



**FCT**  
FACULDADE DE CIÊNCIAS  
E TECNOLOGIA  
UNIVERSIDADE DOS AÇORES

e

## Programação Centrada em Objetos

### Licenciatura em Informática

#### Projeto

O projeto pode ser realizado em grupos de três pessoas e tem como objetivo proporcionar aos alunos a oportunidade para aplicarem os conteúdos lecionados nas aulas em situações próximas da realidade, em que a correção, a eficiência e a qualidade são essenciais para a construção de aplicações. Propomos a elaboração dos artefactos de análise, de desenho e do modelo de implementação do sistema que descrevemos a seguir. Pedimos para elaborarem os artefactos de análise e desenho, a saber: diagrama de casos de uso, diagramas de sequência do sistema, modelo de domínio, contratos, de operações, de interação (comunicação), de classes e de pacotes; e do modelo de implementação, constituído pelo código fonte (em Java) da implementação das operações que vos pedimos.

Para a entrega, devem entregar um relatório em formato *pdf*. Podem usar este documento como esquema de relatório e preencher os “espaços deixados vazios”. Os diagramas UML que tiverem de produzir podem ser feitos “à mão”, não sendo necessário a utilização de qualquer ferramenta gráfica.

Se optarem por criar os diagramas recorrendo a uma ferramenta gráfica, podem recorrer ao *Visual Paradigm* (uma ferramenta de uso profissional), desde que solicitem a licença de estudante.

#### Índice

Índice.....	
2 Visão do projeto e caso de negócio.....	3
3 Introdução.....	3
Utilizadores.....	4
Objectivos-chave para os utilizadores.....	4
Outros requisitos e restrições.....	4
Modelo de casos de uso.....	5
5 Casos de uso.....	5
5 Diagrama de casos de uso.....	5
5 Diagramas de sequência do sistema.....	5
5 Contratos das operações do sistema.....	5
5 Modelo de domínio.....	6
6 Modelo de desenho.....	7
Diagramas de interação.....	7

Diagrama de classes.....	7
Implementação da camada de domínio	
.....8 Implementação da camada de	
apresentação .....	9
Testes e	
execução.....	10
Calendarização.....	1
	1
Observações.....	1
	2

## Visão do projeto e caso de negócio

### Introdução

No atual contexto de elevada procura logística e de eficiência operacional, os centros de distribuição assumem um papel essencial na cadeia de abastecimento, assegurando que as mercadorias chegam às lojas e clientes de forma rápida e controlada.

O processo logístico envolve receção de mercadorias de fornecedores, gestão de stocks em armazém, movimentações internas, e a preparação e expedição de encomendas para lojas. Uma gestão inadequada destas operações pode causar ruturas de stock, atrasos e aumento de custos.

A aplicação a desenvolver tem como objetivo principal fornecer uma solução abrangente e acessível para os diferentes intervenientes na operação de um centro de distribuição. Contudo, vai além do simples registo de entradas e saídas de produtos, permitindo também a gestão de localizações de armazenamento, movimentações internas e a preparação de encomendas para expedição, enriquecendo a rastreabilidade e a qualidade do serviço.

Para atingir este propósito, a aplicação irá disponibilizar informação detalhada sobre fornecedores, produtos, lotes e localizações, permitindo aos operadores e gestores consultar o estado do inventário em tempo real. Para garantir a qualidade e atualização constante destas informações, a indústria fornecedora terá a capacidade de associar novos produtos e lotes às expedições enviadas, enriquecendo a base de dados do armazém.

Na informação sobre os produtos constam o código (SKU), nome, lote, unidades de medida e eventuais restrições, **como necessidade de armazenamento em frio, prazo de validade, temperatura ou se é um produto perigoso**. Além disto, cada produto está associado a um fornecedor, que fornece o nome e um contacto logístico, útil em situações de inconformidade de receção. Um produto pode estar armazenado em várias localizações do centro, garantindo sempre a rastreabilidade. **As unidades de medida definem as variações consoante o tipo do produto podendo estes serem produtos de peso (e.g. kg, g, tonelada), volume (e.g. litros, mililitros e barris), unidade (caixa, palete, peças) ou dimensional (e.g. tecidos, cabos) com o comprimento por unidade**

Os operadores de receção e armazém terão um papel fundamental na execução das operações, registando receções de mercadorias, conferindo quantidades, indicando não conformidades e realizando movimentações de expedições. Os operadores de seleção serão responsáveis pela preparação das expedições, identificando produtos e quantidades e consolidando cargas para expedição. Os gestores logísticos, por sua vez, registam encomendas, acompanham o estado global do inventário, aprovam ajustes de stock e consultam relatórios de desempenho, garantindo o cumprimento das políticas de armazenamento (FIFO/FEFO). Esta colaboração assegura a fiabilidade e a eficiência do processo logístico.

Além disto, a aplicação contará com um administrador que fará a gestão das contas de utilizadores, fornecedores, lojas e configuração das localizações do armazém, assegurando que cada perfil tem acesso controlado à plataforma e respetivas funcionalidades.

Em resumo, a aplicação a desenvolver visa melhorar significativamente a eficiência e segurança dos processos logísticos ao disponibilizar informações abrangentes e confiáveis sobre a gestão de mercadorias num centro de distribuição. A colaboração entre operadores, gestores, transportadores e fornecedores é o alicerce desta iniciativa, promovendo a qualidade do serviço e a satisfação dos clientes finais.

## Utilizadores

Os utilizadores da aplicação incluem:

- Operadores de Receção, que registam mercadorias recebidas, verificam quantidades, identificam lotes e registam não conformidades.
- Operadores de Armazém, que realizam movimentações de produtos e expedições entre localizações, controlam stocks e asseguram a rastreabilidade.
- Operadores de Seleção, que preparam as encomendas das lojas, consolidam produtos e organizam as cargas para expedição.
- Gestores Logísticos, que acompanham o inventário global, aprovam ajustes de stock e consultam relatórios de desempenho do armazém.
- Administrador, que faz a gestão das contas de utilizadores, fornecedores, lojas e localizações do armazém.

## Objectivos-chave para os utilizadores

- Os operadores de receção pretendem uma aplicação que permita registar e validar receções de forma rápida, garantindo a identificação de lotes e o registo de não conformidades.
- Os operadores de armazém querem funcionalidades que facilitem as movimentações de produtos/expedições e permitam consultar o stock disponível por produto e localização.
- Os operadores de seleção necessitam de ferramentas que otimizem a preparação de encomendas, reduzindo erros de seleção e facilitando a consolidação de cargas.
- Os gestores logísticos procuram relatórios claros e detalhados sobre stocks, expedições, receções e desempenho global do armazém, além de mecanismos para aprovar ajustes.
- O administrador precisa de gerir utilizadores, fornecedores, lojas e a configuração das localizações do armazém, assegurando que os perfis têm permissões adequadas.

## Outros requisitos e restrições

O desenvolvimento do sistema deve aplicar as técnicas de análise, desenho e programação orientadas a objetos ensinadas na unidade curricular.

A aplicação, durante o arranque, deverá conseguir carregar um ficheiro no formato JSON, que irá conter uma lista de produtos, fornecedores, lojas, localizações e stocks iniciais. Assim, após o arranque, já haverá informação disponível a ser trabalhada.

É fornecido um ficheiro exemplo no formato JSON com este enunciado.

Pode haver outros requisitos em casos de uso ou que foram omitidos do enunciado. Os docentes estão disponíveis para esclarecer quaisquer dúvidas.

## Modelo de casos de uso

Este capítulo descreve o modelo de casos de uso do sistema. Na tabela seguinte apresentam-se os casos identificados nesta iteração. Na secção casos de uso, descrevem-se os diversos casos de uso e na secção seguinte, mostra-se o diagrama de casos de uso, com identificação dos diversos atores e da sua relação com os casos de uso. Na seção Diagramas de sequência do sistema, apresenta-se os diagramas de sequência do sistema e por fim, na secção seguinte, os contratos das operações do sistema.

## Casos de uso

<b>Id</b>	<b>Nome</b>	<b>Estado</b>
UC01	Autenticar utilizador	Casual
UC02	Registar utilizador	Casual
UC03	Criar fornecedor	Elaborado
UC04	Consultar fornecedores	Casual
UC05	Criar produto	Elaborado
UC06	Consultar produtos	Casual
UC07	Registar receção de mercadoria	Elaborado
UC08	Consultar receções	Casual
UC09	Registar encomenda de loja	Elaborado
UC10	Preparar expedição	Elaborado
UC11	Consultar inventário por localização	Casual
UC12	Mover expedição	Elaborado
UC13	Aprovar ajuste de stock	Elaborado
UC14	Consultar relatórios	Casual
UC15	Mover produto	Casual
UC16	Criar ajuste de stock	Casual
UC17	Editar produto	Casual
UC18	Remover produto	Casual
UC19	Editar fornecedor	Casual
UC20	Remover fornecedor	Casual
UC21	Expedir encomenda	Casual
UC22	Gerir loja	Breve
UC23	Gerir localização	Breve

#### *UC01: Autenticar utilizador*

O utilizador indica ao sistema que pretende iniciar sessão, fornecendo nome de utilizador e password. O sistema valida as credenciais e, em caso de sucesso, apresenta as opções de acordo com o perfil (Operador de Receção, Operador de Armazém, Operador de Seleção, Gestor ou Administrador). Em caso de falha, informa o utilizador e termina o caso de uso.

#### *UC02: Registar utilizador*

O administrador indica que pretende registar um novo utilizador. O sistema solicita o primeiro e o último nome, email, nome de utilizador, password e papel (Operador de Receção, Operador de Armazém, Operador de Seleção, Gestor ou Administrador). O sistema valida unicidade do nome de

utilizador e cria o utilizador. Em caso de dados inválidos, informa o administrador.

#### *UC03: Criar fornecedor*

**Ator principal:** Administrador

##### **Pré-condições:**

- O administrador está autenticado.

##### **Pós-condições:**

- Um fornecedor novo é criado no sistema com nome e contacto logístico.
- O fornecedor fica disponível para associação a receções e produtos.

##### **Cenário principal de sucesso:**

1. O utilizador indica que pretende criar um fornecedor.
2. O sistema pede o nome e o contacto logístico (email e telefone).
3. O utilizador fornece os dados solicitados.
4. O sistema valida unicidade do nome do fornecedor.
5. O sistema apresenta um resumo e solicita confirmação.
6. O utilizador confirma a criação.
7. O sistema cria o novo fornecedor.

##### **Extensões:**

- 3a. Se faltar informação obrigatória, o sistema informa e volta ao passo 2.
- 4a. Se o nome do fornecedor já existir, o sistema informa e volta ao passo 2.
- 5a. O utilizador cancela a criação — o caso de uso termina.

#### *UC04: Consultar fornecedores*

O utilizador autenticado solicita a listagem de fornecedores. O sistema apresenta as primeiras 10 entradas por ordem alfabética e possibilita carregar mais resultados até esgotar a lista. Durante a listagem, o utilizador pode cancelar o carregamento de mais resultados e o caso de uso termina.

#### *UC05: Criar produto*

**Ator principal:** Administrador

##### **Pré-condições:**

- O administrador está autenticado.

##### **Pós-condições:**

- Um novo produto é criado com o código, nome, unidade de medida, restrições e data de criação.
- O produto fica disponível para receções e encomendas.

##### **Cenário principal de sucesso:**

1. O utilizador indica que pretende criar um produto.
2. O sistema pede SKU (código), nome, unidade de medida (e.g. 125ml), restrições (e.g. frio, se exige validade (sim/não)).
3. O utilizador fornece os dados.
4. O sistema valida unicidade do SKU.
5. O sistema apresenta um resumo e pede confirmação.
6. O utilizador confirma a criação.
7. O sistema cria o produto.

**Extensões:**

4a. Se o SKU já existir, o sistema informa e volta ao passo 2.

3a. Se faltar informação obrigatória, o sistema informa e volta ao passo 2.

5a. O utilizador cancela a criação — o caso de uso termina.

**UC06: Consultar produtos**

O utilizador autenticado solicita a listagem de produtos. O sistema mostra SKU, nome, unidade de medida e restrições para os primeiros 10 produtos por ordem alfabética, permitindo pedir mais múltiplos de 10. Durante a listagem, o utilizador pode cancelar o carregamento de mais resultados e o caso de uso termina.

**UC07: Registar receção de mercadoria**

**Ator principal:** Operador de Receção

**Pré-condições:**

- O operador está autenticado.
- Existem fornecedores registados e pelo menos um produto.

**Pós-condições:**

- A receção é registada com fornecedor, linhas recebidas e eventuais não conformidades.
- O stock é atualizado em estado *disponível* ou *quarentena* conforme regras.

**Cenário principal de sucesso:**

1. O utilizador indica que pretende registar uma receção.
2. O sistema pede o fornecedor.
3. O utilizador seleciona o fornecedor.
4. O sistema solicita produtos, por linha, quantidade recebida, lote e validade (quando aplicável).
5. O utilizador introduz os dados enquanto houver linhas para ser inseridas.
6. O sistema valida existência dos produtos e regras de validade para produtos que a exigem.
7. O sistema pergunta se existem não conformidades (falta, excesso, dano, lote/validade inválidos).
8. O utilizador regista as não conformidades se existirem.
9. O sistema regista a receção e atualiza o stock.
10. O utilizador conclui o registo de mercadoria
11. O sistema apresenta resumo e identifica itens em *quarentena* quando aplicável.

**Extensões:**

6a. Se um produto não existir, o sistema informa e solicita correção da linha (volta ao passo 5).

6b. Se a validade estiver ausente/expirada, o sistema marca NC e coloca em quarentena.

8a. Se a quantidade recebida exceder a capacidade imediata, o sistema assinala “a armazenar”.

9a. Se o utilizador cancelar, o caso de uso termina sem alterações.

**UC08: Consultar receções**

O utilizador autenticado solicita a listagem de receções. O sistema mostra as últimas receções com fornecedor, data, número de linhas e existência de não conformidades, permitindo filtrar por período e fornecedor.

**UC09: Registar encomenda de loja**

**Ator principal:** Gestor Logístico

**Pré-condições:**

- O utilizador está autenticado.
- Existem lojas e produtos registados.

**Pós-condições:**

- Uma encomenda é criada com linhas de produto e quantidades solicitadas.
- O stock é reservado por política (FIFO/FEFO) quando possível.

**Cenário principal de sucesso:**

1. O utilizador indica que pretende registar uma encomenda.
2. O sistema pede a loja, prioridade e as linhas (produto, quantidade).
3. O utilizador fornece os dados.
4. O sistema valida existência de loja e produtos.
5. O sistema tenta reservar stock por política definida.
6. O sistema apresenta o resumo da encomenda (linhas, quantidades reservadas, pendentes) e solicita confirmação.
7. O utilizador confirma a criação.
8. O sistema cria a encomenda e mantém as reservas efetuadas.

**Extensões:**

5a. Se não houver stock suficiente, o sistema reserva parcialmente e marca remanescente como pendente.

5b. Se o produto estiver bloqueado, o sistema não permite reserva.

6a. O utilizador cancela a criação — o caso de uso termina.

*UC10: Preparar expedição*

**Ator principal:** Operador de Seleção

**Pré-condições:**

- O utilizador está autenticado.
- Existem encomendas em estado *por preparar* com reservas.

**Pós-condições:**

- As reservas são dadas como finalizadas e consolidadas em expedições.
- É gerado um manifesto de carga associado ao transportador.

**Cenário principal de sucesso:**

1. O utilizador seleciona uma ou mais encomendas.
2. O sistema gera tarefas de seleção por localização/produto.
3. O utilizador executa as tarefas, enquanto houver.
4. O sistema valida quantidades e atualiza o stock.
5. O utilizador confirma a encomenda.
6. O sistema confirma a expedição, com o estado “por expedir” e gera o manifesto.

**Extensões:**

3a. Se a quantidade física não coincidir, o sistema permite correção.

4a. Se a quantidade ficar negativa, a operação é recusada. O caso de uso termina.

#### *UC11: Consultar inventário por localização*

O utilizador autenticado indica a localização a consultar. O sistema apresenta os itens de stock nessa localização (produto, lote, quantidade, estado) e permite avançar paginação de 10 em 10 ou exportar listagem para CSV. Durante a listagem, o utilizador pode cancelar o carregamento de mais resultados e o caso de uso termina.

#### *UC12: Mover expedição*

**Ator principal:** Operador de Armazém

##### **Pré-condições:**

- O utilizador está autenticado.
- Existem expedições criadas no sistema.
- A expedição encontra-se em estado *por expedir*.

##### **Pós-condições:**

- A expedição é movida de um estado ou zona logística para outro (por exemplo: da área de preparação → área de expedição → doca de carga).
- O histórico de movimentação da expedição é registado (quem, quando, de onde, para onde).

##### **Cenário principal de sucesso:**

1. O utilizador indica que pretende mover uma expedição.
2. O sistema apresenta a lista de expedições em estado “por expedir” e respetiva localização atual.
3. O utilizador seleciona a expedição a mover.
4. O sistema pede a nova localização/zona logística (por exemplo: área de consolidação, área de expedição, doca X).
5. O utilizador fornece a nova localização.
6. O sistema valida se a expedição está elegível para movimentação (não foi expedida).
7. O sistema solicita confirmação.
8. O utilizador confirma.
9. O sistema regista a movimentação, atualiza a localização da expedição e guarda no histórico.

##### **Extensões:**

3a. Se a expedição selecionada já estiver carregada, o sistema informa e impede a movimentação — o caso de uso termina.

5a. Se a nova localização não for válida (e.g. inexistente), o sistema recusa a operação — o caso de uso termina.

7a. O utilizador cancela a movimentação — o caso de uso termina.

#### *UC13: Aprovar ajuste de stock*

**Ator principal:** Gestor Logístico

##### **Pré-condições:**

- O utilizador está autenticado.

##### **Pós-condições:**

- O ajuste é aprovado/rejeitado com motivo registado.
- O inventário só é atualizado se aprovado.



**Cenário principal de sucesso:**

1. O utilizador consulta ajustes propostos.
2. O sistema apresenta detalhe de cada ajuste.
3. O utilizador seleciona ajuste e escolhe aprovar/rejeitar.
4. O sistema solicita comentário da decisão.
5. O utilizador confirma.
6. O sistema regista a decisão; se aprovado, atualiza o inventário.

**Extensões:**

3a. Se ajuste levar stock a negativo, o sistema alerta e solicita confirmação explícita.

5a. O utilizador cancela — o caso de uso termina.

**UC14: Consultar relatórios**

O utilizador autenticado solicita relatórios do armazém. O sistema disponibiliza, por exemplo: inventário por produto/localização, receções por período e fornecedor, expedições por período e loja, e rastreabilidade por lote. O utilizador pode aplicar filtros e exportar resultados.

**UC15: Mover produto**

O operador de armazém indica que pretende mover um produto de uma localização para outra dentro do armazém. O sistema solicita o produto, a localização de origem e de destino, e a quantidade a mover. Após confirmação, o sistema atualiza o inventário e regista a movimentação com a data de movimentação.

**UC16: Criar ajuste de stock**

O operador de armazém indica que pretende registar um ajuste manual de stock, normalmente devido a divergências entre contagem física e sistema ou devido a algum eventual dano que possa ter acontecido. O sistema solicita o produto, a localização e a quantidade ajustada. Após confirmação, o sistema cria o ajuste com estado “pendente” de aprovação pelo gestor logístico.

**UC17: Editar produto**

O administrador indica que pretende editar um produto existente. O sistema apresenta a informação atual (nome, unidade, restrições, fornecedor associado). O utilizador altera os campos pretendidos e confirma a edição. O sistema valida os dados e atualiza o produto.

**UC18: Remover produto**

O administrador seleciona um produto e indica que pretende removê-lo. O sistema verifica se o produto não está associado a receções, encomendas ou stock ativo. Se não existirem dependências, o sistema remove o produto; caso contrário, informa que a remoção não é possível.

**UC19: Editar fornecedor**

O administrador seleciona um fornecedor e indica que pretende editar os seus dados. O sistema apresenta o nome e contacto logístico atuais. O utilizador altera as informações e confirma. O sistema valida e atualiza o registo do fornecedor.

**UC20: Remover fornecedor**

O administrador seleciona um fornecedor e solicita a sua remoção. O sistema verifica se o fornecedor tem produtos ou receções associadas. Se não houver dependências, o sistema elimina o fornecedor; caso contrário, informa o utilizador de que a operação é impossível.

**UC21: Expedir encomenda**

O operador de armazém indica que pretende expedir uma encomenda previamente preparada. O sistema verifica se todos os produtos foram conferidos. O operador de armazém seleciona um transportador. Após confirmação, o sistema altera o estado da encomenda para “expedida”, guarda

a data de expedição e atualiza o inventário e gera o comprovativo de expedição.

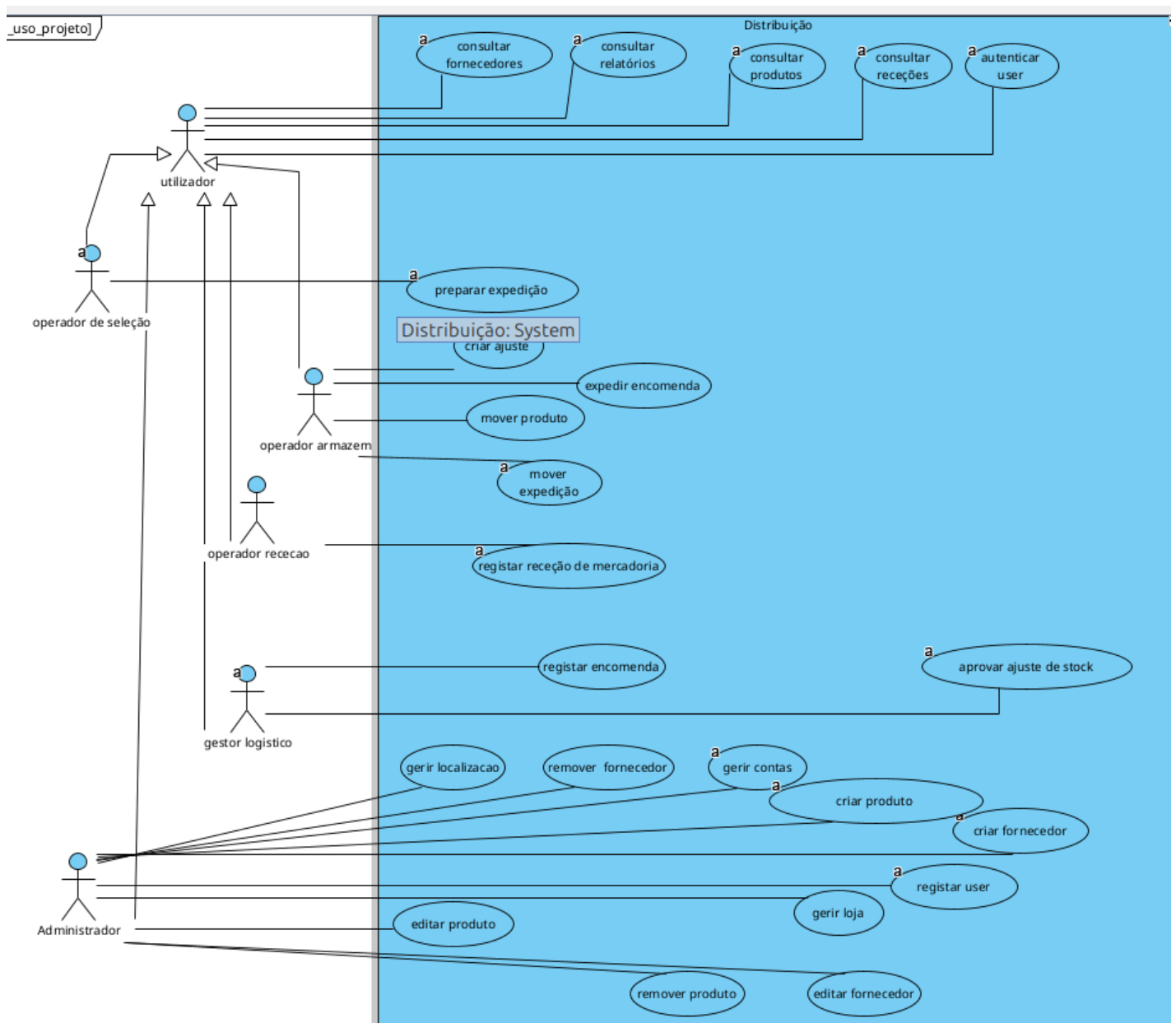
#### UC22: Gerir loja

O administrador acede à área de gestão de lojas. O sistema permite criar, editar ou remover lojas registadas, definindo código, nome e morada. O sistema valida as alterações e atualiza a lista de lojas.

#### UC23: Gerir localização

O administrador acede à gestão de localizações do armazém. O sistema permite criar, editar ou remover localizações físicas, definindo tipo (e.g. estante, solo, frigorífico, doca), capacidade máxima e se suporta algum tipo de restrição (e.g. frio, perigoso). O sistema valida os dados e atualiza o mapa do armazém.

### Diagrama de casos de uso

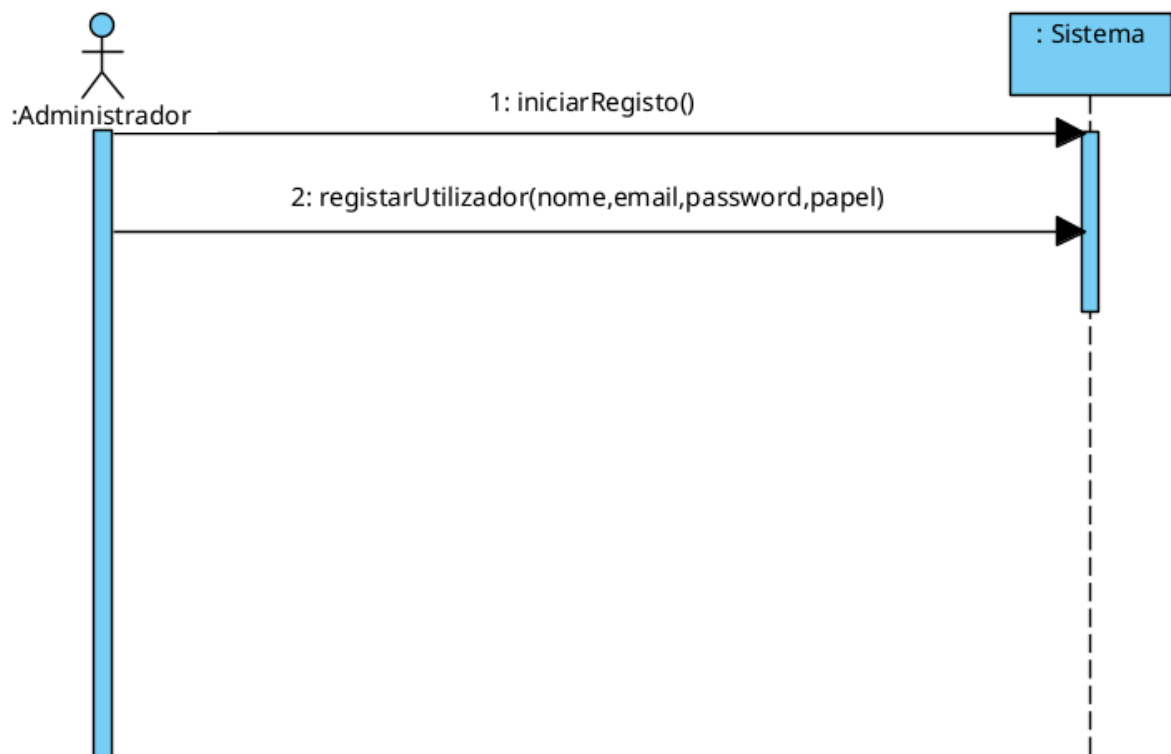


alteracoes : colocou se num sistema os casos de uso , adicionou se os novos casos como se simplificou o diagrama ao colocar uma herança do utilizador para todos os operadores do sistema unutilizando todos os includes e extends que deixavam o diagrama ilegível.

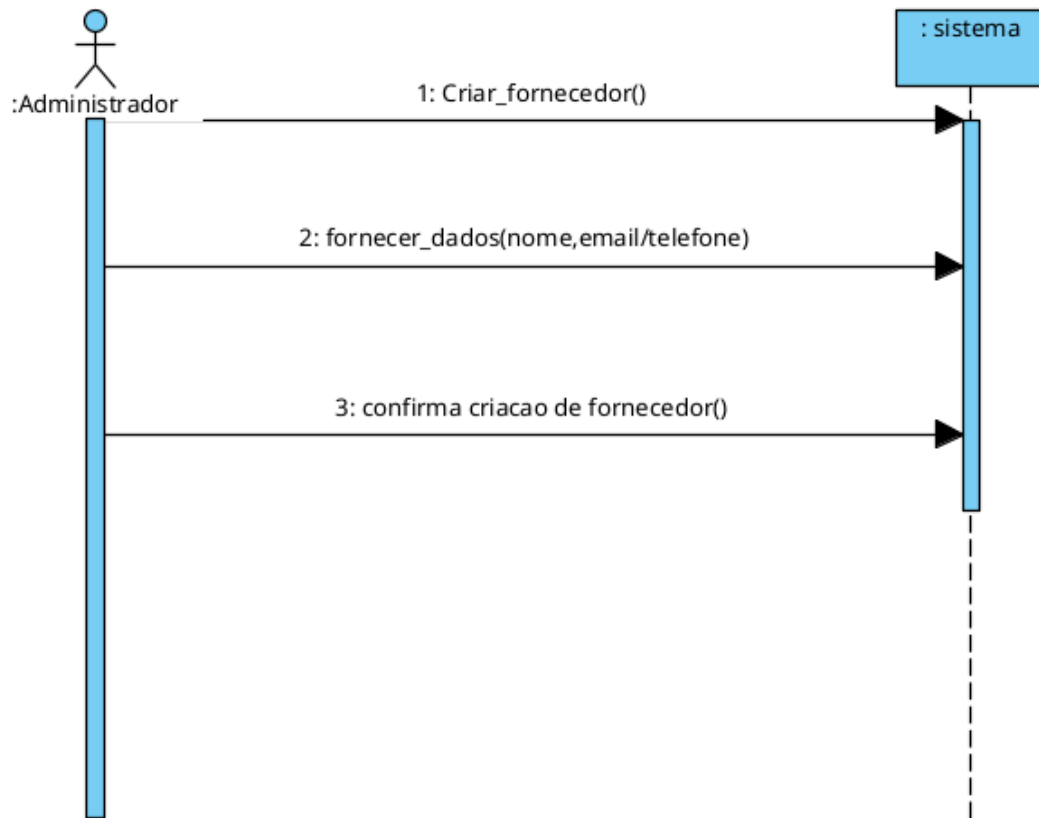
### Diagramas de sequência do sistema



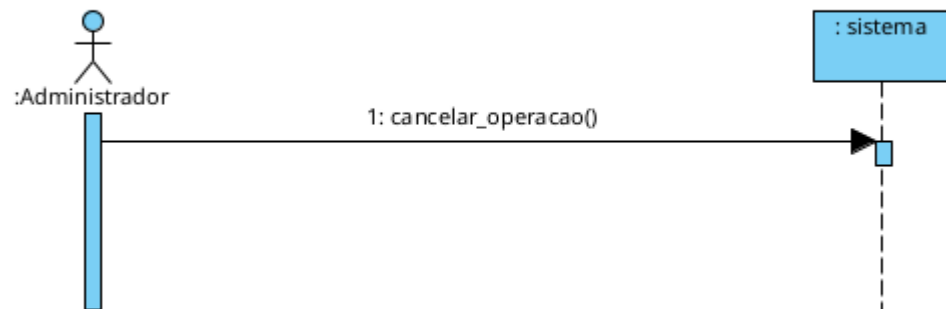
UC01:



UC02:



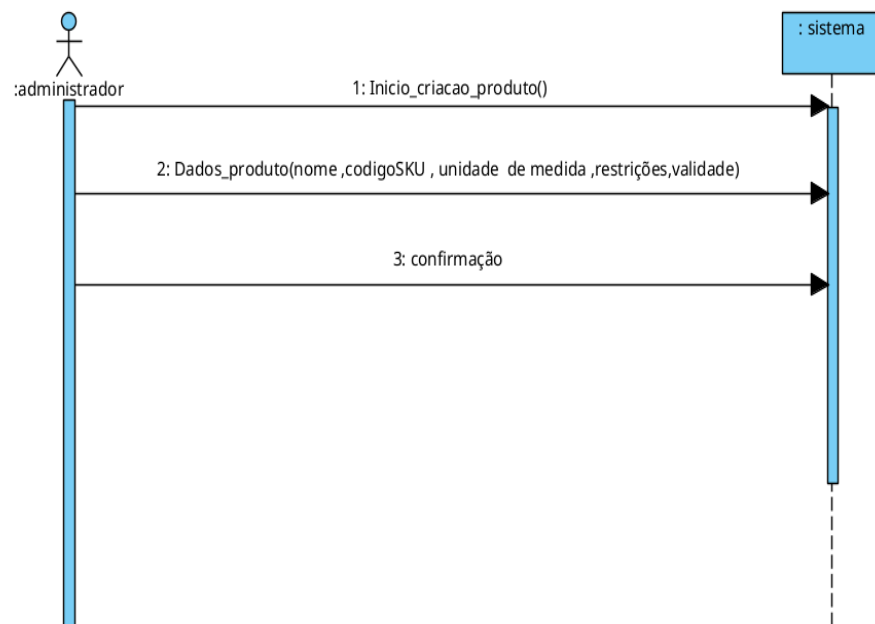
UC03:



extensão:

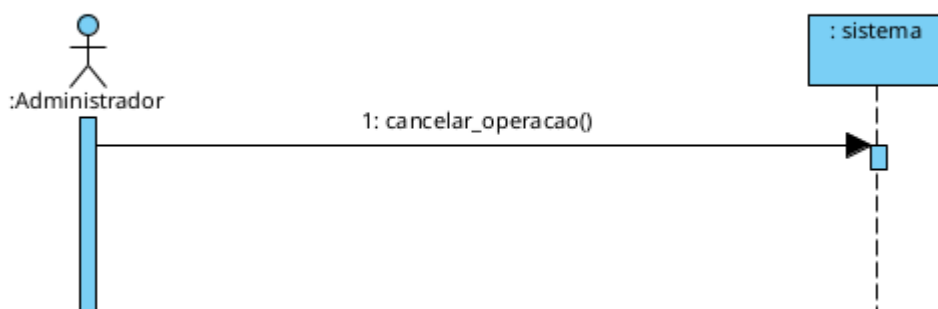


UC04:



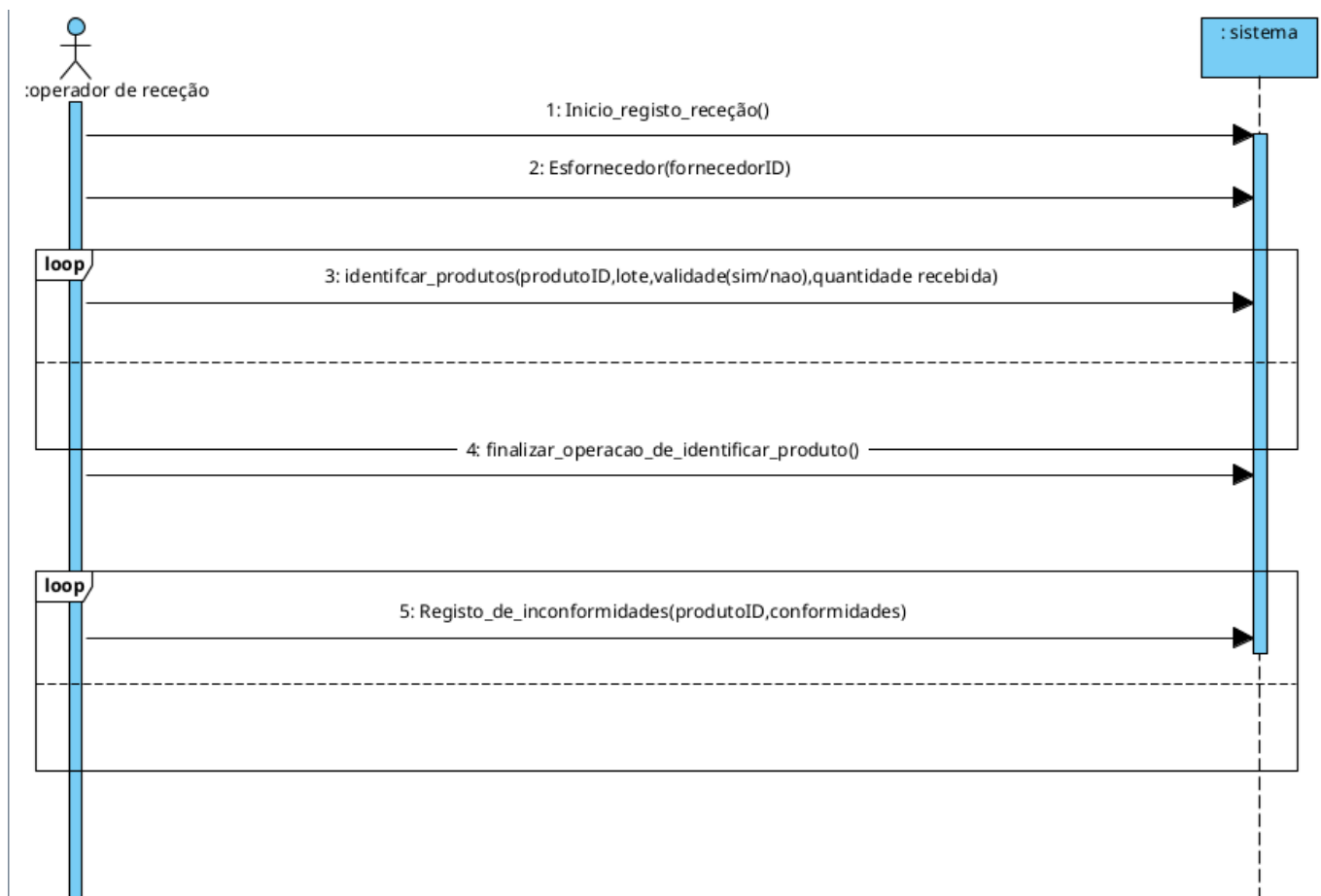
UC05:

extensão UC05:

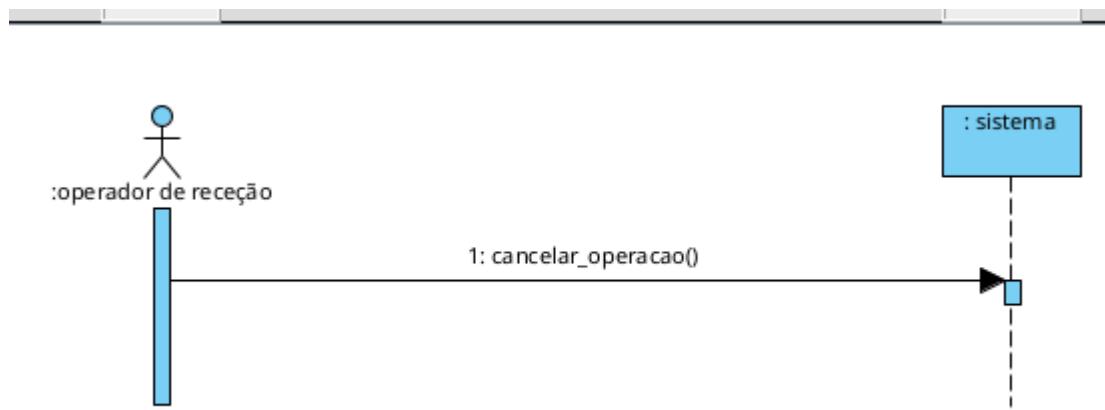




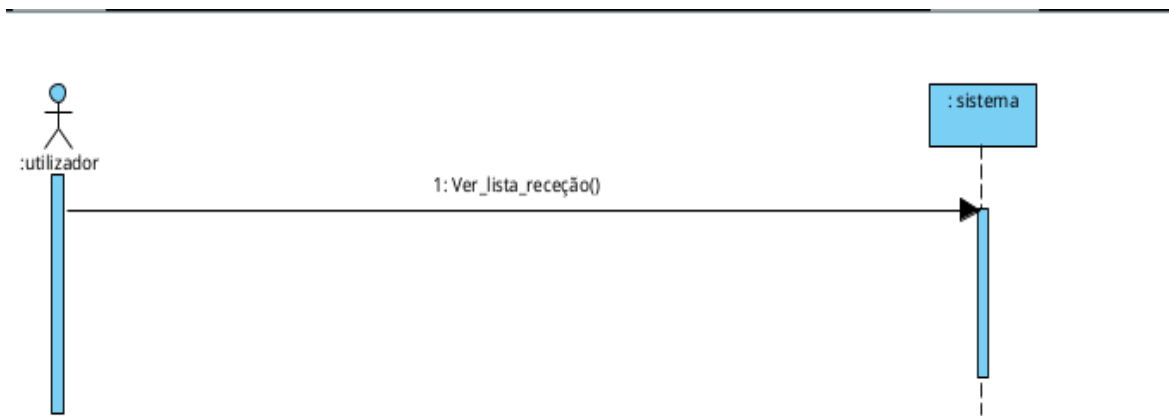
UC06:



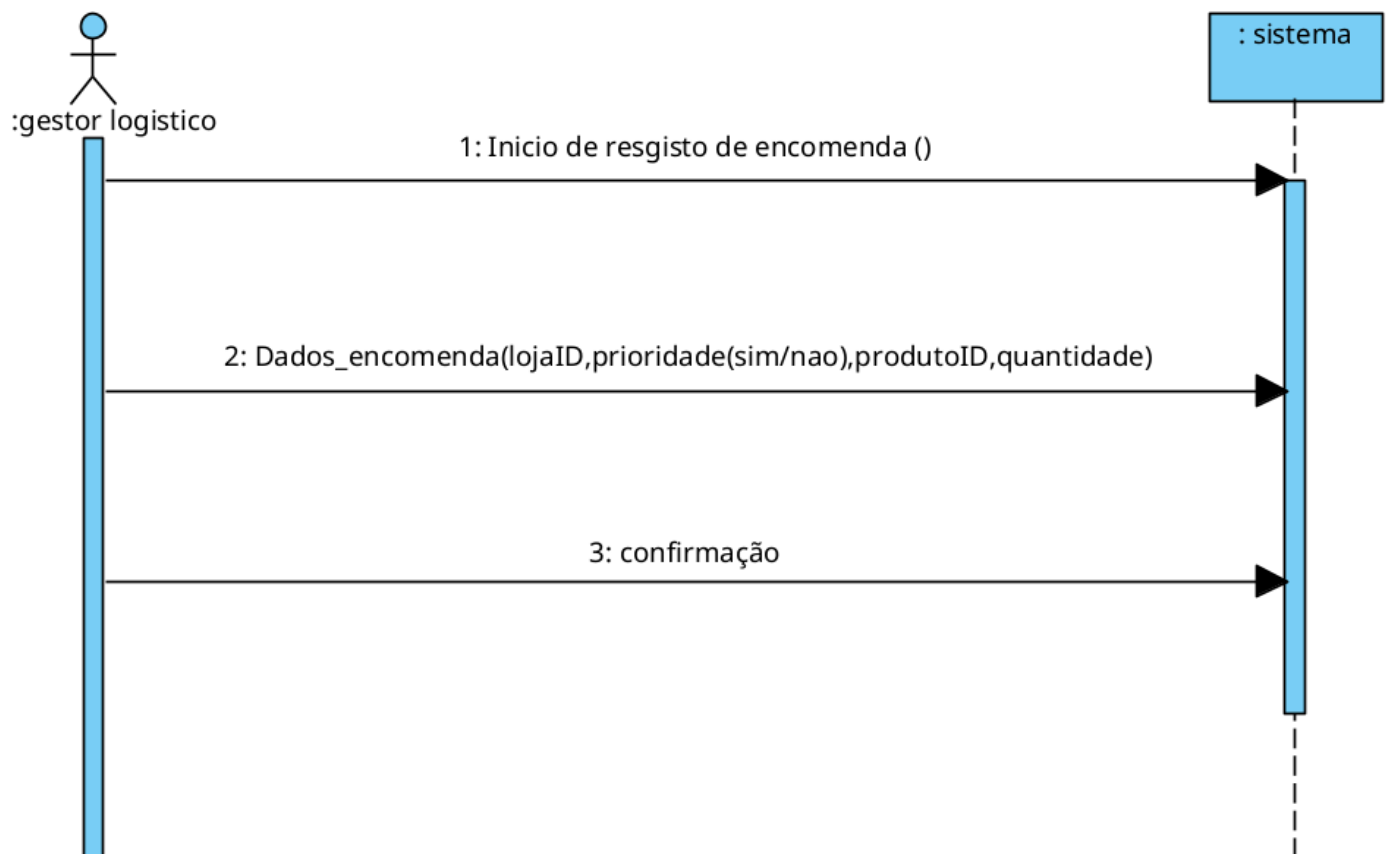
UC07:



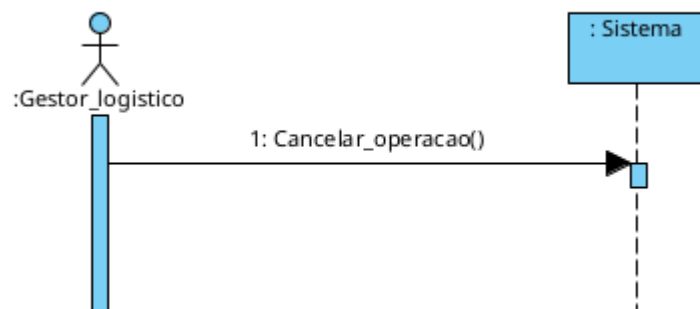
extensão U07:



UC08:

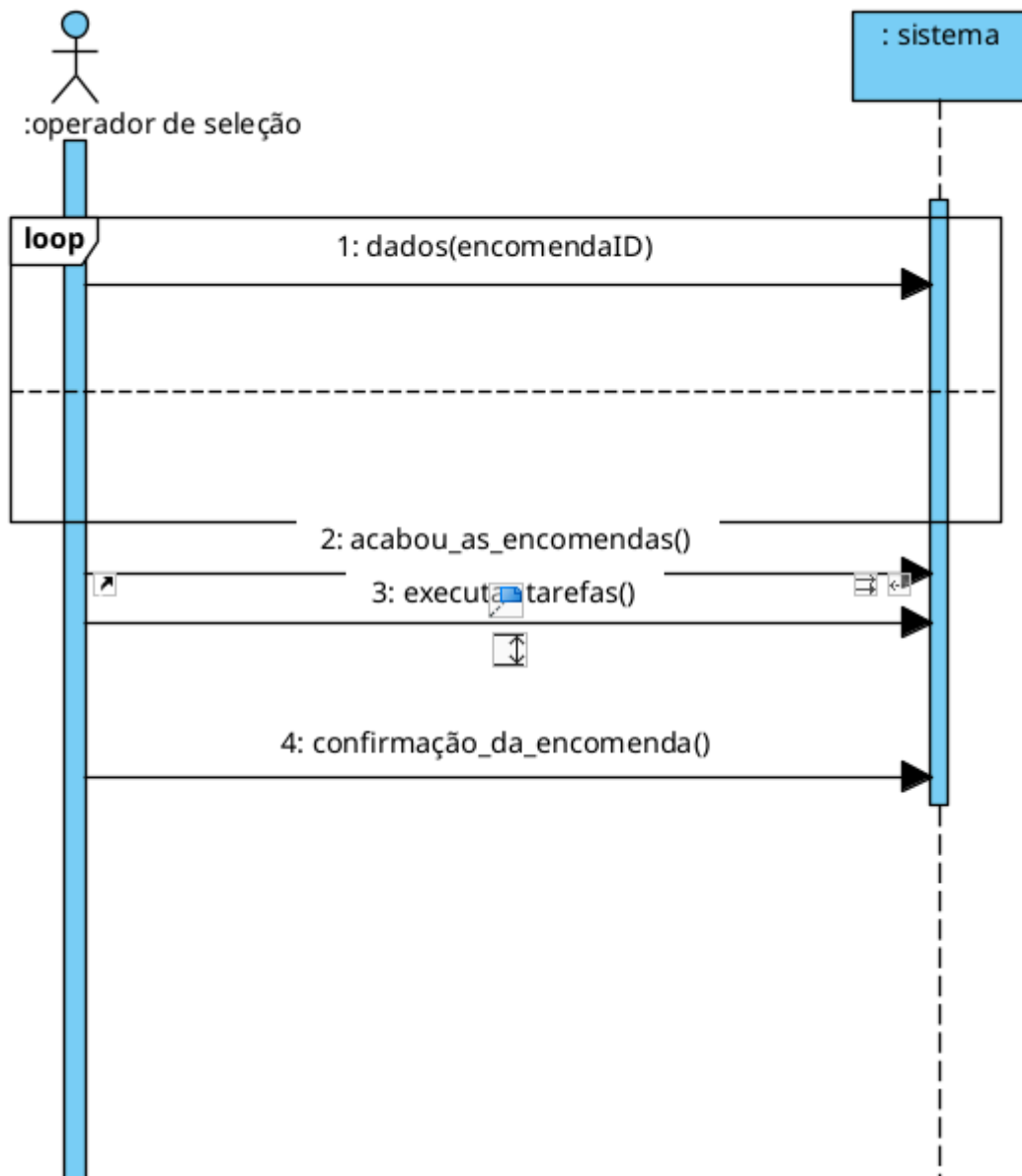


UC09:

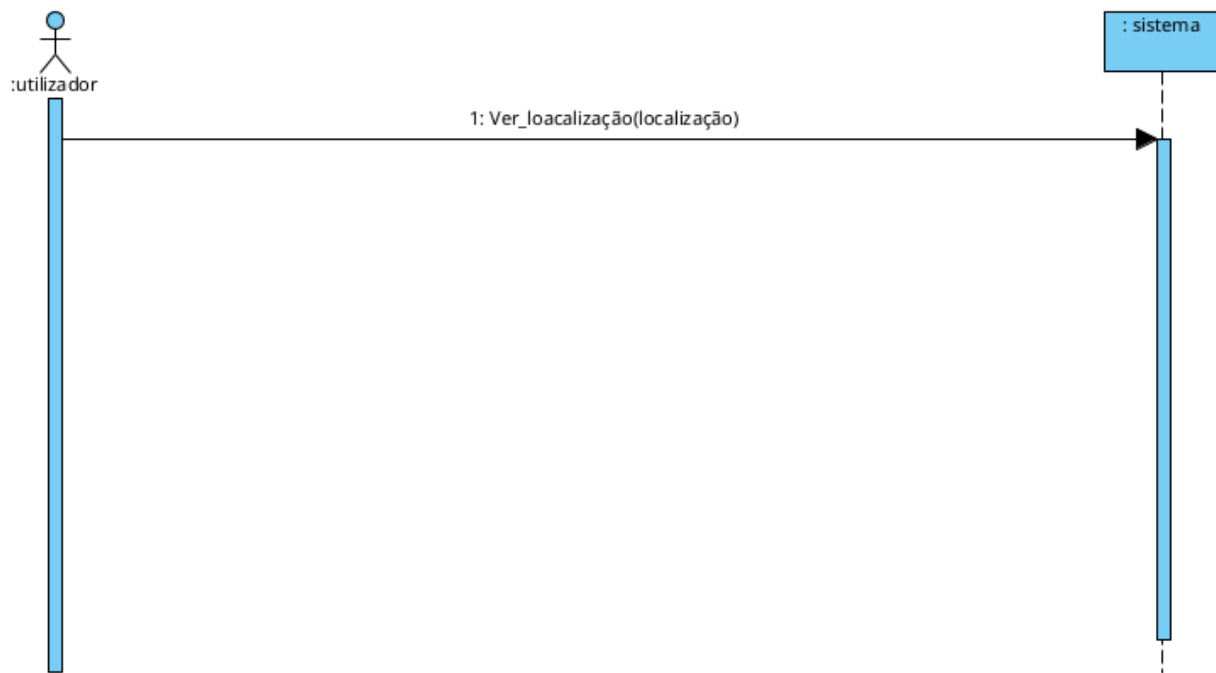


extensão UC09:

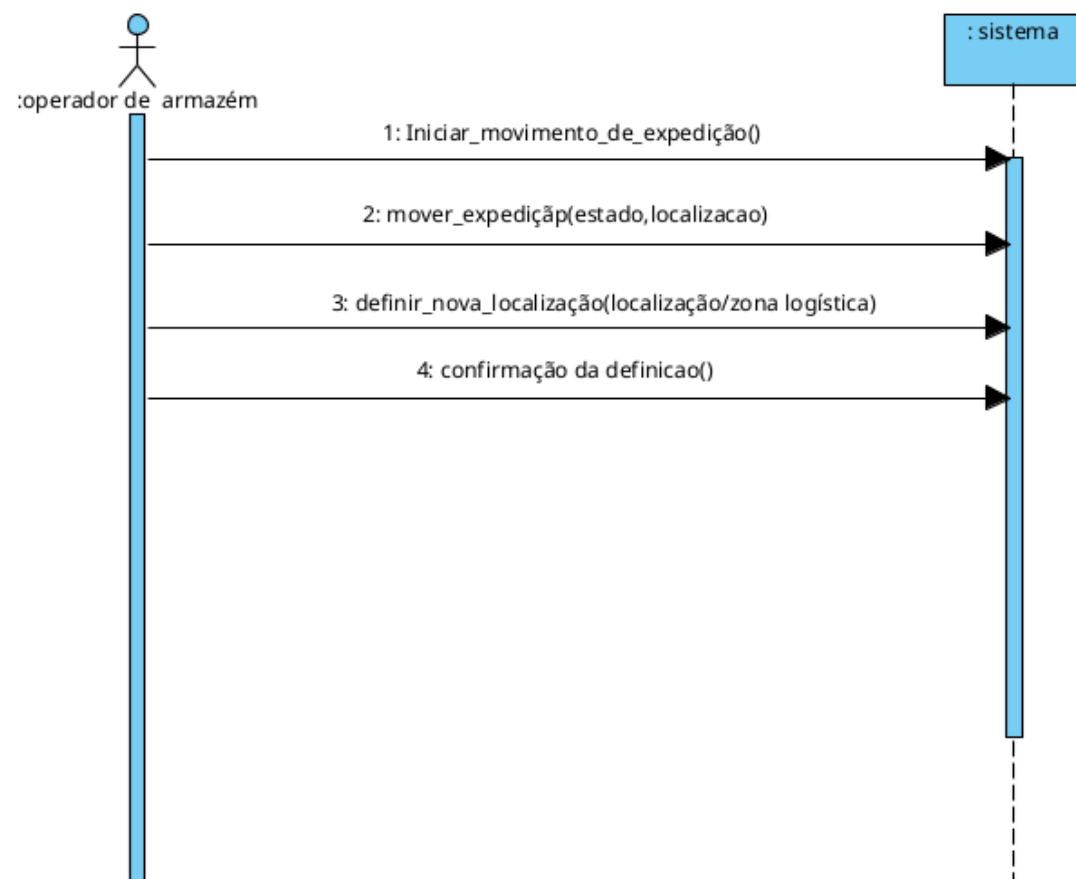




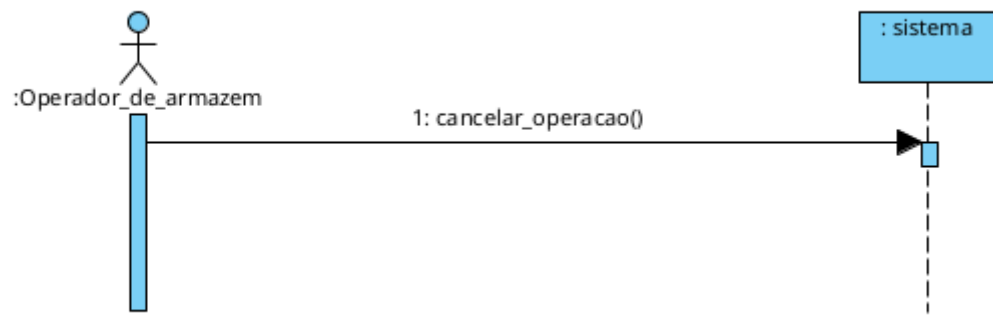
UC10:



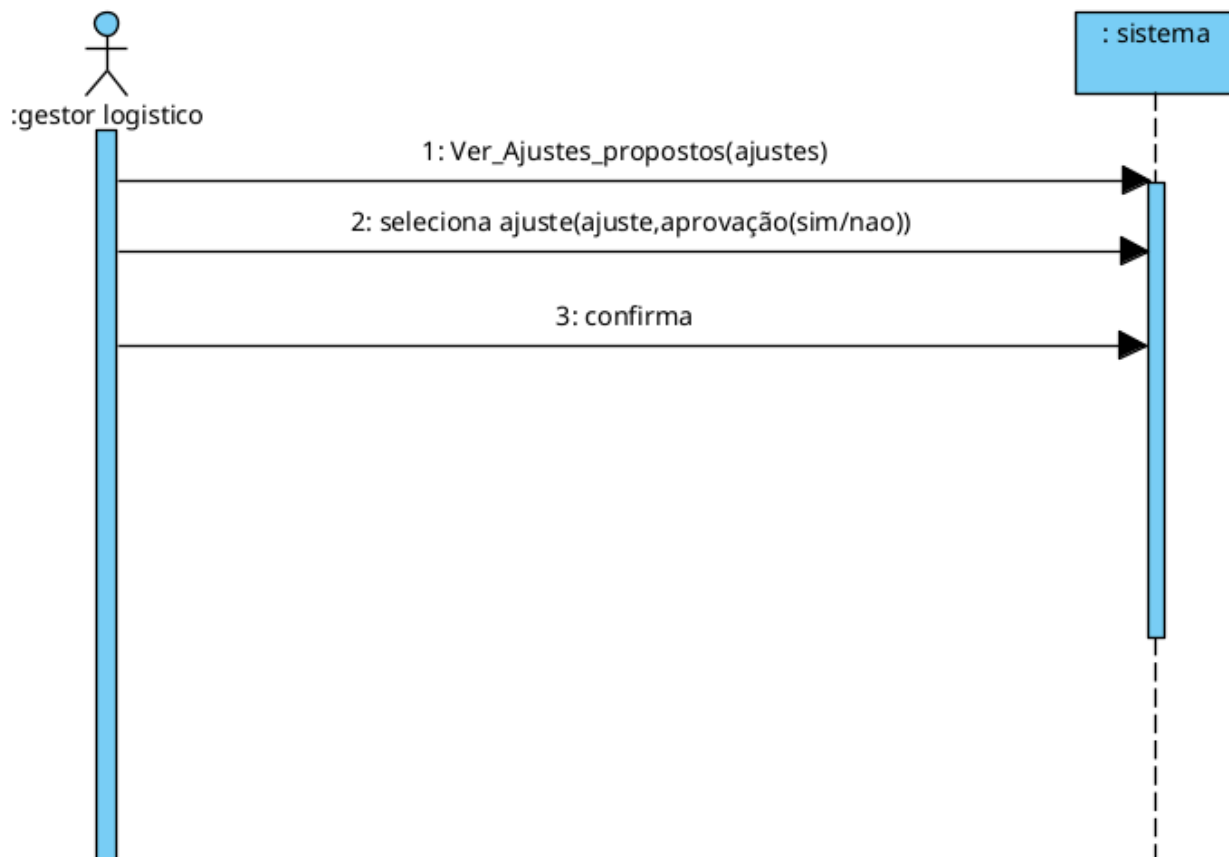
UC11:



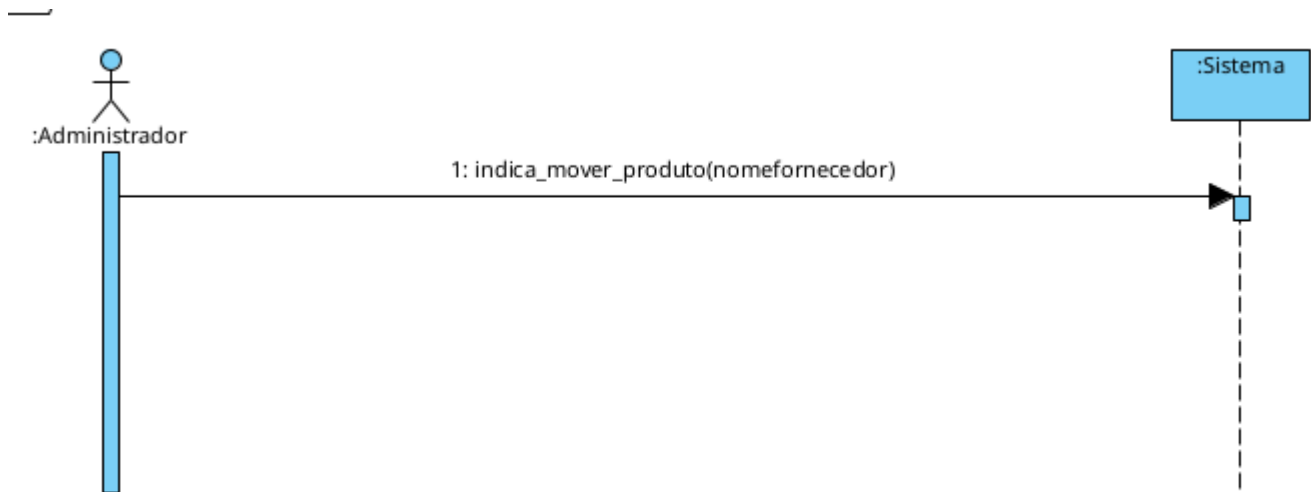
UC12:



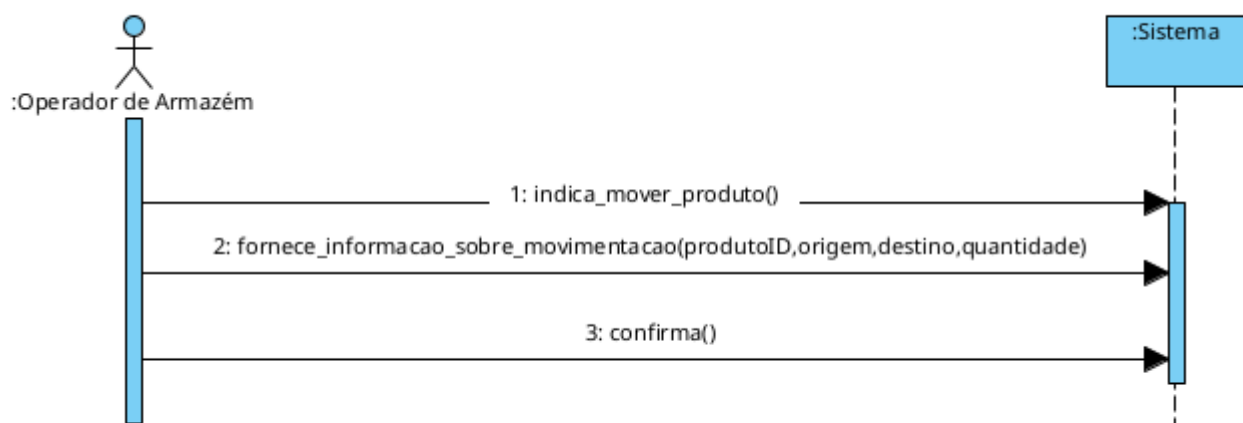
extensão UC12:



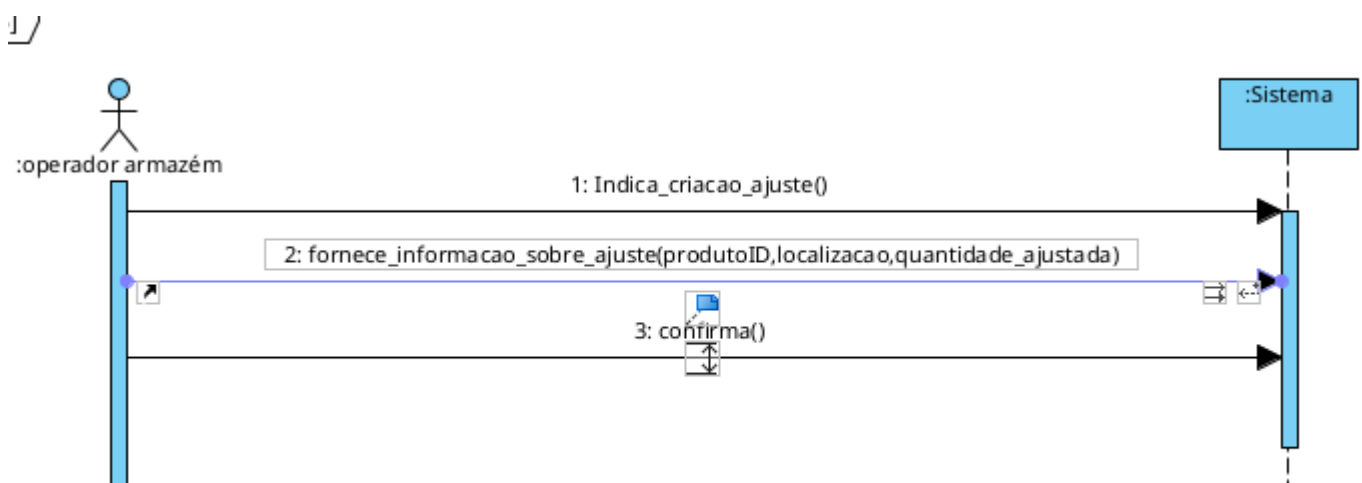
UC13:



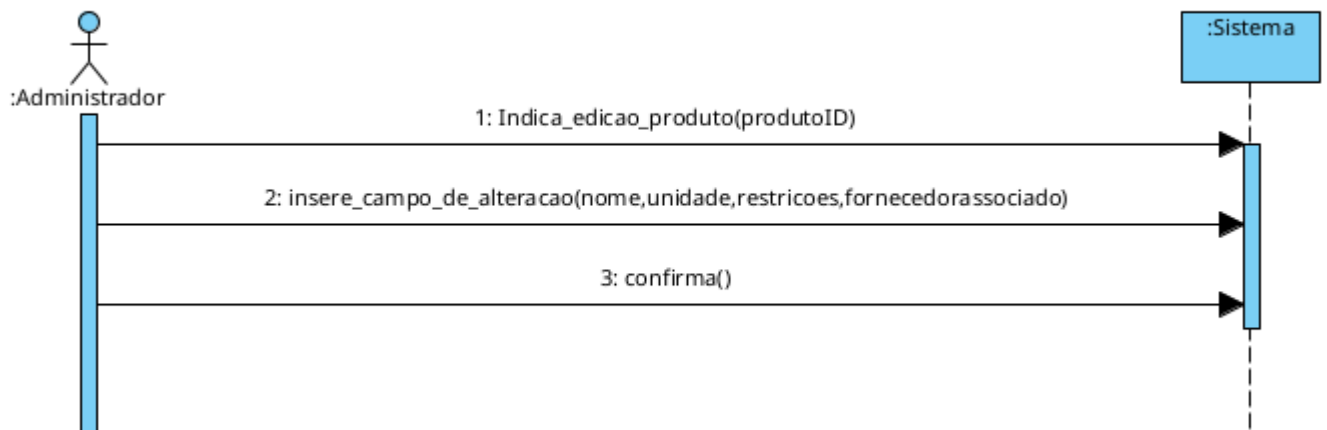
#### UC14:



#### UC15:



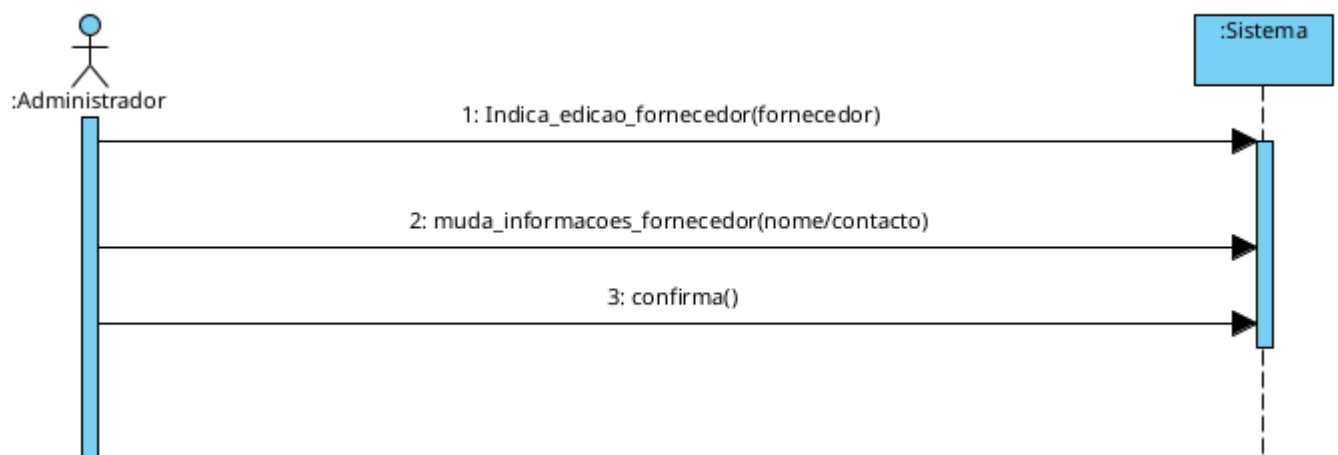
#### UC16:



UC17:



UC18:

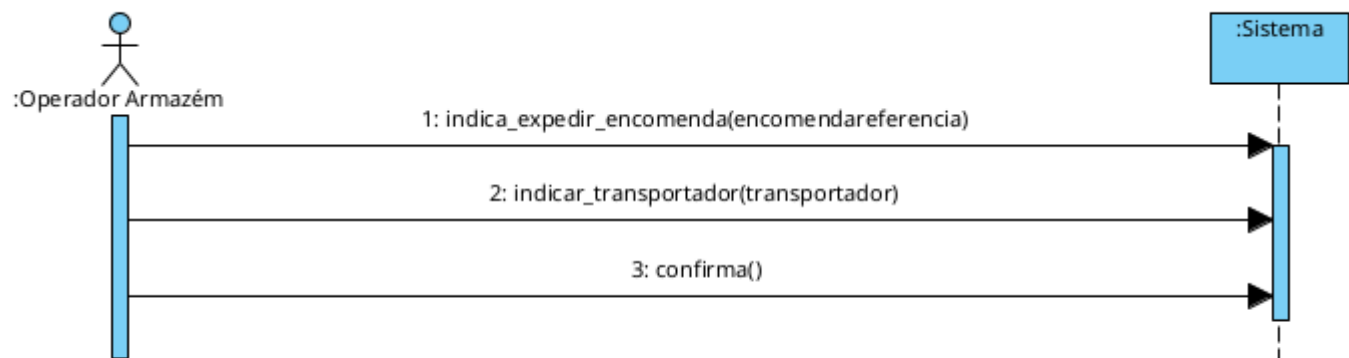


## UC19:

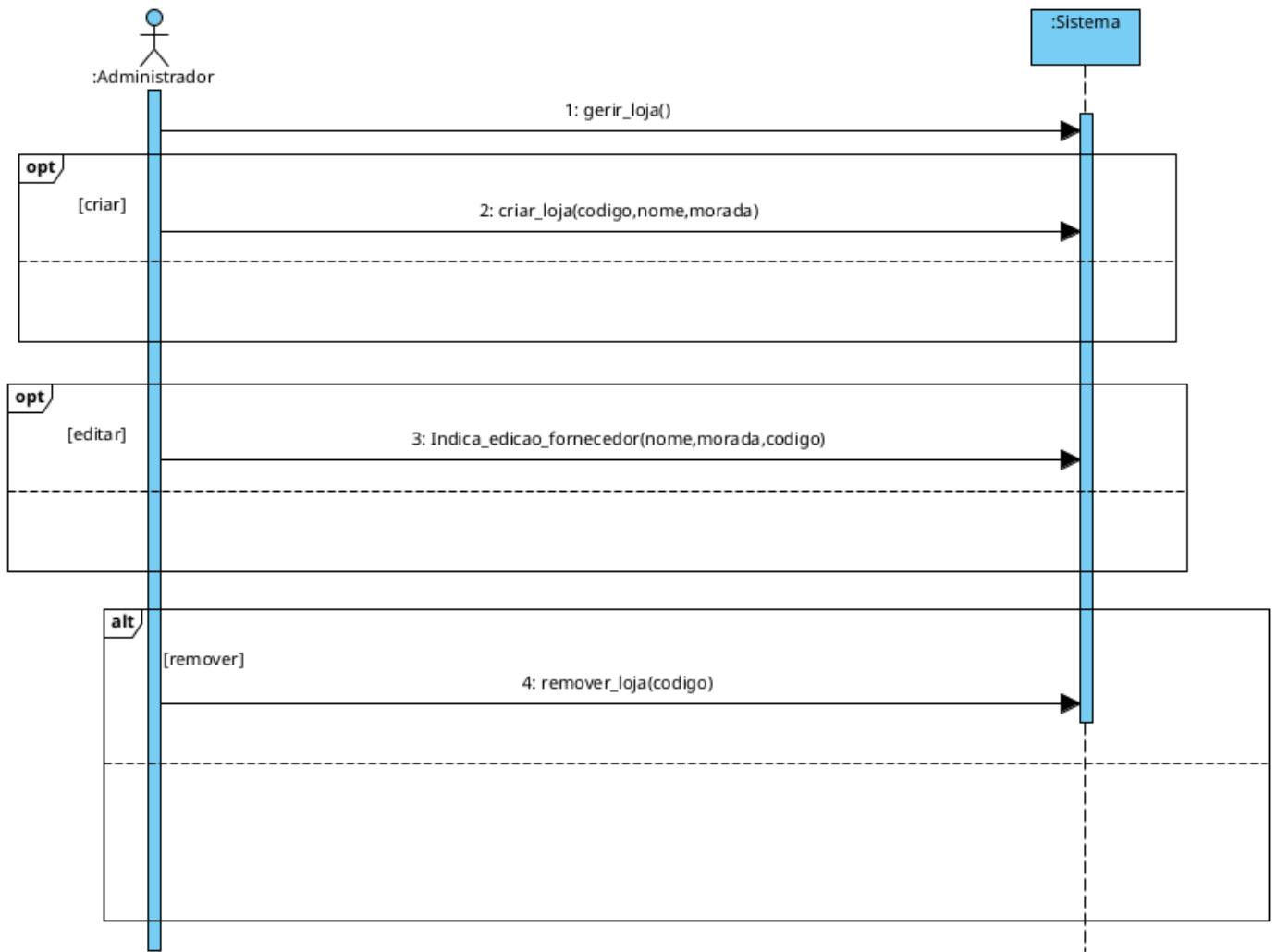
C20]



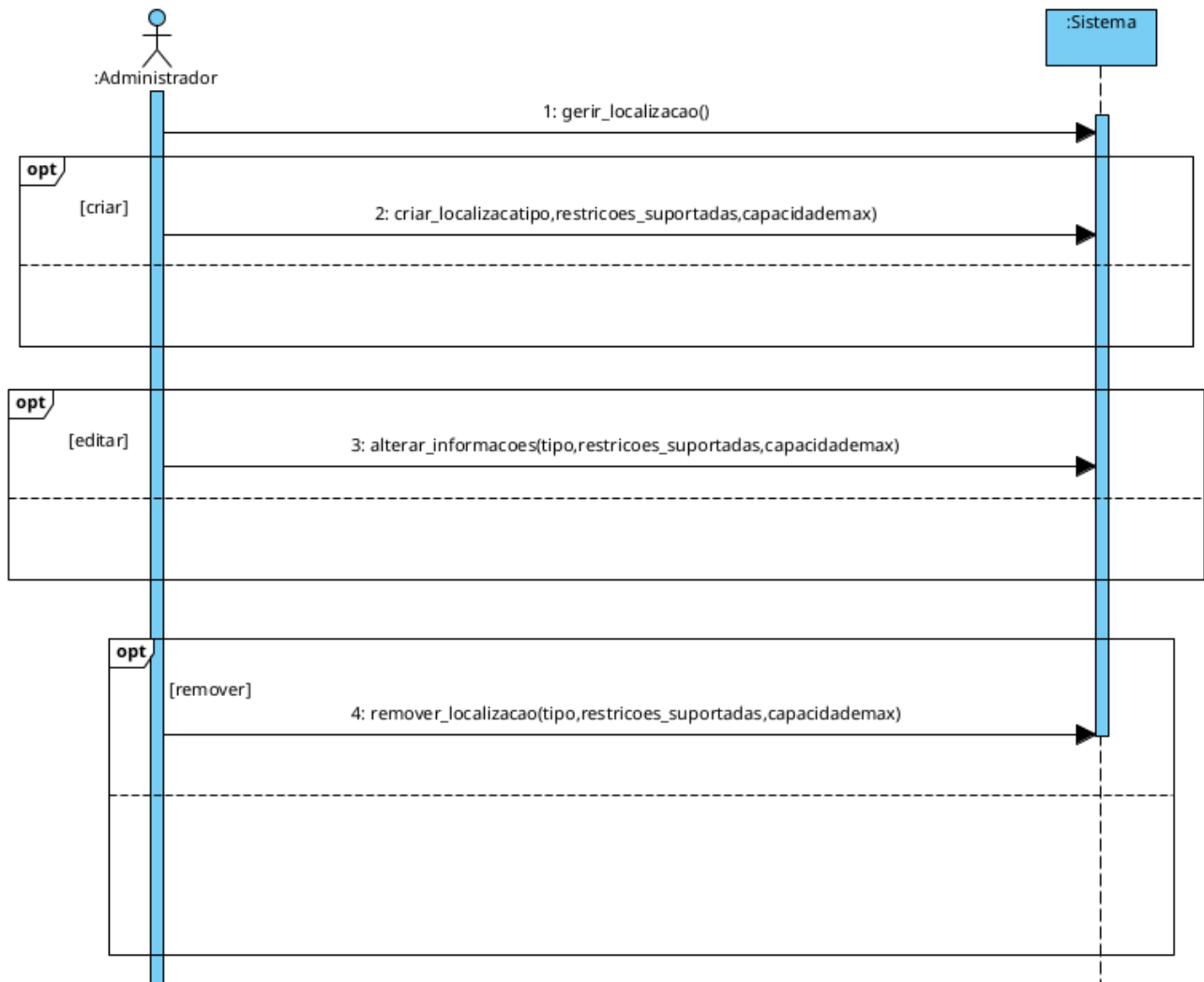
## UC20:



## UC21:



UC22:



UC23:

**Alterações:** Nos diagramas de sequência do sistema do UC01 a UC14 inclusive , foram removidas as respostas do sistema, as funcionalidades que o sistema fazia a si próprio e em relação aos restantes foram apenas adicionados os mesmos visto que não existiam anteriormente a primeira entrega.

#### Contratos das operações do sistema

Nesta secção estão os contratos desenvolvidos para todas as operações.

contratos projeto

#### UC01:

operação: inicio\_sessao(nome,password)

referência cruzada:caso\_uso: iniciar sessao



precondições:

existe uma instância **u** de utilizador com atributos **u.nome** = nome e

**u.password**= password

pós condições:

## UC02:

operação: iniciar\_registo()

referencia cruzada: caso\_uso: registar\_utilizador

precondições:

existe uma instância **a** de administrador.

pós condições:

foi criada uma instância **u** de utilizador

Foi estabelecida uma associação "registar" entre **u** e **a**.

operação: registar\_utilizador(nome,email,password,papel)

referencia cruzada: caso\_uso: registar\_utilizador

precondições:

existe uma instancia **u** de utilizador.

pós condições:

Os atributos **u.nome**, **u.email**, **u.password**, **u.papel** foram alterados respectivamente para nome, email, password, papel.

## UC03:

operação: criar\_fornecedor()

referência cruzada: caso\_uso: criar\_fornecedor

precondições:

existe uma instância **a** de administrador

pós condições:

Foi criada uma instancia **f** de fornecedor.

Foi estabelecida uma associação "criar" entre **administrador** e

## fornecedor.

operação: fornecer\_dados(contacto, nome)

referência cruzada: caso\_uso: criar\_fornecedor

precondições:

existe uma instancia **f** de fornecedor.

pos condicoes:

Os atributos **f.nome** e **f.contacto** foram alterados respectivamente para nome e contacto.

operação: confirmar\_criacao()

referência cruzada: caso\_uso: criar\_fornecedor

precondições:

pós condições:

a associação "criar" entre **f** e **a** foi destruída

## UC04:

operação: indica\_consulta\_fornecedor()

referência cruzada: caso\_uso: Consultar fornecedor

precondições:

existe uma instancia **u** de utilizador.

existe uma instância **f** de fornecedor.

pós condições:

foi estabelecida uma associação "visualiza" entre **u** e **f**

operação: terminar\_consulta()

referência cruzada: caso\_uso: Consultar fornecedor

precondições:

pós condições:

a associação "visualiza" entre **u e f** foi destruída

## UC05:

operação:iniciar\_criacao\_produto()

referência cruzada:caso\_uso: criar\_produto

precondições:

existe uma instância **a** de administrador.

pós condições:

foi criada uma instância **p** de produto e os seus atributos foram inicializados.

foi estabelecida uma associação "cria" entre **a e p**.

operacao:dados\_produto(SKU,nome,unidade\_medida,validade,restricoes)

referência cruzada:caso\_uso: criar\_produto

precondições:

existe uma instância **p** de produto.

pós condições:

os atributos **p.SKU**, **p.nome**, **p.unidade\_medida**, **p.validade** e **p.restricoes** foram alterados respectivamente para SKU , nome , unidade\_medida, validade, restricoes

operação:confirmar\_criacao()

referência cruzada:caso\_uso: criar\_produto

precondições:

pós condições:

a associação "cria" entre **p e a** foi destruída

operação:cancelar operacao()

referência cruzada:caso\_uso:criar produto

precondições:

pós condições:

a instância **p** de produto foi destruída

## UC6:

operação: indica\_consulta\_produtos()

referência cruzada:caso\_uso: Consultar produtos

precondições:

existe uma instancia **u** de utilizador.

existe uma instância **p** de produto.

pós condições:

foi estabelecida uma associação "visualiza" entre **u e p**

operação: fim\_consulta()

referência cruzada:caso\_uso: Consultar produtos

precondições:

pós condições:

a associação "visualiza" entre **u e p** foi destruída

## UC7:

operação: iniciar\_registo\_rececao()

referência cruzada:caso\_uso: registar\_rececao

precondições:

existe uma instancia **or** de operador de recepção.

pós condições:

Foi criada uma instância **r** de recepção e seus atributos foram inicializados.

foi estabelecida uma associação "registra" entre **or** e **r**

operação: associar\_fornecedor(fornecedornome)

referência cruzada: caso\_uso: registrar\_rececao

precondições:

existe uma instância **f** de fornecedor com o atributo **f.nome** =  
fornecedornome

existe uma instância **r** de recepção

pós condições:

foi estabelecida uma associação "tem" entre **f** e **r** através de **f.nome**

operação: identificar\_produtos(lote, validade, quantidade)

referência cruzada: caso\_uso: registrar\_rececao

precondições:

existe uma instância **r** de recepção.

existe uma instância **lr** de linhas de recepção.

pós condições:

Foi criada uma instância **lr** de linhas de recepção.

Foi estabelecida uma associação "tem" entre **r** e **lr**.

o atributo **lr.lote** foi alterado para lote

o atributo **lr.quantidade\_recebida** foi alterado para  
quantidade\_recebida

o atributo **lr.validade** foi alterado para validade

operacao: finalizar\_operacao\_de\_identificar\_produto()

referência cruzada: caso\_uso: registrar\_rececao

precondições:

pós condições:

a associação "tem" entre **r** e **lr** foi destruída

operação: cancelar\_operacao()

referência cruzada:caso\_uso: registrar\_rececao

precondições:

pós condições:

a instancia **r** de registo foi destruída

operacao: registo\_de\_inconformidades(lote,tip,descricao)

referência cruzada:caso\_uso: registrar\_rececao

precondições:

existe uma instância **r** de recepção

existe uma instância **nc** de não conformidades

pós condições:

z

o atributo **lr.lote** foi alterado para lote

o atributo **nc.tipo** foi alterado para inconformidades.

o atributo **nc.descricao** foi alterado para descrição.

Foi estabelecida uma associação "registada em" entre **r e nc**.

operacao: finalizar\_registo\_incoformidades()

referência cruzada:caso\_uso: registrar\_rececao

precondições:

pós condições:

a associação "registada em" entre **r e nc** foi destruída

operação: finalizar\_registo()

referencia cruzada:caso\_uso: registrar\_rececao

precondições: .

pós condições:

a associação "registra" entre **or e r** foi destruída

## UC8:

operação: indica\_consulta\_rececoes()

referência cruzada: caso\_uso: Consultar rececoes

precondições:

existe uma instancia **u** de utilizador.

existe uma instancia **r** de recepção.

pós condições:

foi estabelecida uma associação "visualiza" entre **u e r**

operação: fim\_consulta()

referência cruzada: caso\_uso: Consultar rececoes

precondições:

pós condições:

a associação "visualiza" entre **u e r** foi destruída

## UC9:

operação: inicio\_registo\_encomenda()

referência cruzada: caso\_uso: registrar encomenda

precondições:

existe uma instância **gl** de gestor logístico

existe uma instancia **en** de encomenda.

pós condições:

Foram inicializados os atributos de en.

Foi estabelecida uma associação "registra" entre **gl e en**.

operação: dados\_encomenda(loja,prioridade,produtosku,quantidade)

referência cruzada: caso\_uso: registrar encomenda

precondições:

Existe uma instância **le** de linha de encomenda com os atributos

**le.produtoID** = produtosku e **le.quantidade** = quantidade.

existe uma instancia **en** de encomenda com os atributos **en.lojaID** = lojaID e **en.prioridade** = prioridade

pós condições:

Foi estabelecida uma associacao "registra" entre **en e le**.

operação: fim\_registro\_encomenda()

referência cruzada: caso\_uso: registrar encomenda

precondições:

pós condições:

a associação "registra" entre **gl e en** foi destruída

operação: cancelar operacao()

referência cruzada: caso\_uso: registrar encomenda

precondições:

pós condições:

a instancia **en** foi destruída

## UC10:

operação: dados(encomendaID)

referência cruzada: caso\_uso: preparar expedicao

precondições:

existe uma instancia **en** de encomenda com **en.referencia** = encomendaID

existe uma instância **ex** de expedição

existe uma instancia **os** de operador de seleção.

pós condições:



Foi estabelecida uma associação "associado" entre **en e ex**.

Foi estabelecida uma associação "prepara" entre **ex e os**.

operação: acabou\_de\_encomendas()

referência cruzada:caso\_uso:preparar expedicao

precondições:

existe uma instância **ex** de expedição

pós condições:

a associação "associado" entre **en e ex** foi destruída

operação: executar\_tarefas()

referência cruzada:caso\_uso:preparar expedicao

precondições:

existe uma instância **t** de tarefa

existe uma instancia **os** de operador de seleção.

pós condições:

foi estabelecida uma associação "executado por " entre **os e t**

operação: tarefas\_concluida()

referência cruzada:caso\_uso:preparar expedicao

precondições:

existe uma instância **t** de tarefa

pós condições:

a associação "executado por " entre **os e t** foi destruída

operação: confirmar\_da\_encomenda()

referência cruzada:caso\_uso:preparar expedicao

precondições:

existe uma instância **ex** de expedição

pós condições:

a associação "prepara" entre **ex** e **os** foi destruída

## UC11:

operacao: indica\_consulta\_inventario\_por\_localizacao()

referência cruzada:caso\_uso: Consultar inventário por localização

precondições:

existe uma instância **u** de utilizador.

existe uma instância **upl** de inventário por localização

pós condições:

foi estabelecida uma associação "visualiza" entre **u** e **upl**

operacao: fim\_consulta\_inventario\_por\_localizacao()

referência cruzada:caso\_uso: Consultar inventário por localização

precondições:

pós condições:

a associação "visualiza" entre **u** e **upl** foi destruída

## UC12:

operacao: iniciar\_movimento\_de\_expedicao()

referência cruzada:caso\_uso: mover expedicao

precondições:

existe uma instância **oa** de operador de armazém

existe uma instância **ex** de expedição

pós condições:

Foi estabelecida uma associação "move" entre **oa** e **ex**.

operação: mover\_expedicao(estado,localização)

referência cruzada:caso\_uso:mover expedição

precondições:

existe uma instância **ex** de expedição

pós condições:

o atributo **ex.estado** foi alterado para estado

o atributo **ex.localização** foi alterado para localização

operação: definir\_nova\_localizacao(localizacao)

referência cruzada: caso\_uso: mover expedição

precondições:

existe uma instância **ex** de expedição

pós condições:

o atributo **ex.localização** foi alterado para localização

operação: confirma()

referência cruzada: caso\_uso: mover expedição

precondições:

pós condições:

a associação "move" entre **oa** e **ex** foi destruída.

operação: cancelar\_operacao()

referência cruzada: caso\_uso: mover expedição

precondições:

pós condições:

a instância **ex** de expedição foi destruída.

## UC13:

operação: ver\_asjustes\_propostos(ajustes)

referencia cruzada: caso\_uso: aprovar\_ajuste

precondições:

existe uma instância **gl** de gestor logístico

Existe uma instância **a** de ajuste.

pós condições:

Foi estabelecida uma associação "aprova" entre **gl** e **a**.

operação: seleciona\_ajuste(ajustelD,aprovacao)

referencia cruzada:caso\_uso: aprovar\_ajuste

precondições:

existe uma instância **a** de ajuste

pós condições:

os atributos **a**.ajustelD e **a**.aprovacao foram alterados para ajustelD e aprovação respetivamente

operação: confirma()

referencia cruzada:caso\_uso: aprovar\_ajuste

precondições:

existe uma instância **gl** de gestor logístico

existe uma instância **a** de ajuste

pós condições:

a associação "aprova" entre **gl** e **a**

## UC14:

operação: indica\_consulta\_relatorio()

referência cruzada:caso\_uso: Consultar relatorio

precondições:

Existe uma instância **gl** de gestor logistico.

Existe uma instancia **r** de relatorio.

pós condições:

foi estabelecida uma associação "visualiza" entre **gl** e **r**

operação: fim\_consulta\_relatorio()

referência cruzada:caso\_uso: Consultar relatorio

precondições:

pós condições:

a associação "visualiza" entre **gl e r** foi destruída

## UC15:

operação: indica\_mover\_produto()

referência cruzada: caso\_uso: mover\_produto

precondições:

existe uma instância **oa** de operador de armazém

existe uma instância **p** de produto

existe uma instância **m** de movimentação

pós condições:

os atributos de **p** foram inicializados

foi estabelecida uma associação "faz" entre **oa e m**

operacao:

fornece\_informacao\_sobre\_informacao(produto\_id, origem, destino, quantidade)

referência cruzada: caso\_uso: mover\_produto

precondições:

existe uma instância **p** de produto com **p.sku = produto\_ID**

existe uma instância **m** de movimentação

pós condições:

foi estabelecida uma associação "tem" entre **m e p**

os atributos **m.origem** , **m.destino** e **m.quantidade\_mover** foi alterado respectivamente para origem , destino e quantidade

operação: confirma()

referencia cruzada:caso\_uso: mover\_produto

precondições:

pós condições:

a associação "faz" entre **oa** e **m** foi destruída

## UC16:

operação: indicar\_criacao()

referência cruzada:caso\_uso: criar ajuste

precondições:

existe uma instância **oa** de operador de armazem

pós condições:

foi criada uma instância **a** de ajuste e seus atributos foram

inicializados

Foi estabelecida uma associação "cria" entre **oa** e **a** .

operacao:

fornece\_informacao\_sobre\_ajuste(produtoID,localizacao,quantidade\_ajustada)

referência cruzada:caso\_uso: criar ajuste

precondições:

existe uma instância **a** de ajuste

pós condições:

os atributos **a.produto\_codigo** , **a.localizacao** , **a.quantidade** foram alterados para produtoID, localizacao,quantidade\_ajustada

operação: confirma()

referência cruzada:caso\_uso: criar ajuste

precondições:

pós condições:

a associacao "cria" entre **oa** e **a** foi destruída.

## UC17:

operação: indica\_edicao\_produto(sku)

referência cruzada: caso\_uso: editar produto

precondições:

existe uma instância adm de administrador

existe uma instância p de produto com p.sku = sku

pós condições:

foi estabelecida uma associação "edita" entre **adm** e **p**

operacao:

insere\_campo\_de\_alteracao(nome, unidade, restricoes, fornecedorassociado)

referência cruzada: caso\_uso: editar produto

precondições:

existe uma instância **p** de produto

pós condições:

os atributos **p.nome**, **p.unidade**, **p.restricoes**, **p.fornecedorassociado** foram alterados para nome, unidade, restricoes, fornecedorassociado respectivamente

operação: confirma()

referência cruzada: caso\_uso: editar produto

precondições:

pós condições:

a associação "edita" entre adm e p foi destruída

## UC18:

operação: indica\_remocao(sku)

referência cruzada:caso\_uso: remover produto()

precondições:

existe uma instância **p** de produto

existe uma instância **adm** de administrador

pós condições:

foi estabelecida uma associação "remover" entre **adm e p**

## UC19:

operação: Indica\_edicao\_fornecedor(fornecedorID)

referencia cruzada:caso\_uso: editar\_fornecedor

precondições:

existe uma instância **adm** de administrador

existe uma instância **f** de fornecedor com **f.nome** = fornecedorID

pós condições:

foi estabelecida uma associação "edita" entre **adm e f**

operação: mudar\_informacoes\_fornecedores(nome,contacto)

referência cruzada:caso\_uso: editar fornecedor

precondições:

existe uma instância **f** de fornecedor

pós condições:

os atributos **f.nome** e **f.contacto** foram alterados para nome e



contacto respectivamente

operação: confirma()

referência cruzada: caso\_uso: editar fornecedor

precondições:

None

pós condições:

a associação "edita" entre **adm** e **f** foi destruída

## UC20:

operação: remover\_fornecedor(nomefornecedor)

referência cruzada: caso\_uso: remover fornecedor()

precondições:

existe uma instância **f** de fornecedor

existe uma instância **adm** de administrador

pós condições:

foi estabelecida uma associação "remover" entre **adm** e **f**

## UC21:

operação: indica\_expedir\_encomenda(encomendareferencia)

referência cruzada:caso\_uso: expedir encomenda

precondições:

existe uma instância **oa** de operador de armazém

existe uma instancia **en** de encomenda com

**en**.referencia=encomendareferencia

pós condições:

foi estabelecida uma associacao "expedir" entre **oa** e **en**

operação:indicar\_transportador(transportador)

referência cruzada:caso\_uso: expedir encomenda

precondições:

existe uma instancia **en** de encomenda

pós condições:

o atributo **en**.transportador foi alterado para transportador

operação: confirma()

referência cruzada:caso\_uso: editar fornecedor

precondições:

None

pós condições:

a associação "edita" entre **adm** e **f** foi destruída

## UC22:

operação: gerir\_loja()

referência cruzada:caso\_uso: gerir loja

precondições:

existe uma instância adm de administrador

pós condições:

foi criada uma instância l de loja

foi estabelecida uma associação gere entre adm e l

## UC22a:

operação:criar\_loja(codigo,nome,morada)

referência cruzada:caso\_uso: gerir loja

precondições:

existe uma instância adm de administrador

pós condições:

foi criada uma instância l de loja e seus atributos l.nome , l.codigo e l.morada foram alterados para código,nome,morada

foi estabelecida uma associação "criar" entre l e adm

## UC22b:

operação:editar\_loja(codigo,nome,morada)

referência cruzada:caso\_uso: gerir loja

precondições:

existe uma instância adm de administrador

existe uma instância l de loja

pós condições:

foi estabelecida uma associação "editar" entre l e adm

os atributos l.nome , l.código e l.morada foram alterados para  
codigo,nome,morada

## UC22c:

operacao:remover\_loja(codigo)

referência cruzada:caso\_uso: gerir loja

precondições:

existe uma instância adm de administrador

existe uma instância l de loja com l.codigo = codigo

pós condições:

foi estabelecida uma associação "remover" entre l e adm

## UC23:

operação: indica\_operacao()

referência cruzada:caso\_uso:

precondições:

pós condições:

## UC23a:

operacao:criar\_localizacao(tipo,restricoes\_suportadas,capacidademax)

referencia cruzada:caso\_uso: gerir localizacao

precondições:

existe uma instância adm de administrador

pós condições:

foi criada uma instância l de localização atributos l.tipo,  
l.restricoes\_suportadas, l.capacidademax foram alterados para  
tipo,restricoes\_suportadas,capacidademax

foi estabelecida uma associação "criar" entre **I** e **adm**

## UC23b:

operação:editar\_localizacao(tipo,restricoes\_suportadas,capacidademax)

referencia cruzada:caso\_uso: gerir localizacao

precondições:

existe uma instancia **adm** de administrador

existe uma instância **I** de localização

pós condições:

foi estabelecida uma associação "editar" entre **I** e **adm**

os atributos **I.tipo**, **I.restrições suportadas**, **I.capacidade máx** foram alterados para **tipo,restricoes\_suportadas,capacidademax**

## UC23c:

operação:remover\_localização(tipo,restricoes\_suportadas,capacidadem  
ax)

referência cruzada:caso\_uso: gerir localização

precondições:

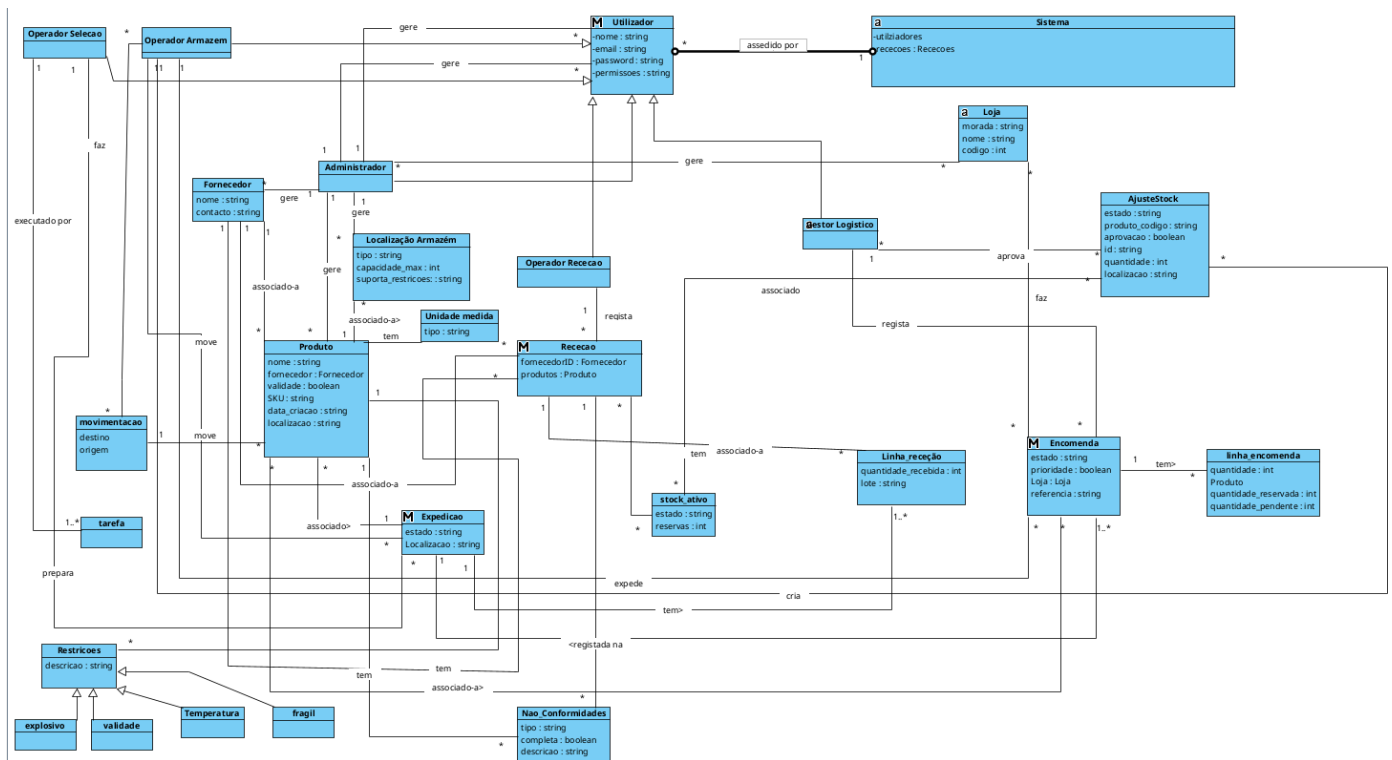
existe uma instância **adm** de administrador

existe uma instância **I** de localização com

pós condições:

foi estabelecida uma associação "remover" entre **I** e **adm**

Este capítulo descreve o modelo de domínio do sistema.



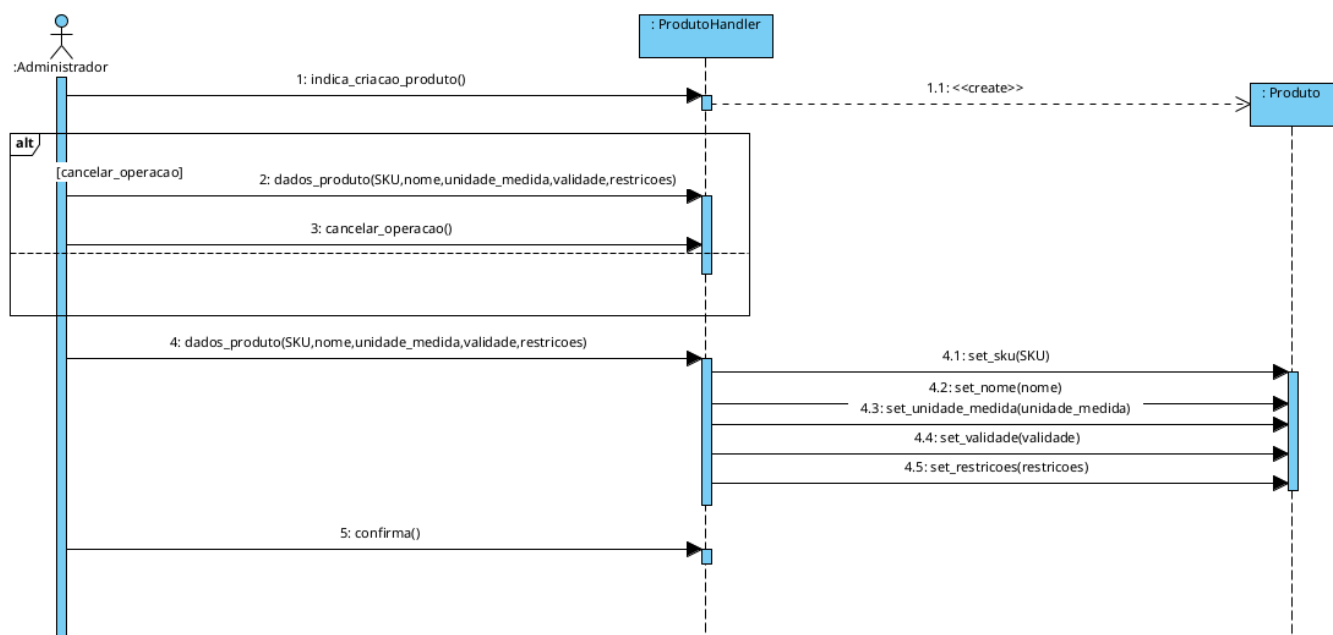
Alterações: Foram retiradas as classes conceituais confirmação recolha e de entrega , transportado e todas as classes conceituais de consulta pois não seriam úteis pois foram alteradas por associações. Após a adição dos novos casos de uso foram adicionadas as classes tarefa, não conformidades stock ativo , restrições e suas subclasses,linha encomenda e recepção,utilizador, sistema localização Armazém ,unidade de medida e movimentação.

## Modelo de desenho

Este capítulo, a entregar na segunda parte do projeto.

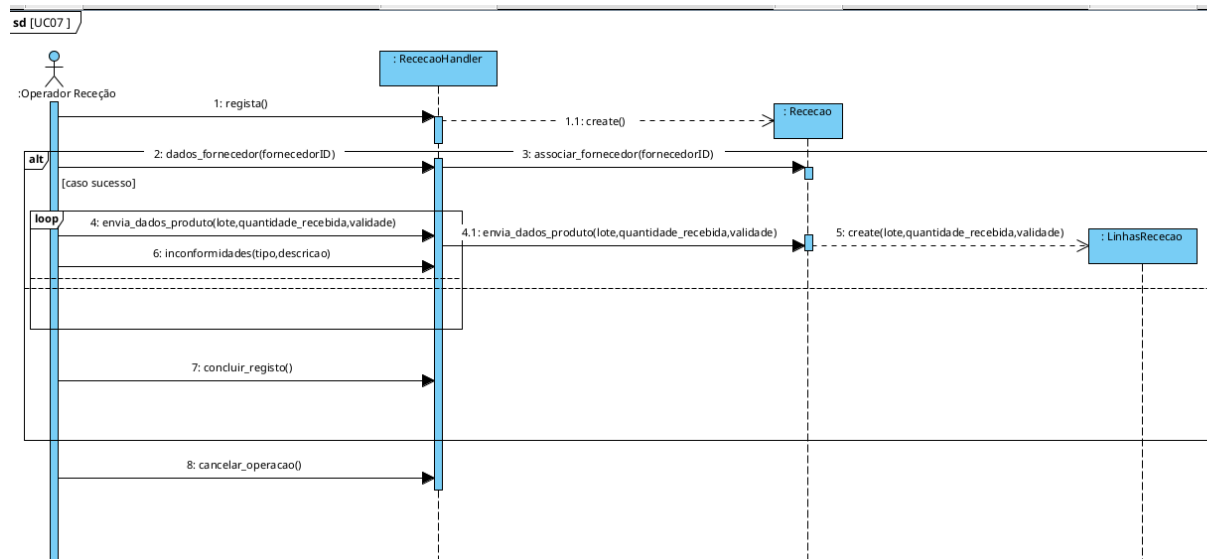
## Diagramas de interação

Esta secção deve conter os diagramas de interação para os casos de uso UC05, UC07, UC09, UC10 e UC12.



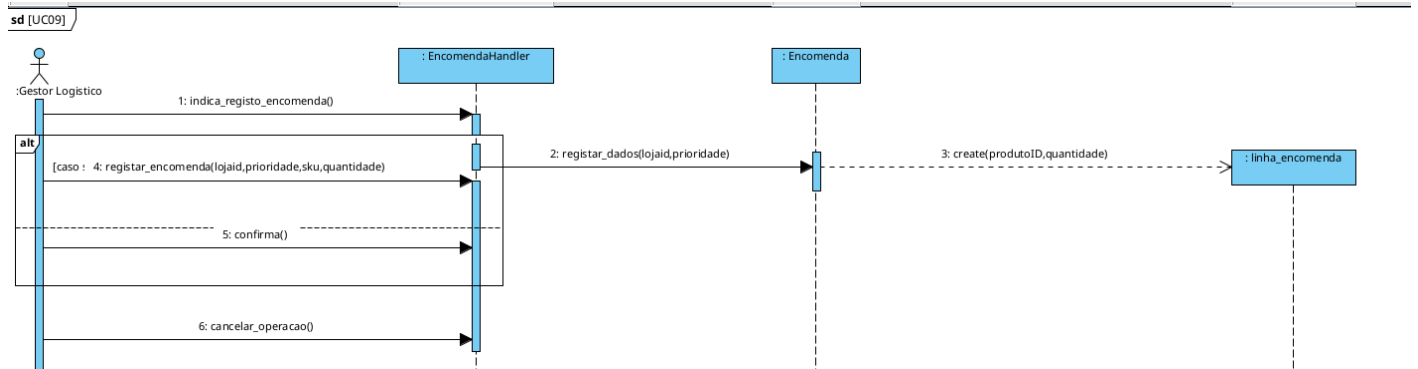
Padrões de graps: information expert é o ProdutoHandler visto que o mesmo sabe as informações dos produtos que são criados sendo ao mesmo tempo o creator pois o mesmo cria os produtos

UC05:



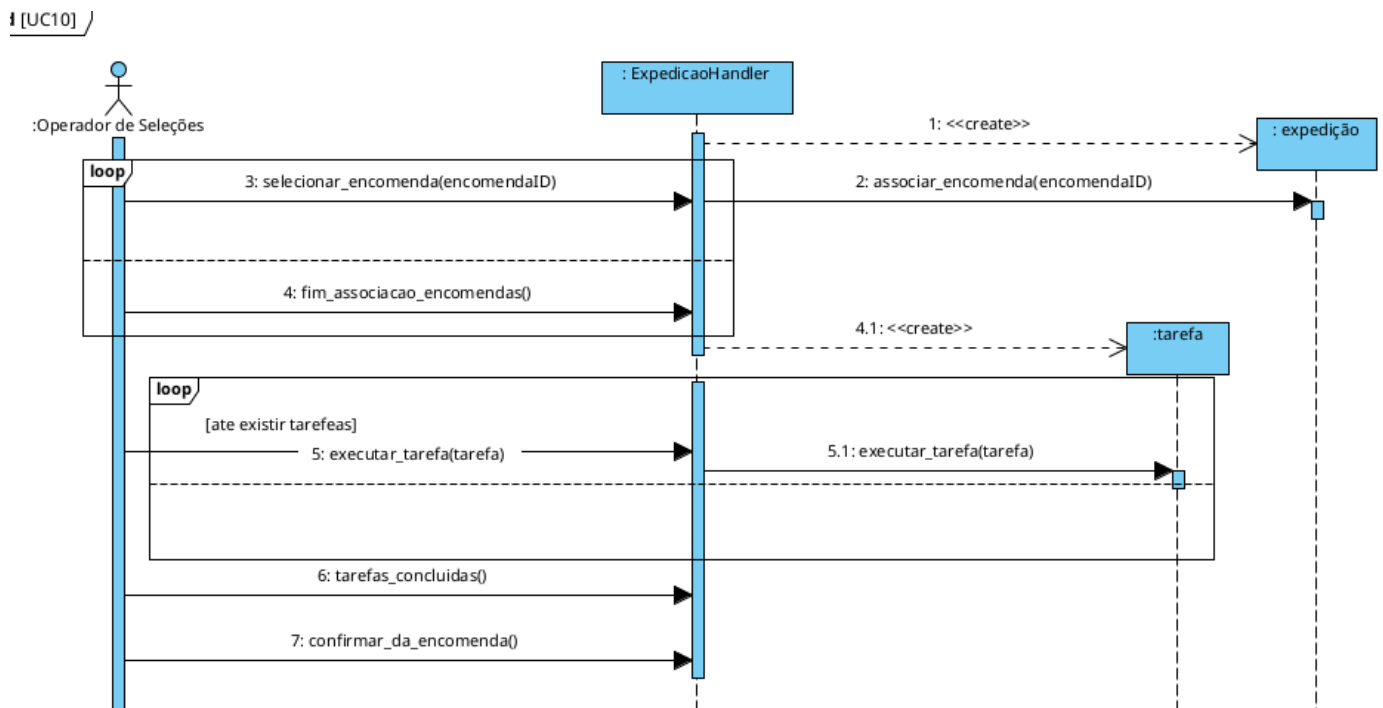
Padrões de graps: informação expert é o RececaoHandler visto que o mesmo sabe as informações das recessões que são criados sendo ao mesmo tempo o creator pois o mesmo cria as recepções.

UC07:



Padrões de graps: informação expert é o RececaoHandler visto que o mesmo sabe as informações das encomendas que são criados sendo a encomenda o creator pois a mesmo cria as linhas de encomenda através das informações enviadas pelo EncomendasHandler

UC09:



Padrões de graps: informação expert é o ExpedicaoHandler visto que o mesmo sabe as informações das expedições que são criados sendo o mesmo o creator pois a mesmo cria as expedições através das informações enviadas pelo operador de seleção para além de criar também tarefas que irão ser executadas pelo Operador de seleção.

UC10:





## Implementação da camada de domínio

Este capítulo, a entregar na terceira parte do projeto, descreve a implementação da camada de domínio do sistema, tendo em conta os artefactos UML apresentados anteriormente e seguindo o padrão arquitetural Model-View-Controller. As classes estão devidamente documentadas recorrendo ao javadoc. São explicadas as decisões tomadas ao nível da implementação que não estão detalhadas no desenho (e.g. acessibilidade das classes, redefinição de métodos da classe Object).

## Implementação da camada de apresentação

Este capítulo descreve a implementação da camada de apresentação do sistema para os casos de uso do modelo de domínio indicados segunda parte, seguindo o padrão arquitetural *Model-View Controller*.

## Testes e execução

Este capítulo descreve os testes implementados para os casos de uso UC03, UC05, UC07, UC09, UC10, UC12 e UC13 quais as situações anómalas e limitações encontradas durante a execução dos testes.

## Calendarização

13 de outubro de 2025 – Primeira entrega do relatório, contendo toda a análise da aplicação, exceto contratos.

18 de novembro de 2025 – Segunda entrega do relatório, contendo todo o desenho da aplicação, o modelo de domínio revisto e os contratos para todas as operações identificadas. Será atribuído um bónus a quem desenvolver os novos casos de uso.

até 9 de janeiro de 2026 – Entrega final do relatório e do código fonte.

Reuniões de acompanhamento facultativas mediante inscrição prévia no moodle ou por marcação por e-mail com os docentes e sujeitas a confirmação.

## Observações

A submissão do projeto pressupõe o compromisso de honra de que o trabalho foi feito pelos alunos que o assinam. Cada grupo de trabalho é responsável por assegurar que os seus trabalhos são atribuíveis apenas a si. Assim, qualquer tentativa de fraude (e.g. copiar ou deixar copiar trabalhos de desenho e programação) tem como consequência imediata a anulação do trabalho.

Hélia Guerra ([helia.mg.guerra@uac.pt](mailto:helia.mg.guerra@uac.pt))

Vítor Sousa ([viktor.hm.sousa@uac.pt](mailto:viktor.hm.sousa@uac.pt))