

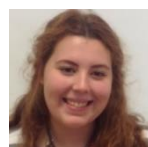
**Universidade do Minho**  
Escola de Engenharia

Desenvolvimento de Sistemas de Software, DSS

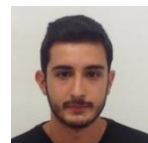
## **Relatório do projeto prático**

### **Gestão de despesas num apartamento**

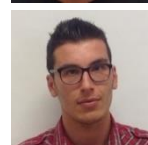
#### **Grupo de Trabalho 38**



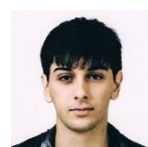
Ana Esmeralda Fernandes A74321



Diogo Alexandre Machado A75399



Miguel Dias Miranda A74726



Rui Filipe Castro Leite A75551

Mestrado Integrado em Engenharia Informática

dezembro de 16



**Universidade do Minho**  
Escola de Engenharia

**Desenvolvimento de Sistemas de Software**  
Relatório do trabalho prático



# Índice

|   |    |
|---|----|
| Introdução                                | 7  |
| 1. Modelo de Domínio                      | 8  |
| 1.1. Breve descrição do problema          | 8  |
| 1.2. Descrição do modelo de Domínio       | 8  |
| 1.3. Apresentação do modelo               | 9  |
| 2. Use Cases                              | 10 |
| 2.1. Diagrama de Use Cases                | 11 |
| 2.2. Especificação dos Use Cases          | 12 |
| 2.2.1. Use Cases comuns a todos os atores | 12 |
| 2.2.2. Use Cases de Utilizador            | 14 |
| 2.2.3. Use Cases de Inquilino             | 14 |
| 2.2.4. Use Cases de Administrador         | 18 |
| 2.2.5. Use Cases adicionais               | 22 |
| 3. Mockups                                | 23 |
| 3.1. Login                                | 23 |
| 3.2. Registar administrador               | 23 |
| 3.3. Página de início do administrador    | 24 |
| 3.4. Página de início do inquilino        | 24 |
| 3.5. Associar categorias a inquilino      | 25 |
| 3.6. Registar inquilino                   | 25 |
| 3.7. Editar utilizador                    | 26 |
| 3.8. Depositar valor                      | 26 |
| 3.9. Remover inquilino                    | 26 |
| 3.10. Registar categoria                  | 27 |
| 3.11. Registar despesa                    | 27 |
| 3.12. Lista de despesas                   | 28 |
| 3.13. Lista de inquilinos                 | 29 |
| 3.14. Pagar despesa                       | 29 |
| 3.15. Pagar prestação                     | 30 |
| 3.16. Mockups de avisos e confirmações    | 30 |
| 4. Diagramas de Máquinas de Estado        | 31 |
| 4.1. Geral                                | 31 |



|  |    |
|--|----|
| 4.2. Administrador                                 | 32 |
| 4.3. Inquilino                                     | 33 |
| 5. Definição dos subsistemas                       | 34 |
| 6. Diagramas de Sequência com subsistemas          | 35 |
| 6.1. Autenticar                                    | 35 |
| 6.2. Fechar sessão                                 | 35 |
| 6.3. Editar utilizador                             | 36 |
| 6.4. Obter lista de despesas                       | 36 |
| 6.5. Consultar saldo                               | 36 |
| 6.6. Depositar quantia                             | 37 |
| 6.7. Consultar movimentos                          | 37 |
| 6.8. Registar inquilino                            | 37 |
| 6.9. Registar categoria                            | 38 |
| 6.10. Obter lista dos Inquilinos                   | 38 |
| 6.11. Associar categorias a inquilino              | 38 |
| 6.12. Remover inquilino                            | 39 |
| 6.13. Registar despesa                             | 39 |
| 6.14. Pagar despesa                                | 40 |
| 6.15. Pagar prestação                              | 40 |
| 6.16. Gerar valores de pagamento                   | 41 |
| 6.17. Registar administrador                       | 41 |
| 7. Diagrama de Package                             | 42 |
| 8. Diagrama de Classes                             | 43 |
| 9. Diagramas de Sequência para a lógica de negócio | 46 |
| 9.1. Pagar despesa                                 | 46 |
| 9.2. Registar inquilino                            | 47 |
| 9.3. Registar Categoria                            | 47 |
| 10. Diagrama de Instalação                         | 48 |
| 11. Implementação da base de dados                 | 49 |
| Conclusão  | 50 |

## Índice de figuras

|  |    |
|--|----|
| Figura 1 – Modelo de Domínio.....                            | 9  |
| Figura 2 – Diagrama de Use Cases.....                        | 11 |
| Figura 3 – Use Case “Autenticar” .....                       | 12 |
| Figura 4 – Use Case “Fechar sessão” .....                    | 12 |
| Figura 5 – Use Case “Editar Utilizador” .....                | 13 |
| Figura 6 – Use Case “Obter lista das despesas” .....         | 13 |
| Figura 7 – Use Case “Registar Administrador” .....           | 14 |
| Figura 8 – Use Case “Depositar quantia” .....                | 14 |
| Figura 9 – Use Case “Consultar movimentos” .....             | 15 |
| Figura 10 – Use Case “Consultar saldo” .....                 | 15 |
| Figura 11 – Use Case “Pagar despesa” .....                   | 16 |
| Figura 12 – Use Case “Pagar prestação” .....                 | 17 |
| Figura 13 – Use Case “Registar inquilino” .....              | 18 |
| Figura 14 – Use Case “Registar categoria” .....              | 18 |
| Figura 15 – Use Case “Remover Inquilino” .....               | 19 |
| Figura 16 – Use Case “Obter lista dos Inquilinos” .....      | 19 |
| Figura 17 – Use Case "Associar categorias a Inquilino" ..... | 20 |
| Figura 18 – Use Case “Registar despesa” .....                | 21 |
| Figura 19 – Use Case “Gerar valores de pagamento” .....      | 22 |
| Figura 20 – Mockup “Login” .....                             | 23 |
| Figura 21 – Mockup "Registar administrador" .....            | 23 |
| Figura 22 – Mockup “Página de início do administrador” ..... | 24 |
| Figura 23 – Mockup “Página de início do inquilino” .....     | 24 |
| Figura 24 – Mockup “Associar categorias a inquilino” .....   | 25 |
| Figura 25 – Mockup “Registar inquilino” .....                | 25 |
| Figura 26 – Mockup “Editar utilizador” .....                 | 26 |
| Figura 27 – Mockup “Depositar valor” .....                   | 26 |
| Figura 28 – Mockup “Remover inquilino” .....                 | 26 |
| Figura 29 – Mockup “Registar categoria” .....                | 27 |
| Figura 30 – Mockup “Registar despesa extraordinária” .....   | 27 |
| Figura 31 – Mockup “Registar despesa recorrente” .....       | 28 |
| Figura 32 – Mockup “Lista de despesas” .....                 | 28 |
| Figura 33 – Mockup “Lista de inquilinos” .....               | 29 |
| Figura 34 – Mockup “Pagar despesa” .....                     | 29 |
| Figura 35 – Mockup “Pagar prestação” .....                   | 30 |
| Figura 36 – Mockup “Sair” .....                              | 30 |
| Figura 37 – Mockup “Passwords não coincidem” .....           | 30 |
| Figura 38 – Mockup “Despesa já registada” .....              | 30 |
| Figura 39 – Mockup "Sem saldo" .....                         | 30 |

|  |    |
|--|----|
| Figura 40 – Diagrama de Máquina de Estado Geral.....   | 31 |
| Figura 41 – Diagrama de Máquina de Estado do Administrador.....  | 32 |
| Figura 42 – Diagrama de Máquina de Estado do Inquilino.....  | 33 |
| Figura 43 – Modelo de Domínio com subsistemas .....  | 34 |
| Figura 44 – Diagrama de Sequência referente ao Use Case “Autenticar” .....                                 | 35 |
| Figura 45 – Diagrama de Sequência referente ao Use Case “Fechar sessão” .....                              | 35 |
| Figura 46 – Diagrama de Sequência referente ao Use Case “Editar utilizador” .....                          | 36 |
| Figura 47 – Diagrama de Sequência referente ao Use Case “Obter lista de despesas” .....                    | 36 |
| Figura 48 – Diagrama de Sequência referente ao Use Case “Consultar saldo” .....                            | 36 |
| Figura 49 – Diagrama de Sequência referente ao Use Case “Depositar quantia” .....                          | 37 |
| Figura 50 – Diagrama de Sequência referente ao Use Case “Consultar movimentos” .....                       | 37 |
| Figura 51 – Diagrama de Sequência referente ao Use Case “Registar inquilino” .....                         | 37 |
| Figura 52 – Diagrama de Sequência referente ao Use Case “Registar categoria” .....                         | 38 |
| Figura 53 – Diagrama de Sequência referente ao Use Case “Obter lista dos Inquilinos” .....                 | 38 |
| Figura 54 – Diagrama de Sequência referente ao Use Case “Associar categorias a inquilino” .....            | 38 |
| Figura 55 – Diagrama de Sequência referente ao Use Case “Remover inquilino” .....                          | 39 |
| Figura 56 – Diagrama de Sequência referente ao Use Case “Registar despesa” .....                           | 39 |
| Figura 57 – Diagrama de Sequência referente ao Use Case “Pagar Despesa” .....                              | 40 |
| Figura 58 – Diagrama de Sequência referente ao Use Case “Pagar prestação” .....                            | 40 |
| Figura 59 – Diagrama de Sequência referente ao Use Case “Gerar valores de pagamento” .....                 | 41 |
| Figura 60 – Diagrama de Sequência referente ao Use Case “Registar administrador” .....                     | 41 |
| Figura 61 – Diagrama de Package .....  | 42 |
| Figura 62 – Diagrama de Classes .....  | 43 |
| Figura 63 – Diagrama de Classes com DAO's .....  | 44 |
| Figura 64 – Diagrama de Classes com DAO's e métodos .....  | 45 |
| Figura 65 – Diagrama de Sequência para a lógica de negócio referente ao Use Case “Pagar despesa” .....     | 46 |
| Figura 66 – Diagrama de Sequência para a lógica de negócio referente ao Use Case “Registar inquilino” .... | 47 |
| Figura 67 – Diagrama de Sequência para a lógica de negócio referente ao Use Case “Registar Categoria” ..   | 47 |
| Figura 68 – Diagrama de Instalação .....   | 48 |
| Figura 69 – Modelo lógico da Base de Dados implementada .....  | 49 |



## Introdução

Este relatório visa apresentar as decisões tomadas na realização do trabalho prático da Unidade Curricular de Desenvolvimento de Sistemas de Software. Procuramos justificar todas as considerações feitas na formulação do problema, na elaboração do modelo de Domínio, de Use Cases e proposta de interface com o utilizador.

Toda a modulação do problema e a apresentação neste relatório de diagramas, esquemas e especificações, são feitas com recurso à linguagem de modelação UML, abordada na UC.

O documento está estruturado em onze capítulos. No primeiro é descrito o problema e é feita uma proposta de modelo de domínio. São justificadas as inclusões de cada entidade e dos relacionamentos entre elas, conforme os requisitos do problema. No segundo capítulo são detalhados os atores do sistema que se consideraram, bem como a especificação dos Use Cases de cada um. No terceiro são apresentados todos os *Mockups* idealizados. No quarto, os diagramas de Máquina de Estado. No quinto, a definição dos subsistemas, tendo por base o modelo de Domínio. No sexto, o diagrama de Package. No sétimo, os diagramas de Classes, com e sem DAO's e com os vários métodos implementados. No nono os diagramas de sequência para lógica de negócio de alguns Use Cases relevantes. No décimo é apresentado o diagrama de Instalação. E por fim, no décimo primeiro, o modelo lógico da base de dados relacional que suporta a aplicação.



## 1. Modelo de Domínio

O sistema a implementar destina-se a suportar a configuração e gestão da partilha de despesas num apartamento, possivelmente de estudantes.

Neste capítulo são apresentados os requisitos do problema e uma proposta de modelo de Domínio. Os requisitos do sistema tiveram como fundamento a experiência própria dos elementos do grupo e observações de determinadas situações comuns no contexto de um apartamento partilhado por estudantes, bem como a interação com os docentes da UC.

### 1.1. Breve descrição do problema

O problema proposto é desenvolver um sistema para a gestão de despesas num apartamento. Após uma análise das necessidades e de factos reais, percebeu-se a existência dos seguintes conceitos importantes na modulação do problema:

- Inquilinos;
- Despesas;
- Pagamentos;
- Conta corrente.

### 1.2. Descrição do modelo de Domínio

A par da elaboração do modelo de Domínio é necessário estabelecer quais as entidades principais do sistema. Tem-se que um Apartamento é habitado por vários inquilinos e gerido por um administrador. Tanto o administrador como o inquilino são utilizadores da aplicação e têm a si associados um nome, um *username* e uma *password*.

Cada inquilino tem a si associado um período de presença no imóvel (desde o dia em que entrou até ao dia em que saiu) e uma conta corrente, de onde são descontados todos os pagamentos relativos a Despesas, ficando registado como movimentos. Cada um destes possui informação relativa à descrição do movimento, o valor associado e a data em que foi feito. A percentagem paga de uma despesa por cada um dos inquilinos a ela associados não é necessariamente igual.

Existem dois tipos de despesas: recorrentes e extraordinárias. Uma despesa extraordinária tem uma designação e uma Recorrente tem uma categoria, que por sua vez também pode ser caracterizada por uma designação. Cada inquilino paga (ou não) a percentagem de uma despesa recorrente conforme a categoria dela. Ambos os tipos de despesas têm uma data de emissão, uma data limite de pagamento, uma referência de fatura, um estado (paga ou não paga) e um preço.

As despesas extraordinárias podem ser pagas por duas modalidades: com ou sem prestações. Cada prestação possui um valor, um estado e um número que a identifica numa sequência de várias prestações.



### 1.3. Apresentação do modelo

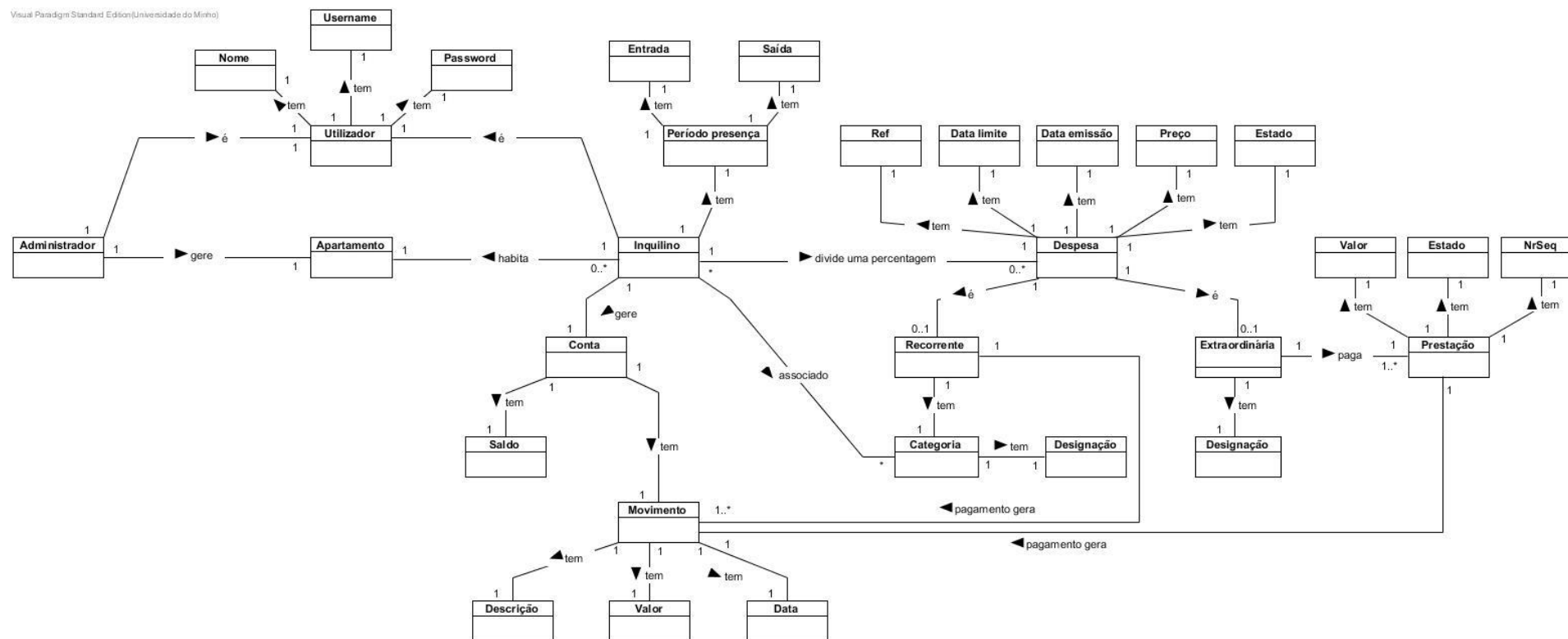


Figura 1 – Modelo de Domínio

## 2. Use Cases

Depois de conseguido o Modelo de Domínio, é necessário pensar-se nos atores que irão interagir com o sistema a desenvolver e os “casos de uso” de cada um. Foram considerados três atores:

- **Inquilino:** habitante do apartamento, responsável por disponibilizar as quantias necessárias para pagar as despesas e por pagá-las;
- **Administrador:** responsável por gerir os inquilinos do apartamento e as respetivas despesas;
- **Utilizador:** designa a entidade que utiliza o sistema, antes de se registar como administrador ou de iniciar sessão.

Por forma a modelar todas a funcionalidades pensadas para a aplicação, foram descritos os seguintes Use Cases:

- **Use cases de Utilizador**
  - Registrar um novo administrador.
- **Use cases comuns a Inquilinos e Administrador**
  - Autenticar;
  - Fechar a sessão;
  - Editar dados de utilizador;
  - Obter a lista das despesas.
- **Use cases de Administrador**
  - Registrar um novo inquilino;
  - Registrar a saída de um inquilino do apartamento;
  - Obter a lista dos inquilinos;
  - Registrar uma categoria;
  - Registrar uma despesa;
  - Associar categorias e percentagens de pagamento a um inquilino.
- **Use cases de Inquilino**
  - Consultar o saldo da sua conta;
  - Depositar uma quantia na sua conta;
  - Consultar os movimentos da sua conta;
  - Pagar uma despesa;
  - Pagar uma prestação de uma despesa.

## 2.1. Diagrama de Use Cases

Visual Paradigm Standard Edition (Universidade do Minho)

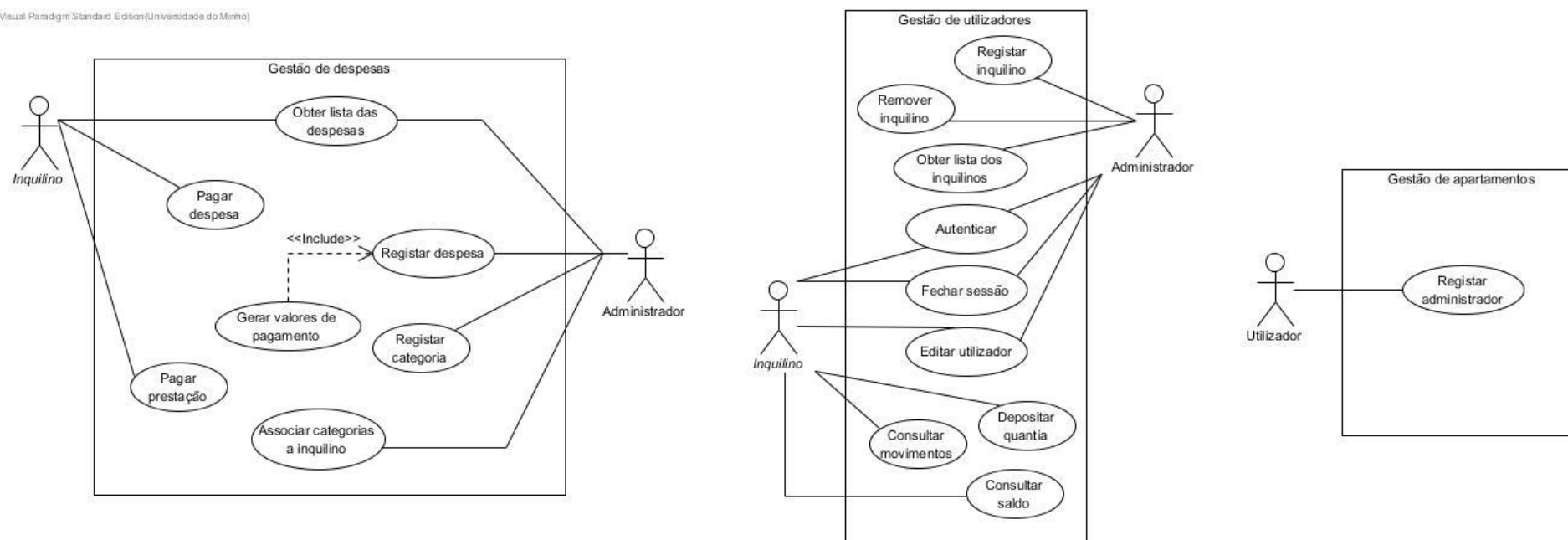


Figura 2 – Diagrama de Use Cases

## 2.2. Especificação dos Use Cases

### 2.2.1. Use Cases comuns a todos os atores

|  |                             |  |
|--|-----------------------------|--|
| <b>Super Use Case</b>                              |                             |  |
| <b>Brief Description</b>                           |                             |  |
| <b>Preconditions</b>                               | Utilizador existe           |  |
| <b>Post-conditions</b>                             | Utilizador fica autenticado |  |
| <b>Flow of Events</b>                              | 1                           | <b>Actor Input</b>   |
|  | 1                           | Fornece nome de utilizador e password                      |
|  | 2                           | Valida nome de utilizador e password                       |
| <b>Exceção 1</b><br>[Dados Inválidos]<br>(passo 1) | 1                           | <b>Actor Input</b>   |
|  | 1                           | Informa que o nome de utilizador ou password são inválidos |
|  |                             |  |

Figura 3 – Use Case “Autenticar”

|  |                                       |   |
|--|---------------------------------------|---|
| <b>Brief Description</b>   |                                       |   |
| <b>Preconditions</b>   | Utilizador autenticado                |   |
| <b>Post-conditions</b>   | Utilizador deixa de estar autenticado |   |
| <b>Flow of Events</b>  | 1                                     | <b>Actor Input</b>                            |
|  | 1                                     | Indica intenção de fechar sessão              |
|  | 2                                     | Pede confirmação da intenção de fechar sessão |
| <b>Alternativa 1</b><br>[Utilizador confirma que pretende fechar sessão]   | 1                                     | <b>Actor Input</b>                            |
|  | 1                                     | Confirma                                      |
|  | 2                                     | Desautentica utilizador                       |
| <b>Alternativa 2</b><br>[Utilizador desiste de fechar sessão]<br>(Passo 3) | 1                                     | <b>Actor Input</b>                            |
|  | 1                                     | Nega confirmação                              |
|  | 2                                     | Cancela operação                              |

Figura 4 – Use Case “Fechar sessão”

|                          |  |                                    |                            |
|--------------------------|--|------------------------------------|----------------------------|
| <b>Super Use Case</b>    |  |                                    |                            |
| <b>Brief Description</b> |  |                                    |                            |
| <b>Preconditions</b>     | Utilizador autenticado                   |                                    |                            |
| <b>Post-conditions</b>   | Utilizador fica com os dados atualizados |                                    |                            |
| <b>Flow of Events</b>    |  | <b>Actor Input</b>                 | <b>System Response</b>     |
|                          | 1  | Indica intenção de editar os dados |                            |
|                          | 2  |                                    | Apresenta informação atual |
|                          | 3  | Fornece dados a alterar            |                            |
|                          | 4  |                                    | Regista alterações         |

Figura 5 – Use Case “Editar Utilizador”

|  |  |  |  |
|--|--|--|--|
| Super Use Case   |  |  |  |
| Brief Description  |  |  |  |
| Preconditions  | Utilizador autenticado                               |  |  |
| Post-conditions  | Utilizador toma conhecimento das despesas registadas |  |  |
| Flow of Events   |  | Actor Input                              | System Response                                      |
|  | 1  | Indica intenção de verificar as despesas |  |
|  | 2  |  | Verifica permissões do utilizador                    |
| Alternativa 1<br>[Utilizador é Administrador]<br>(Passo 2) |  | Actor Input                              | System Response                                      |
|  | 1  |  | Devolve a lista de todas as despesas registadas      |
| Alternativa 2<br>[Utilizador é Inquilino]<br>(Passo 2)     |  | Actor Input                              | System Response                                      |
|  | 1  |  | Devolve a lista das despesas associadas ao inquilino |

Figura 6 – Use Case “Obter lista das despesas”

### 2.2.2. Use Cases de Utilizador

|   |                               |   |  |
|---|-------------------------------|---|--|
| <b>Super Use Case</b>                                   |                               |   |  |
| <b>Brief Description</b>                                |                               |   |  |
| <b>Preconditions</b>                                    | Novo administrador a registar |   |  |
| <b>Post-conditions</b>                                  | Administrador registado       |   |  |
| <b>Flow of Events</b>                                   | ↳                             | <b>Actor Input</b>                              | <b>System Response</b>                             |
|   | 1                             | Manifesta intenção de registar um administrador |  |
|   | 2                             |   | Solicita dados do administrador                    |
|   | 3                             | Fornece username, password e nome               |  |
|   | 4                             |   | Valida dados                                       |
|   | 5                             |   | Regista administrador                              |
|   |                               |   |  |
| <b>Exceção 1</b><br>[Utilizador já existe]<br>(Passo 4) |                               | <b>Actor Input</b>                              | <b>System Response</b>                             |
|   | 1                             |   | Informa que o nome de utilizador já está registado |

Figura 7 – Use Case “Registar Administrador”

### 2.2.3. Use Cases de Inquilino

|                   |                           |   |                                    |
|-------------------|---------------------------|---|------------------------------------|
| Super Use Case    |                           |   |                                    |
| Brief Description |                           |   |                                    |
| Preconditions     | Inquilino autenticado     |   |                                    |
| Post-conditions   | Conta corrente atualizada |   |                                    |
| Flow of Events    |                           | Actor Input   | System Response                    |
|                   | 1                         | Manifesta vontade de depositar um valor na sua conta corrente |                                    |
|                   | 2                         |   | Pede quantia a depositar           |
|                   | 3                         | Fornece montante  |                                    |
|                   | 4                         |   | Regista montante na conta corrente |

Figura 8 – Use Case “Depositar quantia”

|                          |   |  |                        |
|--------------------------|---|--|------------------------|
| <b>Super Use Case</b>    |   |  |                        |
| <b>Brief Description</b> |   |  |                        |
| <b>Preconditions</b>     | Inquilino autenticado                           |  |                        |
| <b>Post-conditions</b>   | Inquilina toma conhecimento dos seus movimentos |  |                        |
| <b>Flow of Events</b>    | 1   | <b>Actor Input</b>   | <b>System Response</b> |
|                          | 1   | Demonstra intenção em consultar movimentos da sua conta corrente |                        |
|                          | 2   |  | Fornece movimentos     |

*Figura 9 – Use Case “Consultar movimentos”*

|                          |   |                                      |                                   |
|--------------------------|---|--------------------------------------|-----------------------------------|
| <b>Super Use Case</b>    |   |                                      |                                   |
| <b>Brief Description</b> |   |                                      |                                   |
| <b>Preconditions</b>     | Inquilino autenticado                             |                                      |                                   |
| <b>Post-conditions</b>   | Inquilino toma conhecimento do saldo da sua conta |                                      |                                   |
| <b>Flow of Events</b>    |   | <b>Actor Input</b>                   | <b>System Response</b>            |
|                          | 1   | Indica intenção de consultar o saldo |                                   |
|                          | 2   |                                      | Fornece o saldo da conta corrente |

*Figura 10 – Use Case “Consultar saldo”*

|   |                            |                                     |   |
|---|----------------------------|-------------------------------------|---|
| <b>Super Use Case</b>   |                            |                                     |   |
| <b>Brief Description</b>  |                            |                                     |   |
| <b>Preconditions</b>  | Inquilino autenticado      |                                     |   |
| <b>Post-conditions</b>  | Parte da despesa fica paga |                                     |   |
| <b>Flow of Events</b>   | 1                          | <b>Actor Input</b>                  | <b>System Response</b>                          |
|   | 1                          | Manifesta intenção de pagar despesa |   |
|   | 2                          |                                     | Fornecer lista das despesas por pagar           |
|   | 3                          | Escolhe despesa                     |   |
|   | 4                          | Confirma intenção de pagar          |   |
|   | 5                          |                                     | Desconta valor da conta e regista movimento     |
|   | 6                          |                                     | Informa sucesso no pagamento                    |
| <b>Exceção 3</b><br>[Inquilino não tem saldo]<br>(Passo 5)      | 1                          | <b>Actor Input</b>                  | <b>System Response</b>                          |
|   | 1                          |                                     | Indica ao utilizador que não tem saldo na conta |
| <b>Alternativa 1</b><br>[Ultimo inquilino a pagar]<br>(Passo 6) |                            | <b>Actor Input</b>                  | <b>System Response</b>                          |
|   | 1                          |                                     | Marca o estado da despesa como paga             |

Figura 11 – Use Case “Pagar despesa”



|   |                       |   |
|---|-----------------------|---|
| <b>Super Use Case</b>                                   |                       |   |
| <b>Brief Description</b>                                |                       |   |
| <b>Preconditions</b>                                    | Inquilino autenticado |   |
| <b>Post-conditions</b>                                  | Prestação paga        |   |
| <b>Flow of Events</b>                                   | 1                     | <b>Actor Input</b>                                  |
|   | 1                     | Demonstra intenção de pagar uma prestação           |
|   | 2                     |   |
|   | 3                     | Escolhe prestação                                   |
|   | 4                     | Desconta valor da prestação e marca-a como paga     |
| <b>Exceção 1</b><br>[Inquilino sem saldo]<br>(Passo 1)  |                       | <b>Actor Input</b>                                  |
|   | 1                     | Informa que não tem saldo suficiente na conta       |
| <b>Alternativa 1</b><br>[Última prestação]<br>(Passo 4) |                       | <b>Actor Input</b>                                  |
|   | 1                     | Regista que utilizador pagou a sua parte da despesa |

Figura 12 – Use Case “Pagar prestação”

#### 2.2.4. Use Cases de Administrador

|  |                           |                                  |  |
|--|---------------------------|----------------------------------|--|
| <b>Super Use Case</b>                                      |                           |                                  |  |
| <b>Brief Description</b>                                   |                           |                                  |  |
| <b>Preconditions</b>                                       | Administrador autenticado |                                  |  |
| <b>Post-conditions</b>                                     | Inquilino fica registado  |                                  |  |
| <b>Flow of Events</b>                                      | 1                         | <b>Actor Input</b>               | <b>System Response</b>                     |
|  | 1                         | Indica nome, username e password |  |
|  | 2                         |                                  | Valida dados                               |
|  | 3                         |                                  | Informa sucesso no registo                 |
| <b>Exceção 1</b><br>[Utilizador já existente]<br>(Passo 2) | 1                         | <b>Actor Input</b>               | <b>System Response</b>                     |
|  | 1                         |                                  | Informa que o utilizador já está registado |

Figura 13 – Use Case “Registar inquilino”

|  |  |                                   |                                 |
|--|--|-----------------------------------|---------------------------------|
| <b>Super Use Case</b>                                  |  |                                   |                                 |
| <b>Brief Description</b>                               |  |                                   |                                 |
| <b>Preconditions</b>                                   | Administrador autenticado                  |                                   |                                 |
| <b>Post-conditions</b>                                 | Categoria de despesa registada com sucesso |                                   |                                 |
| <b>Flow of Events</b>                                  |  | <b>Actor Input</b>                | <b>System Response</b>          |
|  | 1  | Fornece nova categoria a registar |                                 |
|  | 2  |                                   | Regista com sucesso             |
| <b>Exceção 1</b><br>[Categoria já existe]<br>(Passo 2) |  | <b>Actor Input</b>                | <b>System Response</b>          |
|  | 1  |                                   | Informa que categoria já existe |

Figura 14 – Use Case “Registar categoria”

|  |                                |                              |                                    |
|--|--------------------------------|------------------------------|------------------------------------|
| <b>Super Use Case</b>                                    |                                |                              |                                    |
| <b>Brief Description</b>                                 |                                |                              |                                    |
| <b>Preconditions</b>                                     | Administrador está autenticado |                              |                                    |
| <b>Post-conditions</b>                                   | Saída do inquilino registada   |                              |                                    |
| <b>Flow of Events</b>                                    | 1                              | <b>Actor Input</b>           | <b>System Response</b>             |
|  | 1                              | Fornece username             |                                    |
|  | 2                              |                              | Procura inquilino                  |
|  | 3                              |                              | Apresenta informação               |
|  | 4                              | Confirma intenção de remover |                                    |
|  | 5                              |                              | Regista data de saída do inquilino |
| <b>Exceção 1</b><br>[Utilizador não existe]<br>(Passo 2) | 1                              | <b>Actor Input</b>           | <b>System Response</b>             |
|  | 1                              |                              | Indica que inquilino não existe    |
|  |                                |                              |                                    |

Figura 15 – Use Case “Remover Inquilino”

|                          |                           |  |                                |
|--------------------------|---------------------------|--|--------------------------------|
| <b>Super Use Case</b>    |                           |  |                                |
| <b>Brief Description</b> |                           |  |                                |
| <b>Preconditions</b>     | Administrador autenticado |  |                                |
| <b>Post-conditions</b>   | Inquilinos listados       |  |                                |
| <b>Flow of Events</b>    | 1                         | <b>Actor Input</b>                                 | <b>System Response</b>         |
|                          | 1                         | Manifesta intenção de obter a lista dos inquilinos |                                |
|                          | 2                         |  | Apresenta lista dos inquilinos |

Figura 16 – Use Case “Obter lista dos Inquilinos”

|                          |  |  |  |
|--------------------------|--|--|--|
| <b>Super Use Case</b>    |  |  |  |
| <b>Brief Description</b> |  |  |  |
| <b>Preconditions</b>     | Administrador autenticado                |  |  |
| <b>Post-conditions</b>   | O inquilino fica associado às categorias |  |  |
| <b>Flow of Events</b>    | 1  | <b>Actor Input</b>   | <b>System Response</b>                       |
|                          | 1  | Demonstra intenção de associar categorias de despesas a um inquilino |  |
|                          | 2  |  | Apresenta inquilinos registados e categorias |
|                          | 3  | Escolhe o inquilino  |  |
|                          | 4  | Seleciona categorias e fornece percentagem para cada uma             |  |
|                          | 5  | Confirma intenção de associar  |  |
|                          | 6  |  | Associa inquilino às categorias escolhidas   |

*Figura 17 – Use Case "Associar categorias a Inquilino"*

|   |                           |  |  |
|---|---------------------------|--|--|
| <b>Super Use Case</b>   |                           |  |  |
| <b>Brief Description</b>                                      |                           |  |  |
| <b>Preconditions</b>  | Administrador autenticado |  |  |
| <b>Post-conditions</b>  | Despesa fica registada    |  |  |
| <b>Flow of Events</b>   | 1                         | <b>Actor Input</b>   | <b>System Response</b>                                 |
|   | 1                         | Fornece preço, referência e data limite de pagamento                   |  |
|   | 2                         |  | Valida dados da despesa                                |
|   | 3                         | Escolhe tipo da despesa  |  |
|   | 4                         | Confirma registo   |  |
|   | 5                         |  | Regista despesa para os inquilinos                     |
| <b>Exceção 1</b><br>[Despesa já inserida]<br>(passo 2)        | 1                         | <b>Actor Input</b>   | <b>System Response</b>                                 |
|   | 1                         |  | Informa que a referência já está inserida numa despesa |
| <b>Alternativa 1</b><br>[Despesa recorrente]<br>(passo 3)     | 1                         | <b>Actor Input</b>   | <b>System Response</b>                                 |
|   | 1                         |  | <<include>> Gerar valores de pagamento                 |
| <b>Alternativa 2</b><br>[Despesa extraordinária]<br>(passo 3) | 1                         | <b>Actor Input</b>   | <b>System Response</b>                                 |
|   | 1                         | Informa designação   |  |
|   | 2                         |  | Regista designação                                     |
|   | 3                         |  | Fornece lista dos inquilinos registados                |
|   | 4                         | Seleciona inquilinos e a percentagem e número de prestações de cada um |  |
|   | 5                         | Confirma inquilinos  |  |

Figura 18 – Use Case “Registar despesa”

### 2.2.5. Use Cases adicionais

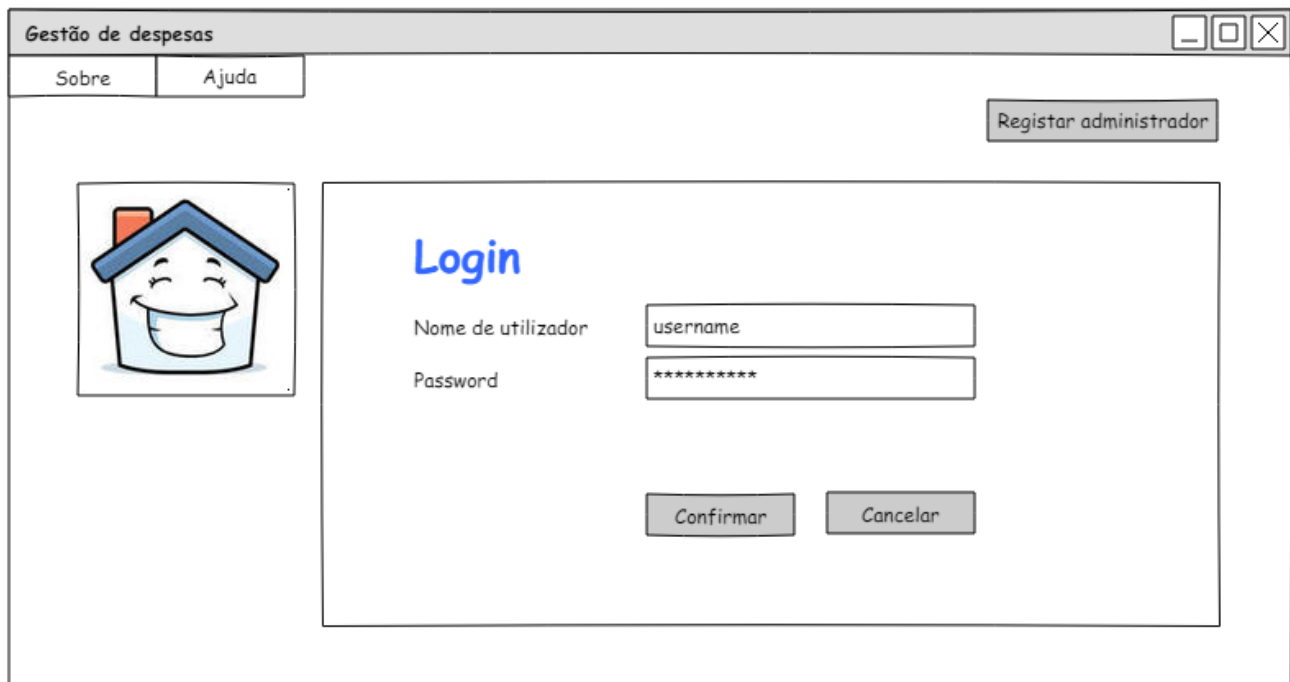
|                          |                                    |                              |   |
|--------------------------|------------------------------------|------------------------------|---|
| <b>Super Use Case</b>    |                                    |                              |   |
| <b>Brief Description</b> |                                    |                              |   |
| <b>Preconditions</b>     | Inquilinos e categorias registadas |                              |   |
| <b>Post-conditions</b>   | Valores são gerados                |                              |   |
| <b>Flow of Events</b>    | 1                                  | <b>Actor Input</b>           | <b>System Response</b>                              |
|                          | 1                                  |                              | Apresenta categorias disponíveis                    |
|                          | 2                                  | Fornece categoria da despesa |   |
|                          | 3                                  |                              | Verifica os utilizadores que pagam a categoria dada |
|                          | 4                                  |                              | Verifica percentagens de pagamento de cada um       |
|                          | 5                                  |                              | Devolve os resultados                               |

*Figura 19 – Use Case “Gerar valores de pagamento”*

### 3. Mockups

Neste capítulo são apresentadas algumas propostas da interface da aplicação. Estes *Mockups* foram melhorados e completados em relação à primeira fase.

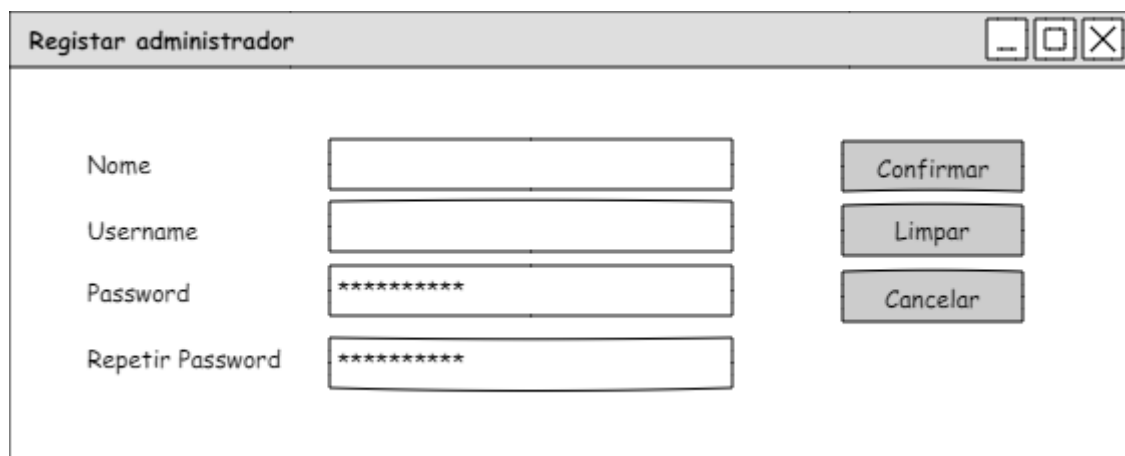
#### 3.1. Login



The mockup shows a window titled "Gestão de despesas" with standard Windows-style window controls (minimize, maximize, close). Below the title bar are two menu items: "Sobre" and "Ajuda". In the top right corner, there is a button labeled "Registrar administrador". On the left side of the main content area, there is a cartoon illustration of a smiling house with a blue roof and a red chimney. To the right of the illustration, the word "Login" is displayed in a large blue font. Below "Login", there are two labels: "Nome de utilizador" and "Password". Each label is followed by a text input field. The "Nome de utilizador" field contains the text "username", and the "Password" field contains eight asterisks "\*\*\*\*\*". At the bottom of the login section, there are two buttons: "Confirmar" and "Cancelar".

Figura 20 – Mockup "Login"

#### 3.2. Registrar administrador



The mockup shows a window titled "Registrar administrador" with standard Windows-style window controls (minimize, maximize, close). The main content area contains four labels on the left: "Nome", "Username", "Password", and "Repetir Password". Each label is followed by a text input field. The "Password" and "Repetir Password" fields contain eight asterisks "\*\*\*\*\*". To the right of the input fields, there are three buttons stacked vertically: "Confirmar", "Limpar", and "Cancelar".

Figura 21 – Mockup "Registrar administrador"

### 3.3. Página de início do administrador

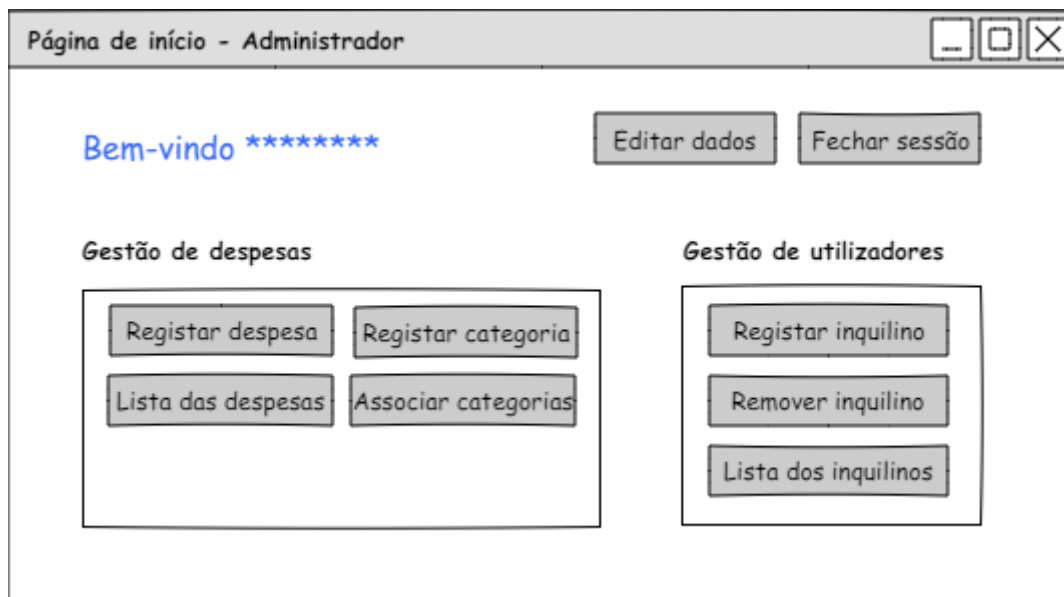


Figura 22 – Mockup “Página de início do administrador”

### 3.4. Página de início do inquilino

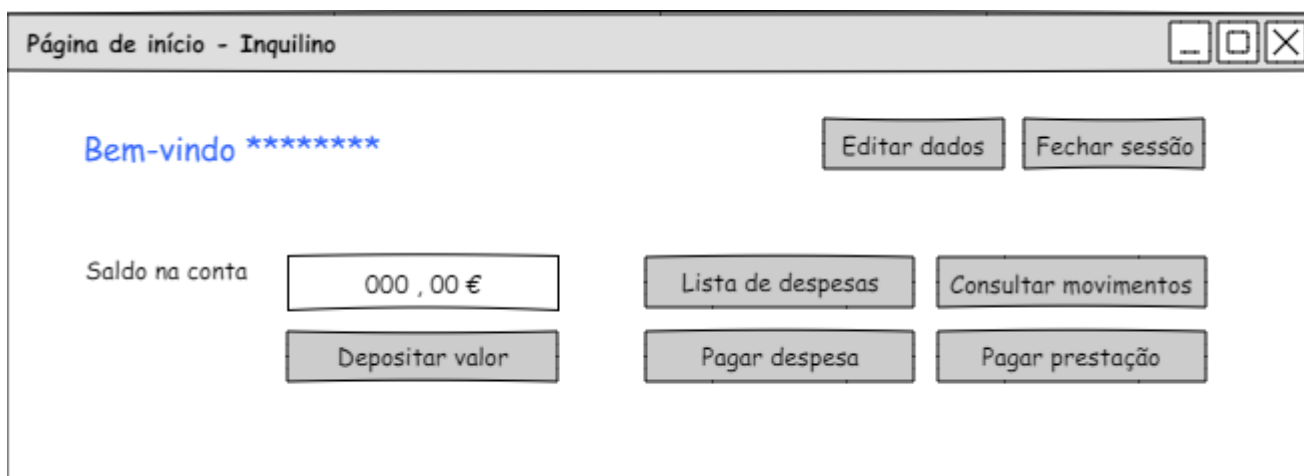
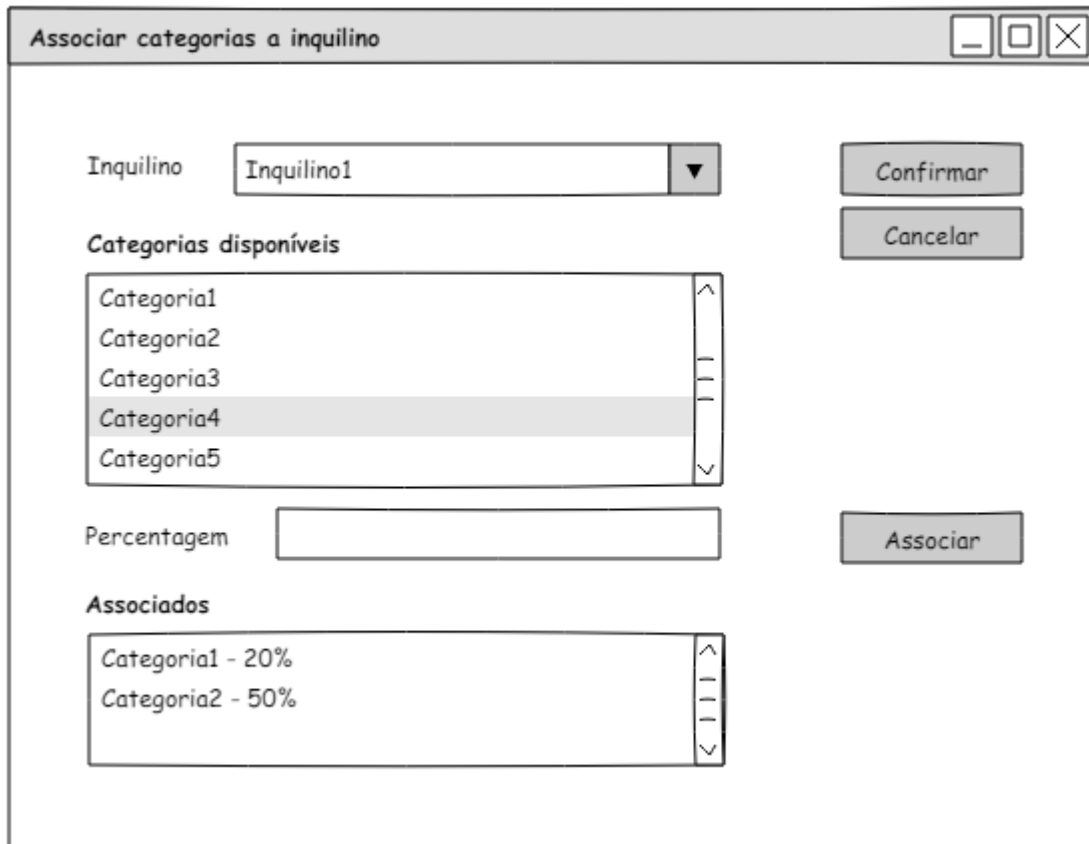


Figura 23 – Mockup “Página de início do inquilino”



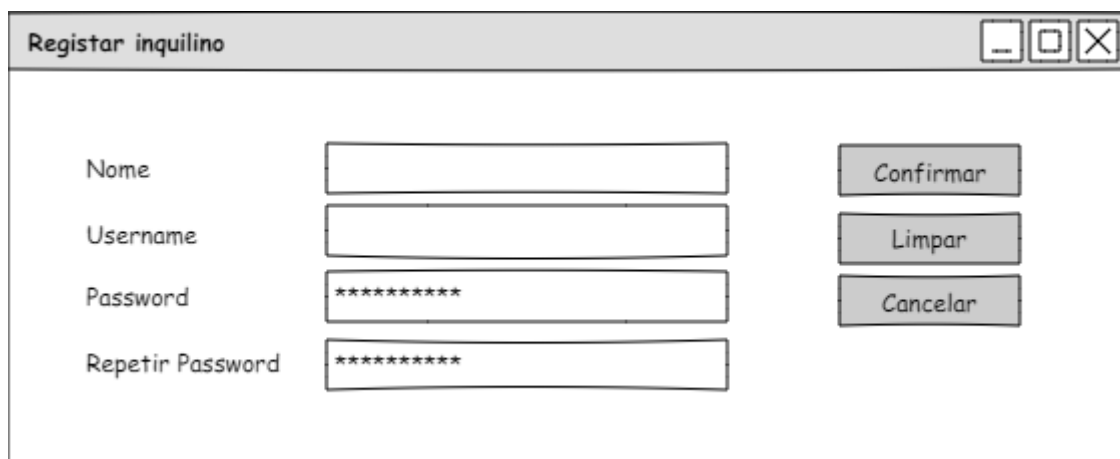
### 3.5. Associar categorias a inquilino



The mockup shows a window titled "Associar categorias a inquilino". It contains a dropdown menu for "Inquilino" with "Inquilino1" selected. Below it is a list box for "Categorias disponíveis" with "Categoria4" selected. To the right of these are "Confirmar" and "Cancelar" buttons. Below the list box is a text input for "Percentagem" and an "Associar" button. At the bottom is a list box for "Associados" showing "Categoria1 - 20%" and "Categoria2 - 50%".

Figura 24 – Mockup “Associar categorias a inquilino”

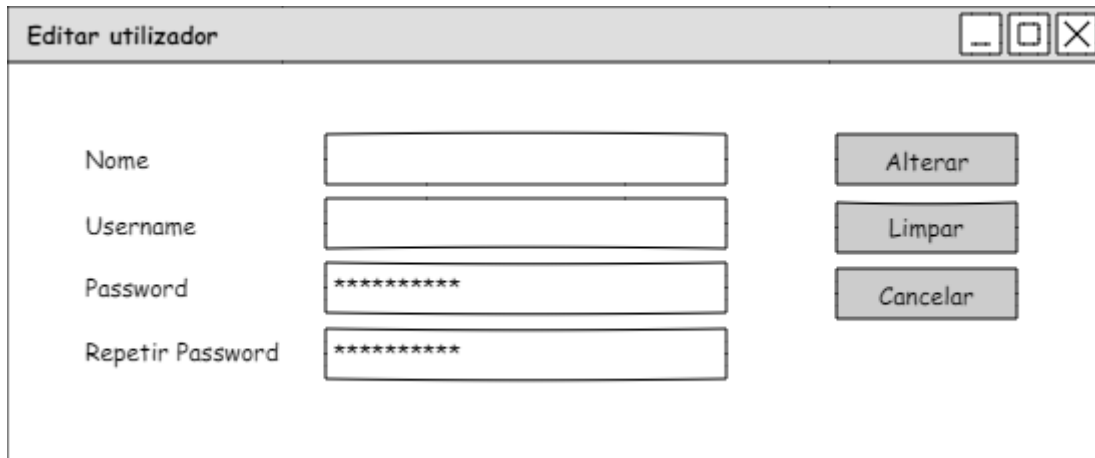
### 3.6. Registrar inquilino



The mockup shows a window titled "Registrar inquilino". It contains four text input fields: "Nome", "Username", "Password" (masked with asterisks), and "Repetir Password" (masked with asterisks). To the right of these fields are three buttons: "Confirmar", "Limpar", and "Cancelar".

Figura 25 – Mockup “Registrar inquilino”

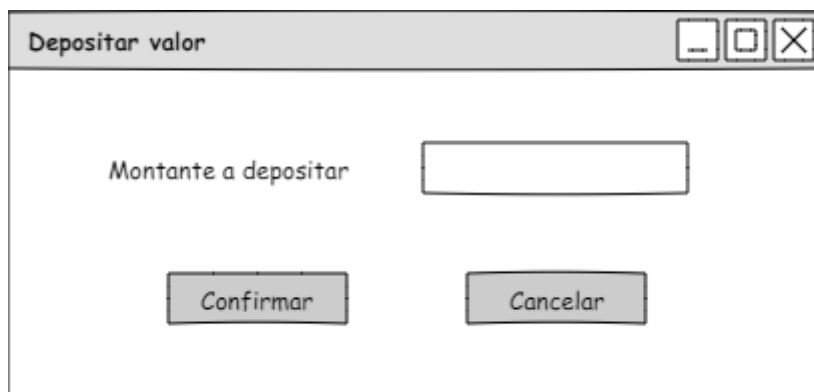
### 3.7. Editar utilizador



The 'Editar utilizador' dialog box features a title bar with standard window controls. It contains four input fields on the left: 'Nome', 'Username', 'Password', and 'Repetir Password'. The 'Password' and 'Repetir Password' fields are masked with asterisks. On the right, there are three buttons: 'Alterar', 'Limpar', and 'Cancelar'.

Figura 26 – Mockup “Editar utilizador”

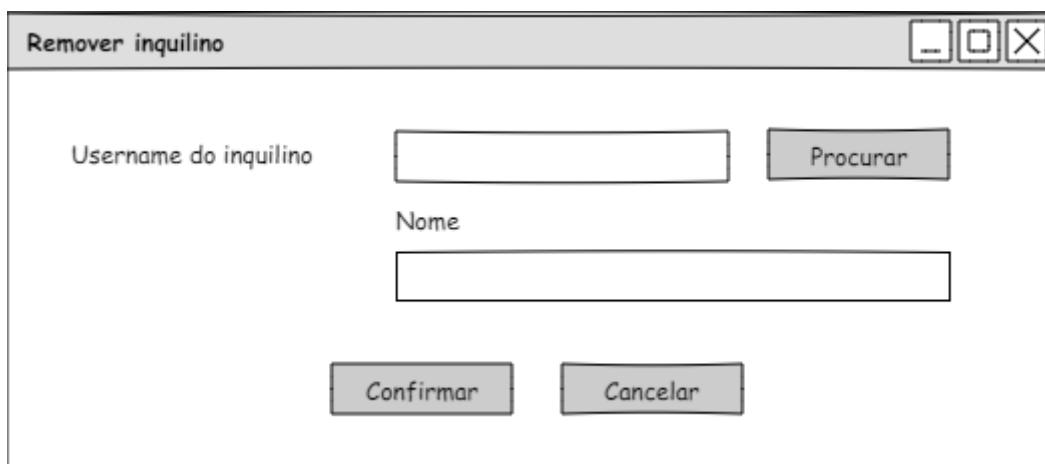
### 3.8. Depositar valor



The 'Depositar valor' dialog box has a title bar with standard window controls. It contains a single input field labeled 'Montante a depositar'. Below the input field are two buttons: 'Confirmar' and 'Cancelar'.

Figura 27 – Mockup “Depositar valor”

### 3.9. Remover inquilino



The 'Remover inquilino' dialog box features a title bar with standard window controls. It contains two input fields: 'Username do inquilino' and 'Nome'. To the right of the 'Username do inquilino' field is a 'Procurar' button. Below the input fields are two buttons: 'Confirmar' and 'Cancelar'.

Figura 28 – Mockup “Remover inquilino”

### 3.10. Registar categoria

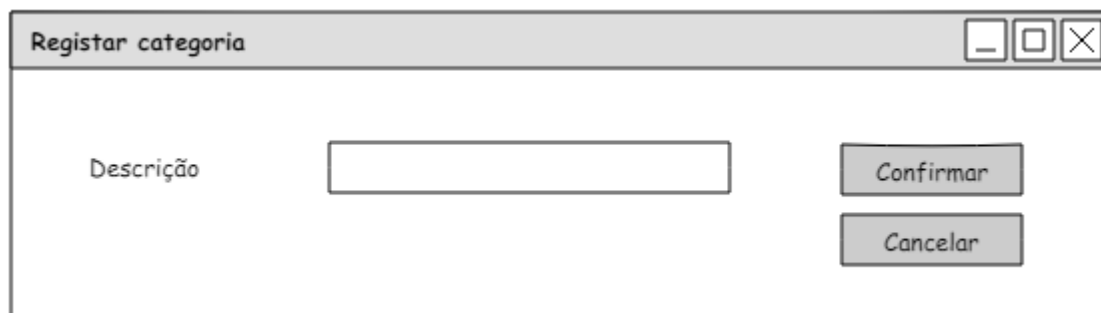


Figura 29 – Mockup “Registar categoria”

### 3.11. Registar despesa

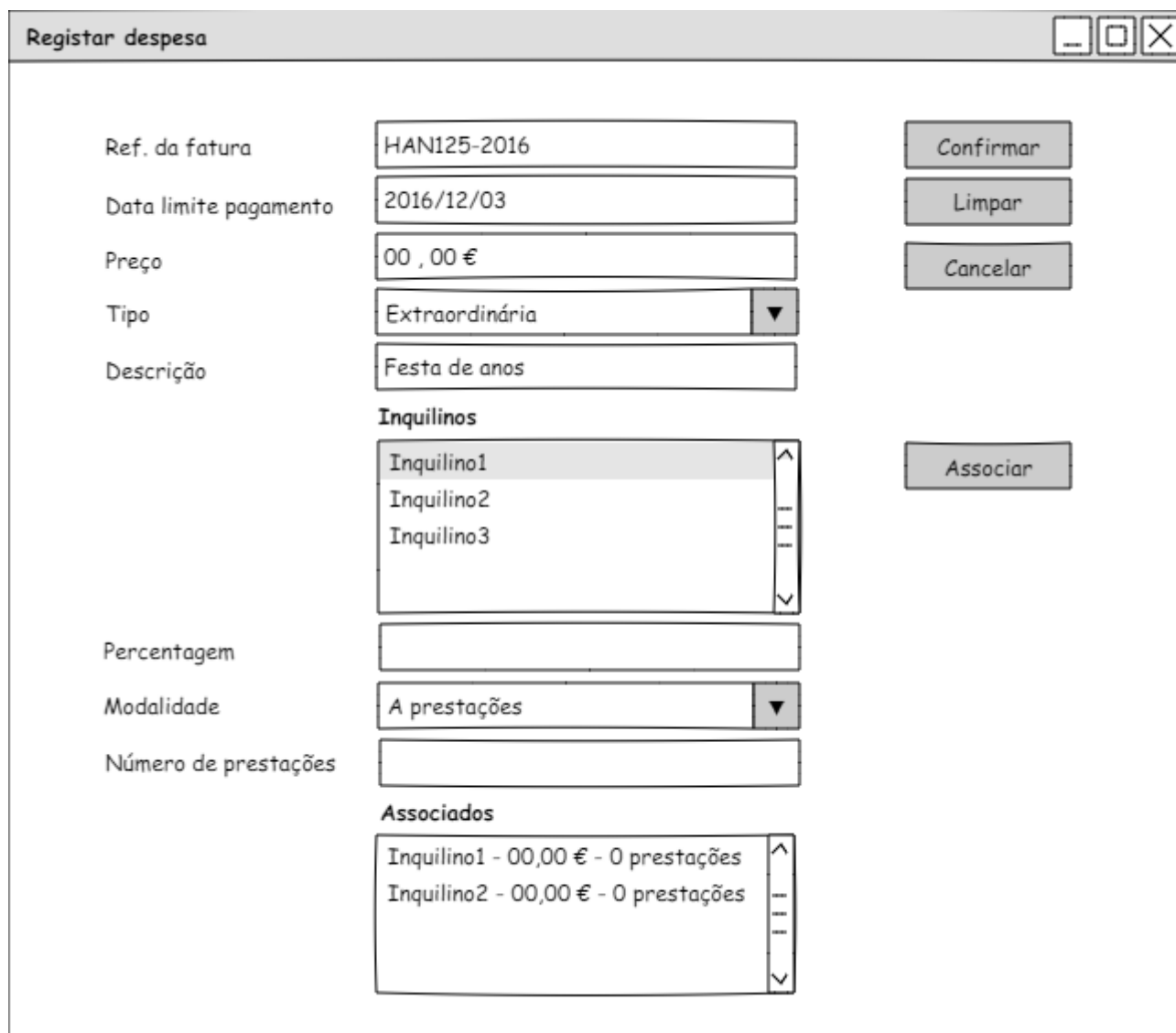
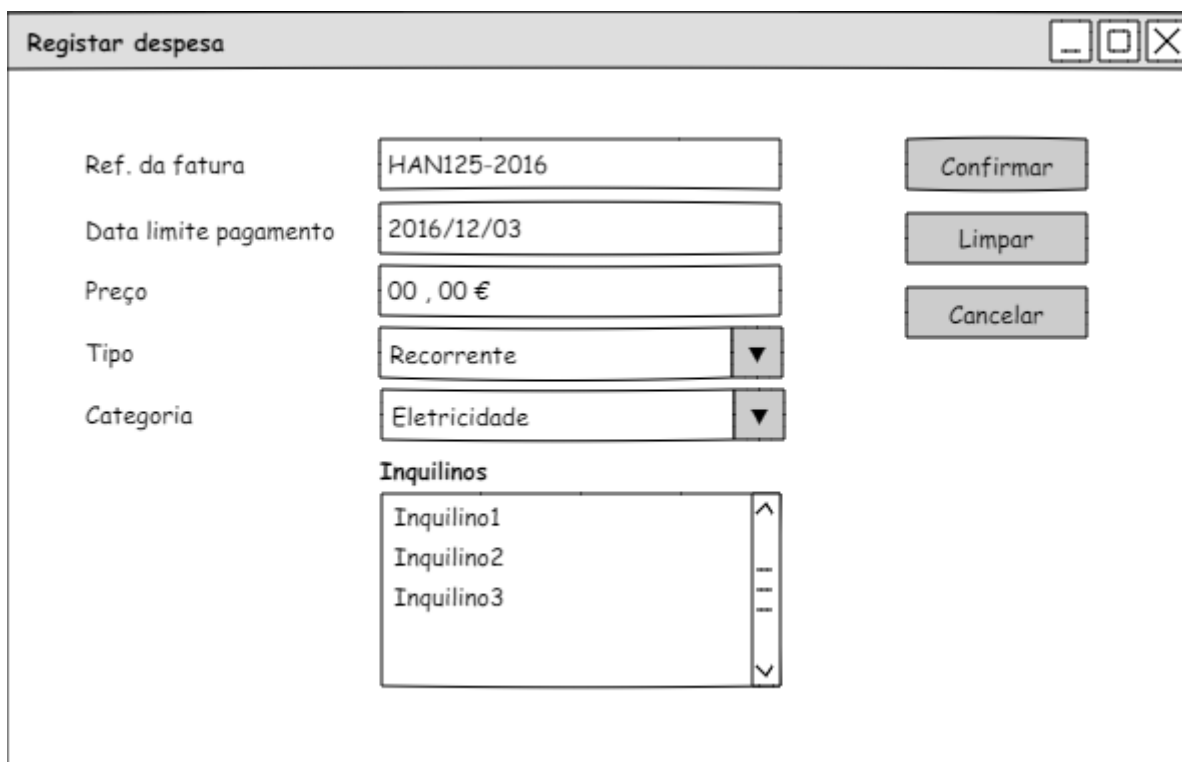


Figura 30 – Mockup “Registar despesa extraordinária”



**Registrar despesa**

Ref. da fatura: HAN125-2016

Data limite pagamento: 2016/12/03

Preço: 00,00 €

Tipo: Recorrente

Categoria: Eletricidade

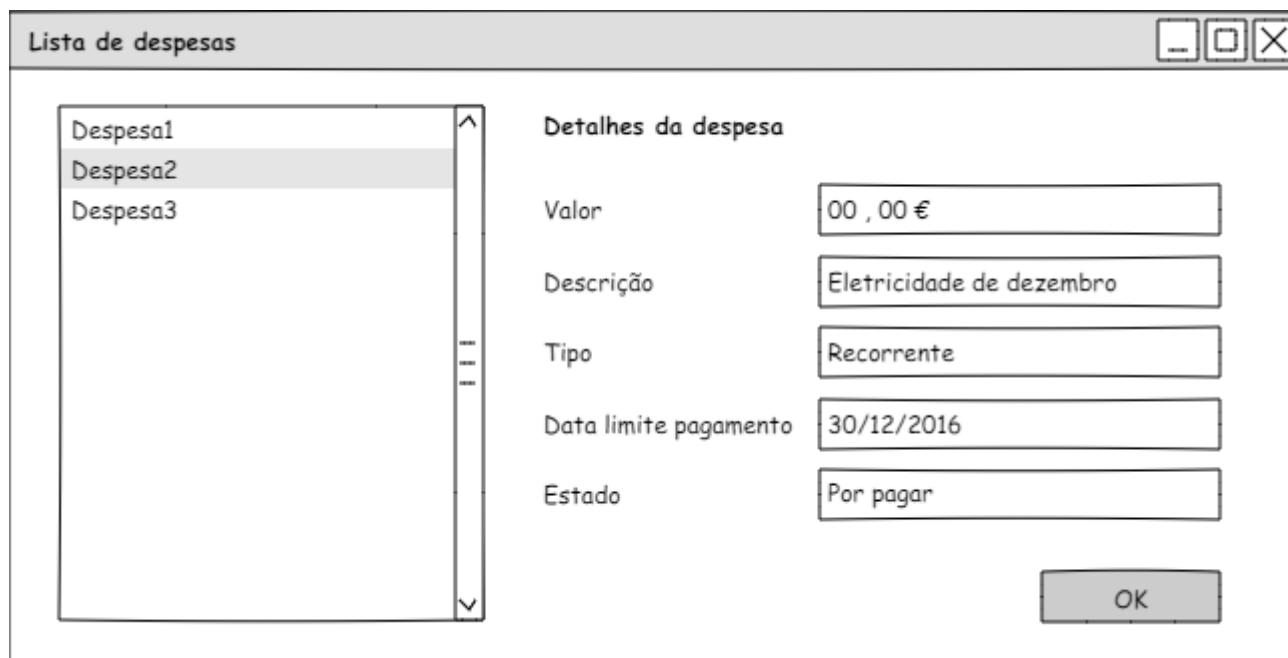
**Inquilinos**

- Inquilino1
- Inquilino2
- Inquilino3

Buttons: Confirmar, Limpar, Cancelar

Figura 31 – Mockup “Registrar despesa recorrente”

### 3.12. Lista de despesas



**Lista de despesas**

Despesa1

Despesa2

Despesa3

**Detalhes da despesa**

Valor: 00,00 €

Descrição: Eletricidade de dezembro

Tipo: Recorrente

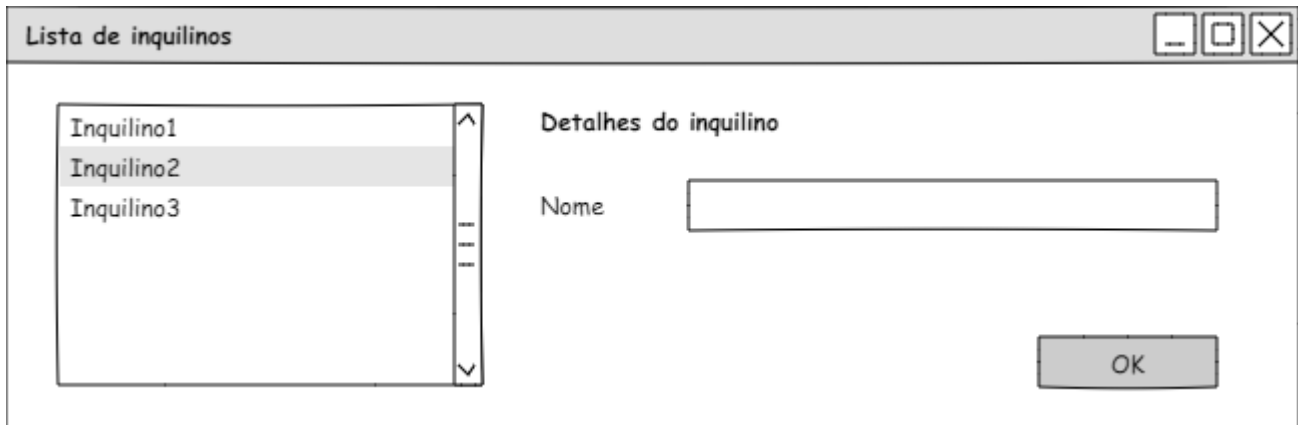
Data limite pagamento: 30/12/2016

Estado: Por pagar

OK

Figura 32 – Mockup “Lista de despesas”

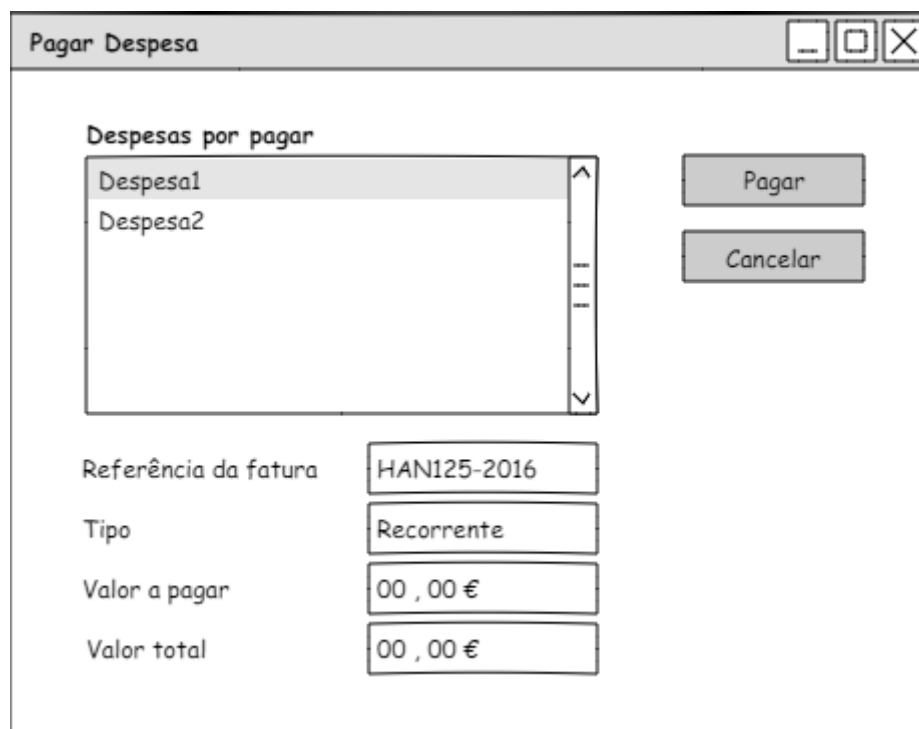
### 3.13. Lista de inquilinos



The mockup shows a window titled "Lista de inquilinos" with standard window controls (minimize, maximize, close). On the left, there is a list box containing three items: "Inquilino1", "Inquilino2" (which is selected), and "Inquilino3". To the right of the list box is a section titled "Detalhes do inquilino" containing a text input field labeled "Nome". At the bottom right of the window is an "OK" button.

Figura 33 – Mockup “Lista de inquilinos”

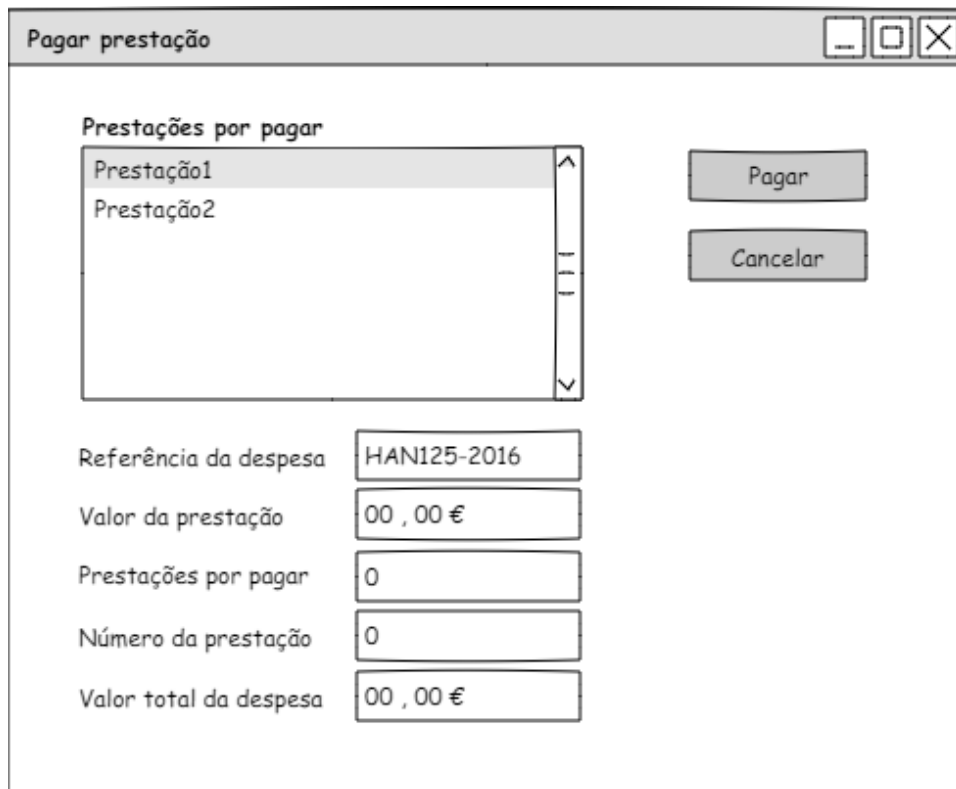
### 3.14. Pagar despesa



The mockup shows a window titled "Pagar Despesa" with standard window controls (minimize, maximize, close). Inside the window, there is a section titled "Despesas por pagar" with a list box containing two items: "Despesa1" (selected) and "Despesa2". To the right of this list box are two buttons: "Pagar" and "Cancelar". Below the list box, there are four rows of labels and text input fields: "Referência da fatura" with the value "HAN125-2016", "Tipo" with the value "Recorrente", "Valor a pagar" with the value "00 , 00 €", and "Valor total" with the value "00 , 00 €".

Figura 34 – Mockup “Pagar despesa”

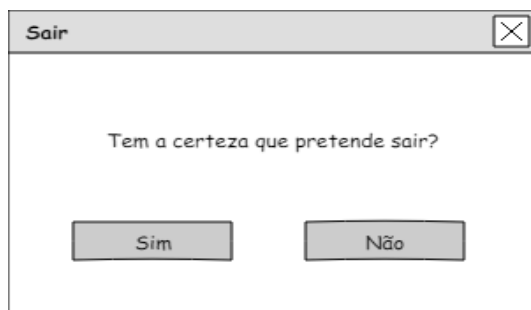
### 3.15. Pagar prestação



The mockup shows a dialog box titled "Pagar prestação". It contains a list box labeled "Prestações por pagar" with two items: "Prestação1" and "Prestação2". To the right of the list are two buttons: "Pagar" and "Cancelar". Below the list box are five input fields with labels: "Referência da despesa" (containing "HAN125-2016"), "Valor da prestação" (containing "00 , 00 €"), "Prestações por pagar" (containing "0"), "Número da prestação" (containing "0"), and "Valor total da despesa" (containing "00 , 00 €").

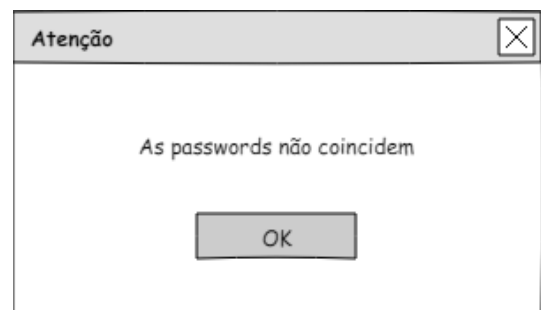
Figura 35 – Mockup “Pagar prestação”

### 3.16. Mockups de avisos e confirmações



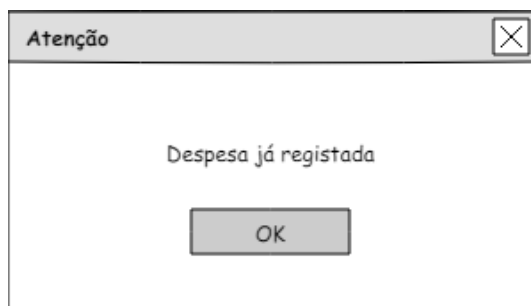
The mockup shows a dialog box titled "Sair". It contains the text "Tem a certeza que pretende sair?". Below the text are two buttons: "Sim" and "Não".

Figura 36 – Mockup “Sair”



The mockup shows a dialog box titled "Atenção". It contains the text "As passwords não coincidem". Below the text is a single button: "OK".

Figura 37 – Mockup “Passwords não coincidem”



The mockup shows a dialog box titled "Atenção". It contains the text "Despesa já registada". Below the text is a single button: "OK".

Figura 38 – Mockup “Despesa já registada”



The mockup shows a dialog box titled "Atenção". It contains the text "Sem saldo". Below the text is a single button: "OK".

Figura 39 – Mockup “Sem saldo”

## 4. Diagramas de Máquinas de Estado

Decidiu-se esboçar os diagramas de máquinas de estado de forma a modelar o comportamento do sistema como um todo. Acharmos importante fazer estes diagramas pois permite modelar todos os estados possíveis que o sistema atravessa em resposta aos eventos que podem ocorrer.

### 4.1. Geral

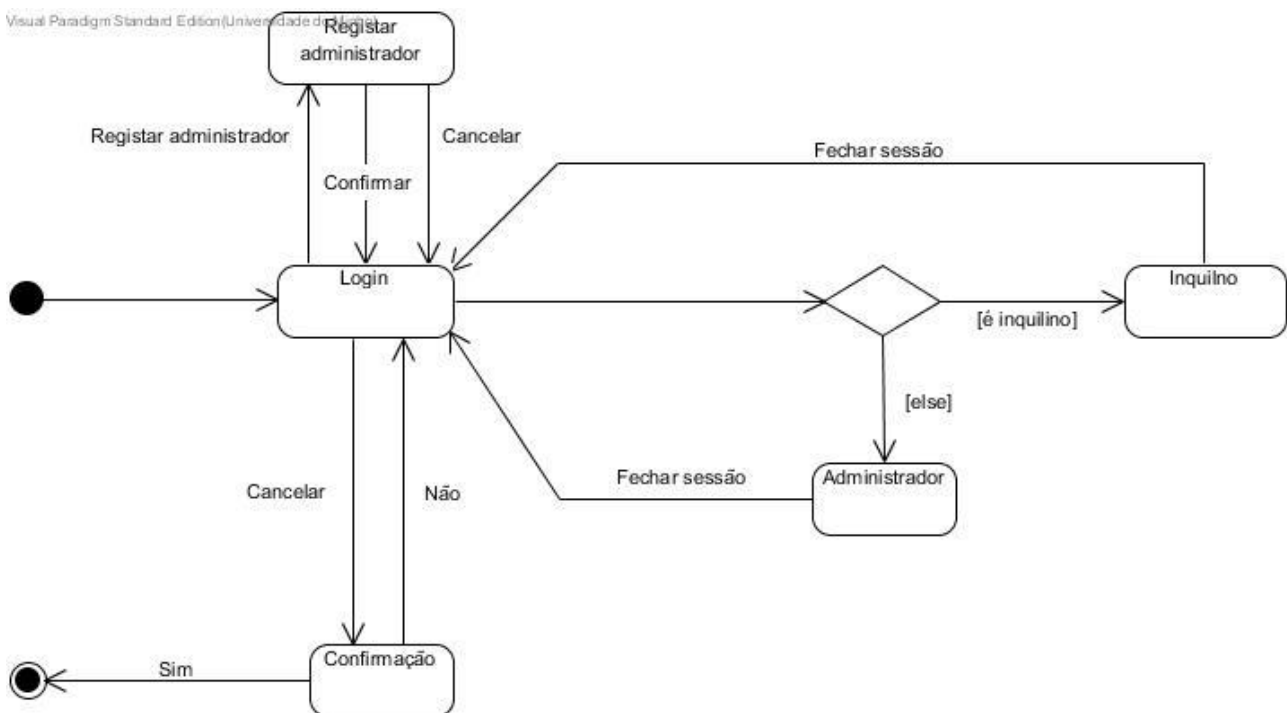


Figura 40 – Diagrama de Máquina de Estado Geral

## 4.2. Administrador

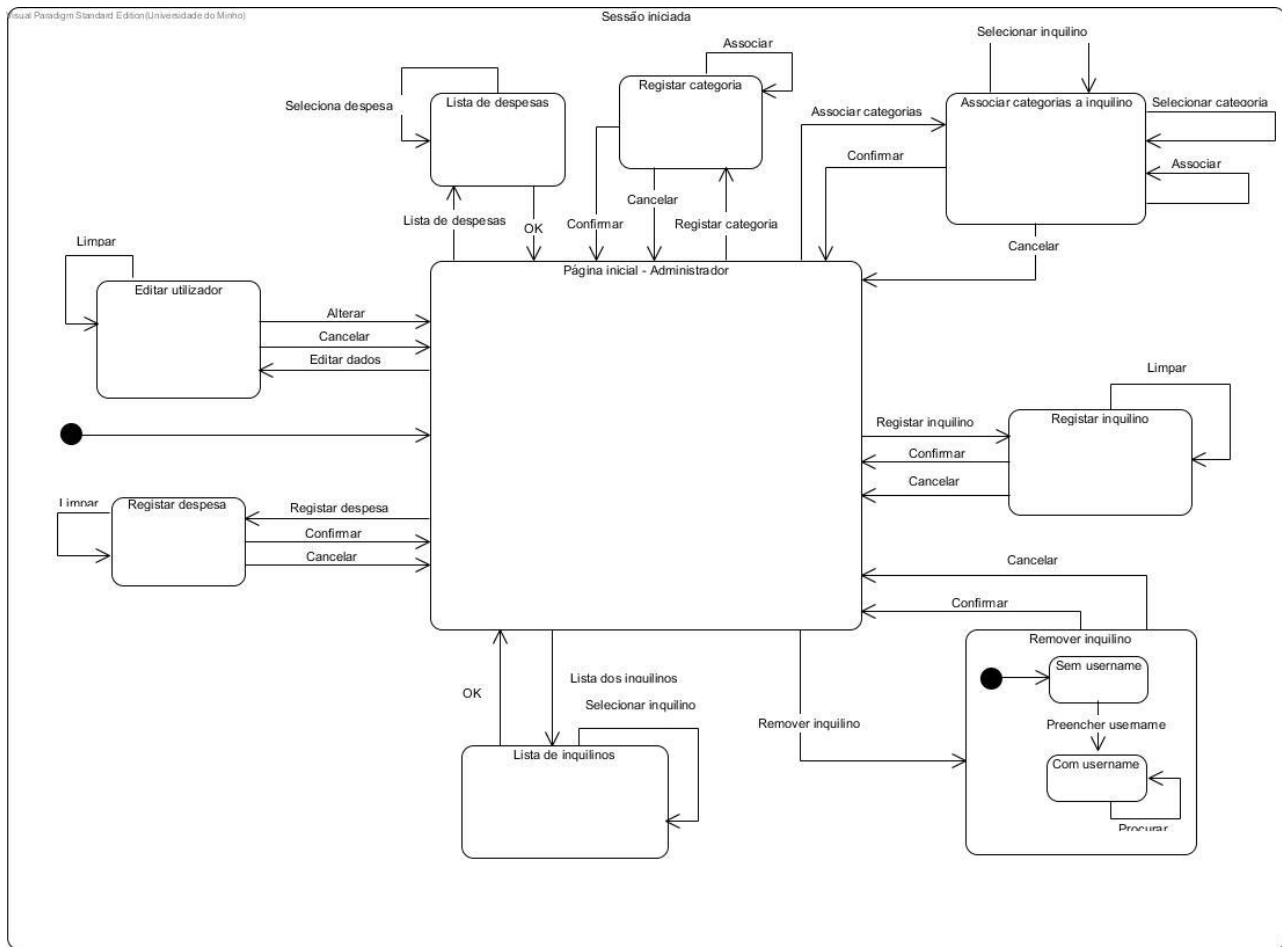


Figura 41 – Diagrama de Máquina de Estado do Administrador



### 4.3. Inquilino

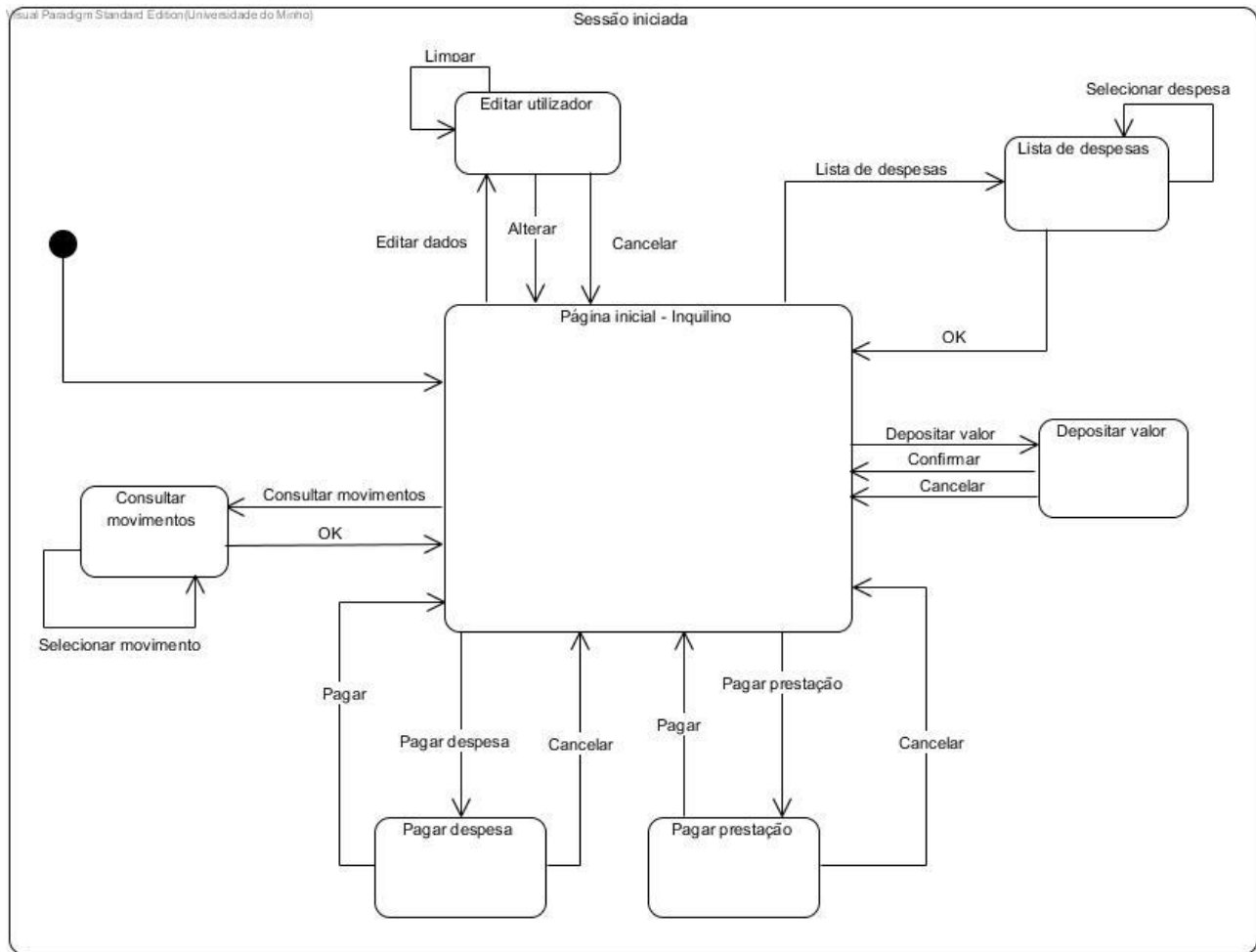


Figura 42 – Diagrama de Máquina de Estado do Inquilino

## 5. Definição dos subsistemas

No ponto de aprofundar a modelação da aplicação com os diagramas de sequência, foi necessário definir quais os subsistemas que se conseguem retirar do modelo de domínio definido. Assim, da análise do modelo de Domínio, foram considerados dois subsistemas: o subsistema dos utilizadores, que inclui toda a informação relativa a um utilizador, incluindo a sua conta e os movimentos que ocorrem sobre ela e o subsistema das despesas, que inclui toda a informação sobre as despesas registadas e sobre quem as paga.

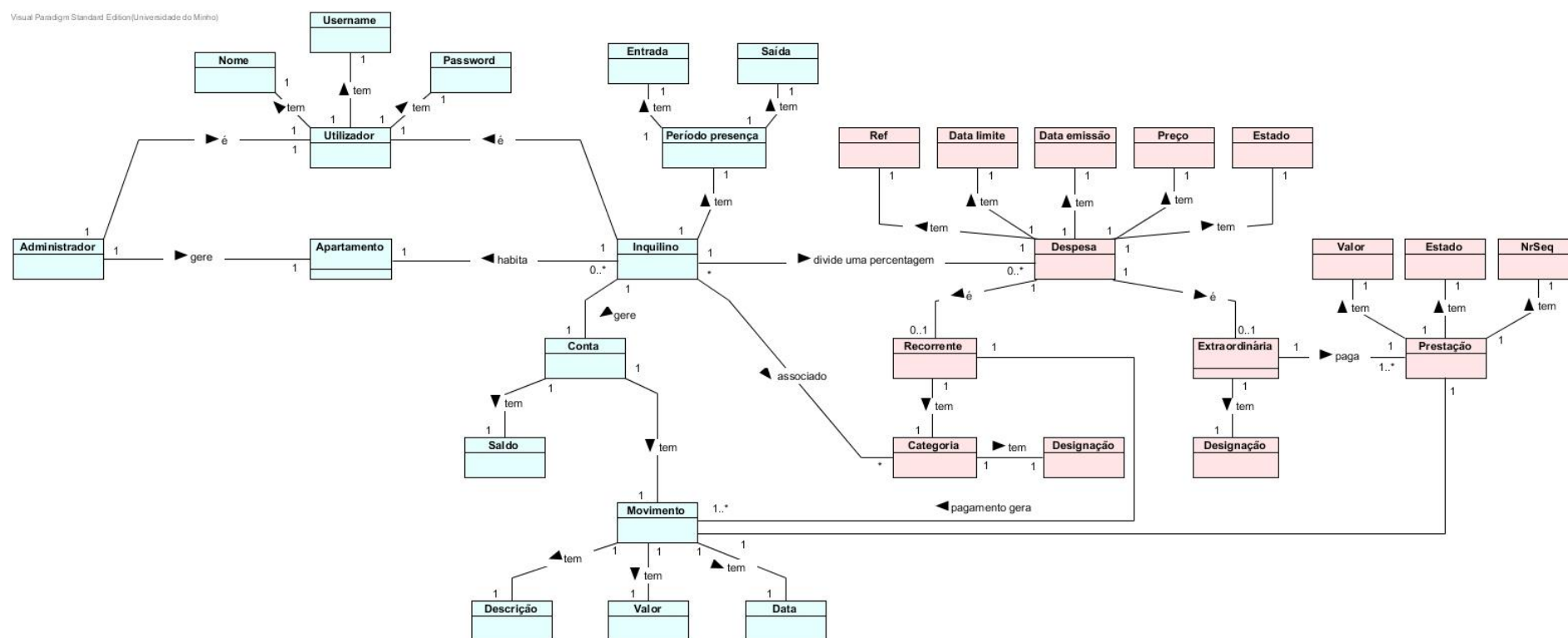


Figura 43 – Modelo de Domínio com subsistemas

## 6. Diagramas de Sequência com subsistemas

Tendo sido definidos os dois subsistemas que constituirão a aplicação, foram construídos os diagramas de sequência para cada um dos *Use Cases*, com o planeamento dos métodos que cada subsistema terá que suportar e a sequenciação temporal de todos os passos até se usar cada funcionalidade.

### 6.1. Autenticar

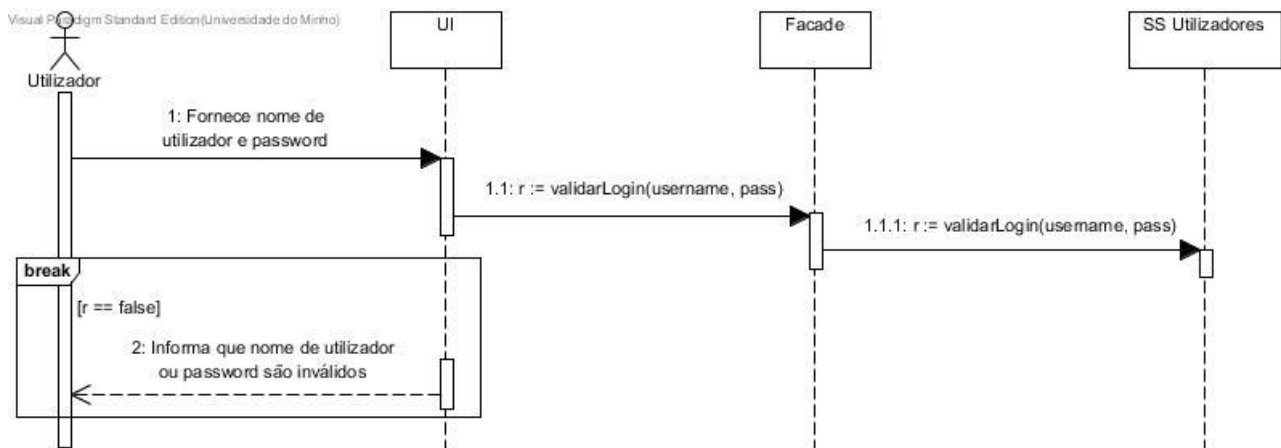


Figura 44 – Diagrama de Sequência referente ao Use Case “Autenticar”

### 6.2. Fechar sessão

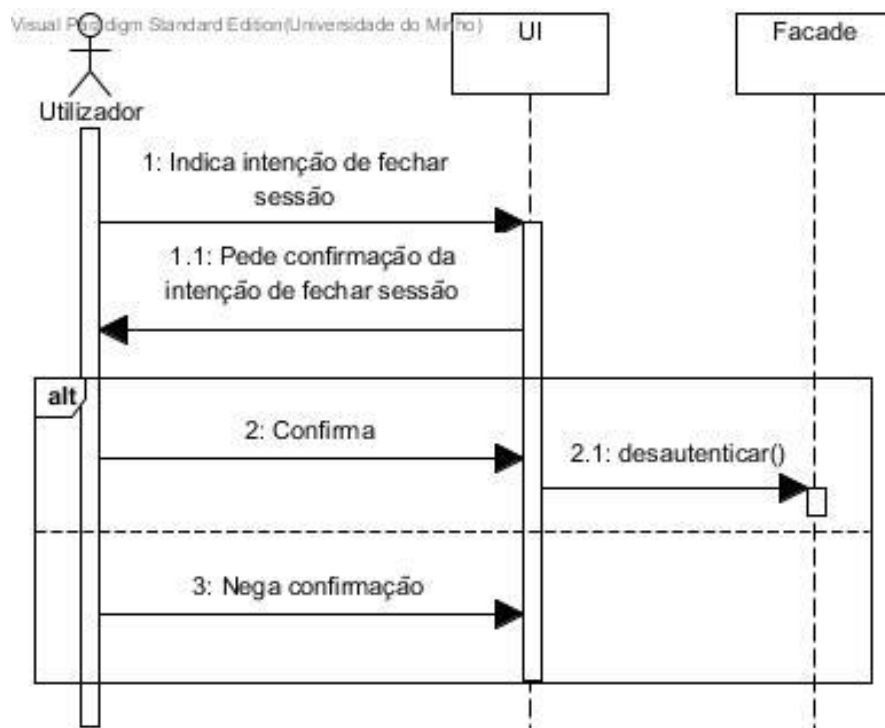


Figura 45 – Diagrama de Sequência referente ao Use Case “Fechar sessão”

### 6.3. Editar utilizador

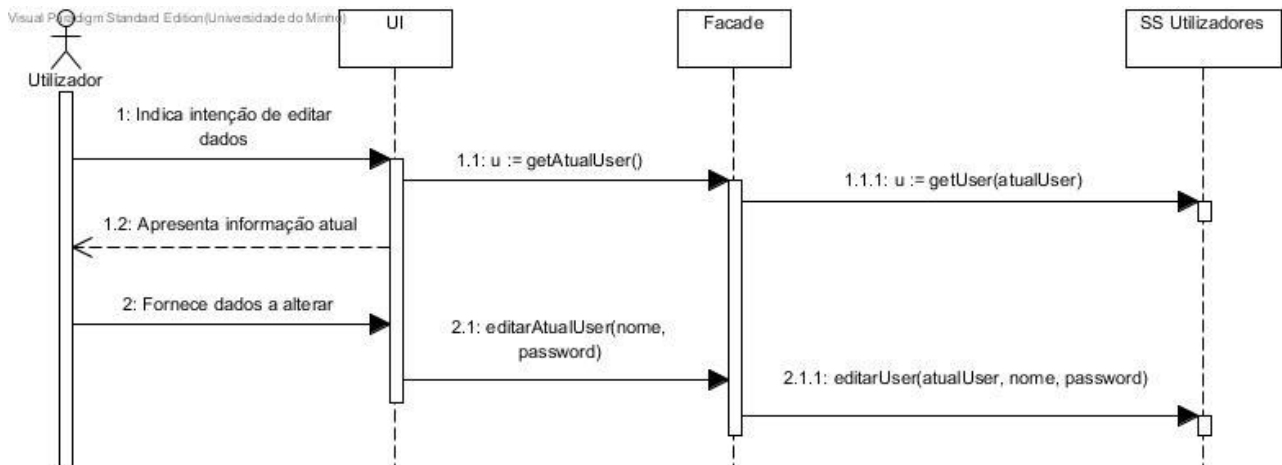


Figura 46 – Diagrama de Sequência referente ao Use Case “Editar utilizador”

### 6.4. Obter lista de despesas

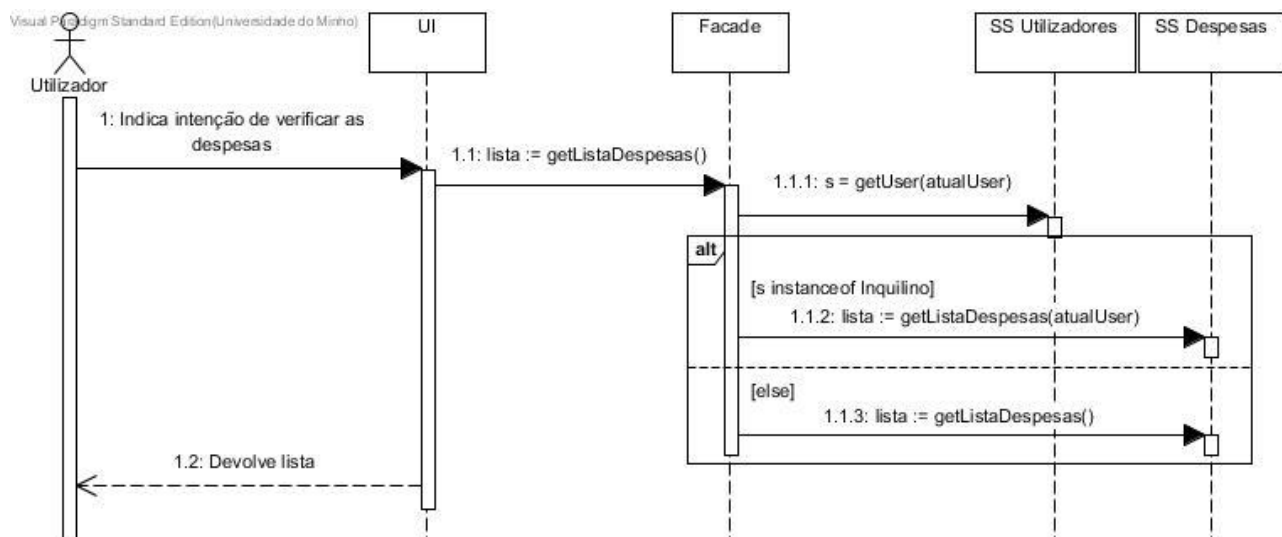


Figura 47 – Diagrama de Sequência referente ao Use Case “Obter lista de despesas”

### 6.5. Consultar saldo

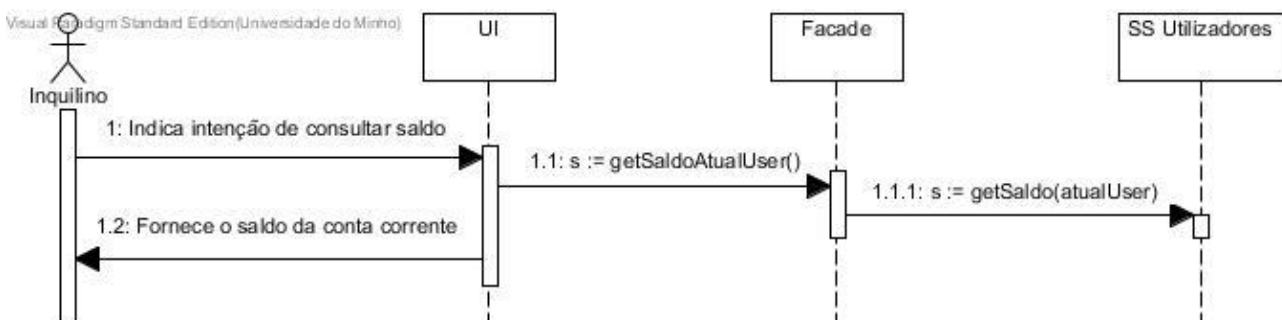


Figura 48 – Diagrama de Sequência referente ao Use Case “Consultar saldo”

## 6.6. Depositar quantia

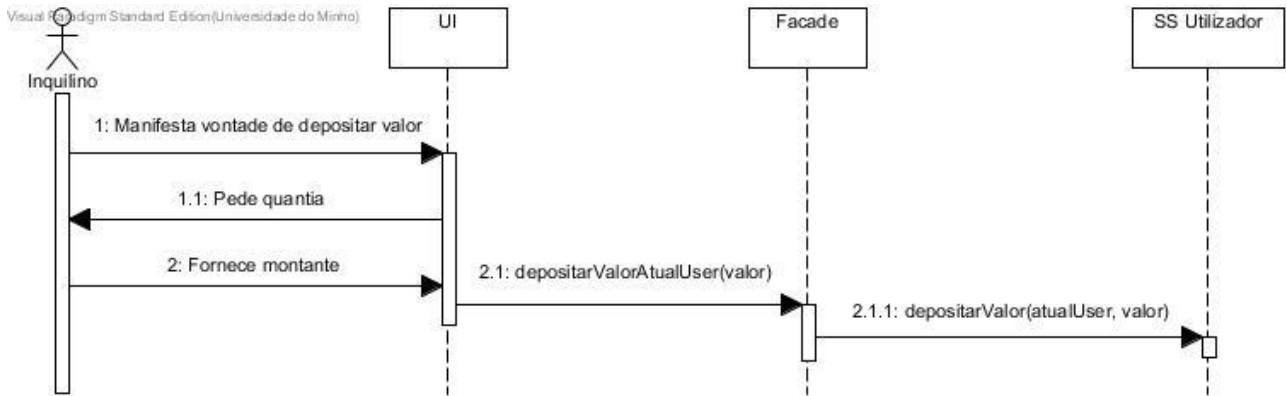


Figura 49 – Diagrama de Sequência referente ao Use Case “Depositar quantia”

## 6.7. Consultar movimentos

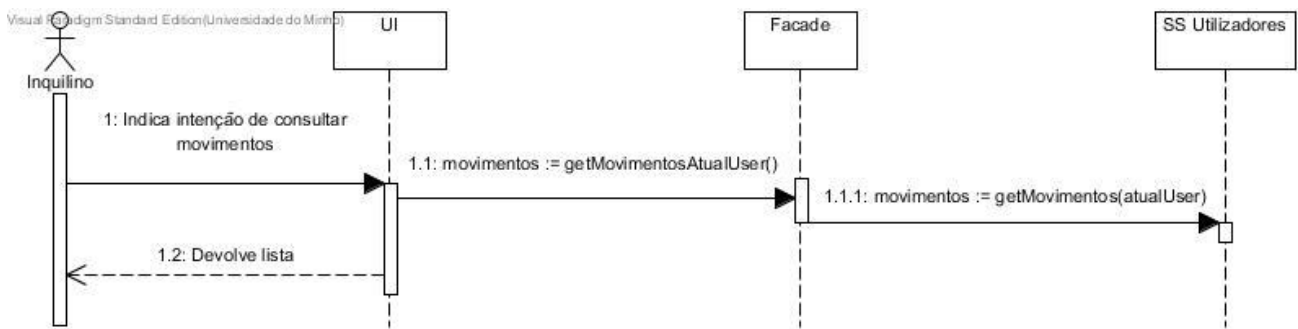


Figura 50 – Diagrama de Sequência referente ao Use Case “Consultar movimentos”

## 6.8. Registrar inquilino

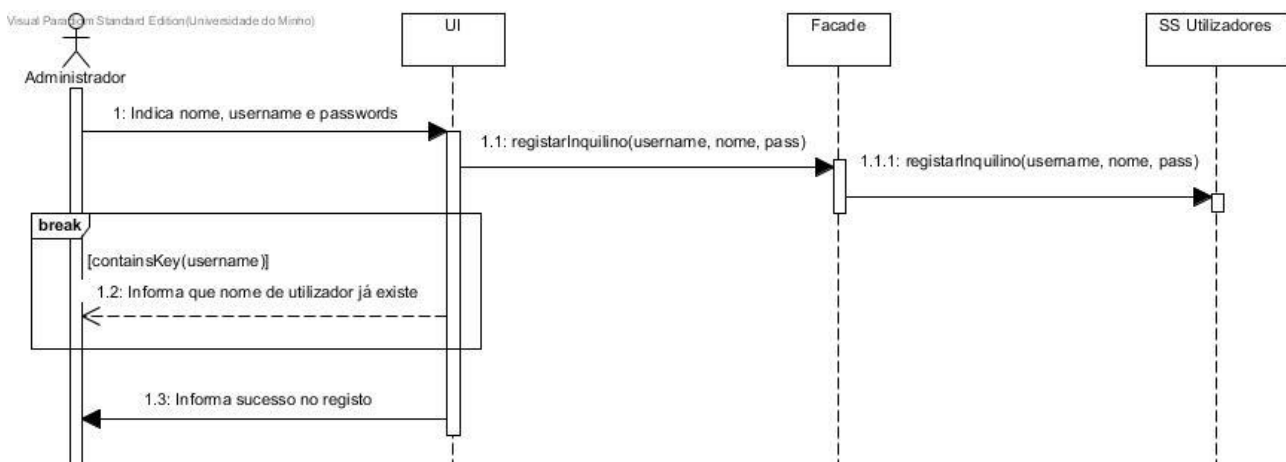


Figura 51 – Diagrama de Sequência referente ao Use Case “Registrar inquilino”

## 6.9. Registrar categoria

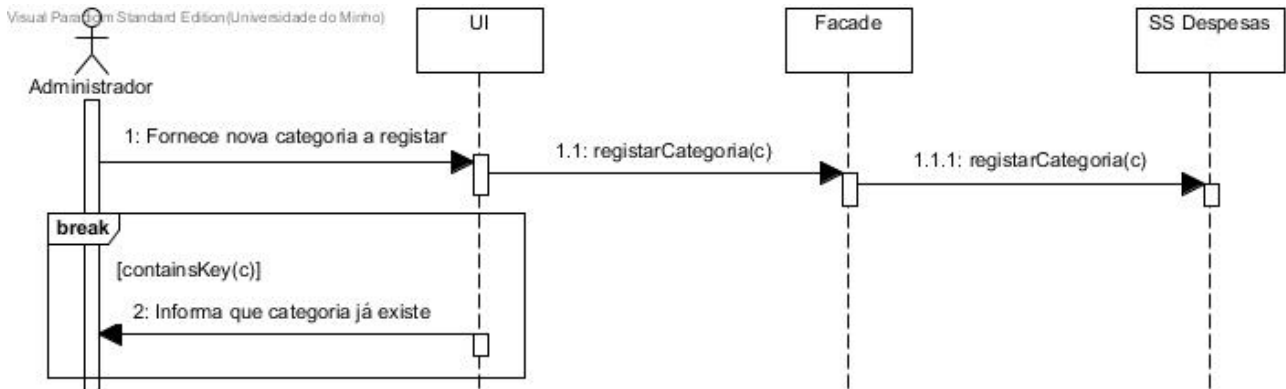


Figura 52 – Diagrama de Sequência referente ao Use Case “Registrar categoria”

## 6.10. Obter lista dos Inquilinos

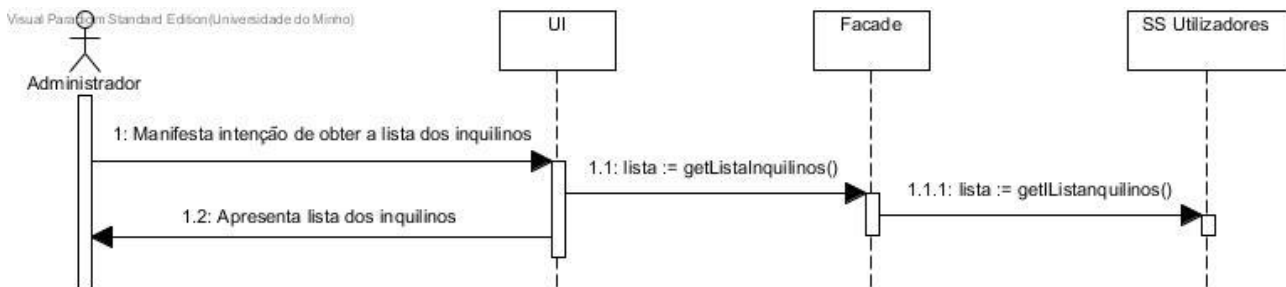


Figura 53 – Diagrama de Sequência referente ao Use Case “Obter lista dos Inquilinos”

## 6.11. Associar categorias a inquilino

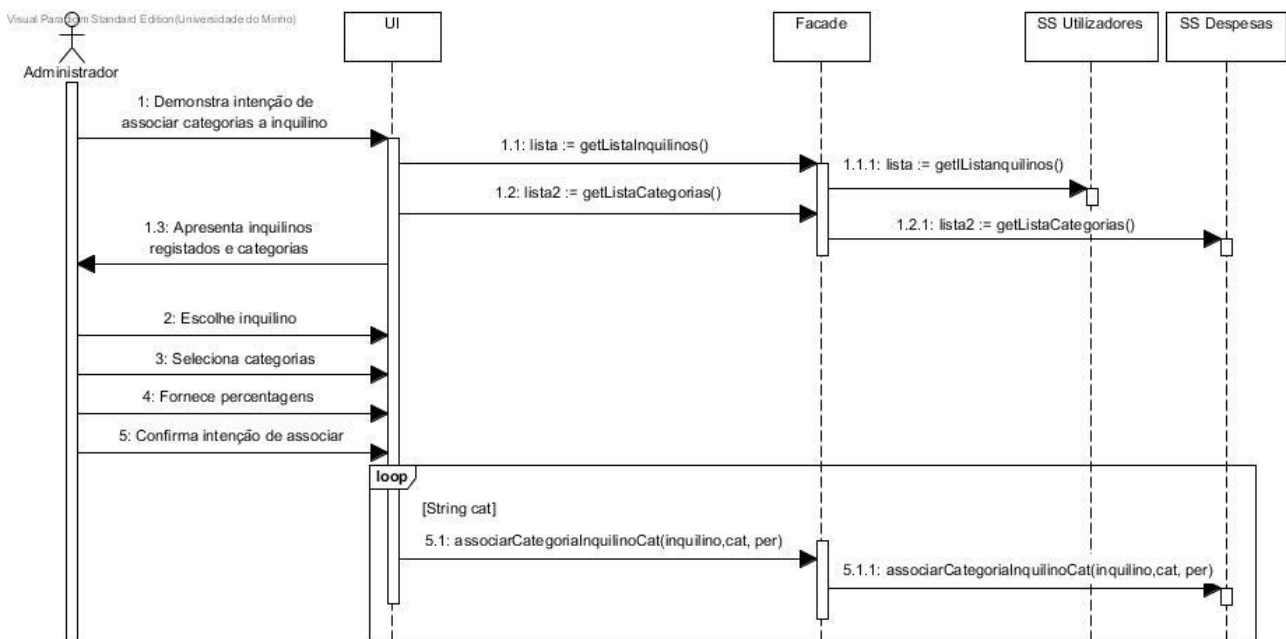


Figura 54 – Diagrama de Sequência referente ao Use Case “Associar categorias a inquilino”

## 6.12. Remover inquilino

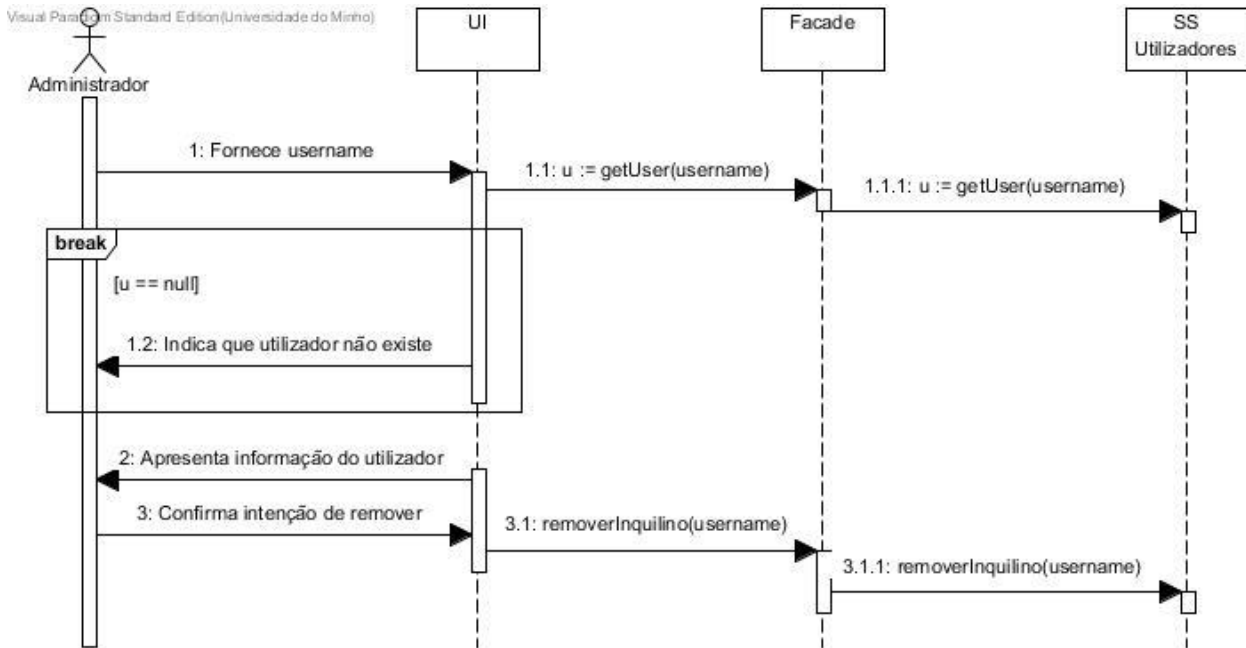


Figura 55 – Diagrama de Sequência referente ao Use Case “Remover inquilino”

## 6.13. Registar despesa

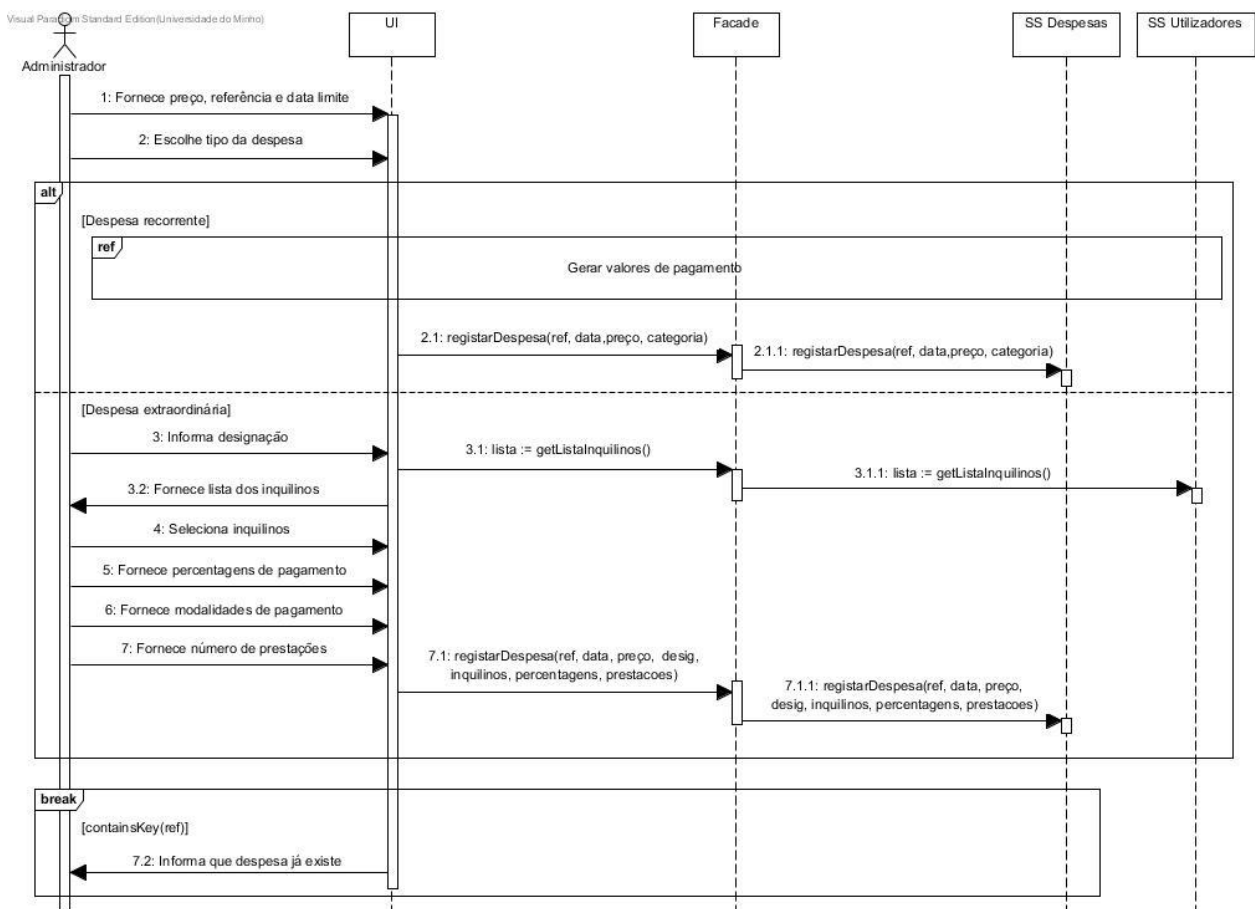


Figura 56 – Diagrama de Sequência referente ao Use Case “Registar despesa”

## 6.14. Pagar despesa

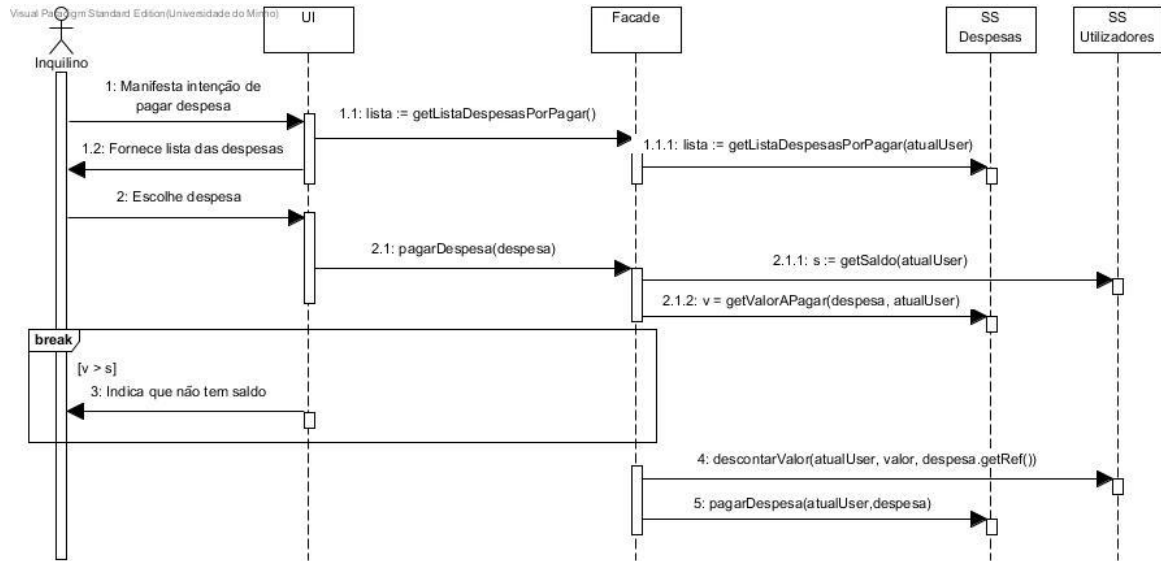


Figura 57 – Diagrama de Sequência referente ao Use Case “Pagar Despesa”

## 6.15. Pagar prestação

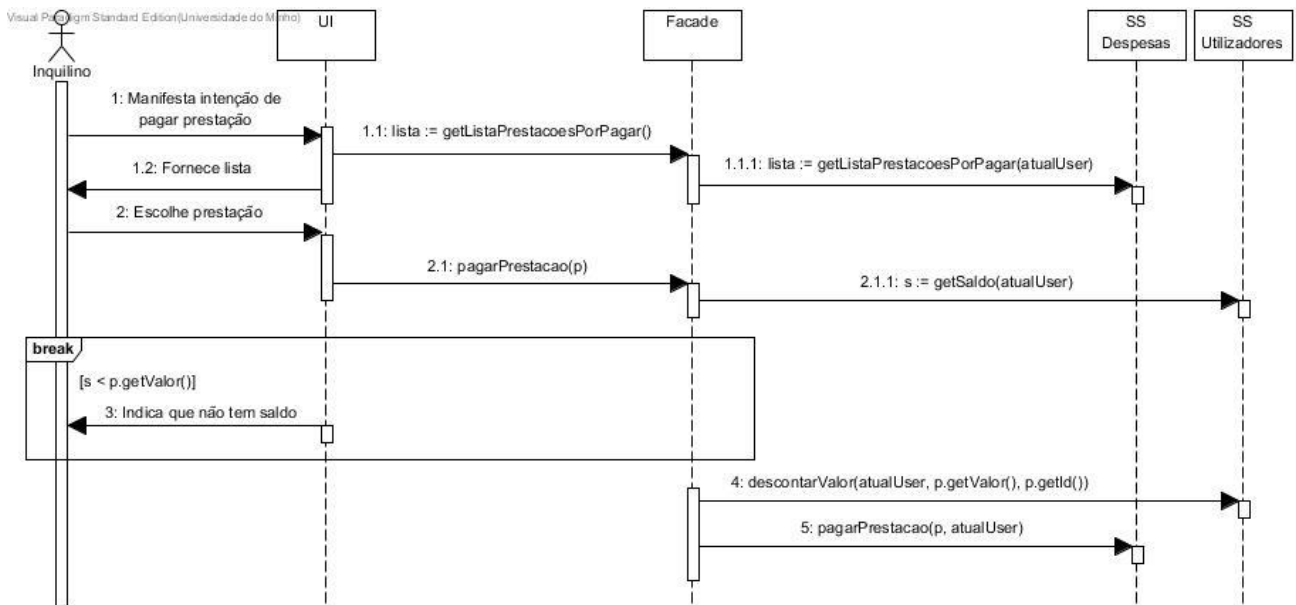


Figura 58 – Diagrama de Sequência referente ao Use Case “Pagar prestação”



## 6.16. Gerar valores de pagamento

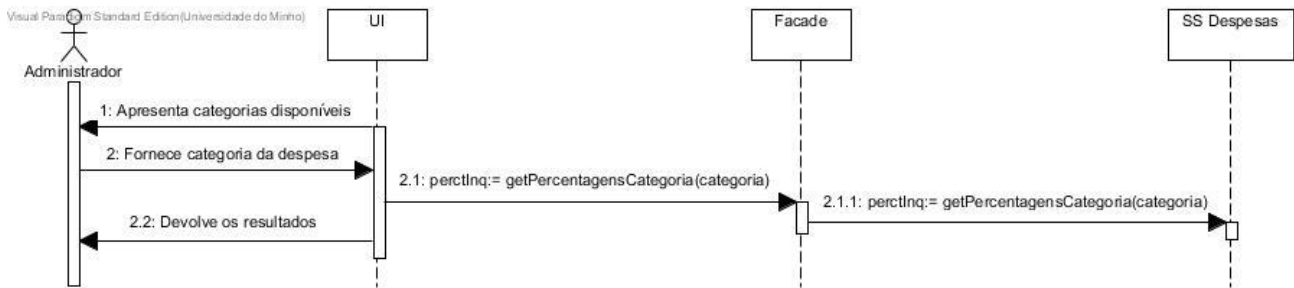


Figura 59 – Diagrama de Sequência referente ao Use Case “Gerar valores de pagamento”

## 6.17. Registar administrador

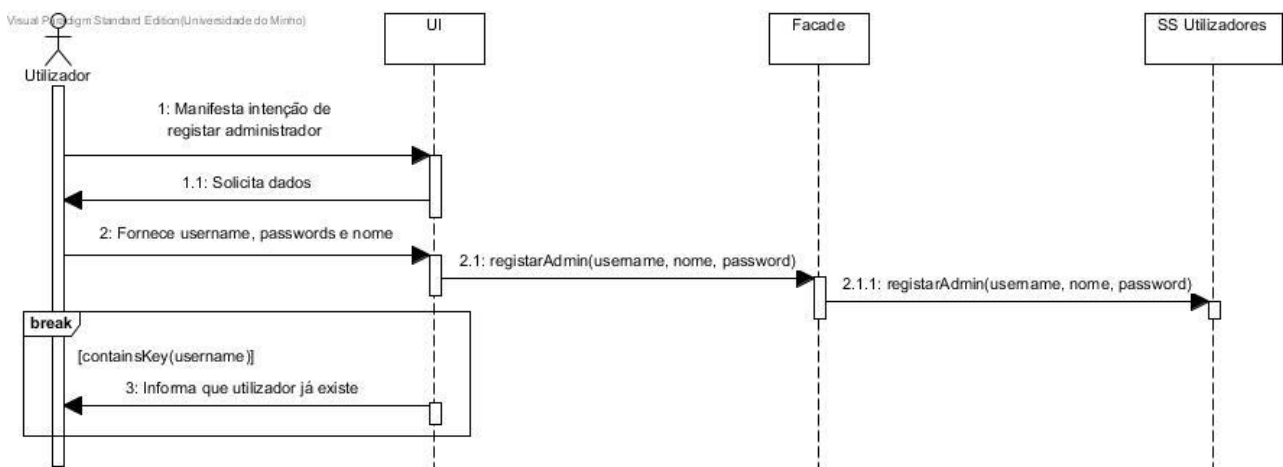


Figura 60 – Diagrama de Sequência referente ao Use Case “Registar administrador”

## 7. Diagrama de Package

Mediante os subsistemas anteriormente considerados e a divisão em *packages* que se conseguiu planejar (nomeadamente a camada de apresentação (UI), a camada de negócio, explicitada nos diagramas de sequência, e a camada de dados, com os respetivos DAO's), constrói-se o seguinte diagrama de Package:

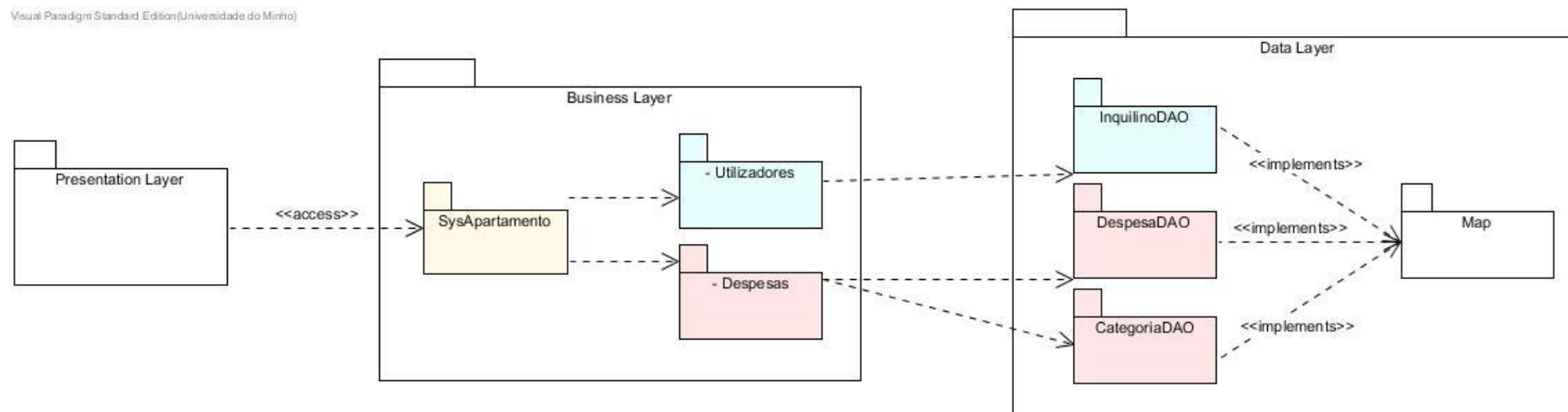


Figura 61 – Diagrama de Package

## 8. Diagrama de Classes

Depois de analisados os diagramas anteriores, foi possível delinear as seguintes classes e os respetivos atributos. Denote-se a existência de uma Facade SysApartamento e duas “sub-Facades” SubSysUsers e SubSysDespesas (representativas dos subsistemas considerados).

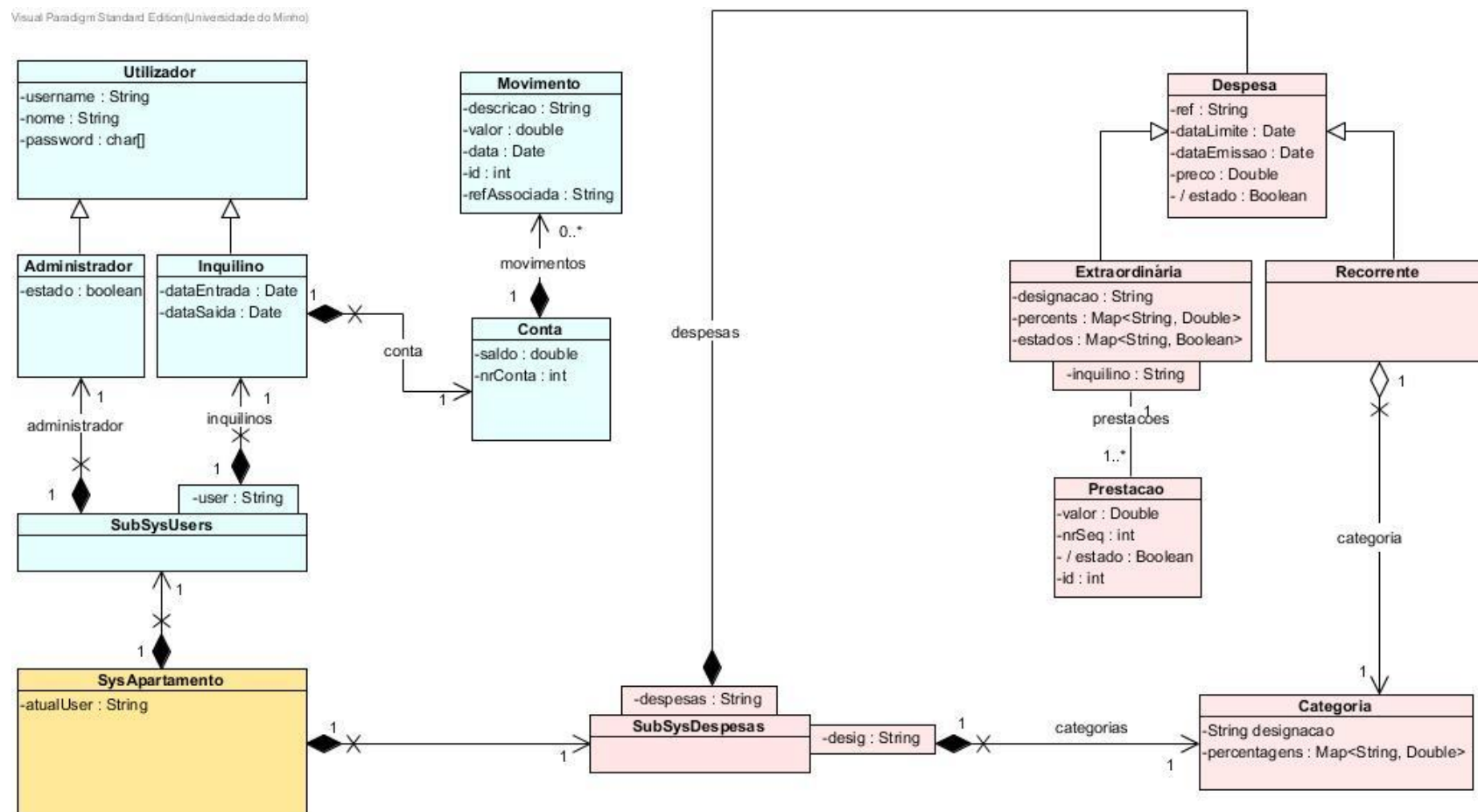


Figura 62 – Diagrama de Classes

De seguida apresenta-se o diagrama de Classes com a implementação dos DAO's. Decidiu-se, por uma questão de simplificação, ter um DAO para a classe Administrador e outro para Inquilino. Como o sistema só pode ter um administrador ativo, no AdministradorDAO consegue-se ter esse maior controlo.

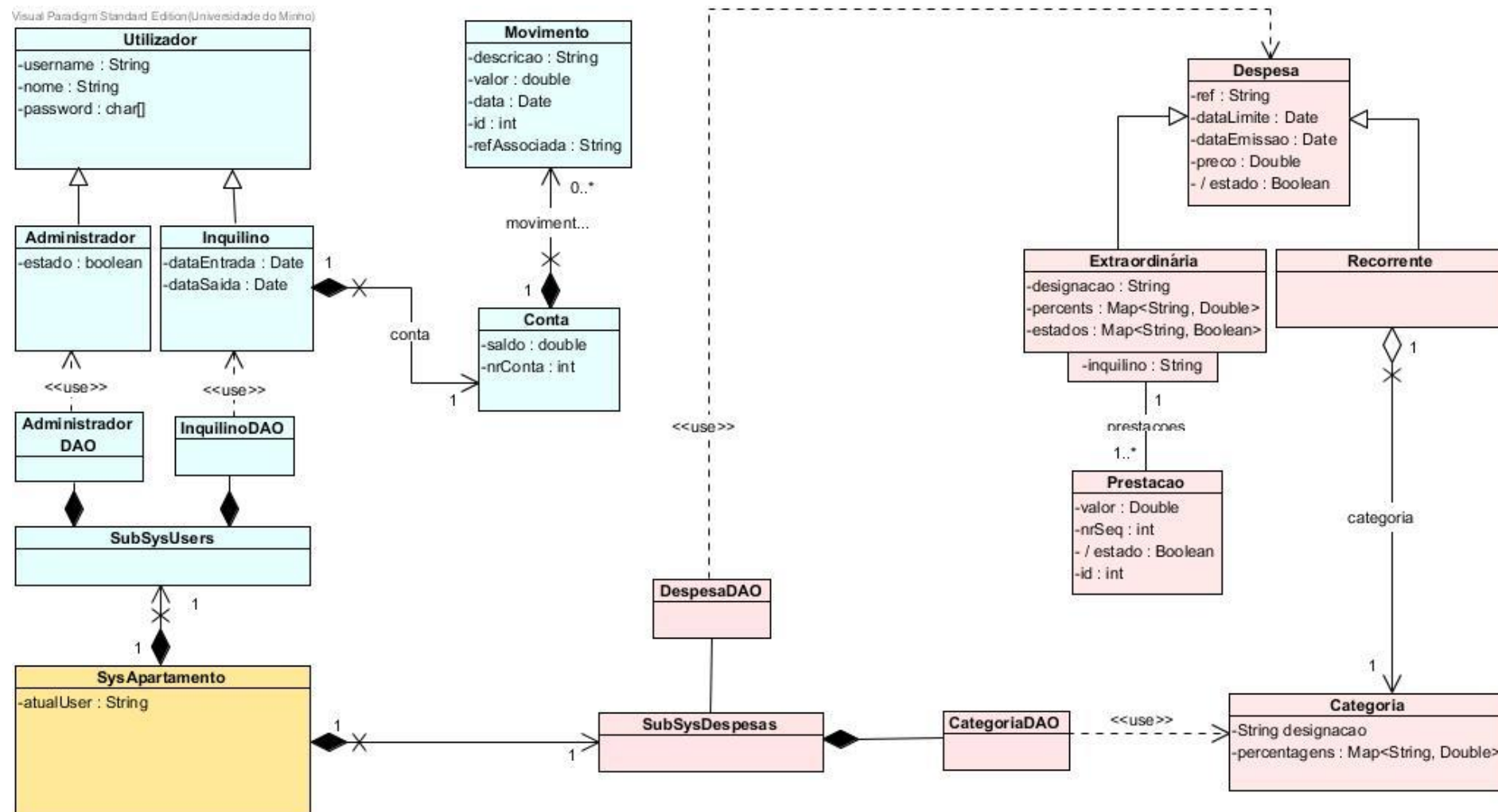


Figura 63 – Diagrama de Classes com DAO's

De seguida é apresentado o diagrama de classes com os DAO's e com alguns dos métodos de cada classe:

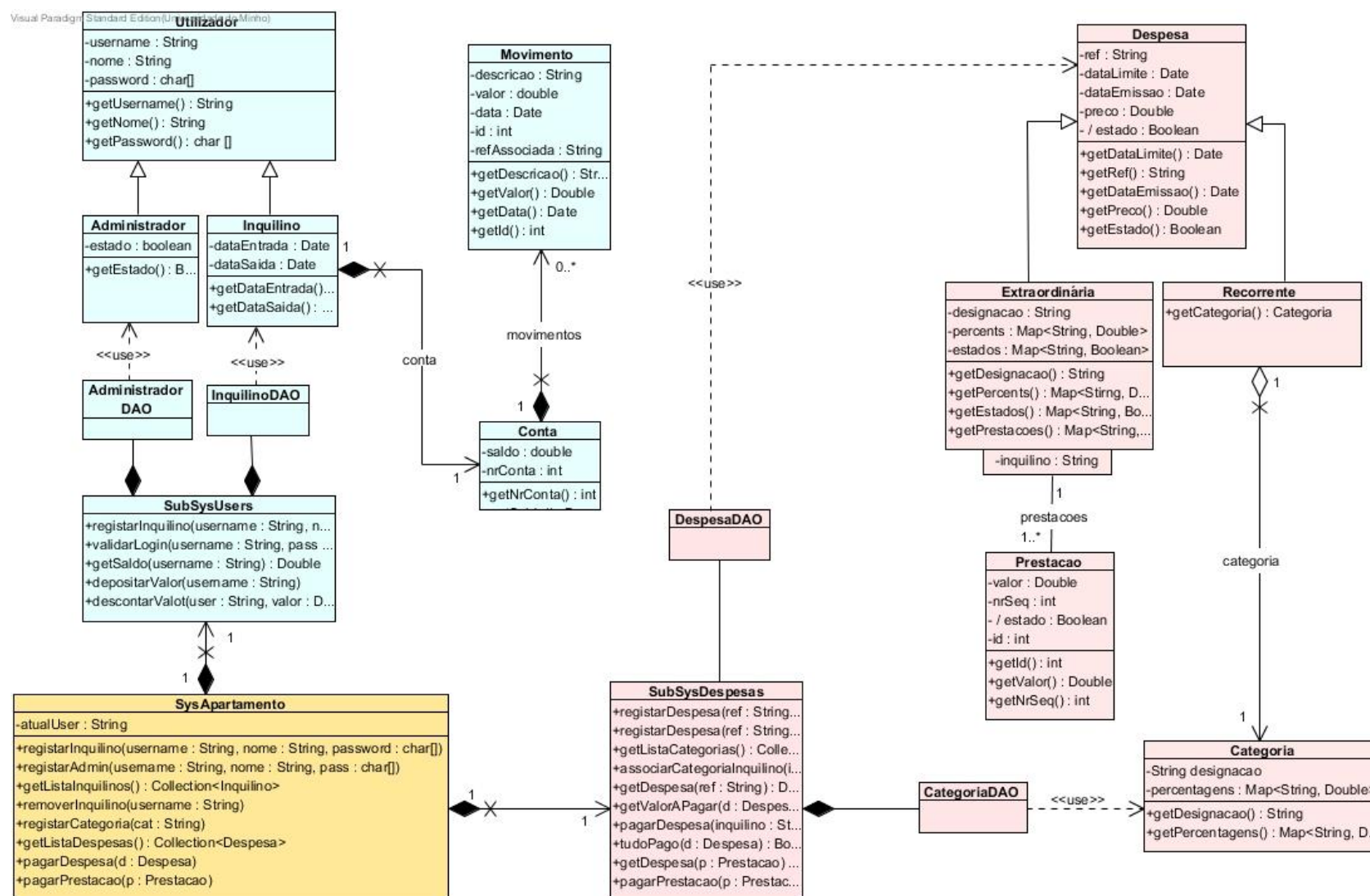


Figura 64 – Diagrama de Classes com DAO's e métodos

## 9. Diagramas de Sequência para a lógica de negócio

De seguida são apresentados três diagramas de sequência que são a especificação dos respectivos DSS apresentados anteriormente, aos quais foram acrescentados os DAO's usados e os métodos invocados.

### 9.1. Pagar despesa

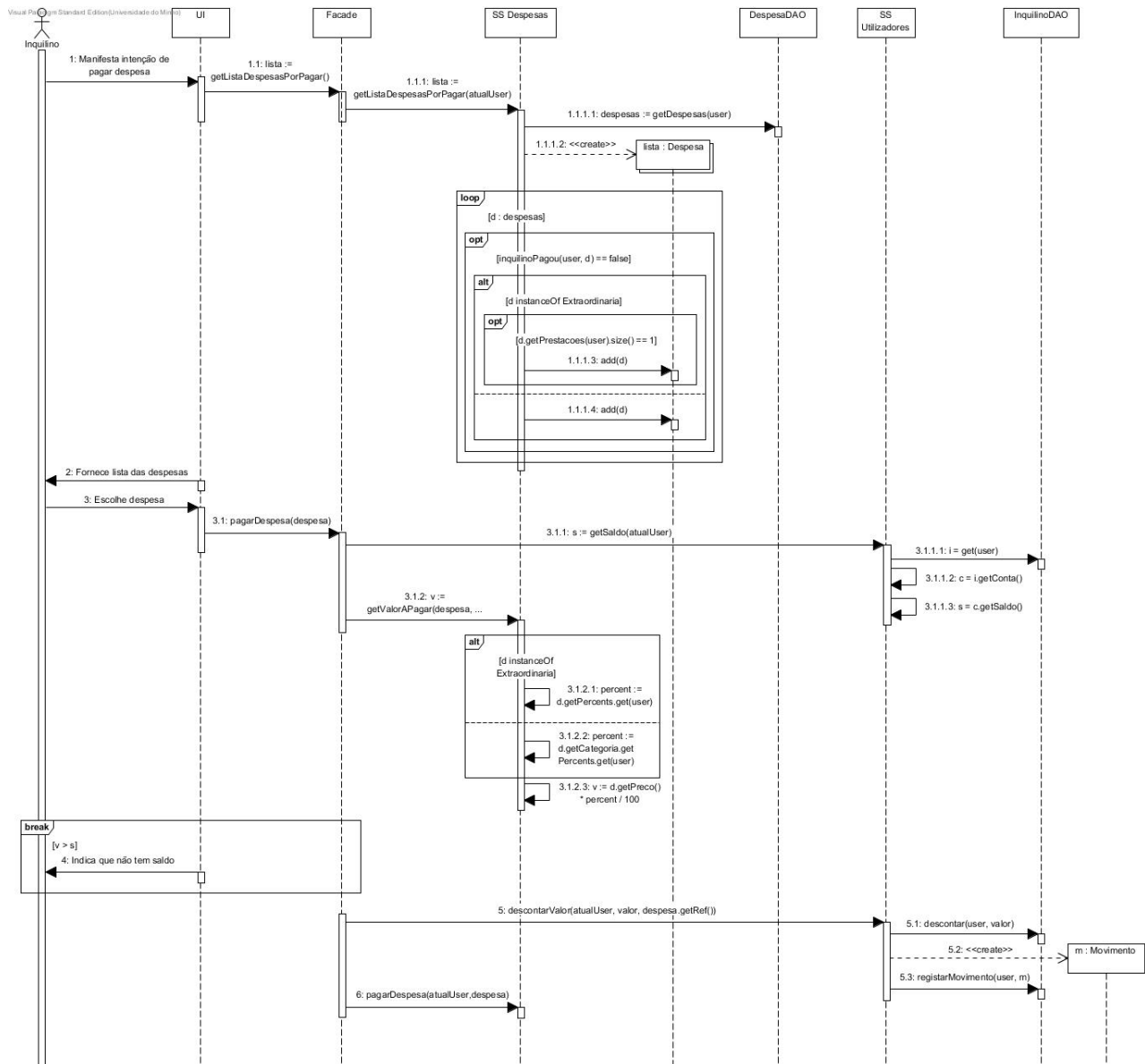


Figura 65 – Diagrama de Sequência para a lógica de negócio referente ao Use Case “Pagar despesa”

## 9.2. Registrar inquilino

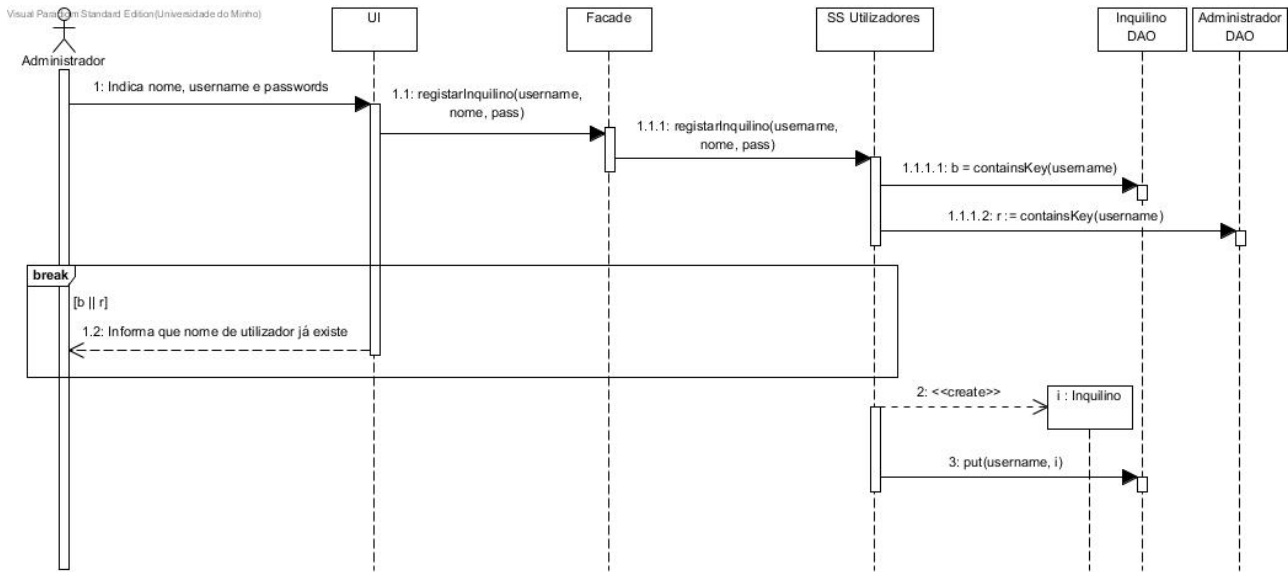


Figura 66 – Diagrama de Sequência para a lógica de negócio referente ao Use Case “Registrar inquilino”

## 9.3. Registrar Categoria

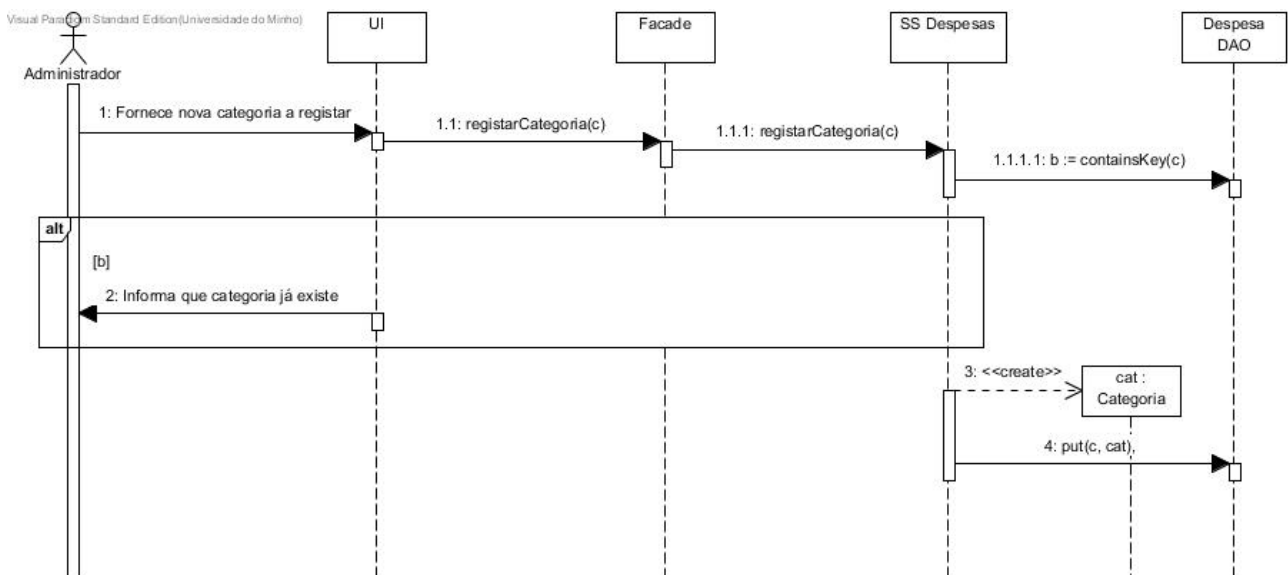


Figura 67 – Diagrama de Sequência para a lógica de negócio referente ao Use Case “Registrar Categoria”

## 10. Diagrama de Instalação

Tendo em conta que a aplicação desenvolvida foi planeada para apenas interagir com a base de dados e que é instalada numa única máquina, o diagrama de instalação é o seguinte:

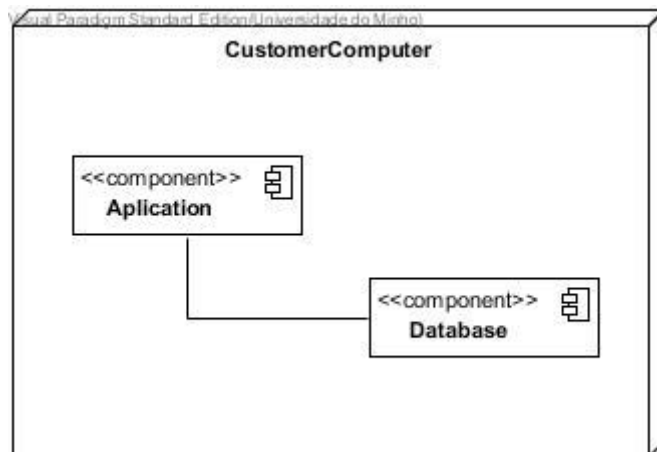


Figura 68 – Diagrama de Instalação



## 11. Implementação da base de dados

Para o suporte da persistência dos dados na aplicação, foi criada uma base de dados capaz de responder aos requisitos definidos no momento do planeamento das funcionalidades do sistema.

O uso de uma Base de Dados permite isolar a camada de dados, de forma a que o resto da aplicação não esteja dependente da forma como os dados são armazenados. Este formato é escalável, ao contrário do que aquilo que aconteceria se fossem usados ficheiros objeto.

De seguida é apresentado o modelo lógico, com a explicitação das tabelas consideradas e dos atributos de cada uma. Esta representação é idêntica àquela que foi apresentada no diagrama de classes, uma vez que se seguiram as regras de criação de uma base de dados relacional, a partir de um projeto de uma linguagem orientada aos objetos.

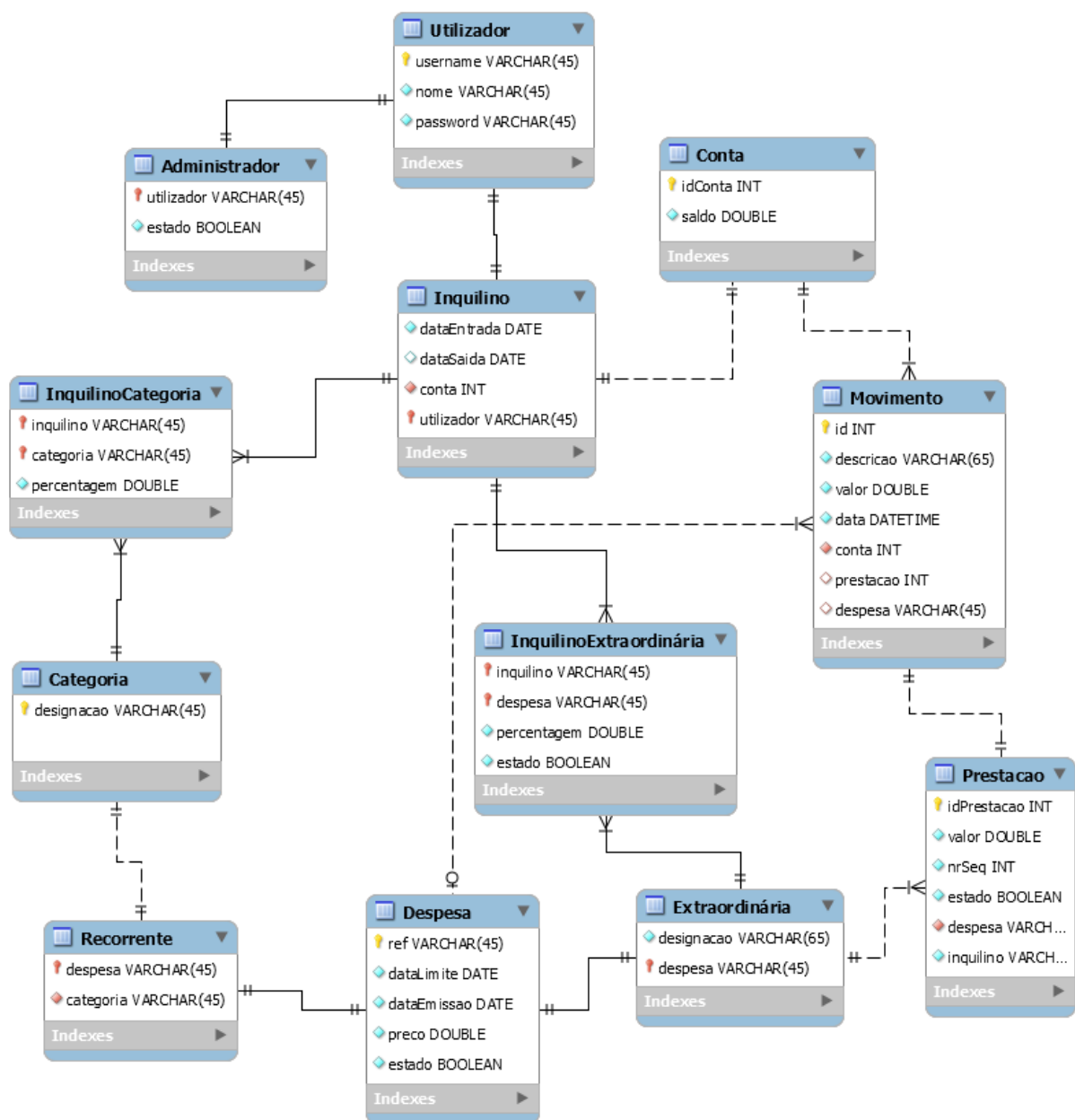


Figura 69 – Modelo lógico da Base de Dados implementada



## Conclusão

Com a realização deste trabalho é possível concluir que a modelação e planeamento de uma aplicação tem uma importância tão alta quanto a escrita do seu código fonte. É através da realização de um diagrama de domínio que se define o problema e da especificação dos use cases que se constrói as funcionalidades da aplicação e as permissões de cada ator. Os diagramas de sistema são uma mais valia para a construção do código, pois nestes é possível definir métodos e a sequência das operações na concretização de uma funcionalidade da aplicação. Os diagramas de classes permitem definir as classes e os respetivos atributos e métodos que constituirão a aplicação numa linguagem orientada por objetos.

O desenvolvimento da aplicação em *Swing*, permitiu a aprendizagem da sintaxe usada, as operações que são possíveis e as ferramentas que existem ao dispor de qualquer programador.

De forma geral é possível concluir que o objetivo proposto foi alcançado. Foi desenvolvida uma aplicação que responde aos requisitos impostos de forma a permitir uma gestão eficiente das despesas num apartamento. Tentou-se que a aplicação não contivesse apenas funcionalidades de consulta e registo. Para tal, implementou-se um sistema que gera automaticamente os inquilinos que paga uma dada despesa recorrente de acordo com as categorias a que estão associados e da qual a despesa faz parte. Porém, por questões de falta de tempo, não foi possível implementar mais funcionalidades que automatizem o sistema, como por exemplo: gerar automaticamente os pagamentos de despesas que sejam periódicas; notificar os inquilinos do registo de novas despesas, entre outras