

Desenvolvimento de Sistemas de Software

Grupo 40

4 de agosto de 2023



Diogo Manuel Brito Pires
a93239



Gonçalo André Rodrigues Soares
a93286



Guilherme Lima Barros Gomes Fernandes
a93216



Rita Celeste Perucho Teixeira
a89494

Conteúdo

1	Introdução e Objetivos	3
2	Abordagem dos Objetivos	4
3	Alterações Realizadas	5
3.1	Modelo de Domínio	5
3.1.1	Primeira Fase	5
3.1.2	Segunda Fase	6
4	Modelo Use Case	7
5	Especificação de Use Cases	8
5.1	Especificação de <i>Use Cases</i>	8
5.1.1	Autenticar Operário	8
5.1.2	Registrar Cliente	8
5.1.3	Registrar Entrega de Equipamento	9
5.1.4	Registrar Pedido de Orçamento	9
5.1.5	Realizar Orçamento de Equipamento	10
5.1.6	Registrar Pedido de Reparação	11
5.1.7	Reparar Equipamento de Serviço Regular	12
5.1.8	Registrar Levantamento de Equipamento	13
5.1.9	Registrar Pedido de Serviço Expresso	13
5.1.10	Reparar Equipamento de Serviço Expresso	14
5.1.11	Consultar Registros	14
6	Diagrama de Componentes	15
7	Diagrama de Classes	16
8	Diagramas de Sequência de Implementação	19
8.1	Autenticar Operário	19
8.2	Registrar Cliente	20
8.3	Registrar Entrega de Equipamento	21
8.4	Realizar Orçamento de Equipamento	22
8.5	Registrar Pedido de Reparação	23
8.6	Registrar Levantamento de Equipamento	23
9	Base de Dados	24
10	Implementação	25
11	Conclusão e Análise Crítica	26

1 Introdução e Objetivos

O trabalho prático que nos foi proposto trata da conceção e modelação de um Sistema de Gestão para Centros de Reparação de equipamentos eletrónicos. Foi-nos dado o objetivo final de um sistema que garanta a gestão de todo o processo de reparação de equipamentos, desde a orçamentação até à entrega do equipamento.

Numa primeira fase, utilizamos a ferramenta Visual Paradigm de modo a elaborar os diagramas que compõem este projeto. Para a especificação dos Use Cases utilizamos tabelas simples que mais tarde foram geradas em código *latex*.

Quanto à segunda fase do projeto, ser-nos-à necessário desenvolver os Diagramas de Sequência e implementar o código em *Java* que dará origem ao Sistema de Gestão do Centro de Reparação. Os Diagramas de Sequência serão desenvolvidos com ajuda da ferramenta *Visual Paradigm*.

2 Abordagem dos Objetivos

De maneira a documentar os requerimentos do sistema e conseguir uma ideia geral do *UI* (User Interface) do projeto, utilizamos os Diagramas de Sequência. Assim, também percebemos a interação lógica entre os objetos no sistema aquando da interação que os chama.

Dada a diversidade de cenários, conseguimos separar e estruturar diversas interações. Com os Diagramas de Sequência, já nos é possível descrever o funcionamento interno do sistema e como este se procede no tempo.

Os Diagramas de Sequência capturam:

- A interação descrita por um Use Case;
- Interações entre o utilizador e o sistema ou entre subsistemas.

Assim, esta segunda fase do trabalho trata de modelar a complexidade do problema e implementá-lo, o que se torna mais simples com as interações já desenvolvidas.

3 Alterações Realizadas

3.1 Modelo de Domínio

3.1.1 Primeira Fase

Durante a primeira fase do projeto e para compreendermos melhor as várias entidades deste sistema, utilizamos um diagrama de domínio. Assim, também percebemos quais as relações estabelecidas entre as diferentes entidades. Este diagrama também nos ajuda a compreender o vocabulário do enunciado, criando uma terminologia comum.

Para percebermos o funcionamento do sistema quando interage com os vários atores (clientes, operários, técnicos e gestores), utilizamos *Use Cases*. Os Use Cases fornecem um *outline* de como o sistema se comporta e como responde aos pedidos realizados pelos atores. Cada Use Case é representado por uma sequência de passos simples, com a pré-condição do sistema sendo o que é tomado como garantido antes do sistema iniciar os passos a tomar para efetuar um procedimento específico e com uma pós-condição do objetivo a ser concretizado. Os Use Cases providenciam uma lista de objetivos que podem ser usados para prever o custo e complexidade do sistema.

Após realizar os *Use Cases*, há que estruturar num diagrama. *Diagramas de Use Case* são valiosos no que toca a visualizar os requerimentos de um sistema e como interagem entre si. Para além disso, ajudam a identificar fatores internos e externos que poderão influenciar o sistema e que devem ser tomados em consideração.

Assim, a primeira fase do trabalho permitiu modelar o problema e compreendê-lo de forma mais simples e clara.

Após algumas tentativas, decidimos começar o nosso diagrama de domínio pelo **Cliente**. Achamos que um Cliente pode ter vários equipamentos por reparar entregues na Loja, mas cada equipamento é único graças ao seu código de registo. Também os Clientes são únicos através do NIF que é único a cada um e que serve como identificador. Por outro lado, achamos que um Cliente pode ter mais que um contacto associado, porque pode ter um contacto telefónico ou um email associado a si, em função do pedido realizado.

Depois começamos a estudar o **Centro de reparações**, que emprega trabalhadores denominados de **Operários**. Ao analisar a situação, achamos que não faz sentido um centro de reparações ter zero funcionários, nem um operário sem centros de reparação. No entanto, achamos que um operário pode trabalhar em mais que um centro de reparações. No nosso entender, existem três tipos de Operários, cada um com ações diferentes: funcionários; técnicos e gestores. O funcionário é o operário que aparece nos cenários 1, 2 e 5. Este realiza pedidos, porém não achamos que seria relevante representar esta interação neste diagrama.

Por outro lado, as relações dos técnicos são mais complexas. Primeiro, quando o técnico interage com o pedido de orçamento, origina um plano de trabalhos. Achamos que esta relação deve ser ternária porque o plano de trabalhos, e o respetivo orçamento, só acontecem quando o técnico interage com o pedido de orçamento. A outra relação do técnico é com o pedido de reparação. Nas duas situações achamos que os pedidos devem ser feitos por apenas um técnico, para a reparação ser mais eficiente.

O último operário que descrevemos é o gestor, responsável pelo centro. Consideramos que existem vários centros de reparações, logo um gestor pode ser responsável por vários centros, mas consideramos que cada centro só pode ter um gestor. Esse gestor participa no cenário 5.

Por fim, o centro de reparações também armazena alguns registos relativos aos funcionários e equipamentos. Nós consideramos alguns destes registos necessários por causa do cenário 5 do enunciado. Apesar de não estar descrito no diagrama de domínio, estes registos incluem informação sobre algumas das ações dos operários. Um exemplo de uma informação útil é saber as reparações efetuadas por um técnico. Por fim, consideramos mais fácil interpretar o diagrama com uma entidade chamada registo e as sub-entidades associadas a ele. A outra hipótese viável para nós era o centro de reparações estar relacionada com os vários tipos de registos diretamente, sem haver uma entidade chamada registo, mas consideramos demasiado confuso. Esta confusão teria origem na quantidade de ligações que o centro de reparações teria só com registos.

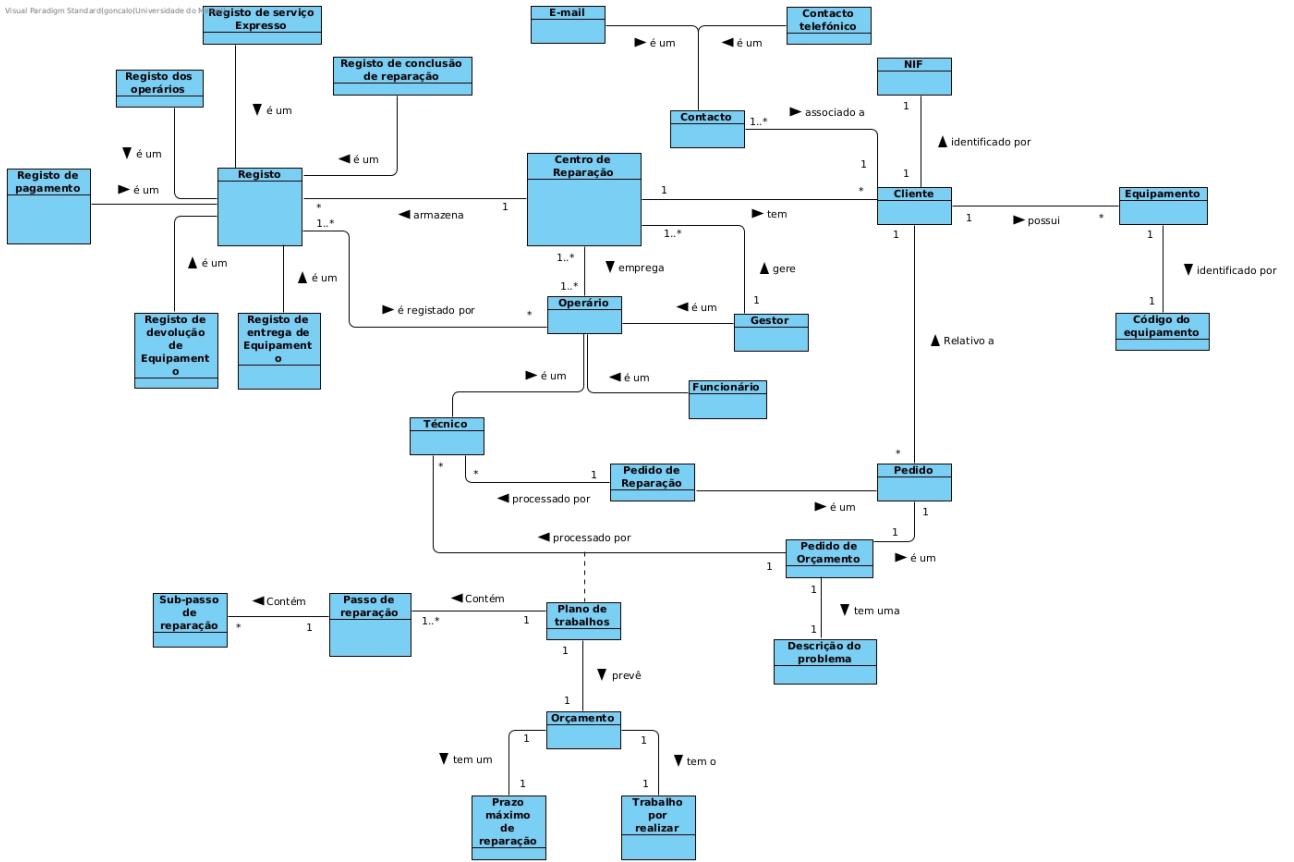


Figura 1: *Modelo de Domínio*

3.1.2 Segunda Fase

A segunda fase segue-se descrita através de um **Diagrama de componentes**, vários **Diagramas de classes** para o sistema e os seus subsistemas, alguns **Diagramas de Sequência** que realizamos para os *Use cases* que consideramos mais importantes e da **implementação** de todo o planeamento realizado anteriormente. Embora tenhamos tentado ao máximo respeitar o Diagrama de Domínio e os Use Cases, houveram detalhes que apenas nos foram apontados quando desenvolvemos os Diagramas de Sequência e o código de implementação. Assim, foram necessárias algumas alterações aos processos da primeira fase do trabalho tal como o facto de termos adicionado um novo tipo de operário, o Admin.

4 Modelo Use Case

Depois de finalizado o Modelo de Domínio, debruçámo-nos sobre o Modelo de Use Cases.

Dada a diversidade de cenários, conseguimos separar e estruturar as diversas interações. Apesar de não descrever o funcionamento interno do sistema, permite compreender o que ele faz e como os atores interagem com ele. Para além disso, é possível perceber as alternativas possíveis a um cenário normal, através de fluxos alternativos e de exceção. Após realizar os *Use Cases*, há que estruturar num diagrama. *Diagramas de Use Case* são valiosos no que toca a visualizar os requerimentos de um sistema e como interagem entre si. Para além disso, ajudam a identificar fatores internos e externos que poderão influenciar o sistema e que devem ser tomados em consideração. Portanto, o nosso passo final foi gerar os *Diagramas de Use Case*.

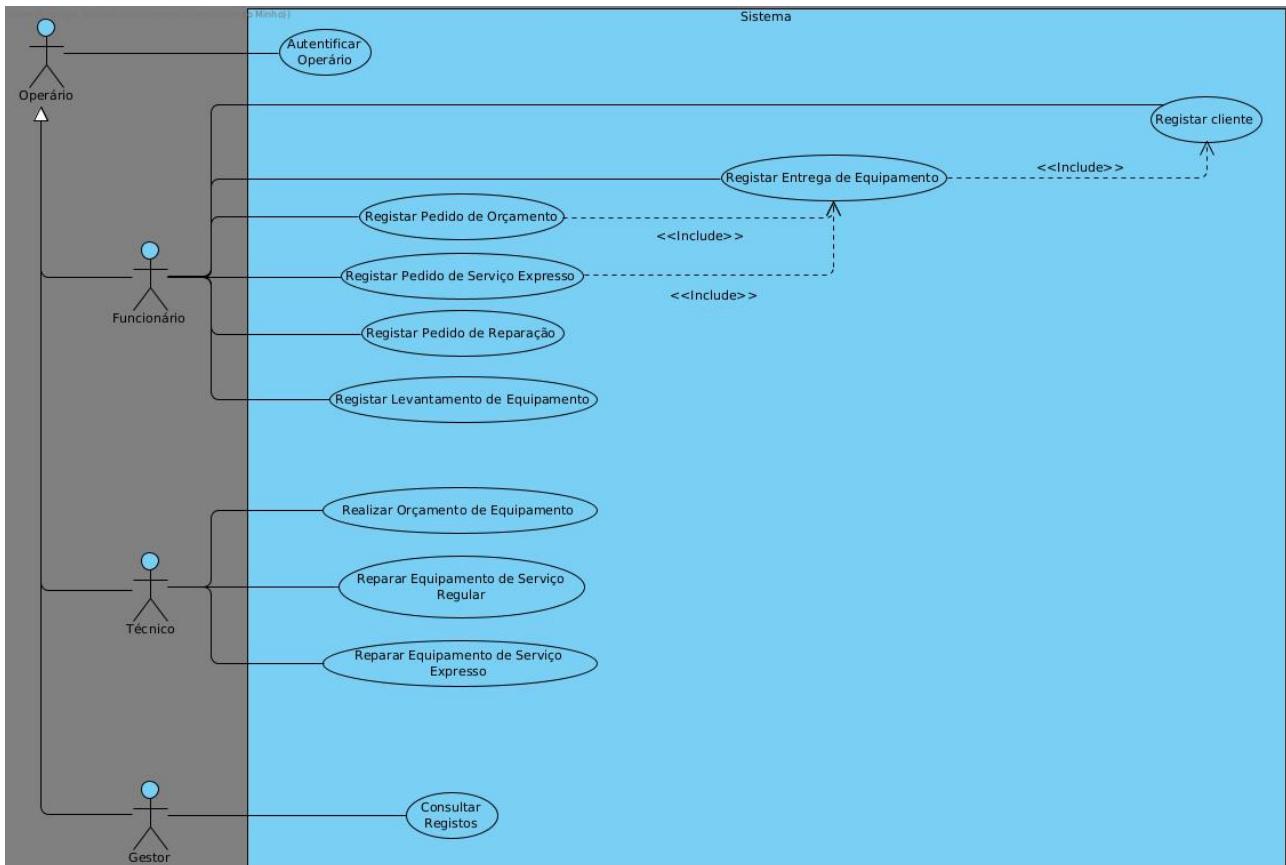


Figura 2: Diagrama de Use Case

5 Especificação de Use Cases

Iremos nesta secção proceder à especificação de cada um dos Use Cases presentes no Modelo de Use Case

5.1 Especificação de *Use Cases*

5.1.1 Autenticar Operário

Cenário	1
Ator	Operário
Pré-Condição	O Operário não está autenticado no Sistema.
Pós-Condição	O Operário fica autenticado no Sistema.

	Autor	Sistema
Fluxo Normal	1. Informa o Sistema que deseja efetuar a sua autenticação.	
		2. Apresenta o formulário de autenticação.
	3. Insere o seu identificador e senha de acesso.	
		4. Valida o par identificador/senha.
		5. Autentica o Operário.
Fluxo Exceção 1 [Os dados indicados são inválidos] (passo 4)		4.1. Informa que o par identificador/senha é inválido.

5.1.2 Registar Cliente

Cenários	1 e 2
Autor	Funcionário
Pré-Condição	O Funcionário está autenticado no Sistema.
Pós-Condição	O Sistema fica com mais um registo de Cliente.

	Autor	Sistema
Fluxo Normal	1. Informa o Sistema que deseja efetuar o registo de um Cliente.	
		2. Apresenta o formulário de registo de um Cliente.
	3. Insere o NIF, nome, email e número de telefone do Cliente.	
		4. Verifica que não existe nenhum utilizador com o NIF inserido.
		5. Guarda os dados do Cliente.
		6. Informa que o registo foi feito com sucesso.
Fluxo Exceção 1 [Já existe um utilizador com o mesmo NIF] (passo 4)		4.1. Informa que já existe um utilizador com o NIF inserido.

5.1.3 Registar Entrega de Equipamento

Cenários	1, 2 e 5
Autor	Funcionário
Pré-Condição	O Funcionário está autenticado.
Pós-Condição	O Sistema fica com mais um registo de equipamento.

	Autor	Sistema
Fluxo Normal	1. Insere o NIF do Cliente, assim como a informação do equipamento.	
		2. Cria o registo de um novo equipamento, associado ao registo do Cliente.
		3. Devolve o código de registo do equipamento.
		4. Incrementa o número de receções que o funcionário realizou.
Fluxo Alternativo 1 [O Cliente não está registado no Sistema] (passo 1)		1.1. «include» Registar Cliente. 1.2. Voltar ao passo 1.

5.1.4 Registar Pedido de Orçamento

Cenários	1
Autor	Funcionário
Pré-Condição	O Funcionário está autenticado.
Pós-Condição	O Sistema fica com mais um registo de pedido de orçamento.

	Autor	Sistema
Fluxo Normal	1. «include» Registar Entrega de Equipamento.	
		2. Regista pedido de orçamento para o dado equipamento.
		3. Informa que o registo foi feito com sucesso.

5.1.5 Realizar Orçamento de Equipamento

Cenários	1 e 3	
Autor	Técnico	
Pré-Condição	O Técnico está autenticado e existe pelo menos um equipamento à espera de orçamento.	
Pós-Condição	O Sistema fica com mais um orçamento e um plano de trabalho relativo a um equipamento.	
Fluxo Normal	Autor	Sistema
	1. Mostra todos os pedidos de orçamento pendentes numa lista.	
	2. Escolhe um pedido de orçamento da lista.	
		3. Fornece o código de registo do equipamento associado a esse pedido de orçamento.
	4. Indica que quer iniciar um plano de trabalhos para a reparação.	
		5. Cria novo plano de trabalhos para a reparação.
	6. Regista os passos e subpassos necessários para reparar o equipamento.	
	7. Regista os custos e as horas necessárias associadas a cada passo.	
		8. Calcula e regista o orçamento.
		9. Calcula e regista o prazo máximo de execução de reparação do orçamento
Fluxo Exceção 1 [Equipamento não pode ser reparado] (passo 6)	10. Remove o pedido de orçamento do equipamento da lista de pedidos de orçamento.	
	11. Envia o orçamento ao cliente, através de um email.	
	6.1. Indica que o equipamento não tem reparação.	
		6.2. Envia email a notificar que o equipamento não pode ser reparado e com pedido de levantamento do mesmo.
		6.3. Cancela plano de trabalhos para a reparação.

5.1.6 Registar Pedido de Reparação

Cenários	1
Autor	Funcionário
Pré-Condição	O Funcionário está autenticado e o Sistema recebe um email de resposta.
Pós-Condição	O Sistema fica com mais um registo de pedido de reparação.

	Autor	Sistema
Fluxo Normal		1. Redireciona o email recebido para um funcionário.
	2. Indica que o pedido de reparação foi aceite.	
		3. Regista pedido de reparação para o dado equipamento.
Fluxo Exceção 1 [Cliente não confirma a reparação num espaço de 30 dias] (passo 1)		1.1. Arquiva o orçamento do equipamento.
Fluxo Exceção 2 [Cliente recusa o orçamento] (passo 2)	2.1 Indica que o pedido de reparação foi recusado.	
		2.2. Envia email para o Cliente com pedido de recolha do equipamento.

5.1.7 Reparar Equipamento de Serviço Regular

Cenários	1, 4 e 5	
Autor	Técnico	
Pré-Condição	Existe um equipamento na lista de reparações programadas.	
Pós-Condição	É registada a conclusão de reparação de um equipamento.	
Fluxo Normal	Autor	Sistema
		1. Fornece o código do equipamento mais urgente na lista de equipamentos a reparar.
	2. Assinala os passos e tempo demorado conforme realiza a reparação, assinalando o custo das peças utilizadas.	
	3. Indica a conclusão da reparação no sistema.	
		4. Armazena a informação relativa à reparação e marca como terminada.
		5. Envia um email ao cliente a notificar a conclusão da reparação.
Fluxo Exceção 1 [Reparação interrompida] (passo 2)	2.1. Indica a interrupção da reparação.	
		2.2. Insere o equipamento na lista de equipamento a reparar.
Fluxo Alternativo 2 [Orçamento ultrapassado em 120%] (passo 2)		2.1. Mostra o contacto do Cliente.
		2.2. Regista a data e hora do contacto.
		2.3. Regista o identificador do Técnico que estava a reparar o equipamento.
Fluxo Exceção 3 [Cliente rejeita continuação da reparação] (passo 2.4 do Fluxo Alternativo 2)	2.4. Indica confirmação do cliente para continuar a reparação.	
	2.5. Voltar ao passo 2.	
		2.4.1. Envia email para o Cliente com pedido de recolha do equipamento.

5.1.8 Registar Levantamento de Equipamento

Cenários	1
Autor	Funcionário
Pré-Condição	O Funcionário está autenticado e existe um registo do Cliente no Sistema.
Pós-Condição	O Sistema fica com mais um registo de entrega e pagamento de equipamento.

	Autor	Sistema
Fluxo Normal	1. Insere o NIF do Cliente.	2. Indica os equipamentos a ser recolhidos, assim como o preço da sua reparação.
	3. Indica que o equipamento foi levantado e pago.	4. Regista o equipamento como levantado e pago.
		5. Incrementa o número de entregas que o funcionário realizou.
Fluxo Exceção 1 [O Cliente não vai levantar o equipamento no espaço de 90 dias] (passo 1)		1.1. Regista o equipamento como abandonado.

5.1.9 Registar Pedido de Serviço Expresso

Cenários	2 e 5
Autor	Funcionário
Pré-Condição	O Funcionário está autenticado no Sistema.
Pós-Condição	O Sistema fica com mais um pedido de serviço expresso.

	Autor	Sistema
Fluxo Normal		1. Verifica que há disponibilidade para a realização do serviço de imediato.
	2. «include» Registar Entrega de Equipamento.	
		3. Regista pedido de serviço expresso para o dado equipamento.
		4. Informa que o registo foi feito com sucesso.
		5. Incrementa o numero de atendimentos que o funcionário realizou.
Fluxo Exceção 1 [Não existe disponibilidade para realizar o serviço de imediato] (passo 2)		1.1. Informa que não é possível realizar o serviço de imediato.

5.1.10 Reparar Equipamento de Serviço Expresso

Cenários	2, 4 e 5
Autor	Técnico
Pré-Condição	Existe equipamento a aguardar por reparação expresso.
Pós-Condição	É registada a conclusão de reparação de um equipamento e o Cliente é notificado por SMS.

	Autor	Sistema
Fluxo Normal		1. Fornece o código do equipamento a reparar.
	2. Indica a conclusão da reparação.	3. Regista a conclusão da reparação.
		4. Envia SMS ao Cliente a notificar a conclusão do serviço expresso.
		5. Atualiza os registo relativos ao técnico com esta nova reparação.

5.1.11 Consultar Registros

Cenários	5
Autor	Gestor
Pré-Condição	O Gestor está registado no Sistema.
Pós-Condição	O Sistema apresenta a informação para análise.

	Autor	Sistema
Fluxo Normal	1. Pede a informação sobre o funcionamento do Centro de Reparações.	
		2. Apresenta listas com informação sobre os Funcionários e Técnicos.
	3. Escolhe uma das listas para análise	
		4. Apresenta informação mais detalhada da lista escolhida.

6 Diagrama de Componentes

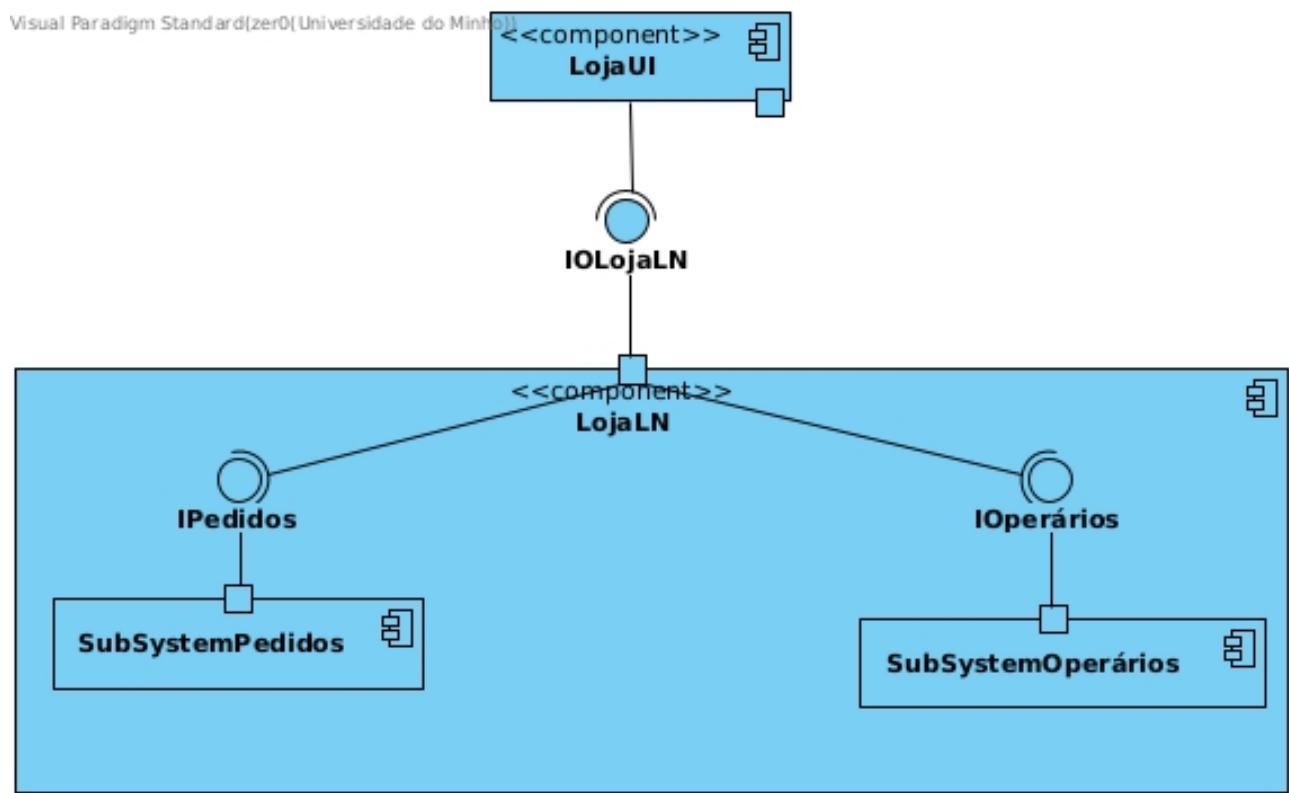


Figura 3: Diagrama de Componentes

7 Diagrama de Classes

Com base no modelo de Domínio, foram selecionadas as componentes que dariam origem a classes, como por exemplo, *GestLoja* e *Operario*. Para definir a interface da lógica de negócio, foram analisados todos os *Use Case's*. Foram também identificados os subsistemas e a sua definição. Seguidamente, foram modeladas as classes com atributos e operações. Deste processo resultaram os seguintes Diagramas de Classes.

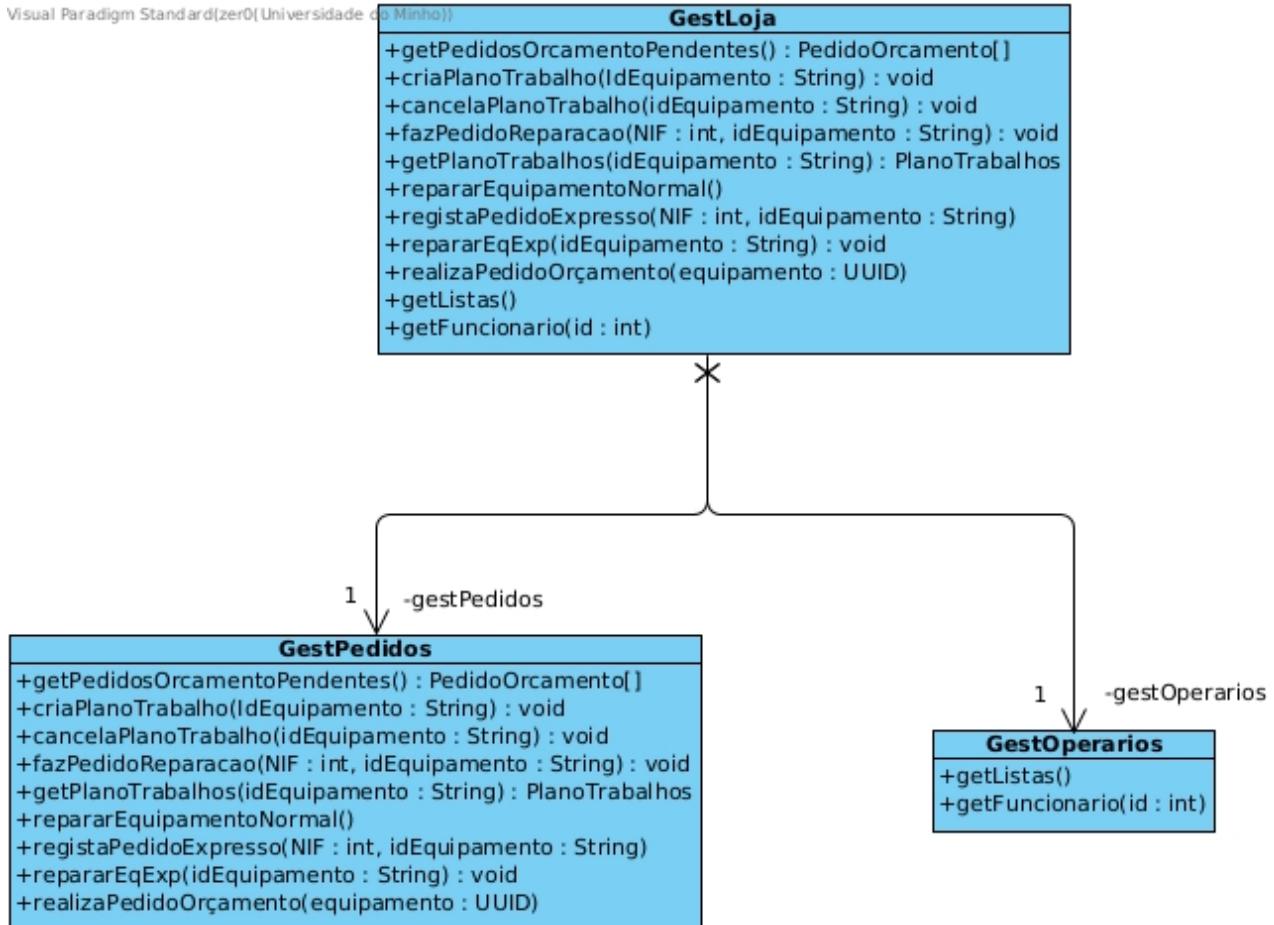


Figura 4: Classe Loja

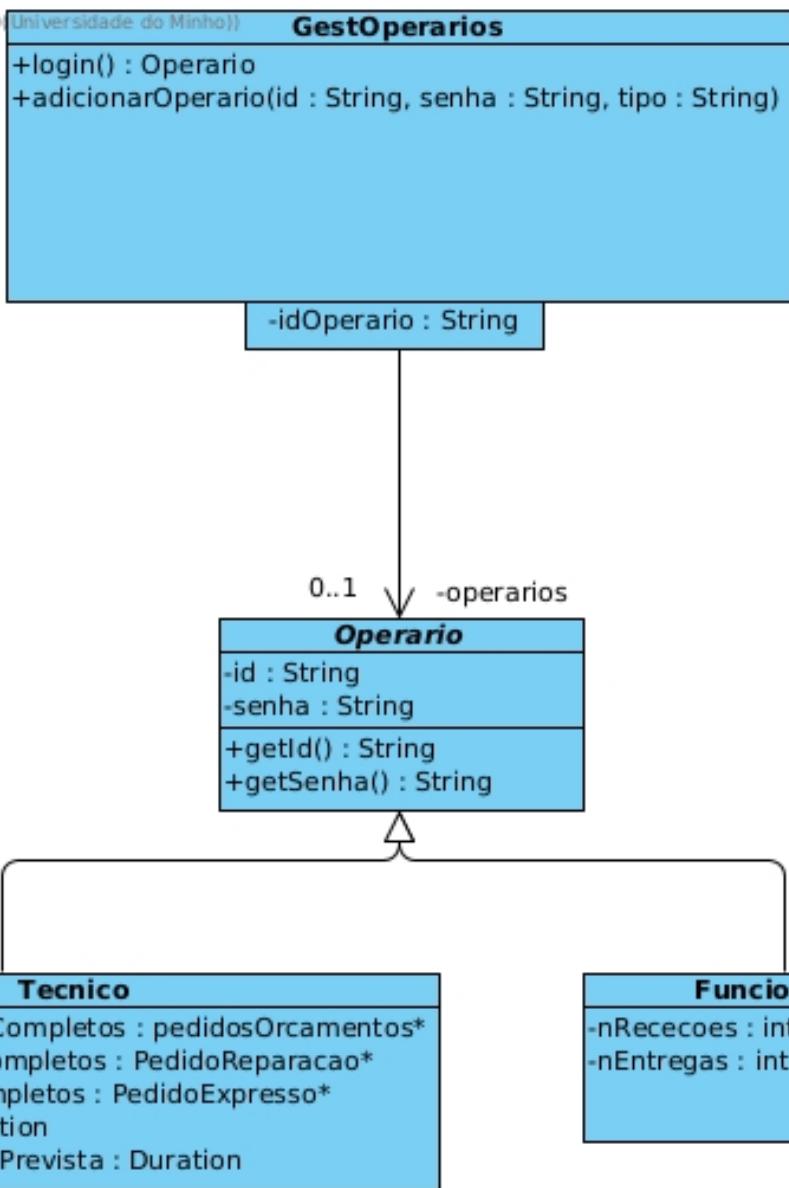


Figura 5: Classe Operários

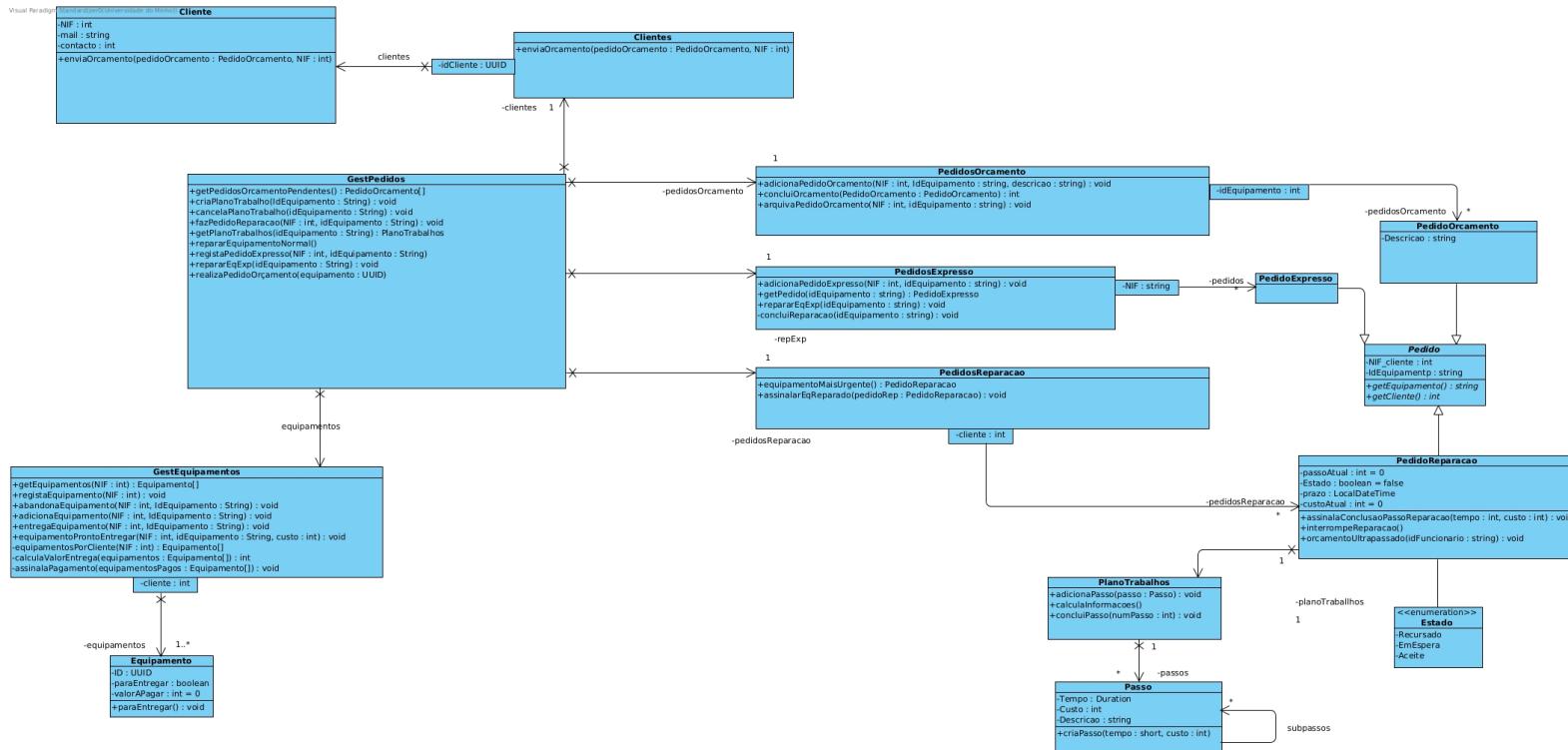


Figura 6: Classe Pedidos

8 Diagramas de Sequência de Implementação

Seguindo os Diagramas de Sequência de Subsistemas, foram desenvolvidos os Diagramas de Sequência de implementação. Estes distinguem-se por serem uma refinação dos anteriores. Nesta fase, já se referenciam as classes e as suas instâncias específicas. É também já utilizada uma sintaxe parecida à que será utilizada na implementação Java.

Estes diagramas fazem com que o processo de transição para a implementação do projeto esteja já muito perto de acontecer. Estruturalmente, já está tudo muito bem definido e, após a geração automática dos packages e classes no IDE do Java, irá apenas faltar desenvolver as funções que realizam os vários passos necessários ao bom funcionamento da aplicação.

8.1 Autenticar Operário

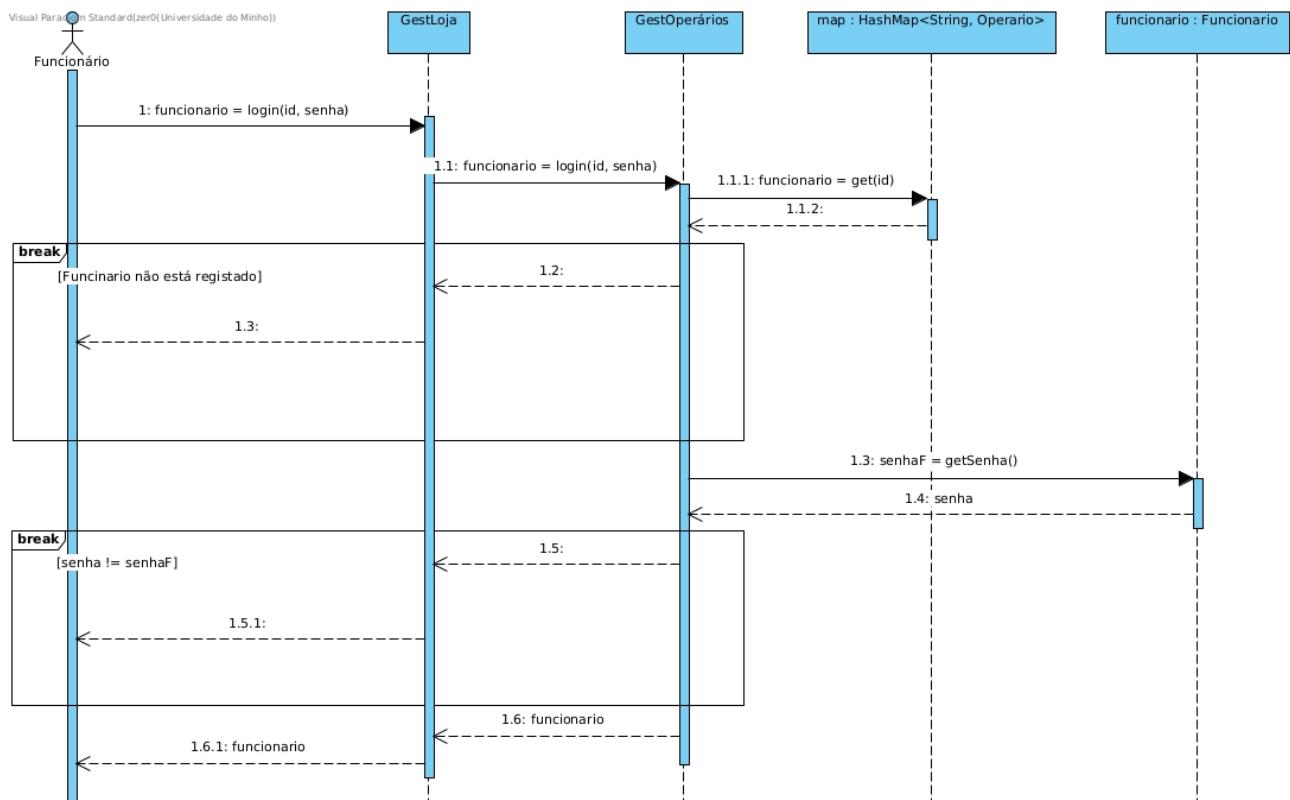


Figura 7: Autenticar Operário

8.2 Registrar Cliente

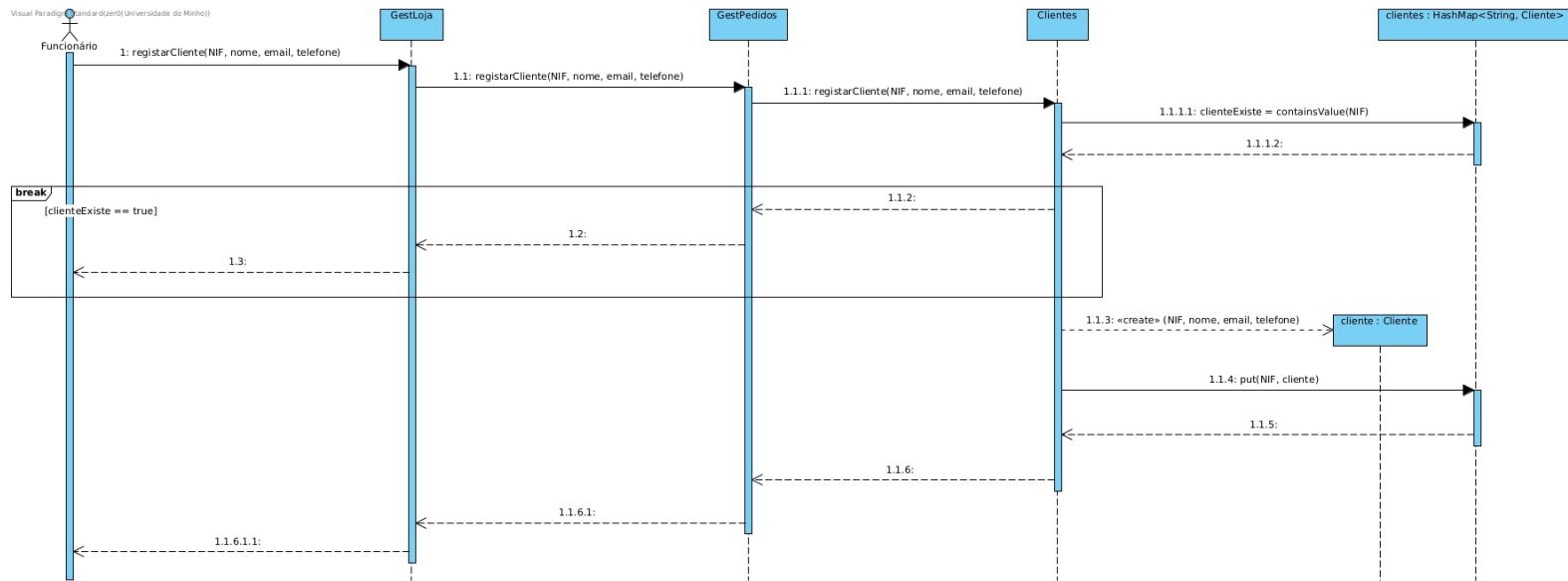


Figura 8: Registrar Cliente

8.3 Registar Entrega de Equipamento

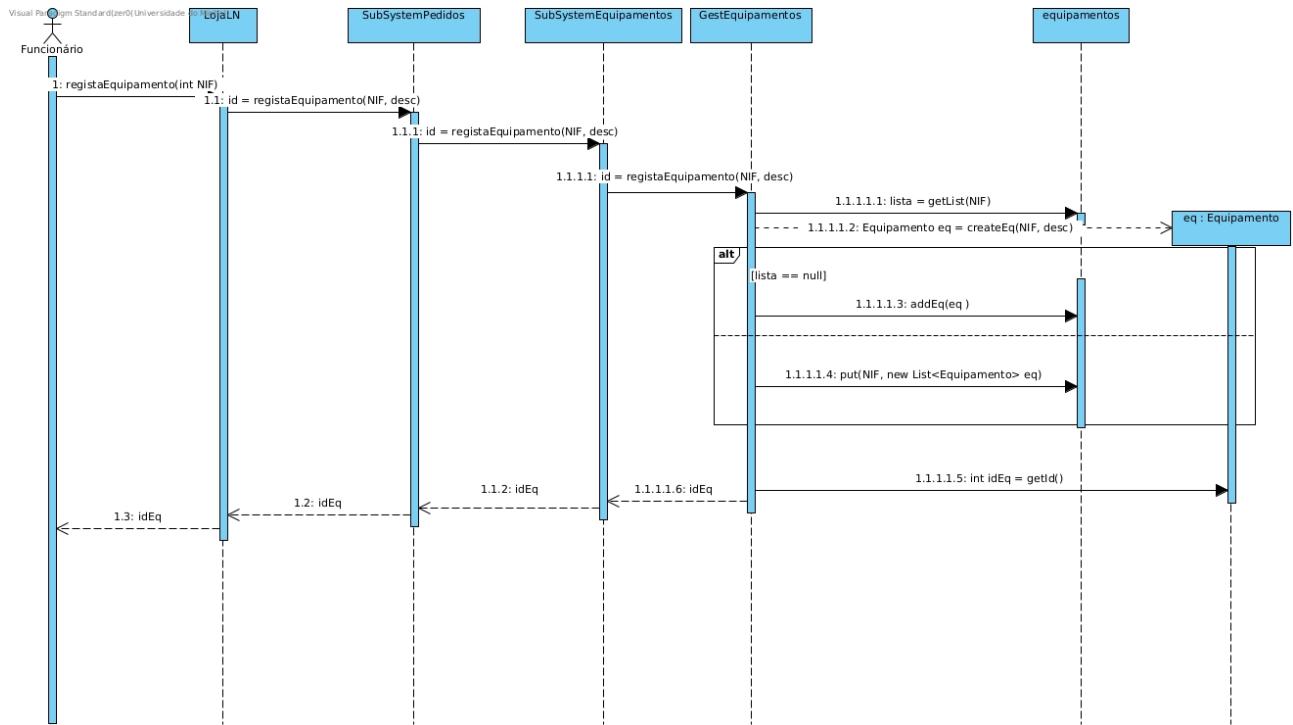


Figura 9: *Registrar Entrega de Equipamento*

8.4 Realizar Orçamento de Equipamento

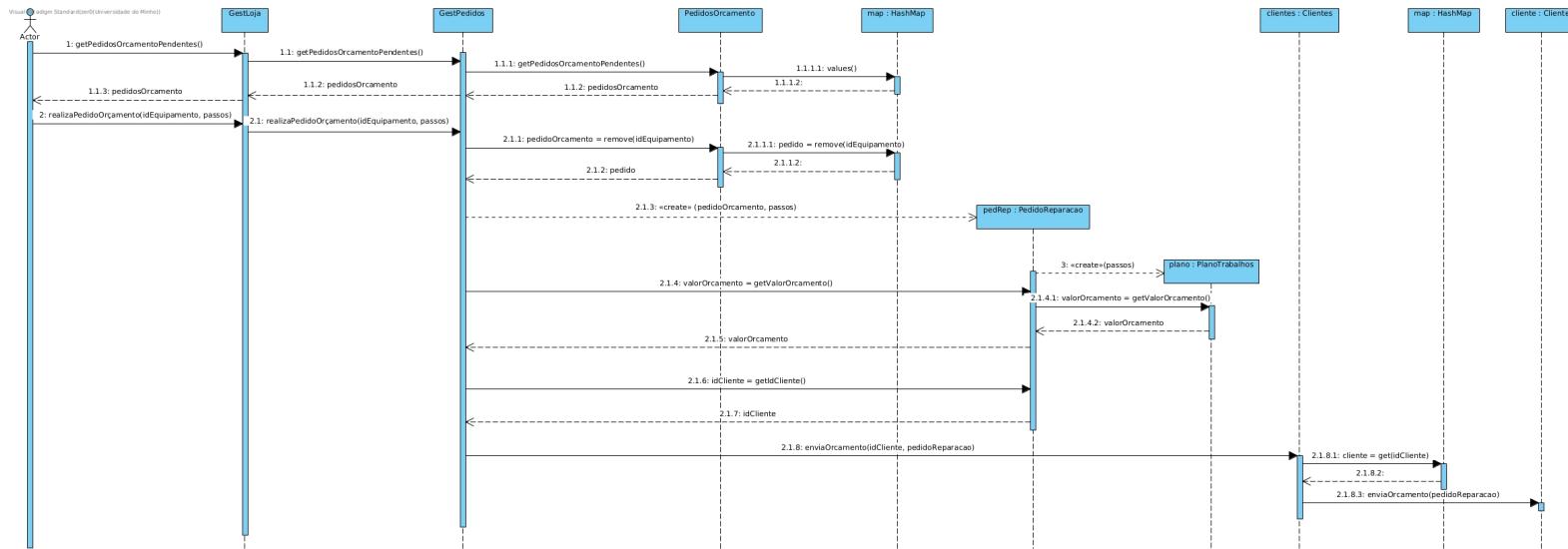


Figura 10: Realizar Orçamento de Equipamento

8.5 Registar Pedido de Reparação

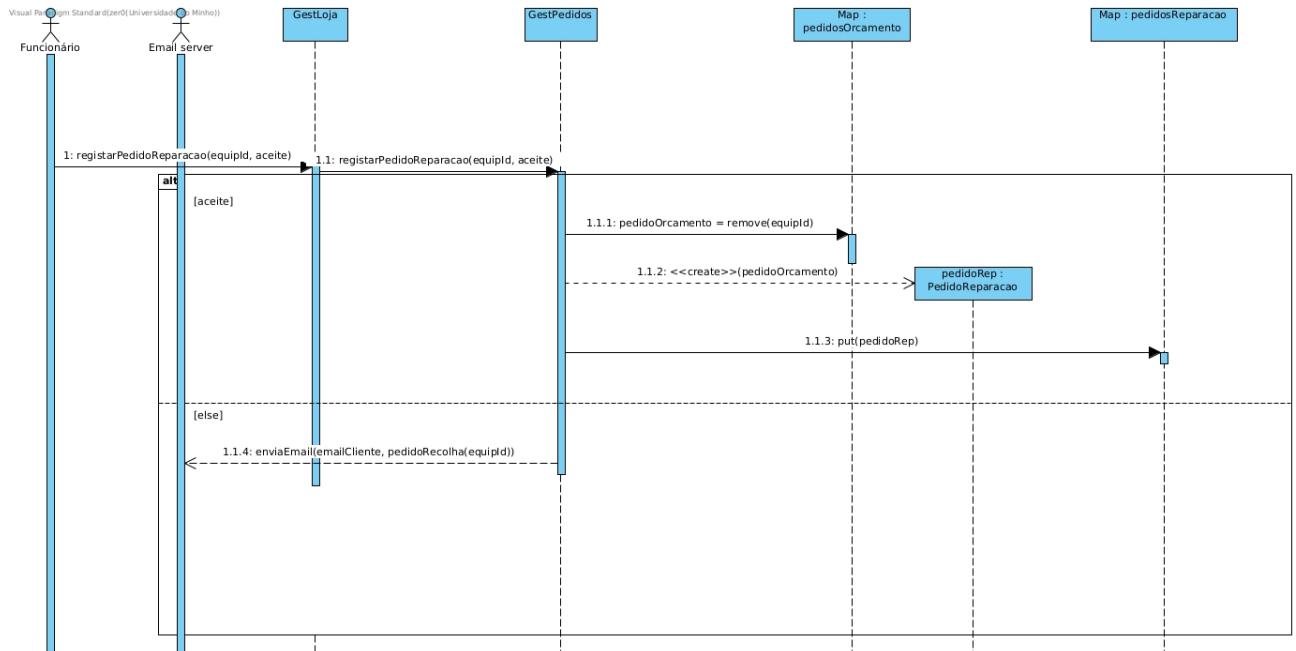


Figura 11: Registar Pedido de Reparação

8.6 Registar Levantamento de Equipamento

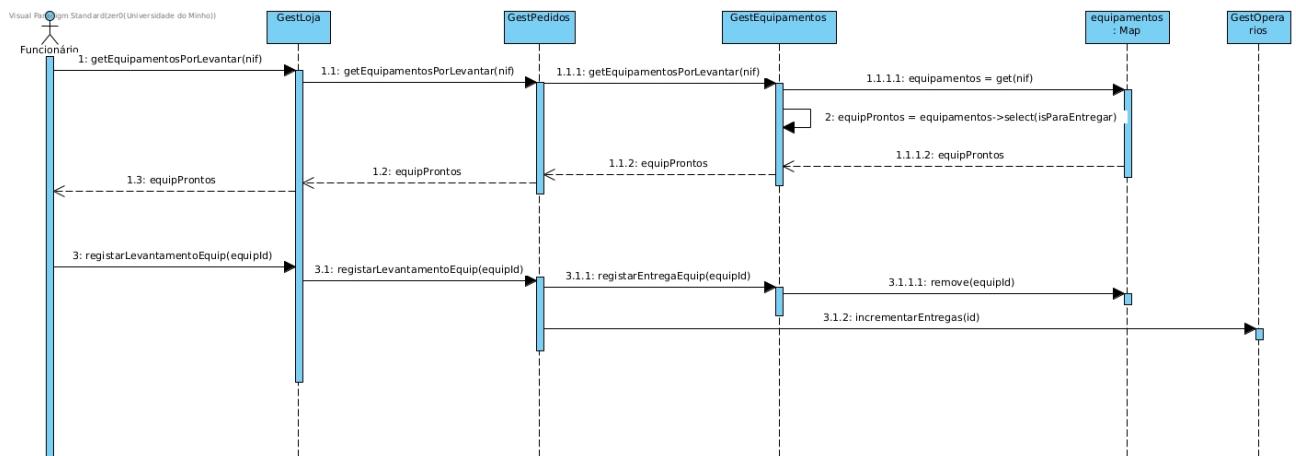


Figura 12: Registar Levantamento de Equipamento

9 Base de Dados

De maneira ao Software ser posto em prática, necessitamos de dados que possam ser utilizados, manipulados e analisados. Assim, o nosso grupo gerou duas classes: *PedidosDAO* e *OperariosDAO*, uma para os Operários e outra para os Pedidos. Deste modo, os dados são carregados em memória a partir de dois ficheiros existentes. Caso os ficheiros não existam, estas duas classes retornam uma instância vazia, no sentido de não conterem operarios ou pedidos já carregados mas durante a execução do programa terem a possibilidade de adicionar tais. Posteriormente, ao sair do programa, os dados voltam a ser guardados num ficheiro já com as alterações que foram realizadas durante a execução do programa.

10 Implementação

Em termos da implementação do projeto, o primeiro passo foi a geração automática das *packages* e classes no IDE através dos vários Diagramas que tinham sido desenvolvidos anteriormente. Neste momento, deveriam já ter sido desenvolvidos todos os métodos necessários.

O segundo passo, foi então a implementação dos métodos, de cada função. Os Diagramas de Sequência serviram de grande ajuda neste passo pois já tínhamos a sequência cronológica do que seria preciso acontecer, faltava então fazer funções que aplicassem esses processos.

Por fim, construímos os Menus que levam ao procedimento de cada necessidade do Funcionário, Técnico ou Gestor.

11 Conclusão e Análise Crítica

Revendo o tema dado para este trabalho prático, conseguimos compreender todo o processo que gira em torno do funcionamento e modelação de um Sistema de Gestão para Centros de Reparação de equipamentos eletrónicos. Para além disso, aprendemos a utilizar ferramentas importantes para o planeamento de um sistema de software, como *Diagramas de Sequência* e *Diagramas de Packages*. Também desenvolvemos as nossas competências de implementação de um programa.

Com este trabalho podemos concluir que a modelação é uma excelente ferramenta no auxílio do desenvolvimento de software de média a grande dimensão visto que permite haver um faseamento do desenvolvimento, o que facilita a correção de erros em qualquer momento, sem implicar grandes custos. A modelação também permite a deteção de erros numa fase inicial, facilitando a sua correção.

Do nosso ponto de vista, o desempenho que demonstramos neste trabalho é positivo, uma vez que conseguimos obter um resultado final completamente modelado e funcional. Também se encontra de acordo com os requisitos.

A nossa maior dificuldade foi a criação dos diagramas e modelos, visto que há maneiras diferentes de modelar e todas podem estar corretas, o que nos atrapalhou um bocado e nos levou a fazer algumas modificações recorrentes nos diagramas (especialmente os de sequência).

Finalmente, com este trabalho, conseguimos compreender melhor e aplicar a matéria lecionada na Unidade Curricular de Desenvolvimento de Sistemas de Software.