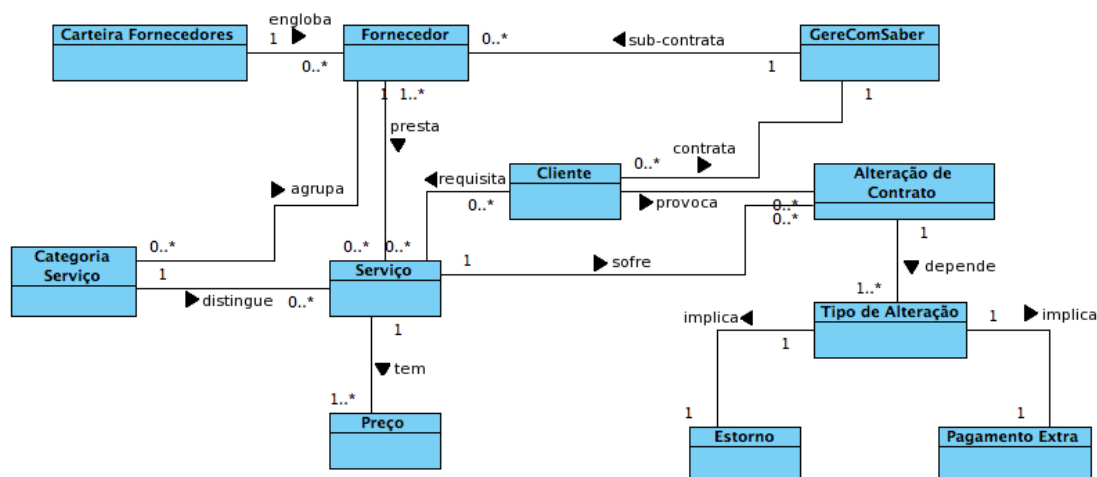


Figura 2.1: Modelo de domínio

2.3 Use Cases

Na fase seguinte, num contexto real, finaliza-se o contacto com o cliente esboçando a aplicação final na forma de diagramas de Use Cases. O desenvolvimento dos Use Cases foi realizado em UML, usando o Visual Paradigm.

Na identificação dos Uses Cases foram seguidas as indicações propostas no enunciado do trabalho a par com as decisões que o grupo, segundo a sua perspectiva, foi tomando relativamente à melhor forma de implementar o sistema.

No diagrama principal o utilizador tem a possibilidade de se autenticar como funcionário, gerente, cliente ou administrador de sistema, fazendo login e inserindo a respectiva password.

Registando-se como funcionário, o utilizador tem acesso a um Menu de Funcionário onde pode registar um novo cliente, adicionar serviços a um determinado cliente, alterar dados pessoais de um cliente, cancelar um serviço a pedido de um cliente, consultar a ficha e o

histórico de serviços de um cliente e estender ou reduzir o prazo de duração de um serviço a pedido de um cliente.

Registando-se como gerente, o utilizador tem acesso a um Menu de Gerente e também ao Menu de Funcionário. No menu de gerente ele pode escolher sub-menus dependendo do tipo de operação que deseja efectuar. Escolhendo o menu Gerir Clientes, o gerente tem a possibilidade de cancelar um contrato de um cliente, isto é, marcá-lo como inactivo no sistema. Pode também consultar a lista de clientes que estão activos ou inactivos e também activar um serviço, de um cliente, que esteja pendente. Escolhendo o menu Gerir Carteira Fornecedores, o gerente tem a possibilidade de adicionar um registo de um novo fornecedor ao sistema, alterar o registo de um fornecedor e remover um, caso deseje. Tem também a hipótese de listar os fornecedores activos no sistema. Escolhendo o menu Gerir Serviços, o gerente pode manipular toda a informação relativa a categorias de serviço (ex: Jardinagem) no sistema. Isto é, adicionar novas categorias e remover categorias. Aparte disto, o gerente pode inserir novos serviços a um fornecedor, alterar os serviços por eles prestados e também remover serviços aos fornecedores. Escolhendo o menu Financeiro, o gerente pode calcular a margem de lucro da GereComSaber. No menu inicial do gerente, ele pode ainda actualizar a data do sistema.

Registando-se como cliente, o utilizador tem a possibilidade de contratar serviços, cancelá-los, estender ou reduzir os seus períodos de duração e ainda consultar a lista de serviços disponíveis de contratar assim como, claro, aceder à sua ficha informativa.

Registando-se como administrador de sistema, o utilizador tem a possibilidade adicionar novos utilizadores ao sistema de software, remover utilizador e alterar os seus registos.

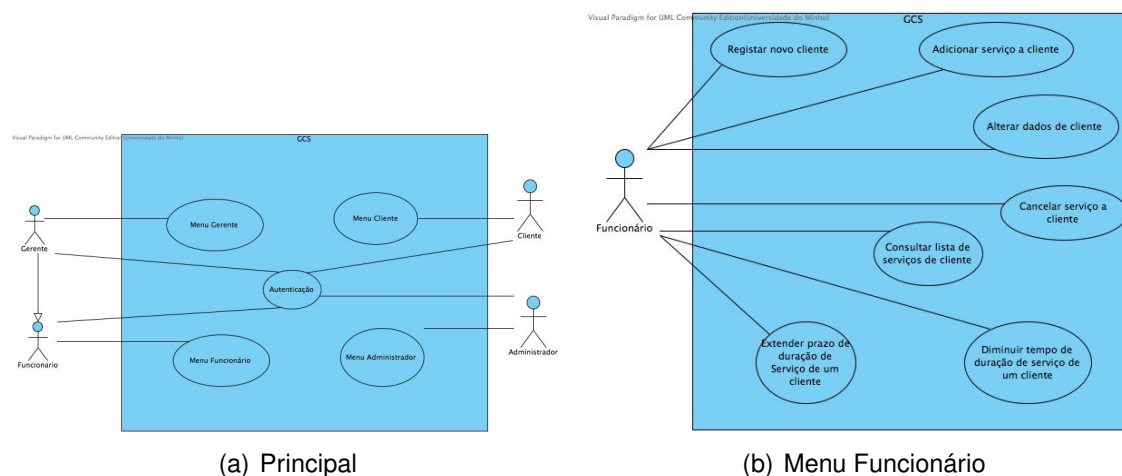


Figura 2.2: Menu Funcionário

3

Arquitectura

Finalizada a fase de diálogo entre o cliente e o engenheiro de software, segue-se a fase de desenho da arquitectura que o sistema necessita possuir de forma a facultar todas as funcionalidades requeridas.

Pelo facto de o paradigma mais apropriado para o projecto ser OO, começou-se por aperfeiçoar o modelo de domínio, obtendo assim uma primeira versão do diagrama de classes, que representa os objectos, e a sua constituição, que vão interagir de forma a obter o sistema pretendido.

O sistema começou como uma black box, isto é, não se sabia muito bem quais os sub-sistemas que o definiam e como eles iriam estar estruturados. Para se perceber isso fizeram-se os Diagramas de Sequência de Subsistema em que dependendo da funcionalidade, se começou a definir vários packages (ou sub-sistemas) que tratariam das suas respectivas funcionalidades. Foram definidos 5 sub-sistemas sendo eles:

- Clientes
- Contratos
- Servicos
- Fornecedores
- Dados

O sub-sistema **Clientes** trata de toda a gestão relativa a fichas de clientes, inserir novos registos de clientes no sistema, desactivá-los, referenciar contratos no cliente etc. O sub-sistema **Contratos** tem como objectivo criar, manter, alterar e cancelar contratos efectuados

por clientes. É o responsável por construir um novo contrato, associar-lhe um serviço, um cliente e tratar de calcular estorno ou pagamentos extra, se necessário. O sub-sistema **Servicos** é o responsável por gerir e manter o leque de serviços disponíveis no sistema. Cria novos registos, associa-lhes tipo de serviço, categoria e fornecedor, altera registos e remove-os. O sub-sistema **Fornecedores** gere a informação relativa aos fornecedores sub-contratados pela GereComSaber. Funciona um pouco como o sub-sistema Clientes na medida em que faz um gestão semelhante mas, neste caso, relativa a fornecedores. É o sub-sistema responsável por referenciar serviços aos fornecedores. Por fim temos o sub-sistema **Dados** que foi criado para gerir toda a informação no sistema. Os objectos existem neste sub-sistema sendo que nos outros apenas são usados os códigos destes objectos para os referenciar. Este package recebe pedidos de objectos para que os outros packages lhes façam alterações, pedidos de inserção e remoção de novos dados. O package **Dados** carrega toda a informação contida na base de dados para as estruturas de dados e submete toda a informação à base de dados.

Depois de desenhada, em linhas gerais, a estrutura do sistema, prosseguiu-se a especificação detalhada da interacção entre os objectos nele presentes, originando os Diagramas de Sequência de Implementação.

Finalizadas a primeiras fazes, aprofundou-se o detalhe do diagrama de classes, cruzando a informação comportamental dos objectos presentes nos diagrama de sequência e estrutural presente na primeira fase do diagrama de classes.

No final, o digrama de classes final e os diagramas de sequência - um para cada funcionalidade requerida - definem detalhadamente a arquitectura do sistema. Desta forma, desambigua-se qualquer promenor presente na implementação dos requisitos e propociona-se uma fácil e rápida implementação.

Em anexo podem ser vistos todos os diagramas usados na fase de especificação.

3.1 Diagrama de Classes

A definição estrutural do sistema em classes foi um processo iterativo. Em primeiro lugar, começamos por definir o esqueleto geral da arquitectura, anotando as classes que evidentemente iriam constar no diagrama. De seguida, e à medida que fomos avançando na elaboração dos diagramas de sequência, os aperfeiçoamentos foram surgindo. No final o sistema ficou dividido em 5 subsistemas representados em pacotes de classes, são eles: **Clientes, Servicos, Contratos, Fornecedores e Dados**.

Em cada pacote de classes consta uma classe dotada da gestão dos objectos em questão (e.g. adicionar/remover/editar clientes). Esta abordagem à modelação visa uma melhor organização lógica e encapsulamento das decisões das estruturas de dados utilizadas. No pacote **Dados** toda a informação relativa a determinada entidade é guardada numa classe respectiva ListaX que contém HashMaps para guardar a informação.

3.2. DIAGRAMAS DE SEQUÊNCIA

De notar que fora do package Dados, as classes que tiverem de conter determinados objectos apenas conhecem os seus códigos. Por exemplo, na classe Cliente a ListaServicos-Activos é uma lista de códigos de Contratos. Para manipular o objecto respectivo o package tem de comunicar com o package Dados e pedir-lhe esse mesmo objecto.

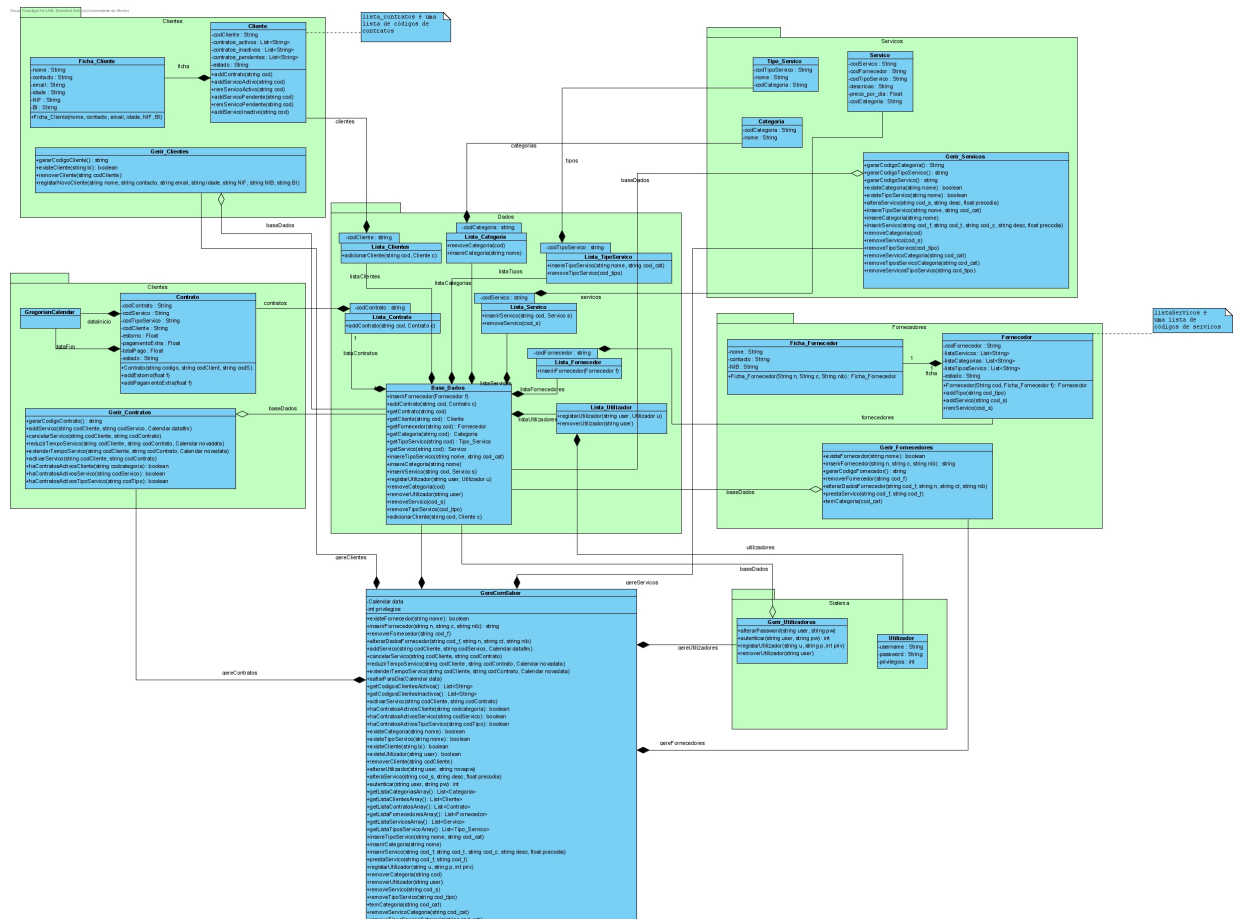


Figura 3.1: Diagrama de classes

3.2 Diagramas de Sequência

O sistema GereComSaber é constituído por três camadas: Presentation, Business, Data. Foram criados os diagramas de sequência que modelam a camada Business, assim como as operações que envolvem o sucesso dos use cases. A vantagem da modelação da camada Business é a de ser possível aperceber que o sistema está dividido em sub-sistemas que agrupam classes com a mesma semântica lógica.

Os respectivos diagramas foram criados com base nas interações das classes que constituem a camada, assim como nos algoritmos que realizam as operações. A modelação

3.2. DIAGRAMAS DE SEQUÊNCIA

da camada Presentation não foi realizada e a camada de Business foi modelada em duas fases:

- 1ª Numa primeira fase, assumiu-se que o sistema era uma caixa negra e construíram-se os diagramas de sequência respectivos.
- 2ª Numa segunda fase dividiu-se o sistema em subsistemas alterando os diagramas de sequência nos quais se mostra as interações entre as várias classes que constituem o sistema. Nesta fase, houve uma ligação entre os vários diagramas de sequência e o diagrama de classes, ou seja, houve um refinamento dos vários diagramas. Também foi possível perceber que houve algumas falhas na primeira fase e, como se trata de um processo iterativo, alterámos os uses cases necessários.

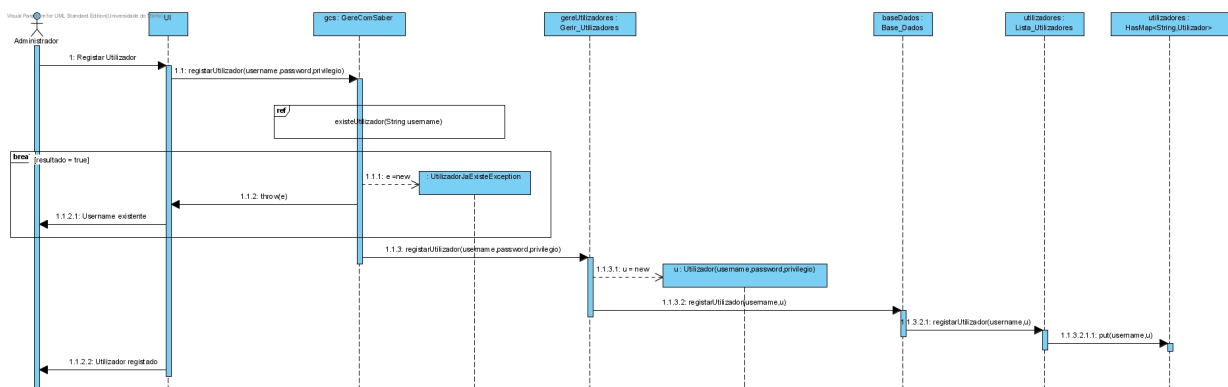


Figura 3.2: Diagrama de sequência para o registo de um novo Utilizador

4

Alterações à Modelação Inicial

Nesta segunda fase ao ser definida a Arquitectura do sistema e as interacções entre os objectos, assim como os algoritmos, o grupo apercebeu-se que teriam de ser adicionados novos Use Cases. Eles foram:

- Inserir Tipo de Serviço
- Remover Tipo de Serviço

Esta modificação foi necessária pois um serviço tem uma categoria associada, p ex: Jardinagem, e tem também uma actividade. Ela pode ser Cortar Relva, Apanhar Ervas Daninhas, etc Na nossa modelação inicial esta actividade tomava um valor alfabético introduzido na criação de um serviço. Ora, "Cortar Relva"ou "cortar relvado"iriam ser actividades diferentes pois a string é diferente mas referem-se à mesma actividade logo era preciso controlar isto. Com a adição do atributo Tipo de Serviço aos serviços passou a ser único de cada Serviço a descrição em que cada fornecedor propoe mais detalhes sobre o serviço.

5

Conclusões e Trabalho Futuro

A maior parte do tempo dedicado ao projecto durante a primeira fase foi aplicado na especificação do projecto e na coordenação intra-grupo. O aspecto mais difícil foi perceber por completo o funcionamento e a estrutura do sistema a implementar e as suas funcionalidades do que propriamente detalhar essas funcionalidades.

Nesta segunda fase, houve bastante trabalho que não foi utilizado pois começou-se um pouco a fazer os diagramas sem ter bem definida e bem presente a arquitectura do sistema. Sendo que quando se mudou significativamente esta arquitectura, os diagramas tiveram de ser todos refeitos e bastante tempo se perdeu com isso. Com isso facilmente se constatou que o processo deve ser iterativo e cada passo deve ficar completamente definido e não haver dúvidas, pois uma alteração num passo anterior leva a bastante perda de tempo a corrigir os passos seguintes.

O projecto, tal como é suposto, tem ajudado os elementos do grupo a ganhar traquejo na modelagem de sistemas de informação usando UML e a aprender a trabalhar em grupo num projecto razoavelmente grande. Ganhou-se consciência de uma vez por todas que a especificação é obrigatória no desenvolvimento de sistemas de informação estáveis.

Prevê-se que na fase seguinte, e última, se implemente o sistema proposto. A corrente fase foi cumprida dentro dos prazos e seguindo uma metodologia adequada.



Anexos

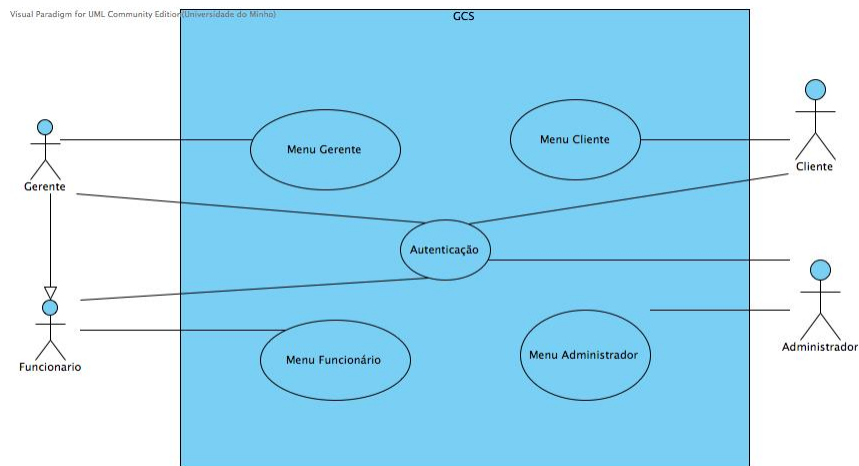


Figura 6.1: Diagrama Menu Principal

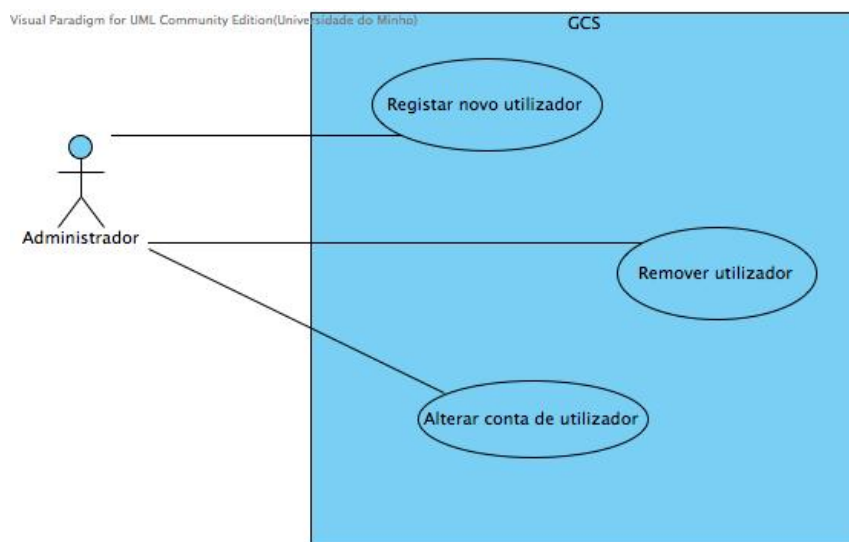


Figura 6.2: Diagrama Menu Administrador

Super Use Case	Menu Administrador	
Author	João	
Date	24/Nov/2009 12:29:31	
Brief Description	O Administrador pretende alterar conta de utilizador de sistema	
Preconditions	O utilizador deve estar autenticado com privilégios de Administrador	
Post-conditions		
Flow of Events		Actor Input
	1	Alterar conta de utilizador de sistema
	2	
	3	Insere código do utilizador
	4	
	5	Altera os campos que deseja
	6	Confirma a alteração dos dados
	7	
	8	
	9	
		System Response
Alternative Flow of Events 1 7 - Os campos obrigatórios não estão todos preenchidos	1	Verifica se todos os campos obrigatórios estão preenchidos
	2	Verifica se os dados são consistentes com o formulário
		Informa que as alterações foram efectuadas com sucesso
Alternative Flow of Events 2 8 - Dados inconsistentes com o formulário	1	Verifica se os campos obrigatórios não estão todos preenchidos
	2	Volta ao passo 4.
Alternative Flow of Events 3 9 - Dados inconsistentes com o formulário	1	Verifica se os dados não são consistentes com o formulário
	2	Volta ao passo 4.

Figura 6.3: Descrição Use Case: Alterar Conta de Utilizador

Super Use Case	Menu <u>Administrador</u>	
Author	<u>Joao</u>	
Date	<u>24/Nov/2009 12:02:40</u>	
Brief Description	<u>O Administrador pretende efectuar o registo de um novo utilizador.</u>	
Preconditions	<u>O utilizador deve estar autenticado com privilégios de Administrador.</u>	
Post-conditions	<u>Número de utilizadores do sistema = Número de utilizadores do sistema + 1.</u>	
Flow of Events	Actor Input	System Response
	1 <u>Inserir novo utilizador do sistema.</u>	
	2	<u>Pergunta que tipo de privilégios irá ter o novo utilizador.</u>
	3 <u>Escolhe privilégios de novo utilizador.</u>	
	4	<u>Mostra formulário de registo de novo utilizador dependendo do tipo de privilégios escolhidos.</u>
	5 <u>Preenche os campos.</u>	
	6 <u>Confirma os dados</u>	
	7	<u>Verifica se os campos obrigatórios estão todos preenchidos.</u>
	8	<u>Verifica se os dados estão consistentes com o formulário.</u>
	9	<u>Cria novo registo de utilizador no sistema.</u>
	10	<u>Informa que o novo registo foi criado com sucesso.</u>
Alternative Flow of Events	Actor Input	System Response
<u>7 - Os campos não estão todos preenchidos.</u>	1	<u>Informa que houve uma falha no preenchimento do formulário.</u>
	2	<u>Volta ao passo 4.</u>
Alternative Flow of Events 2	Actor Input	System Response
<u>8 - Os dados não são consistentes com o formulário.</u>	1	<u>Informa que os dados estão incorrectos.</u>
	2	<u>Volta ao passo 4.</u>

Figura 6.4: Descrição Use Case: Registrar novo Utilizador

Super Use Case	Menu <u>Administrador</u>	
Author	<u>Joao</u>	
Date	<u>24/Nov/2009 12:15:42</u>	
Brief Description	<u>O Administrador pretende remover um utilizador do sistema.</u>	
Preconditions	<u>O utilizador deve estar autenticado com privilégios de Administrador.</u> <u>Existe pelo menos um utilizador de sistema.</u>	
Post-conditions	<u>Utilizadores do sistema = Utilizadores do sistema - 1</u>	
Flow of Events	Actor Input	System Response
	1 <u>Remover utilizador do sistema.</u>	
	2	<u>Pede código do utilizador a remover</u>
	3 <u>Inserir código do utilizador.</u>	
	4	<u>Verifica se o utilizador existe.</u>
	5	<u>Remove o utilizador escolhido do sistema.</u>
Alternative Flow of Events 1 <u>4 - O utilizador não existe no sistema.</u>	6	<u>Informa que o utilizador foi removido com sucesso.</u>
	Actor Input	System Response
	1	<u>Informa que o utilizador não existe.</u>
	2	<u>Volta ao passo 2.</u>

Figura 6.5: Descrição Use Case: Remover Utilizador

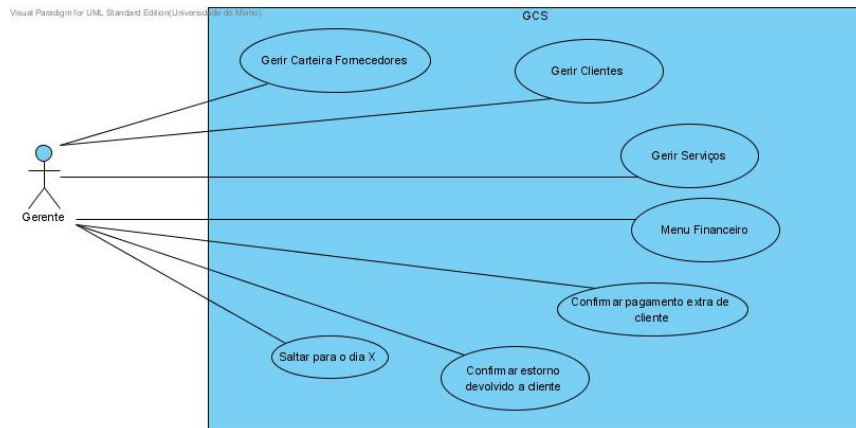


Figura 6.6: Diagrama Menu Gerente

Super Use Case	Gerir Serviços	
Author	ase	
Date	24/Nov/2009 14:49:30	
Brief Description	O gerente pretende actualizar o seu sistema para um determinado dia	
Preconditions	O utilizador está autenticado com privilégios de gerente	
Post-conditions	O dia do sistema é actualizado	
Flow of Events	Actor Input	System Response
	1 Saltar para o dia X	
	2	Mostra um formulário para introdução de data
	3 Preenche formulário	
	4	Vai buscar as informações de todos os clientes no sistema e agrupa-os
	5	Percorre essa lista e verifica se data de fim de todos os serviços activos é anterior à nova data do sistema
	6	Marca serviços activos como serviços inactivos

Figura 6.7: Descrição Use Case: Saltar para o dia X

Super Use Case	Menu <u>Gerente</u>	
Author	<u>João Fernandes, João Gonçalves, José Pereira, Ricardo Nogueira, Vasco Grilo</u>	
Date	1/Dez/2009 13:50:19	
Brief Description	O <u>gerente</u> pretende <u>marcar no sistema</u> que um <u>cliente</u> já <u>pagou o pagamento extra devido</u> .	
Preconditions	O <u>utilizador</u> está autenticado com <u>privilégios de gerente</u> . A <u>lista de clientes</u> com <u>pagamentos extra em atraso</u> tem <u>pelo menos 1 cliente</u> .	
Post-conditions	<u>Numero clientes</u> com <u>pagamentos extra</u> a <u>efectuar decrementou</u> em 1 unidade.	
Flow of Events		Actor Input
	1	<u>Confirmar pagamento extra de cliente</u>
	2	
	3	<u>Escolhe cliente</u>
	4	
	5	<u>Confirma pagamento</u>
	6	
	7	
	8	
		System Response
		<u>Mostra lista de clientes com pagamentos extra a efectuar</u>
		<u>Mostra ficha informativa</u>
		<u>Actualiza o registo de pagamento extra na ficha do cliente para zero</u>
		<u>Retira o cliente da lista de clientes com pagamentos extra a efectuar</u>
		<u>Informa que tudo correu com sucesso</u>

Figura 6.8: Descrição Use Case: Confirmar Pagamento Extra de Cliente

Super Use Case	Menu <u>Gerente</u>	
Author	<u>João Fernandes, João Gonçalves, José Pereira, Ricardo Nogueira, Vasco Grilo</u>	
Date	1/Dez/2009 13:53:44	
Brief Description	O <u>gerente</u> <u>devolveu o dinheiro</u> a um <u>cliente</u> e pretende <u>actualizar essa informação no sistema</u> .	
Preconditions	O <u>utilizador</u> está autenticado com <u>privilégios de gerente</u> . A <u>lista de clientes</u> com <u>estorno a receber</u> tem <u>pelo menos 1 cliente</u> .	
Post-conditions	<u>Número de clientes</u> na <u>lista de clientes</u> a <u>receber estorno</u> <u>decrementou</u> em 1 unidade.	
Flow of Events		Actor Input
	1	<u>Confirmar estorno devolvido a cliente</u>
	2	
	3	<u>Escolhe cliente</u>
	4	
	5	<u>Confirma devolução</u>
	6	
	7	
	8	
		System Response
		<u>Mostra lista de clientes com dinheiro a ser devolvido</u>
		<u>Mostra ficha informativa</u>
		<u>Actualiza o registo na ficha do cliente de estorno para zero</u>
		<u>Retira cliente da lista de clientes a receber estorno</u>
		<u>Informa que tudo correu com sucesso</u>

Figura 6.9: Descrição Use Case: Confirmar Estorno devolvido a Cliente

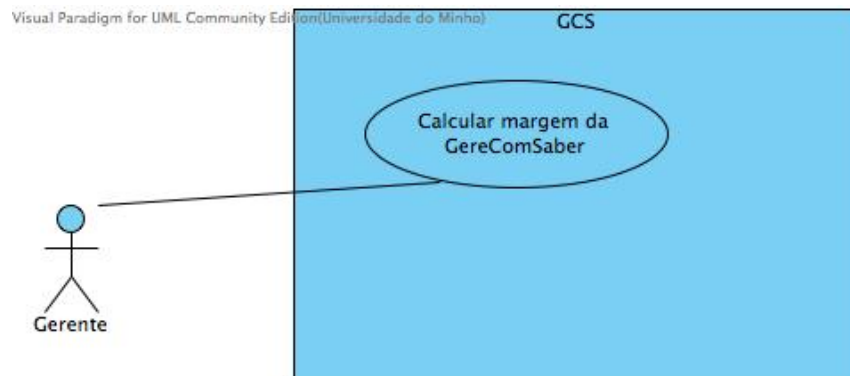


Figura 6.10: Diagrama Menu Financeiro

Super Use Case	Menu Financeiro	
Author	vasco	
Date	20/Nov/2009 16:51:23	
Brief Description	O gerente pretende calcular a margem de lucro da GereComSaber. O utilizador está autenticado com privilégios de gerente.	
Preconditions	Existem fornecedores no sistema. Existem clientes no sistema. Existem serviços no sistema.	
Post-conditions	Registo de margem de lucro actualizado. Registo de total estorno actualizado. Registo de total pagamentos extra actualizado. Lista de dinheiro a devolver a cada fornecedor actualizada.	
Flow of Events	Actor Input	System Response
	1 Calcular margem da GereComSaber.	
	2	Agrupa todos os clientes numa estrutura de dados auxiliar.
	3	Agrupa todos os fornecedores numa lista.
	4	Para cada cliente, vai incrementando os totais de estorno, pagamentos extra e total pagamentos com base nas informações da ficha de cada cliente dos serviços activos.
	5	Subtrai o estorno ao total de pagamentos.
	6	Soma ao novo total os pagamentos extra.
	7	Soma PagamentosExtra às receitas totais e subtrai Estorno.
	8	Associa a cada fornecedor da lista os valores monetários dos serviços prestados.
	9	Para cada valor a receber pelos fornecedores, calcula a taxa que fica para a GereComSaber.
	10	Actualiza o registo dessa margem de lucro.
	11	Mostra uma lista contendo os fornecedores e o dinheiro a devolver a cada um.
	12	Apresenta o total de lucro da GereComSaber.

Figura 6.11: Descrição Use Case: Calcular Margem da GereComSaber

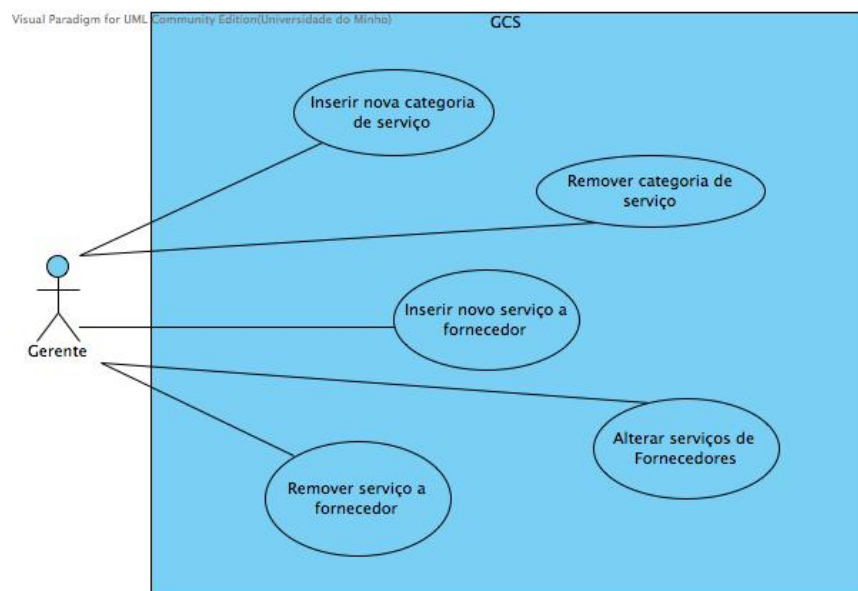


Figura 6.12: Diagrama Gerir Serviços

Super Use Case	Gerir Serviços	
Author	Noor	
Date	Nov 20, 2009 5:41:27 PM	
Brief Description	O utilizador pretende adicionar um novo serviço a um determinado fornecedor.	
Preconditions	O utilizador está autenticado com privilégios de gerente. Existem categorias no sistema. Existem fornecedores no sistema.	
Post-conditions	Número de serviços de fornecedor = Número de serviços de fornecedor +1	
Flow of Events		Actor Input
	1	Inserir novo serviço a fornecedor.
	2	
	3	
	4	Escolhe fornecedor.
	5	
	6	Escolhe categoria.
	7	
	8	
	9	Completa formulário.
	10	
	11	
	12	
	13	
	14	
Alternative Flow of Events 1 11-Campos obrigatórios em falta		System Response
	1	Informa campos obrigatórios em falta.
Alternative Flow of Events 2 12-Dados não são consistentes	2	Volta ao passo 9
Alternative Flow of Events 3 13 - Fornecedor não tem serviços na categoria escolhida		System Response
	1	Informa dados mal inseridos.
	2	Volta ao passo 9
		Actor Input
	1	
	2	
		System Response
		Cria categoria escolhida no registo do fornecedor.
		Adiciona novo serviço a essa categoria.

Figura 6.13: Descrição Use Case: Inserir Novo Serviço a Fornecedor

Super Use Case	Gerir Carteira de Fornecedores	
Author	João	
Date	20/Nov/2009 16:54:47	
Brief Description	O Gerente pretende alterar os serviços de um fornecedor	
Preconditions	Existem fornecedores no sistema O utilizador está autenticado com privilégios de gerente	
Post-conditions	O registo do serviço escolhido é actualizado	
Flow of Events		Actor Input
	1	Alterar dados de serviço de um fornecedor
	2	
	3	Escolhe um fornecedor da lista
	4	
	5	Escolhe categoria de serviço
	6	
	7	Escolhe serviço
	8	
	9	Altera os dados
	10	
		System Response
		Mostra uma lista com os fornecedores
		Mostra categorias de serviços prestados pelo fornecedor escolhido
		Mostra serviços dentro da categoria que o fornecedor presta
		Mostra dados sobre o serviço a alterar
		Regista as mudanças efectuadas

Figura 6.14: Descrição Use Case: Alterar Serviços de Fornecedores

Super Use Case	chinelô	
Author	chinelô	
Date	9/Dez/2009 18:02:54	
Brief Description	O utilizador autentica-se no sistema	
Preconditions	O sistema tem utilizadores registados	
Post-conditions	O utilizador autentica-se no sistema e obtém os privilégios associados ao seu tipo de registo	
Flow of Events		Actor Input
	1	Autenticar
	2	
	3	Preenche formulário
	4	
	5	
	6	
	7	
	8	
		System Response
		Mostra formulário de autenticação(username e password)
		Verifica se todos os campos obrigatórios estão preenchidos
		Procura o username inserido no sistema
		Compara a password inserida com a password existente no sistema associada ao username escolhido
		Identifica os privilégios associados ao tipo de conta acedida e atribui os mesmos ao utilizador
		Informa o utilizador do sucesso da sua autenticação e tipo de privilégios que lhe foram concedidos
Alternative 1		
4 - Campos vazios		Actor Input
	1	
		System Response
		Informa o utilizador que existem campos obrigatórios não preenchidos
		Volta ao passo 2
Alternative 2		
5 - username não existente		Actor Input
	1	
	2	
		System Response
		Informa o utilizador que o username introduzido não se encontra registado
		Volta ao passo 2
Alternative 3		
6 - password errada		Actor Input
	1	
		System Response
		Informa o utilizador que a password inserida não corresponde à password registada no sistema para o username inserido
		Volta ao passo 2

Figura 6.15: Descrição Use Case: Autenticação

Super Use Case	Gerir Serviços		
Author	Noor		
Date	Nov 23, 2009 10:53:24 PM		
Brief Description	O utilizador pretende remover um serviço a um fornecedor		
Preconditions	O utilizador tem que estar autenticado com privilégios de Gerente. Existem serviços registados no sistema		
Post-conditions	Número de serviços do sistema = Número de serviços do sistema - 1		
Flow of Events		Actor Input	System Response
	1	Remover serviço a fornecedor	
	2		Mostra lista de fornecedores
	3	Escolhe fornecedor	
	4		Mostra lista de categorias nos quais o fornecedor escolhido opera
	5	Escolhe categoria	
	6		Mostra lista de serviços dessa categoria que o fornecedor presta
	7	Escolhe serviço	
	8		Verifica se existem contratos activos do serviço escolhido
	9		Pede confirmação da remoção do serviço
	10	Confirma remoção	
	11		Remove o serviço da lista de serviços do fornecedor
	12		Remove serviço do sistema
	13		Informa o utilizador do sucesso da remoção
Alternative 1 9 - Remoção cancelada		Actor Input	System Response
	1		Informa utilizador do cancelamento da remoção
	2		Regressa ao passo 2
Exception 8 - Existem contratos activos do serviço a remover		Actor Input	System Response
	1		Informa o utilizador que existem contratos activos do serviço a remover

Figura 6.16: Descrição Use Case: Remover Serviço a Fornecedor

Super Use Case	Gerir Serviços	
Author	Noor	
Date	Nov 20, 2009 5:23:56 PM	
Brief Description	O gerente pretende adicionar uma nova categoria de serviços à sua empresa.	
Preconditions	O utilizador está autenticado com privilégios de gerente.	
Post-conditions	Numero de categorias totais = Numero de categorias totais +1	
Flow of Events		Actor Input
	1	Inserir nova categoria.
	2	
	3	Preenche formulário
	4	
	5	
	6	
		System Response
		Mostra formulário de inserção de categoria.
		Verifica validade dos dados inseridos e preenchimento de campos obrigatórios.
		Verifica que a categoria que se pretende inserir não existe não está registada no sistema.
		Adiciona nova categoria ao sistema.
Alternative Flow of Events 1 4- Dados invalidos ou campos obrigatórios não preenchidos.		Actor Input
	1	
	2	
		System Response
		Informa utilizador do erro sucedido.
		Voltar ao passo 2
Alternative 2 5 - Categoria existente.		Actor Input
	1	
	2	
		System Response
		Informa utilizador que a categoria que inseriu já se encontra registada.
		Volta ao passo 2

Figura 6.17: Descrição Use Case: Inserir Nova Categoria de Serviço

Super Use Case	Gerir Serviços	
Author	ase	
Date	24/Nov/2009 14:38:03	
Brief Description	O gerente pretende remover uma categoria de serviços da sua empresa	
Preconditions	Existem categorias no sistema O utilizador está autenticado com privilégios de Gerente	
Post-conditions	Num categorias.servico.sistema = Num categorias.servico.sistema - 1	
Flow of Events		Actor Input
	1	Remover categoria de serviço
	2	
	3	Escolhe categoria
	4	
	5	
	6	Confirma remoção
	7	
	8	
	9	
	10	
Alternative 1 5 - Remoção cancelada		Actor Input
	1	
	2	
Exception 4 - Contratos activos da categoria a remover		Actor Input
	1	
		System Response
		Mostra lista de categorias existentes no sistema
		Verifica se há contratos activos cujo serviço pertence à categoria a remover
		Pede confirmação da remoção
		Remove serviços cujo tipo de serviço pertence à categoria a remover
		Remove tipos de serviço que pertençam à categoria a remover
		Remove categoria do sistema
		Informa sucesso da remoção
		Informa cancelamento da remoção
		Volta ao passo 2
		Informa utilizador que ha contratos activos cujo serviço pertence à categoria a remover

Figura 6.18: Descrição Use Case: Remover Categoria de Serviço

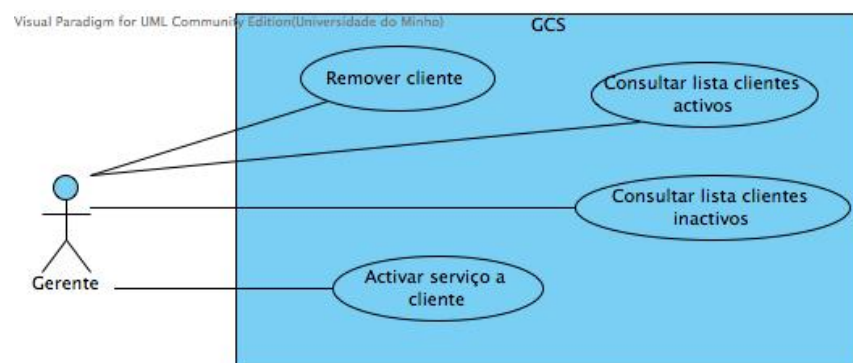


Figura 6.19: Diagramas Gerir Clientes

Brief Description	O gerente pretende activar um serviço solicitado por um cliente após confirmação de pagamento.	
Preconditions	O utilizador está autenticado com privilégios de gerente. Existem clientes no sistema. Existem serviços no sistema. Existem fornecedores no sistema.	
Post-conditions	Sucesso: Lista de serviços pendentes do cliente actualizada. Lista de serviços activos do cliente actualizada. Num serviços activos cliente = Num serviços activos cliente + 1 Ficha de cliente actualizada. Insucesso: O sistema não se altera.	
Flow of Events		Actor Input
	1	Activar serviço a cliente.
	2	
	3	Inserir código de cliente.
	4	
	5	
	6	
	7	
	8	Selecciona serviço.
	9	Activar serviço seleccionado.
	10	
	11	
	12	
	13	
	14	
Alternative Flow of Events 1 4 - Código inserido não existe.		Actor Input
	1	
	2	
Exception 1 6- Cliente não tem serviços pendentes.		Actor Input
	1	

System Response	
Pede código de cliente.	
Procura cliente por código no sistema.	
Mostra ficha de cliente.	
Verifica se cliente tem serviços pendentes.	
Mostra lista de serviços pendentes.	
Marca serviço como "Activo".	
Remove serviço da lista de serviços pendentes.	
Inserir serviço na lista de serviços activos.	
Actualiza registo de montantes pagos pelo cliente.	
Informa que activação de serviço foi bem sucedida.	
System Response	
Informa que código inserido não existe.	
Voltar ao passo 2.	
System Response	
Informa que cliente não possui serviços à espera de activação.	

Figura 6.20: Descrição Use Case: Activar Serviço a Cliente

Super Use Case	Gerir Clientes	
Author	chinalo	
Date	21/Nov/2009 18:15:52	
Brief Description	O gerente pretende consultar a lista de clientes activos no sistema	
Preconditions	Existem clientes activos no sistema O utilizador está autenticado com privilégios de gerente	
Post-conditions		
Flow of Events		Actor Input
	1	Consultar lista de clientes activos
	2	
	3	Escolhe cliente
	4	
	5	
		System Response
		Mostra lista de clientes activos
		Vai buscar ficha do cliente ao sistema
		Mostra ficha do cliente

Figura 6.21: Descrição Use Case: Consultar Lista de Clientes Activos

Super Use Case	Gerir Clientes	
Author	ase	
Date	24/Nov/2009 10:39:10	
Brief Description	O gerente pretende consultar a lista de clientes inactivos no sistema	
Preconditions	Existem clientes inactivos no sistema O utilizador está autenticado com privilégios de gerente	
Post-conditions		
Flow of Events		Actor Input
	1	Consultar lista de clientes inactivos
	2	
	3	Escolhe cliente
	4	
	5	
		System Response
		Mostra lista de clientes inactivos
		Vai buscar ficha do cliente ao sistema
		Mostra ficha do cliente

Figura 6.22: Descrição Use Case: Consultar Lista de Clientes Inactivos

Super Use Case	Gerir Clientes	
Author	chinelo	
Date	21/Nov/2009 18:03:17	
Brief Description	O gerente pretende marcar um cliente como "Nao Cliente"	
Preconditions	Existem clientes no sistema O utilizador está autenticado com privilégios de gerente	
Post-conditions	Sucesso: Registo do cliente actualizado Num clientes activos sistema = Num clientes activos sistema - 1 Num clientes inactivos sistema = Num clientes inactivos sistema + 1	
Flow of Events		Actor Input
	1	Remover cliente
	2	
	3	Inserir código do cliente
	4	
	5	
	6	
	7	Confirma
	8	
	9	
		System Response
		Pede código do cliente
		Procura cliente no sistema pelo código introduzido
		Verifica se cliente já está marcado como "Nao Cliente"
		Pede confirmação de remoção
		Marca cliente como "Nao Cliente"
		Informa que cliente foi "removido" com sucesso
Alternative Flow of Events 1 4 - Cliente não encontrado		Actor Input
	1	
	2	
		System Response
		Informa o gerente que não existe nenhum cliente com o código introduzido
		Volta ao passo 2
Exception 1 5 - Cliente já é "Nao Cliente"		Actor Input
	1	
		System Response
		Informa que o utilizador relativo ao código digitado já se encontra em estado "Nao Cliente".
Alternative Flow of Events 2 7 - Não confirma		Actor Input
	1	
		System Response
		Volta ao passo 2

Figura 6.23: Descrição Use Case: Remover Cliente

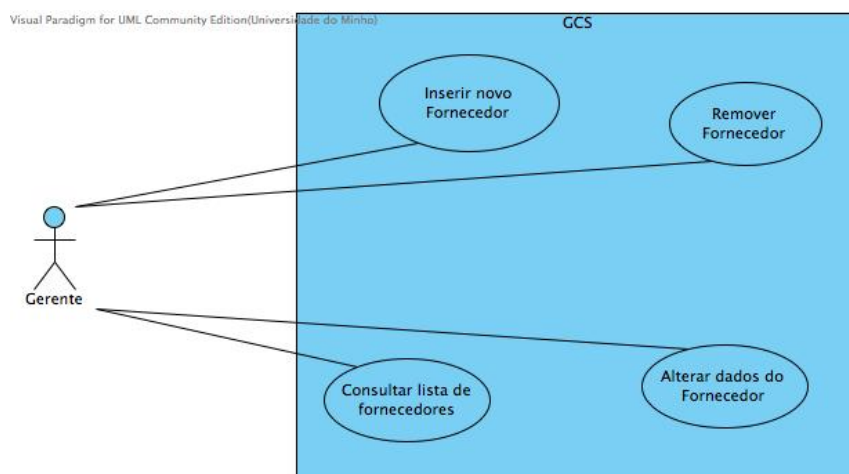


Figura 6.24: Diagrama Gerir Carteira de Fornecedores

Super Use Case	Gerir Carteira de Fornecedores	
Author	João	
Date	20/Nov/2009 16:05:58	
Brief Description	O Gerente pretende inserir um novo fornecedor na lista de fornecedores.	
Preconditions	O utilizador deve estar autenticado com privilégios de gerente.	
Post-conditions	Número de Fornecedores = Número de Fornecedores + 1.	
Flow of Events		Actor Input
	1	Inserir novo fornecedor.
	2	
	3	Inserir o nome da empresa.
	4	
	5	
	6	Preenche o formulário.
	7	Confirma o preenchimento do formulário.
	8	
	9	
	10	
Alternative Flow of Events 1 4- Empresa já existe na lista de fornecedores.		Actor Input
	1	
	2	
Alternative Flow of Events 2 8 - Os campos não estão todos preenchidos ou os dados não fazem sentido.		Actor Input
	1	
	2	

	System Response	
	Pede o nome da empresa.	
	Verifica se o nome da empresa já existe na lista de fornecedores.	
	Mostra formulário de registo de novo fornecedor.	
	Verifica se todos os campos estão preenchidos e se o input é consistente com o esperado.	
	Regista o novo fornecedor na lista de fornecedores.	
	Informa a inserção do fornecedor com sucesso.	
	Informa que a empresa já existe na lista de fornecedores.	
	Volta ao passo 2.	
	Informa que os campos não estão todos preenchidos ou os dados não fazem sentido.	
	Volta ao passo 5.	

Figura 6.25: Descrição Use Case: Inserir Novo Fornecedor

Super Use Case		
Author	asa	
Date	24/Nov/2009 9:48:25	
Brief Description	O utilizador pretende alterar os dados informativos de um fornecedor.	
Preconditions	O utilizador está autenticado com privilégios de gerente. Existem fornecedores no sistema.	
Post-conditions	O registo do fornecedor escolhido é actualizado.	
Flow of Events		Actor Input
		System Response
	1	Alterar dados do Fornecedor.
	2	Percorre a lista de todos os fornecedores.
	3	Apresenta uma lista com esses fornecedores.
	4	Selecciona fornecedor.
	5	Vai buscar dados relativos a esse fornecedor.
	6	Apresenta um formulário com os dados recolhidos.
	7	Altera dados
	8	Verifica se todos os campos obrigatórios estão preenchidos.
	9	Verifica se os campos fazem sentido com o input necessário.
	10	Actualiza o registo do fornecedor.
Alternative Flow of Events 1 8a - Campos obrigatórios em falta.		Actor Input
		System Response
	1	Informa os campos obrigatórios que estão vazios.
Alternative Flow of Events 2 9a - Dados inconsistentes.		Actor Input
		System Response
	1	Informa que campos estão inconsistentes com os dados introduzidos.
	2	Voltar ao passo 6

Figura 6.26: Descrição Use Case: Alterar Dados do Fornecedor

Super Use Case	<u>Gerir Carteira de Fornecedores</u>	
Author	<u>João</u>	
Date	<u>20/Nov/2009 16:43:07</u>	
Brief Description	<u>O Gerente pretende remover um fornecedor</u>	
Preconditions	<u>Existem fornecedores</u> <u>O utilizador está autenticado com privilégios de gerente</u>	
Post-conditions	<u>Número de fornecedores = Número de fornecedores - 1</u> <u>Estado do fornecedor = "Não contratado"</u>	
Flow of Events		Actor Input
	1	<u>Remover fornecedor</u>
	2	
	3	<u>Escolhe nome de fornecedor a remover</u>
	4	
	5	<u>Confirma</u>
	6	
System Response		<u>Mostra lista de nomes de fornecedores</u>
		<u>Pede confirmação</u>
		<u>Marca fornecedor como "Não contratado"</u>
		<u>Informa que o fornecedor foi removido com sucesso</u>
Alternative Flow of Events 1		Actor Input
<u>5 - Não confirma</u>	1	
		<u>Volta ao passo 2</u>

Figura 6.27: Descrição Use Case: Remover Fornecedor

Super Use Case	<u>Gerir Carteira Fornecedores</u>	
Author	<u>asa</u>	
Date	<u>24/Nov/2009 9:44:23</u>	
Brief Description	<u>O utilizador pretende obter uma lista com todos os fornecedores sub-contratados</u>	
Preconditions	<u>O utilizador está autenticado com privilégios de gerente</u> <u>Existem fornecedores no sistema</u>	
Post-conditions		
Flow of Events		Actor Input
	1	<u>Consultar lista fornecedores</u>
	2	
	3	<u>Seleciona fornecedor</u>
	4	
System Response		<u>Mostra uma lista contendo todos os fornecedores sub-contratados pela empresa</u>
		<u>Procura os dados relativos a esse fornecedor</u> <u>Mostra uma ficha informativa sobre o fornecedor</u>

Figura 6.28: Descrição Use Case: Consultar Lista de Fornecedores

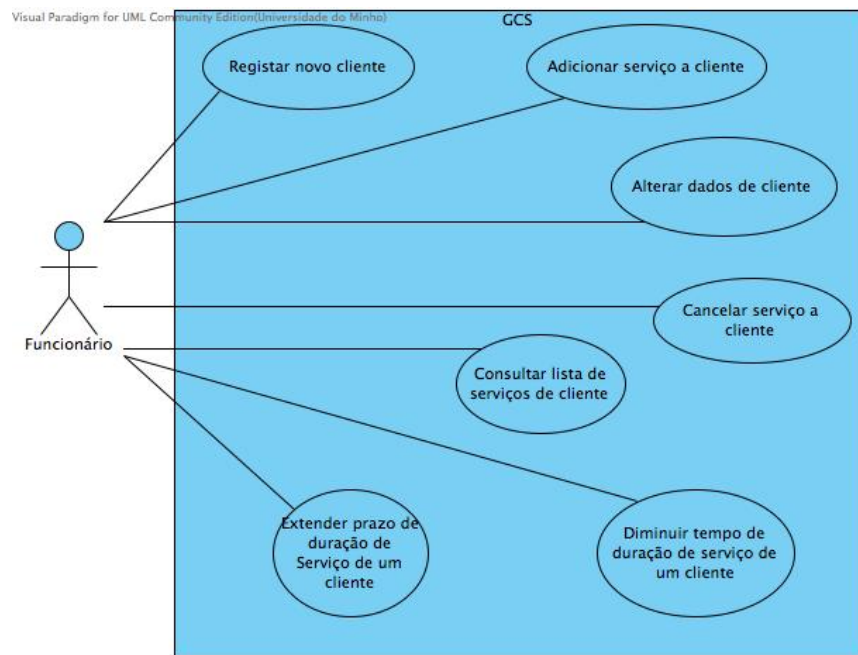


Figura 6.29: Diagrama Menu Funcionario

Super Use Case	Menu Funcionário	
Author	João Pedro	
Date	22/Nov/2009 17:13:56	
Brief Description	O utilizador pretende registar um novo cliente no sistema	
Preconditions	O utilizador está autenticado com privilégios iguais ou superiores a Funcionário	
Post-conditions	Sucesso: Num clientes activos = Num clientes activos + 1 Insucesso: Sistema não se altera	
Flow of Events		Actor Input
	1	Inserir novo cliente
	2	
	3	Preenche formulário
	4	
	5	
	6	
	7	
	8	
	9	
	10	
	11	Confirma
	12	
	13	
	14	Inserir username e password
	15	
	16	
Exception 1 6 - Já existe cliente com a mesma morada		Actor Input
	1	
Alternative Flow of Events 1 4 - Campos obrigatórios, em falta		System Response
	1	Informa que já existe um registo de cliente com a mesma morada e que não é possível adicionar o cliente pretendido.
Alternative Flow of Events 2 5 - Dados inconsistentes com o formulário		System Response
	1	Informa campos obrigatórios que estão em falta
		System Response
	2	Volta ao passo 2
		System Response
	1	Informa campos que estão inconsistentes com o tipo de dados requerido
		System Response
	2	Voltar ao passo 2

Figura 6.30: Descrição Use Case: Registar Novo Cliente

Super Use Case	Menu Funcionário	
Author	João Pedro	
Date	22/Nov/2009 17:54:11	
Brief Description	O utilizador pretende alterar dados pessoais da ficha de cliente	
Preconditions	Existem clientes no sistema O utilizador está autenticado com privilégios iguais ou superiores a Funcionário	
Post-conditions	Sucesso; Ficha de cliente actualizada. Insucesso; O sistema permanece inalterado.	
Flow of Events	Actor Input	System Response
	1 Alterar dados de cliente	
	2	Pede código do cliente
	3 Inserir código de cliente	
	4	Procura cliente no sistema pelo código introduzido
	5	Recolhe informações sobre cliente
	6	Mostra ficha do cliente com os dados recolhidos
	7 Altera ficha de cliente	
	8	Actualiza registo do cliente no sistema
Alternative Flow of Events 1	Actor Input	System Response
4 - Código de cliente não existente	1	Informa que código introduzido não existe no sistema
	2	Voltar ao passo 2

Figura 6.31: Descrição Use Case: Alterar Dados de Cliente

Super Use Case	Menu Funcionário	
Author	João Pedro	
Date	22/Nov/2009 17:40:48	
Brief Description	O funcionário pretende adicionar um serviço a um determinado cliente. O utilizador está autenticado com privilégios de cliente.	
Preconditions	Existem clientes no sistema. Existem fornecedores no sistema. Existem serviços no sistema.	
Post-conditions	Sucesso: Num serviços pendentes activação = Num serviços pendentes activação + 1 Registo de cliente actualizado. Insucesso: Sistema mantém-se inalterado.	
Flow of Events		Actor Input
	1	Adicionar serviço a cliente
	2	
	3	Inserir Código do cliente
	4	
	5	
	6	Escolhe categoria
	7	
	8	Escolhe serviço
	9	
	10	
	11	Escolhe fornecedor
	12	
	13	Inserir tempo de duração serviço
	14	
	15	Confirma adição de serviço
	16	
	17	
	18	
	19	
Alternative Flow of Events 1 4 - Cliente já existe		Actor Input
	1	
	2	
Exception 1 16 - Cliente já tem serviço		Actor Input
	1	
Alternative Flow of Events 2 15 - Não confirma adição de serviço		Actor Input
	1	

System Response	
Pede código do cliente	
Procura cliente no sistema pelo código introduzido	
Mostra lista de categorias de serviço	
Mostra lista de todos os serviços correspondentes a essa categoria	
Procura fornecedores que prestam esse serviço	
Mostra lista de fornecedores que prestam esse serviço	
Mostra detalhes sobre o serviço	
Calcula preço do serviço para a duração escolhida	
Verifica se cliente já tem serviço	
Insere novo serviço na lista de serviços pendentes do cliente	
Incrementa número de serviços no cliente em 1 unidade	
Informa que após confirmação de pagamento, o serviço passará a estar activo	
Informa que código de cliente não existe	
Volta para 2	
Informa que o cliente já tem o serviço	
Voltar ao passo 10	

Figura 6.32: Descrição Use Case: Adicionar Serviço a Cliente

Super Use Case	Menu Funcionário	
Author	João Pedro	
Date	22/Nov/2009 18:03:29	
Brief Description	O utilizador pretende cancelar um serviço adquirido por um cliente	
Preconditions	Existem clientes no sistema O utilizador está autenticado com privilégios iguais ou superiores a Funcionário O cliente seleccionado tem serviços associados	
Post-conditions	Sucesso: O serviço é cancelado a um cliente O estorno de um cliente é actualizado $\text{Num serviços activos cliente} = \text{Num serviços activos} - 1$ $\text{Num serviços inactivos cliente} = \text{Num serviços inactivos} + 1$ Insucesso: O sistema permanece inalterado	
Flow of Events	Actor Input	System Response
	1 Remover serviço a cliente	
	2	Pede código do cliente
	3 Insere código do cliente	
	4	Procura o cliente no sistema pelo código introduzido
	5	Mostra lista de serviços do cliente
	6 Escolhe serviço a cancelar	
	7	Calcula tempo que cliente usufruiu do serviço
	8	Regista na data de cancelamento a data actual
	9	Calcula o tempo que faltava para acabar o serviço
	10	Calcula estorno
	11	Incrementa estorno a devolver na ficha do cliente
	12	Marca serviço como "Inactivo" e remove-o da lista de serviços activos
	13	Insere o serviço na lista de serviços inactivos
	14	Informa que o serviço foi removido com sucesso e o estorno calculado
Alternative Flow of Events 1 4 - Cliente inexistente	Actor Input	System Response
	1	Informa que o cliente não existe
	2	Volta para 2

Figura 6.33: Descrição Use Case: Remover Serviço a um Cliente

Super Use Case	Menu Funcionário	
Author	ase	
Date	24/Nov/2009 12:25:08	
Brief Description	O cliente pretende estender o prazo de duração de um serviço.	
Preconditions	Existem clientes no sistema Existem serviços no sistema	
Post-conditions	Sucesso: Registo de serviço no cliente actualizado. Ficha do cliente actualizada. Insucesso: O sistema não se altera.	
Flow of Events		Actor Input
	1	Extender prazo de serviço.
	2	
	3	Insere código de cliente
	4	
	5	
	6	Escolhe o serviço
	7	
	8	Insere data
	9	
	10	
	11	
	12	
	13	
		System Response
		Pede código de cliente
		Procura cliente no sistema pelo código introduzido
		Mostra ficha do cliente e lista dos serviços activos
		Pede nova data de fim de serviço
		Verifica se data excede data de fim de serviço previamente registada
		Calcula diferença de tempo entre a data de fim original e a nova data de fim
		Calcula pagamento extra do cliente
		Regista nova data de fim de serviço
		Informa que extensão de prazo ocorreu com sucesso
Alternative Flow of Events 2 8 - Nova data de fim de serviço não excede data original de fim		Actor Input
	1	
	2	
		System Response
		Informa que nova data deve exceder data original de fim de serviço
		Voltar ao passo 6
Alternative Flow of Events 1 4 - Código de cliente não existe		Actor Input
	1	
	2	
		System Response
		Informa que código introduzido não existe no sistema
		Voltar ao passo 2

Figura 6.34: Descrição Use Case: Estender Prazo de Duração de Serviço de um Cliente

Super Use Case	Menu Funcionário	
Author	ase	
Date	24/Nov/2009 12:31:04	
Brief Description	O cliente pretende reduzir o prazo de fim de um serviço adquirido	
Preconditions	Existem clientes no sistema Existem serviços no sistema	
Post-conditions	Sucesso: Registo de cliente actualizado Data de fim de serviço do serviço seleccionado alterada Estorno actualizado Insucesso: Sistema não se altera	
Flow of Events		Actor Input
	1	Diminuir tempo duração serviço
	2	
	3	Inserir código cliente
	4	
	5	
	6	Selecciona serviço
	7	
	8	
	9	Inserir data
	10	
	11	
	12	
	13	
	14	
	15	
Alternative Flow of Events 1 4 - Código de cliente não existe		Actor Input
	1	
Alternative Flow of Events 2 10 - Nova data de fim de serviço excede data prévia de fim de serviço		Actor Input
	1	
		System Response
	1	Informa que código introduzido não existe no sistema
		System Response
	2	Informa que nova data inserida para fim de serviço é posterior à data previamente definida
		System Response
	2	Informa que data de fim de serviço foi alterada com sucesso
		System Response
	2	Informa que código introduzido não existe no sistema
		System Response
	2	Informa que nova data inserida para fim de serviço é posterior à data previamente definida
		System Response
	2	Informa que data de fim de serviço foi alterada com sucesso
		System Response
	2	Informa que nova data inserida para fim de serviço é posterior à data previamente definida
		System Response
	2	Informa que data de fim de serviço foi alterada com sucesso

Figura 6.35: Descrição Use Case: Diminuir Tempo de Duração de Serviço de um Cliente

Super Use Case	Menu Funcionário	
Author	João Pedro	
Date	22/Nov/2009 18:08:44	
Brief Description	O utilizador pretende consultar a lista de serviços adquiridos por um cliente	
Preconditions	Existem clientes no sistema Existem serviços no sistema O utilizador está autenticado com privilégios iguais ou superiores a Funcionário	
Post-conditions		
Flow of Events		Actor Input
	1	Consultar lista de Serviços de cliente
	2	
	3	Insere código do cliente
	4	
	5	
	6	Selecciona serviço
Alternative Flow of Events 1 4 - Cliente inexistente		System Response
	1	
	2	
		System Response
	1	Pede Código do Cliente
	2	Procura cliente no sistema pelo código introduzido
		System Response
	1	Mostra lista dos serviços adquiridos pelo cliente
	2	Mostra detalhes sobre o serviço seleccionado
		System Response
	1	Informa que o cliente não existe
	2	Volta para o passo 2

Figura 6.36: Descrição Use Case: Consultar Lista de Serviços de um Cliente

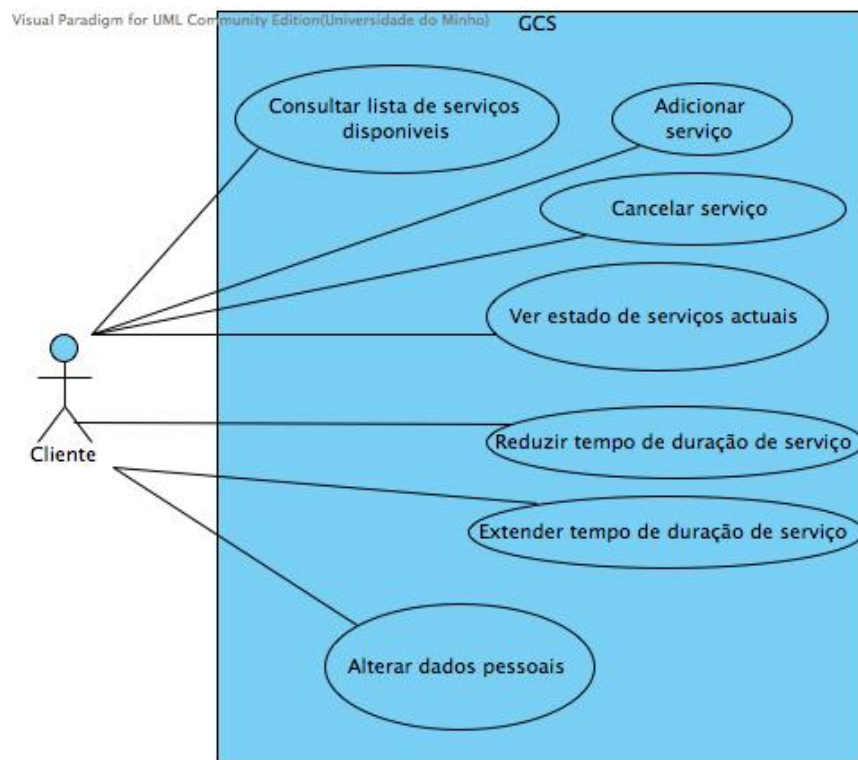


Figura 6.37: Diagrama Menu Cliente

Super Use Case	Menu <u>Cliente</u>	
Author	<u>asa</u>	
Date	24/Nov/2009 16:20:37	
Brief Description	O <u>cliente</u> pretende adicionar um novo <u>serviço</u> ao seu contrato	
Preconditions	O <u>utilizador</u> está autenticado com <u>privilegios</u> de <u>cliente</u> Existem <u>categorias</u> de <u>serviço</u> no sistema Existem <u>fornecedores</u> <u>ativos</u> no sistema Existem <u>serviços</u> no sistema	
Post-conditions	<u>Sucesso</u> : Num <u>serviços</u> <u>cliente</u> = Num <u>serviços</u> <u>cliente</u> + 1 <u>Registo</u> de <u>cliente</u> <u>actualizado</u> Num <u>serviços</u> <u>pendentes</u> <u>activação</u> = Num <u>serviços</u> <u>pendentes</u> <u>activação</u> + 1 <u>Insucesso</u> : Sistema mantém-se inalterado	
Flow of Events		Actor Input
	1	<u>Adicionar serviço</u>
	2	
	3	<u>Escolhe categoria</u>
	4	
	5	<u>Escolhe serviço</u>
	6	
	7	<u>Escolhe fornecedor</u>
	8	
	9	
	10	<u>Confirma</u>
	11	
	12	<u>Escolhe tempo de duração do serviço</u>
	13	
	14	<u>Insere morada</u>
	15	
	16	
	17	
	18	<u>Confirma serviço</u>
	19	
	20	
		System Response
		Mostra lista de <u>categorias</u> de <u>serviço</u>
		Mostra lista de <u>serviços</u> dessa categoria
		Mostra lista de <u>fornecedores</u> que prestam esse <u>serviço</u>
		Mostra detalhes sobre <u>serviço</u>
		Pede <u>confirmação</u>
		Pede tempo de <u>duração</u>
		Pede <u>morada</u> onde irá ser prestado o <u>serviço</u>
		Verifica se o <u>fornecedor</u> escolhido já está a prestar o <u>serviço</u> nessa <u>morada</u>
		Calcula <u>montante</u> a pagar
		Mostra <u>montante</u> a pagar
		Adiciona <u>serviço</u> a lista de <u>serviços</u> <u>pendentes</u> do <u>cliente</u>
		Informa que após <u>confirmação</u> de pagamento o <u>serviço</u> passará a estar <u>activo</u>
Alternative Flow of Events 1 12 - <u>Não confirma compra</u>		Actor Input
	1	
		System Response
		<u>Voltar ao passo 8</u>
Alternative Flow of Events 2 15 - <u>Fornecedor já presta serviço</u>		Actor Input
	1	
	2	
		System Response
		Informa que <u>fornecedor</u> já lhe está a prestar esse <u>serviço</u>
		<u>Volta ao passo 6</u>

Figura 6.38: Descrição Use Case: Adicionar Serviço

Super Use Case	Menu <u>Cliente</u>		
Author	<u>ase</u>		
Date	24/Nov/2009 16:28:54		
Brief Description	O cliente pretende cancelar um serviço adquirido		
Preconditions	Existem clientes no sistema O utilizador está autenticado com privilégios de cliente O cliente seleccionado tem serviços associados		
Post-conditions	Sucesso: O serviço é cancelado ao cliente. O estorno do cliente é actualizado. $\text{Num. serviços activos cliente} = \text{Num. serviços activos} - 1$ $\text{Num. serviços inactivos cliente} = \text{Num. serviços inactivos} + 1$ Insucesso: O sistema permanece inalterado		
Flow of Events		Actor Input	System Response
	1	<u>Cancelar serviço</u>	
	2		<u>Mostra lista de serviços associados ao utilizador</u>
	3	<u>Escolhe serviço</u>	
	4		<u>Mostra detalhes sobre serviço</u>
	5	<u>Cancela serviço</u>	
	6		<u>Regista na data de cancelamento a data actual</u>
	7		<u>Calcula quanto tempo faltava para data original de fim de serviço</u>
	8		<u>Calcula montante a devolver (estorno)</u>
	9		<u>Actualiza na ficha do cliente o campo Estorno</u>
	10		<u>Marca serviço como "Inactivo"</u>
	11		<u>Remove serviço da lista de serviços activos</u>
	12		<u>Insere serviço na lista de serviços inactivos</u>
	13		<u>Informa que serviço foi cancelado com sucesso</u>

Figura 6.39: Descrição Use Case: Cancelar Serviço

Super Use Case	Menu <u>Cliente</u>	
Author	<u>ase</u>	
Date	24/Nov/2009 16:35:38	
Brief Description	<u>O cliente pretende alterar dados pessoais do seu registo</u>	
Preconditions	<u>O utilizador está autenticado com privilégios de cliente</u>	
Post-conditions	<u>Sucesso: Ficha do cliente actualizada. Insucesso: O sistema não se altera.</u>	
Flow of Events		Actor Input
	1	<u>Alterar dados pessoais</u>
	2	
	3	
	4	<u>Altera formulário</u>
	5	
	6	
	7	
	8	
		System Response
		<u>Procura no sistema a ficha do cliente</u>
		<u>Mostra formulário com dados alteráveis sobre o cliente</u>
		<u>Verifica se campos obrigatórios estão preenchidos</u>
		<u>Verifica se dados são consistentes com o formulário</u>
		<u>Actualiza ficha de cliente</u>
		<u>Informa que ficha do cliente foi actualizada com sucesso</u>
Alternative Flow of Events 1 <u>5 - Campos obrigatórios vazios</u>		Actor Input
	1	
	2	
		System Response
		<u>Informa quais os campos obrigatórios que estão em falta</u>
		<u>Voltar ao passo 3</u>
Alternative Flow of Events 2 <u>6 - Dados inconsistentes com o formulário</u>		Actor Input
	1	
	2	
		System Response
		<u>Informa campos que não são consistentes com o tipo de dados</u>
		<u>Voltar ao passo 3</u>

Figura 6.40: Descrição Use Case: Alterar Dados Pessoais

Super Use Case		
Author	Noor	
Date	Nov 30, 2009 11:21:20 AM	
Brief Description	O cliente <u>extende o tempo de duração de um serviço</u> .	
Preconditions	O cliente <u>tem que ter serviços activos</u> . O utilizador <u>tem que estar autenticado com privilégios de cliente</u> . O sistema <u>tem serviços</u> . O sistema <u>tem fornecedores</u> .	
Post-conditions	O tempo de <u>duração do serviço</u> = tempo de <u>duração do serviço</u> + tempo de <u>extensão</u> .	
Flow of Events		Actor Input
	1	O cliente <u>extende o tempo de duração de um serviço</u> .
	2	
	3	<u>Escolhe serviço</u> .
	4	
	5	
	6	<u>Insere data</u> .
	7	
	8	
	9	
	10	
	11	
Alternative Flow of Events		Actor Input
	1	
	2	

		System Response
		<u>Mostra lista de serviços activos</u> .
		<u>Mostra informação de serviço</u> .
		<u>Pede nova data de fim de serviço</u> .
		<u>Verifica se data excede data de fim de serviço registada</u> .
		<u>Calcula diferença de tempo entre a data de fim original e a nova data de fim</u> .
		<u>Calcula pagamento extra do cliente</u> .
		<u>Regista nova data de fim de serviço</u> .
		<u>Informa que extensão de prazo ocorreu com sucesso</u> .
		System Response
		<u>Informa que nova data deve exceder data original de fim de serviço</u> .
		<u>Volta ao passo 6</u> .

Figura 6.41: Descrição Use Case: Extender Tempo de Duração de Serviço

Super Use Case		
Author Noor		
Date Nov 30, 2009 11:33:05 AM		
Brief Description O cliente pretende reduzir o tempo de duração de um serviço activo		
Preconditions O cliente tem serviços activos O sistema tem serviços O sistema tem fornecedores O utilizador tem que estar autenticado com privilégios de cliente		
Post-conditions O tempo de duração de um serviço = tempo de duração de serviço - tempo de redução		
Flow of Events		Actor Input
	1	O cliente pretende reduzir o tempo de duração de um serviço activo
	2	
	3	Escolhe serviço
	4	
	5	
	6	Insere data
	7	
	8	
	9	
	10	
	11	
	12	
	13	
Alternative Flow of Events 8-Verifica se data é menor que a data de fim de serviço registada		Actor Input
	1	
	2	

		System Response
		Mostra lista de serviços activos
		Mostra informação do serviço
		Pede nova data de fim de serviço
		Verifica se data é menor que a data de fim de serviço registada
		Calcula diferença de tempo entre data original e a nova data de fim
		Calcula diferença de pagamento do cliente
		Actualiza ficha do cliente
		Regista nova data de fim de serviço
		Informa que redução do prazo ocorreu com sucesso

		System Response
		Informa que nova data tem que ser menor do que a data original de fim de serviço
		Volta ao passo 6

Figura 6.42: Descrição Use Case: Reduzir Tempo de Duração de Serviço

Super Use Case	Menu <u>Cli</u> ente	
Author	ase	
Date	24/Nov/2009 16:33:43	
Brief Description	O cliente pretende ver o estado do seu contrato	
Preconditions	O utilizador está autenticado com privilégios de cliente	
Post-conditions		
Flow of Events		Actor Input
	1	Ver estado dos serviços actuais
	2	
	3	
	4	
	5	
	6	
	7	Escolhe um dos serviços
	8	
		System Response
		Procura ficha de cliente no sistema
		Apresenta ficha de cliente
		Mostra a lista de serviços activos
		Mostra a lista de serviços inactivos
		Mostra a lista de serviços pendentes
		Mostra detalhes sobre o serviço

Figura 6.43: Descrição Use Case: Ver Estado Actual dos Serviços

Super Use Case	Menu <u>Cli</u> ente	
Author	ase	
Date	24/Nov/2009 16:10:08	
Brief Description	O cliente pretenda consultar a lista de serviços disponíveis	
Preconditions	O utilizador está autenticado com privilégios de cliente Existem categorias de serviço no sistema Existem fornecedores no sistema Existem serviços no sistema	
Post-conditions		
Flow of Events		Actor Input
	1	Consultar lista de serviços
	2	
	3	Escolhe categoria
	4	
	5	Escolhe serviço
	6	
	7	Escolhe fornecedor
	8	
		System Response
		Mostra lista com todas as categorias de serviço no sistema
		Mostra lista de serviços dessa categoria
		Mostra lista de fornecedores que prestam esse serviço
		Mostra detalhes sobre serviço prestado por esse fornecedor

Figura 6.44: Descrição Use Case: Consultar Lista de Serviços Disponíveis

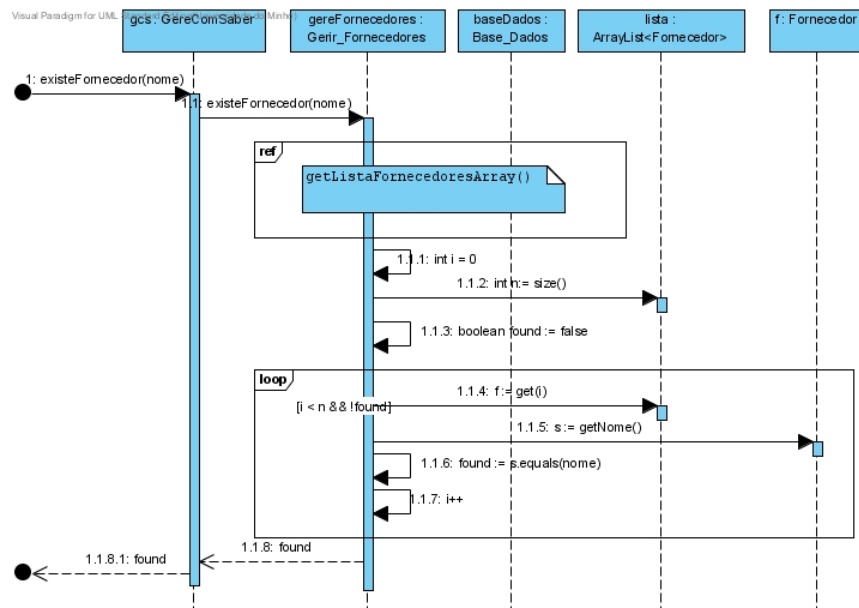


Figura 6.45: Diagrama de sequência : existeFornecedor(nome)

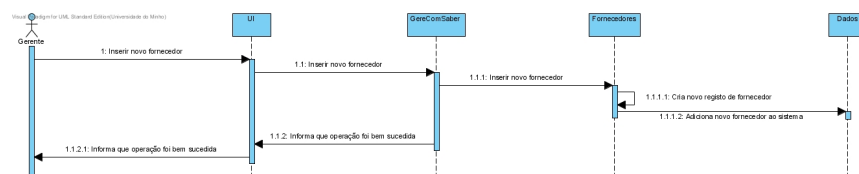


Figura 6.46: Diagrama de sequência de Subsistema : Inserir Novo Fornecedor

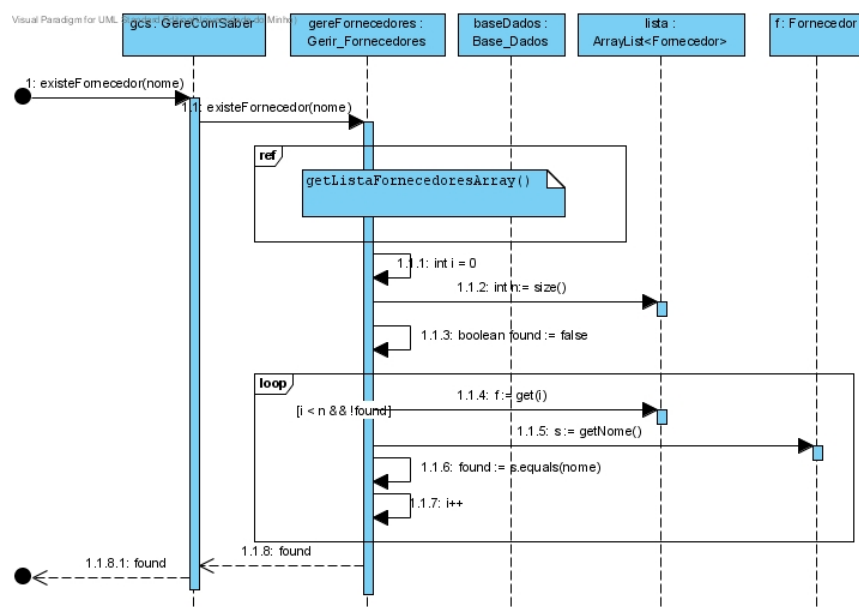


Figura 6.47: Diagrama de sequência : inserirFornecedor(nome,contacto,NIB)

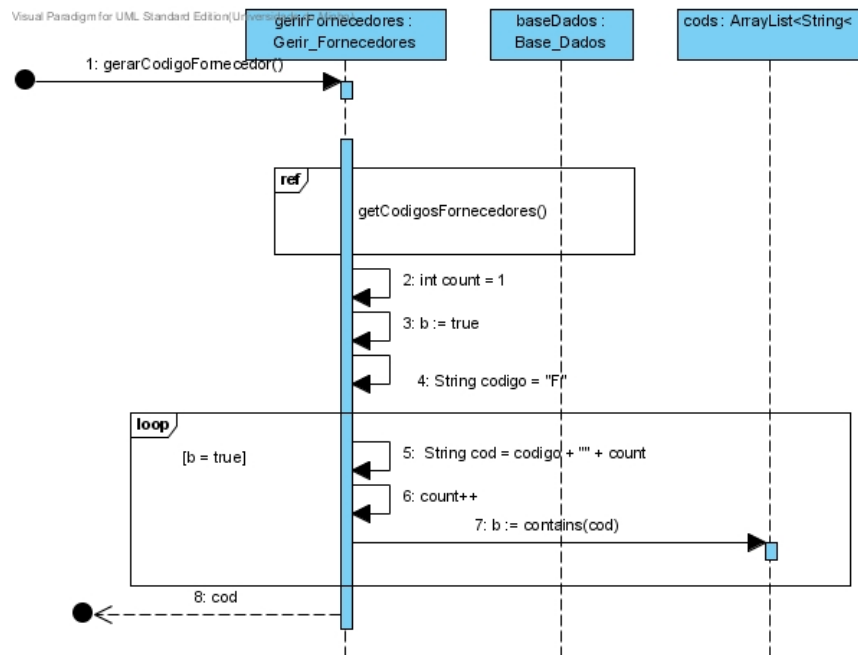


Figura 6.48: Diagrama de sequência : gerarCodigoFornecedor()

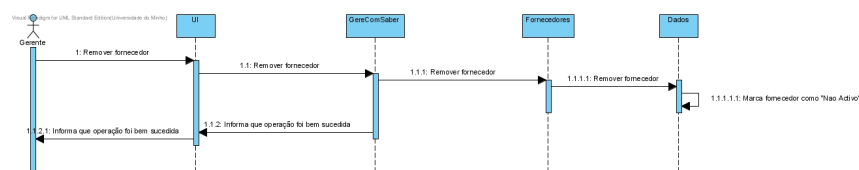


Figura 6.49: Diagrama de sequência Subsistema : Remover Fornecedor

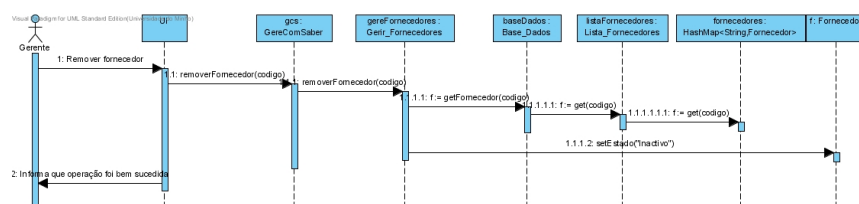


Figura 6.50: Diagrama de sequência : removerFornecedor(codigo)

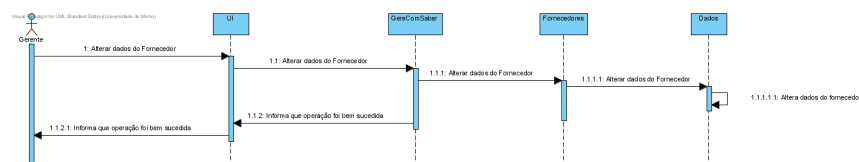


Figura 6.51: Diagrama de sequência Subsistema : Alterar Dados Fornecedor

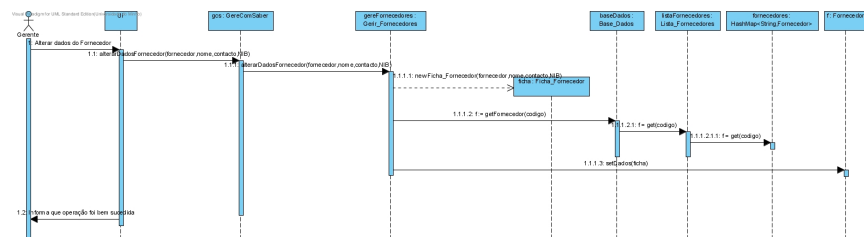


Figura 6.52: Diagrama de sequência : Alterar Dados Fornecedor

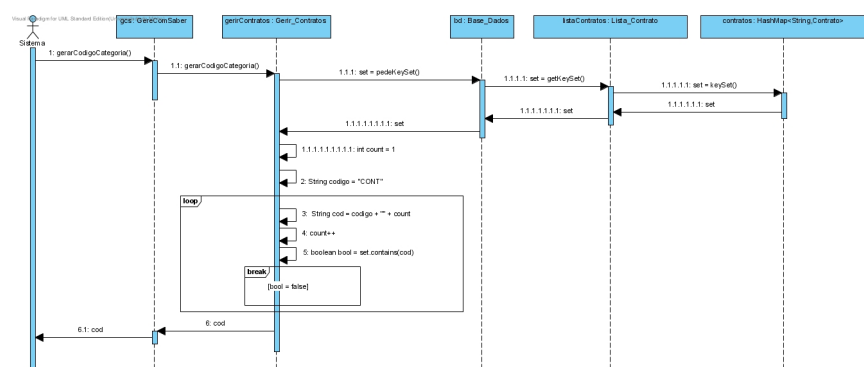


Figura 6.53: Diagrama de sequência : Gerar Codigo Contrato

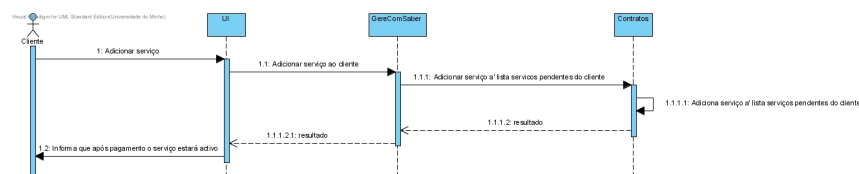


Figura 6.54: Diagrama de sequência Subsistema : Adicionar Serviço

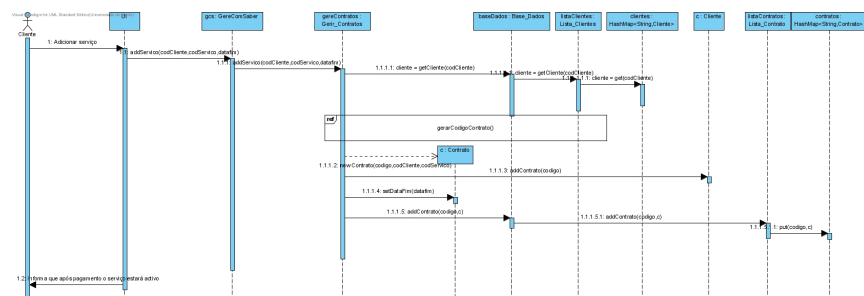


Figura 6.55: Diagrama de sequência : Adicionar Serviço

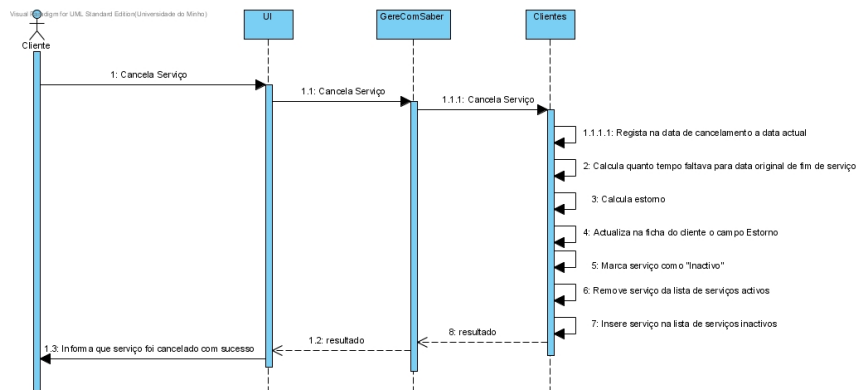


Figura 6.56: Diagrama de sequência Subsistema : Cancelar Serviço

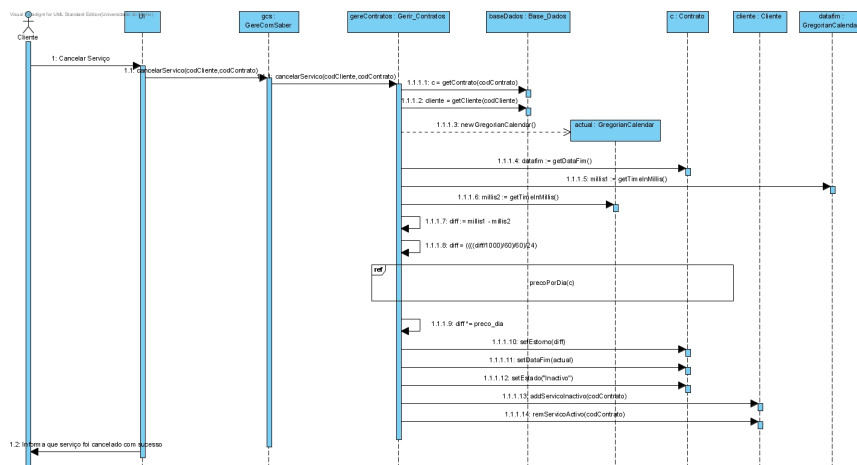


Figura 6.57: Diagrama de sequência : cancelarServico(codCliente,codContrato)

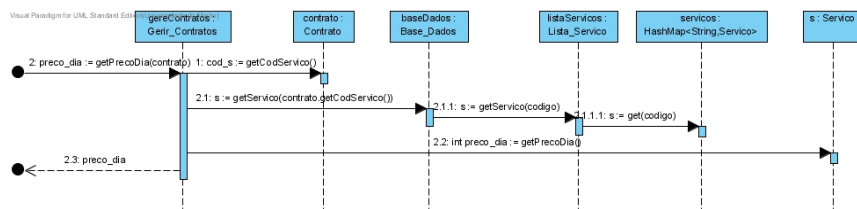
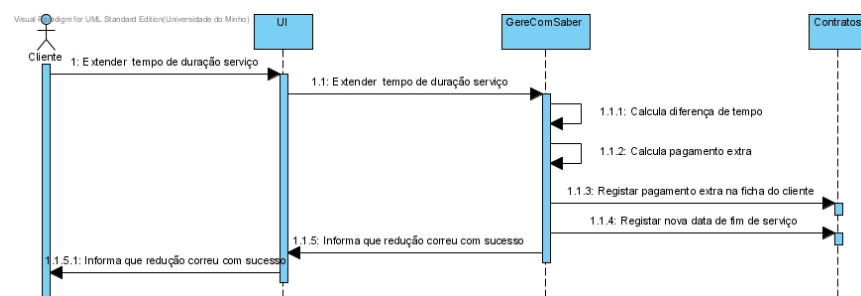
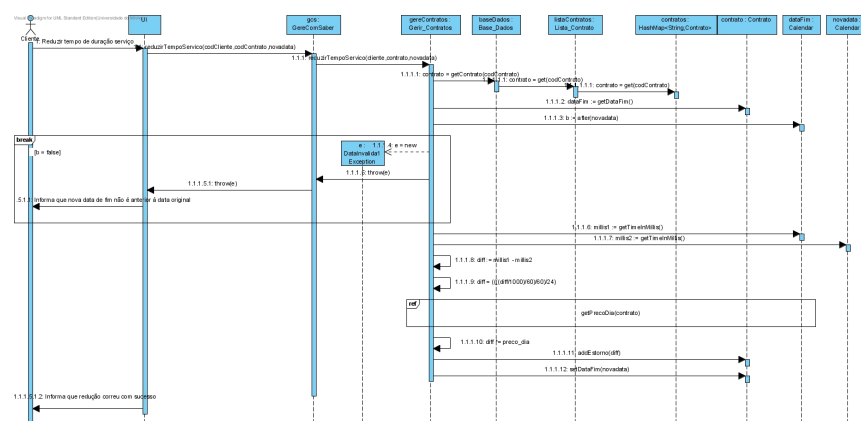
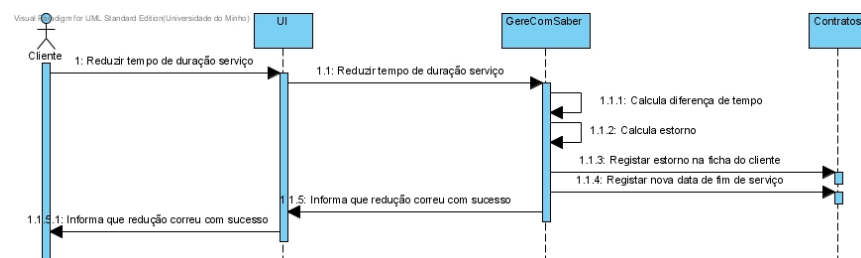


Figura 6.58: Diagrama de sequência : getPrecoDia(contrato)



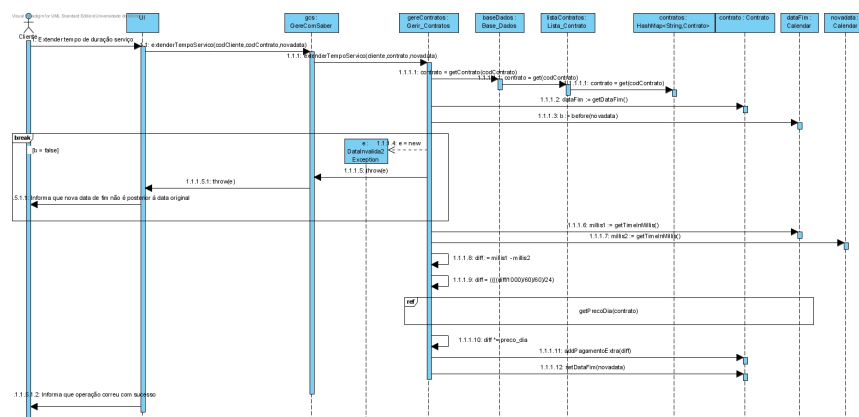


Figura 6.62: Diagrama de sequência : extenderTempoServico(codcliente,codcontrato,novadata)

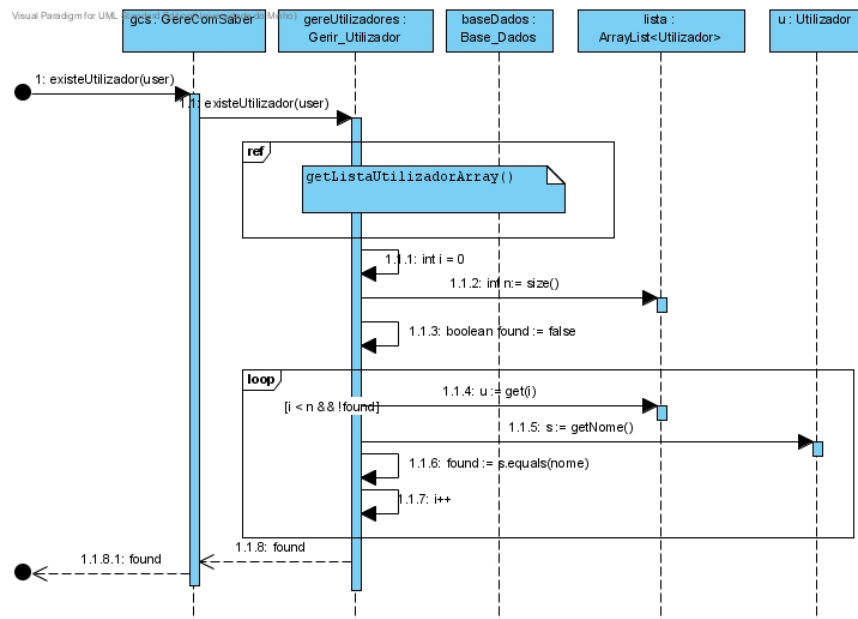


Figura 6.63: Diagrama de sequência : existeUtilizador(user)

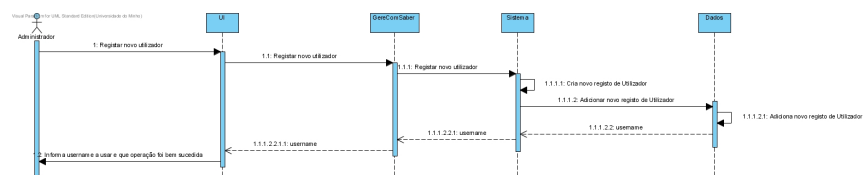


Figura 6.64: Diagrama de sequência Subsistema : Registrar Novo Utilizador

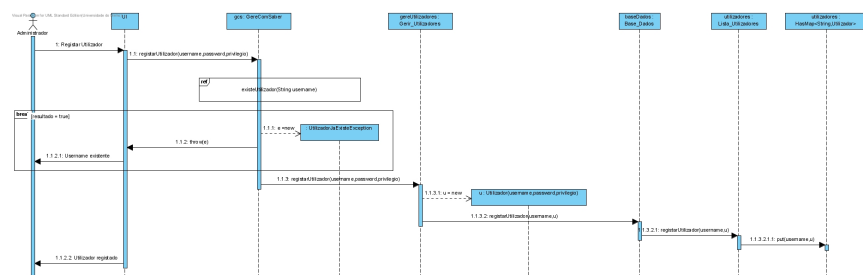


Figura 6.65: Diagrama de sequência : registrarUtilizador(username,password)

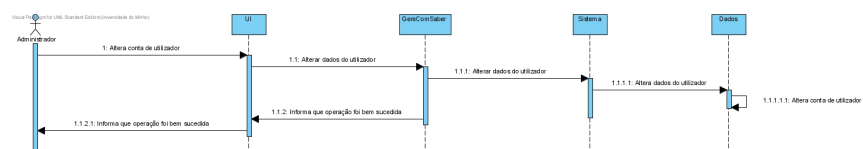


Figura 6.66: Diagrama de sequência Subsistema : Alterar Conta de Utilizador

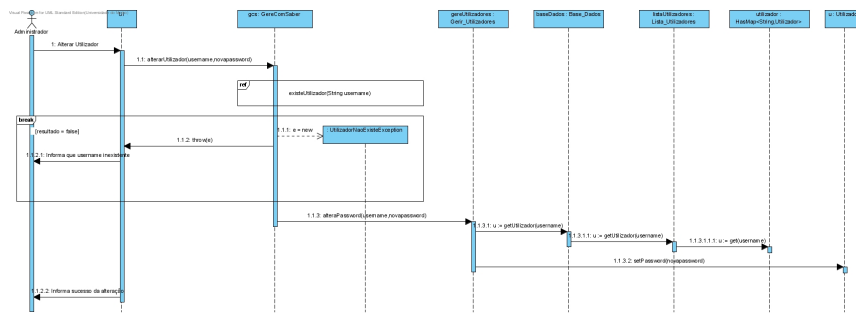


Figura 6.67: Diagrama de sequência : alterarUtilizador(username,novapassword)

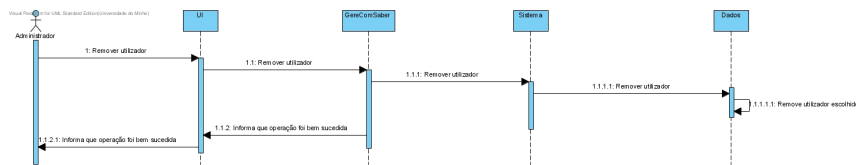


Figura 6.68: Diagrama de sequência Subistema : Remover Utilizador

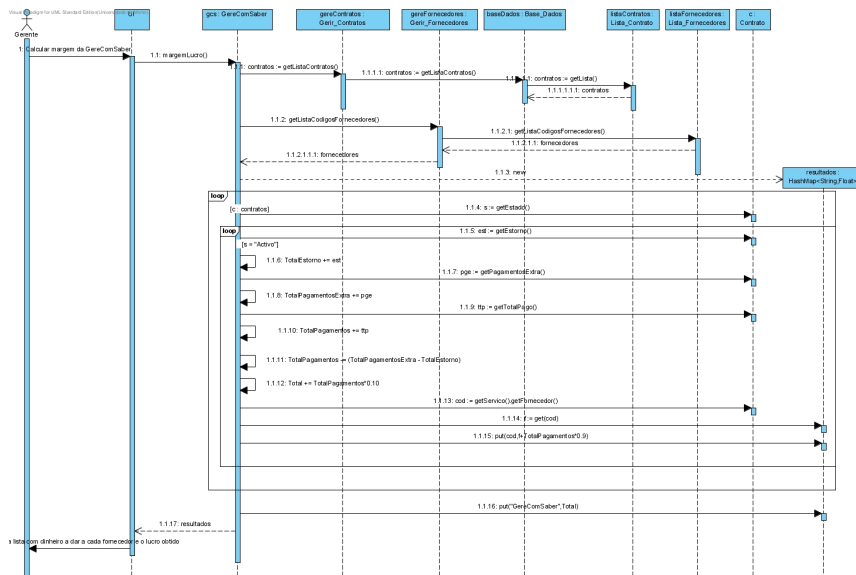


Figura 6.69: Diagrama de sequência : removerUtilizador(username)

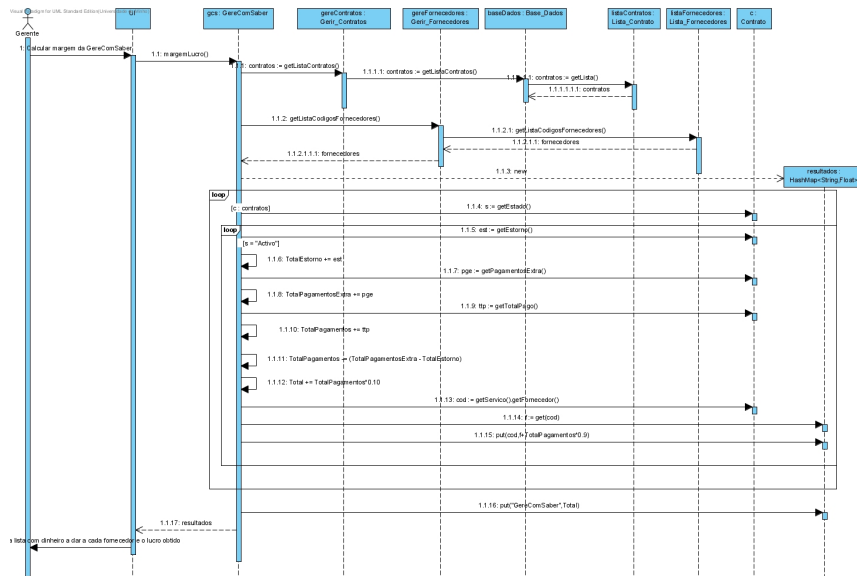


Figura 6.70: Diagrama de sequência : margemLucro()

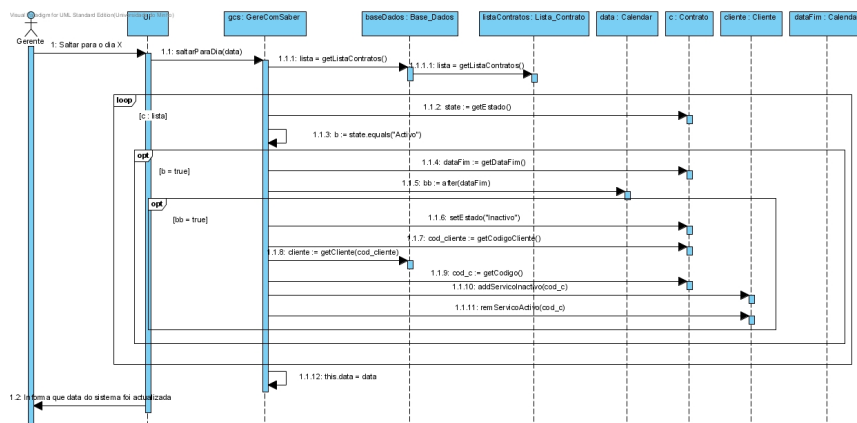


Figura 6.71: Diagrama de sequência : saltarParaDia(data)

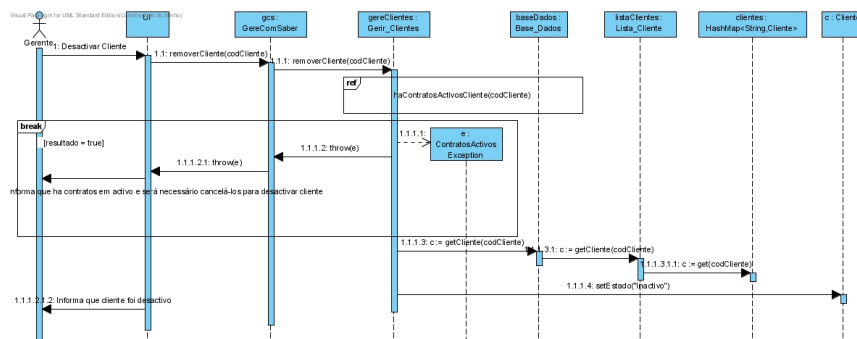


Figura 6.72: Diagrama de sequência : removerCliente(codCliente)

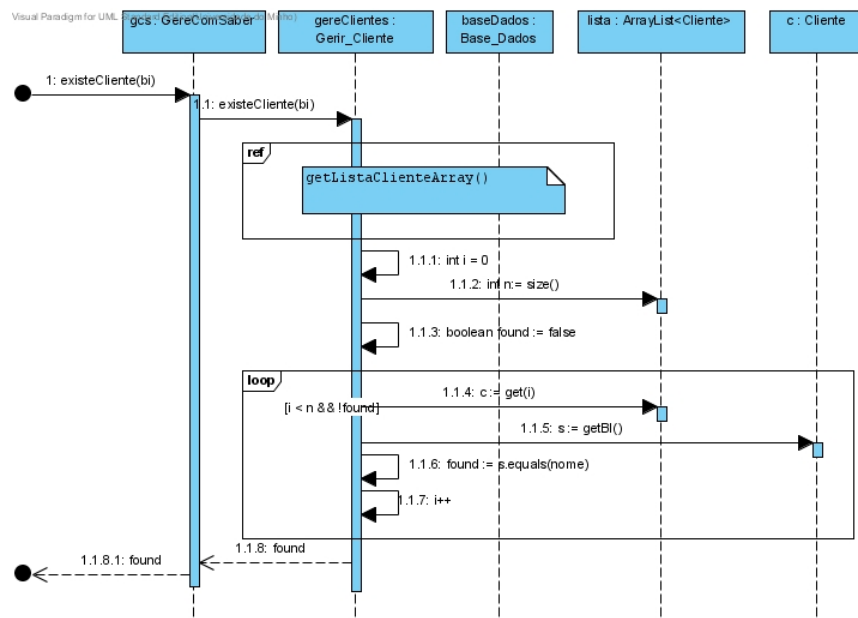


Figura 6.73: Diagrama de sequência : existeCliente(bi)

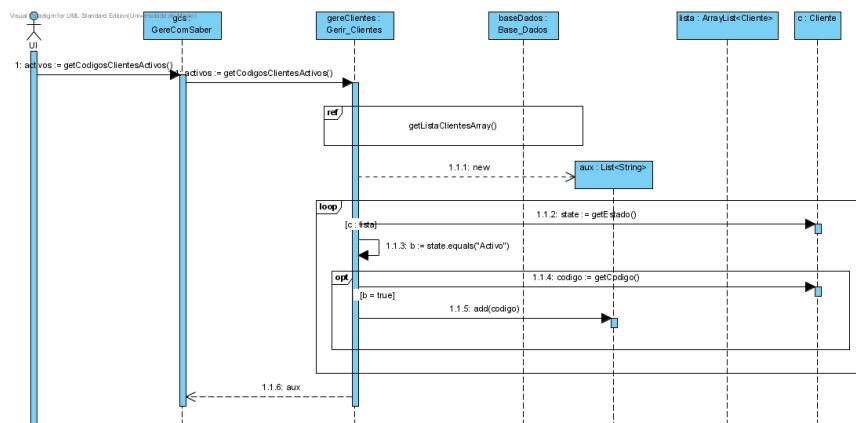


Figura 6.74: Diagrama de sequência : getCodigosClientesActivos()

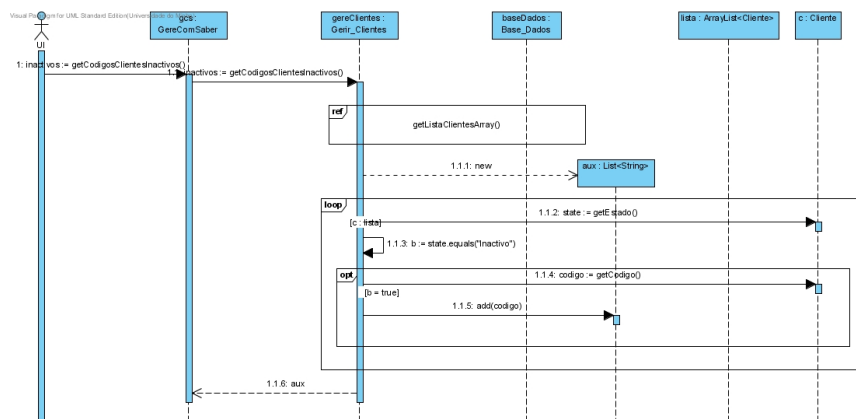


Figura 6.75: Diagrama de sequência : getCodigosClientesInativos()

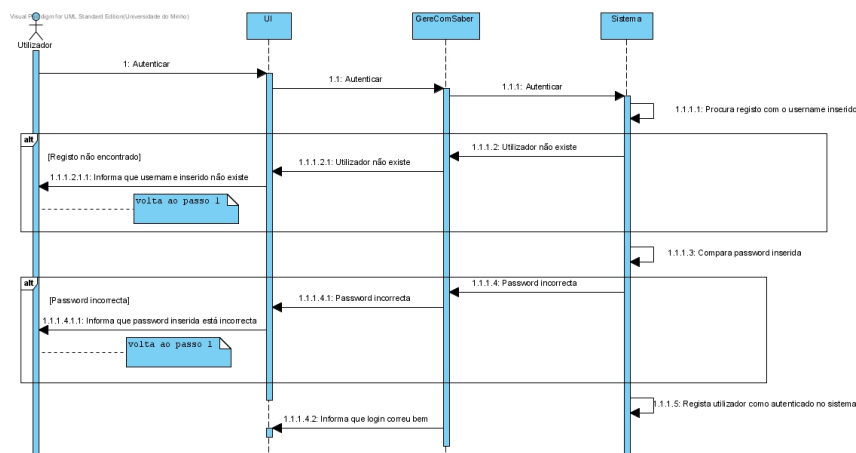


Figura 6.76: Diagrama de sequência Subsistema : Autenticação



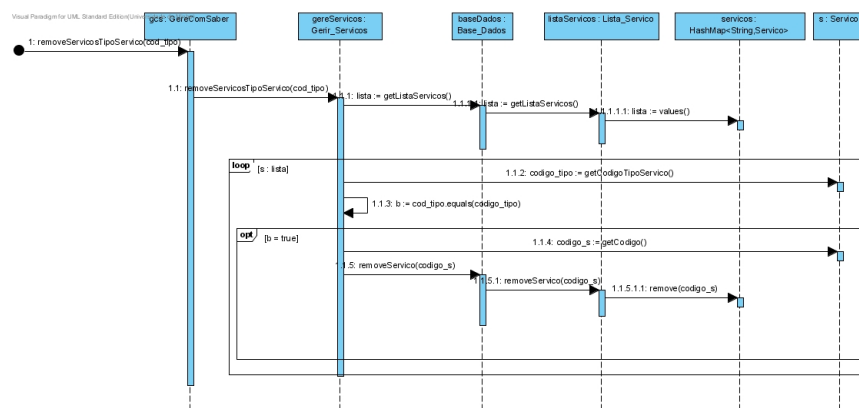


Figura 6.80: Diagrama de sequência : removeServicosTipoServico(codtipo)

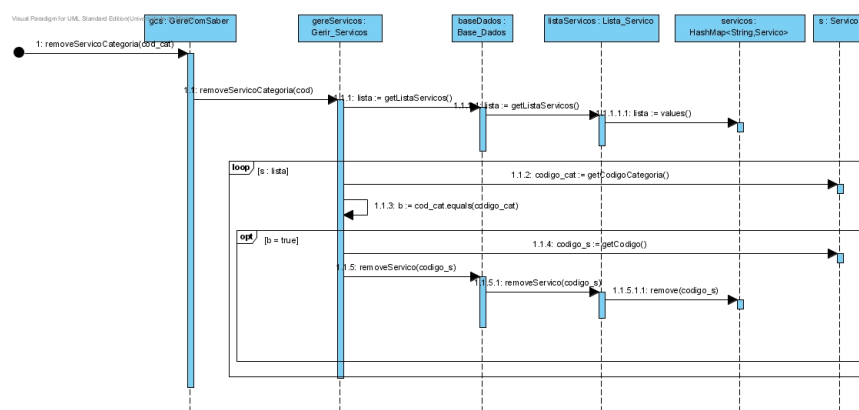


Figura 6.81: Diagrama de sequência : removeServicoCategoria(codcat)

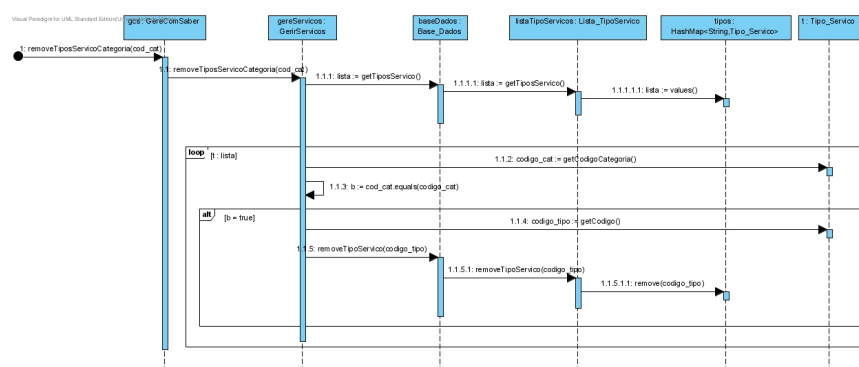


Figura 6.82: Diagrama de sequência : removeTiposServicoCategoria(codcat)

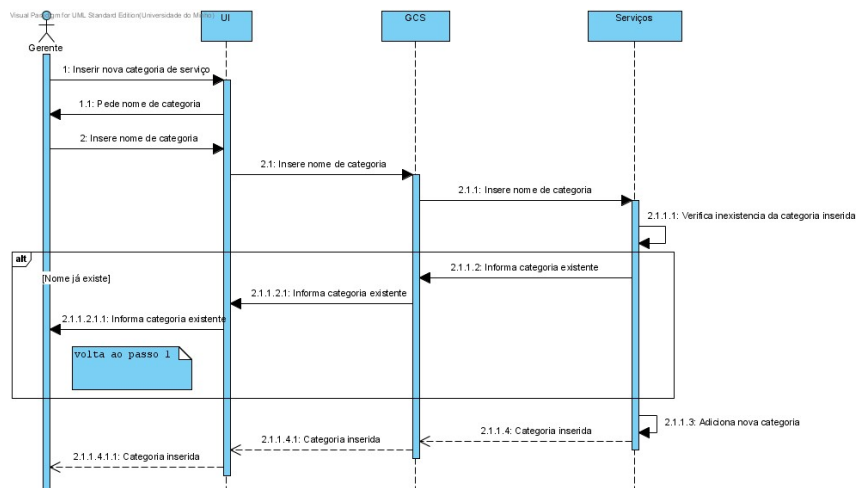


Figura 6.83: Diagrama de sequência Subsistema : Inserir nova categoria de serviço

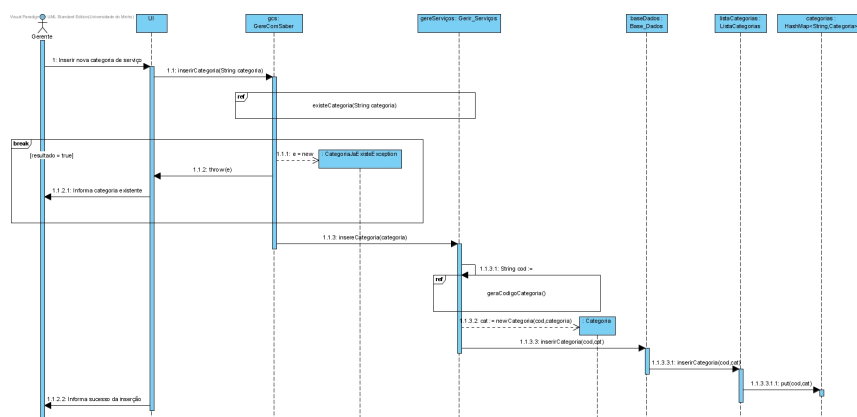


Figura 6.84: Diagrama de sequência : inserirCategoria(nome)

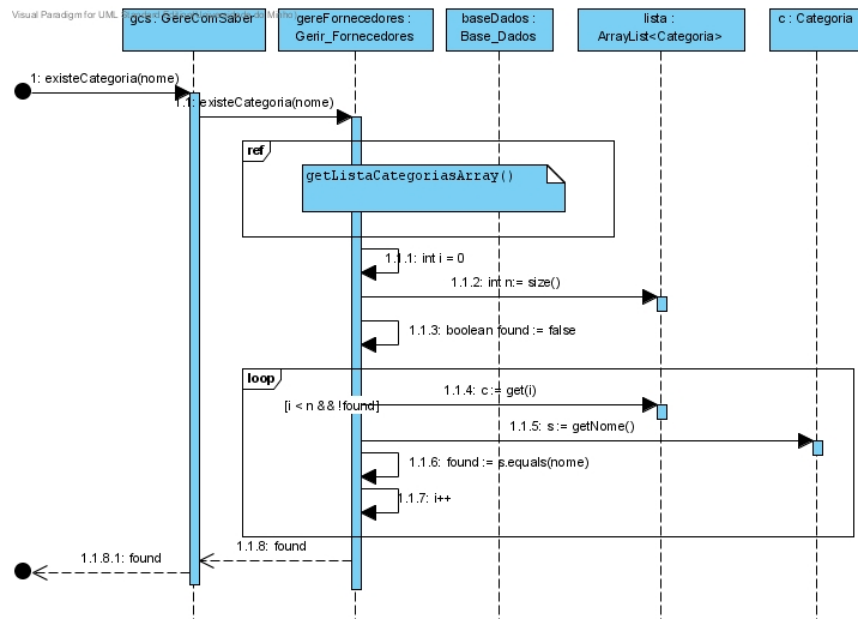


Figura 6.85: Diagrama de sequência : existeCategoria(nome)

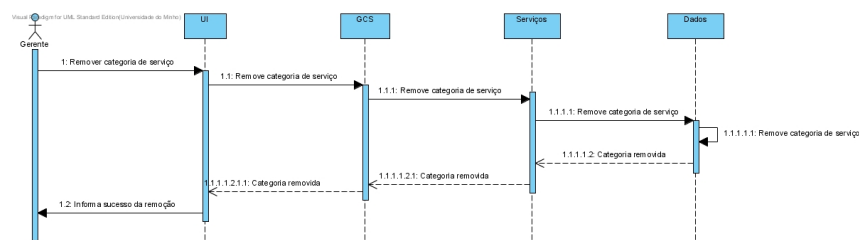


Figura 6.86: Diagrama de sequência Subsistema : Remove Categoria

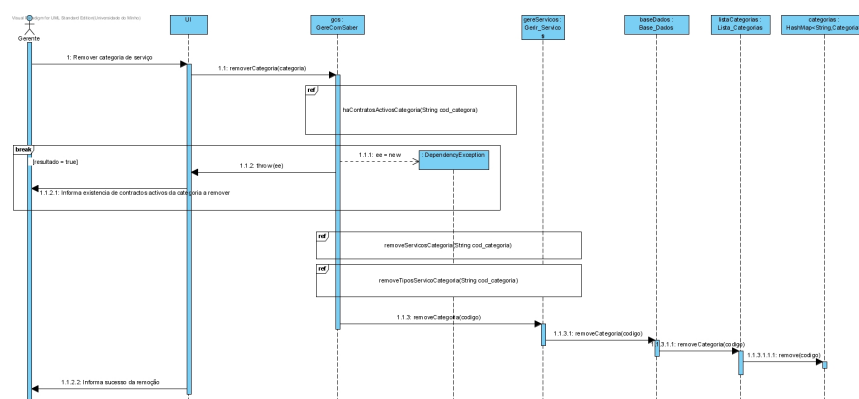


Figura 6.87: Diagrama de sequência : removeCategoria(codcategoria)

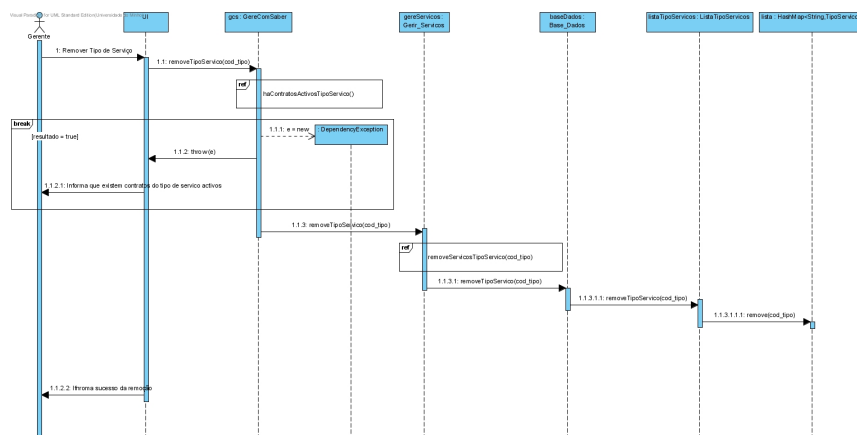


Figura 6.90: Diagrama de sequência : removeTipoServico(codtipo)

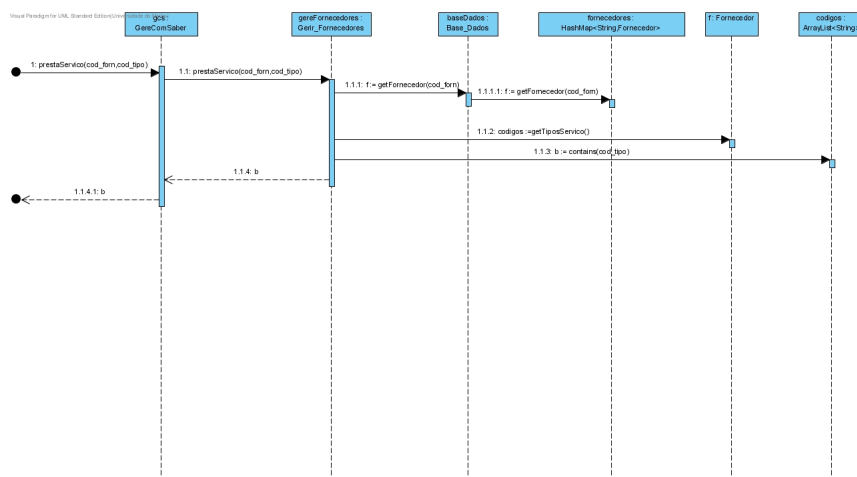


Figura 6.91: Diagrama de sequência : prestaSe rvico(codforn,codtipo)

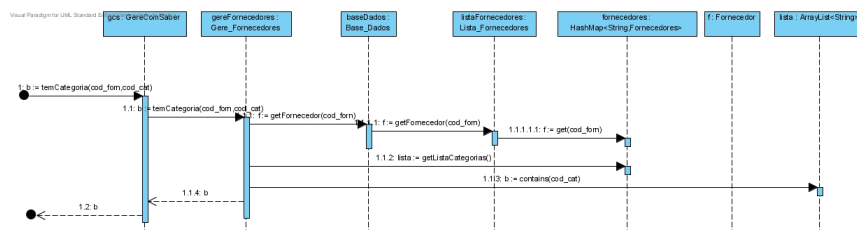


Figura 6.92: Diagrama de sequência : temCategoria(codforn,codcat)

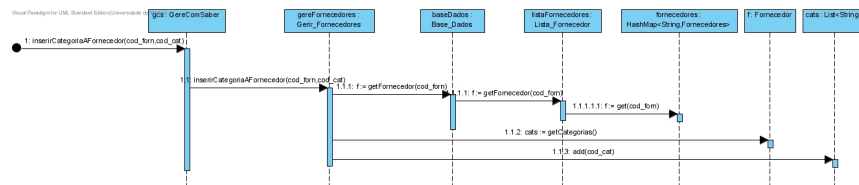


Figura 6.93: Diagrama de sequência : inserirCategoriaAFornecedor(codforn,codcat)

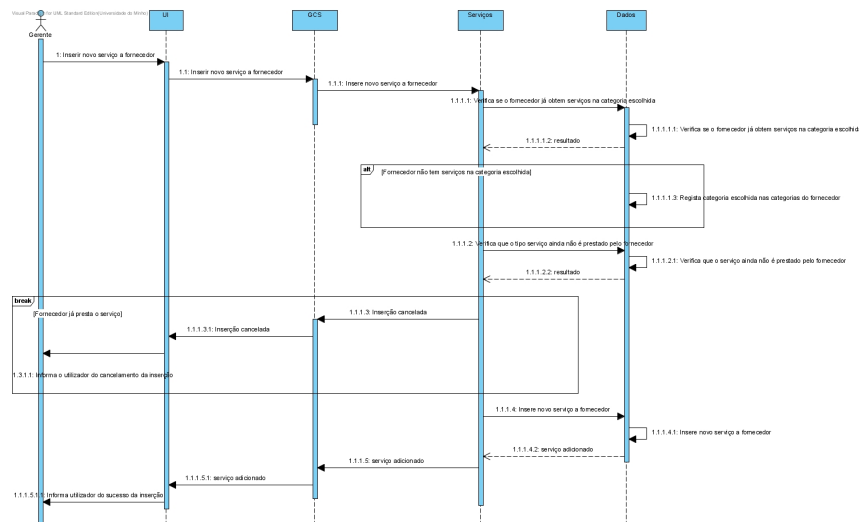


Figura 6.94: Diagrama de sequência Subsistema : Inserir Novo Serviço A Fornecedor

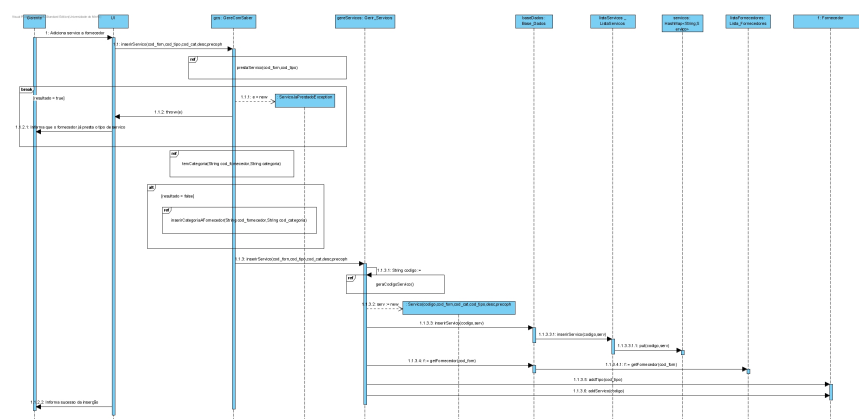


Figura 6.95: Diagrama de sequência : inserirServicoFornecedor(codforn,codtipo,codcat,desc,precoph)

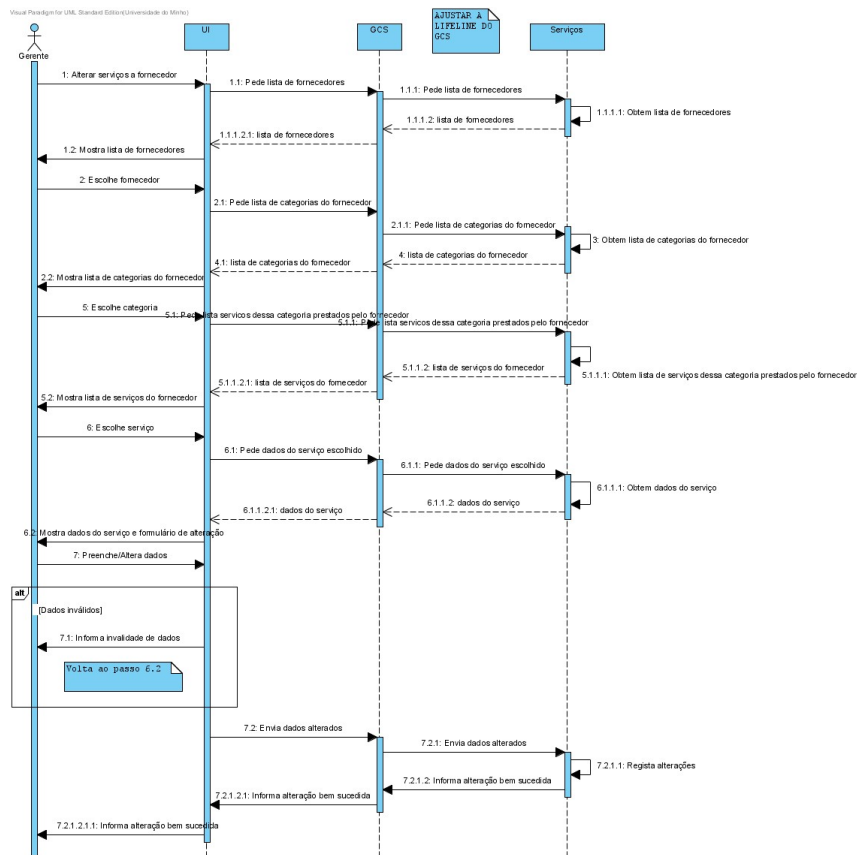


Figura 6.96: Diagrama de sequência Subistema : Alterar Serviços

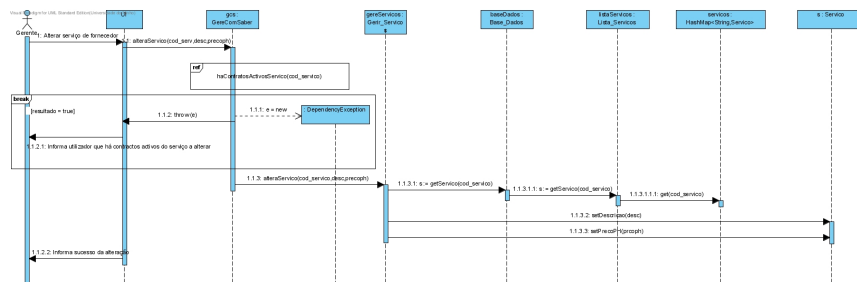


Figura 6.97: Diagrama de sequência : alteraServico(String codservico,String desc,float precoph)

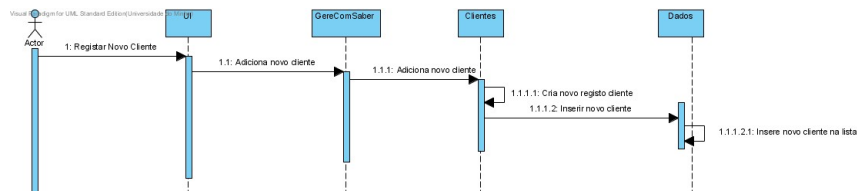
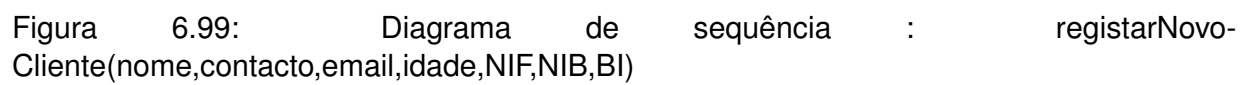


Figura 6.98: Diagrama de sequência Subistema : Registrar Novo Cliente



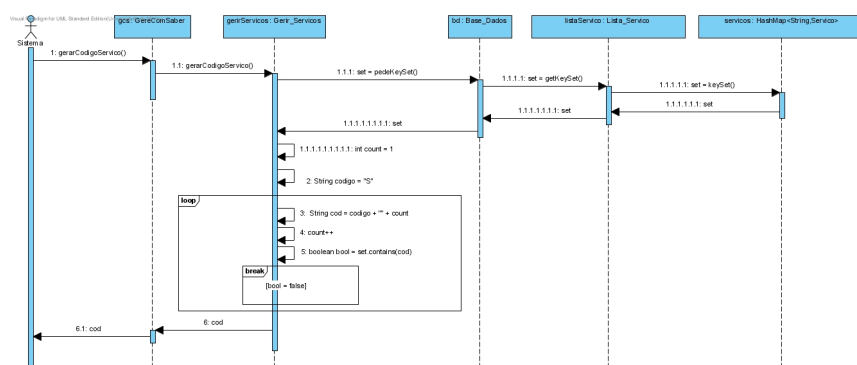


Figura 6.102: Diagrama de sequência : gerarCodigoServico()

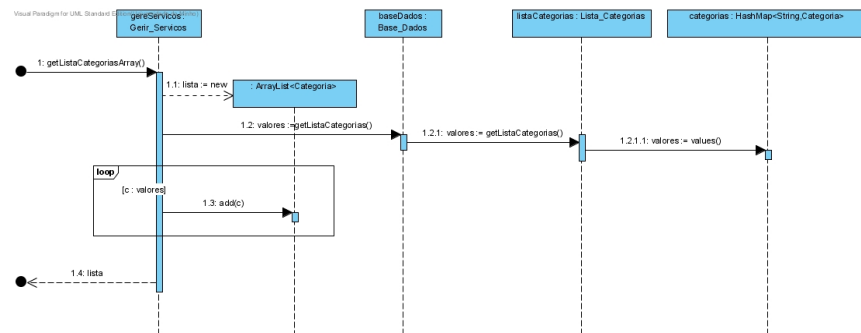


Figura 6.103: Diagrama de sequência : getListaCategoriasArray()

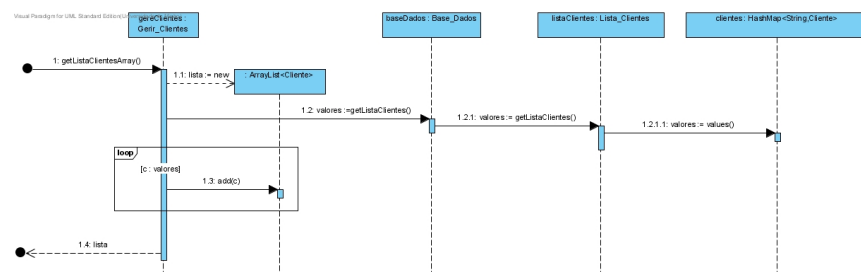


Figura 6.104: Diagrama de sequência : getListaClientesArray()

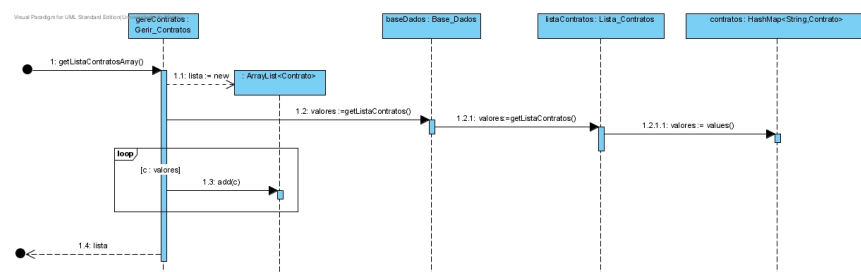


Figura 6.105: Diagrama de sequência : getListaContratosArray()

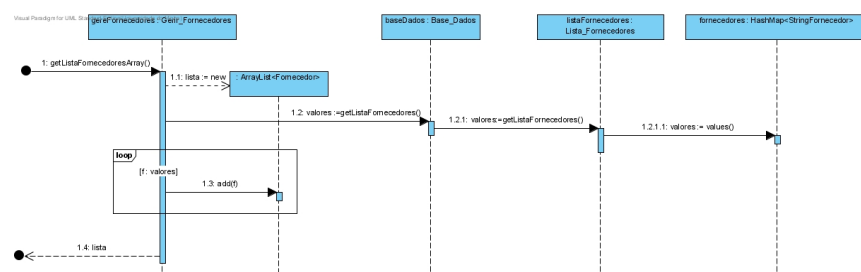


Figura 6.106: Diagrama de sequência : getListaFornecedoresArray()

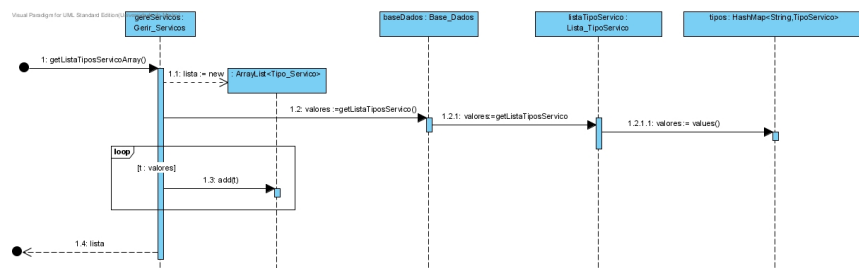


Figura 6.107: Diagrama de sequência : `getListaTiposServicoArray()`

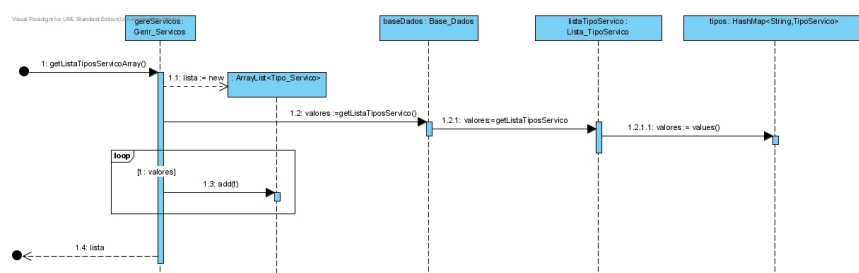


Figura 6.108: Diagrama de sequência : `getListaTiposServicoArray()`

Bibliografia

[1] Apontamentos teóricos da cadeira.

[2] M. Fowler. UML Distilled, Third Edition. Addison-Wesley, 2004.