

Universidade do Minho  
Licenciatura em Engenharia Informática  
Desenvolvimento de Sistemas de Software



## **Trabalho Prático**

Ano Lectivo de 2010/2011

54738 **João Gomes**  
54745 **André Pimenta**  
54802 **Milton Nunes**  
54808 **Cedric Pimenta**  
54825 **Daniel Santos**

2 de Janeiro de 2011

## **Resumo**

Este relatório aborda e explica de forma sucinta a criação de um software para um parque de estacionamento (**PARKUM**).

A criação do software seguiu o **RUP** como linha de guia, e o uso de UML como linguagem de modelado.

Desta forma neste relatório é apresentado a fase de recolha e captura de requisitos, e só após esta fase a implementação do software.

# Conteúdo

<b>Conteúdo</b>	<b>i</b>
<b>Lista de Figuras</b>	<b>ii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Contextualização e apresentação do caso de estudo . . . . .	1
1.2 Motivação e objectivos . . . . .	1
1.3 Estrutura . . . . .	1
<b>2 Desenvolvimento</b>	<b>2</b>
2.1 Análise do problema . . . . .	2
2.1.1 Captura e análise de requisitos . . . . .	2
2.1.2 Modelo de domínio . . . . .	2
2.1.3 Diagrama de Use Case do sistema . . . . .	4
2.1.4 Diagramas de Sequencia . . . . .	22
2.1.5 Refinamento . . . . .	22
2.1.6 Diagramas de sequencia finais . . . . .	30
2.1.7 Sub-Sistemas . . . . .	52
2.1.8 Diagrama de Use Case de sistema refinado . . . . .	53
2.1.9 Diagramas de actividades . . . . .	57
2.1.10 Diagramas de estado . . . . .	61
2.2 Solução do problema . . . . .	62
2.2.1 Base de dados persistente . . . . .	66
2.2.2 Interface . . . . .	67
2.2.3 Manual de utilização . . . . .	67
2.3 Ferramentas utilizadas . . . . .	69
<b>3 Conclusão</b>	<b>70</b>
<b>Bibliografia</b>	<b>71</b>
<b>4 ANEXOS</b>	<b>72</b>

# **Lista de Figuras**

2.1	Modelo de dominio . . . . .	3
2.2	Diagrama de Use Case de sistema . . . . .	4
2.3	Descrição textual Use Case de Entrada . . . . .	5
2.4	Descrição textual Use Case de Entrada pré-registado . . . . .	5
2.5	Descrição textual Use Case de Entrada pré-registado através de chamada wirless . . . . .	6
2.6	Descrição textual Use Case de Entrada pré-registado através de cartão . . . . .	6
2.7	Descrição textual Use Case de Entrada pré-registado através de pré-registo . . . . .	7
2.8	Descrição textual Use Case de Entrada pré-registado através de via verde . . . . .	7
2.9	Descrição textual Use Case de Entrada anonima . . . . .	8
2.10	Descrição textual Use pedir bilhete . . . . .	8
2.11	Descrição textual Use Case de sair do parque . . . . .	9
2.12	Descrição textual Use Case de sair do parque pré-registado . . . . .	9
2.13	Descrição textual Use Case de sair do pré-registado através de chamada wirless . . . . .	10
2.14	Descrição textual Use Case de sair do pré-registado através de cartão . . . . .	10
2.15	Descrição textual Use Case de sair do pré-registado através de via verde . . . . .	11
2.16	Descrição textual Use Case de sair do pré-registado através de matricula pré registada	11
2.17	Descrição textual Use Case de sair do parque anonimamente . . . . .	12
2.18	Descrição textual Use Case de efectuar pagamento . . . . .	12
2.19	Descrição textual Use Case de efectuar pagamento com dinheiro . . . . .	13
2.20	Descrição textual Use Case de efectuar pagamento com cartão . . . . .	13
2.21	Descrição textual Use Case de efectuar pagamento extra . . . . .	14
2.22	Descrição textual Use Case de emitir recibo . . . . .	14
2.23	Descrição textual Use Case de listar clientes com pagamento em atraso . . . . .	15
2.24	Descrição textual Use Case de processar pagamentos via verde . . . . .	15
2.25	Descrição textual Use Case de gerar relatorios sobre clientes . . . . .	15
2.26	Descrição textual Use Case de gerir fichas de clientes . . . . .	16
2.27	Descrição textual Use Case de criar fichas de clientes . . . . .	16
2.28	Descrição textual Use Case editar de fichas clientes . . . . .	16
2.29	Descrição textual Use Case de remover fichas de clientes . . . . .	17
2.30	Descrição textual Use Case tartar estatisticas . . . . .	17
2.31	Descrição textual Use Case de elaborar relatorios de maquinas de pagamento . . . . .	18
2.32	Descrição textual Use Case de elaborar relatorios . . . . .	18
2.33	Descrição textual Use Case de elaborar relatorios de entradas e saidas . . . . .	19
2.34	Descrição textual Use Case de processar facturação . . . . .	19
2.35	Descrição textual Use Case de processar facturação em dinheiro . . . . .	20
2.36	Descrição textual Use Case de processar facturação resultante de via verde . . . . .	20
2.37	Descrição textual Use Case de processar facturação de debito de cartão . . . . .	21

2.38	Diagrama de sequencia de Entrada versão 1.0 . . . . .	22
2.39	Diagrama de sequencia de Entrada versão 2.0 . . . . .	23
2.40	Diagrama de sequencia de Entrada versão 3.0 . . . . .	24
2.41	Diagrama de sequencia de Efectuar pagamento versão 1.0 . . . . .	25
2.42	Diagrama de sequencia de Efectuar pagamento versão 2.0 . . . . .	26
2.43	Diagrama de sequencia de Efectuar pagamento versão 3.0 . . . . .	27
2.44	Diagrama de sequencia de Efectuar pagamento versão 4.0 . . . . .	28
2.45	Diagrama de sequencia de Saída versão 1.0 . . . . .	29
2.46	Diagrama de sequencia de Saída versão 2.0 . . . . .	29
2.47	Diagrama de sequencia de Entrada . . . . .	30
2.48	Diagrama de sequencia de Entrar no parque pre-registado . . . . .	31
2.49	Diagrama de sequencia de Entrar chamada wireless . . . . .	32
2.50	Diagrama de sequencia de Entrar cartão de cliente . . . . .	33
2.51	Diagrama de sequencia de Entrar matricula pre-registada . . . . .	34
2.52	Diagrama de sequencia de Entrar via-verde . . . . .	35
2.53	Diagrama de sequencia de Entrar no parque anonima . . . . .	35
2.54	Diagrama de sequencia de Pedir bilhete . . . . .	36
2.55	Diagrama de sequencia de Actualizar info nível e painéis - Entrada . . . . .	36
2.56	Diagrama de sequencia de Estacionar . . . . .	37
2.57	Diagrama de sequencia de Subir piso . . . . .	37
2.58	Diagrama de sequencia de Descer piso . . . . .	38
2.59	Diagrama de sequencia de Registar Movimento . . . . .	38
2.60	Diagrama de sequencia de Informar Paineis Movimento . . . . .	38
2.61	Diagrama de sequencia de Consultar ocupação . . . . .	39
2.62	Diagrama de sequencia de Sair do parque . . . . .	39
2.63	Diagrama de sequencia de Sair do parque pre-registado . . . . .	40
2.64	Diagrama de sequencia de Sair através chamada wireless . . . . .	40
2.65	Diagrama de sequencia de Sair através cartão cliente . . . . .	41
2.66	Diagrama de sequencia de Sair através Via Verde . . . . .	41
2.67	Diagrama de sequencia de Sair do parque normal . . . . .	42
2.68	Diagrama de sequencia de Registar Saída . . . . .	42
2.69	Diagrama de sequencia de Sair através matricula pré-registada . . . . .	43
2.70	Diagrama de sequencia de Actualizar info nos painéis de nível e de entrada - Saída . . . . .	43
2.71	Diagrama de sequencia de efectuar pagamento . . . . .	44
2.72	Diagrama de sequencia de Pagar com dinheiro . . . . .	45
2.73	Diagrama de sequencia de Pagar cartão . . . . .	45
2.74	Diagrama de sequencia de Pagar extra . . . . .	46
2.75	Diagrama de sequencia de Emitir recibo . . . . .	47
2.76	Diagrama de sequencia de Atribuir troco . . . . .	47
2.77	Diagrama de sequencia de Registar pagamento . . . . .	47
2.78	Diagrama de sequencia de Manutencao . . . . .	47
2.79	Diagrama de sequencia de Listar clientes com pagamentos em atraso . . . . .	48
2.80	Diagrama de sequencia de Gerir pagamentos dos clientes via verde . . . . .	48
2.81	Diagrama de sequencia de Gerar relatorios de cliente . . . . .	48
2.82	Diagrama de sequencia de Gerir fichas de cliente . . . . .	49
2.83	Diagrama de sequencia de Criar ficha . . . . .	49
2.84	Diagrama de sequencia de Editar ficha . . . . .	50

2.85	Diagrama de sequencia de Remover ficha . . . . .	50
2.86	Diagrama de Tratar Estatisticas . . . . .	50
2.87	Diagrama de sequencia elaborar relatorios de maquinas de pagamentos . . . . .	51
2.88	Diagrama de elaborar relatorios . . . . .	51
2.89	Diagrama de elaborar Relatorios diarios de entradas e saidas . . . . .	51
2.90	Diagrama de Processar facturacao . . . . .	52
2.91	Diagrama de Processar Facturacao em dinheiro . . . . .	52
2.92	Diagrama de Processar Facturacao atraves de via-verde . . . . .	52
2.93	Diagrama de Processar Facturacao atraves de debito . . . . .	53
2.94	Diagrama de Use Case de sistema refinado . . . . .	54
2.95	Use cases de entrada no parque . . . . .	55
2.96	Use cases de estacionamento . . . . .	55
2.97	Use cases de pagamento . . . . .	56
2.98	Use cases de saida do parque . . . . .	56
2.99	Use cases de gestão de clientes . . . . .	57
2.100	Use cases de facturação e relatorios . . . . .	57
2.101	Diagrama de Actividade de entrada no parque . . . . .	58
2.102	Diagrama de Actividade de estacionamento . . . . .	59
2.103	Diagrama de Actividade de saida . . . . .	60
2.104	Diagrama de Actividade de pagamento . . . . .	61
2.105	Diagrama de Estado do portal de entrada . . . . .	61
2.106	Diagrama de Estado da Maquina de pagamento . . . . .	62
2.107	Diagrama de Estado do portal de saida . . . . .	62
2.108	Diagrama de Class . . . . .	63
2.109	Diagrama de Class package gestão de lugares . . . . .	64
2.110	Diagrama de Class package pagamentos . . . . .	64
2.111	Diagrama de Class package interface . . . . .	64
2.112	Diagrama de Class package sensores . . . . .	65
2.113	Diagrama de Class package Parkum . . . . .	65
2.114	Diagrama de Class package Parkum . . . . .	66
2.115	Opções de interface garfica 1 . . . . .	67
2.116	Opções de interface garfica 2 . . . . .	67
2.117	Opções de interface garfica 3 . . . . .	68
2.118	Opções de interface garfica 4 . . . . .	68
2.119	Opções de interface garfica 5 . . . . .	68
2.120	Opções de interface garfica 6 . . . . .	68
2.121	Opções de interface garfica 7 . . . . .	68
2.122	Opções de interface garfica 8 . . . . .	68
2.123	Opções de interface garfica 9 . . . . .	69
2.124	Opções de interface garfica 10 . . . . .	69
2.125	Opções de interface garfica 11 . . . . .	69
2.126	Opções de interface garfica 12 . . . . .	69
4.1	André Pimenta - a54745 . . . . .	73
4.2	Cedric Pimenta - a54808 . . . . .	73
4.3	Daniel Santos - a54825 . . . . .	73
4.4	João Miguel - a54738 . . . . .	73



# Capítulo 1

## Introdução

### 1.1 Contextualização e apresentação do caso de estudo

Neste trabalho prático foi pedido o desenvolvimento de um software para a gestão de um parque de estacionamento, **PARKUM**.

Sendo este software um pouco complexo, temos de ser capazes de o criar de forma organizada, eficaz e que satisfaça todos os requisitos apresentados pelo cliente( neste caso o enunciado do trabalho prático). Para isso devemos ser capazes de identificar todos os utilizadores do software, todas as tarefas que este deve satisfazer e a melhor forma de o fazer.

Desta forma temos de saber identificar **QUEM** faz o **QUE**, **QUANDO** e deve-o fazer **DURANTE** quanto tempo. Para isto devemos seguir Rational Unifield Process (RUP), apostando numa modelação orientada aos objectos e seguir o RUP usando UML como notação de modelação.

### 1.2 Motivação e objectivos

Este trabalho prático tem como principal objectivo a utilização do RUP como processo de base para o desenvolvimento de software. Seguido o RUP devemos então apostar na modelação orientada a objectos.

Para a modelação deste projecto devemos também usar UML(standard da OMG) como notação de modelação. Devemos também tentar definir alguma metodologia nos modelos UML.

Para além do uso do RUP este trabalho prático é uma forma de nos preparar para o desenvolvimento de grandes projectos seguindo guias lógicas e coerentes para que possamos desenvolver este da forma mais correcta possível mas também ser capazes de ter uma comunicação com o requerente do projecto e tentar satisfazer todas as necessidades que este pretende no software.

Podemos então de forma resumida afirmar que este trabalho prático tem como principais objectivo e motivação a preparação e criação de software de forma eficaz e satisfazendo todos os requisitos do cliente. Isto seguindo o RUP e usando UML como linguagem de modelação.

### 1.3 Estrutura

O presente relatório é constituído por 3 capítulos: **Introdução**, onde apresentamos o nosso problema e a forma como o tencionamos resolver, **Desenvolvimento** que se deve essencialmente duas secções, análise do problema e resolução deste, e por ultimo a **Conclusão** onde analisamos de forma sucinta a criação do software.

# Capítulo 2

## Desenvolvimento

### 2.1 Análise do problema

#### 2.1.1 Captura e análise de requisitos

Está é uma das etapas mais importantes de todo o desenvolvimento do projecto.

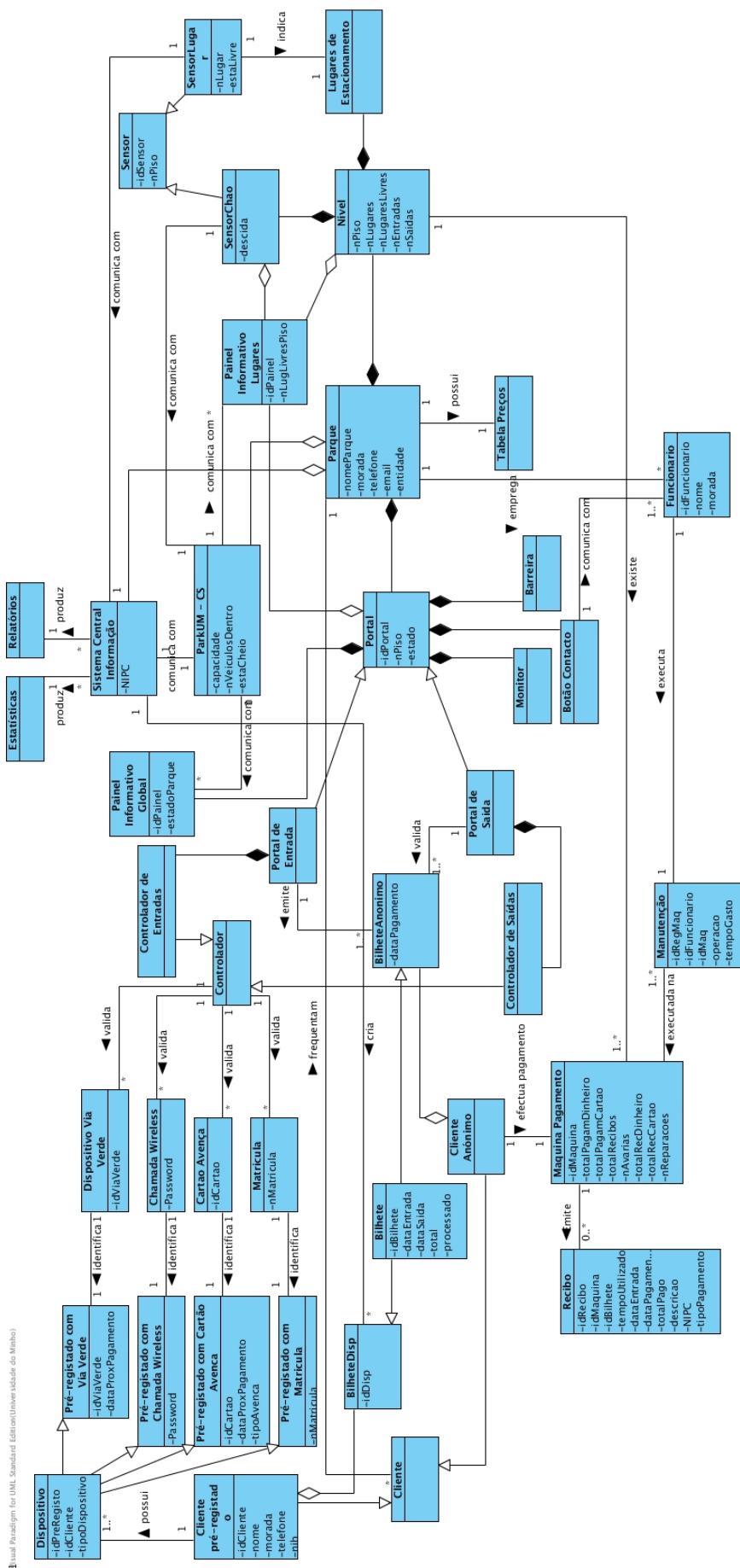
O projecto que desenvolvemos depende da forma com analisamos o problema em si, e fundamental dos dados que capturamos, neste caso através da descrição do funcionamento do parque que é apresentada no enunciado no enunciado.

Sendo que a etapa de captura e análise de requisitos é fundamental no trabalho, esta pode estar mais sujeita a erros e consequentemente a uma maior carga de tempo.

De forma a conseguirmos capturar todos os requesitos e funcionalidades que o parque vai necessitar devemos começar por criar o **modelo de dominio** juntamente com o **diagrama de use case** do sistema que irá funcionar como uma "blackbox"do sistema.

#### 2.1.2 Modelo de domínio

No modelo de dominio tentamos englobar todo o vocabulário fundamental do dominio do problema, ou seja temos todas as potencias entidades, funconalidades, utilizadores, dispositos entre outros de todos os sistema. Este são apresentados de forma associada entre eles para que se possa perceber a ligação entre os mesmos, ou seja porque se vai ligar e o porquê de se ligar e com quem vai interagir. Desta forma apresentamos o seguinte modelo de dominio do **PARKUM**.



**Figura 2.1:** Modelo de dominio

### 2.1.3 Diagrama de Use Case do sistema

Podemos começar por explicar os utilizadores fundamentais do sistema, que serão os actores do nosso diagrama.

O principal actor do sistema será sem qualquer pé de duvida o cliente do parque. Este poderá ser anonimo caso entre de forma anónima ou um cliente pre-registado caso entre com um dispositivo no parque e possua um pré-registo, desta forma pode colocar o cliente como principal actor do diagrama de use case, sendo que este poderá ou não ser cliente pré-registado.

No entanto teremos tarefas que têm de ser executadas como trabalho de fundo, são estas as tarefas de gestão. Estas tarefas serão realizadas pelo actor gestor informatico que é responsável pelas tarefas de gestão do parque.

No diagrama de use case do sistema (PARKUM) referimos os use case fundamentais, que de forma objectiva podemos observar que *Entrar*, *Estacionar*, *Pagar* e *Sair* serão os use case principais em que os actores *cliente anonimo* e *cliente pré-registado* vão interagir com o sistema.

No entanto o sistema não funciona apenas com as "acções" dos clientes sendo que é necessário as operações de gestão do sistema.

Aqui podemos destacar os use case *Porcessar facturação*, *Elaborar relatorios*, *Tratar estatisticas* e *Gerir fichas de clientes*. Estes use cases retratam as funcionalidades fundamentais que o actor *Gestor informatico* vai efectuar no sistema.

Desta forma podemos ver este diagrama como uma "blackbox" do sistema, onde as funções fundamentais e os utilizadores fundamentais para o sistema estão presentes.

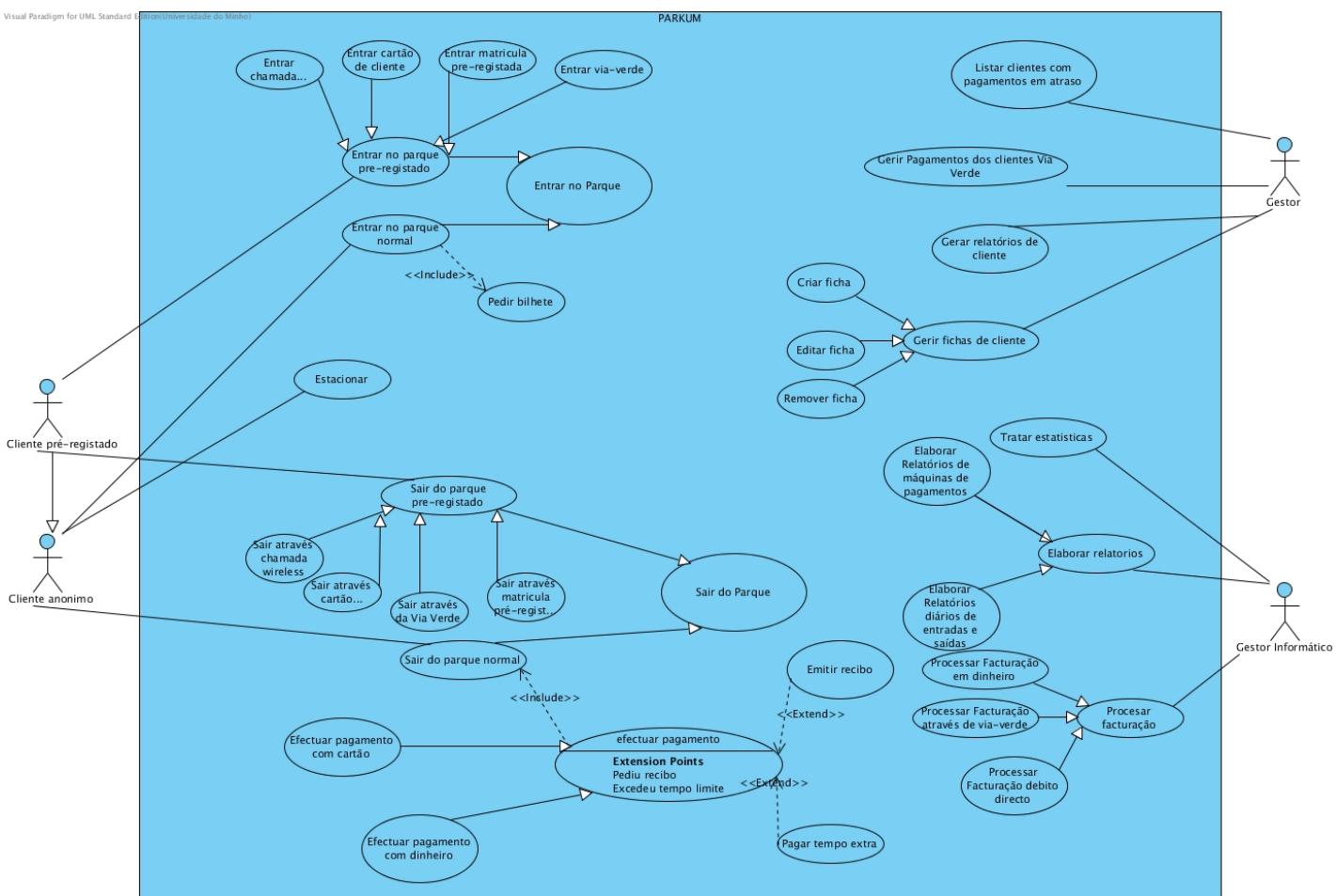


Figura 2.2: Diagrama de Use Case de sistema

Após a construção do diagrama de use case principal fizemos a descrição textual de cada um, de forma a explicar de forma mais detalhada o funcionamento de cada use case.

<b>Super Use Case</b>		
<b>Author</b>	ricardo	
<b>Date</b>	Dec 9, 2010 11:49:50 PM	
<b>Brief Description</b>		
<b>Preconditions</b>		
<b>Post-conditions</b>	O Cliente entra no Parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 O Cliente obtém identificação.	O Sistema confirma identificação.
	2	O Sistema obtém data de entrada.
	3	O Sistema cria registo de entrada.
	4	O Sistema levanta a barreira.
	5	
	6 O Cliente entra no parque.	O Sistema detecta a passagem do Cliente.
	7	O Sistema baixa a barreira.
	8	
	9	<<include>>: actualizar info nível e painéis.
<b>Exception 1</b>	<b>Actor Input</b>	<b>System Response</b>
2a - Não é possível confirmar identificação.	1	O Sistema apresenta mensagem de erro.
	2	STOP
<b>Exception 2</b>	<b>Actor Input</b>	<b>System Response</b>
3a - Não é possível criar registo.	1	apresenta mensagem de erro.
<b>Exception 3</b>	<b>Actor Input</b>	<b>System Response</b>
2b - Extension Point: Não é confirmada a identificação.	1	extended by: Resolver bloqueio/emergência.
<b>Exception 4</b>	<b>Actor Input</b>	<b>System Response</b>
5a - Extension Point: Barreira não levanta.	1	extended by: Resolver bloqueio/emergência.

Figura 2.3: Descrição textual Use Case de Entrada

<b>Super Use Case</b>	Entrar.	
<b>Author</b>	ricardo	
<b>Date</b>	Dec 9, 2010 11:43:19 PM	
<b>Brief Description</b>		
<b>Preconditions</b>	O Cliente encontra-se registado no parque.	
<b>Post-conditions</b>	O Cliente entra no parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 O Cliente usa a identificação que lhe está atribuída.	O Sistema valida identificação.
	2	O Sistema obtém hora e data.
	3	O Sistema cria bilhete de dispositivo.
	4	O Sistema regista bilhete.
	5	
	6	O Sistema levanta barreira.
	7 O Cliente entra no parque.	O Sistema baixa barreira.
	8	
	9	<include> actualizar info nível e painéis.
<b>Exception 1</b>	<b>Actor Input</b>	<b>System Response</b>
2a - O Sistema não valida identificação.	1	O Sistema mostra mensagem de erro de leitura no visor.
	2	STOP
<b>Exception 2</b>	<b>Actor Input</b>	<b>System Response</b>
6a - Barreira não levanta.	1	O Sistema remove referência ao bilhete do conjunto de bilhetes registados.
	2	O Sistema elimina bilhete anteriormente criado.
	3	extended by: Resolver Bloqueio/emergência.
	4	STOP

Figura 2.4: Descrição textual Use Case de Entrada pré-registrado

<b>Super Use Case</b>	Entra no parque pre-registrado.	
<b>Author</b>	ricardo	
<b>Date</b>	Dec 9, 2010 9:36:15 PM	
<b>Brief Description</b>	Entrar no parque através de dispositivo com wireless	
<b>Preconditions</b>	Existem lugares livres no parque.	
<b>Post-conditions</b>	O Cliente entra no parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 O Cliente conecta-se à rede.	O Sistema pede código acesso.
	2 O Cliente digita código acesso.	O Sistema verifica password
	3	O Sistema gera id bilhete.
	4	O Sistema cria registo de bilhete.
	5	O Sistema guarda registo.
	6	O Sistema levanta barreira.
	7	
	8	
	9 O Cliente entra no parque.	O Sistema baixa barreira.
	10	<<include>>: Atualizar info nível e painéis
	11	
<b>Exception 1</b> 1a - Password não é válida.	<b>Actor Input</b>	<b>System Response</b>
	1	O Sistema notifica falha de autenticação.
<b>Exception 2</b> 8a - Barreira danificada.	2	STOP
	<b>Actor Input</b>	<b>System Response</b>
	1	O Sistema remove referência ao bilhete do conjunto de bilhetes registados.
<b>Exception 3</b> 9a - Bloqueio emergencia.	2	O Sistema elimina bilhete.
	3	<b>extended by:</b> Resolver Bloqueio/emergencia.

Figura 2.5: Descrição textual Use Case de Entrada pré-registrado através de chamada wireless

<b>Super Use Case</b>	Entra no parque pre-registrado.	
<b>Author</b>	ricardo	
<b>Date</b>	Dec 9, 2010 10:51:20 PM	
<b>Brief Description</b>		
<b>Preconditions</b>	Existem lugares livres no parque.	
<b>Post-conditions</b>	O Cliente entra no parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 O Cliente encosta cartão ao detector de cartões.	
	2	O Sistema obtém ID do cartão.
	3	O Sistema verifica se cartão está registado.
	4	O Sistema gera id bilhete.
	5	O Sistema cria registo bilhete.
	6	O Sistema guarda registo.
	7	O Sistema abre barreira.
	8 O Cliente entra no parque.	O Sistema fecha barreira.
	9	<include>: Atualizar info nível e painéis.
	10	
<b>Exception 1</b> 3a - O Sistema não consegue ler o cartão.	<b>Actor Input</b>	<b>System Response</b>
	1	O Sistema mostra mensagem de erro de leitura no visor.
<b>Exception 2</b> 4a - Dispositivo não está registrado.	2	STOP
	<b>Actor Input</b>	<b>System Response</b>
<b>Exception 3</b> 6a - Barreira está danificada.	1	O Sistema mostra mensagem de falta de registo.
	2	STOP
	<b>Actor Input</b>	<b>System Response</b>
<b>Exception 4</b> 7a - Bloqueio emergencia.	1	O Sistema remove referência ao bilhete do conjunto de bilhetes registados.
	2	O Sistema elimina bilhete.
	3	<b>extended by:</b> Resolver Bloqueio/emergencia.

Figura 2.6: Descrição textual Use Case de Entrada pré-registrado através de cartão

<b>Super Use Case</b>	Entrar no parque pré-registrado.	
<b>Author</b>	ricardo	
<b>Date</b>	Dec 9, 2010 11:28:36 PM	
<b>Brief Description</b>	Cliente entra no parque através de detecção de matrícula.	
<b>Preconditions</b>	Existem lugares livres no parque.	
<b>Post-conditions</b>	O Cliente entra no parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 O Cliente pára veículo em frente a barreira.	
	2	O Sistema fotografa veículo.
	3	O Sistema obtém código da matrícula.
	4	O Sistema verifica se matrícula se encontra registada.
	5	O Sistema cria registo de entrada.
	6	O Sistema guarda registo de entrada.
	7	O Sistema levanta a barreira.
	8 O Cliente entra no parque.	
	9	O Sistema baixa barreira.
	10	<<include>>: Atualizar info nível e painéis.
<b>Exception 1</b> 3a - matrícula ilegível	<b>Actor Input</b>	<b>System Response</b>
1		O Sistema apresenta mensagem de erro de matrícula ilegível.
	2	STOP
<b>Exception 2</b> 4a - Não existe registo da matrícula	<b>Actor Input</b>	<b>System Response</b>
1		O Sistema apresenta mensagem erro veículo não registado.
	2	STOP
<b>Exception 3</b> 7a - Barreira está danificada	<b>Actor Input</b>	<b>System Response</b>
1		O Sistema remove referência ao bilhete do conjunto de bilhetes registados.
	2	O Sistema elimina bilhete.
	3	<b>extended by:</b> Resolver Bloqueio/emergencia.

Figura 2.7: Descrição textual Use Case de Entrada pré-registrado através de pré-registro

<b>Super Use Case</b>	Entrar.	
<b>Author</b>	ricardo	
<b>Date</b>	Dec 9, 2010 11:43:19 PM	
<b>Brief Description</b>		
<b>Preconditions</b>	O Cliente encontra-se registrado no parque.	
<b>Post-conditions</b>	O Cliente entra no parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 O Cliente usa a identificação que lhe está atribuída.	
	2	O Sistema valida identificação.
	3	O Sistema obtém hora e data.
	4	O Sistema cria bilhete de dispositivo.
	5	O Sistema regista bilhete.
	6	O Sistema levanta barreira.
	7 O Cliente entra no parque.	
	8	O Sistema baixa barreira.
	9	<include> actualizar info nível e painéis.
<b>Exception 1</b> 2a - O Sistema não valida identificação.	<b>Actor Input</b>	<b>System Response</b>
1		O Sistema mostra mensagem de erro de leitura no visor.
	2	STOP
<b>Exception 2</b> 6a - Barreira não levanta.	<b>Actor Input</b>	<b>System Response</b>
1		O Sistema remove referência ao bilhete do conjunto de bilhetes registados.
	2	O Sistema elimina bilhete anteriormente criado.
	3	<b>extended by:</b> Resolver Bloqueio/emergência.
	4	STOP

Figura 2.8: Descrição textual Use Case de Entrada pré-registrado através de via verde

<b>Super Use Case</b>	Entrar	
<b>Author</b>	ricardo	
<b>Date</b>	Dec 10, 2010 12:10:20 AM	
<b>Brief Description</b>		
<b>Preconditions</b>		
<b>Post-conditions</b>	O Cliente entra no parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 <include>: Pedir bilhete	
	2 O Cliente retira bilhete	
	3	O Sistema levanta barreira
	4 O Cliente entra no parque	
	5	O Sistema baixa barreira
	6	<include>: actualizar info nível e painéis
<b>Exception 1</b> 6a - Barreira não levanta	<b>Actor Input</b>	<b>System Response</b>
	1	O Sistema elimina bilhete anteriormente criado
	2	<b>extended by:</b> Resolver Bloqueio/emergência
	3	STOP

Figura 2.9: Descrição textual Use Case de Entrada anónima

<b>Super Use Case</b>		
<b>Author</b>	ricardo	
<b>Date</b>	Dec 12, 2010 11:58:29 AM	
<b>Brief Description</b>		
<b>Preconditions</b>		
<b>Post-conditions</b>	O Cliente recebe o bilhete e entra no parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 O Cliente pressiona botão de emissão de bilhete	
	2	O Sistema verifica se o parque não está completo
	3	O Sistema gera identificador
	4	O Sistema obtém data e hora actual
	5	O Sistema cria bilhete
	6	O Sistema imprime bilhete
<b>Exception 1</b> 2a - O Parque está completo	<b>Actor Input</b>	<b>System Response</b>
	1	O Sistema apresenta mensagem de parque cheio nos painéis
	2	STOP

Figura 2.10: Descrição textual Use Case pedir bilhete

<b>Super Use Case</b>		
<b>Author</b>	Cedric	
<b>Date</b>	14/Dez/2010 17:01:50	
<b>Brief Description</b>	O Cliente pretende sair do parque.	
<b>Preconditions</b>	O Cliente está no parque e ou entrou pedindo bilhete ou entrou como sendo cliente pré-registrado.	
<b>Post-conditions</b>	<b>Sucesso:</b> A barreira é levantada e o Cliente sai do parque. <b>Insucesso:</b> A barreira não levanta e o Cliente não sai do parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 <<include>>: Registrar Saída	
	2	O Sistema levanta a barreira.
	3 O Cliente sai do parque.	
	4	O Sistema detecta a passagem do Cliente.
	5	O Sistema baixa a barreira.
	6	<<include>>: Actualizar info nos painéis de nível e de entrada.
<b>Exception 1</b> 2a - Extension Point: Barreira está danificada	<b>Actor Input</b>	<b>System Response</b>
	1	<b>extended by:</b> Resolver Bloqueio/emergencia
	2	STOP

Figura 2.11: Descrição textual Use Case de sair do parque

<b>Super Use Case</b>	Sair do parque.	
<b>Author</b>	Cedric	
<b>Date</b>	14/Dez/2010 21:56:15	
<b>Brief Description</b>	O Cliente pretende sair do parque.	
<b>Preconditions</b>	O Cliente está no parque e entrou como sendo um cliente pré-registrado.	
<b>Post-conditions</b>	<b>Sucesso:</b> A barreira é levantada e o Cliente sai do parque. <b>Insucesso:</b> A barreira não levanta e o Cliente não sai do parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1	<<include>>: Registrar Saída
	2	O Sistema calcula e guarda o preço da estadia no parque.
	3	O Sistema levanta a barreira.
	4 O Cliente sai do parque.	
	5	O Sistema detecta a passagem do Cliente.
	6	O Sistema baixa a barreira.
	7	<<include>>: Actualizar info nos painéis de nível e de entrada.
<b>Exception 1</b> 4a - Extension Point: Barreira está danificada	<b>Actor Input</b>	<b>System Response</b>
	1	<b>extended by:</b> Resolver Bloqueio/emergencia
	2	STOP

Figura 2.12: Descrição textual Use Case de sair do parque pré-registrado

<b>Super Use Case</b>	Sair do parque pré-registrado.	
<b>Author</b>	Cedric	
<b>Date</b>	14/Dez/2010 22:44:51	
<b>Brief Description</b>	O Cliente pretende sair do parque.	
<b>Preconditions</b>	O Cliente está no parque e entrou como cliente pré-registrado através de chamada wireless	
<b>Post-conditions</b>	<b>Sucesso:</b> A barreira é levantada e o Cliente sai do parque. <b>Insucesso:</b> A barreira não levanta e o Cliente não sai do parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 O Cliente liga o sistema wireless	O Sistema recebe a conexão wireless
	2 O Cliente escolhe a rede wireless do parque	O Sistema valida a conexão
	3	O Sistema guarda data de saída
	4	O Sistema calcula e guarda o preço da estadia no parque
	5	O Sistema levanta a barreira
	6	O Sistema calcula e guarda o preço da estadia no parque
	7	O Sistema baixa a barreira
	8 O Cliente sai do parque	<<include>>: Atualizar info nos painéis de nível e de entrada
	9	O Sistema detecta a passagem do Cliente
	10	O Sistema baixa a barreira
	11	<<include>>: Atualizar info nos painéis de nível e de entrada
<b>Exception 1</b> <b>7a - Extension Point:</b> Barreira está danificada	<b>Actor Input</b>	<b>System Response</b>
	1	extended by: Resolver Bloqueio/emergência
	2	STOP
<b>Exception 2</b> <b>4a - O Sistema não valida a conexão</b>	<b>Actor Input</b>	<b>System Response</b>
	1	O Sistema apresenta mensagem de erro
	2	STOP

Figura 2.13: Descrição textual Use Case de sair do pré-registrado através de chamada wireless

<b>Super Use Case</b>	Sair do parque pré-registrado.	
<b>Author</b>	Cedric	
<b>Date</b>	14/Dez/2010 23:15:05	
<b>Brief Description</b>	O Cliente pretende sair do parque.	
<b>Preconditions</b>	O Cliente está no parque e entrou como cliente pré-registrado através do cartão cliente.	
<b>Post-conditions</b>	<b>Sucesso:</b> A barreira é levantada e o Cliente sai do parque. <b>Insucesso:</b> A barreira não levanta e o Cliente não sai do parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 O Cliente aproxima o cartão cliente do local devido no portal de saída	O Sistema lê o cartão cliente
	2	O Sistema valida o cartão cliente
	3	O Sistema guarda data de saída
	4	O Sistema calcula e guarda o preço da estadia no parque
	5	O Sistema levanta a barreira
	6	O Sistema detecta a passagem do Cliente
	7 O Cliente sai do parque	O Sistema baixa a barreira
	8	<<include>>: Atualizar info nos painéis de nível e de entrada
	9	
	10	
<b>Alternative 1</b> <b>6a - Extension Point:</b> Barreira está danificada	<b>Actor Input</b>	<b>System Response</b>
	1	extended by: Resolver Bloqueio/emergência
	2	STOP
<b>Exception 1</b> <b>3a - O Sistema não valida o cartão cliente</b>	<b>Actor Input</b>	<b>System Response</b>
	1	O Sistema apresenta mensagem de erro
	2	STOP

Figura 2.14: Descrição textual Use Case de sair do pré-registrado através de cartão

<b>Super Use Case</b>	Sair do parque pre-registrado.	
<b>Author</b>	Cedric	
<b>Date</b>	14/Dez/2010 23:58:40	
<b>Brief Description</b>	O Cliente pretende sair do parque.	
<b>Preconditions</b>	O Cliente está no parque e entrou como cliente pré-registrado através da Via Verde	
<b>Post-conditions</b>	<b>Sucesso:</b> A barreira é levantada e o Cliente sai do parque. <b>Insucesso:</b> A barreira não levanta e o Cliente não sai do parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 O Cliente pára junto do portal de saída	O Sistema detecta o dispositivo da Via Verde
	2	O Sistema valida o dispositivo da Via Verde
	3	O Sistema guarda a data de saída
	4	O Sistema calcula e guarda o preço da estadia no parque
	5	O Sistema levanta a barreira
	6	
	7 O Cliente sai do parque	O Sistema detecta a passagem do Cliente
	8	O Sistema baixa a barreira
	9	<<include>>: Actualizar info nos painéis de nível e de entrada
	10	
<b>Exception 1</b> 3a - O Sistema não valida dispositivo Via Verde	<b>Actor Input</b>	<b>System Response</b>
	1	O Sistema apresenta mensagem de erro
	2	STOP
<b>Alternative 2</b> 6a - Extension <b>Point:</b> Barreira está danificada	<b>Actor Input</b>	<b>System Response</b>
	1	extended by: Resolver Bloqueio/emergência
	2	STOP

Figura 2.15: Descrição textual Use Case de sair do pré-registrado através de via verde

<b>Super Use Case</b>	Sair do parque pre-registrado.	
<b>Author</b>	Cedric	
<b>Date</b>	14/Dez/2010 23:42:01	
<b>Brief Description</b>	O Cliente pretende sair do parque.	
<b>Preconditions</b>	O Cliente está no parque e entrou como cliente pré-registrado através do reconhecimento de matrícula.	
<b>Post-conditions</b>	<b>Sucesso:</b> A barreira é levantada e o Cliente sai do parque. <b>Insucesso:</b> A barreira não levanta e o Cliente não sai do parque.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 O Cliente pára junto a um portal de saída	
	2	O Sistema lê a matrícula do veículo do Cliente
	3	O Sistema valida a matrícula
	4	O Sistema guarda a data de saída
	5	O Sistema calcula e guarda o preço da estadia no parque
	6	O Sistema levanta a barreira
	7 O Cliente sai do parque	O Sistema detecta a passagem do Cliente
	8	O Sistema baixa a barreira
	9	<<include>>: Actualizar info nos painéis de nível e de entrada
	10	
<b>Exception 1</b> 3a - O Sistema não valida matrícula	<b>Actor Input</b>	<b>System Response</b>
	1	O Sistema apresenta mensagem de erro
	2	STOP
<b>Exception 2</b> 6a - Extension <b>Point:</b> Barreira está danificada	<b>Actor Input</b>	<b>System Response</b>
	1	extended by: Resolver Bloqueio/emergência
	2	STOP

Figura 2.16: Descrição textual Use Case de sair do pré-registrado através de matricula pré registada

<b>Super Use Case</b>	Sair do parque																									
<b>Author</b>	Cedric																									
<b>Date</b>	15/Dez/2010 0:20:03																									
<b>Brief Description</b>	O Cliente pretende sair do parque.																									
<b>Preconditions</b>	O Cliente está no parque.																									
<b>Post-conditions</b>	<b>Sucesso:</b> A barreira é levantada e o Cliente sai do parque. <b>Insucesso:</b> A barreira não levanta e o Cliente não sai do parque.																									
<b>Flow of Events</b>		<table border="1"> <thead> <tr> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1 &lt;&lt;include&gt;&gt;: efectuar pagamento.</td> <td></td></tr> <tr> <td>2 O Cliente insere o bilhete.</td> <td>O Sistema lê o bilhete.</td></tr> <tr> <td>3</td> <td>O Sistema valida o bilhete.</td></tr> <tr> <td>4</td> <td>O Sistema verifica que a data de pagamento foi à menos de 10 minutos.</td></tr> <tr> <td>5</td> <td>O Sistema guarda a data de saída.</td></tr> <tr> <td>6</td> <td>O Sistema levanta a barreira.</td></tr> <tr> <td>7</td> <td></td></tr> <tr> <td>8 O Cliente sai do parque.</td> <td>O Sistema detecta a passagem do Cliente.</td></tr> <tr> <td>9</td> <td>O Sistema baixa a barreira.</td></tr> <tr> <td>10</td> <td>&lt;&lt;include&gt;&gt;: Actualizar info nos painéis de nível e de entrada.</td></tr> <tr> <td>11</td> <td></td></tr> </tbody> </table>	Actor Input	System Response	1 <<include>>: efectuar pagamento.		2 O Cliente insere o bilhete.	O Sistema lê o bilhete.	3	O Sistema valida o bilhete.	4	O Sistema verifica que a data de pagamento foi à menos de 10 minutos.	5	O Sistema guarda a data de saída.	6	O Sistema levanta a barreira.	7		8 O Cliente sai do parque.	O Sistema detecta a passagem do Cliente.	9	O Sistema baixa a barreira.	10	<<include>>: Actualizar info nos painéis de nível e de entrada.	11	
Actor Input	System Response																									
1 <<include>>: efectuar pagamento.																										
2 O Cliente insere o bilhete.	O Sistema lê o bilhete.																									
3	O Sistema valida o bilhete.																									
4	O Sistema verifica que a data de pagamento foi à menos de 10 minutos.																									
5	O Sistema guarda a data de saída.																									
6	O Sistema levanta a barreira.																									
7																										
8 O Cliente sai do parque.	O Sistema detecta a passagem do Cliente.																									
9	O Sistema baixa a barreira.																									
10	<<include>>: Actualizar info nos painéis de nível e de entrada.																									
11																										
<b>Exception 1:</b> 7a - Extension Point: Barreira está danificada		<table border="1"> <thead> <tr> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>extended by: Resolver Bloqueio/emergência</td></tr> <tr> <td>2</td> <td>O Sistema devolve bilhete.</td></tr> <tr> <td>3</td> <td>STOP</td></tr> </tbody> </table>	Actor Input	System Response	1	extended by: Resolver Bloqueio/emergência	2	O Sistema devolve bilhete.	3	STOP																
Actor Input	System Response																									
1	extended by: Resolver Bloqueio/emergência																									
2	O Sistema devolve bilhete.																									
3	STOP																									
<b>Alternative 2</b> 5a - Já passou 10 minutos desde que o bilhete foi pago.		<table border="1"> <thead> <tr> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>O Sistema informa o cliente que passou o limite de tempo.</td></tr> <tr> <td>2</td> <td>O Sistema coloca o atributo "processado" como "false".</td></tr> <tr> <td>3</td> <td>O Sistema devolve o bilhete.</td></tr> <tr> <td>4 Volta ao passo 1</td> <td></td></tr> </tbody> </table>	Actor Input	System Response	1	O Sistema informa o cliente que passou o limite de tempo.	2	O Sistema coloca o atributo "processado" como "false".	3	O Sistema devolve o bilhete.	4 Volta ao passo 1															
Actor Input	System Response																									
1	O Sistema informa o cliente que passou o limite de tempo.																									
2	O Sistema coloca o atributo "processado" como "false".																									
3	O Sistema devolve o bilhete.																									
4 Volta ao passo 1																										
<b>Exception 2</b> 4a - O Sistema não valida o bilhete.		<table border="1"> <thead> <tr> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>O Sistema imprime mensagem de erro no ecrã.</td></tr> <tr> <td>2</td> <td>O Sistema devolve o bilhete.</td></tr> <tr> <td>3</td> <td>STOP</td></tr> </tbody> </table>	Actor Input	System Response	1	O Sistema imprime mensagem de erro no ecrã.	2	O Sistema devolve o bilhete.	3	STOP																
Actor Input	System Response																									
1	O Sistema imprime mensagem de erro no ecrã.																									
2	O Sistema devolve o bilhete.																									
3	STOP																									

Figura 2.17: Descrição textual Use Case de sair do parque anonimamente

<b>Author</b>	pimenta																													
<b>Date</b>	Dec 14, 2010 4:23:02 PM																													
<b>Brief Description</b>	O Cliente entrou no parque requisitando um bilhete.																													
<b>Preconditions</b>																														
<b>Post-conditions</b>	Insucesso: Sistema não se altera Sucesso: Bilhete é registado como pago.																													
<b>Flow of Events</b>		<table border="1"> <thead> <tr> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1 Cliente Introduz bilhete.</td> <td>MPA</td></tr> <tr> <td>2</td> <td>Sistema conta tempo de estadia.</td></tr> <tr> <td>3</td> <td>Sistema calcula valor a pagar.</td></tr> <tr> <td>4</td> <td>Sistema informa do valor a pagar.</td></tr> <tr> <td>5 Cliente paga quantia.</td> <td></td></tr> <tr> <td>6</td> <td>Sistema processa pagamento.</td></tr> <tr> <td>7</td> <td>Sistema pergunta se pretende recibo.</td></tr> <tr> <td>8</td> <td>Sistema devolve bilhete.</td></tr> <tr> <td>9 Cliente recebe bilhete.</td> <td></td></tr> <tr> <td>10</td> <td>Sistema regista pagamento.</td></tr> <tr> <td>11 Cliente recebe recibo.</td> <td></td></tr> <tr> <td>12 Cliente recebe bilhete.</td> <td></td></tr> <tr> <td>13</td> <td>Sistema regista pagamento.</td></tr> </tbody> </table>	Actor Input	System Response	1 Cliente Introduz bilhete.	MPA	2	Sistema conta tempo de estadia.	3	Sistema calcula valor a pagar.	4	Sistema informa do valor a pagar.	5 Cliente paga quantia.		6	Sistema processa pagamento.	7	Sistema pergunta se pretende recibo.	8	Sistema devolve bilhete.	9 Cliente recebe bilhete.		10	Sistema regista pagamento.	11 Cliente recebe recibo.		12 Cliente recebe bilhete.		13	Sistema regista pagamento.
Actor Input	System Response																													
1 Cliente Introduz bilhete.	MPA																													
2	Sistema conta tempo de estadia.																													
3	Sistema calcula valor a pagar.																													
4	Sistema informa do valor a pagar.																													
5 Cliente paga quantia.																														
6	Sistema processa pagamento.																													
7	Sistema pergunta se pretende recibo.																													
8	Sistema devolve bilhete.																													
9 Cliente recebe bilhete.																														
10	Sistema regista pagamento.																													
11 Cliente recebe recibo.																														
12 Cliente recebe bilhete.																														
13	Sistema regista pagamento.																													
<b>Alternative 1</b> 6a. Sistema processa dinheiro a mais.		<table border="1"> <thead> <tr> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Sistema informa do troco.</td></tr> <tr> <td>2</td> <td>Sistema dá troco.</td></tr> <tr> <td>3 Cliente recebe troco.</td> <td></td></tr> <tr> <td>4 Cliente recebe troco.</td> <td></td></tr> <tr> <td>5</td> <td></td></tr> <tr> <td>4</td> <td>Voltar ao passo 7</td></tr> </tbody> </table>	Actor Input	System Response	1	Sistema informa do troco.	2	Sistema dá troco.	3 Cliente recebe troco.		4 Cliente recebe troco.		5		4	Voltar ao passo 7														
Actor Input	System Response																													
1	Sistema informa do troco.																													
2	Sistema dá troco.																													
3 Cliente recebe troco.																														
4 Cliente recebe troco.																														
5																														
4	Voltar ao passo 7																													
<b>Alternative 2</b> 7a - Extension Point: Cliente pede recibo.		<table border="1"> <thead> <tr> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>extended by: Emitir recibo.</td></tr> <tr> <td>2</td> <td>Voltar ao passo 8</td></tr> </tbody> </table>	Actor Input	System Response	1	extended by: Emitir recibo.	2	Voltar ao passo 8																						
Actor Input	System Response																													
1	extended by: Emitir recibo.																													
2	Voltar ao passo 8																													
<b>Alternative 3</b> 2a - Extension Point: Excedeu tempo de estadia.		<table border="1"> <thead> <tr> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>extended by: Pagar tempo extra</td></tr> </tbody> </table>	Actor Input	System Response	1	extended by: Pagar tempo extra																								
Actor Input	System Response																													
1	extended by: Pagar tempo extra																													

Figura 2.18: Descrição textual Use Case de efectuar pagamento

Author	pimenta	
Date	Dec 14, 2010 4:23:02 PM	
Brief Description		
Preconditions	O Cliente entrou no parque, requisitando um bilhete.	
Post-conditions	Insucesso: Sistema não se altera Sucesso: Bilhete é registado como pago.	
Flow of Events	1 Cliente Introduz bilhete	MPA
	2	Sistema conta tempo de estadia
	3	Sistema calcula valor a pagar
	4	Sistema informa do valor a pagar
	5 Cliente paga quantia	
	6	Sistema processa pagamento
	7	Sistema pergunta se pretende recibo
	8	Sistema devolve bilhete
	9 Cliente recebe bilhete	
	10	Sistema regista pagamento
	11 Cliente recebe recibo	
	12 Cliente recebe bilhete	
	13	Sistema regista pagamento
Alternative 1 6a. Sistema processa dinheiro a mais	1	Actor Input
	2	System Response
	3 Cliente recebe troco	Sistema informa do troco
	4 Cliente recebe troco	Sistema dá troco
	5	
4	Voltar ao passo 7	
Alternative 2 7a - Extension Point: Cliente pede recibo	1	Actor Input
	2	System Response extended by: Emitir recibo
Alternative 3 2a - Extension Point: Excedeu tempo de estadia	1	Actor Input
	1	System Response extended by: Pagar tempo extra

Figura 2.19: Descrição textual Use Case de efectuar pagamento com dinheiro

Super Use Case	Efectuar pagamento.	
Author	pimenta	
Date	Dec 14, 2010 5:31:36 PM	
Brief Description		
Preconditions	Cliente possui cartão magnético e entrou no parque como cliente anónimo.	
Post-conditions	Dinheiro é debitado da conta e bilhete é registado como pago.	
Flow of Events	1 Cliente insere cartão	MPA
	2 <<include>>: Inserir PIN	
	3	Sistema pede para confirmar quantia
	4 Cliente confirma quantia	
	5	Sistema debita dinheiro
	6	Sistema regista operação
	7	Sistema devolve cartão
	8 Cliente recebe cartão	
	Exception 1 3a. Cartão não possui dinheiro suficiente	Actor Input
1		Sistema informa que a conta não possui quantidade de dinheiro suficiente
2		Sistema devolve cartão
3		STOP
Exception 2 4a. Não confirma operação	Actor Input	System Response
1 Cliente não confirma quantia		
2		Sistema devolve cartão
3		STOP

Figura 2.20: Descrição textual Use Case de efectuar pagamento com cartão

<b>Super use case</b>		
<b>Author</b>	pimenta	
<b>Date</b>	Dec 14, 2010 5:39:34 PM	
<b>Brief Description</b>		
<b>Preconditions</b>	Passou do limite de tempo permitido Bilhete marcado com tempo extra	
<b>Post-conditions</b>	Pagou a multa	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 Cliente introduz bilhete	Sistema calcula tempo extra
	2	Sistema calcula pagamento extra
	3	Sistema informa de quantia extra
	4	Sistema pergunta como pretende efectuar pagamento
	5	
	6 Cliente paga com dinheiro	
	7	Sistema processa pagamento
	8	Sistema calcula troco
	9 Cliente recebe troco	
	10	Sistema pergunta se pretende recibo
	11 Cliente pede recibo	Sistema emite recibo
	12	Sistema devolve bilhete
	13	
	14 Cliente recebe bilhete	
	15 Cliente recebe recibo	
	16	Sistema regista pagamento
<b>Alternative 1:</b> 6a. Cliente paga com cartão magnético	<b>Actor Input</b>	<b>System Response</b>
	1	Sistema processa pagamento
	2	Avança para o passo 10
<b>Alternative 2:</b> 11a. Cliente não pede recibo	<b>Actor Input</b>	<b>System Response</b>
	1 Cliente não pede recibo	
	2	Sistema devolve bilhete

Figura 2.21: Descrição textual Use Case de efectuar pagamento extra

<b>Super use case</b>		
<b>Author</b>	pimenta	
<b>Date</b>	Dec 14, 2010 5:39:34 PM	
<b>Brief Description</b>		
<b>Preconditions</b>	Passou do limite de tempo permitido Bilhete marcado com tempo extra	
<b>Post-conditions</b>	Pagou a multa	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 Cliente introduz bilhete	Sistema calcula tempo extra
	2	Sistema calcula pagamento extra
	3	Sistema informa de quantia extra
	4	Sistema pergunta como pretende efectuar pagamento
	5	
	6 Cliente paga com dinheiro	
	7	Sistema processa pagamento
	8	Sistema calcula troco
	9 Cliente recebe troco	
	10	Sistema pergunta se pretende recibo
	11 Cliente pede recibo	Sistema emite recibo
	12	Sistema devolve bilhete
	13	
	14 Cliente recebe bilhete	
	15 Cliente recebe recibo	
	16	Sistema regista pagamento
<b>Alternative 1:</b> 6a. Cliente paga com cartão magnético	<b>Actor Input</b>	<b>System Response</b>
	1	Sistema processa pagamento
	2	Avança para o passo 10
<b>Alternative 2:</b> 11a. Cliente não pede recibo	<b>Actor Input</b>	<b>System Response</b>
	1 Cliente não pede recibo	
	2	Sistema devolve bilhete

Figura 2.22: Descrição textual Use Case de emitir recibo

<b>Super Use Case</b>		
<b>Author</b>	miguel	
<b>Date</b>	19/Dez/2010 0:40:15	
<b>Brief Description</b>		
<b>Preconditions</b>		
<b>Post-conditions</b>	É apresentada a lista de clientes com pagamentos em atraso.	
	<b>Actor Input</b>	<b>System Response</b>
<b>Flow of Events</b>	1 Requer clientes com pagamentos em atraso	
	2	Pede ao sistema central dados dos clientes com pagamentos em atraso.
	3	Lista clientes com pagamentos em atraso.
<b>Alternative 1</b> 3a - Não existem cliente com pagamentos em atraso	<b>Actor Input</b>	<b>System Response</b>
	1	O Sistema apresenta mensagem de que não foram encontrados resultados.

Figura 2.23: Descrição textual Use Case de listar clientes com pagamento em atraso

<b>Super Use Case</b>		
<b>Author</b>	miguel	
<b>Date</b>	19/Dez/2010 0:39:56	
<b>Brief Description</b>		
<b>Preconditions</b>		
<b>Post-conditions</b>	É efectuado o pagamento dos clientes pré-registrados com via verde.	
	<b>Actor Input</b>	<b>System Response</b>
<b>Flow of Events</b>	1 Requer lista de entradas de um cliente.	
	2	Calcula o total que o cliente paga.
	3	Envia o total a ser pago à via verde.
	4	Recebe a percentagem a que tem direito.

Figura 2.24: Descrição textual Use Case de processar pagamentos via verde

<b>Super Use Case</b>		
<b>Author</b>	miguel	
<b>Date</b>	18/Dez/2010 23:57:30	
<b>Brief Description</b>		
<b>Preconditions</b>		
<b>Post-conditions</b>	É gerado um relatório sobre um cliente.	
	<b>Actor Input</b>	<b>System Response</b>
<b>Flow of Events</b>	1 Requer relatório de um determinado cliente.	
	2	Requer dados ao sistema central de informação.
	3	Gera relatório.
	4	Apresenta relatório do respetivo cliente.

Figura 2.25: Descrição textual Use Case de gerar relatórios sobre clientes

<b>Super Use Case</b>		
<b>Author</b>	miguel	
<b>Date</b>	18/Dez/2010 13:53:54	
<b>Brief Description</b>		
<b>Preconditions</b>	Tipo de gestão válida.	
<b>Post-conditions</b>	O Sistema é alterado.	
	<b>Actor Input</b>	<b>System Response</b>
<b>Flow of Events</b>	1 Requer opções de gestão.	Mostra opções de gestão.
	2	
	3 Escolhe tipo de gestão.	Verifica validade da opção.
	4	Gera ficha de cliente.
	5	Informa o sucesso da operação.
	6	
	<b>Actor Input</b>	<b>System Response</b>
<b>Alternative 1</b> 4a - se opção inválida	1	Informa do erro.
	2	voltar ao passo 1

Figura 2.26: Descrição textual Use Case de gerir fichas de clientes

<b>Super Use Case</b>	Gerir fichas de cliente.	
<b>Author</b>	miguel	
<b>Date</b>	18/Dez/2010 13:34:56	
<b>Brief Description</b>		
<b>Preconditions</b>		
<b>Post-conditions</b>	Ficha de cliente criada no sistema.	
	<b>Actor Input</b>	<b>System Response</b>
<b>Flow of Events</b>	1 Envia os dados.	Validação dos dados.
	2	Cria uma nova ficha de cliente.
	3	Informa sucesso da operação.
	4	
	<b>Actor Input</b>	<b>System Response</b>
<b>Exception 1</b> 2a - dados inválidos.	1	Informa que os dados estão inválidos.
	2	STOP

Figura 2.27: Descrição textual Use Case de criar fichas de clientes

<b>Super Use Case</b>	Gerir fichas de cliente.	
<b>Author</b>	miguel	
<b>Date</b>	18/Dez/2010 13:44:28	
<b>Brief Description</b>		
<b>Preconditions</b>	Ficha existe.	
<b>Post-conditions</b>	Ficha editada com sucesso.	
	<b>Actor Input</b>	<b>System Response</b>
<b>Flow of Events</b>	1 Requer lista de fichas.	Mostra lista de fichas.
	2	
	3 Escolhe ficha a editar.	Verifica se ficha valida.
	4	Disponibiliza ficha.
	5	
	6 Edita ficha.	
	7 Envia novos dados.	Verifica dados enviados.
	8	Altera ficha.
	9	Informa sucesso de edição da ficha.
	10	
	<b>Actor Input</b>	<b>System Response</b>
<b>Exception 1</b> 8a - dados inválidos.	1	Informa do erro.
	2	STOP

Figura 2.28: Descrição textual Use Case editar de fichas clientes

<b>Super Use Case</b>	Gerir fichas de cliente.	
<b>Author</b>	miguel	
<b>Date</b>	18/Dez/2010 13:51:37	
<b>Brief Description</b>		
<b>Preconditions</b>	Ficha de cliente existe.	
<b>Post-conditions</b>	Ficha de cliente é removida do sistema.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 Requer fichas.	
	2	Mostra lista de fichas.
	3 Escolhe ficha a remover	
	4	Verifica validade da ficha.
	5	Remove ficha de cliente.
	6	Informa ficha removida com sucesso.
<b>Exception 1</b> 4a - ficha inválida	<b>Actor Input</b>	<b>System Response</b>
	1	Mostra mensagem de erro.
	2	STOP

Figura 2.29: Descrição textual Use Case de remover fichas de clientes

<b>Super Use Case</b>		
<b>Author</b>	Milton	
<b>Date</b>	18/Dez/2010 15:48:48	
<b>Brief Description</b>	Gerar tabela de distribuições de lugares por níveis e por horas.	
<b>Preconditions</b>		
<b>Post-conditions</b>	Gera tabela de distribuições de lugares por níveis e por horas.	
<b>Flow of Events</b>	<b>Actor Input</b>	<b>System Response</b>
	1 Funcionário solicita criação da tabela de distribuição de determinado dia.	
	2	Sistema requisita dados de ocupação de lugares ao sistema central de informação.
	3	Sistema recebe dados do RegOcupHoraPiso.
	4	Sistema gera tabela de distribuição.
	5	Sistema imprime tabela.
<b>Exception 1</b> 2a - Sistema não recebe dados.	<b>Actor Input</b>	<b>System Response</b>
	1	Sistema informa que sistema central informação não enviou dados.
	2	STOP

Figura 2.30: Descrição textual Use Case tartar estatisticas

<b>Super Use Case</b>	Elaborar relatórios	
<b>Author</b>	Milton	
<b>Date</b>	18/Dez/2010 16:39:51	
<b>Brief Description</b>	Elaborar relatórios de máquinas automáticas de pagamentos	
<b>Preconditions</b>		
<b>Post-conditions</b>	Relatórios são elaborados.	
<b>Flow of Events</b>	Actor Input	System Response
	1 Funcionário requisita elaboração de relatório de determinada máquina.	
	2	Sistema requisita dados de máquinas de pagamentos ao sistema central de informação.
	3	Sistema recebe dados de MagPagamento.
	4	Sistema gera relatório.
	5	Sistema imprime relatório.
<b>Exception:</b> 2a - Sistema não recebe dados.	Actor Input	System Response
	1	Sistema informa que sistema central de informação não enviou dados.
	2	STOP

Figura 2.31: Descrição textual Use Case de elaborar relatórios de maquinas de pagamento

<b>Super Use Case</b>		
<b>Author</b>	Milton	
<b>Date</b>	18/Dez/2010 15:22:22	
<b>Brief Description</b>	Elaboração de relatórios	
<b>Preconditions</b>	Impressora está operacional	
<b>Post-conditions</b>	Relatório elaborado	
<b>Flow of Events</b>	Actor Input	System Response
	1 Funcionário solicita elaboração de relatório.	
	2	Sistema gera relatório.
	3	Sistema imprime relatório.

Figura 2.32: Descrição textual Use Case de elaborar relatórios

<b>Super Use Case</b>	Elaborar relatórios	
<b>Author</b>	Milton	
<b>Date</b>	18/Dez/2010 16:32:07	
<b>Brief Description</b>	Elaborar relatórios diários de entrada e saída.	
<b>Preconditions</b>		
<b>Post-conditions</b>	Elabora relatório.	
<b>Flow of Events</b>	Actor Input	System Response
	1 Funcionário requisita elaboração do relatório de entrada e saída diário.	
	2	Sistema requisita dados ao sistema central de informação.
	3	Sistema recebe dados de RegDiarioPiso.
	4	Sistema recebe dados de RegMovimento.
	5	Sistema recebe dados de Bilhete.
	6	Sistema gera o relatório.
	7	Sistema imprime relatório.
<b>Exception:</b> 2a - Sistema não consegue os dados.	Actor Input	System Response
	1	Sistema informa que não recebeu dados do sistema central de informação.
	2	STOP.

Figura 2.33: Descrição textual Use Case de elaborar relatórios de entradas e saídas

<b>Super Use Case</b>		
<b>Author</b>	Milton Nunes	
<b>Date</b>	18/Dez/2010 15:01:08	
<b>Brief Description</b>	Processamento da facturação do Parkum	
<b>Preconditions</b>	Impressora está operacional	
<b>Post-conditions</b>	Sistema processa facturação.	
<b>Flow of Events</b>	Actor Input	System Response
	1 Funcionário solicita processamento da facturação.	
	2	Sistema gera facturação.
	3	Sistema imprime facturação.

Figura 2.34: Descrição textual Use Case de processar facturação

<b>Super Use Case</b>	Procesar facturação	
<b>Author</b>	Milton	
<b>Date</b>	18/Dez/2010 15:29:32	
<b>Brief Description</b>	Elabora facturação em dinheiro.	
<b>Preconditions</b>		
<b>Post-conditions</b>	Facturação criada.	
<b>Flow of Events</b>	Actor Input	System Response
	1 Funcionário solicita processamento facturação de pagamentos em dinheiro.	
	2	Sistema requisita dados dos pagamentos em dinheiro ao sistema central de informação.
	3	Sistema recebe dados de Recibo.
	4	Sistema gera facturação de pagamentos em dinheiro.
	5	Sistema imprime facturação.
<b>Exception:</b> 2a - Sistema não recebe dados.	Actor Input	System Response
	1	Sistema informa que não recebeu dados.
	2	STOP.

Figura 2.35: Descrição textual Use Case de processar facturação em dinheiro

<b>Super Use Case</b>	Procesar facturação	
<b>Author</b>	Milton	
<b>Date</b>	18/Dez/2010 15:35:42	
<b>Brief Description</b>	Gerar facturação de pagamentos efectuados através de via-verde.	
<b>Preconditions</b>		
<b>Post-conditions</b>	Facturação gerada.	
<b>Flow of Events</b>	Actor Input	System Response
	1 Funcionário solicita processamento facturação de pagamentos através de via-verde.	
	2	Sistema requisita ao sistema central informação dados de pagamentos efectuados por clientes de via verde.
	3	Sistema gera facturação.
	4	Sistema imprime facturação.
<b>Exception:</b> 2a - Via-verde não envia dados de pagamentos	Actor Input	System Response
	1	Sistema informa que via-verde não enviou dados.
	2	STOP

Figura 2.36: Descrição textual Use Case de processar facturação resultante de via verde

<b>Super Use Case</b>	Procesar facturação																
<b>Author</b>	Milton																
<b>Date</b>	18/Dez/2010 15:40:03																
<b>Brief Description</b>																	
<b>Preconditions</b>																	
<b>Post-conditions</b>																	
<b>Flow of Events</b>	<table border="1"> <thead> <tr> <th></th> <th><b>Actor Input</b></th> <th><b>System Response</b></th> </tr> </thead> <tbody> <tr> <td>1</td><td>Funcionário solicita processamento facturação de pagamentos de clientes pré-registrados.</td><td></td></tr> <tr> <td>2</td><td></td><td>Sistema requisita ao sistema central informação dados dos pagamentos efectuados por débito directo.</td></tr> <tr> <td>3</td><td></td><td>Sistema gera facturação.</td></tr> <tr> <td>4</td><td></td><td>Sistema imprime facturação.</td></tr> </tbody> </table>		<b>Actor Input</b>	<b>System Response</b>	1	Funcionário solicita processamento facturação de pagamentos de clientes pré-registrados.		2		Sistema requisita ao sistema central informação dados dos pagamentos efectuados por débito directo.	3		Sistema gera facturação.	4		Sistema imprime facturação.	
	<b>Actor Input</b>	<b>System Response</b>															
1	Funcionário solicita processamento facturação de pagamentos de clientes pré-registrados.																
2		Sistema requisita ao sistema central informação dados dos pagamentos efectuados por débito directo.															
3		Sistema gera facturação.															
4		Sistema imprime facturação.															
<b>Exception:</b> 2a - Sistema bancário não responde.	<table border="1"> <thead> <tr> <th></th> <th><b>Actor Input</b></th> <th><b>System Response</b></th> </tr> </thead> <tbody> <tr> <td>1</td><td></td><td>Sistema informa que sistema bancário não enviou dados.</td></tr> <tr> <td>2</td><td></td><td>STOP</td></tr> </tbody> </table>		<b>Actor Input</b>	<b>System Response</b>	1		Sistema informa que sistema bancário não enviou dados.	2		STOP							
	<b>Actor Input</b>	<b>System Response</b>															
1		Sistema informa que sistema bancário não enviou dados.															
2		STOP															

Figura 2.37: Descrição textual Use Case de processar facturação de débito de cartão

### 2.1.4 Diagramas de Sequencia

Com os diagramas de use case textuais podemos passar a especificação "dinâmica", ou seja ao comportamento do sistema.

Começamos então a fazer a especificação através dos diagramas textuais dos use cases descritos até agora, criando os diagramas de sequencia.

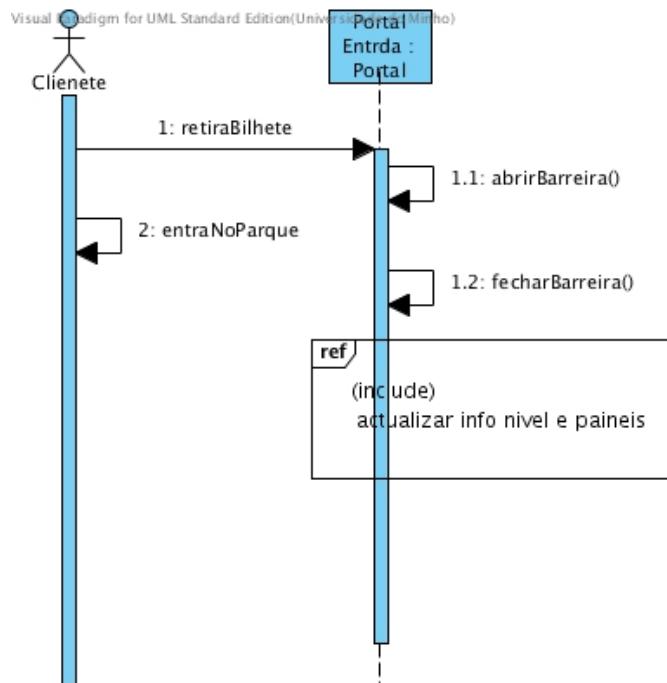
Estes diagramas oferecem uma representação gráfica da interação entre os diferentes objectos.

### 2.1.5 Refinamento

Após a criação dos diagrmas de sequencia do sistema, passamos ao refinamento destes.

Com o refinamento destes será possível obter os varios sub-sistemas em que se divide o sistema principal.

Apresentamos a seguir o refinamento de alguns diagramas de sequencia:



**Figura 2.38:** Diagrama de sequencia de Entrada versão 1.0

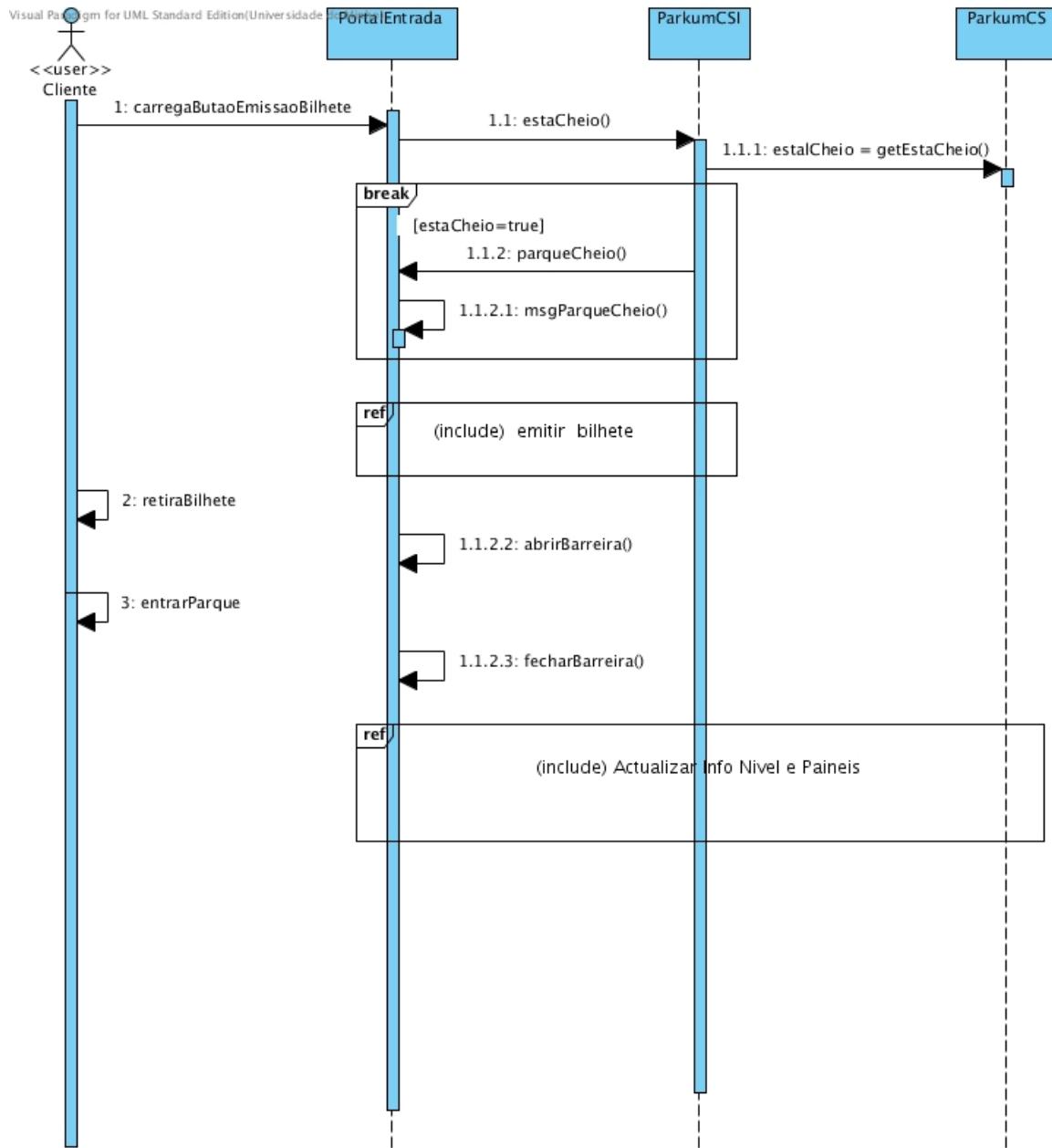


Figura 2.39: Diagrama de sequencia de Entrada versão 2.0

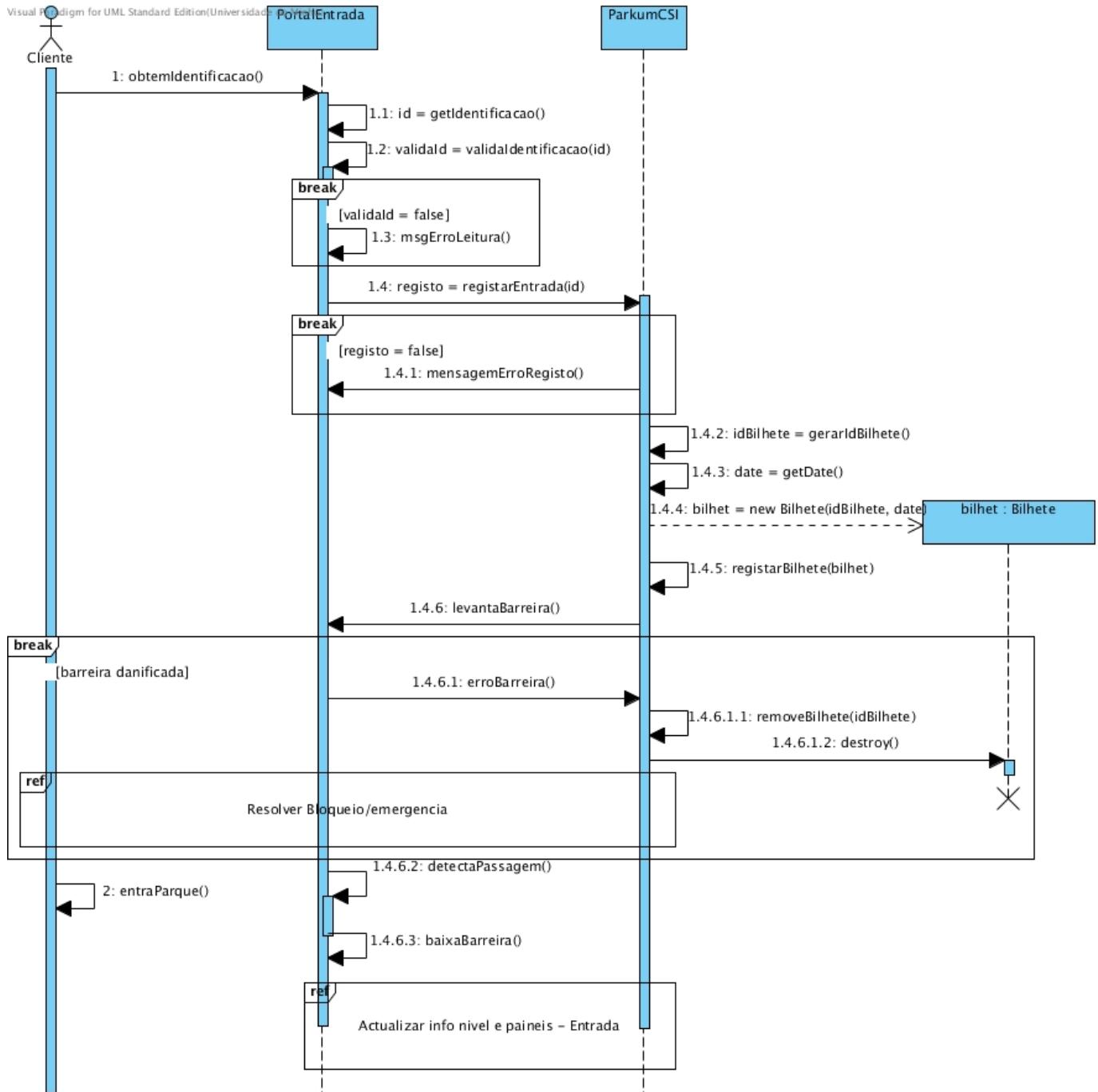
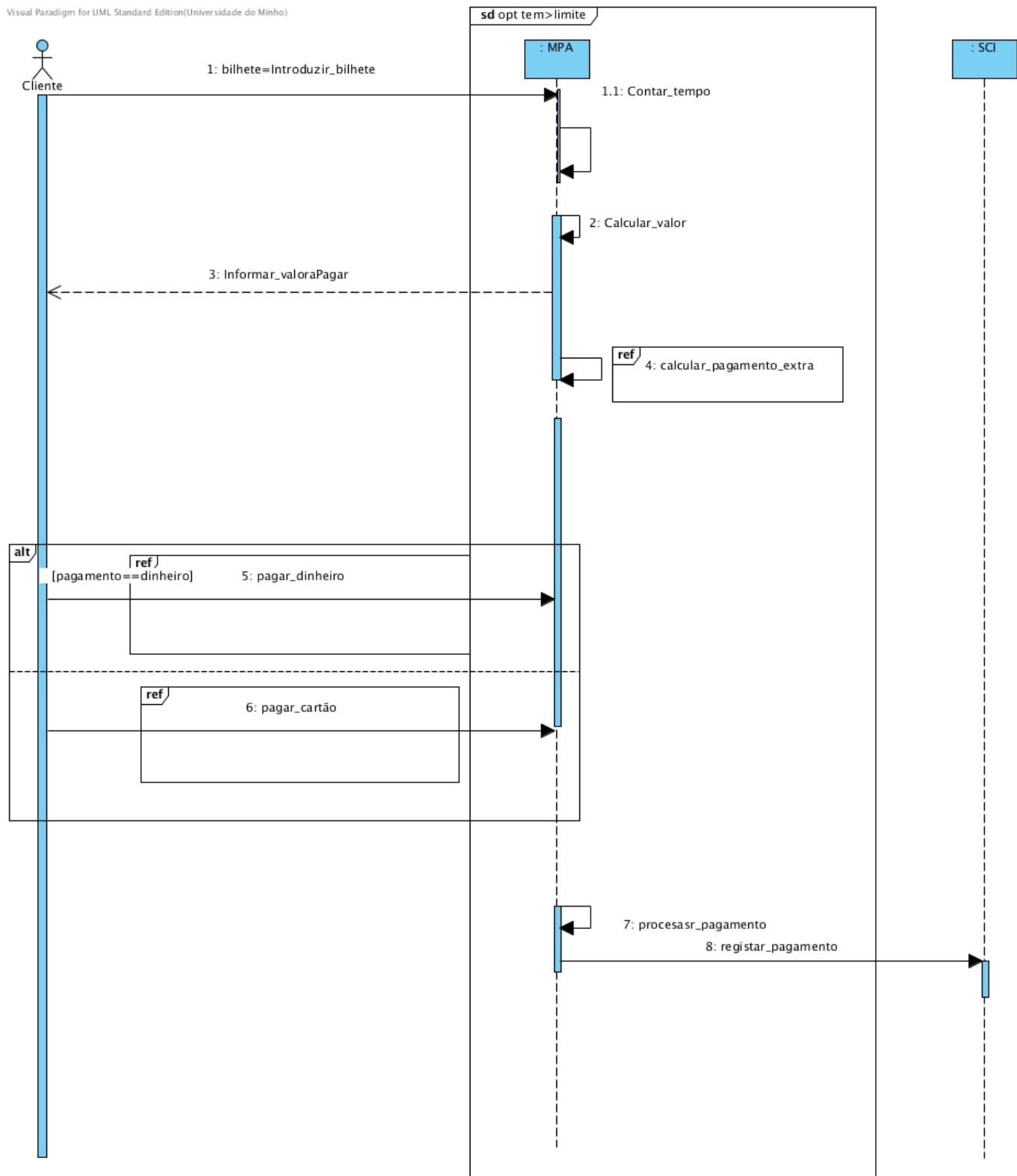
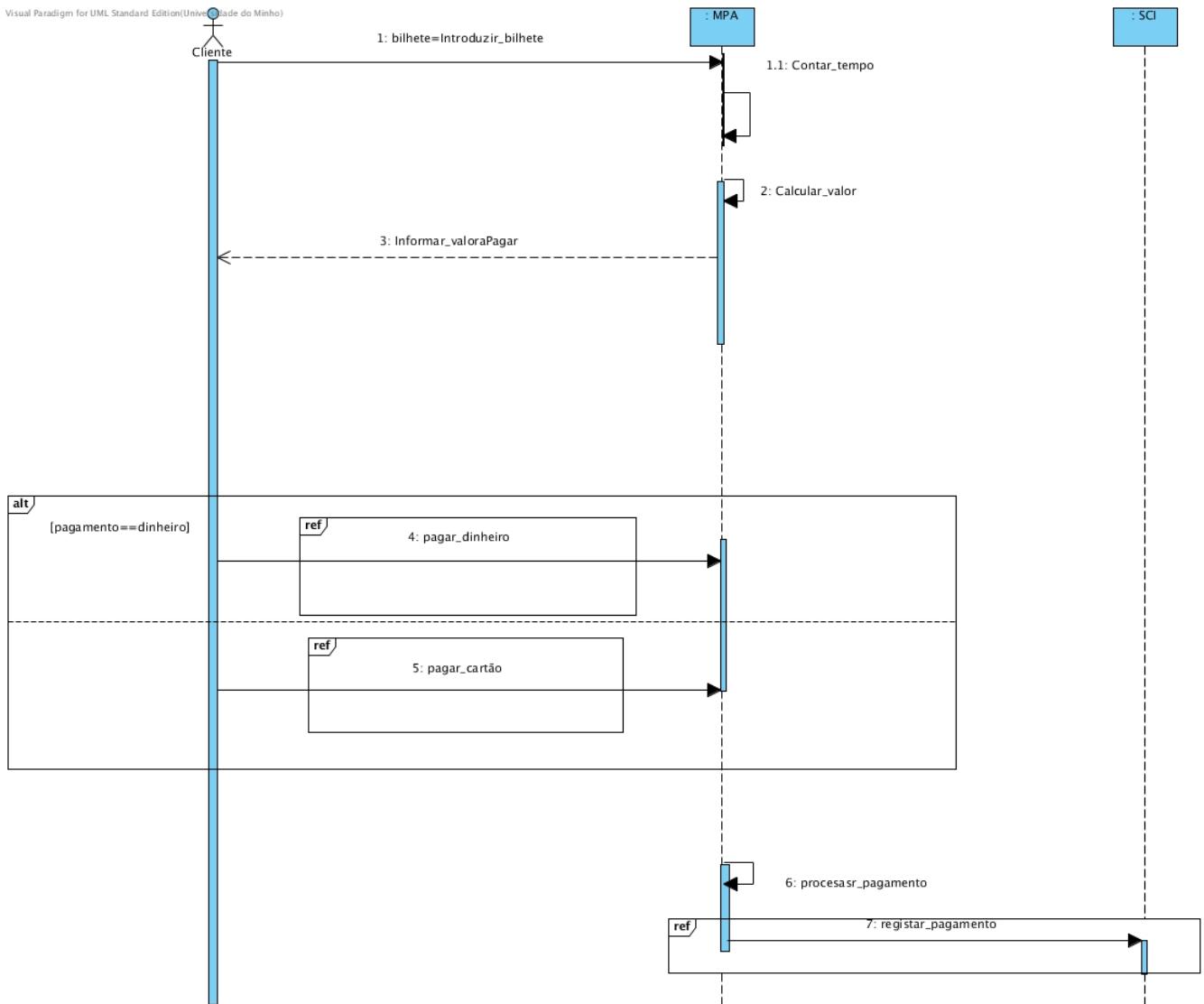


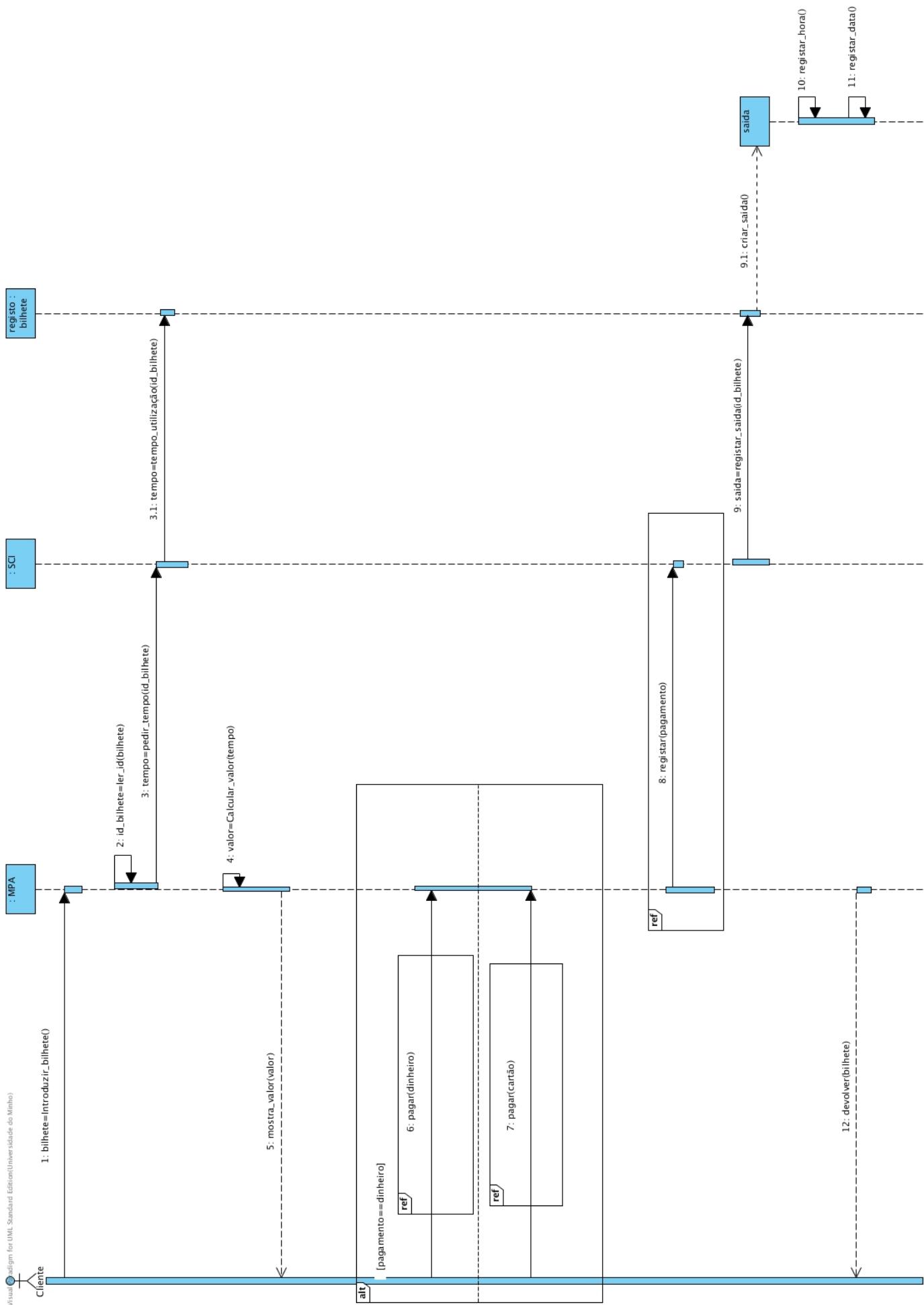
Figura 2.40: Diagrama de sequencia de Entrada versão 3.0

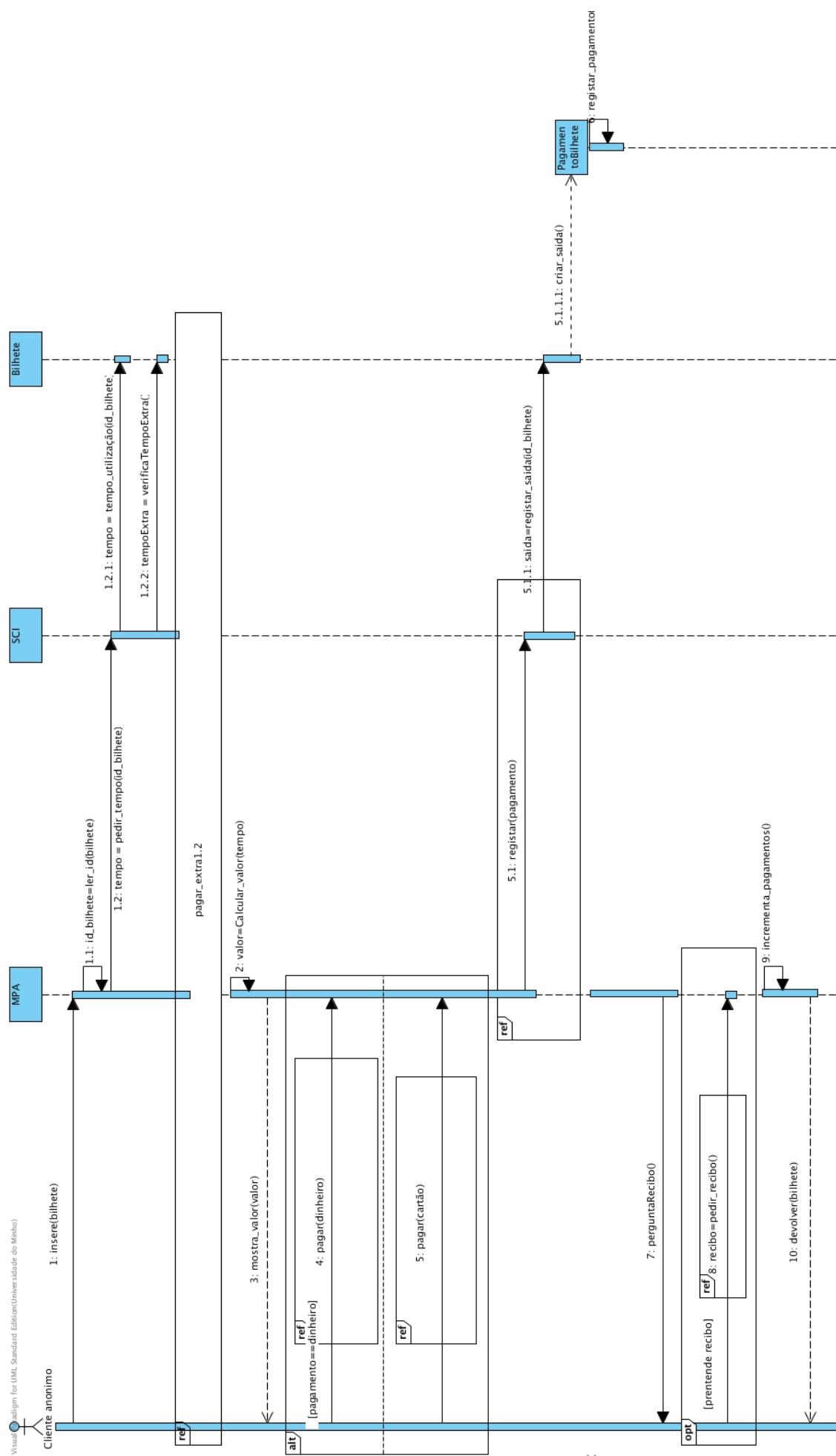


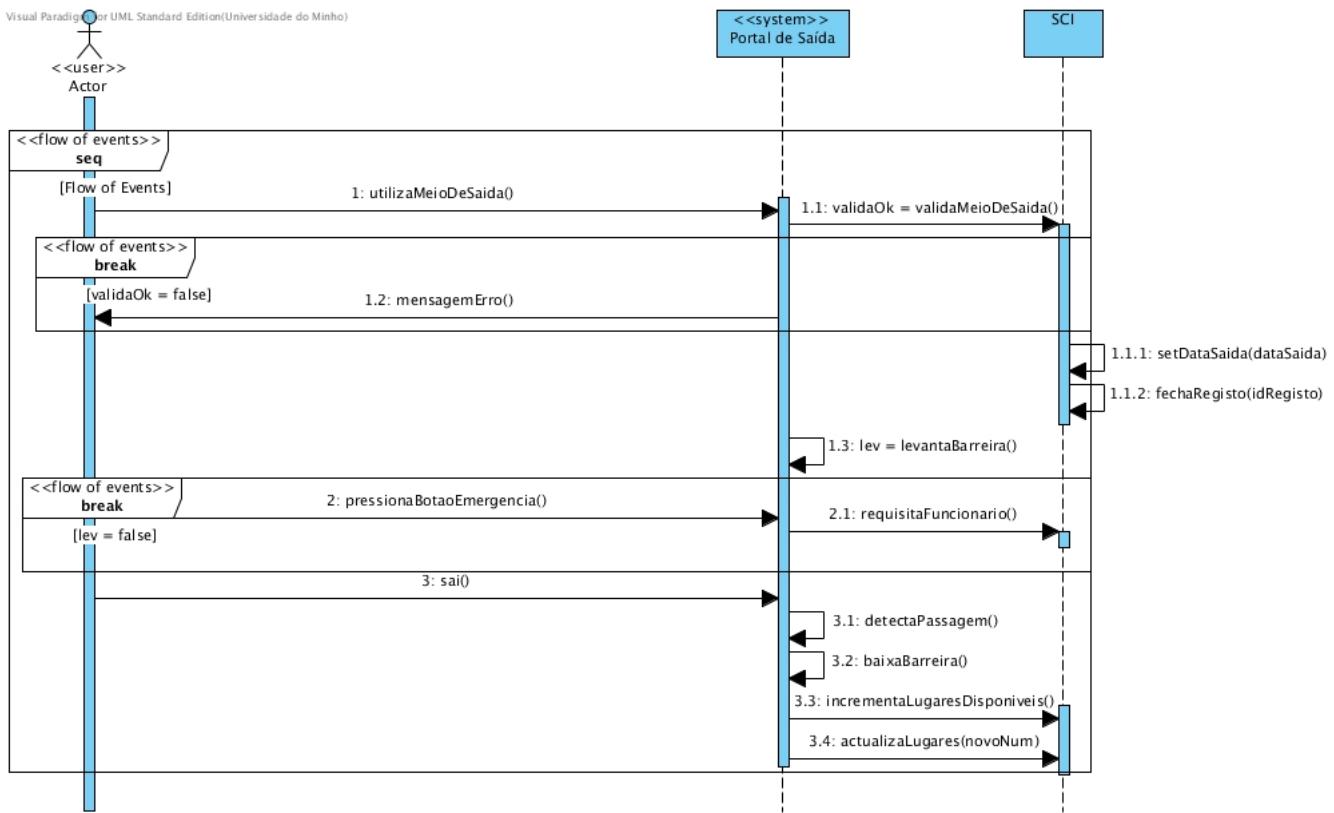
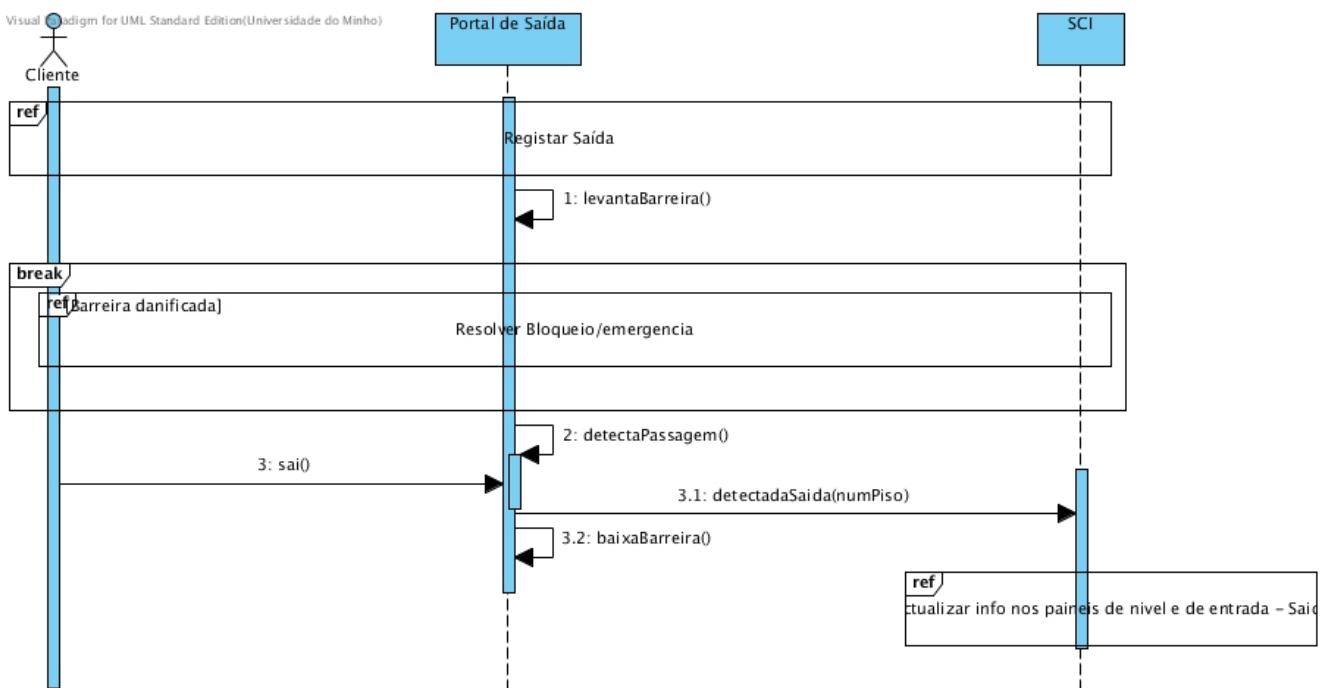
**Figura 2.41:** Diagrama de sequencia de Efectuar pagamento versão 1.0



**Figura 2.42:** Diagrama de sequencia de Efectuar pagamento versão 2.0

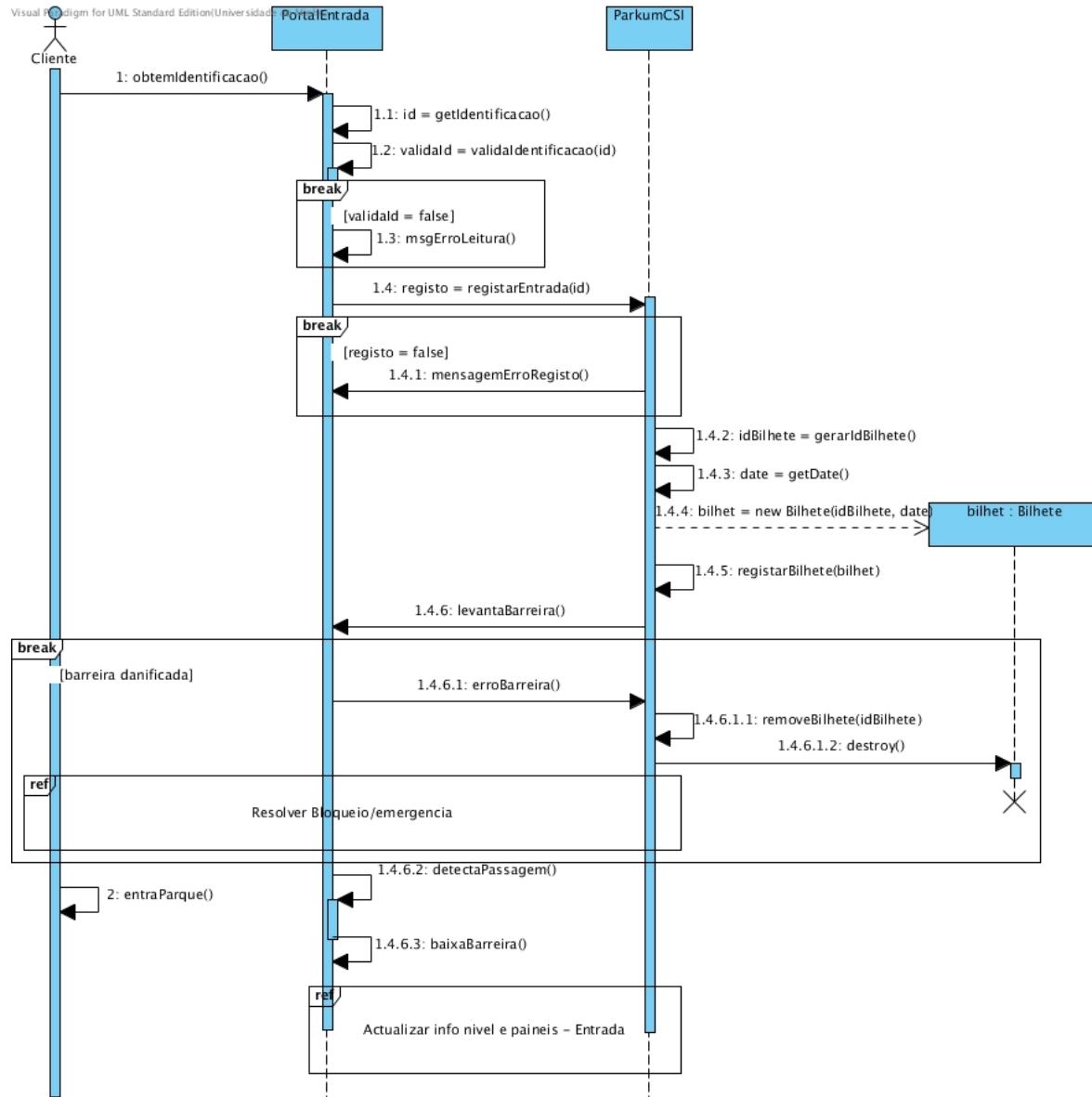
**Figura 2.43:** Diagrama de sequencia de Efectuar pagamento versão 3.0

**Figura 2.44:** Diagrama de sequencia de Efectuar pagamento versão 4.0

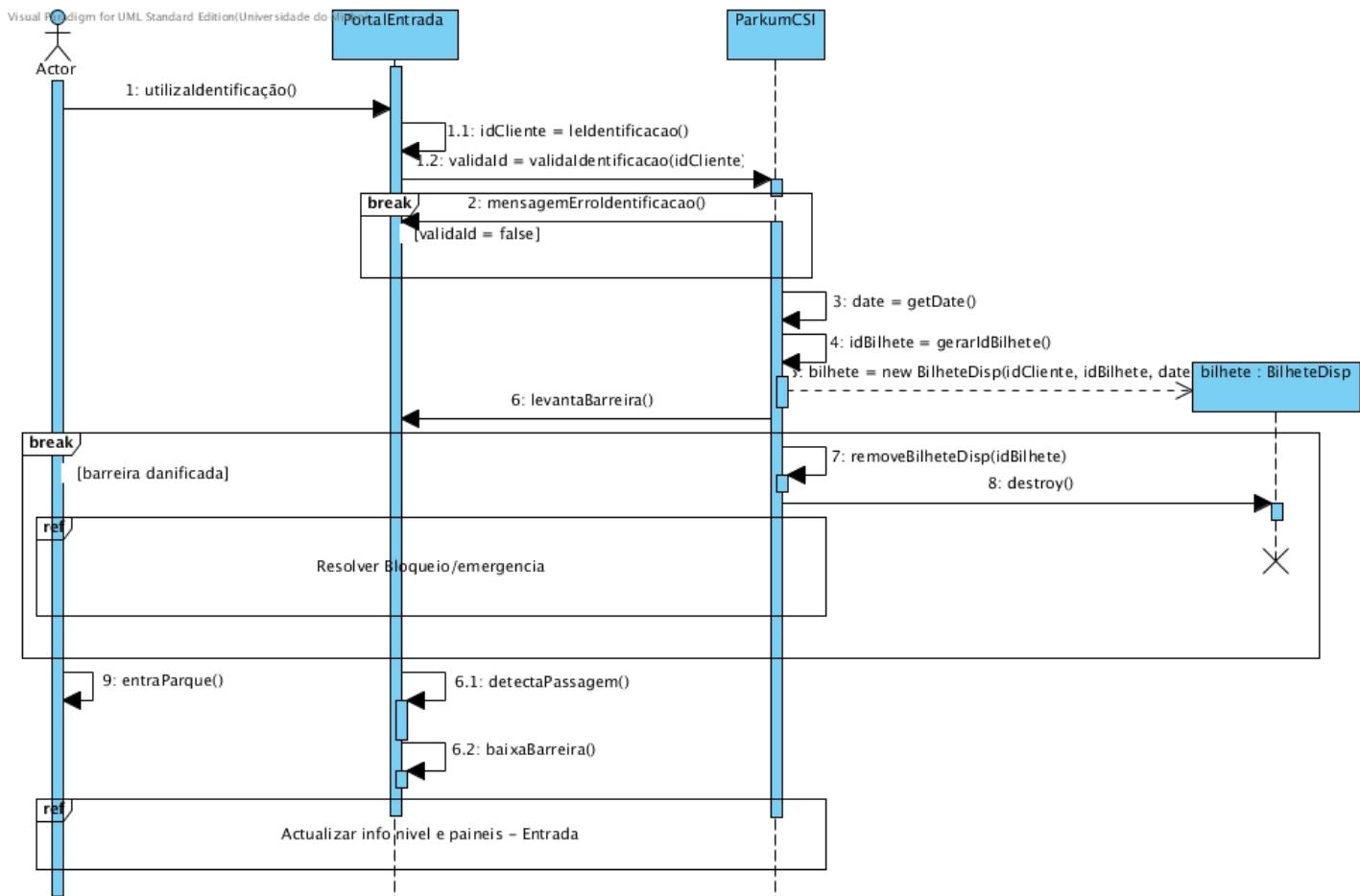
**Figura 2.45:** Diagrama de sequencia de Saida versão 1.0**Figura 2.46:** Diagrama de sequencia de Saida versão 2.0

### 2.1.6 Diagramas de sequencia finais

Após o refinamento dos diagramas de sequência que foram criados através dos use case textuais, ficamos com os diagramas de sequência que apresentamos a seguir, onde já temos uma clara noção de todas as entidades envolvidas em cada um e das suas necessidades.



**Figura 2.47:** Diagrama de sequencia de Entrada



**Figura 2.48:** Diagrama de sequencia de Entrar no parque pre-registrado

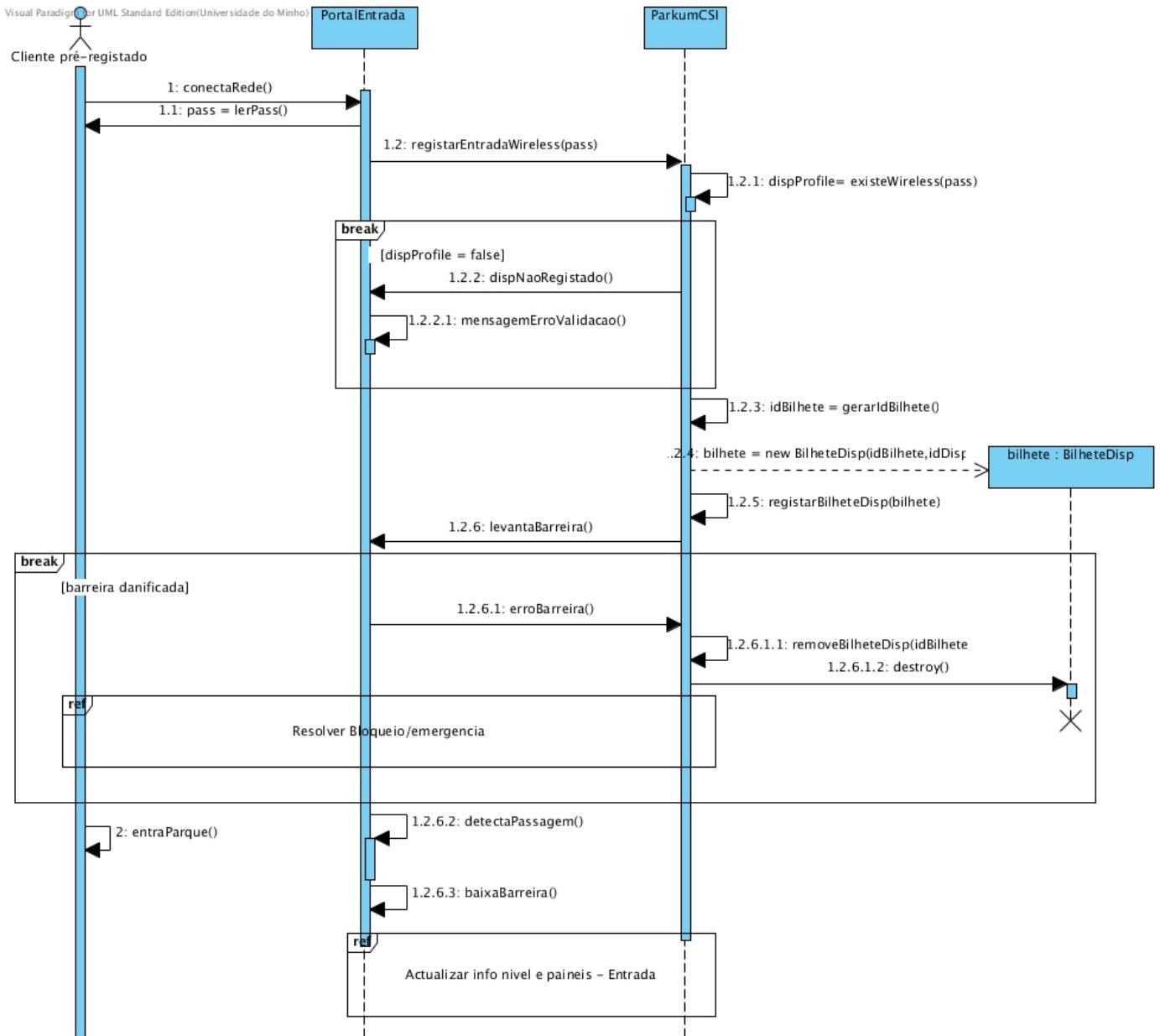
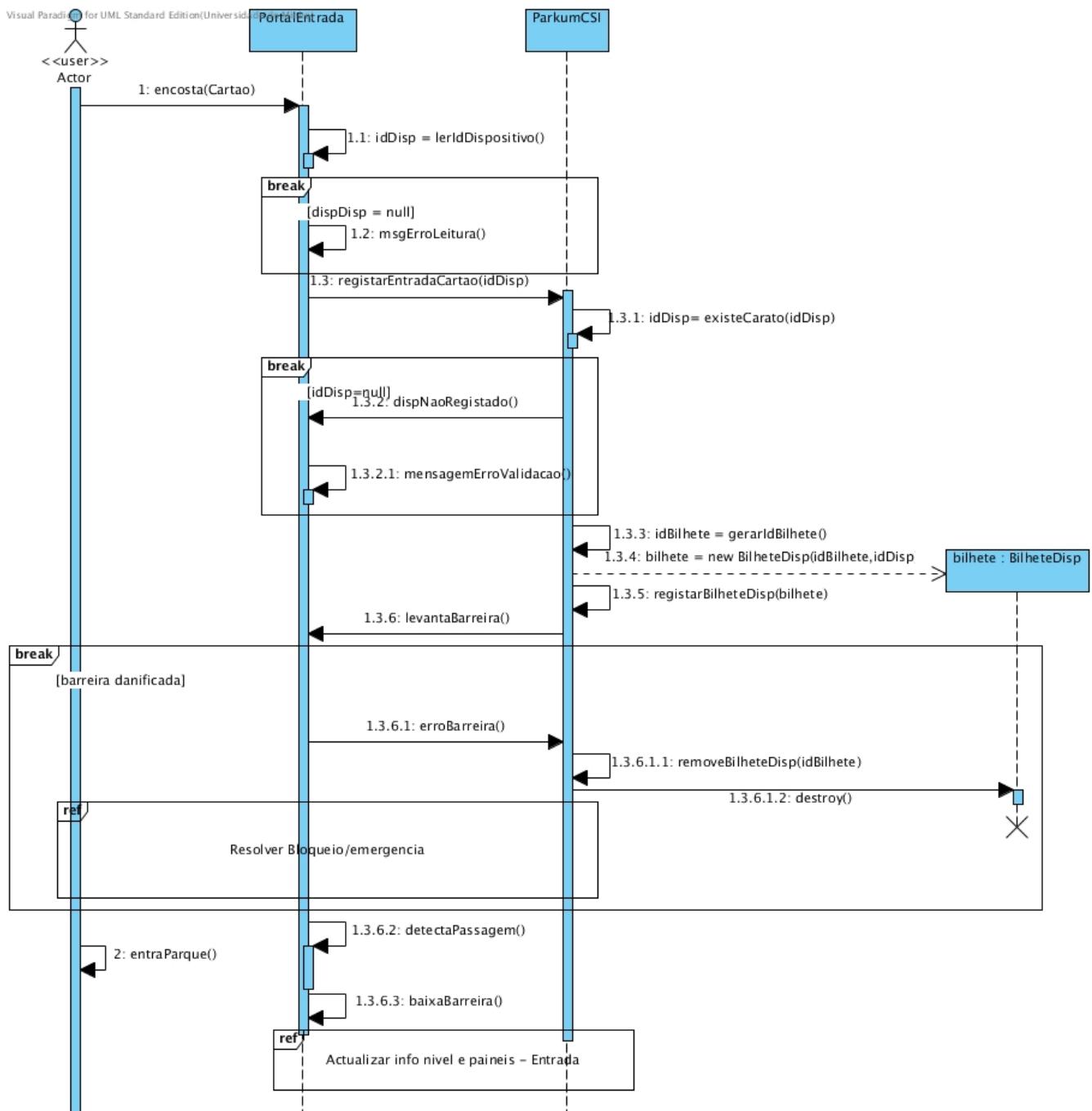
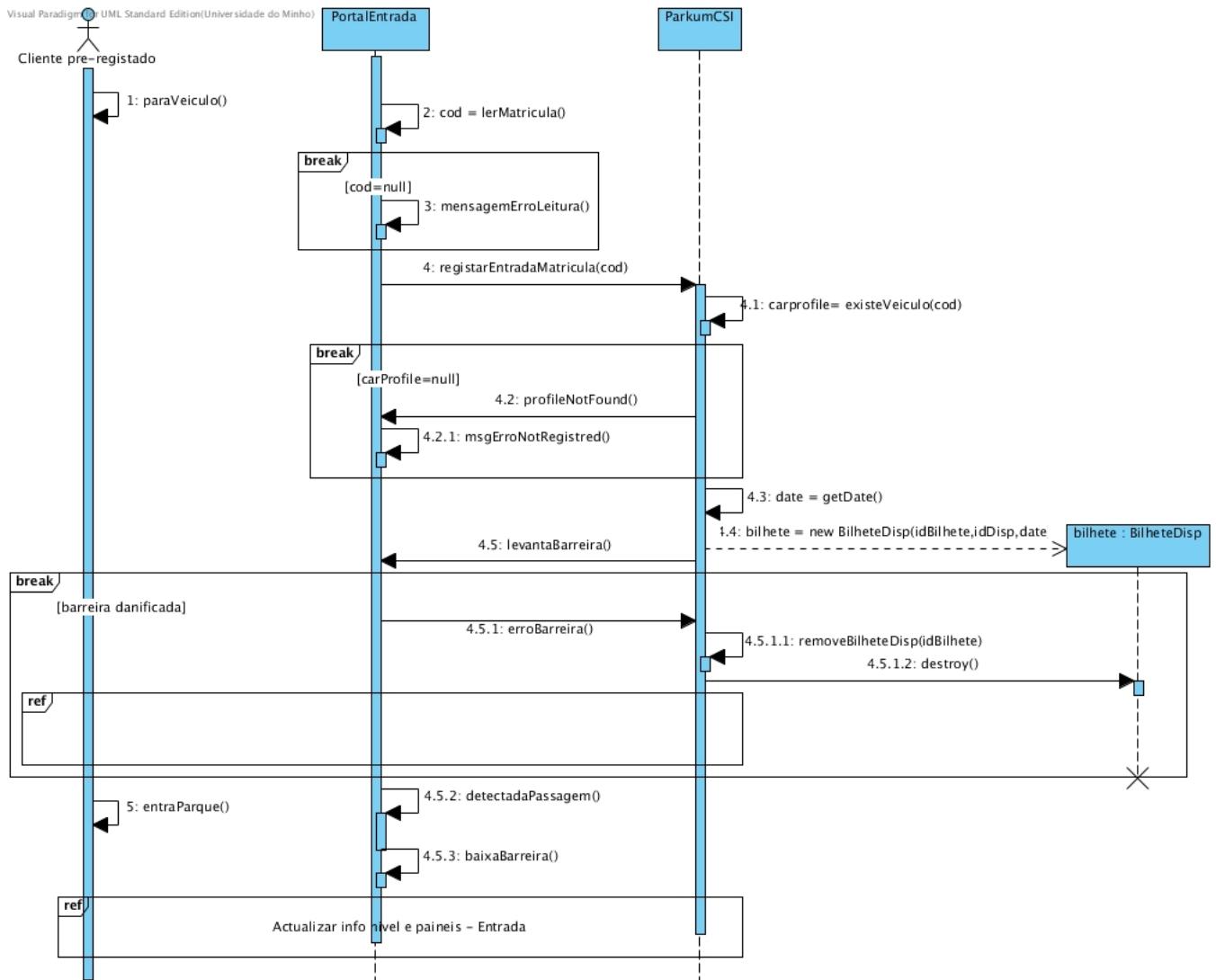


Figura 2.49: Diagrama de sequencia de Entrar chamada wireless



**Figura 2.50:** Diagrama de sequencia de Entrar cartão de cliente



**Figura 2.51:** Diagrama de sequencia de Entrar matricula pre-registada

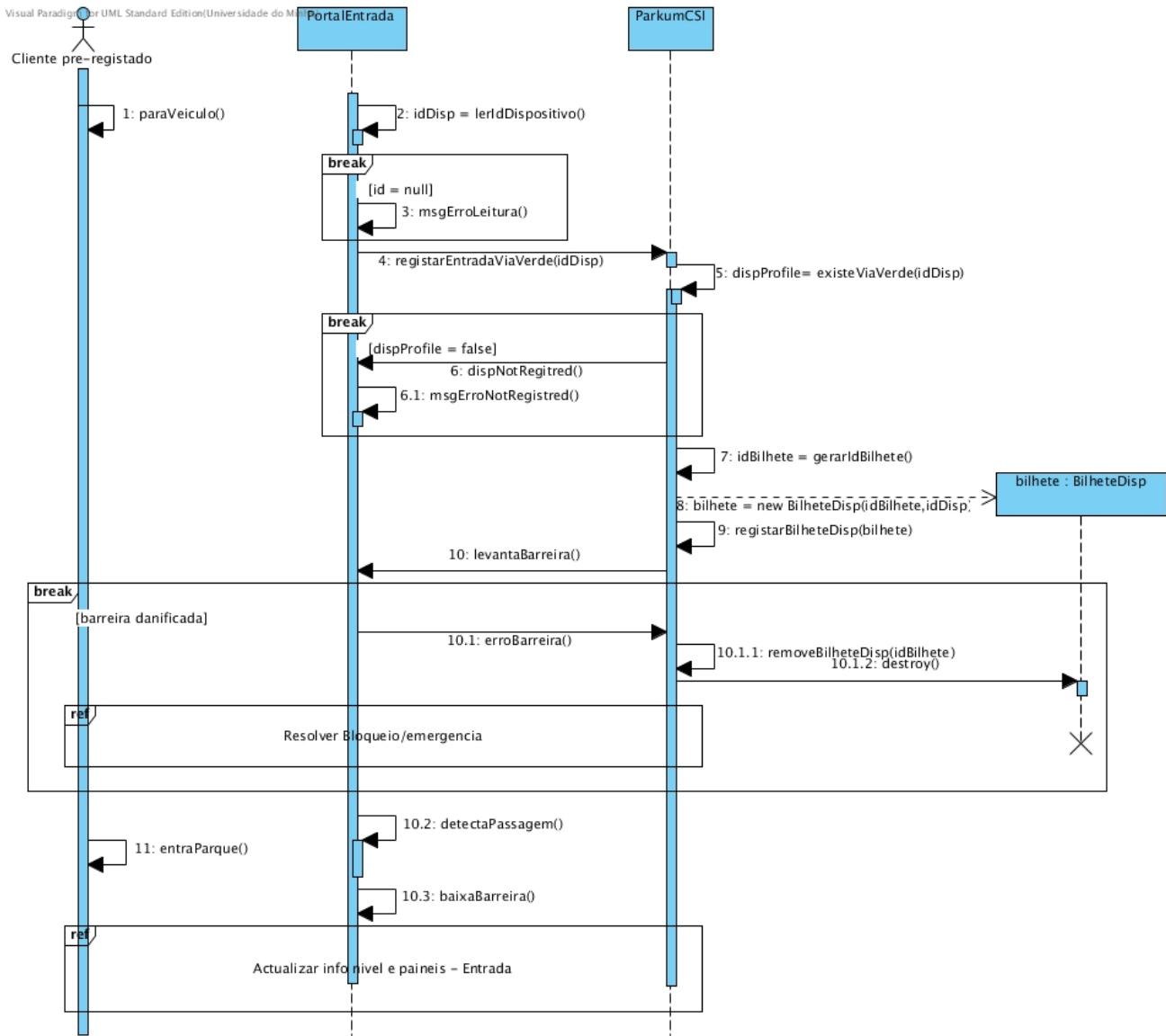


Figura 2.52: Diagrama de sequencia de Entrar via-verde

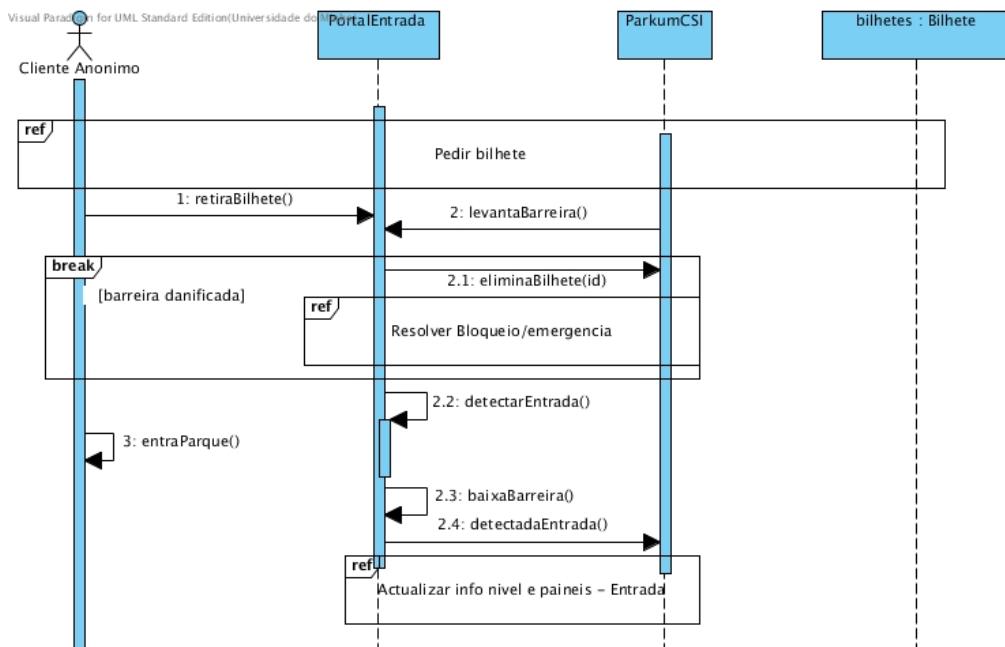


Figura 2.53: Diagrama de sequencia de Entrar no parque anônima

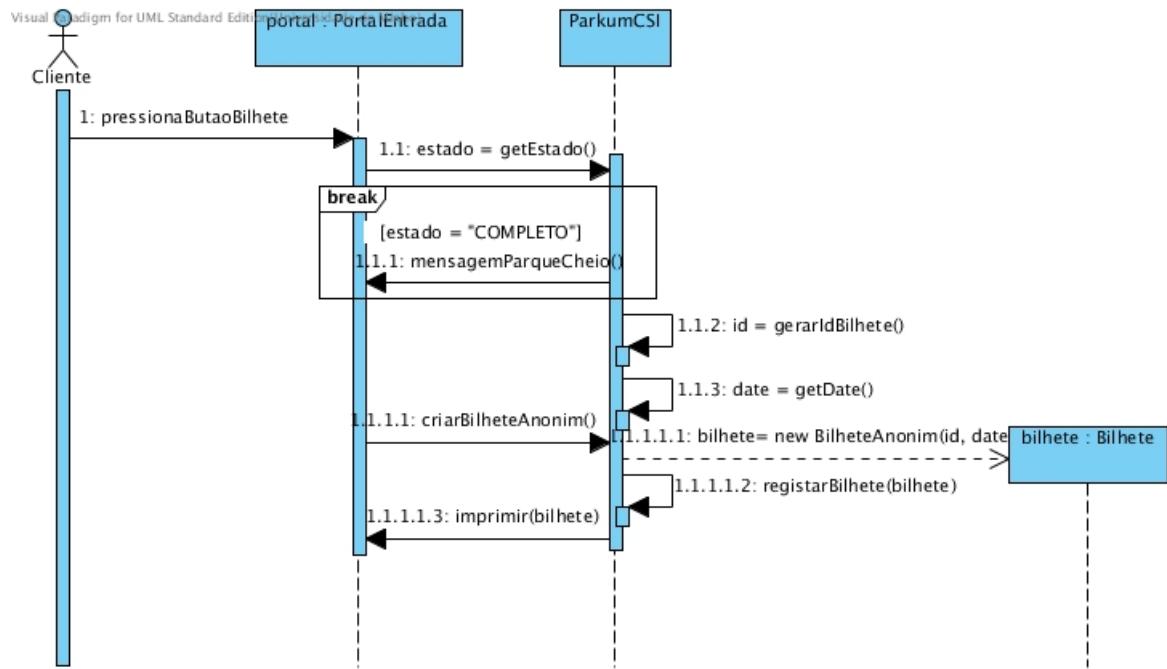


Figura 2.54: Diagrama de sequencia de Pedir bilhete

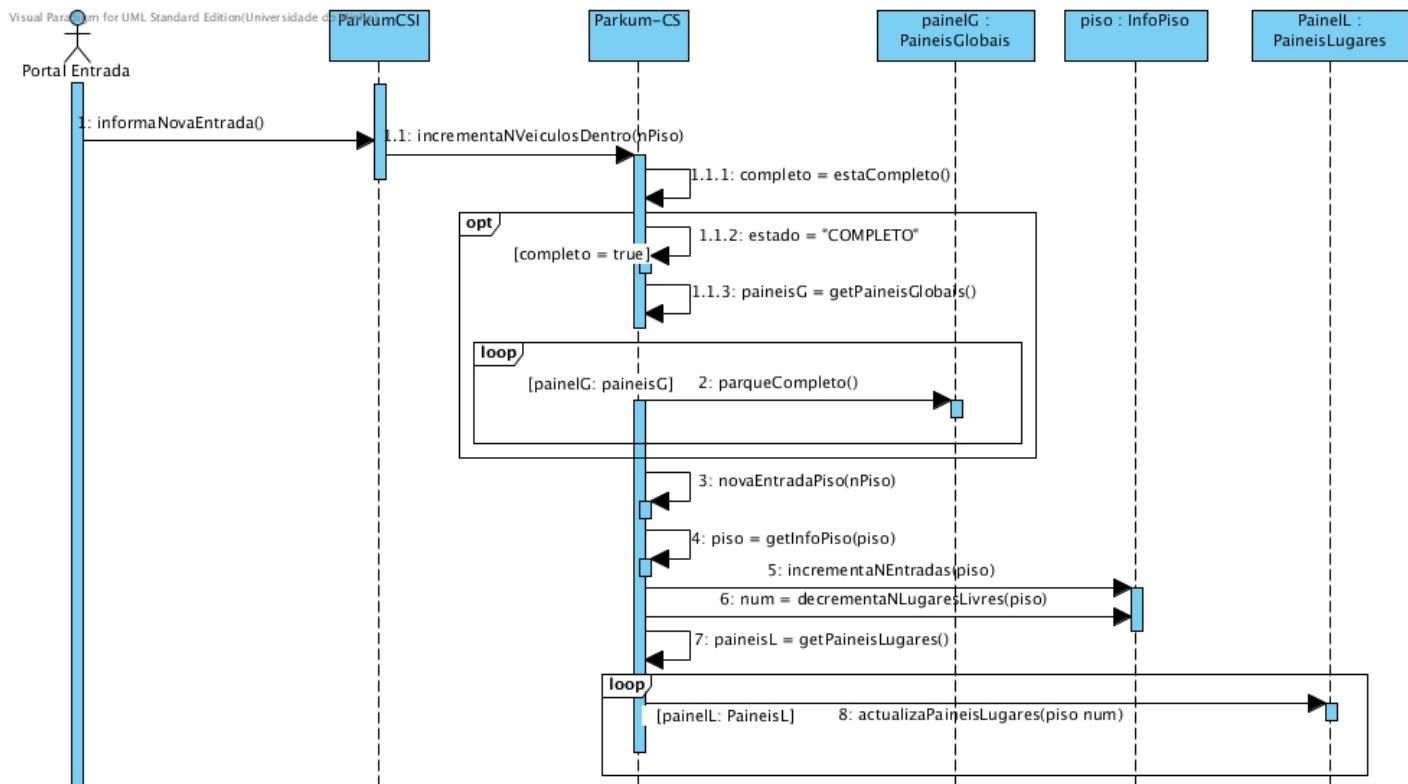


Figura 2.55: Diagrama de sequencia de Actualizar info nível e painéis - Entrada

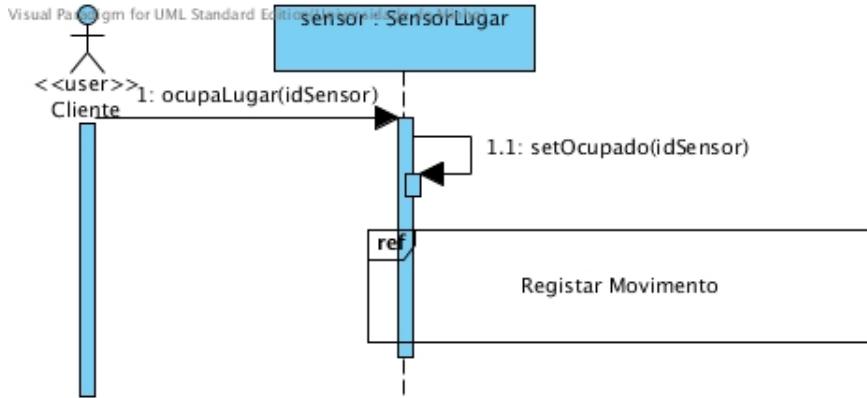


Figura 2.56: Diagrama de sequencia de Estacionar

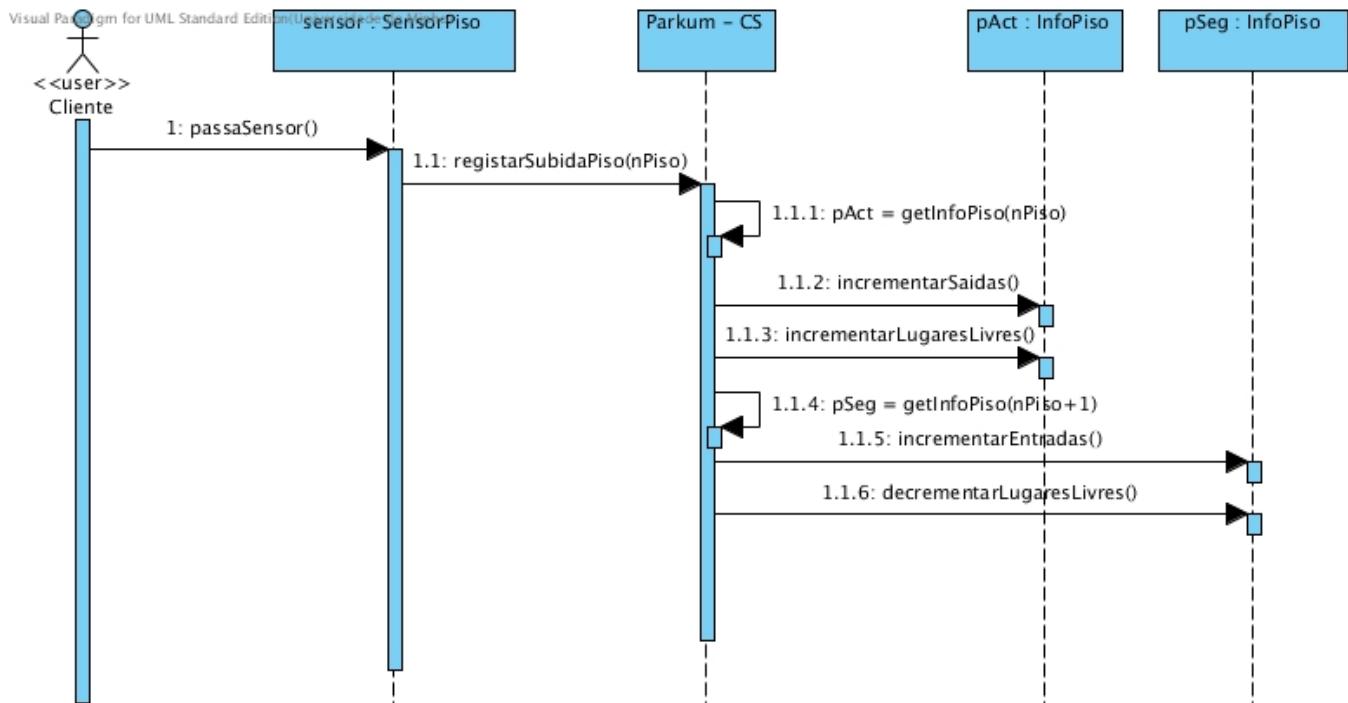


Figura 2.57: Diagrama de sequencia de Subir piso

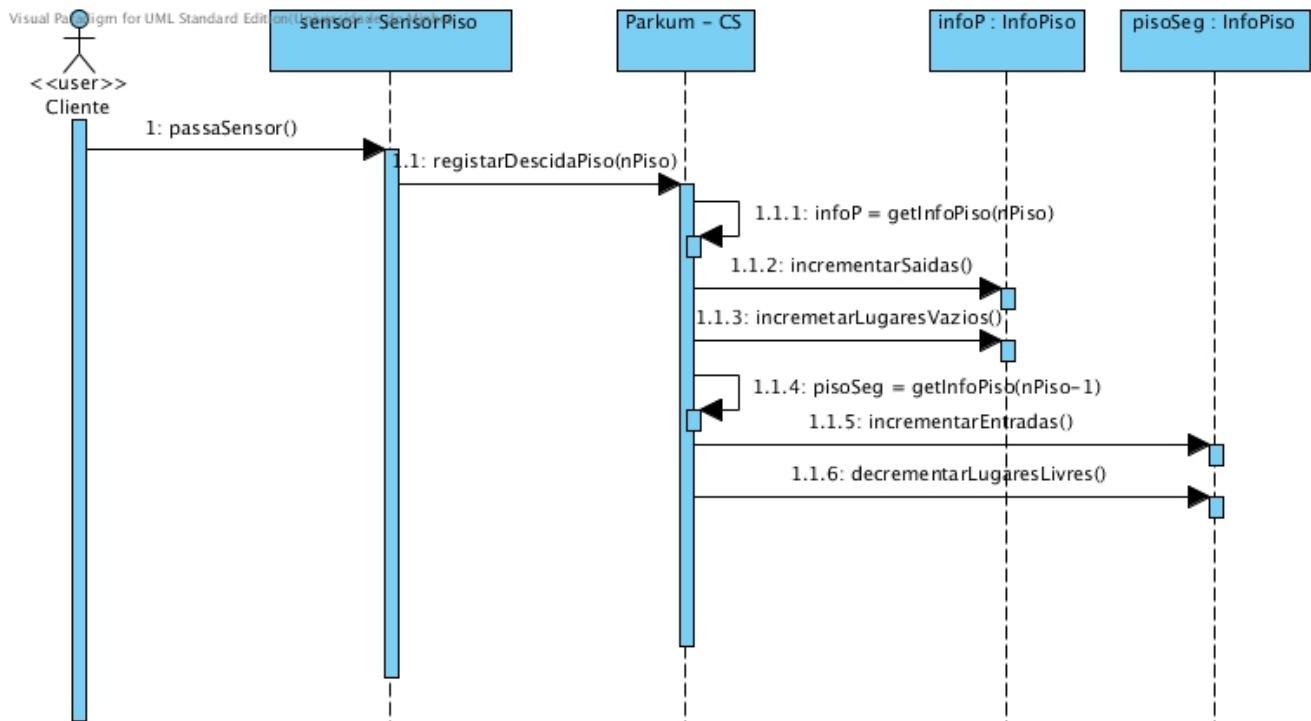


Figura 2.58: Diagrama de sequencia de Descer piso

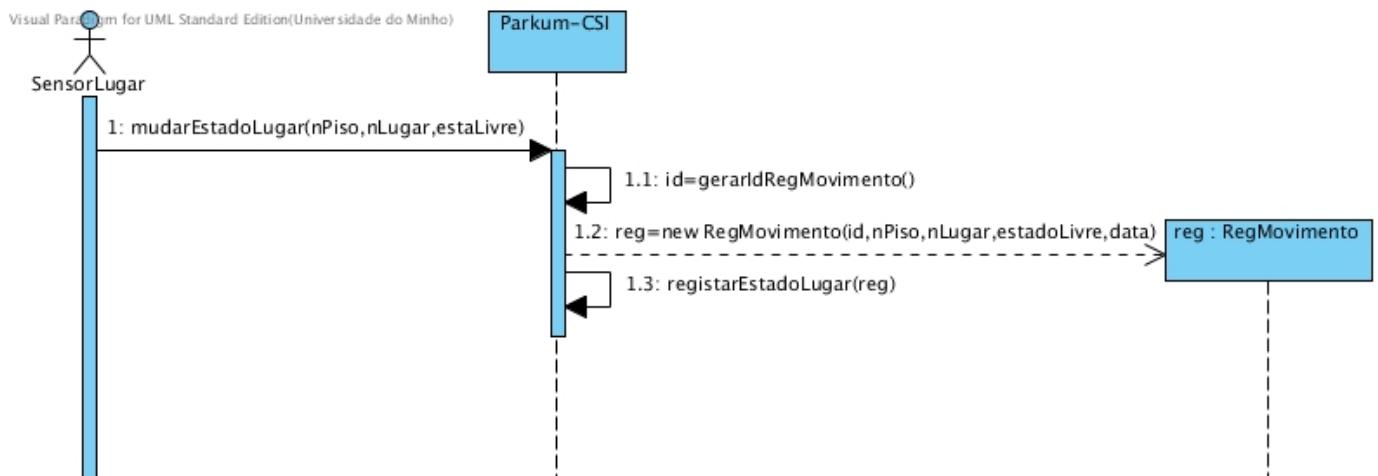


Figura 2.59: Diagrama de sequencia de Registrar Movimento



Figura 2.60: Diagrama de sequencia de Informar Paineis Movimento

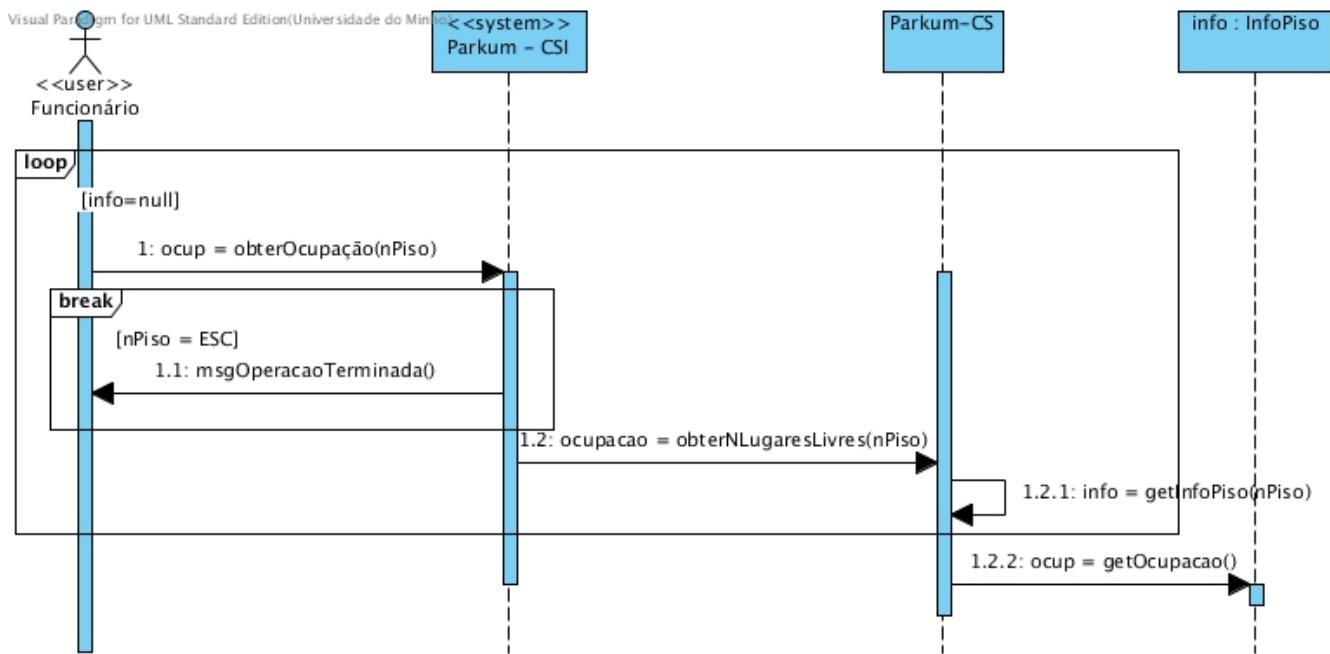


Figura 2.61: Diagrama de sequencia de Consultar ocupação

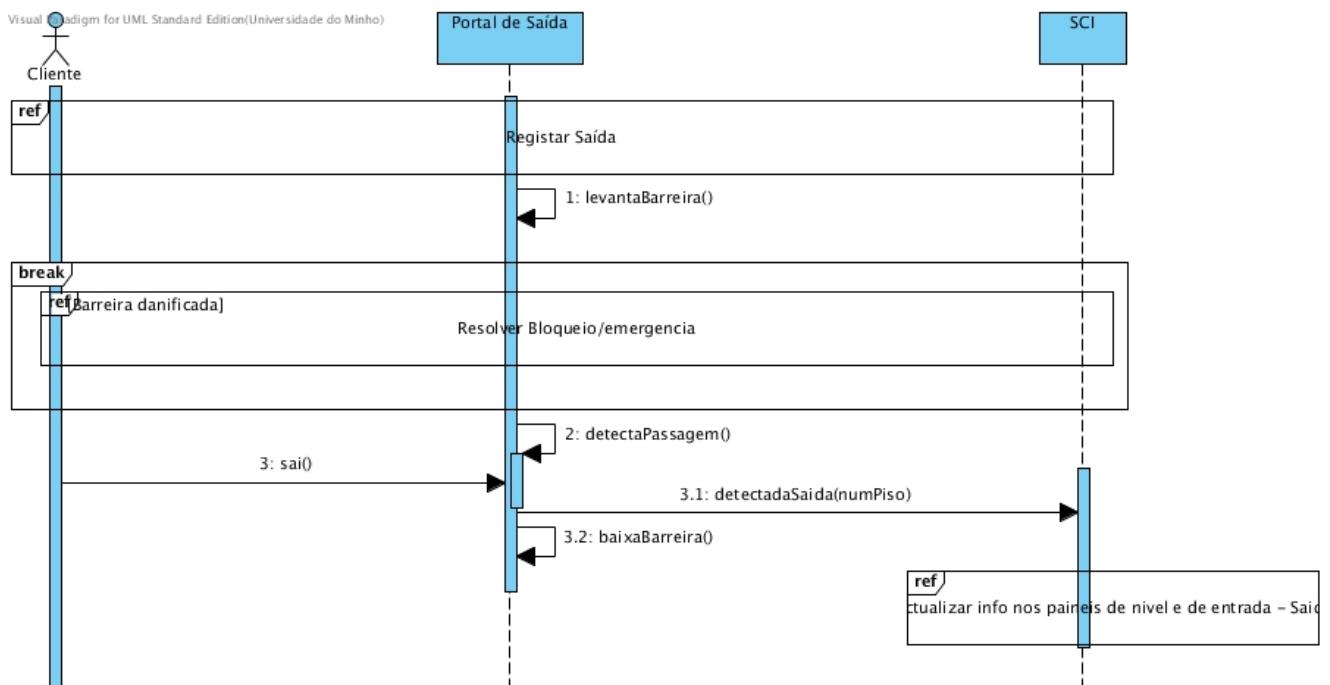
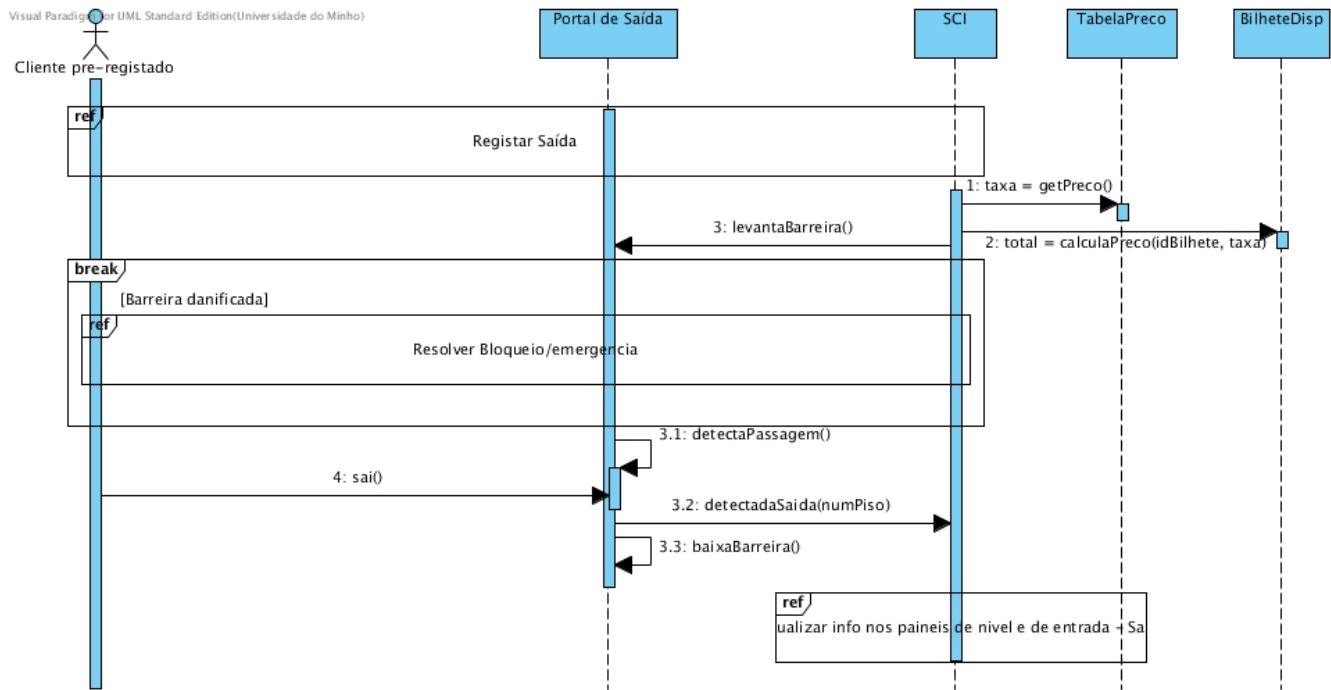
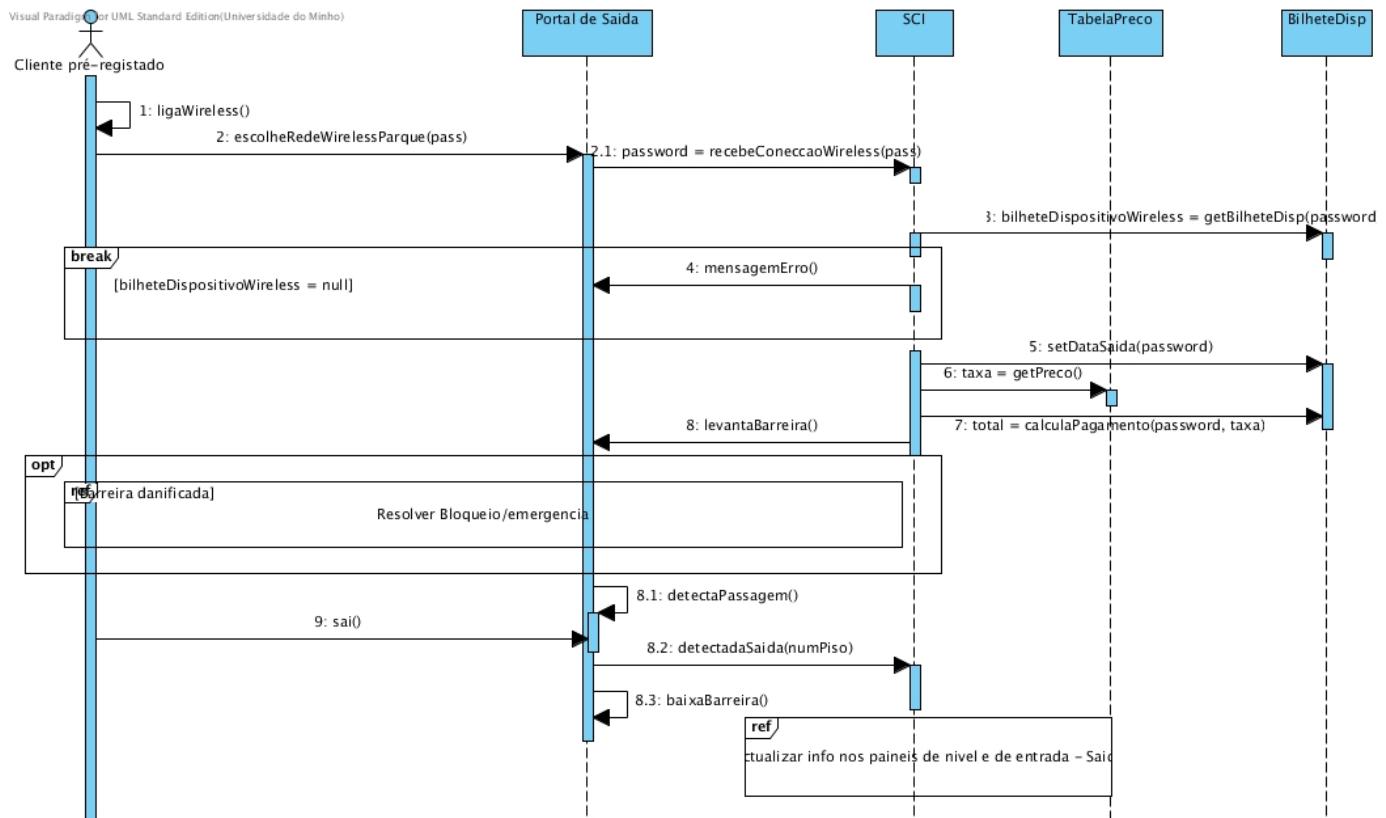


Figura 2.62: Diagrama de sequencia de Sair do parque



**Figura 2.63:** Diagrama de sequencia de Sair do parque pre-registrado



**Figura 2.64:** Diagrama de sequencia de Sair através chamada wireless

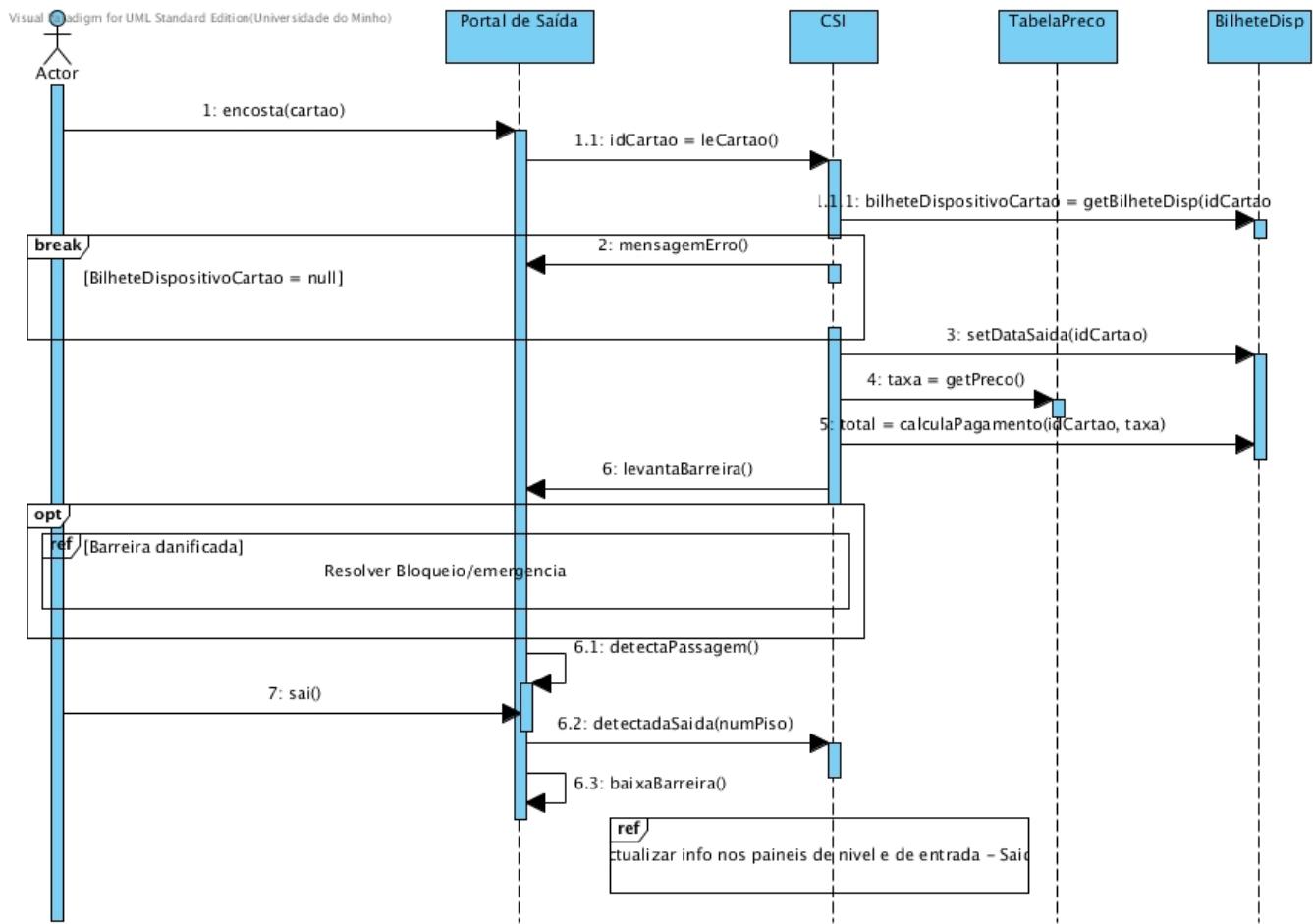


Figura 2.65: Diagrama de sequencia de Sair através cartão cliente

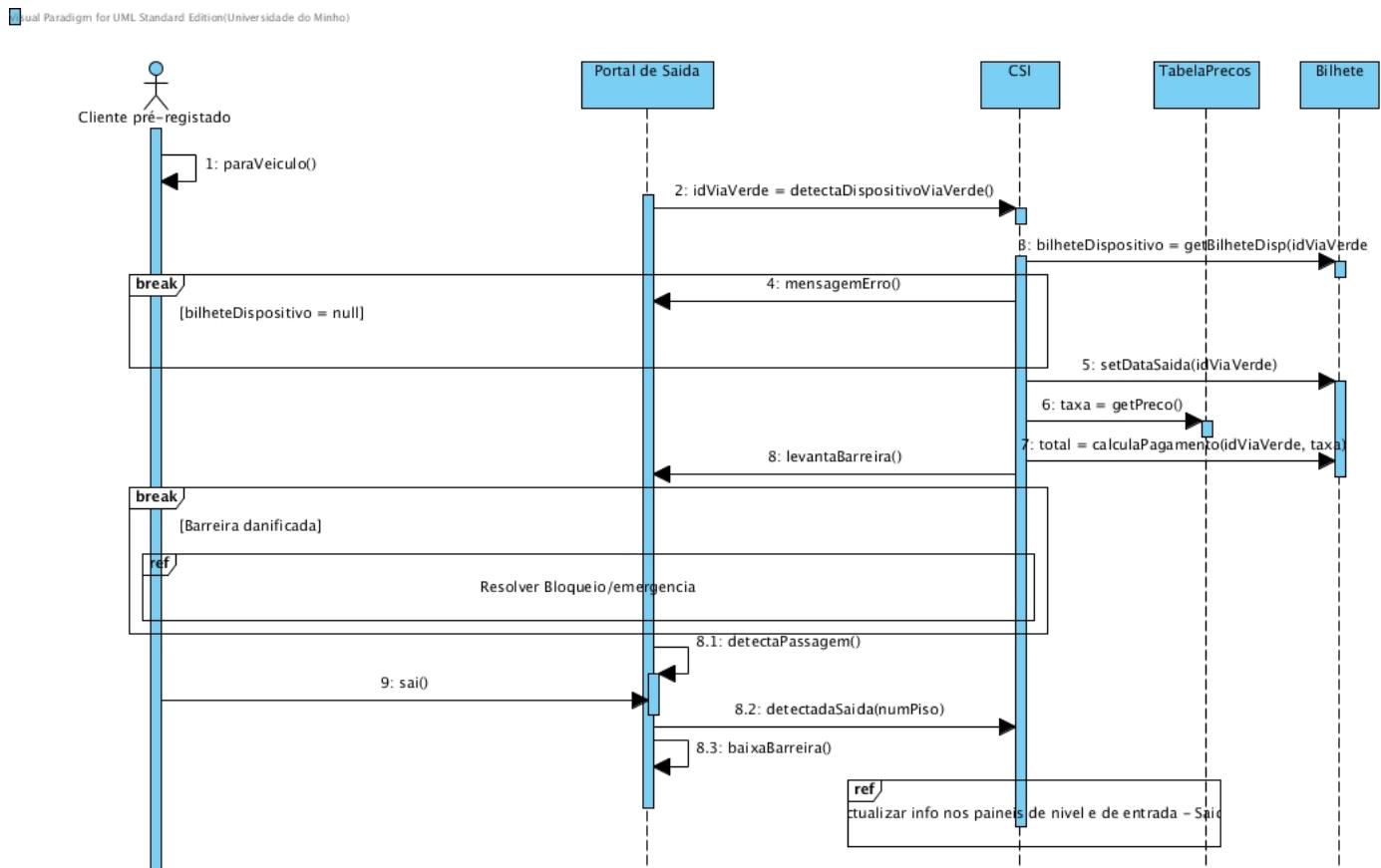


Figura 2.66: Diagrama de sequencia de Sair através Via Verde

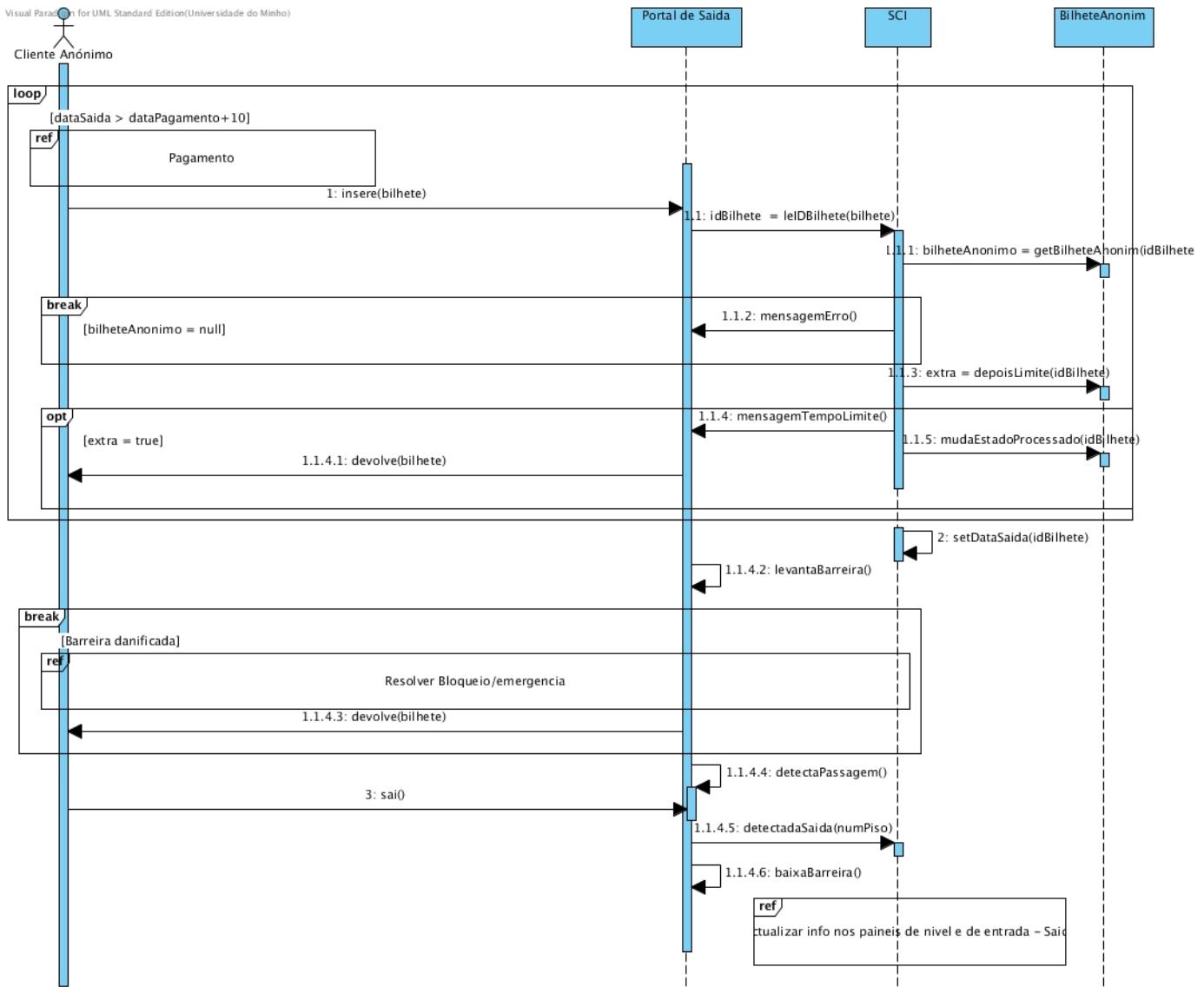


Figura 2.67: Diagrama de sequencia de Sair do parque normal

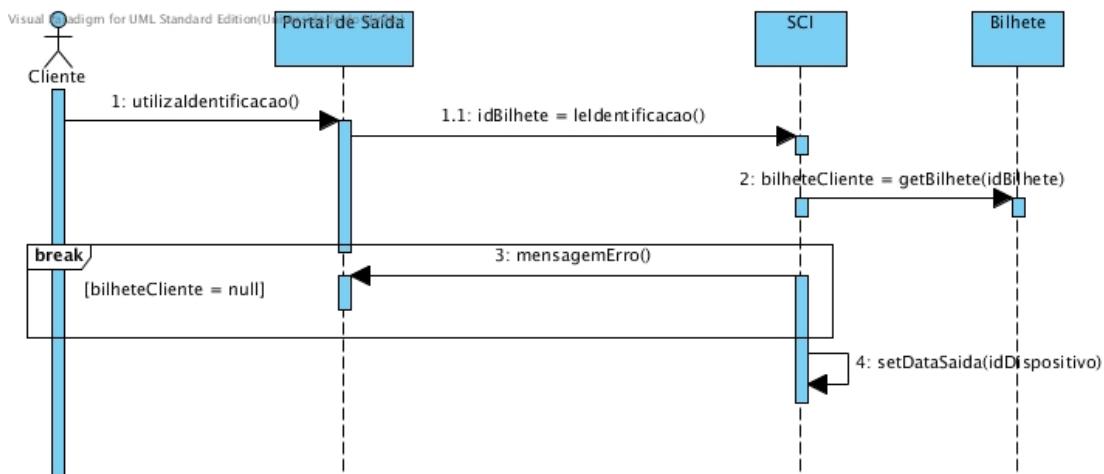
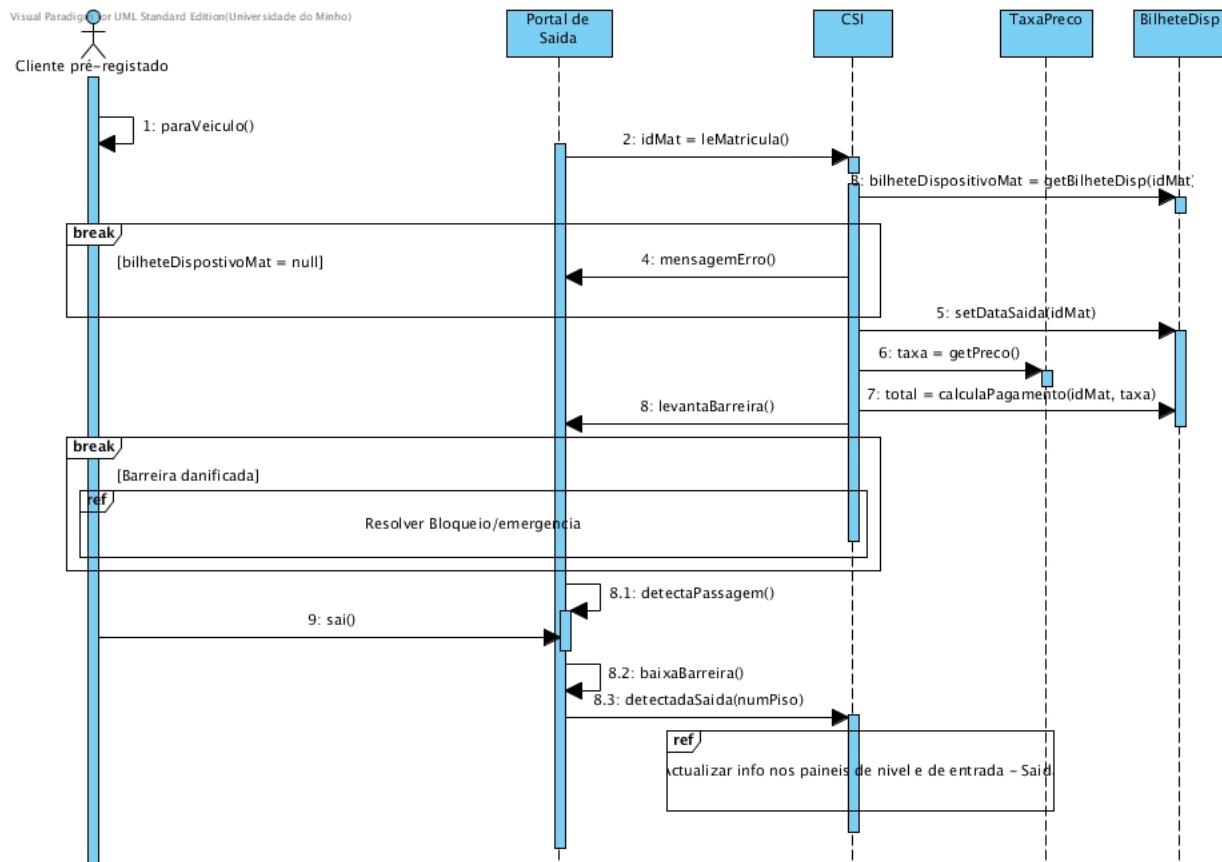
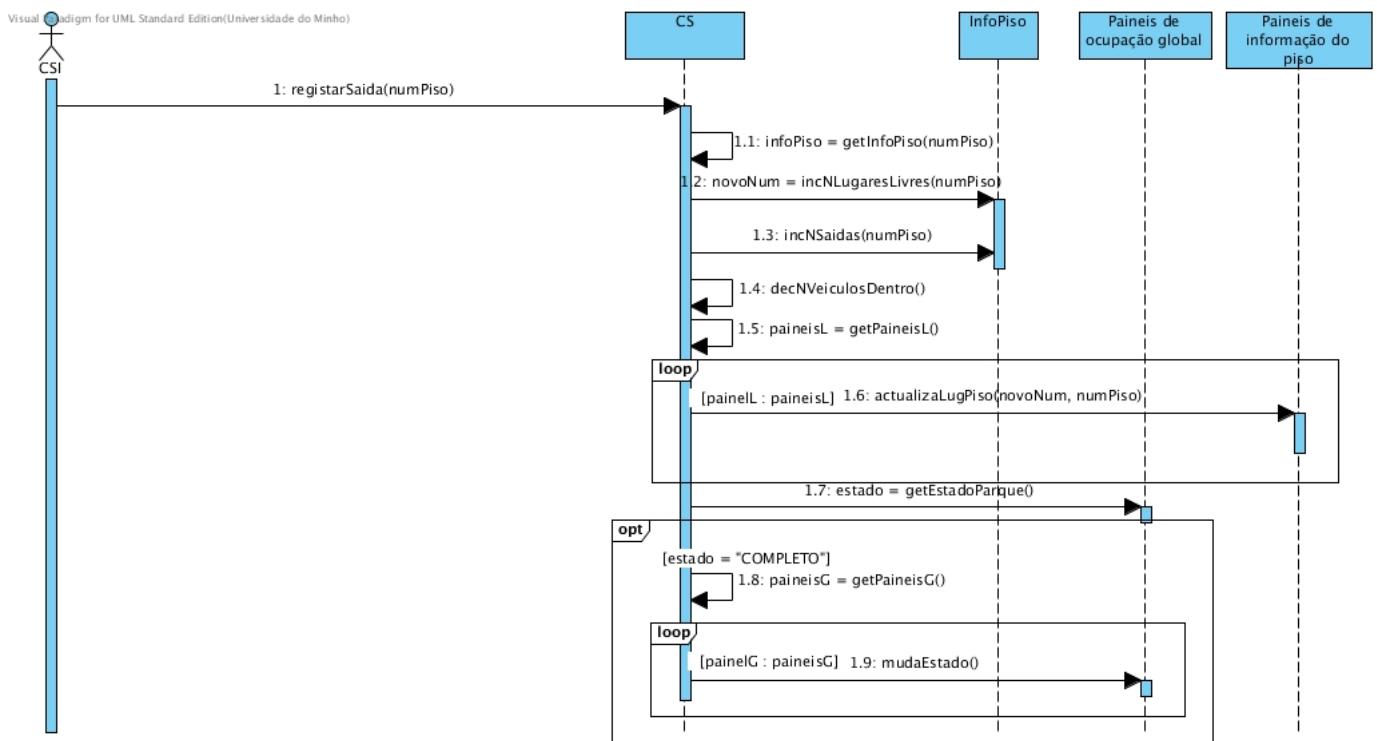


Figura 2.68: Diagrama de sequencia de Registrar Saida



**Figura 2.69:** Diagrama de sequencia de Sair através matricula pré-registada



**Figura 2.70:** Diagrama de sequencia de Actualizar info nos painéis de nível e de entrada - Saida

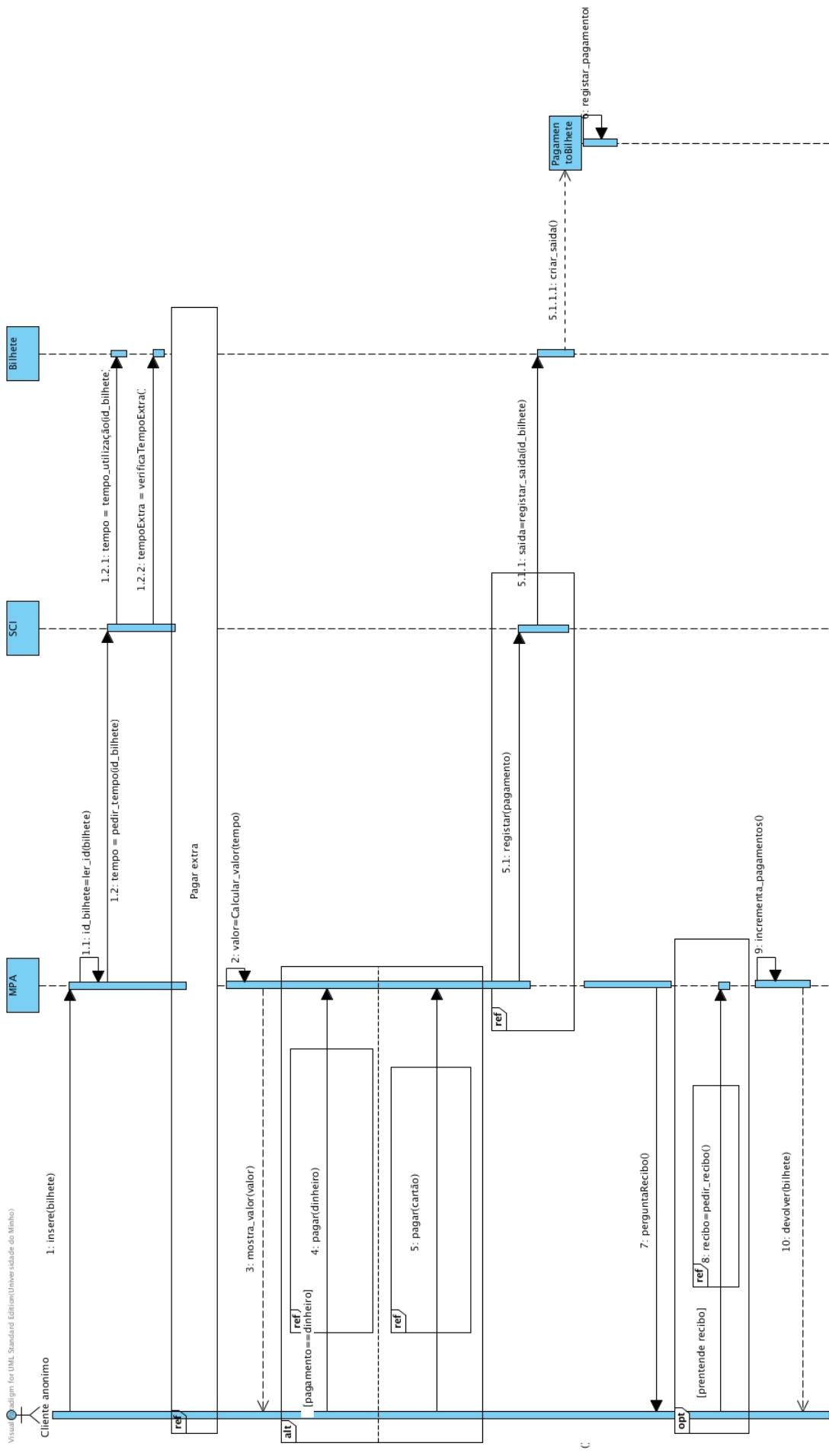


Figura 2.71: Diagrama de sequencia de efectuar pagamento

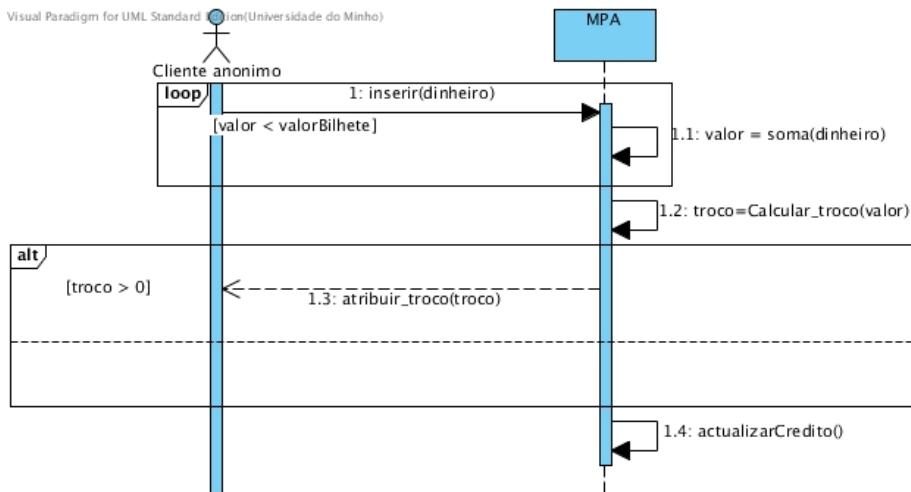


Figura 2.72: Diagrama de sequencia de Pagar com dinheiro

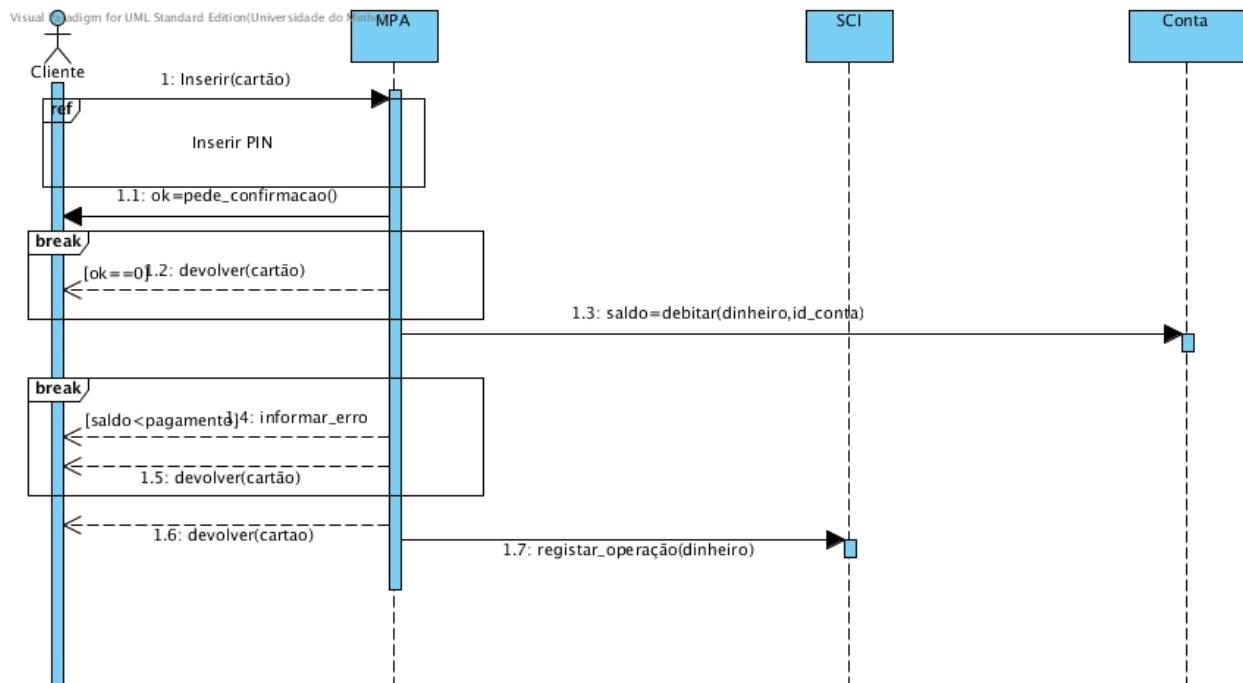
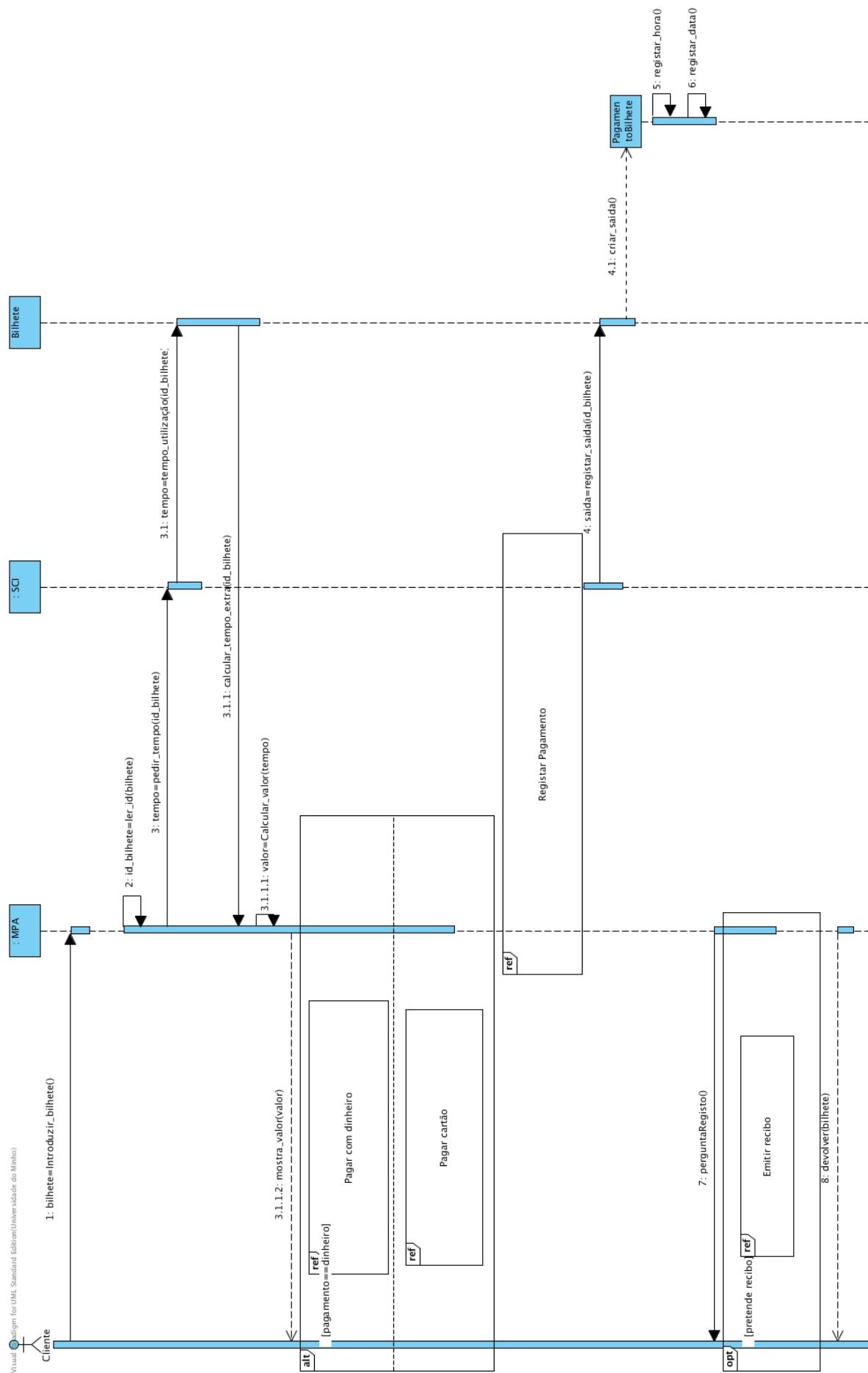


Figura 2.73: Diagrama de sequencia de Pagar cartão



**Figura 2.74:** Diagrama de secuencia de Pagar extra

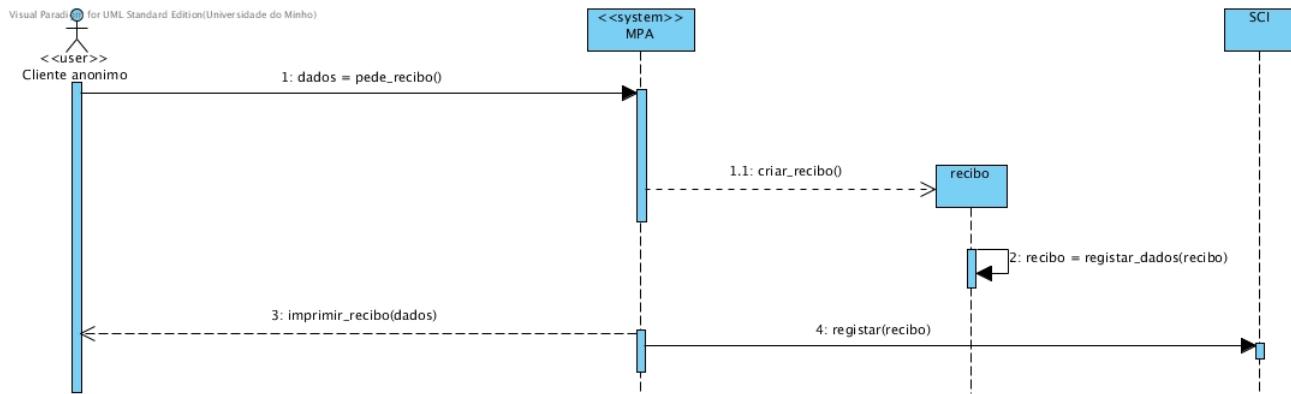


Figura 2.75: Diagrama de sequencia de Emitir recibo

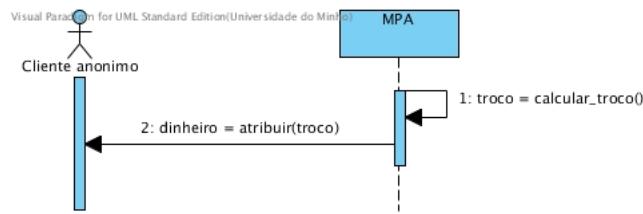


Figura 2.76: Diagrama de sequencia de Atribuir troco

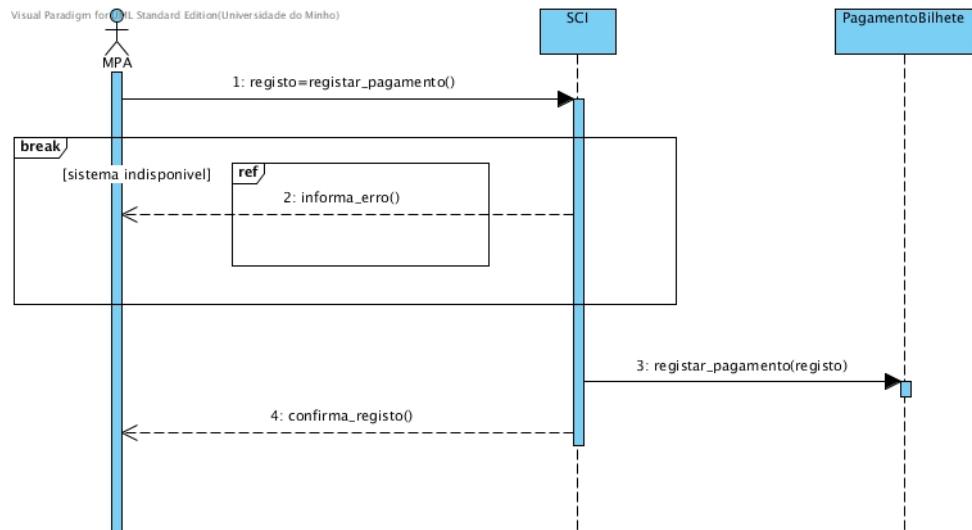


Figura 2.77: Diagrama de sequencia de Registar pagamento

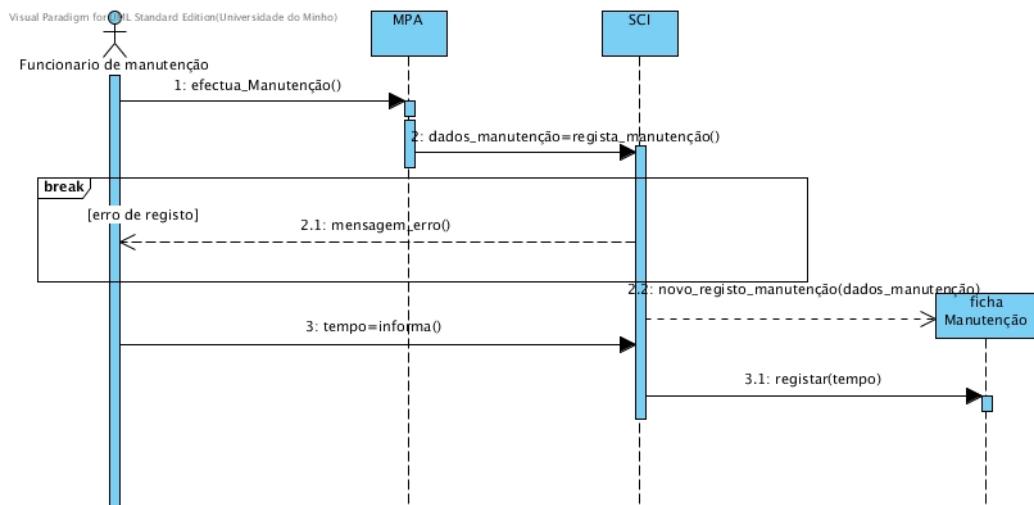
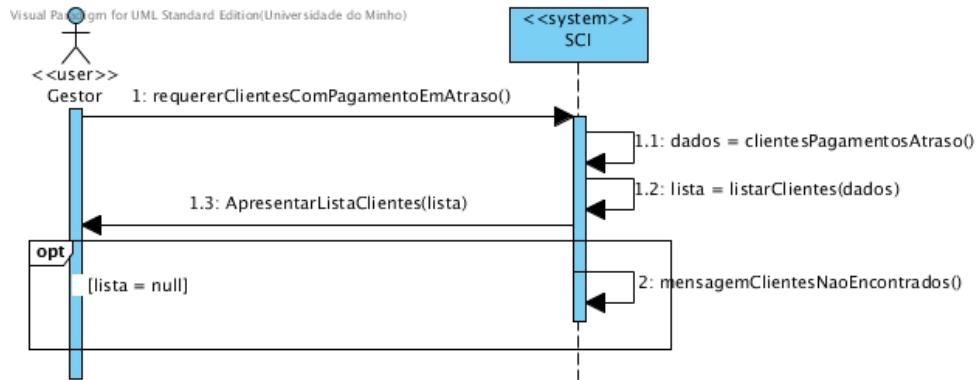
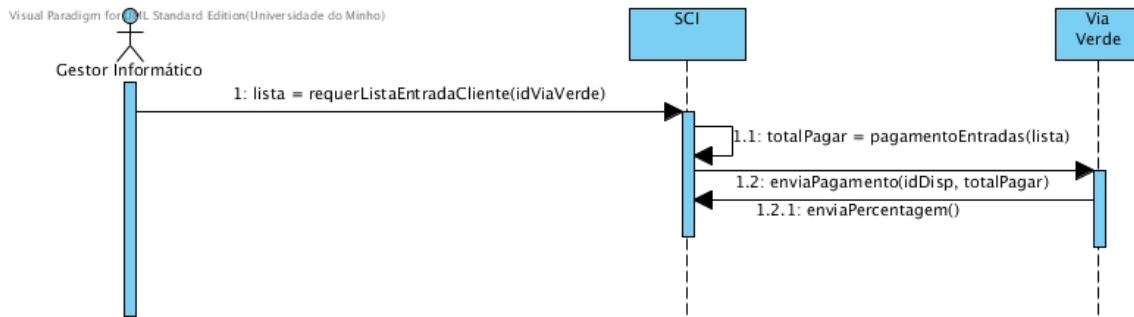


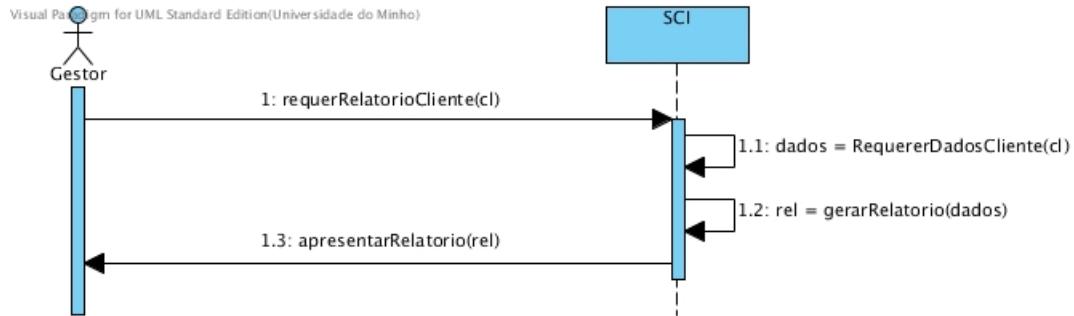
Figura 2.78: Diagrama de sequencia de Manutencao



**Figura 2.79:** Diagrama de sequencia de Listar clientes com pagamentos em atraso



**Figura 2.80:** Diagrama de sequencia de Gerir pagamentos dos clientes via verde



**Figura 2.81:** Diagrama de sequencia de Gerar relatorios de cliente

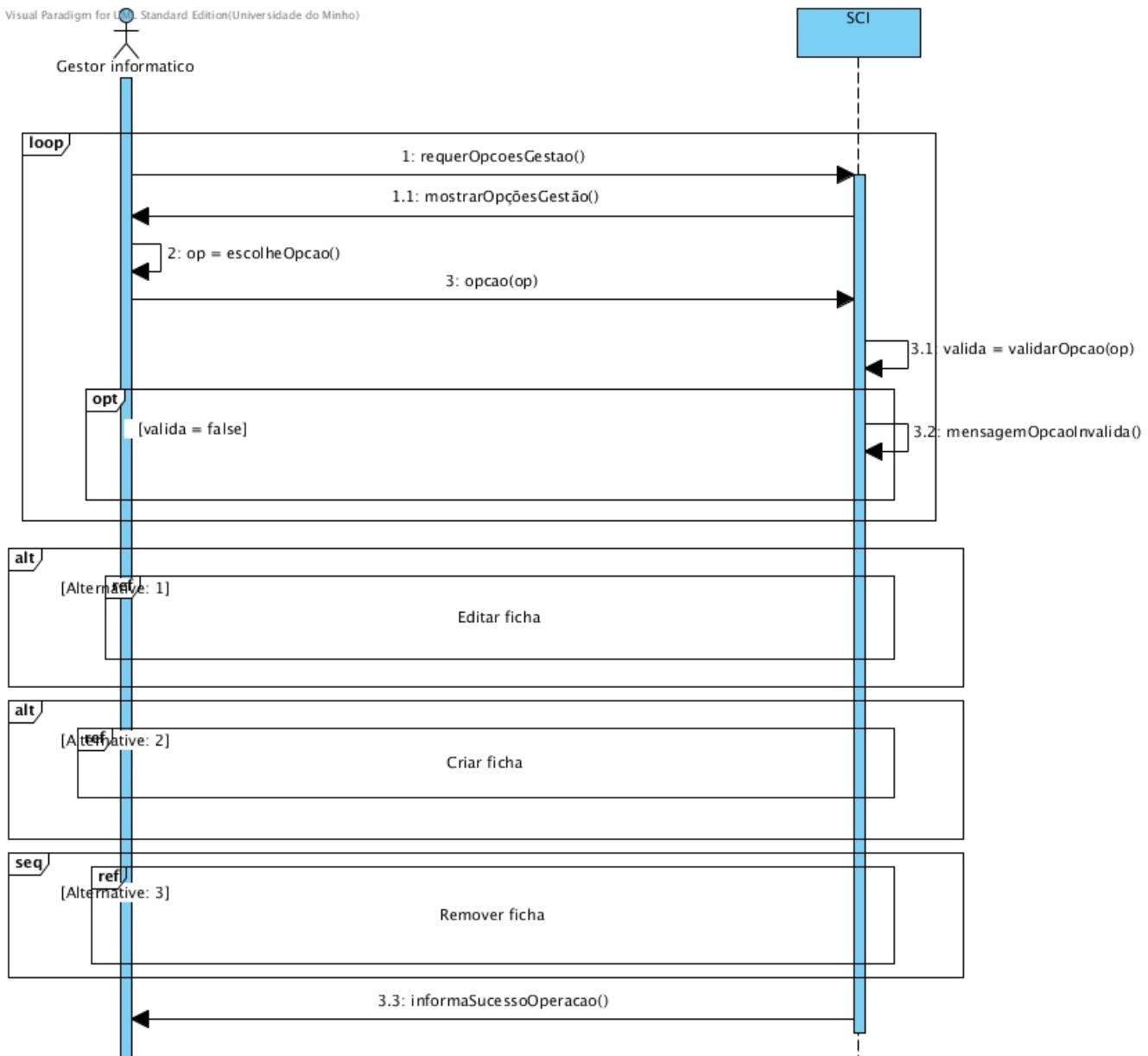


Figura 2.82: Diagrama de sequencia de Gerir fichas de cliente

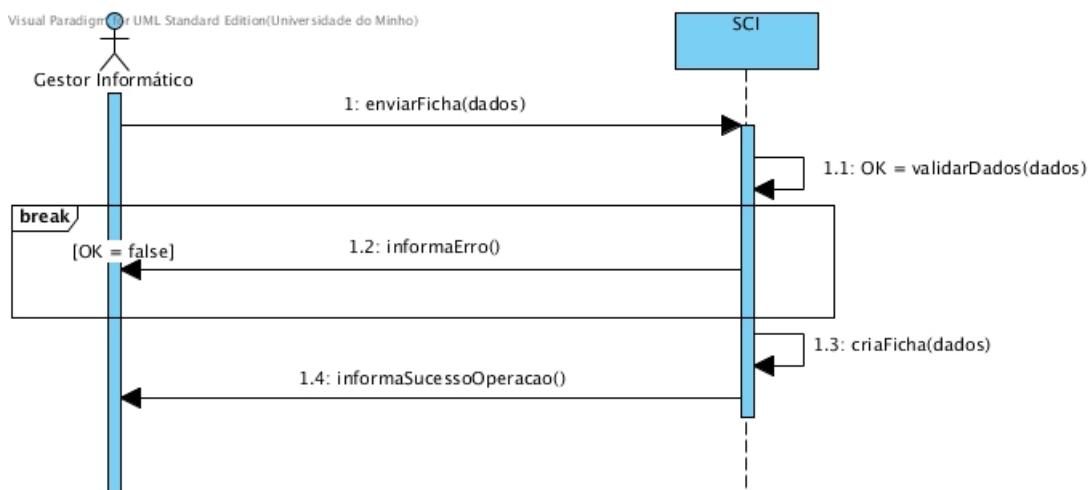
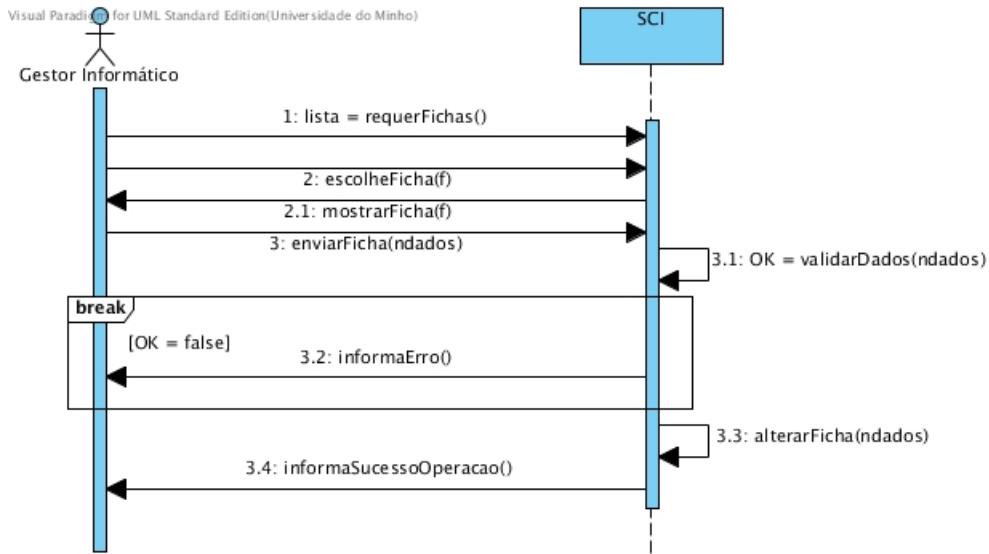
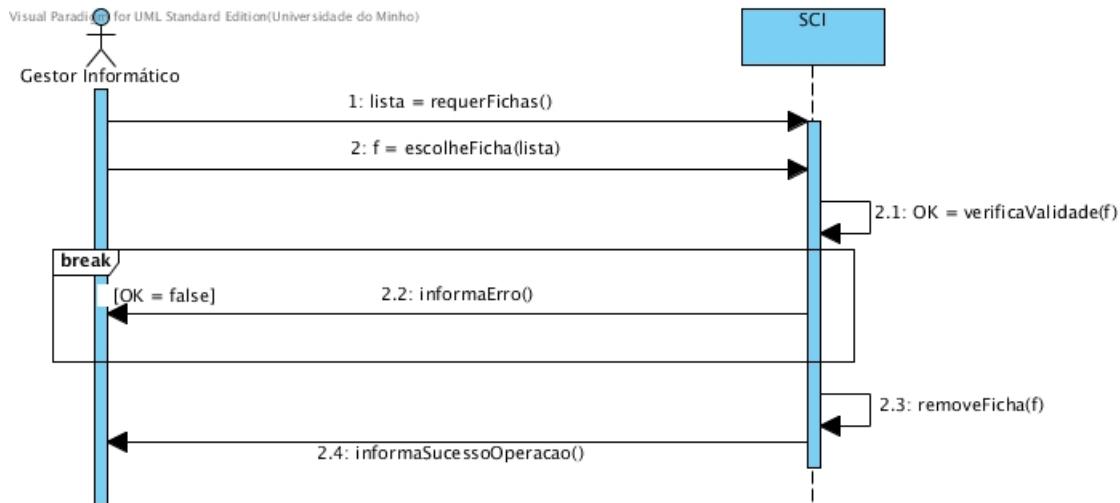
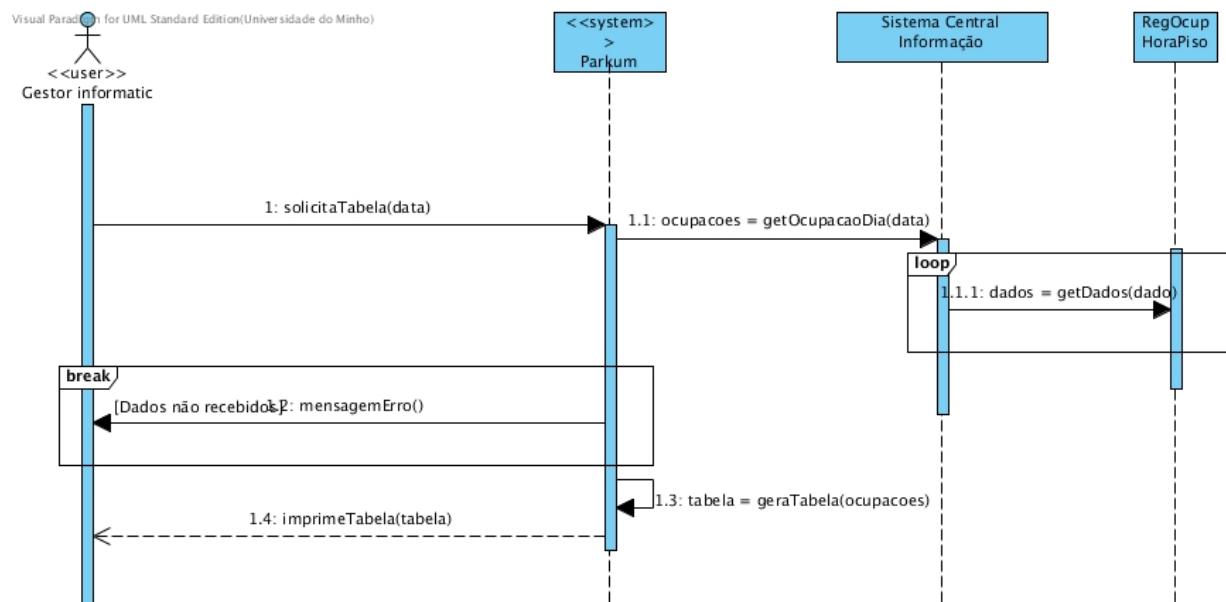
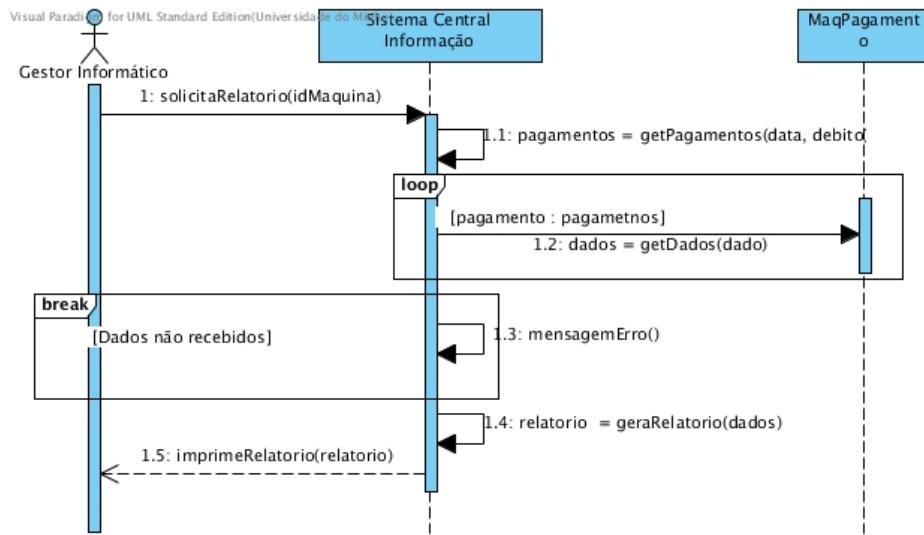
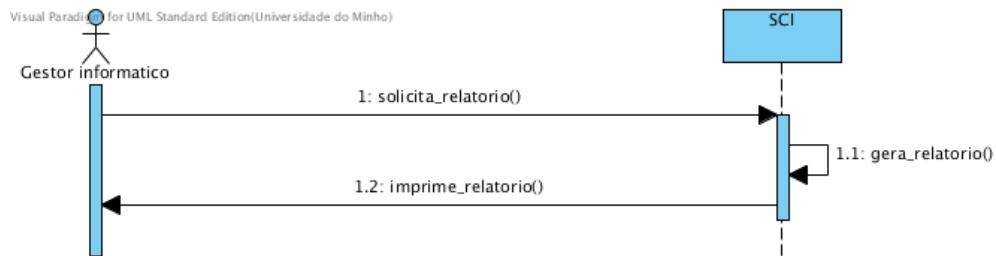


Figura 2.83: Diagrama de sequencia de Criar ficha

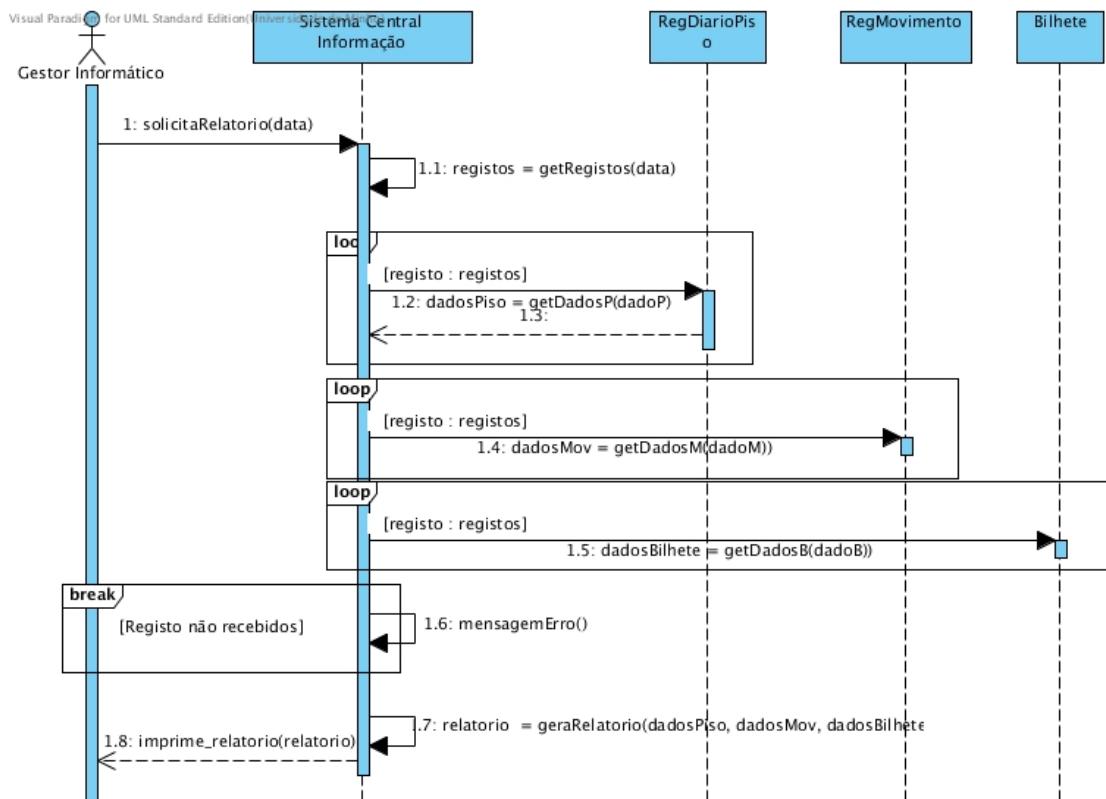
**Figura 2.84:** Diagrama de sequencia de Editar ficha**Figura 2.85:** Diagrama de sequencia de Remover ficha**Figura 2.86:** Diagrama de Tratar Estatísticas



**Figura 2.87:** Diagrama de sequencia elaborar relatorios de maquinas de pagamentos



**Figura 2.88:** Diagrama de elaborar relatorios



**Figura 2.89:** Diagrama de elaborar Relatorios diarios de entradas e saidas

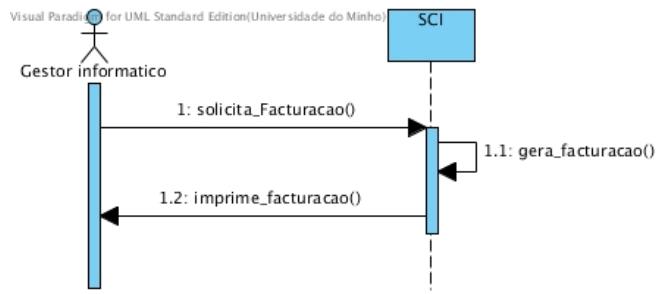


Figura 2.90: Diagrama de Processar facturacao

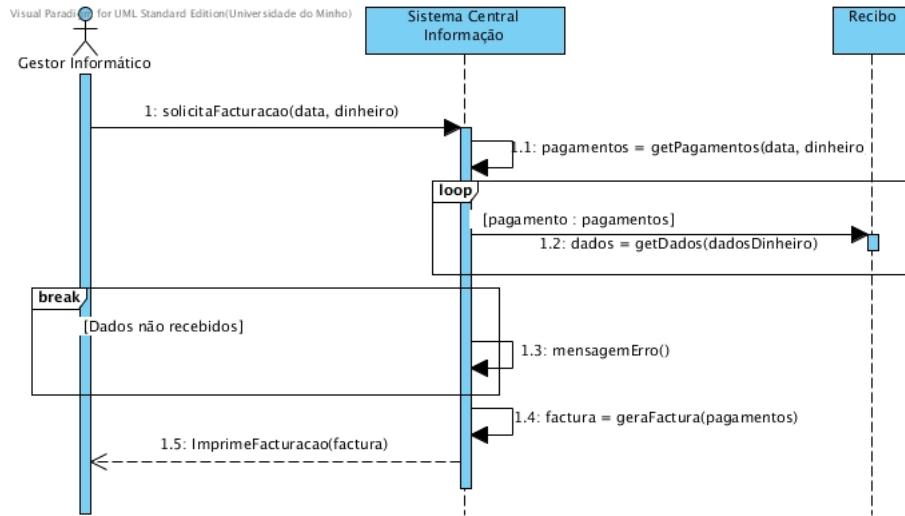


Figura 2.91: Diagrama de Processar Facturacao em dinheiro

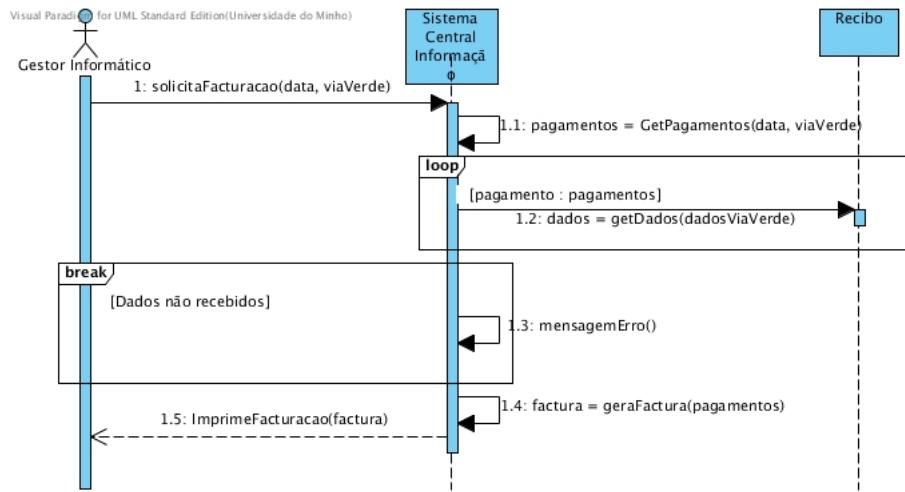


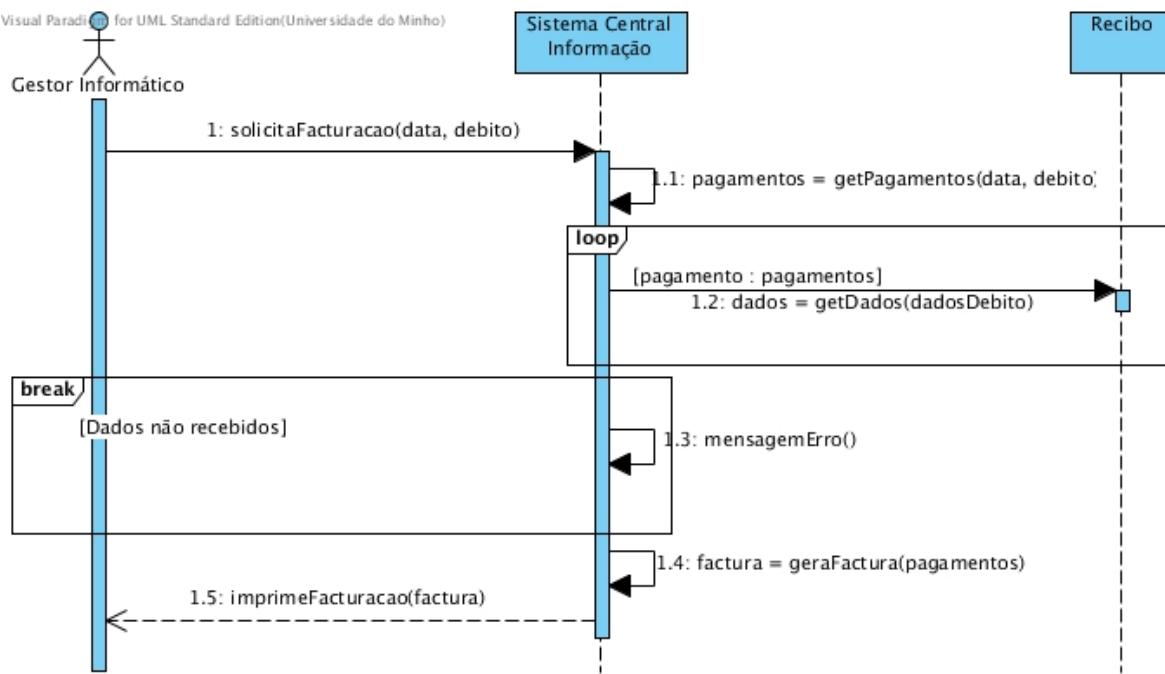
Figura 2.92: Diagrama de Processar Facturacao através de via-verde

### 2.1.7 Sub-Sistemas

Após o refinamento dos diagramas apresentados anteriormente pode-se facilmente identificar quais os sub-sistemas do sistema **PARKUM**, estes sub-sistemas são intuitivos.

Podemos então dividir os sub-sistemas nos seguintes:

- Portal de entrada (PE)
- Portal de saída (PS)
- Maquina de pagamento (MPA)



**Figura 2.93:** Diagrama de Processar Facturacao atraves de debito

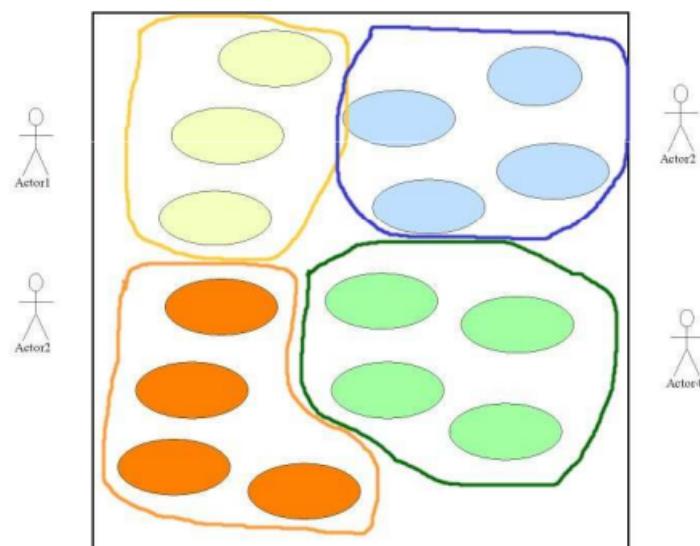
- Sistema central de informação (SCI)
- Parkum-cs (CS)
- Sistema de sensores

O portal de entrada será responsável pelas tarefas de entrada, o portal de saída responsável pelas tarefas de saída, máquina de pagamento pelas tarefas de pagamento, sistema de sensores pelas tarefas de actualização de lugares no parque, o parkum-cs irá tratar das tarefas de lugares no parque e nos diferentes pisos deste e por sim o sistema central de informação que irá interagir com todos os outros sub-sistemas, será o o sub-sistema que terá de certa forma maior destaque no sistema , pois este irá interagir com todos os outros.

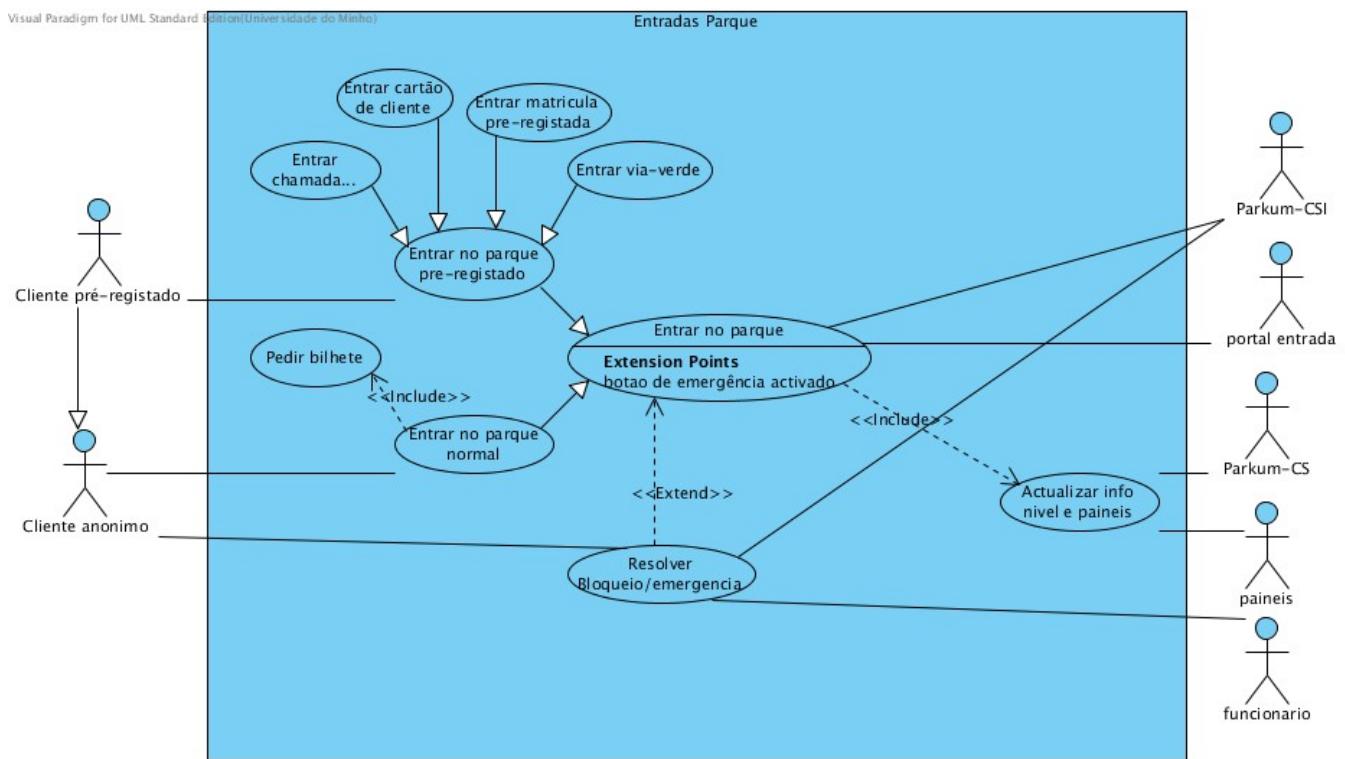
### 2.1.8 Diagrama de Use Case de sistema refinado

Após a identificação dos diferentes sub-sistemas e do refinamento dos diagramas de sequencia, ficamos aptos a refinar o digrama de use case principal.

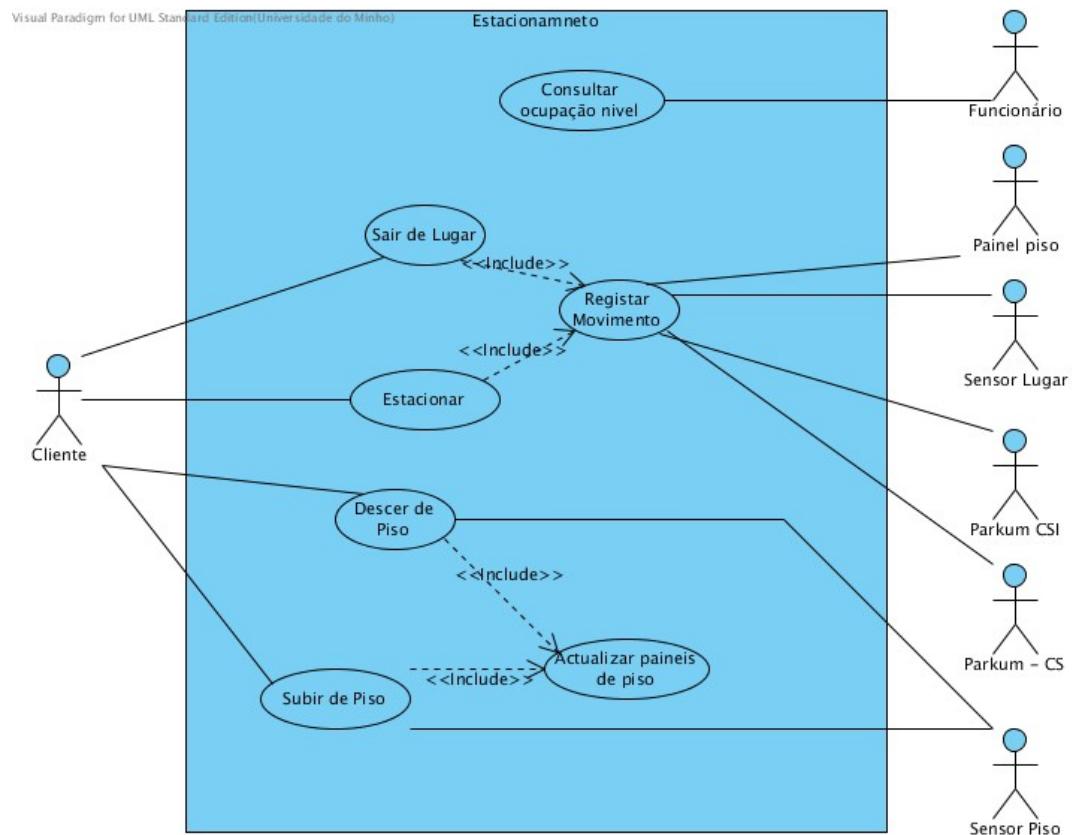
Desta forma dividimos o use case principal por "areas de funcionalidade", sendo que não deixam de pertencer ao sistema principal mas ficando melhor organizadas.



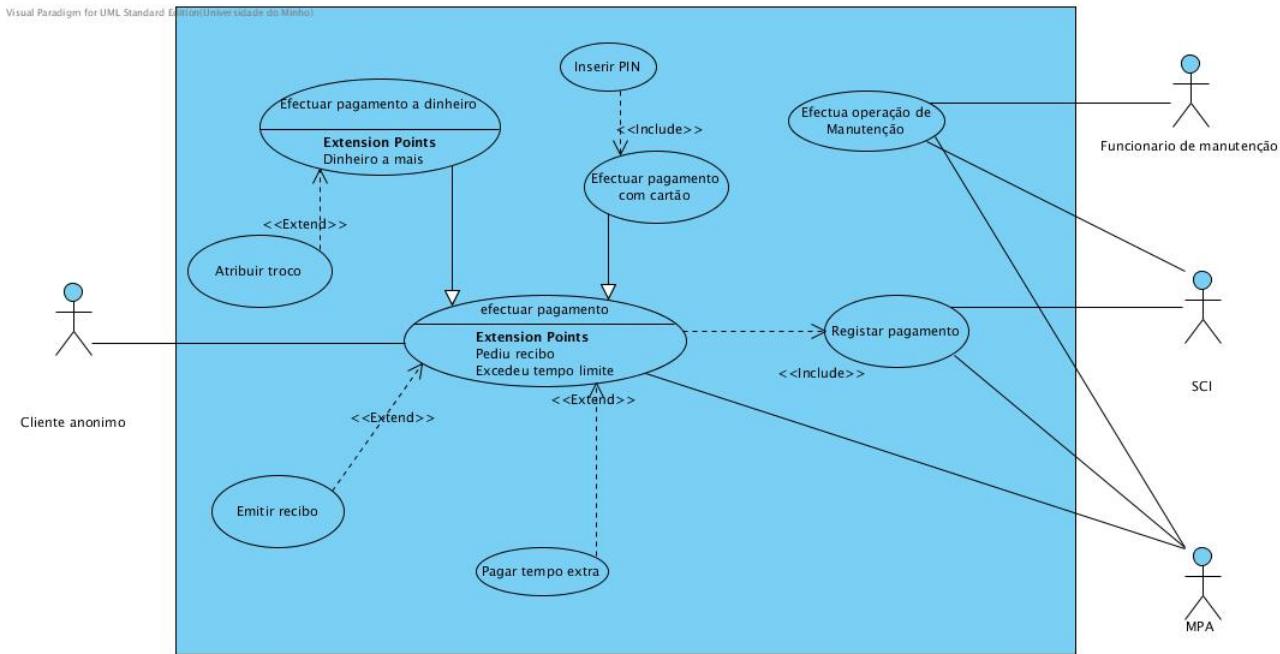
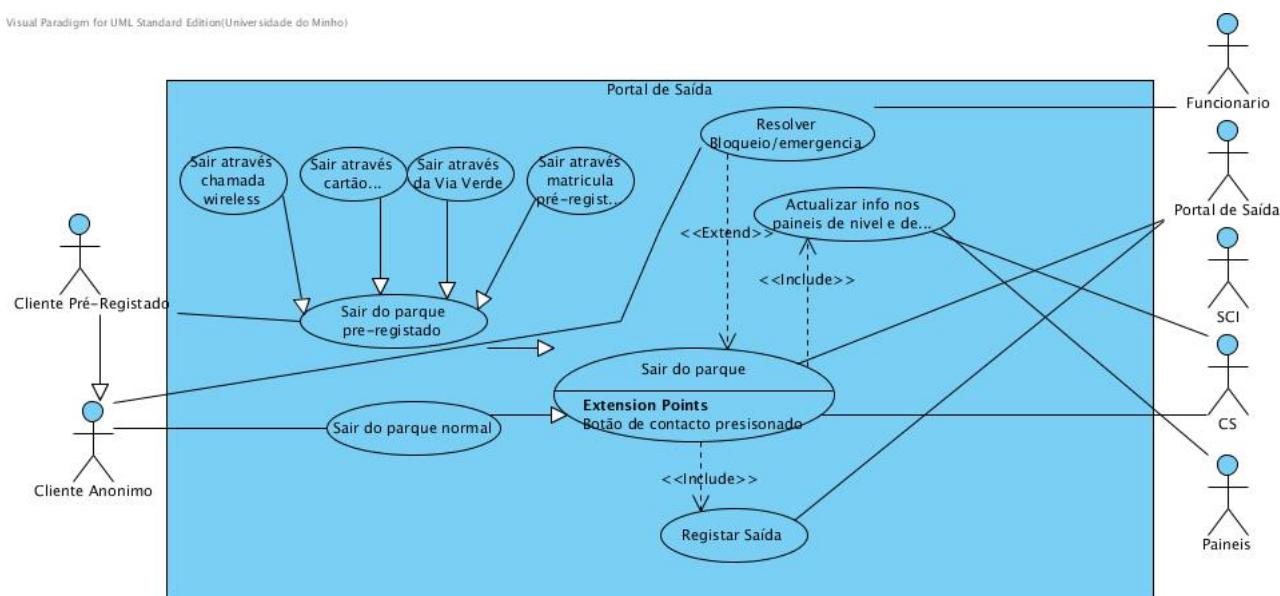
**Figura 2.94:** Diagrama de Use Case de sistema refinado



**Figura 2.95:** Use cases de entrada no parque



**Figura 2.96:** Use cases de estacionamento

**Figura 2.97:** Use cases de pagamento**Figura 2.98:** Use cases de saída do parque

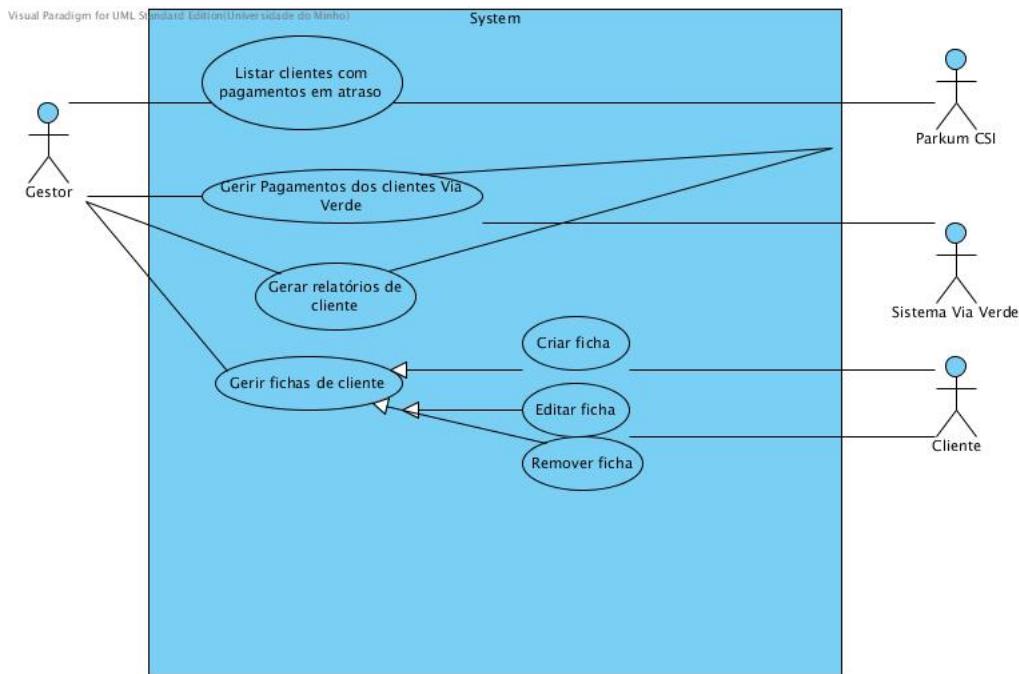


Figura 2.99: Use cases de gestão de clientes

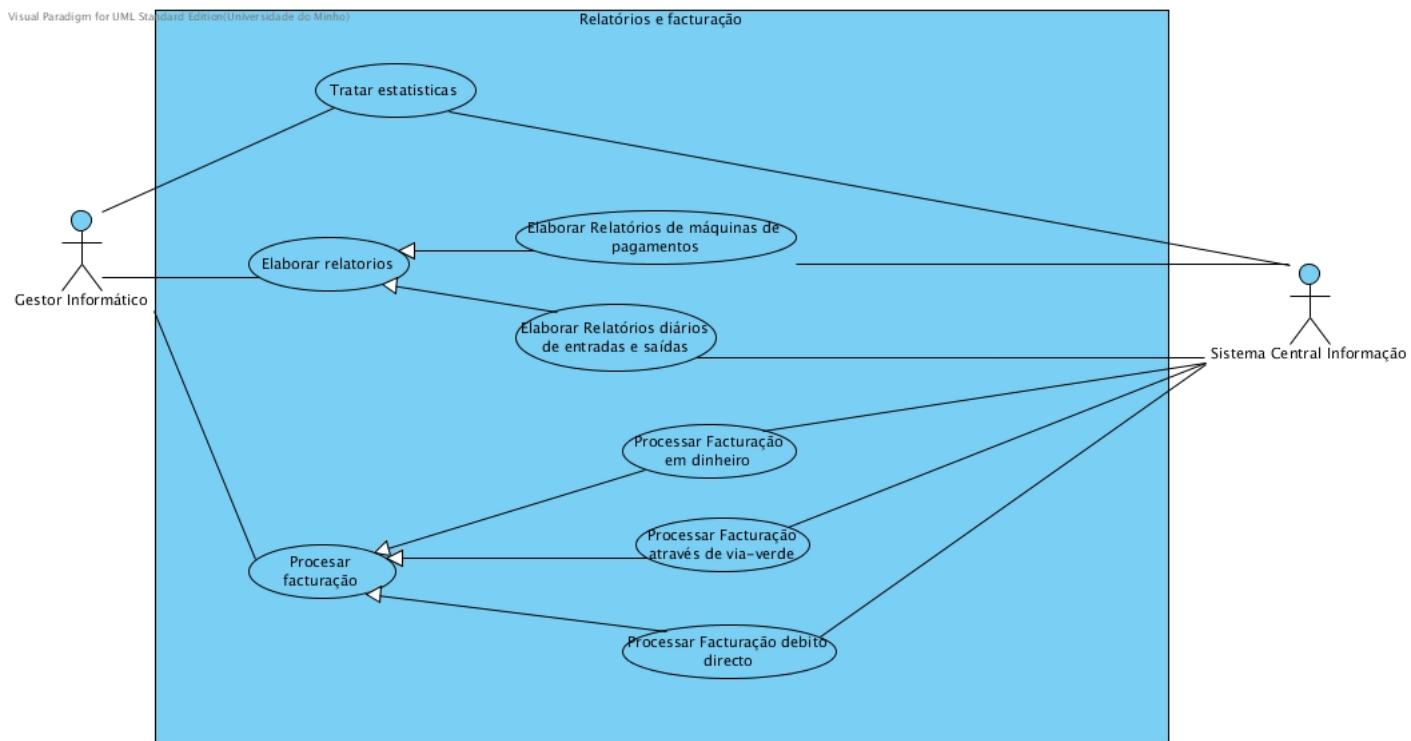


Figura 2.100: Use cases de facturação e relatórios

### 2.1.9 Diagramas de acividades

De forma a perceber melhor o funcionamento de alguns use case (os principais) criamos os diagramas de acitvidade destes que nos vão permitir ter noção mais nitida do funciomamento destes use cases.

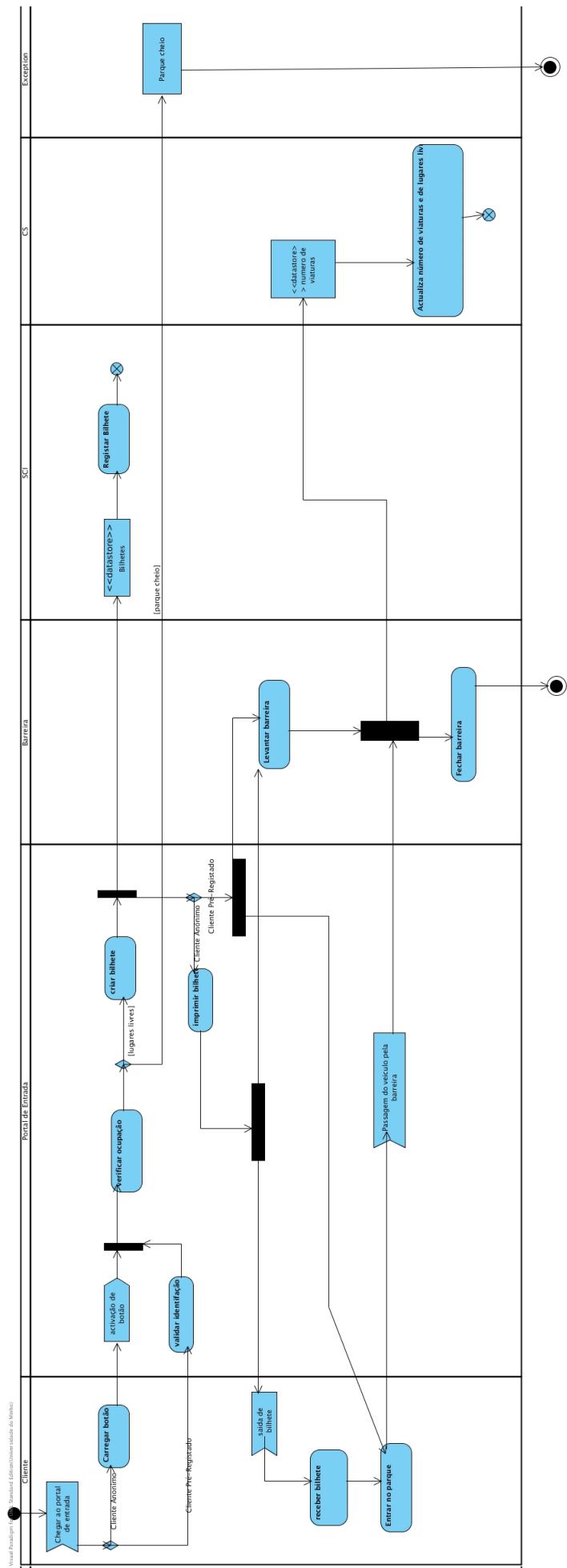


Figura 2.101: Diagrama de Actividade de entrada no parque

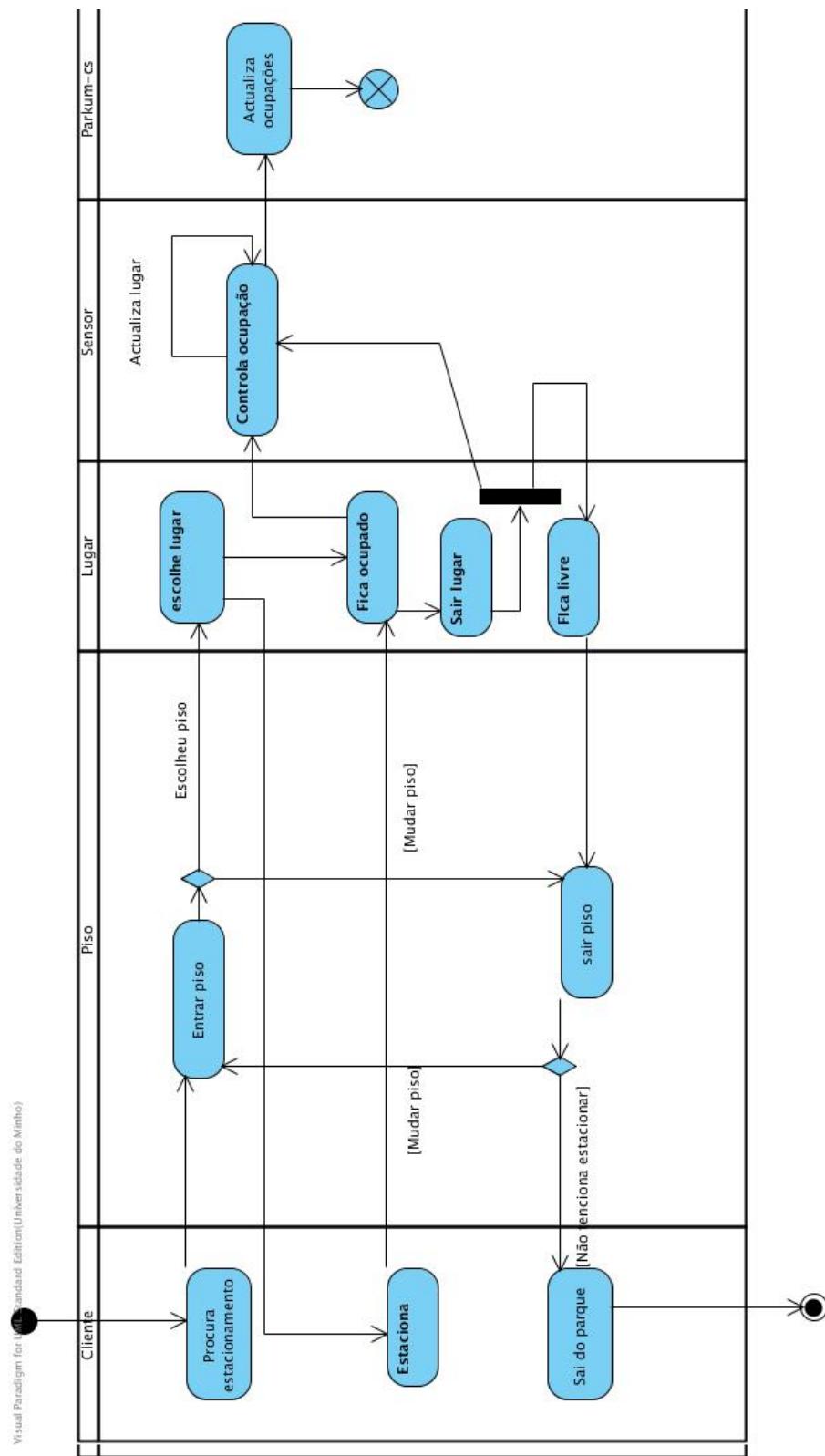


Figura 2.102: Diagrama de Actividade de estacionamento

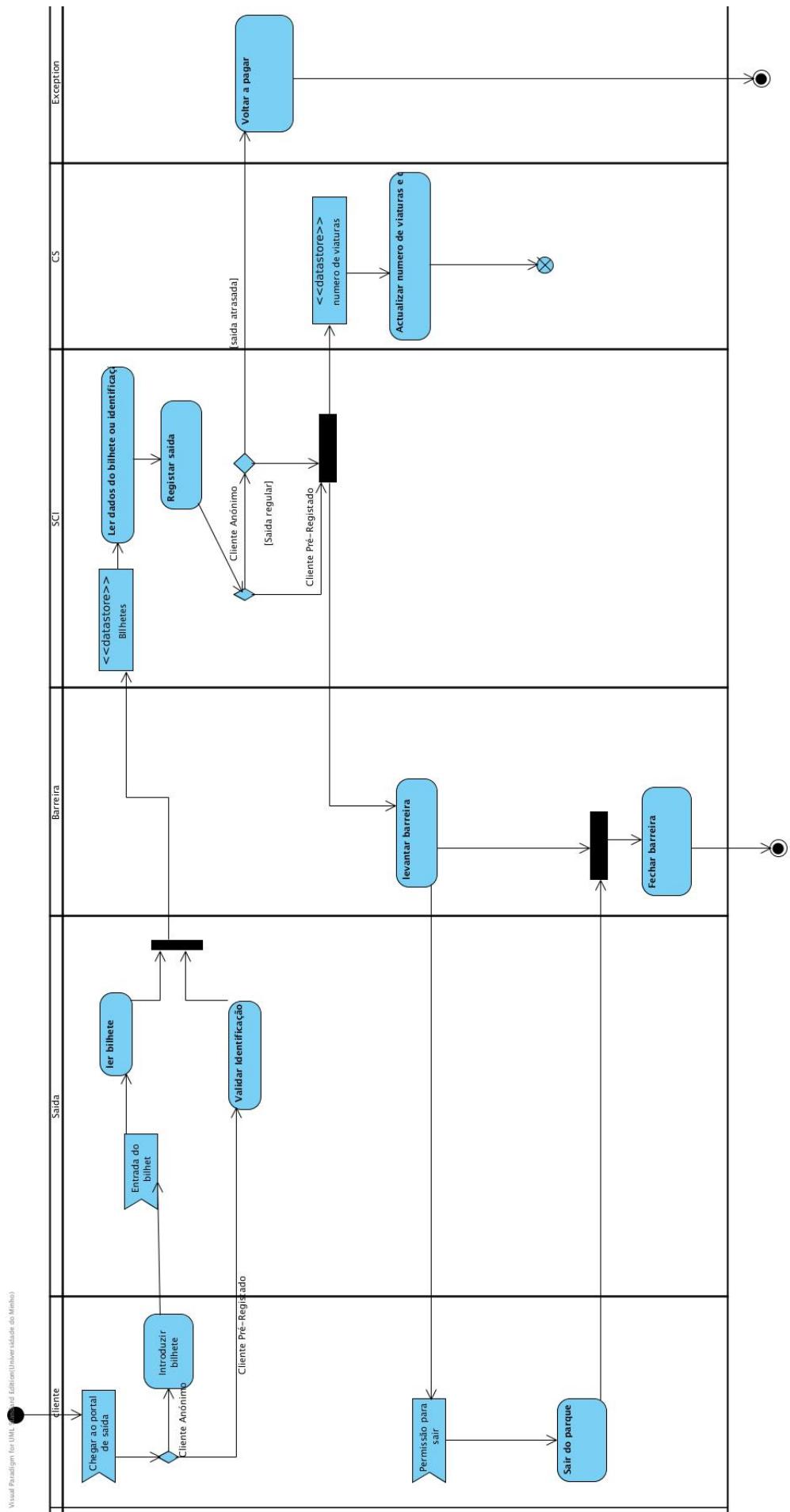
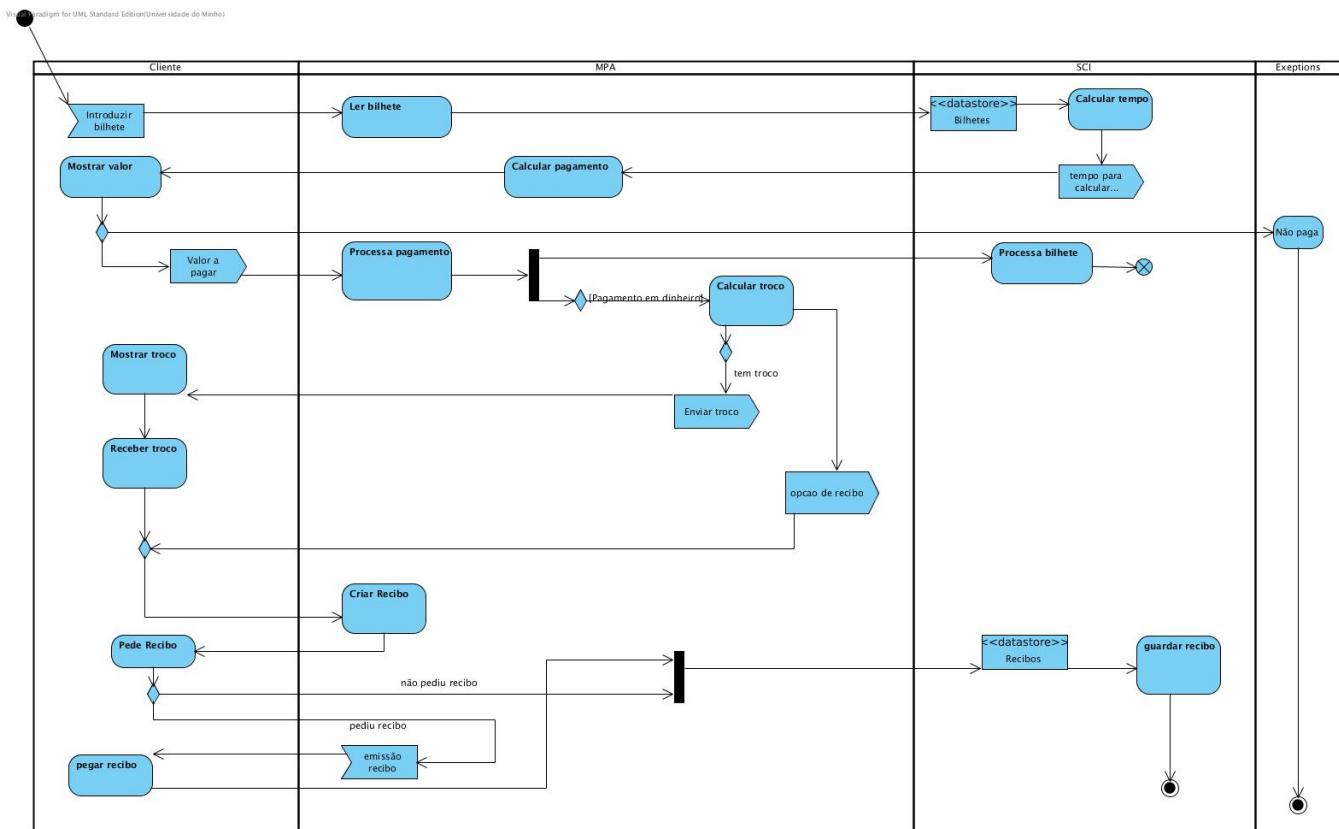


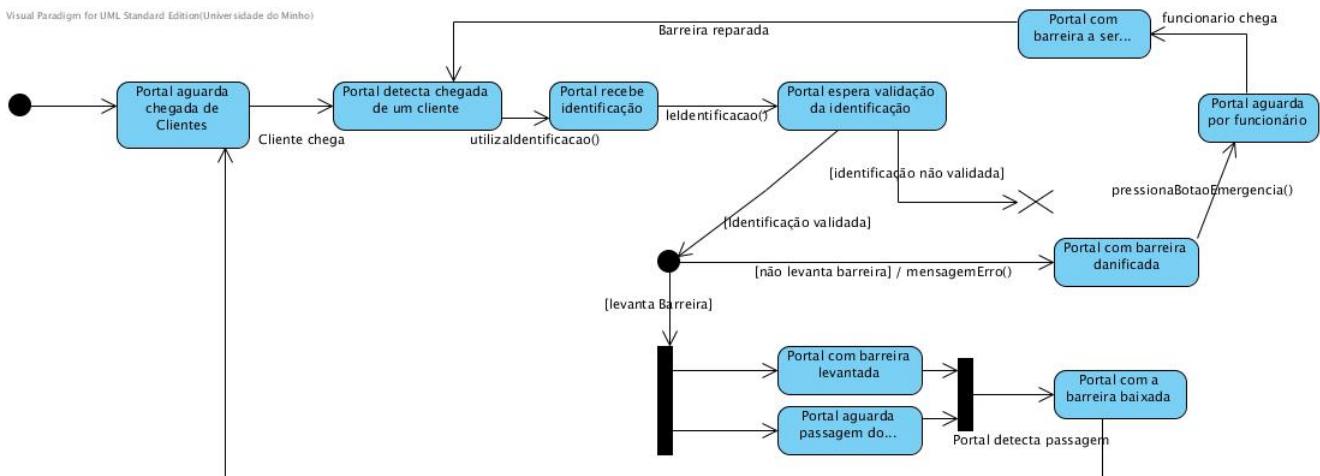
Figura 2.103: Diagrama de Actividade de saída



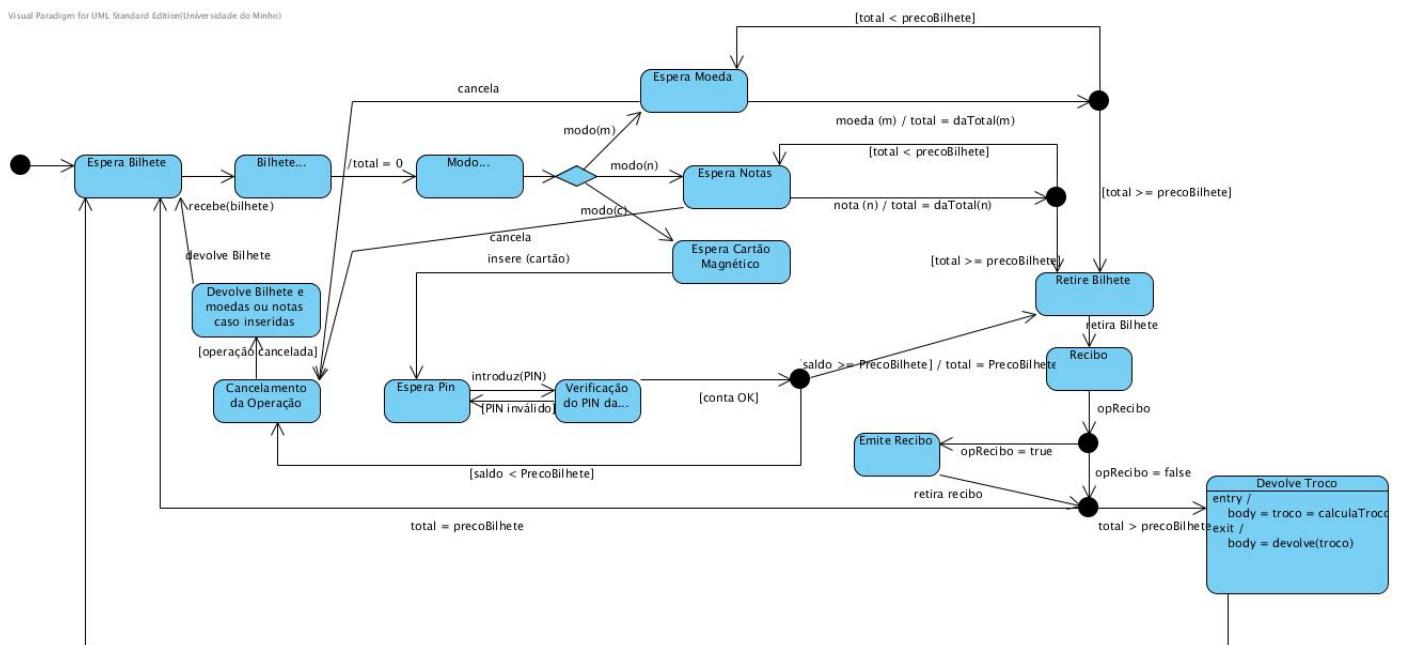
**Figura 2.104:** Diagrama de Actividade de pagamento

### 2.1.10 Diagramas de estado

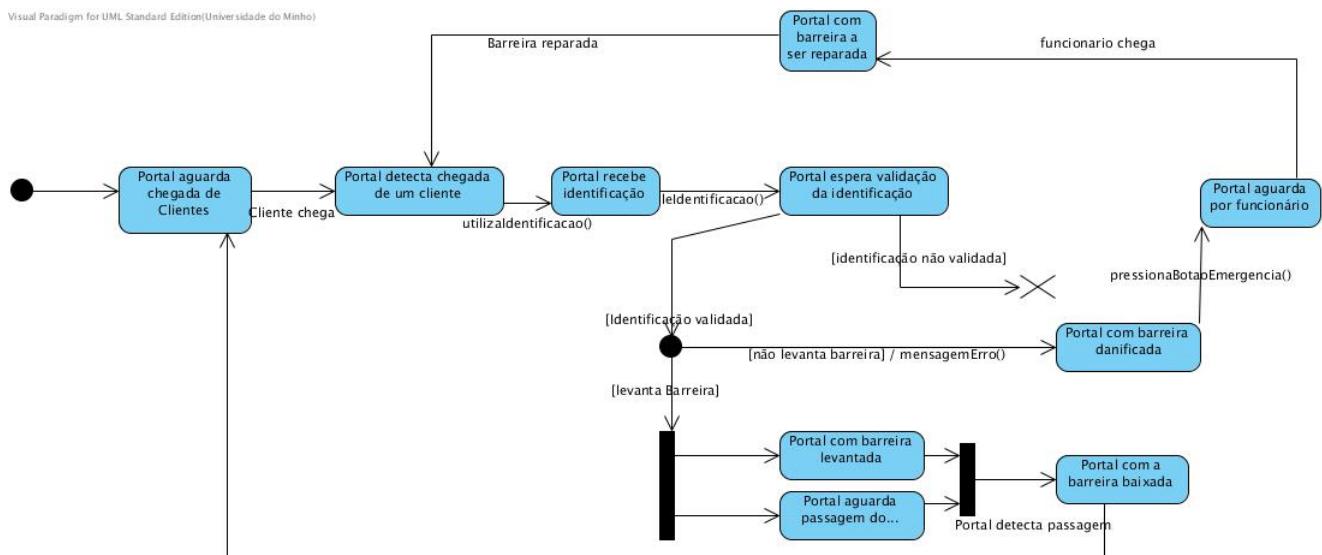
Para percebermos melhor o funcionamento dos sistemas apresentamos os seguintes diagramas de estado:



**Figura 2.105:** Diagrama de Estado do portal de entrada



**Figura 2.106:** Diagrama de Estado da Maquina de pagamento

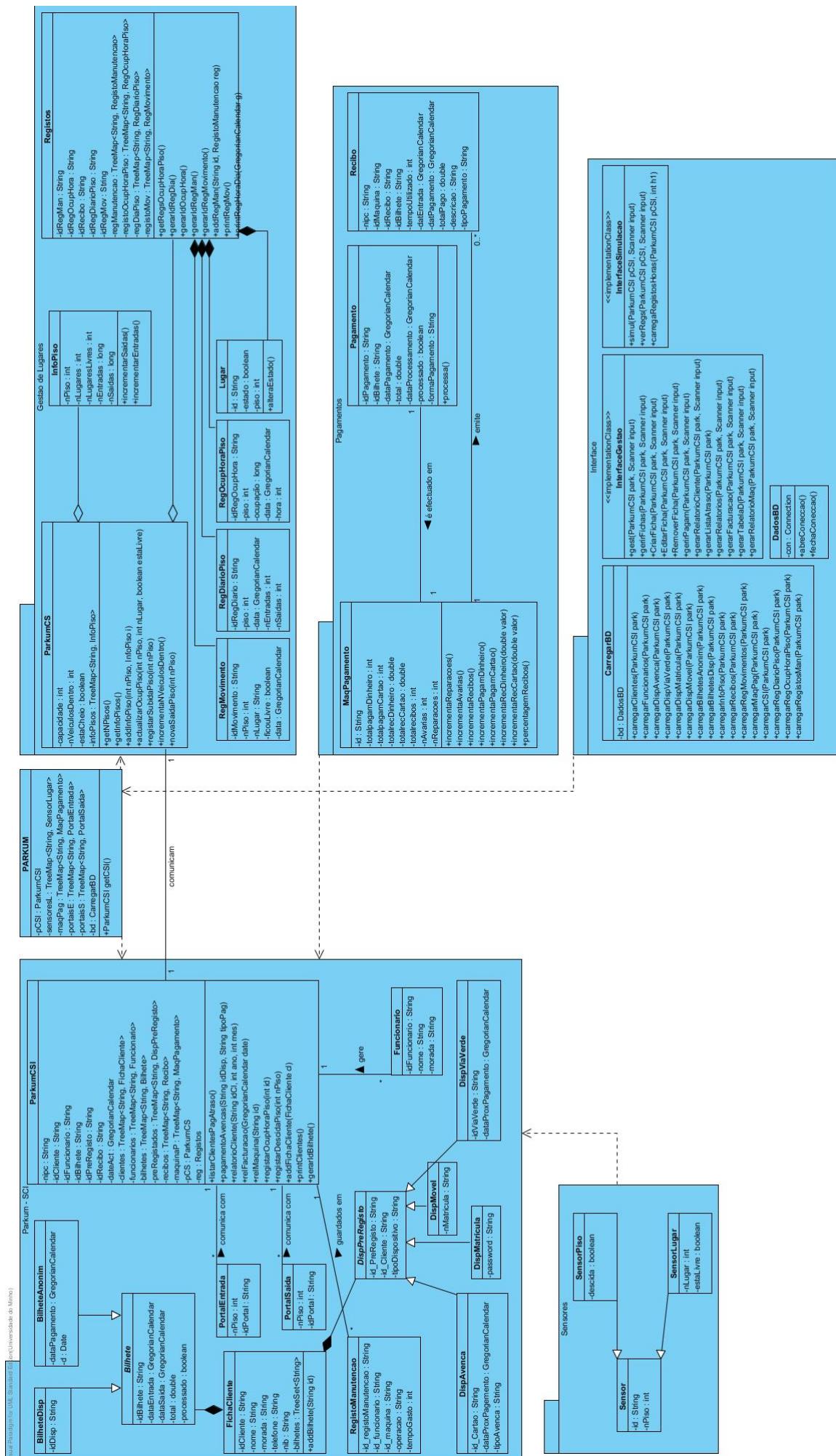


**Figura 2.107:** Diagrama de Estado do portal de saída

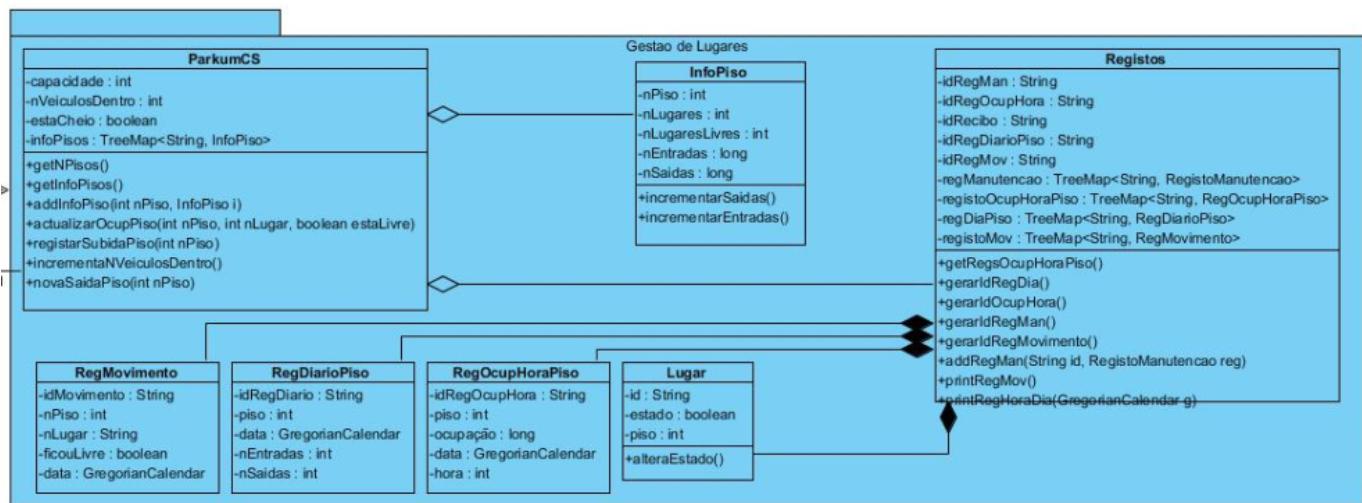
## 2.2 Solução do problema

Após a criação dos diagramas de use case, sequencia, actividade e estado estamos perfeitamente aptos a começar a implementar o software necessário para o **PARKUM**.

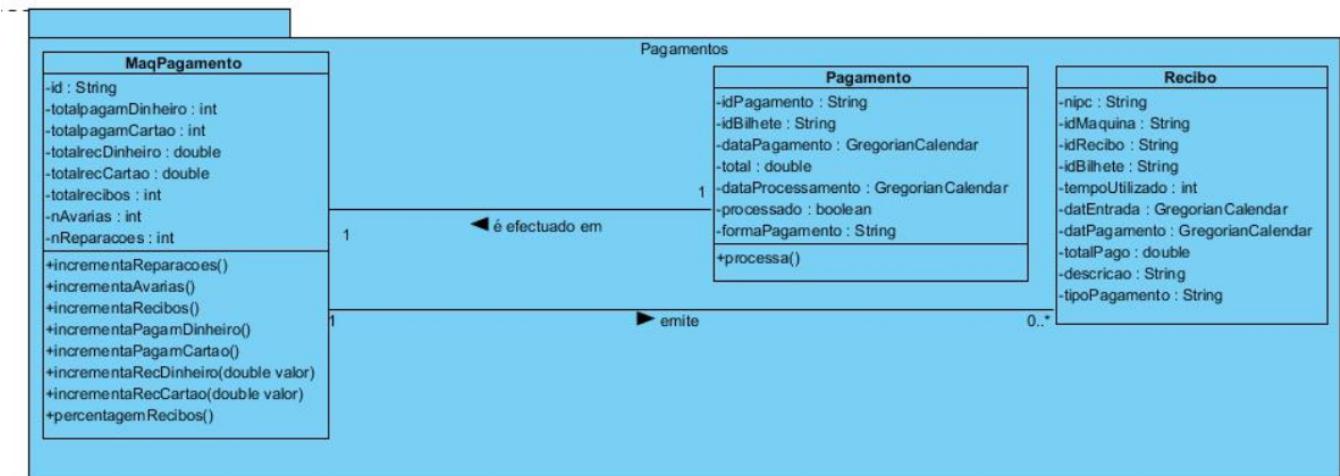
No entanto apesar de já termos muitos métodos definidos nos diagramas de sequência e de termos noção de muitas classes a usar, devemos criar o diagrama de classes deste e estrutural da melhor forma possível de forma a facilitar a implementação do nosso projeto.



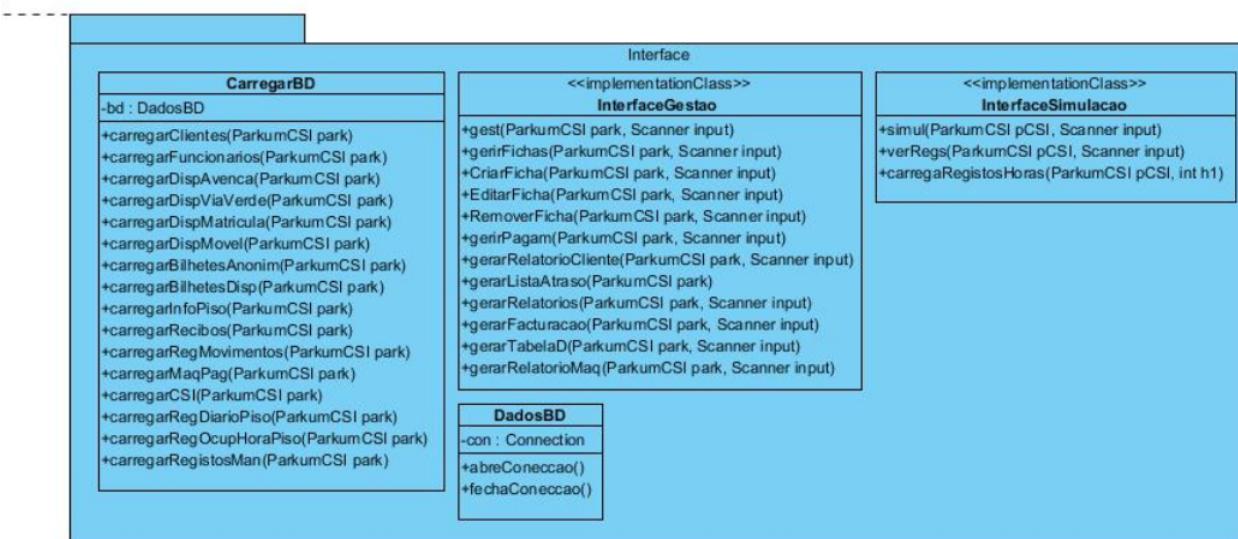
**Figura 2.108:** Diagrama de Class



**Figura 2.109:** Diagrama de Class package gestão de lugares



**Figura 2.110:** Diagrama de Class package pagamentos



**Figura 2.111:** Diagrama de Class package interface

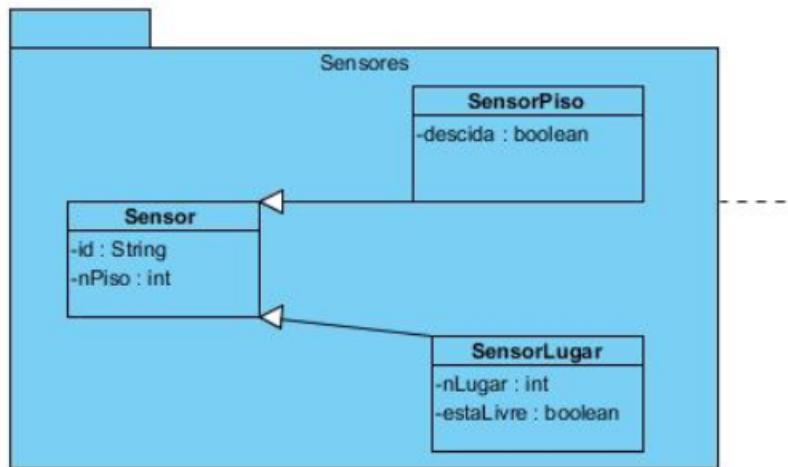


Figura 2.112: Diagrama de Class package sensores

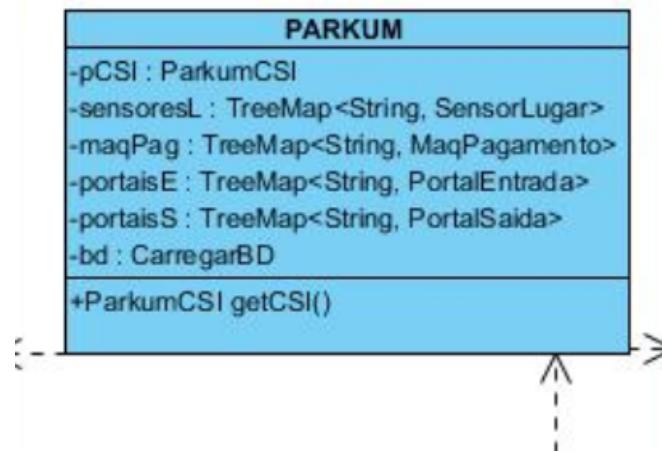


Figura 2.113: Diagrama de Class package Parkum

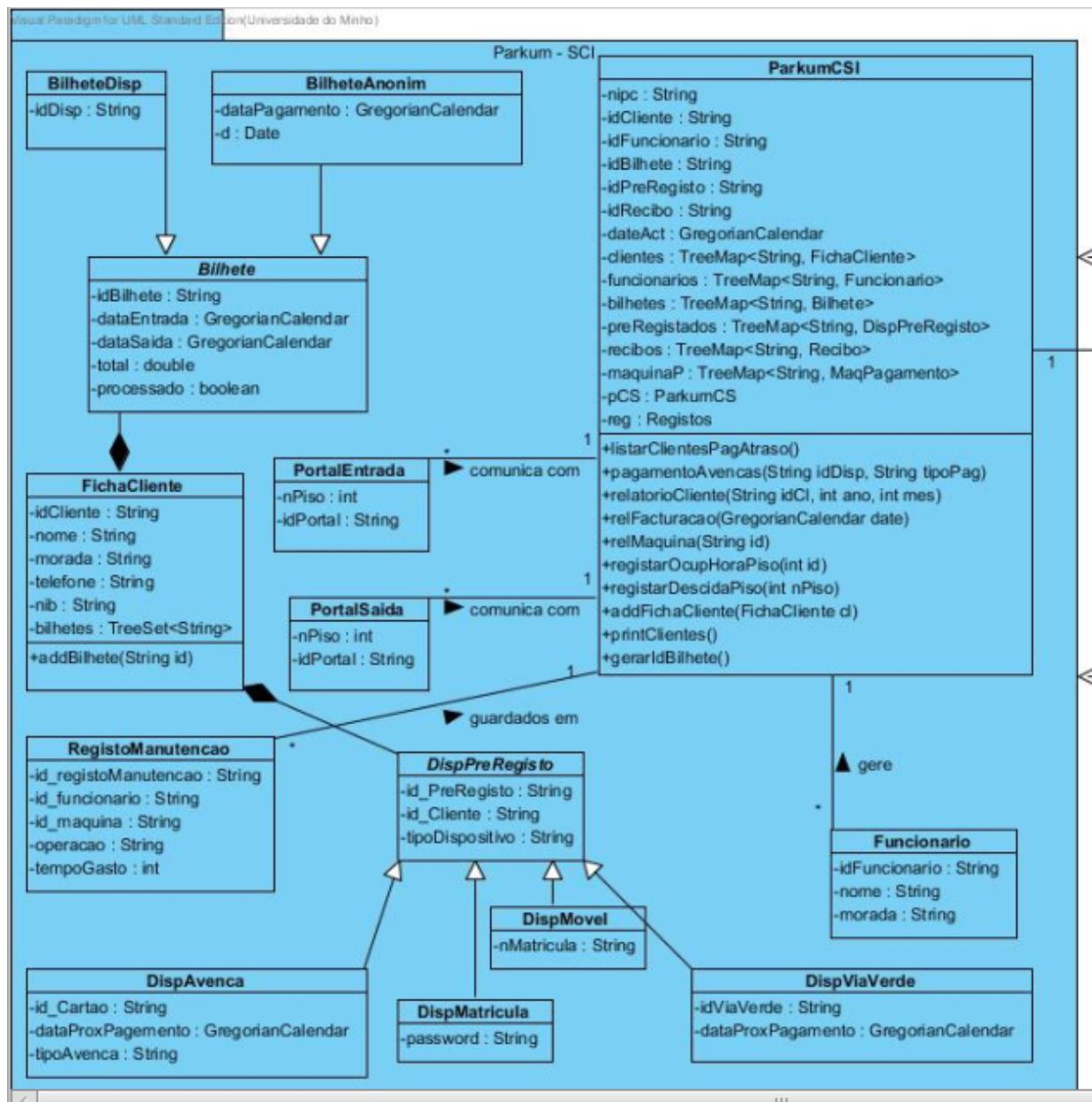


Figura 2.114: Diagrama de Class package Parkum

Como podemos ver possuimos varias class's relativas a diferentes componentes do software. No entanto as classes mais importantes serão o ParkumCsi e o ParkumCs, pois são estas que guardam os registo para a simulação pedida no enunciado.

Sendo que a aplicação se vai centralizar fundamentalmente na simulação de registo já presentes na base de dados os principais métodos estão centralizados nas classes relativas às estatísticas de dados.

### 2.2.1 Base de dados persistente

Já tendo praticamente todas as funcionalidades do sistema criadas, falta guardar os dados do sistema para que possam ser utilizados sempre que se pretenda. Optamos por criar uma base de dados em ORACLE.

Apesar de termos usado uma plataforma para o desenvolvimento da base de dados também poderíamos ter usado ObjectStream visto que apenas se vai destinar a simular a aplicação. No entanto e de forma a ficar o mais próximo de uma aplicação com uso real e de forma a tornar o acesso aos dados melhor, optamos por uma base de dados em ORACLE que irá guardar todas as informações e registo do sistema.

### 2.2.2 Interface

A interface do sistema e a interface para a parte de simulação foram criadas em modo texto, tal como é recomendado no enunciado.

Na interface permitimos todas as operações do funcionamento do sistema, mais as operações para simulação deste.

### 2.2.3 Manual de utilização

A aplicação possui duas interfaces, uma de gestão e uma de simulação de actividades.

A interface de gestão possui funcionalidades para gestão de clientes tais como criação de uma ficha de cliente, a edição de uma ficha ou ainda a remoção de uma ficha. Permite também a criação de relatórios mensais. Depois de introduzido o ano e o mês pretendido é devolvido o número de entradas no parque nesse mês, o tempo médio que o cliente esteve estacionado e ainda o total gasto nesse mês em estacionamentos. Permite a apresentação da lista de clientes em atraso. Permite gestão de pagamentos em dois tipos, via verde e cartões de avença. Por fim, nesta interface é ainda possível gerar relatórios de vários tipos. Relatórios de entradas e saídas do parque, que depois de introduzida uma data nos permitem ver o número de entradas e saídas para cada piso, o número total de entradas e saídas e ainda o número de veículos que se encontram dentro do parque. Relatório sobre o total facturado num determinado dia, para os diferentes tipos de pagamentos. Relatórios sobre as distribuições de lugares em cada piso para cada hora. Por fim relatórios sobre as máquinas de pagamentos. A interface de simulação permite operações como carregar registos entre duas horas do dia e também ver o estado dos vários registos da aplicação tais como: clientes, funcionários, dispositivos pré-registrados, bilhetes, informações dos pisos, recibos emitidos, registos de movimentações nos lugares, registo hístico de cada piso e o estado das máquinas de pagamento.

```

1
-----Sistema de Gestão-----
1 - Gerir Fichas de Clientes
2 - Gerar relatórios de utilização de clientes
3 - Lista clientes com pagamentos em atraso
4 - Processar pagamentos
5 - Gerar Relatórios de controlo e gestão
0 - Sair

```

**Figura 2.115:** Opções de interface gráfica 1

```

1
-----Gerir Fichas de Cliente-----
1 - Criar Ficha
2 - Editar Ficha
3 - Eliminar Ficha
0 - Sair
|
```

**Figura 2.116:** Opções de interface gráfica 2

```

1
-----Nova Ficha Cliente-----
Nome: cliente x
Morada: rua do carmo
Telefone: 9111111111
NIB: 2554199281

```

Figura 2.117: Opções de interface garfica 3

```

-----Editar Ficha Cliente-----
Introduza id ficha: 1

Id: 1
Nome: Clientel
Morada: rua a
Telefone: 912345678
NIB: 12345

1-alterar nome
2-alterar morada
3-alterar telefone
4-alterar nib
0-sair

```

Figura 2.118: Opções de interface garfica 4

```

-----Remover Ficha Cliente-----
Introduza id ficha: 1
Ficha removida com sucesso!

```

Figura 2.119: Opções de interface garfica 5

```

-----Gerar relatórios de Utilização de Cliente-----
Introduza id cliente: 2
Introduza Ano:2010
Introduza Mes:9
id: b3

-----Relatório de Utilização de Cliente2-----
Nº entradas: 1
Tempo médio de estacionamento: 135
Valor mensal: 1.35€

```

Figura 2.120: Opções de interface garfica 6

```

-----Lista Clientes com Pagamentos em Atraso-----
Id Cliente: 1

```

Figura 2.121: Opções de interface garfica 7

```

4
-----Processar Pagamentos-----
1 - Processar Pagamentos de Avença
2 - Processar Pagamentos de Via-Verde
0 - Sair

```

Figura 2.122: Opções de interface garfica 8

```

5
-----Relatórios de Controlo e Gestão-----
1 - Gerar relatório diário de entradas e saídas
2 - Gerar relatório diário de facturação
3 - Gerar tabelas de distribuições de lugares
4 - Gerar relatório de máquina automática
0 - Sair

```

Figura 2.123: Opções de interface garfica 9

```

2
-----Simulação-----
1 - Carregar registos entre 2 horas
2 - Ver estado dos componentes
0 - Sair

```

Figura 2.124: Opções de interface garfica 10

```

2
-----Ver Registro-----
1 - Clinetes
2 - Funcionarios
3 - Dispositivos Registados
4 - Bilhetes
5 - Informacao Piso
6 - Recibos
7 - Registros de estacionamento e saídas
8 - Ver Registros Hora Piso
9 - Ver Estado Maquinas Pagamento
0 - Sair

```

Figura 2.125: Opções de interface garfica 11

```

-----Parkum-----
1 - Gestao
2 - Simulação
0 - Sair

```

Figura 2.126: Opções de interface garfica 12

## 2.3 Ferramentas utilizadas

Para a criar deste projecto usamos as seguintes ferramentas:

- Visual Paradigma -Modelação em UML
- neat beans - Escrita de código JAVA
- geany - escrita do relatório em latex
- Oracle - desenvolvimento da base de dados

# Capítulo 3

## Conclusão

Após a criação do software pedido para o parque de estacionamento(**PARKUM**) seguindo **RUP** algumas conclusões tornaram-se obvias.

Em primmeiro lugar temos de admitir que usando o **RUP** e usando **UML** para a moleção do sistema são optimos e facilitam muito o trabalho de captura de requisitos.

Com a fase de captura e analise de requisitos conseguimos sem menor fundo de duvida ter noção de tudo o que o sistema vai necessitar, sendo o modelo de domínio o maior responsavel por isso.

No entanto apenas com o modelo de domínio só temos uma noção do que o sistema vai necessitar de forma generica e não de uma forma concreta. Foi então que a criação dos diagsmas de use case e posteriormente diagramas de sequencia e de actividade que conseguimos verdadeiramente ter uma visão detalhada das necessidades do sistema.

Com os diagsmas de use case conseguimos ter noção dos verdadeiras utilizadores e das suas acções no sistema, no entanto ainda não temos a total noção de todo o sistema de forma minuciosa.

A partir da descrição textual dos diagsmas de use case elaboramos os diagsmas de sequencia, e com estes conseguimos agora sim ter uma excelente noção do funcionamento do sistema.

Após feitas refinações dos diagramas, conseguimos perceber quais os sub-sistemas do sistema principal e ao mesmo tempo podemos dizer que a informação resultante da elaboração dos diagramas de sequencia já suficiente para "programar"o software. No entanto e apesar de já ser-mos capazes de cirar o software ainda devemos "cirar"mais informação para o desenvolvimento do software para ter noção de quais as Class's a desenvolver no sistema.

É obvio que o diagrama de Class é muito importante e ajuda imenso na fase de implementação, pois dá-nos a noção das Calss's a usar e dos atributos que estas necessitam.

Neste momento podemos dizer e de forma intuitiva que a utilização do **RUP** com linha guia é fundamental na criação de software,e que tudo o que fazemos deve ter uma linha orientação, e nada deve resultar da intuição e sorte, ou se de certa forma resultar provavelmente o risco a erros será muito maior.

A nível pessoal pessoal também foram tiradas algumas conclusões unâmimes pelos elementos do grupo. Sempre que se desenvolva um projecto com dimenões razoaveis e não só devemos inicialmente focarnos na analise e captura de requisitos de forma eficiente e só depois passar a sua implemtação de forma a facilitar e tornar um bom trabalho.

# Bibliografia

- [1] W.Amblor, Scott: The Elements of UML2. Style. Cambrige University Press (2005)
- [2] [www.whitakerparkingsystems.com](http://www.whitakerparkingsystems.com)
- [3] [www.omnitecsecurity.com](http://www.omnitecsecurity.com)
- [4] [www.idonic.com](http://www.idonic.com)

## **Capítulo 4**

# **ANEXOS**

**ANEXOS**

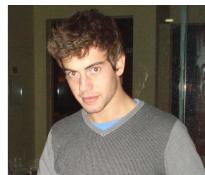
# GRUPO 50



**Figura 4.1:** André Pimenta - a54745



**Figura 4.2:** Cedric Pimenta - a54808



**Figura 4.3:** Daniel Santos - a54825



**Figura 4.4:** João Miguel - a54738



**Figura 4.5:** Milton Nunes - a54802