

UNIVERSIDADE DO MINHO

MESTRADO INTEGRADO EM ENGENHARIA INFORMÁTICA

Desenvolvimento de Sistemas de Software - Trabalho
Prático (Fase Final) - Ano Letivo 2021/2022
Grupo 42

Gonçalo Braz (a93270)

Simão Cunha (a93262)

Tiago Silva (a93277)

Gonçalo Pereira (a93168)

6 de janeiro de 2022



Conteúdo

1	Introdução	3
2	Alterações na fase Intermédia	3
2.1	Modelo de Domínio	3
2.2	Descrição dos <i>Use Cases</i>	3
3	Descrição do trabalho efetuado na fase final	6
3.1	Diagrama de componentes	6
3.2	Diagrama de classes	7
3.3	Diagramas de subssistemas	9
3.3.1	DS #1: Gerar orçamento	9
3.3.2	DS #2: Adicionar pedido de orçamento	10
3.3.3	DS #3: Adicionar pedido de serviço Expresso	10
3.3.4	DS #4: Processar reparação	11
3.3.5	DS #5: Concluir pedido	12
3.3.6	DS #6: Verificar disponibilidade para efetuar serviço Expresso	12
3.3.7	DS #7: Reparar pedido Expresso	13
3.3.8	DS #8: Confirmar orçamento	13
3.3.9	DS #9: Recusar orçamento	13
3.3.10	DS #10: Adicionar cliente	14
3.3.11	DS #11: Executar passo	14
3.3.12	DS #12: Executar subpasso	15
3.4	Implementação	15
4	Conclusão	21

Neste caso de uso, sentimos a necessidade de dizer concretamente quais os dados que são necessários para adicionar um novo utilizador ao sistema: ID, nome e password - algo que não aparecia no *use case* do 1º relatório.

Use Case:	Adicionar novo utilizador	
Ator:	Gestor	
Pré-condição	Gestor autenticado no sistema	
Pós-condição	É adicionado um novo utilizador na camada de persistência de dados do sistema	
	Ator	Sistema
Fluxo Normal	2. Fornece o ID, o nome e a password	1. Pede o ID, o nome e a password do utilizador
		3. Valida o ID, o nome e a password
		4. Adiciona o novo utilizador na camada de persistência
		5. Confirma sucesso na inserção do utilizador
Fluxo Alternativo: 1. (Um dos dados fornecidos é inválido) [passo 3]		3.1. Informa a invalidade de um dos dados inseridos
		3.2. Regressa ao passo 1

Figura 2: *Use Case*: Adicionar novo utilizador

Neste *use case*, reparámos que não indicávamos quais os elementos que são necessários para registar um cliente: NIF, nome, número de telemóvel e email. Procedemos, assim à sua retificação.

Use Case:	Registar novo cliente	
Ator:	Funcionário/Gestor	
Pré-condição	Utilizador autenticado no sistema	
Pós-condição	É adicionado um novo cliente na camada de persistência de dados do sistema	
	Ator	Sistema
Fluxo Normal	2. Fornece o NIF, o nome, o número de telemóvel e o email	1. Pede o NIF, o nome, o número de telemóvel e o email do cliente
		3. Valida o NIF, o nome, o número de telemóvel e o email
		4. Adiciona o novo cliente na camada de persistência
		5. Confirma sucesso na inserção do cliente
Fluxo Alternativo: 1. (Um dos dados fornecidos é inválido) [passo 3]		3.1. Informa a invalidade de um dos dados inseridos
		3.2. Regressa ao passo 1
Fluxo Alternativo: 2. (Cliente já existe) [passo 4]		4.1. Informa que cliente já existe registado na camada de persistência
		4.2. Substitui os antigos dados pelos fornecidos
		4.3. Regressa ao passo 5

Figura 3: *Use Case*: Registar novo cliente

Aqui, observámos que não apontámos quais os dados a utilizar para a autenticação do sistema. Assim, os dados necessários são o ID do utilizador e a sua password.

Use Case:	Autenticar no sistema	
Ator:	Funcionário/Técnico/Gestor	
Pré-condição	Utilizador existe na camada de persistência de dados do sistema	
Pós-condição	Acesso ao sistema e às funcionalidades permitidas	
	Ator	Sistema
Fluxo Normal	2. Fornece o ID e a password	1. Pede o ID e a password do utilizador
		3. Valida o ID e a password
		4. Permite acesso do utilizador ao sistema
Fluxo Alternativo: 1. (Os dados fornecidos são inválidos) [passo 3]		3.1. Informa a invalidade dos dados inseridos
		3.2. Regressa ao passo 1

Figura 4: *Use Case*: Autenticar no sistema

Quando estamos a registar o pedido de orçamento, foi necessário indicar os dados requeridos pelo sistema - NIF do cliente, o equipamento e uma descrição do pedido - pelo que procedemos à sua alteração.

Use Case:	Registar pedido de orçamento	
Ator:	Funcionário	
Pré-condição	Utilizador previamente autenticado	
Pós-condição	Sistema fica com o registo de pedido de orçamento e equipamento	
	Ator	Sistema
Fluxo Normal	2. Fornece o NIF do cliente, equipamento e descrição do pedido	1. Pede o NIF do cliente, equipamento e descrição do pedido 3. Verifica se o cliente existe na camada de persistência de dados do sistema 4. Regista o pedido de orçamento e respetivo equipamento
Fluxo Alternativo: 1. (Cliente não existe na camada de persistência) [passo 3]		3.1. Informa que cliente não existe na camada de persistência 3.2. Regressa ao passo 1

Figura 5: *Use Case*: Registar pedido de orçamento

Neste *use case*, tivemos a necessidade, tal como nos casos de uso anteriores, de especificar quais os dados a utilizar. No entanto, aqui serão necessários o NIF do cliente, o equipamento, o tipo do pedido e a descrição do pedido para efetuar o serviço Expresso. Também sentimos a necessidade verificar a existência do cliente na camada de persistência de dados para continuar com o fluxo da atividade.

Use Case:	Registar pedido do serviço Expresso	
Ator:	Funcionário/Gestor	
Pré-condição	Utilizador previamente autenticado	
Pós-condição	Sistema fica com o registo de serviço Expresso	
	Ator	Sistema
Fluxo Normal	2. Fornece o NIF do cliente, o equipamento, o tipo do pedido e a descrição do pedido	1. Pede o NIF do cliente, o equipamento, o tipo do pedido e a descrição do pedido 3. Verifica se o cliente existe na camada de persistência de dados do sistema 4. Verifica a disponibilidade para efetuar o serviço 5. Aceita o pedido de serviço Expresso
Fluxo Alternativo: 1. (Cliente não existe na camada de persistência) [passo 3]		3.1. Informa que cliente não existe na camada de persistência 3.2. Regressa ao passo 1
Fluxo Exceção: 1. (Não existe disponibilidade) [passo 1]		1.1. Informa que não existe disponibilidade para efetuar serviço Expresso

Figura 6: *Use Case*: Registar pedido de serviço Expresso

Neste caso de uso, decidimos retirar o fluxo de exceção referente à impossibilidade de o equipamento não poder ser reparado oriundo do passo 3 do fluxo normal com o objetivo de poder simplificar na implementação.

Use Case:	Criar orçamento	
Ator:	Técnico	
Pré-condição	Utilizador previamente autenticado	
Pós-condição	Sistema fica com o registo do orçamento no sistema e com menos um registo na lista de pedidos de orçamentos	
	Ator	Sistema
Fluxo Normal	2. Escolhe o pedido 3. Fornece o plano de trabalhos para a reparação	1. Apresenta lista de pedidos de orçamentos 4. Cria o orçamento 5. Regista o orçamento 6. Envia orçamento para o email do cliente 7. Retira registo de pedido da lista de pedidos de orçamentos
Fluxo Exceção: 1. (Não existe pedidos de orçamentos na lista do sistema) [passo 1]		1.1. Informa que não existe pedidos de orçamento 1.2. Recusa a criação de um orçamento
Fluxo Alternativo: 1. (Sistema invalida o plano de trabalhos) [passo 3]		3.1. Informa que o plano de trabalhos é inválido 3.2. Pede novo plano de trabalhos 3.3. Regressa ao passo 3

Figura 7: *Use Case*: Criar orçamento

Nesta descrição, reparámos que tínhamos no antigo diagrama 2 fluxos de exceção iguais, pelo que procedemos à sua retificação.

Use Case:	Registrar a resposta do cliente sobre o orçamento	
Ator:	Funcionário	
Pré-condição:	Cliente responde ao orçamento	
Pós-condição:	Sistema possui mais um equipamento na lista de equipamentos a reparar	
	Ator	Sistema
Fluxo Normal		1. Recebe resposta positiva do cliente 2. Adiciona equipamento do cliente à lista de equipamentos a reparar
Fluxo Alternativo: 1. (Sistema não recebe confirmação de orçamento no espaço de 30 dias) [passo 1]		2.1. Não recebe confirmação de orçamento num espaço de 30 dias 2.2. Arquivo orçamento 2.3. Começa a contagem do prazo de 90 dias para a recolha do equipamento 2.4. Faz checkout ao equipamento
Fluxo Alternativo: 2. (Cliente recusa orçamento) [passo 1]		2.1.b. Regista recusa do orçamento 2.2.b. Solicita recolha do equipamento ao cliente 2.3.b. Começa a contagem do prazo de 90 dias para a recolha do equipamento 2.4.b. Faz checkout ao equipamento
Fluxo Exceção: 1. (Prazo de 90 dias excedido) [passo 2.3]		2.3.1. Coloca equipamento numa lista de equipamentos abandonados

Figura 8: *Use Case*: Registrar a resposta do cliente sobre o orçamento

Aqui, decidimos alterar o fluxo de exceção referente ao cancelamento da reparação por falta de tempo/peças para um fluxo alternativo onde o utilizador coloca a reparação em espera por falta de peças/tempo, tendo em conta que a sua implementação é relativamente mais fácil.

Use Case:	Processamento da reparação	
Ator:	Técnico	
Pré-condição:	Utilizador previamente autenticado	
Pós-condição:	Sistema fica com o registo de reparação e com menos um registo na lista de equipamentos a reparar	
	Ator	Sistema
Fluxo Normal	1. Acede à lista de equipamentos a reparar 2. Escolhe o equipamento 3. Regista a execução dos passos da reparação 4. Regista a conclusão da reparação	3. Compara custo total da reparação com o orçamento 5. Retira registo do equipamento da lista de equipamentos a reparar 6. Cria custo total da reparação 7. Regista a reparação
Fluxo Alternativo: 1. Custo da reparação ultrapassa em 120% [passo 3]		3.1. Notifica cliente do excesso 3.2. Aguarda resposta do cliente 3.3. Continua passo 3
Fluxo Alternativo: 2. (Serviço Expresso) [passo 3]		3.1. Regressa ao passo 4
Fluxo Exceção: 1. (Cliente recusa continuar com reparação devido ao excesso no orçamento) [passo 3.3]		3.3.1. Informa a recusa do cliente em continuar com a reparação 3.3.2. Retira registo do equipamento da lista de equipamentos a reparar
Fluxo Exceção: 2. (Não existem registo na lista de equipamentos a reparar) [passo 1]		1.1. Informa que não existe equipamentos para serem reparados 1.2. Recusa o processamento da reparação
Fluxo Alternativo: 3. (Utilizador põe em espera a reparação por faltas de peças/tempo) [passo 3]		3.1. Regressa ao passo 1

Figura 9: *Use Case*: Processamento da reparação

3 Descrição do trabalho efetuado na fase final

3.1 Diagrama de componentes

Incluindo-se na modelação estrutural da aplicação, o *diagrama de componentes* permite identificar os componentes do sistema, mostrando as dependências entre eles. Cada componente comunica com outro através de *interfaces*. Decidimos, então, identificar 4 subsistemas na lógica de negócio (Clientes, Utilizadores, Equipamentos e Logs).

Assim, elaboramos o seguinte diagrama:

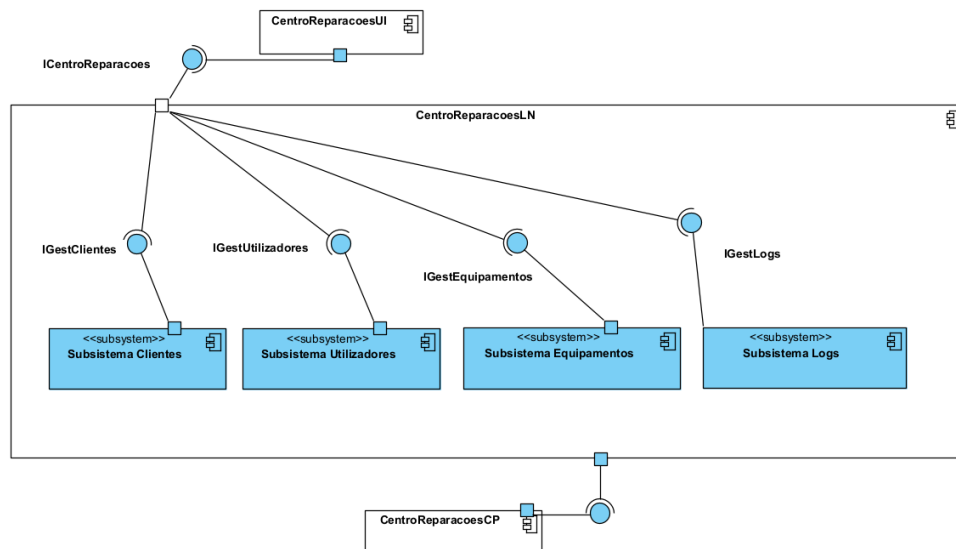


Figura 10: Diagrama de componentes

3.2 Diagrama de classes

Inserindo-se na modelação estrutural do sistema, o *diagrama de classes* tem como função agrupar classes de forma a criar um subsistema. Uma classe serve para facilitar a reutilização de código através da reutilização de classes previamente desenvolvidas, e para facilitar a manutenção da aplicação — a aplicação deve ser desenvolvida de forma a que a alteração de uma classe tenha o menor impacto possível no resto do sistema.

Uma omissão no diagrama de classes é a interface Carregavel que é estendida por quase todas as interfaces e, por este motivo não incluímos devido a esta causar poluição visual no diagrama e pois esta apenas serve para carregar/guardar os dados na camada de persistência do programa.

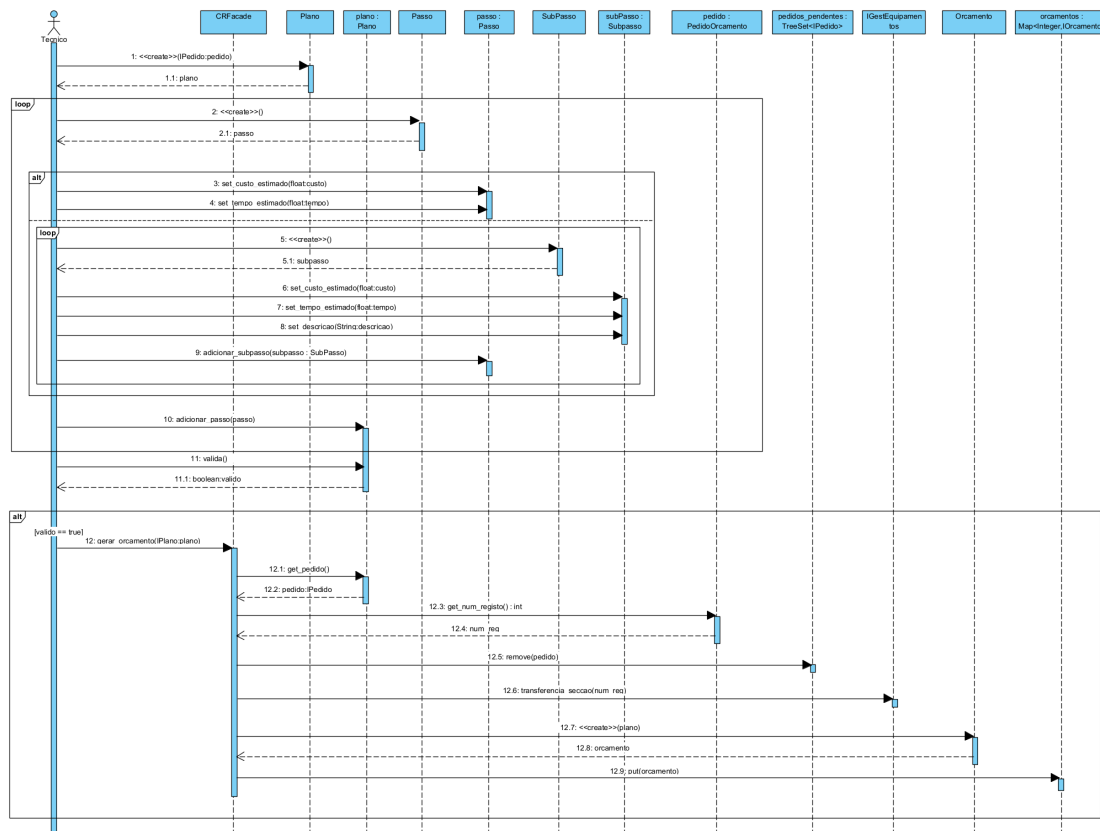
```
public interface Carregavel {
    void carregar(String string);
    String salvar();
}
```

Figura 13: Diagrama de classes final

3.3 Diagramas de subssistemas

Enquadrando-se na modelação comportamental do programa, os **diagramas de subssistemas** descrevem as interacções entre os objetos para atingir um objetivo, focando-se no ordenamento temporal das trocas de mensagens.

3.3.1 DS #1: Gerar orçamento

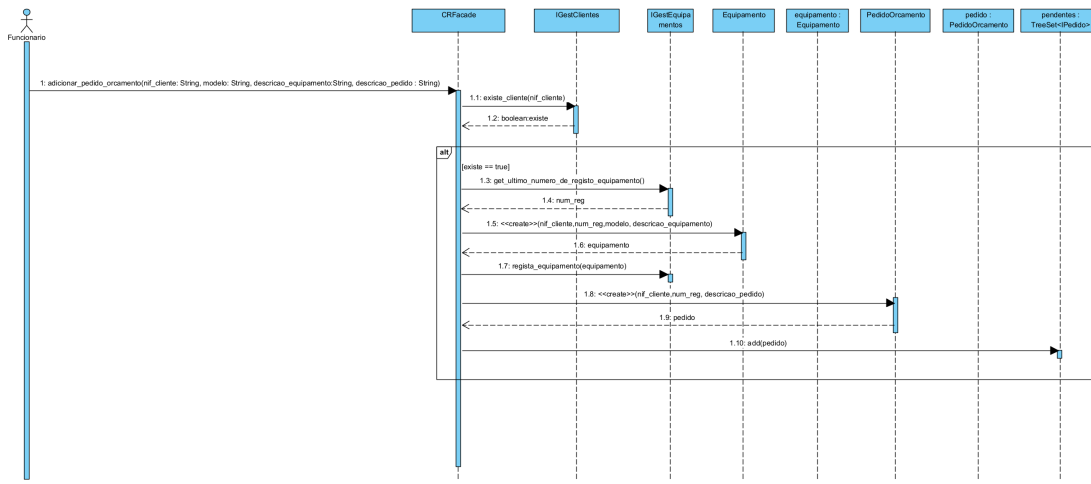


Escolhido um pedido da lista de pedidos pendentes, o técnico irá criar um orçamento para o mesmo. Para isto ele vai formular um plano onde, ele irá criar passos, podendo a estes adicionar

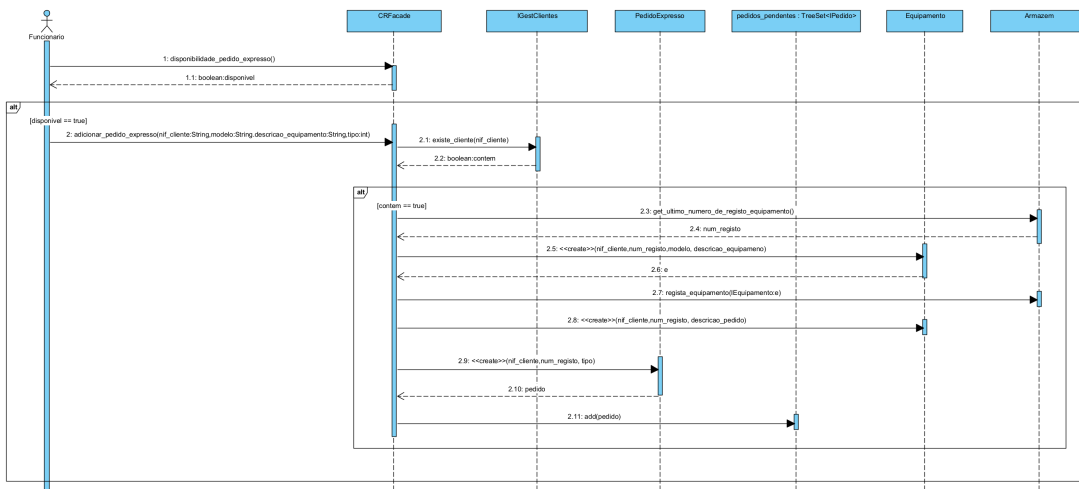
subpassos. Para criar um subpasso ou um passo sem subpassos terá de fornecer um tempo esperado de realização, um custo esperado e uma descrição.

Caso o plano seja válido, o técnico terá a opção de gerar o orçamento, adicionando-o ao conjunto de orçamentos e o pedido é movido de pendente para o conjunto dos pedidos planeados, conjunto este onde ele espera confirmação por parte do cliente.

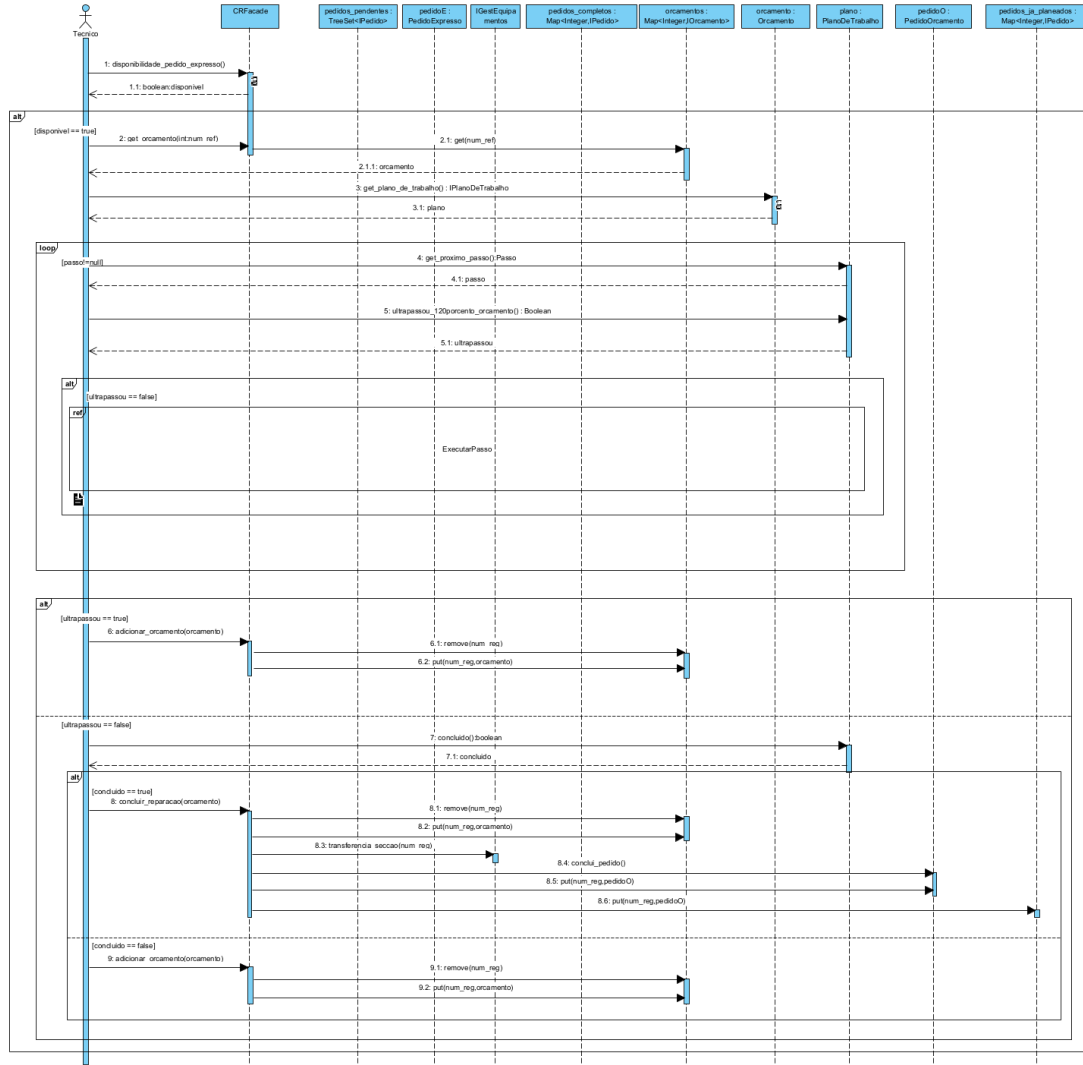
3.3.2 DS #2: Adicionar pedido de orçamento



3.3.3 DS #3: Adicionar pedido de serviço Expresso



3.3.4 DS #4: Processar reparação



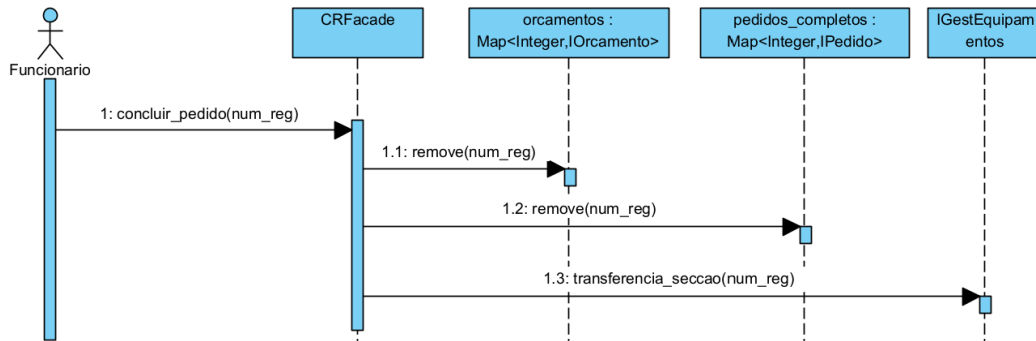
Escolhido um pedido de orçamento confirmado, o técnico irá começar a reparação. Isto implica a execução dos vários passos e correspondentes subpassos do plano.

De igual forma como quando eles são criados, os subpassos e os passos simples requerem que o técnico forneça um tempo e um custo, porém desta vez serão os valores reais da reparação. Ele poderá executar estes apenas enquanto que não ultrapassar em 120% o orçamento definido para o plano.

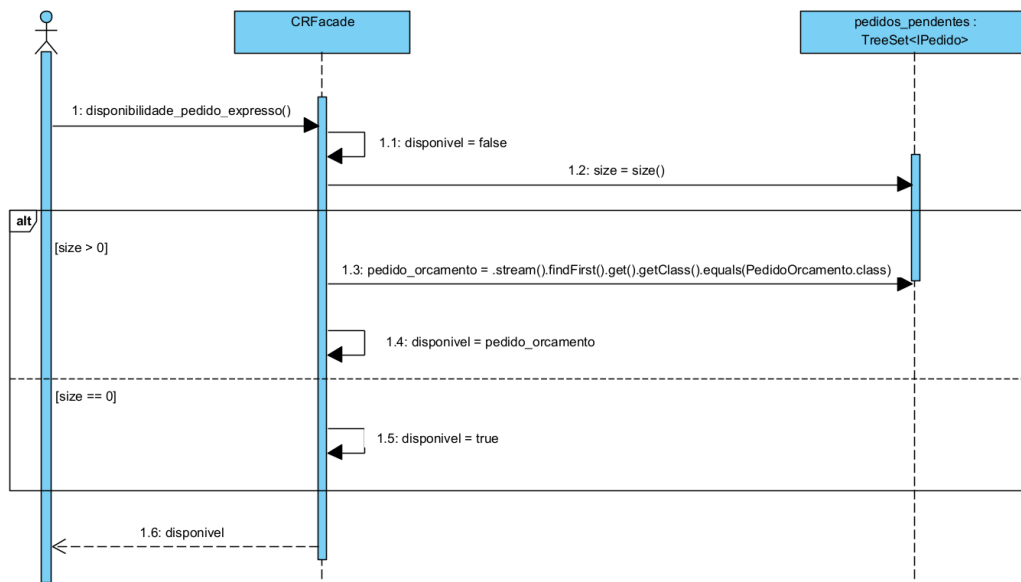
No caso de ultrapassar o orçamento, o técnico apenas tem a opção de notificar o cliente. Ao efetuar esta operação, o pedido volta à lista de orçamentos à espera de confirmação do cliente.

Caso não tenha sido ultrapassado, o técnico poderá guardar o estado da reparação como ela está no momento, ou, caso ela já esteja concluída, pode concluir a reparação. Esta opção irá notificar o cliente que o seu equipamento já está pronto a ser entregue e, o pedido será movido para o conjunto dos pedidos prontos a entregar.

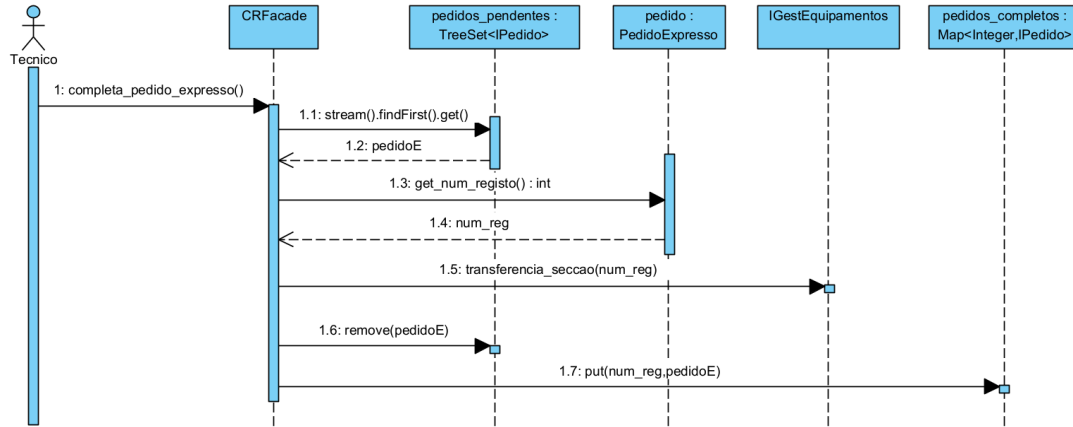
3.3.5 DS #5: Concluir pedido



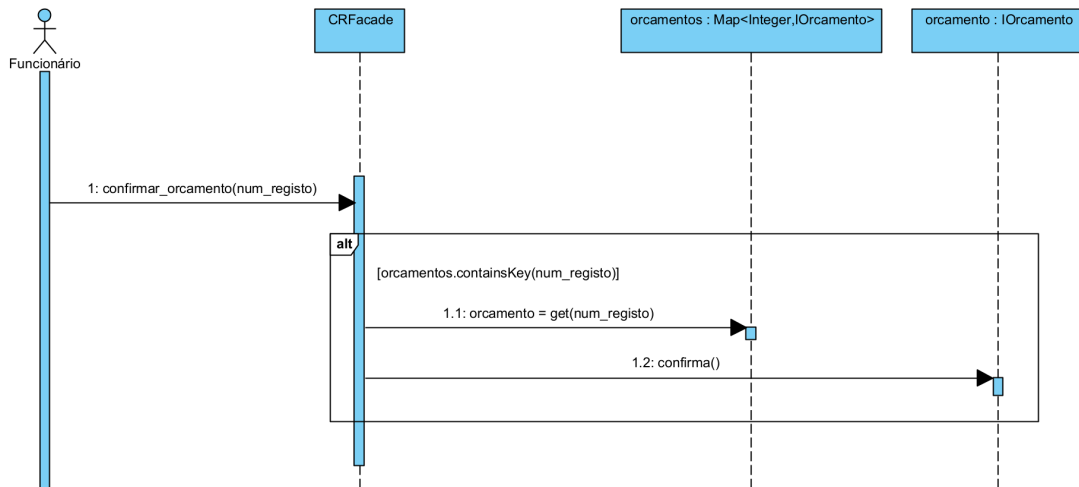
3.3.6 DS #6: Verificar disponibilidade para efetuar serviço Expresso



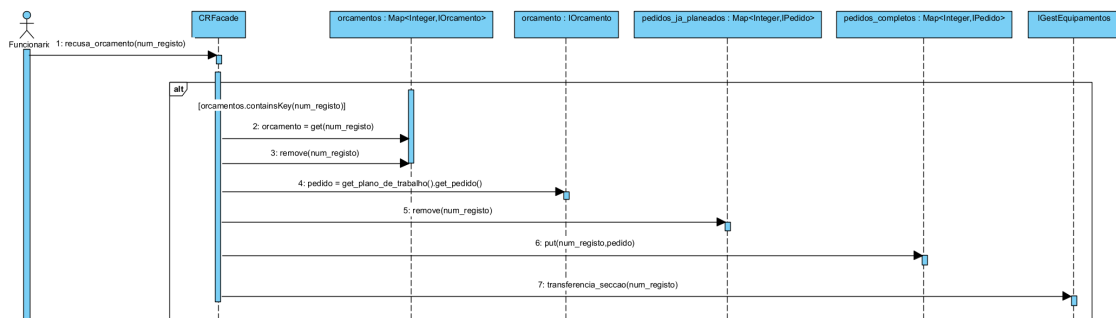
3.3.7 DS #7: Reparar pedido Expresso



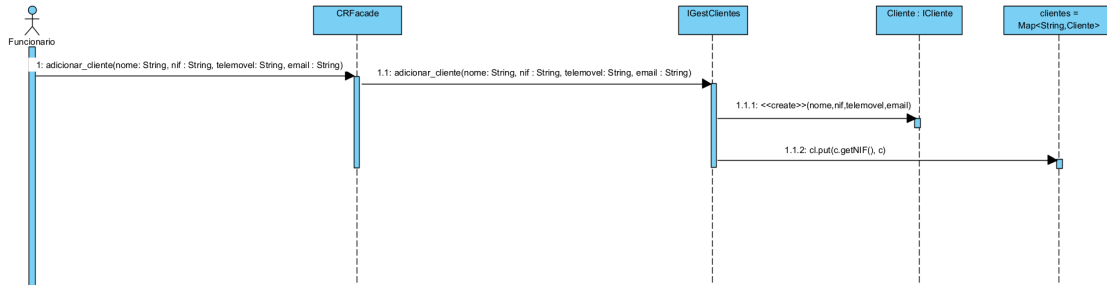
3.3.8 DS #8: Confirmar orçamento



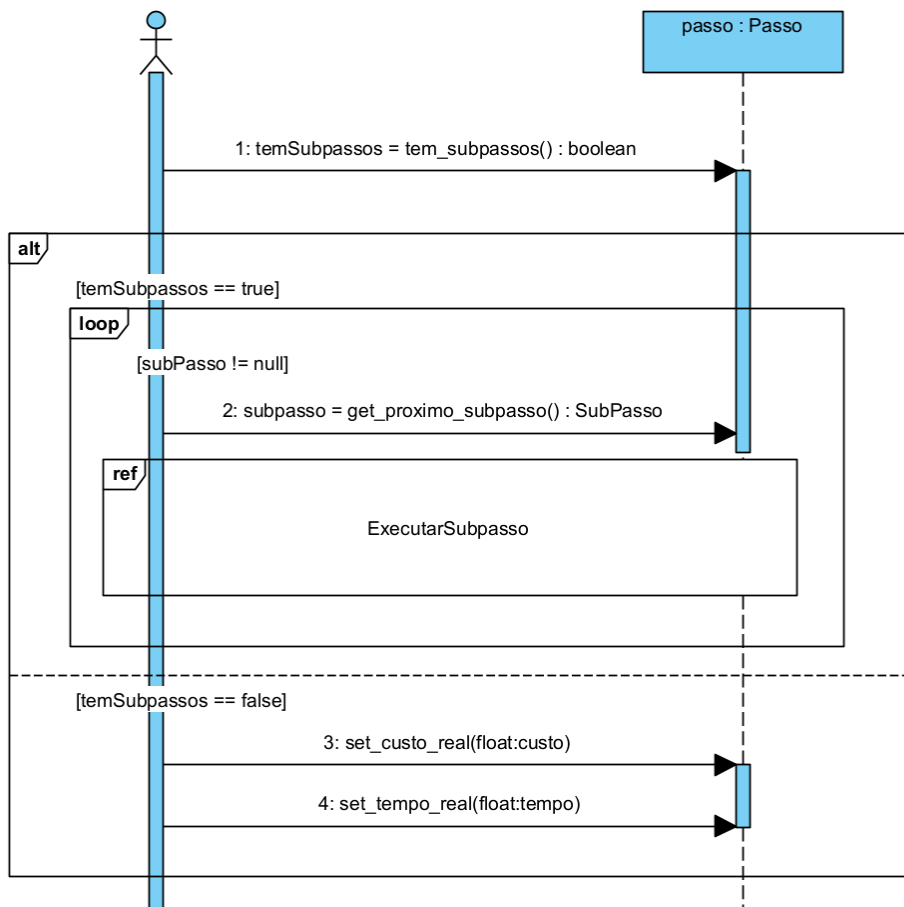
3.3.9 DS #9: Recusar orçamento



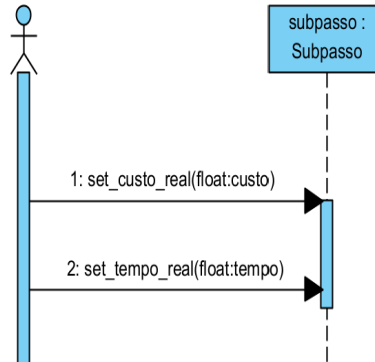
3.3.10 DS #10: Adicionar cliente



3.3.11 DS #11: Executar passo



3.3.12 DS #12: Executar subpasso



3.4 Implementação

Chegando a esta fase, procedemos, então, à implementação da nossa solução para um centro de reparações de equipamentos eletrónicos através da linguagem de programação **Java**.

Quando iniciamos o programa, aparece no ecrã um menu simples para o utilizador autenticar-se no sistema.

```
Centro de Reparções
1 - Autenticação
0 - Sair
Option:
1
```

Figura 14: Menu inicial

Ao seleccionarmos a opção nº1, o utilizador é convidado a inserir o seu ID e a sua password.

```
Autenticação
1 - User id
2 - Password
3 - ---
0 - Sair
Option:
|
```

Figura 15: Menu de autenticação

No caso dos dados corresponderem aos verdadeiros, é permitido o acesso do utilizador ao programa que é definido pelo seu tipo (Técnico, Funcionário ou Gestor). Caso falhe, aparecerá novamente o menu da autenticação.

```
Autenticação
1 - User id: [surumkata]
2 - Password: *****
3 - Entrar
0 - Sair
Option:
```

Figura 16: Menu de autenticação válida

Para além das funcionalidades que os diferentes tipos de utilizador têm (que serão listadas abaixo), todos têm a opção de sair do programa, ou de simplesmente dar logout.

O **técnico** tem ao seu dispor as seguintes funcionalidades: criar um plano (e consequentemente gerar orçamento), processar a realização de um pedido Expresso e processar a reparação do equipamento. De notar que este só pode realizar o processo de um serviço Expresso quando este existe no sistema. Caso haja, o técnico terá de o efetuar em detrimento de qualquer outra reparação, uma vez que este serviço tem prioridade.

<pre>Menu Inicial 1 - Gerar orçamento/Criar plano 2 - --- 3 - Processar reparação 4 - Logout 0 - Sair Option: </pre>	<pre>Menu Inicial 1 - Gerar orçamento/Criar plano 2 - Realizar pedido expresso 3 - --- 4 - Logout 0 - Sair Option: </pre>
---	--

Figura 17: Menus de técnico autenticado

É permitido ao **funcionário** registar clientes, pedidos (e consequentemente o registo do equipamento), confirmação ou a recusa do orçamento por parte de um cliente e a conclusão do pedido, ou seja, a entrega do equipamento ao cliente.


```
Menu Inicial
1 - Registrar cliente
2 - Registrar pedido
3 - Confirmar/Recusar orcamento
4 - Concluir pedido/Entregar equipamento
5 - Logout
0 - Sair
Option:
|
```

Figura 18: Menu de funcionário autenticado

O **gestor**, para além de possuir as mesmas opções que o funcionário e técnico, pode, também, registar novos utilizadores, listar funcionários e técnicos e analisar as estatísticas dos mesmos.

```
Menu Inicial
1 - Registrar cliente
2 - Registrar pedido
3 - Gerar orçamento/Criar plano
4 - Confirmar/Recusar orcamento
5 - ---
6 - Processar reparação
7 - Concluir pedido/Entregar equipamento
8 - Lista de funcionários
9 - Lista de técnicos
10 - Registrar utilizador
11 - Estatísticas de utilizadores
12 - Logout
0 - Sair
Option:
|
```

Figura 19: Menu de gestor autenticado

Ao registar um pedido, o funcionário tem a opção de registar um pedido de serviço Expresso ou um pedido de orçamento, sendo que a primeira opção só está disponível se houver disponibilidade para ser realizado o serviço em questão (atualmente, no sistema, a disponibilidade está limitada a 1 registo de serviço Expresso, ou seja, para se poder registar outro serviço Expresso, é preciso primeiro realizar o já registado).

```
Registo Pedido|
1 - Serviço Express
2 - Pedido de orçamento
0 - Sair
Option:
```

Figura 20: Menu de registo de um pedido

```
Processar Reparacao
1 - Apresentar informações
2 - ---
3 - Notificar cliente
4 - ---
5 - ---
0 - Sair
Option:
1
```

Figura 21: Menu da registo de um pedido quando não é possível efetuar Expresso

Ao processar uma reparação, o menu fica com uma opção bloqueada de notificar o cliente, que só fica disponível quando o orçamento excede os 120%, bloqueando todas as outras opções, exceto a funcionalidade *Apresentar informações* e a de *Sair*.

Nas figuras abaixo, pudemos observar o fluxo de um processo de reparação, onde existe um plano com apenas 1 passo com o custo estimado de 20€, sendo esse valor de custo total estimado do plano e, consequentemente, do orçamento.

```
Processar Reparacao
1 - Apresentar informações
2 - Executar passo
3 - ---
4 - Guardar
5 - ---
0 - Sair
Option:
```

Figura 22: Menu do processamento da reparação de um equipamento

```

Equipamento [#9]
Custo Estimado [20.0]
Custo Real [0.0]
Percentagem gasta [0.0]
Tempo Estimado [20.0]
Tempo Real [0.0]
Orçamento excedido [false]

```

Figura 23: Menu da listagem do plano de trabalho

Quando o técnico guarda o processamento de um passo com o custo real de 25€, tendo em conta que corresponde a 125% do orçamento ($\geq 120\%$), observamos que este fica impossibilitado de continuar o processo de reparação e tem de notificar o cliente do novo orçamento.

```

Executar Passo [1/1]
1 - Apresentar informações
2 - Custo real [0]
3 - Duração real [0]
4 - ---
5 - Guardar
0 - Sair
Option:

```

Figura 24: Menu da execução de um passo

```

1 - Apresentar informações
2 - Custo Real [25.0]
3 - Tempo Real [25.0]
4 - ---
5 - Guardar
0 - Sair
Option:
5

```

Figura 25: Menu da listagem da informação de um passo

O técnico pode guardar o processo de reparação em qualquer passo, e depois, mais tarde, retomar de onde estava, tendo em conta que o sistema lhe dá sempre o próximo passo (ainda não processado) a ser executado.

```
Processar Reparacao
1 - Apresentar informações
2 - ---
3 - Notificar cliente
4 - ---
5 - ---
0 - Sair
Option:
3
```

Figura 26: Menu da execução de um plano quando se excede o orçamento

```
Processar Reparacao
1 - Apresentar informações
2 - ---
3 - Notificar cliente
4 - ---
5 - ---
0 - Sair
Option:
3
Cliente notificado, orcamento retornado a lista de espera.
Press any key to continue
|
```

Figura 27: Menu da notificação de um cliente

4 Conclusão

A realização da segunda fase envolveu uma quantidade de planeamento considerável antes do começo da realização da implementação da solução, especialmente se tomarmos em vista as inconsistências existentes trazidas da primeira fase nos *use case* e o diagrama de classes da avaliação contínua que, embora apenas fosse suposto ser uma ideia cruda do produto final, exigiu uma grande reestruturação.

Através da concretização deste trabalho, fomos capazes de perceber o papel que o planeamento tem para a realização de um projeto, sendo este um alicerce importante para a implementação do código, facilitando a sua escrita e permitindo uma melhor organização e construção deste.

A construção do modelo de domínio na fase inicial simplificou o problema que nos foi proposto. A análise dos *use case*, por sua vez, permitiu clarificar os requisitos e funcionalidades que o programa teria de dispor.

Nesta segunda fase, a construção do diagrama de classes mostrou ser exigente tendo em conta que não foi realizada com tanta antecedência como os outros diagramas. Assim, o diagrama final acabou por divergir um pouco do diagrama de classes planeado.

Em suma, acreditamos que, embora existisse a necessidade de alterar alguns pontos da primeira fase, fomos capazes de implementar um sistema que responde aos requisitos pedidos, auxiliado pelos métodos de planeamento e desenvolvimento adquiridos nesta unidade curricular.