



Universidade do Minho

Departamento de Informática

Configura Fácil

Mestrado Integrado em Engenharia Informática

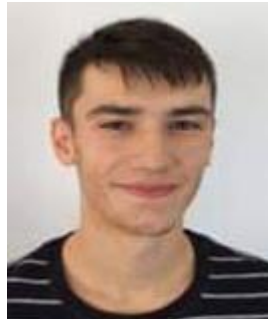
Desenvolvimento de Sistemas de Software

Grupo 28



(A80757)

Jaime Leite



(A81826)

João Marques



(A81918)

Nuno Rei



(A86266)

Rafael Lourenço

Janeiro, 2019

Índice

Índice

Introdução à segunda fase do projeto	4
Modelo de Domínio	5
Use Cases.....	6
Especificação dos Use Cases	7
Login	7
Criar Configuracao.....	7
Passo a passo.....	8
Pacote.....	8
Atualizar Stock.....	8
Adicionar Componente Essencial.....	9
Adicionar Componente Opcional	9
Máquina de Estados.....	11
Interfaces.....	11
Login	11
Informação Login efetuado	12
Informação Logout	12
Erro Credenciais inválidas	12
Atualizar Stock.....	13
Informação de atualização stock e configurações realizadas	13
Erro de armazém cheio	14
Erro de quantidade inválida	14
Criar Configuração (Componentes Essenciais).....	14
Aviso componentes do mesmo tipo.....	15
Configuração por pacotes	15
Informação pacote adicionado com sucesso	16
Aviso de existência de pacote já escolhido	16
Configuração por passo a passo.....	17
Aviso componentes necessários e/ou incompatíveis	17
Informação configuração adicionada com sucesso	18
Erro fila de configurações cheia	18
Aviso componentes iguais a pacote	18
Diagramas de Sequência de Sistemas e com Subsistemas	19
Atualizar Stock.....	19

Adicionar Componente Essencial	20
Adicionar Componente Opcional	21
Criar Configuração.....	23
Pacote.....	25
Passo a Passo.....	26
Diagrama de packages	28
Diagrama de Classes.....	30
Diagrama de classes/DAO	31
Diagramas de implementação	32
Login	32
AtualizarStock.....	33
Criar Configuracao.....	34
Pacote.....	36
Passo a Passo.....	38
Adicionar Componente Essencial.....	39
Adicionar Componente Opcional	40
calculaNecessidades.....	41
calculaIncompatibilidades.....	42
addCompStock	43
calculaProducoes.....	45
verificaCompPacote	46
Base de dados	48
JDBC.....	48
Conclusão	49

Introdução à segunda fase do projeto

Na continuação do trabalho realizado pelo grupo na primeira fase do projeto, o grupo desenvolveu o diagrama de classes, bem como o diagrama de classes DAO, diagramas de sequência, diagramas de implementação e por fim a interface.

Ao longo do desenvolvimento desta fase do trabalho, foram feitas algumas alterações e adições nos Use Cases, diagrama de classes e também em alguns diagramas de sequência para corresponderem da melhor maneira às especificações que se encontravam nos Use Cases.

Modelo de Domínio

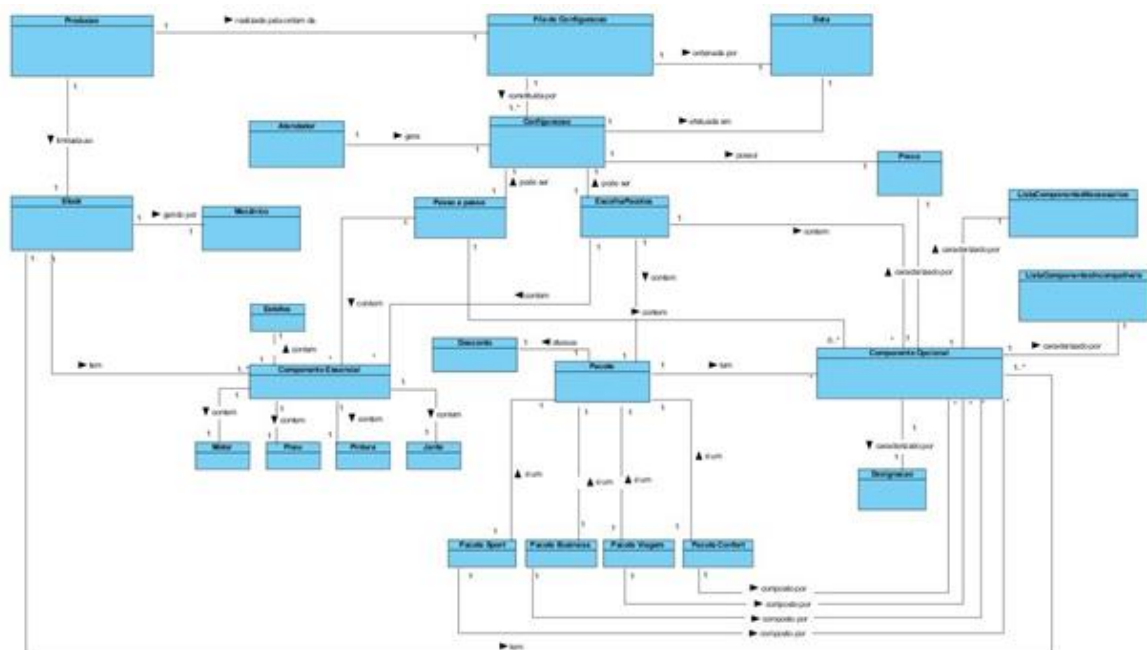
O modelo definido pelo grupo tem a seguinte análise:

A configuração, que é a parte central do modelo, é dividida por dois caminhos:

- Passo a passo, que contém componentes essenciais/básicos (motor, pneu, pintura, jante e estofos) e pode incluir componentes extra (caracterizados por preço, designação, lista de componentes que são incompatíveis com este - ListaComponentesIncompatíveis - e lista de componentes necessários para a introdução deste na configuração - ListaComponentesNecessarios);
- Escolha de Pacote, que também contém componentes de base, mas em alternativa ao passo anterior, pode conter um conjunto de componentes opcionais (Pacote). Este pode ter um de quatro tipos: Sport, Business, Viagem ou Confort. Sobre estes existe um dado desconto.

Existem dois tipos de funcionários: o mecânico que está na fábrica e o atendedor que está à entrada do stand. O primeiro tem como função a atualização do stock de um dado componente (essencial e opcional). A produção é limitada pelo stock e é feita pela ordem que uma configuração aparece na fila de configuração. A ordem é estabelecida pelo identificador atribuído à configuração quando esta é terminada, este identificador vai aumentando conforme são criadas novas configurações. O identificador volta a 1 quando não existir nenhuma configuração para ser executada.

O atendedor tem como única função criar uma configuração.



Use Cases

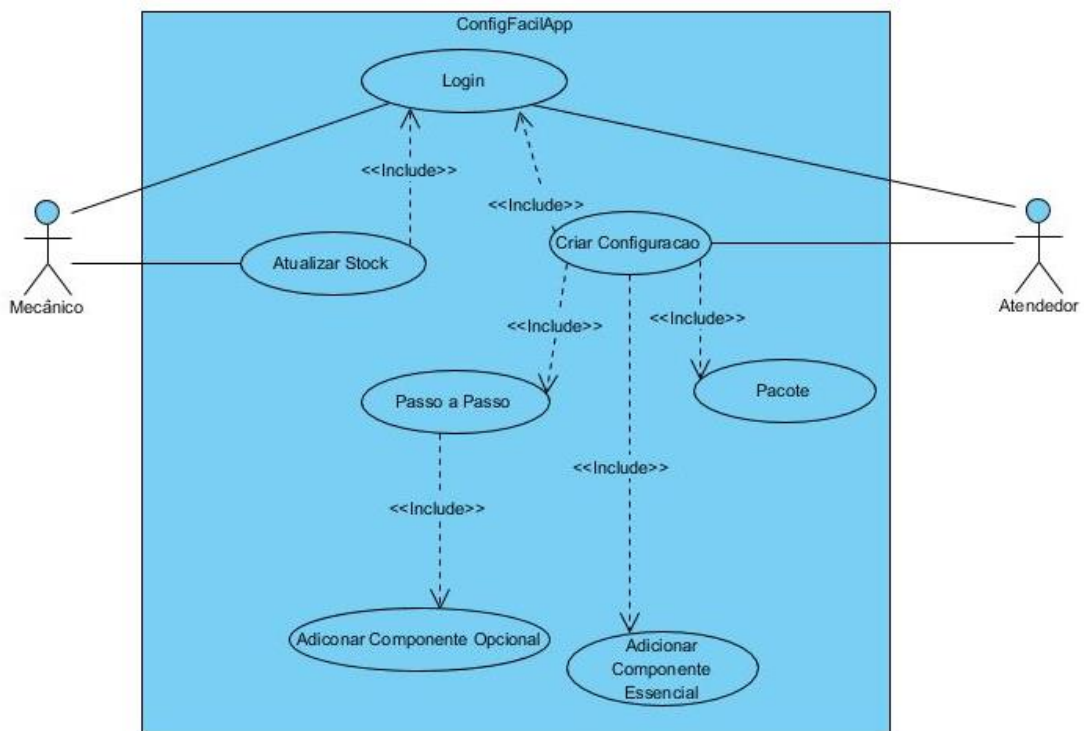
No modelo de use cases desenvolvido, consideramos dois atores, que são o mecânico e o atendedor.

O mecânico interage com o sistema, podendo atualizar o stock de um componente (Atualizar Stock).

O atendedor tem como única funcionalidade criar uma configuração. Esta criação envolve componentes essenciais (Adicionar Componente Essencial) e depois pode ser passo a passo (Passo a Passo) onde são colocados os componentes opcionais (Adicionar Componente Opcional) ou com pacote (Pacote) escolhendo um pacote dos disponibilizados.

Para qualquer uma destas ações, o utilizador necessita de estar autenticado no sistema (Login).

Agrupamos as funcionalidades em sete Use Cases:



Especificação dos Use Cases

Apresentam-se de seguida a especificação, que o grupo acha ser a melhor, de todos os Use Case criados pelo mesmo.

Login

O login é feito por qualquer uns dos atores para ficar autenticado no sistema e então a partir daí poder realizar as suas tarefas.

Use Case:	Login	
Actor:	Mecânico e Atendedor	
Pré condição:	Possuir as credenciais do funcionário	
Pós condição:	Ficar autenticado no sistema	
Cenário Normal	Actor input	System response
	1. Insere credenciais	
		2. Valida credenciais
		3. Informa logado
Exceção 1 [credenciais inválidas] (passo 2)		2.1 Informa sobre credenciais inválidas

Criar Configuracao

O atendedor cria uma configuração, precisando para isso de estar logado no sistema. É feito o include do Use Case que introduz os cinco componentes essenciais, à escolha do utilizador, na configuração. Esta pode passar a incluir um pacote (“«include» Pacote”) ou pode o utilizador continuar a escolher componentes opcionais para a sua configuração (“«include» Passo a Passo”). Caso a fila de configurações esteja lotada, o processo não é concluído com sucesso.

Use Case:	Criar Configuração	
Actor:	Atendedor	
Pré condição:	Estar logado	
Pós condição:	Realizar e introduzir configuração	
Cenário Normal	Actor input	System response
	1. «include» Login	
	2. Indicar componentes essenciais («include» Adicionar Componentes Essenciais)	
	3. Indica que quer utilizar pacote («include» Pacote)	
Alternativo 1 [quer passo a passo] (passo 3)		4. Insere configuração
	3.1 Indica que quer utilizar passo a passo («include» Passo a Passo)	
		3.2 Regressa a 4
Exceção 1 [Fila cheia] (passo 4)		4.1 Informa sobre Fila Cheia

Passo a passo

São introduzidas componentes opcionais (“«include» Adicionar Componente Opcional”) que depois são verificadas se correspondem a um pacote. Se corresponderem, o utilizador tem a opção de as substituir ou não pelo pacote. Para além disto, o atendedor pode não querer escolher nenhuma componente opcional, ficando a configuração sem peças extra

Use Case:	Passo a Passo	
Actor:	Atendedor	
Pré condição:	Ter atendedor livre para fazer a configuração	
Pós condição:	Terminar configuração	
Cenário Normal	Actor input	System response
	1. Indica que quer fazer configuração passo a passo	2. Apresenta componentes opcionais
	3. Indica componentes opcionais («include» Adicionar Componente Opcional)	4. Verifica se componentes iguais a pacote
		5. Regista Componentes
Alternativo 1 [não quer selecionar nenhum componente] (passo 3)	3.1 Indica que não quer adicionar componentes	3.2 Regressa a 5
Alternativo 2 [componentes opcionais iguais a pacote] (passo 4)	4.2 Indica que quer substituir	4.1 Pergunta se quer trocar por pacote
		4.3 Substitui componentes por pacote
		4.4 Regressa a 5
Alternativo 3 [não quer por pacote] (passo 4.2)	4.2.1 Indica que não quer substituir	4.2.2 Regressa a 6

Pacote

O atendedor tem a opção de escolher um pacote, que aglomera componentes opcionais, para introduzir na configuração. Tem para isso de estar autenticado no sistema.

Use Case:	Pacote	
Actor:	Atendedor	
Pré condição:	Ter atendedor livre para fazer a configuração	
Pós condição:	Terminar Configuração	
Cenário Normal	Actor input	System response
	1. Indica que quer fazer configuração com pacote	2. Apresenta pacotes
	3. Indica pacote	4. Regista pacote
Alternativo 1 [não seleciona pacote] (passo 3)	3.1 Indica que não quer pacote	3.2 Regressa a 5

Atualizar Stock

Quando há componentes que chegam à fábrica, o mecânico atualiza o stock dos mesmos. Na atualização do stock de um componente também são calculadas as configurações que podem ser produzidas, dando informação ao utilizador das mudanças ocorridas. A atualização pode não suceder caso o armazém se encontre cheio ou o utilizador indique algo que não corresponda a números inteiros.

Use Case:	Atualizar Stock	
Actor:	Mecânico	
Pré condição:	Saber componente e quantidade do componente a introduzir	
Pós condição:	Stock do componente atualizado	
Cenário Normal	Actor input	System response
	1. «include» Login	
	2. Indica componente e quantidade a introduzir	
		3. Atualiza componente
		4. Calcula configurações que podem ser realizadas
		5. Informa de atualização do componente e produções realizadas
Exceção 1 [armazém cheio] (passo 3)		3.1 Informa armazém cheio
Exceção 2 [quantidade inválida] (passo 3)		3.1 Informa quantidade inválida

Adicionar Componente Essencial

Inserção de um elemento obrigatório na configuração. O elemento escolhido pelo utilizador pode corresponder a um do mesmo tipo já inserido na configuração. Aqui pode continuar e introduzir este novo componente, removendo o mais antigo da configuração ou pode não fazer a introdução deste novo componente.

Use Case:	Adicionar Componente Essencial	
Actor:	Atendedor	
Pré condição:	Conhecer componente essencial a adicionar	
Pós condição:	Componente adicionado	
Cenário Normal	Actor input	System response
	1. Indica componente a adicionar	
		2. Verifica tipo componente
		3. Insere elemento
Alternativo 1 [componente mesmo tipo] (passo 2)	2.2 Indica que quer substituir	2.1 Informa sobre componente do mesmo tipo
		2.3 Remove componente antigo
		2.4 Regressa a 3
Exceção 1 [não quer substituir] (passo 2.2)	2.2.1 Indica que não quer substituir	

Adicionar Componente Opcional

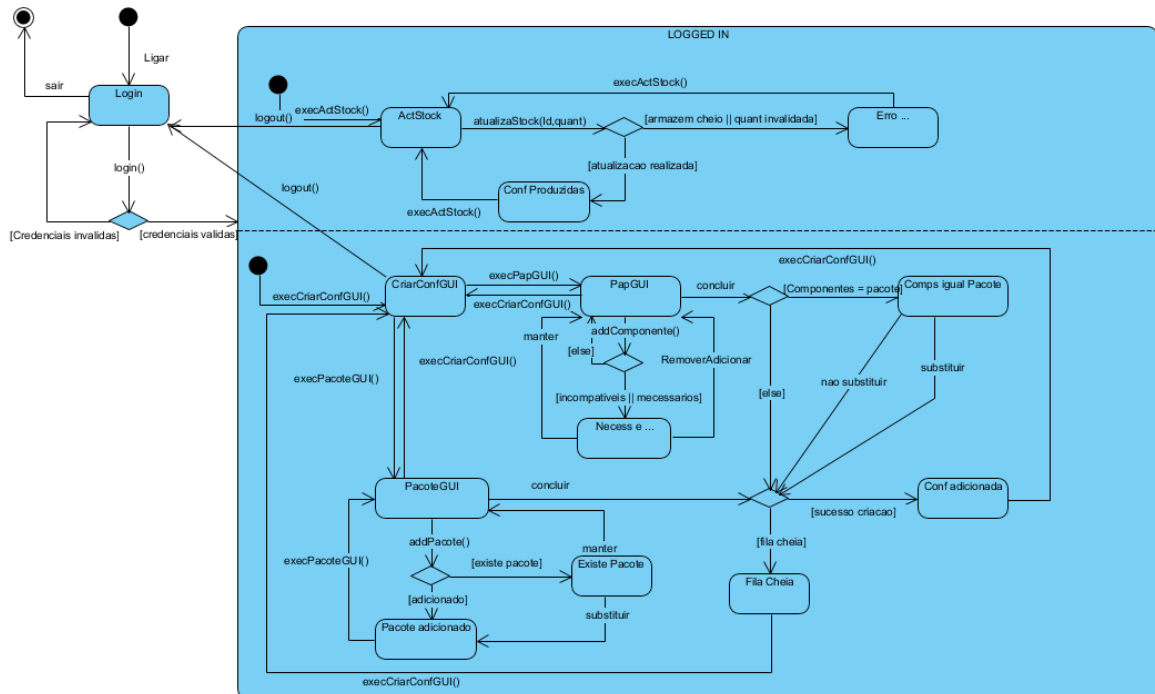
O atendedor indica o componente opcional a introduzir na configuração, sendo apresentados os elementos necessários e/ou incompatíveis com o mesmo. O utilizador tem a opção de remover os componentes incompatíveis e/ou introduzir os necessários para então introduzir inserir este novo elemento ou então não aceitar as modificações e deixar os componentes que estavam.

De realçar que as incompatibilidades e necessidades são apenas entre componentes opcionais.

Use Case:	Adicionar Componente Opcional	
Actor:	Atendedor	
Pré condição:	Conhecer componente opcional a adicionar	
Pós condição:	Componente adicionado	
Cenário Normal	Actor input	System response
	1. Indica componente a adicionar	
		2. Calcula necessidades e incompatibilidades
		3. Insere elemento na configuração
Alternativo 1 [componente necessario e/ou incompatível] (passo 2)		2.1 Informa componentes incompatíveis e/ou necessários
	2.2 Indica que quer componente atual	
		2.3 Remove incompatíveis e insere necessários
		2.4 Volta a 3
Exceção 1 [não quer componente atual] (passo 2.2)	2.2.1 Indica que não quer substituir componente	

Máquina de Estados


A máquina de estados relativa à primeira fase foi bastante mudada para simplificar a implementação. Sendo a nova máquina de estados explicada pelas transições entre interfaces que serão mostradas a seguir.



Interfaces

Os nomes atribuídos às interfaces que se seguem abaixo são os nomes que aparecem sobre as respectivas interfaces.

Login



Login

UserName:

PassWord:

Entrar

Fechar Aplicação

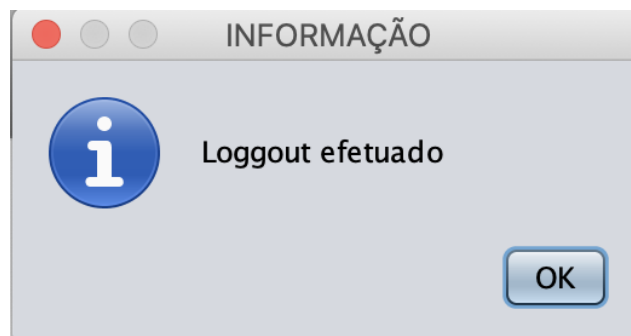
A interface “Login” é a interface inicial do sistema onde todos os funcionários da empresa se autenticam colocando as suas credenciais e clicando no botão “Entrar”. Após carregar no botão “Entrar” se as credenciais estiverem corretas aparece a interface “Informação Login efetuado”, em caso contrário aparece a interface “Erro Credenciais inválidas”. Caso pretenda fechar a aplicação pressiona o botão “Fechar Aplicação”. É através desta interface que se acede à interface “Criar Configuração (Componentes Essenciais)” e “Atualizar Stock”, a distinção para qual das duas encaminhar um funcionário é feita através da primeira letra no “UserName”.

Informação Login efetuado



Interface que mostra ao utilizador que as suas credenciais estavam válidas e que será redirecionado para o seu ambiente de trabalho.

Informação Logout



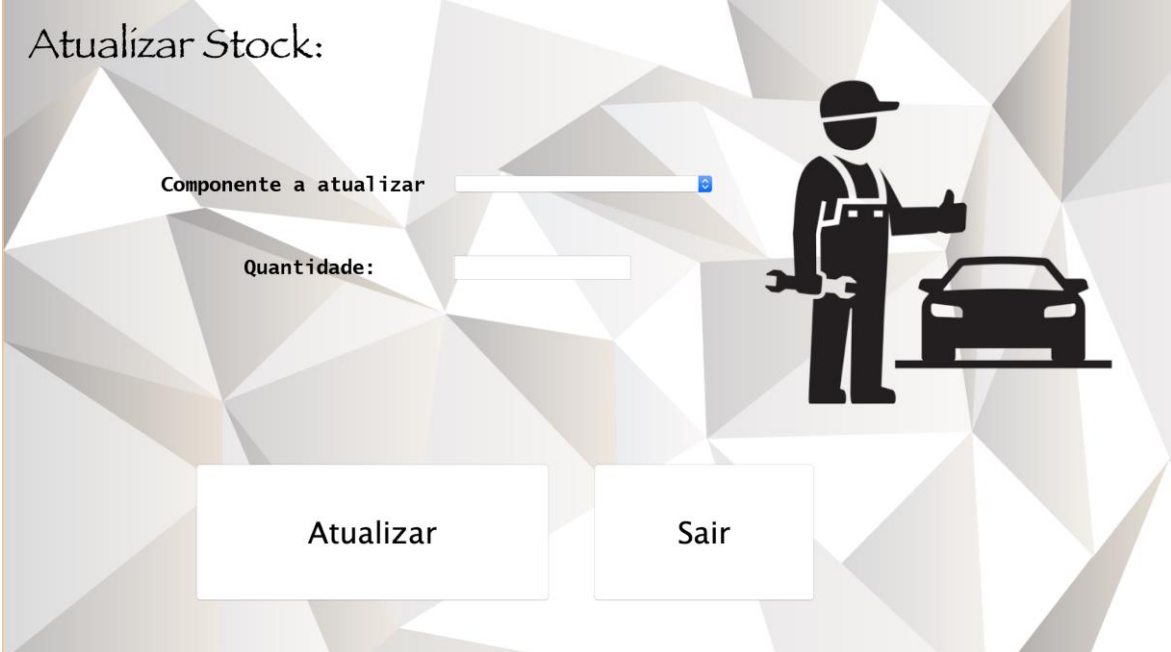
Interface que aparece ao utilizador quando este quer sair do seu ambiente de trabalho.

Erro Credenciais inválidas



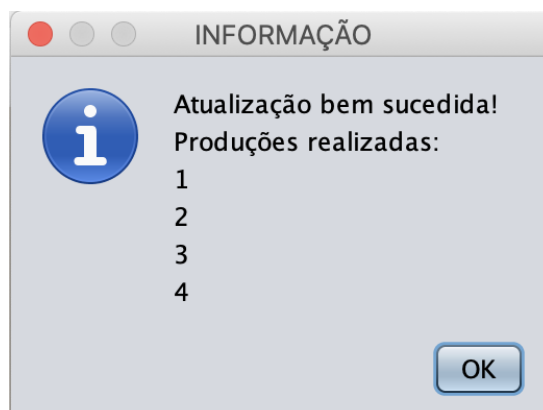
Mensagem de erro que mostra ao utilizador que este se enganou nas suas credenciais.

Atualizar Stock



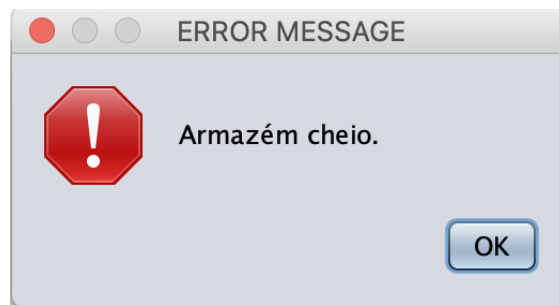
Após efetuar o login como mecânico é esta a interface que o utilizador pode utilizar. Aqui pode selecionar qual o componente que chegou a fábrica na ComboBox “Componente a Atualizar” e colocar a quantidade em “Quantidade”. Depois de mandar atualizar carregando no botão “Atualizar”, se tudo correu bem é mostrado ao utilizador quais foram as configurações realizadas através da interface “Informação de atualização stock e configurações realizadas”. Se por algum motivo correu algo de errado aparece uma das seguintes interfaces: “Erro de armazém cheio” ou “Erro de quantidade inválida”, quando o armazém se encontra lotado de componentes ou quando a quantidade não é um número inteiro, respetivamente. Clicando no botão “Sair” aparece a interface “Informação Logout” e volta para a interface inicial “Login”.

Informação de atualização stock e configurações realizadas



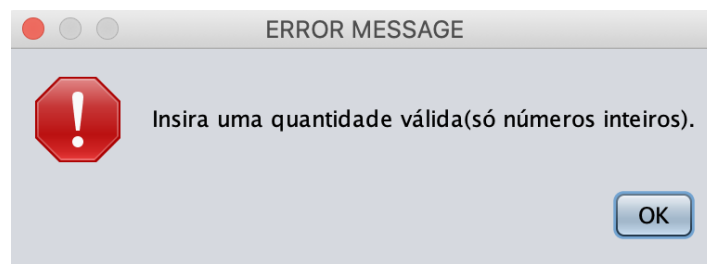
Informa o utilizador que o componente foi atualizado com sucesso e quais foram as configurações que foram possíveis realizar.

Erro de armazém cheio



Mostra ao utilizador que o armazém se encontra lotado para a quantidade de componentes que tentou introduzir.

Erro de quantidade inválida



Informa o utilizador sobre erro na inserção da quantidade do componente a utilizar.

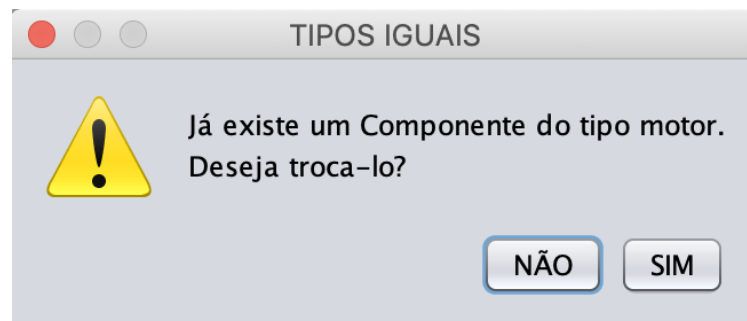
Criar Configuração (Componentes Essenciais)



Interface que aparece ao utilizador quando este está autenticado como atendedor de loja. Aqui o utilizador pode ir inserindo componentes essenciais na configuração que se encontram na lista da esquerda e estes vão aparecendo na lista da direita. Existem 5 tipos de componentes e não podem existir componentes do mesmo tipo numa configuração por isso quando o utilizador tenta inserir um componente e já existe um

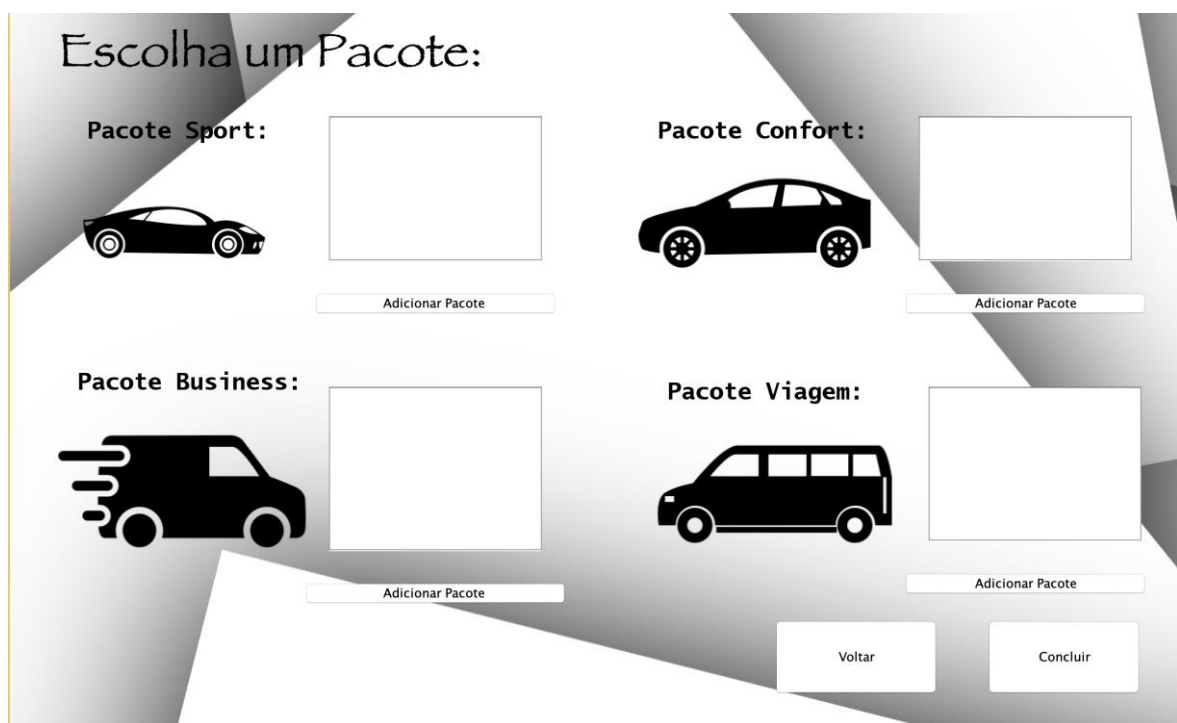
componente do mesmo tipo aparece a interface “Aviso componentes do mesmo tipo”. Quando o utilizador já tiver escolhido os 5 tipos de componentes essenciais este pode passar para a configuração por pacote ou passo a passo, clicando nos botões “Escolher Pacote” ou “Passo a Passo”, respetivamente.

Aviso componentes do mesmo tipo



Interface que informa o utilizador sobre componentes do mesmo tipo, se o utilizador escolher “SIM” os componentes são trocados, se escolher “NÃO” fica como estavam.

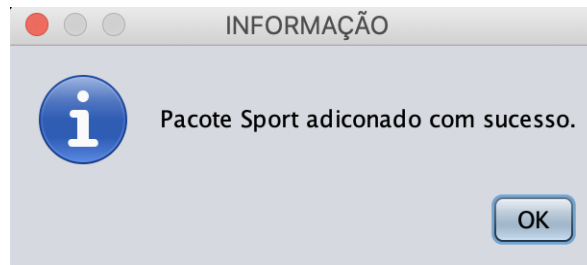
Configuração por pacotes



Quando o utilizador escolhe fazer configuração com pacote é esta interface que aparece podendo o utilizador escolher um dos pacotes Clicando no botão “Adicionar Pacote” relativo ao pacote que quer, mostrando-lhe a interface “Informação pacote adicionado com sucesso”. Se o utilizador mudar de ideias pode selecionar outro pacote onde lhe é mostrado a interface “Aviso de existência de pacote já escolhido”. Depois da decisão tomada o utilizador pode concluir a configuração ou voltar para trás caso queira mudar algum componente essencial, pressionando os botões “Concluir” ou “Voltar”, respetivamente. Quando carrega em “Concluir” pode aparecer a interface “Aviso

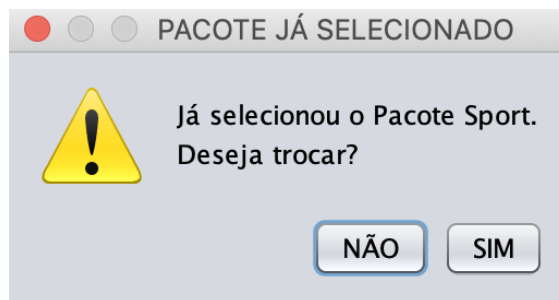
componentes iguais a pacote” onde o utilizador pode decidir se quer mandar os componentes ou substituir por um pacote. De seguida aparece a interface “Informação configuração adicionada com sucesso” e depois volta para a interface “Criar Configuração (Componentes Essenciais)”.

Informação pacote adicionado com sucesso



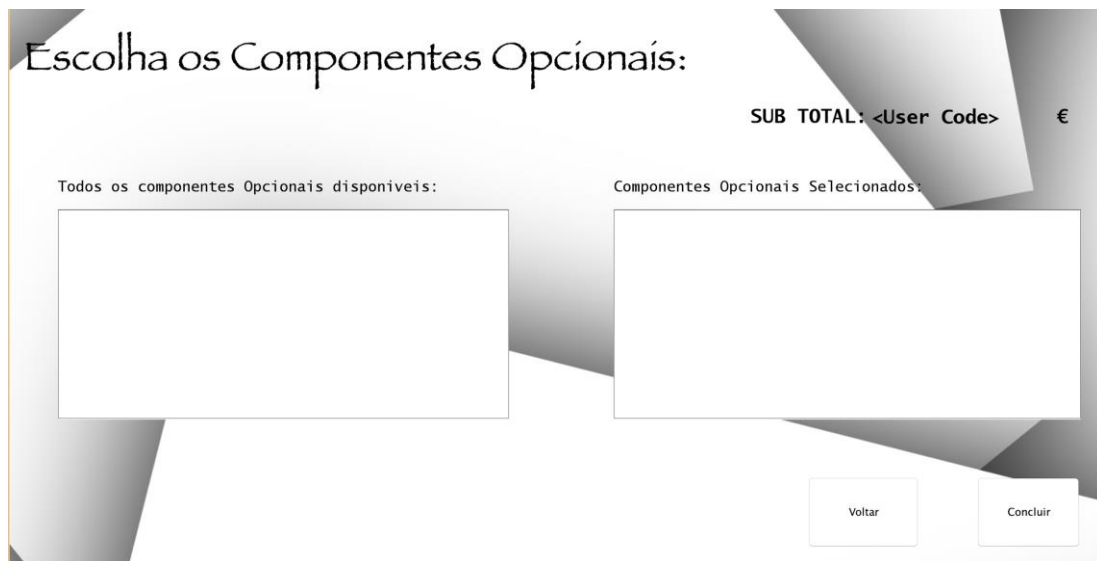
Informa o utilizador sobre o pacote ter sido inserido na configuração com sucesso.

Aviso de existência de pacote já escolhido



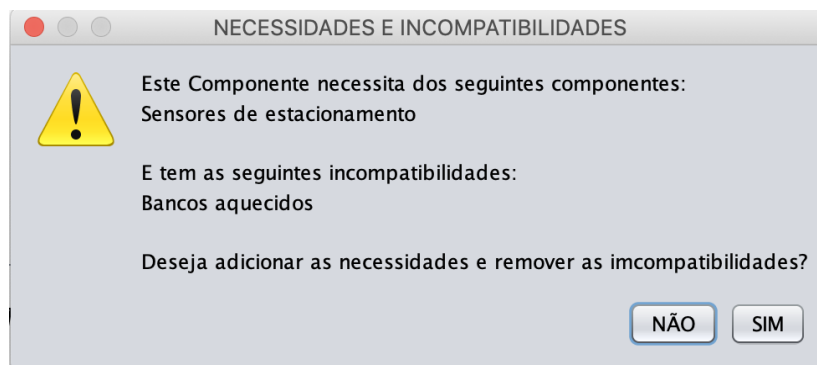
Interface que avisa o utilizador sobre a existência de um pacote já selecionado, caso o utilizador carregue em “SIM” os pacotes são trocados e caso clique em “NÃO” o pacote é mantido.

Configuração por passo a passo



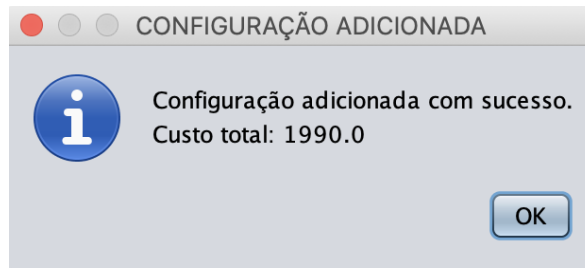
Interface que é mostrada ao utilizador quando quer fazer configuração passo a passo, a forma de inserir os componentes é a mesma de inserir os componentes essenciais, podendo aparecer a interface “Aviso componentes necessários e/ou incompatíveis” por falta de componentes ou por componentes incompatíveis. O utilizador pode voltar para a interface “Criar Configuração (Componentes Essenciais)” carregando no botão “Voltar” ou pode escolher terminar a configuração carregando no botão “Concluir”. Quando faz concluir, pode aparecer a interface “Aviso componentes iguais a pacote”. De seguida aparece a interface “Informação configuração adicionada com sucesso” ou “Erro fila de configuração cheia” caso a fila de configurações esteja cheia.

Aviso componentes necessários e/ou incompatíveis



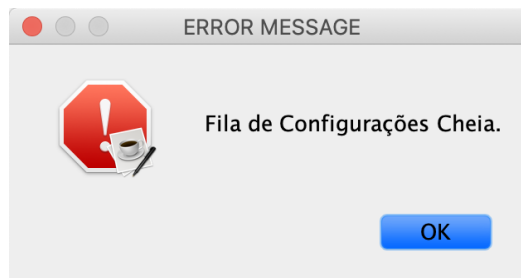
Interface que avisa o utilizador que existem componentes necessários e/ou incompatíveis deixando o utilizador escolher se quer introduzir os necessários e remover os incompatíveis para colocar o componente selecionado ou então deixar ficar os componentes já inseridos não introduzindo o selecionado nem os componentes necessários, carregando no botão “SIM” ou “NÃO”, respetivamente.

Informação configuração adicionada com sucesso



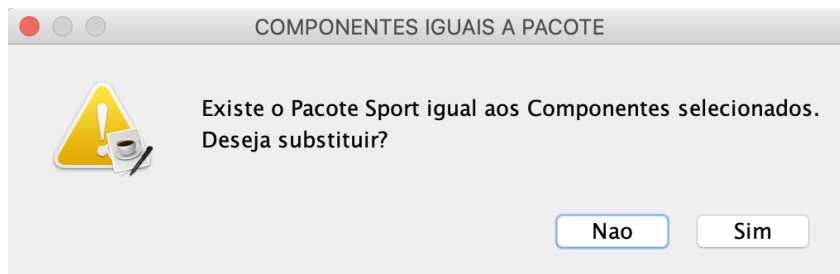
Informa que a configuração foi introduzida na fila e qual o custo total da configuração.

Erro fila de configurações cheia



Informa que a fila de configurações está cheia e não insere a configuração realizada.

Aviso componentes iguais a pacote



Avisa o utilizador sobre componentes iguais a um pacote existente e pergunta se quer substituir, clicando em “SIM” os componentes opcionais são removidos para ser colocado o pacote, em caso de “NÃO” os componentes são mantidos.

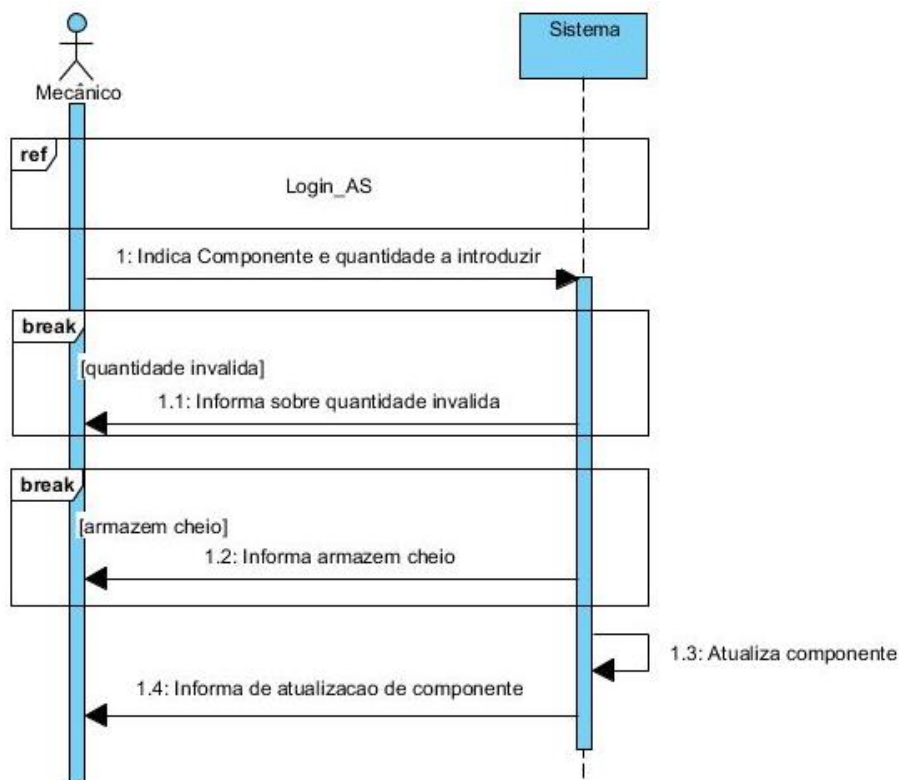
Diagramas de Sequência de Sistemas e com Subsistemas

Partindo da especificação textual dos Use Cases, apresenta-se de seguida, para cada um deles, as três fases da construção dos diagramas de sequência dos mesmos, como sendo:

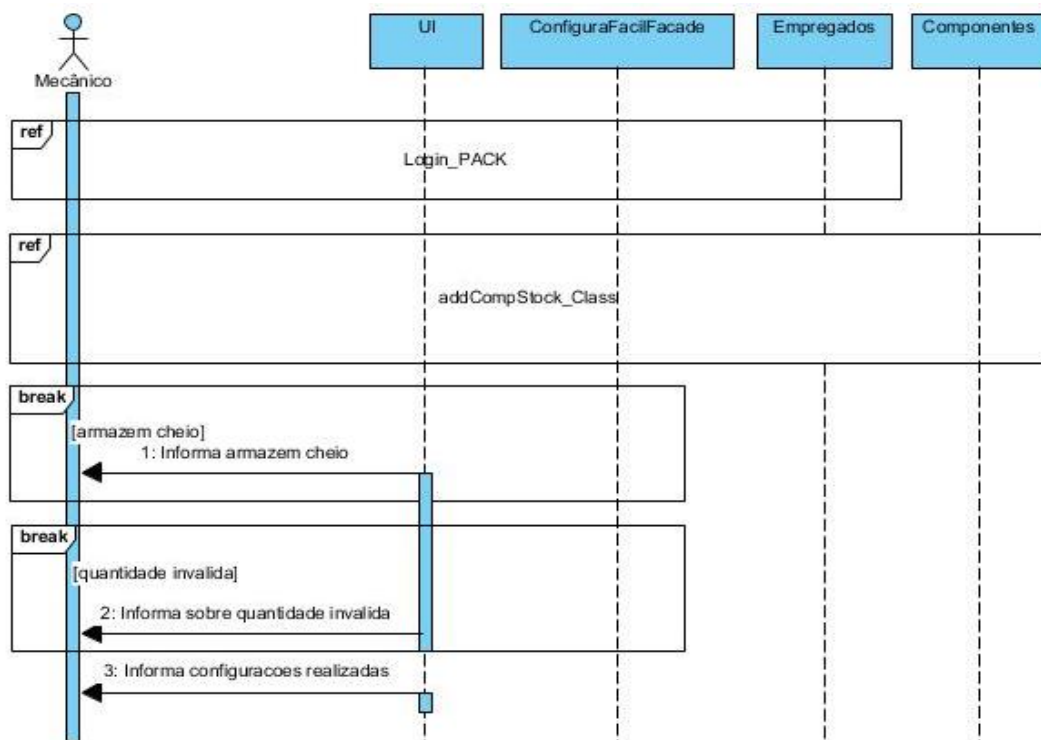
- **Ator-sistema:** o sistema ainda é algo abstrato e o ator, neste caso o empregado de loja/fábrica interage com o mesmo;
- **Ator-UserInterface-Facade-Packages:** aqui é mostrado como a informação introduzida pelo ator na interface é “tratada”, quando se está a considerar “packages” como forma de agrupar representações do sistema, visto que os “packages” contém subdivisões presentes no “facade”.

Atualizar Stock

Ator-Sistema

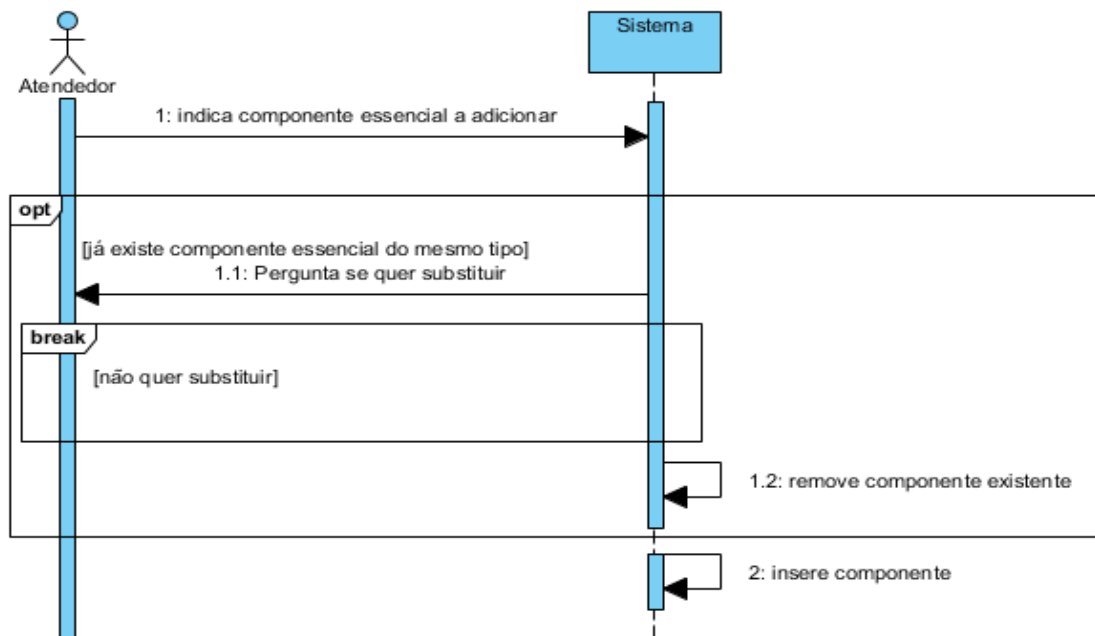


Ator-UserInterface-Facade-Packages

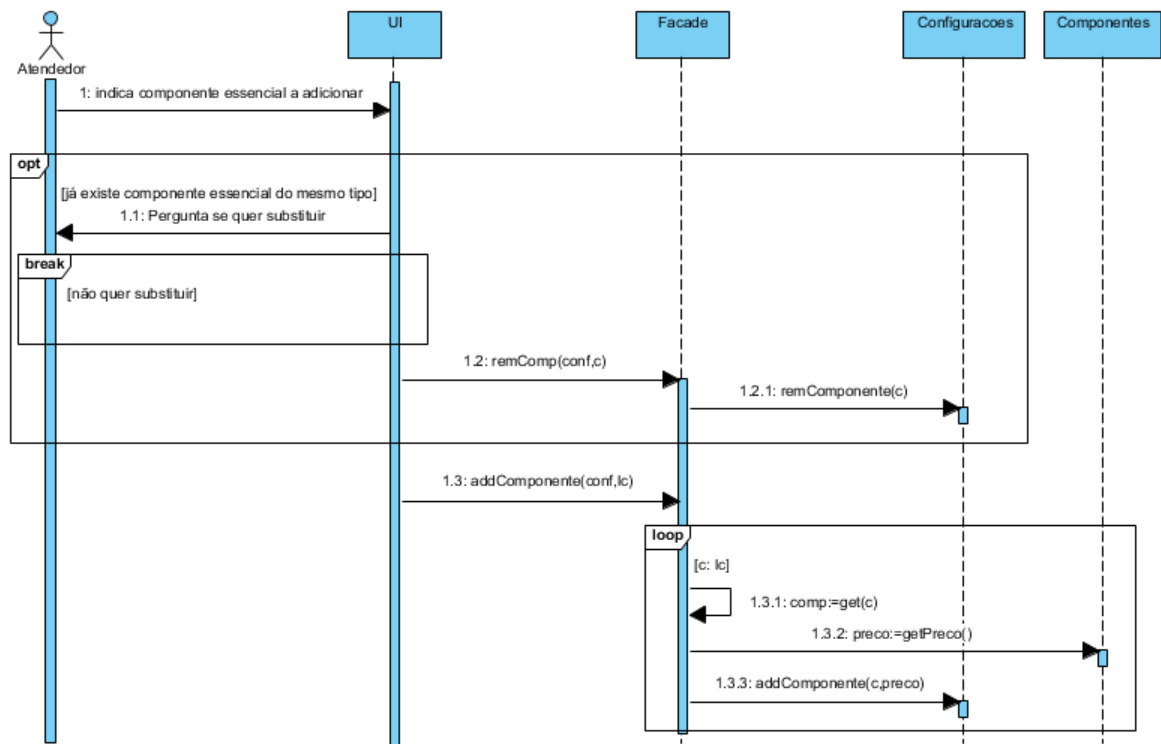


Adicionar Componente Essencial

Ator-Sistema

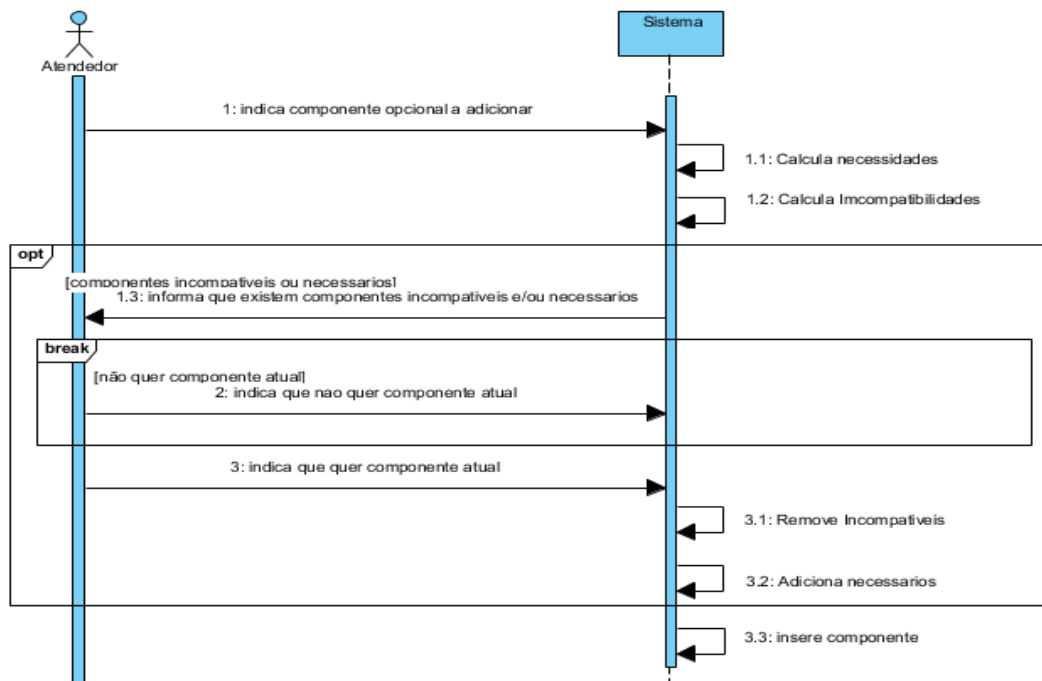


Ator-UserInterface-Facade-Packages

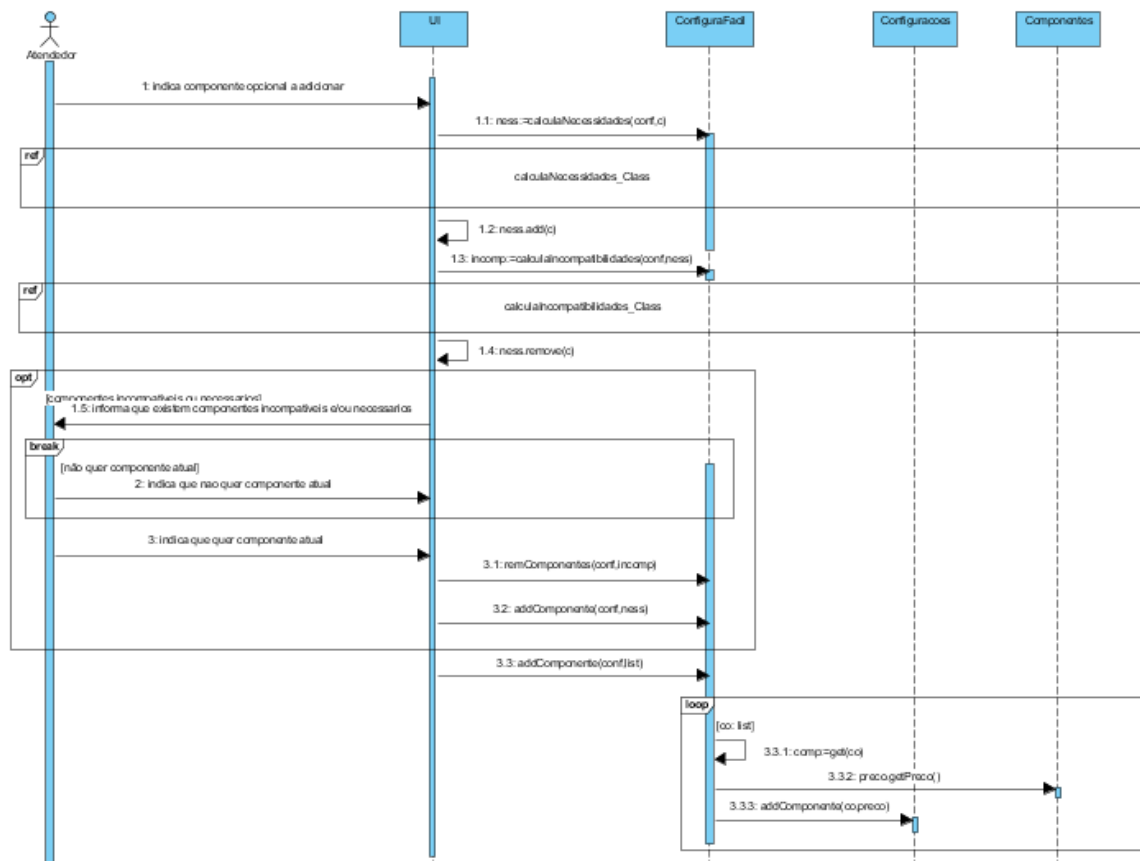


Adicionar Componente Opcional

Ator-Sistema

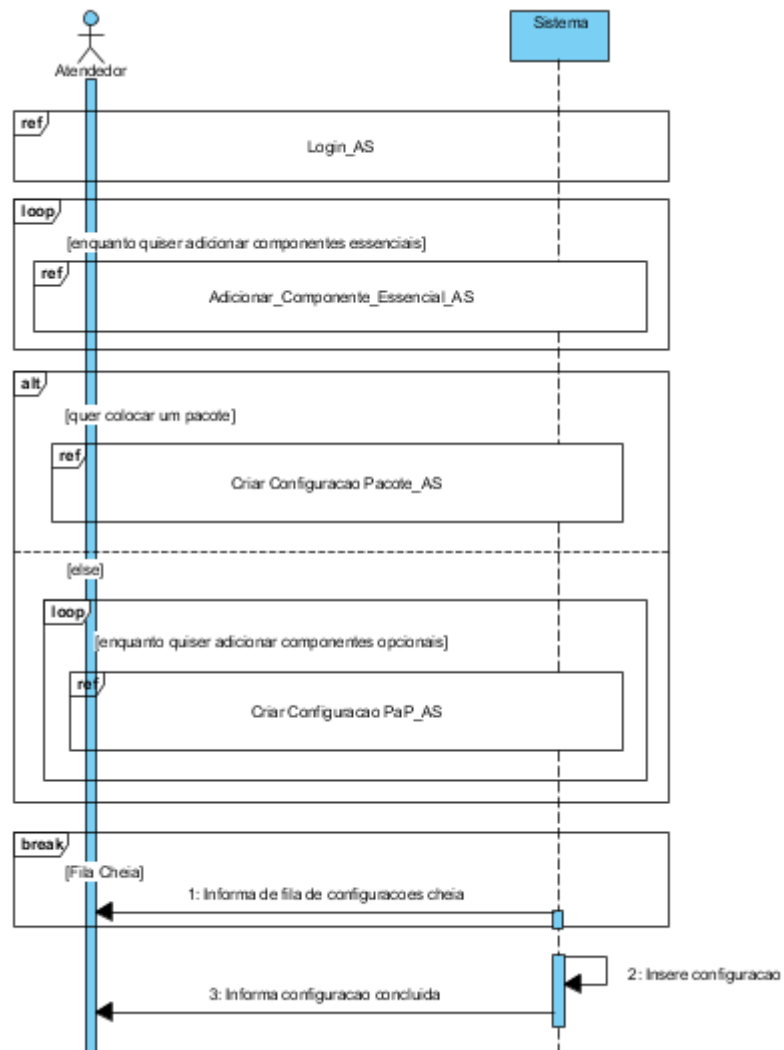


Ator-UserInterface-Facade-Packages

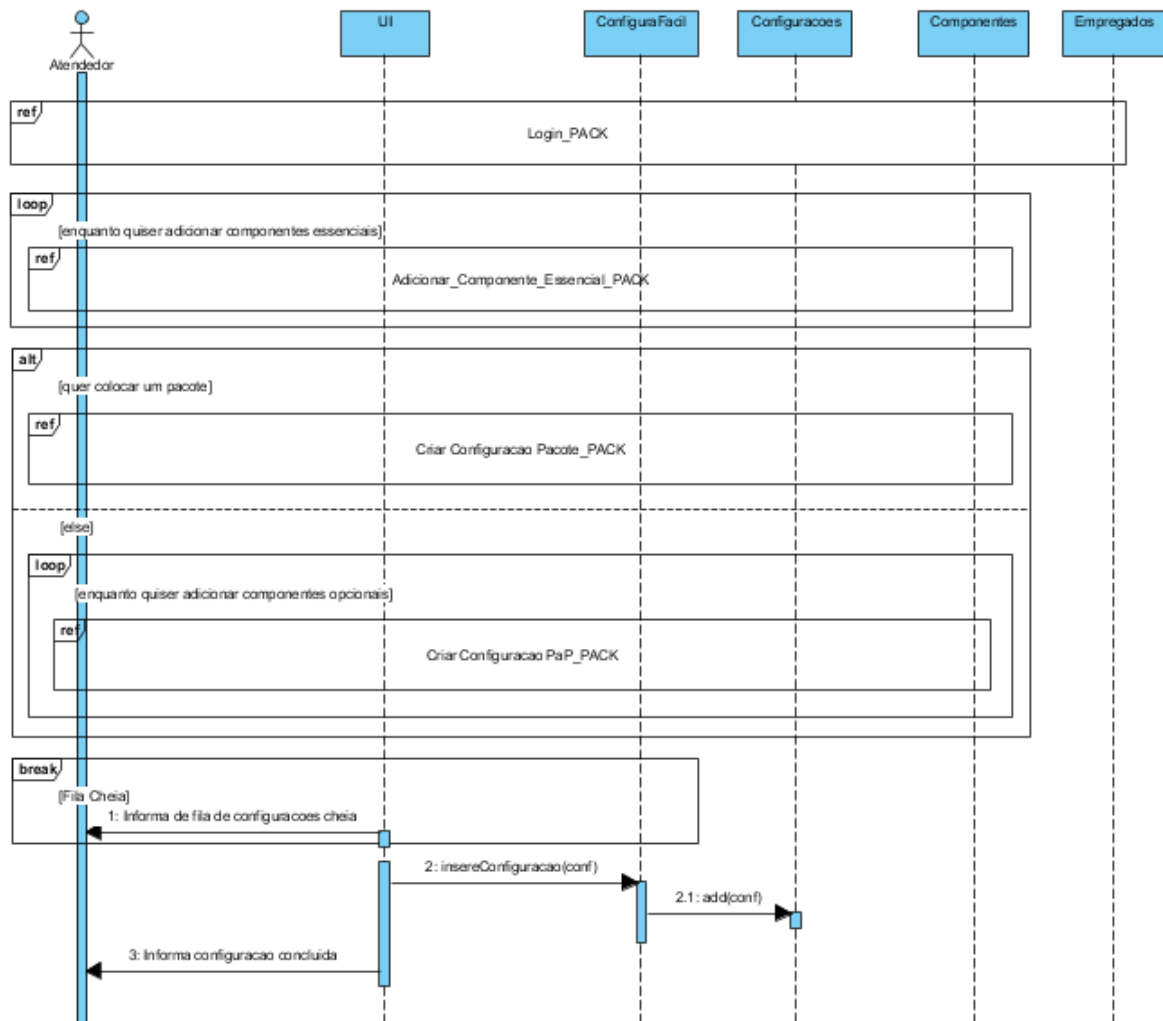


Criar Configuração

Ator-Sistema

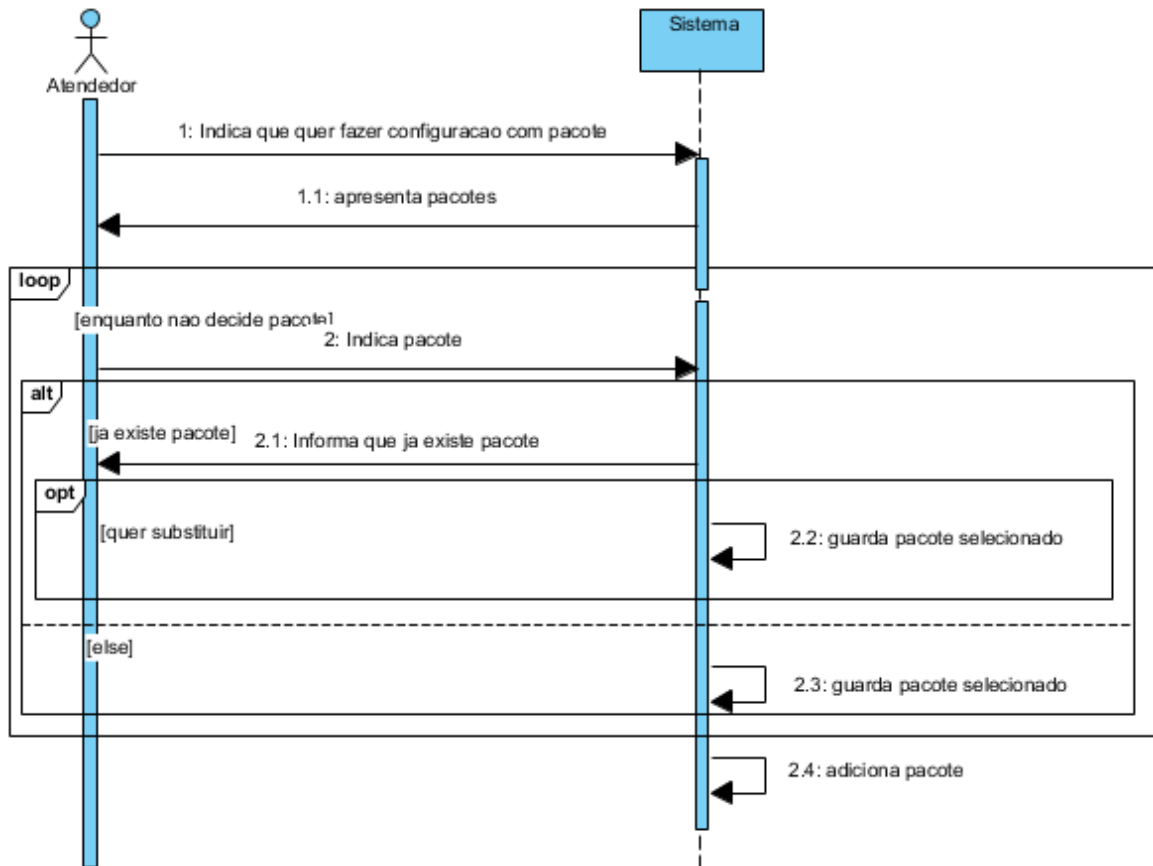


Ator-UserInterface-Facade-Packages

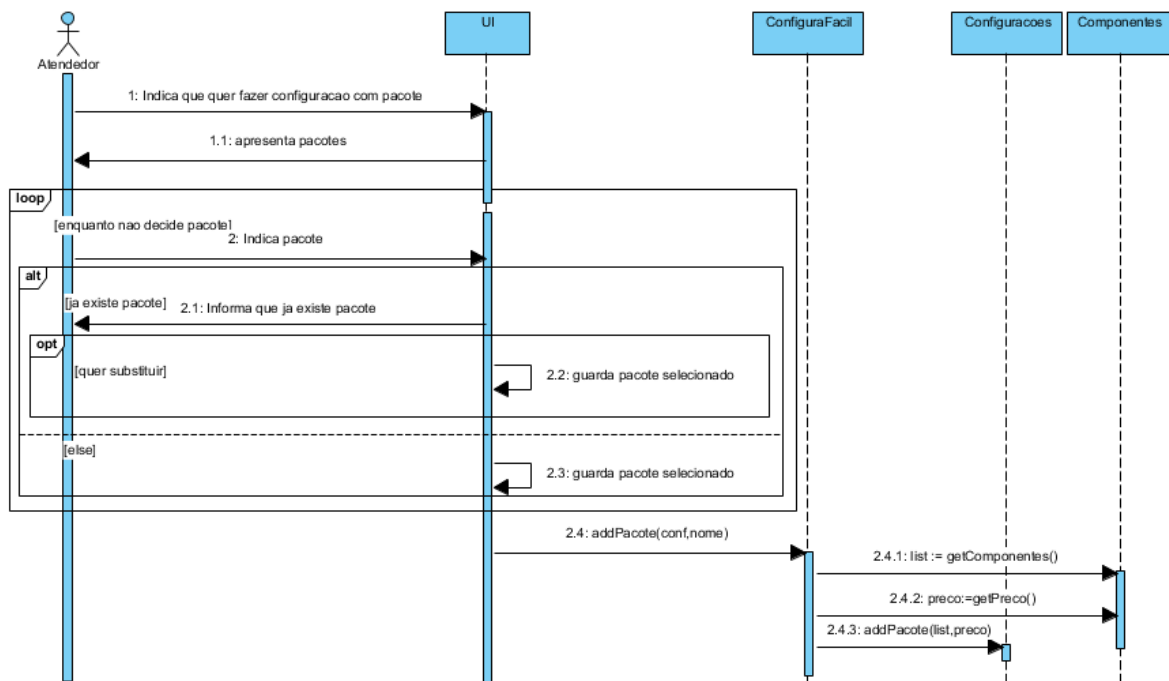


Pacote

Ator-Sistema

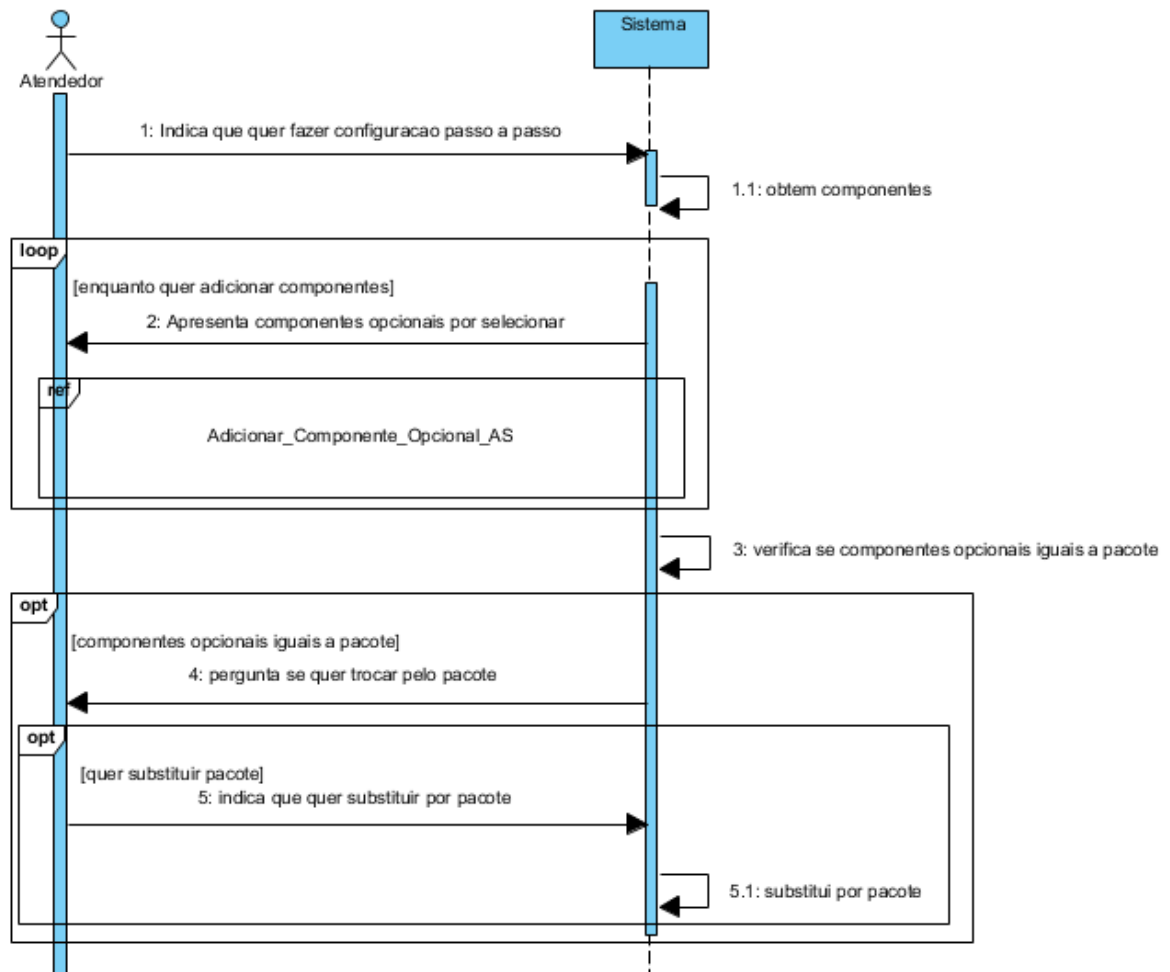


Ator-UserInterface-Facade-Packages



Passo a Passo

Ator-Sistema



Ator-UserInterface-Facade-Packages

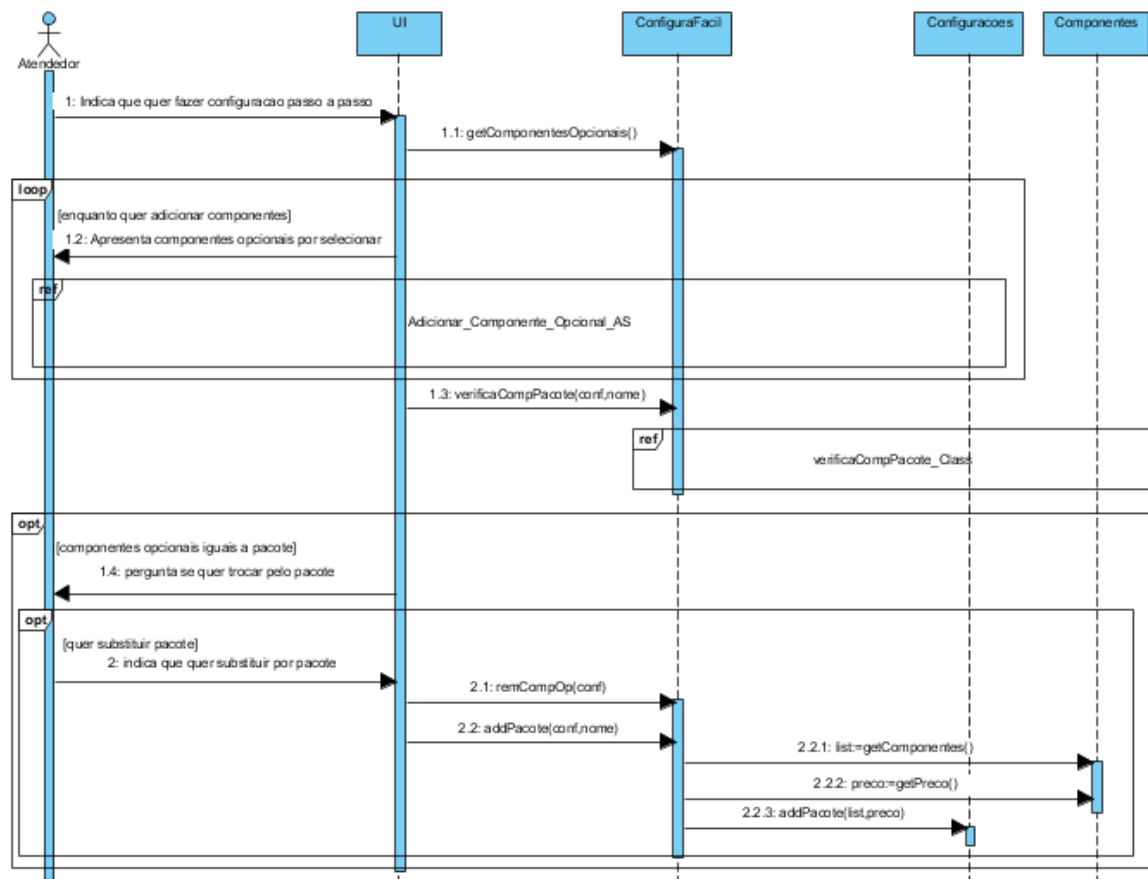
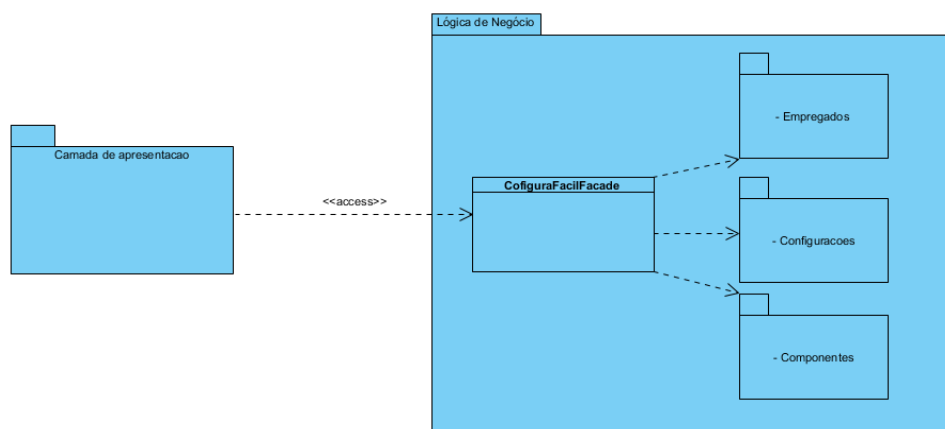


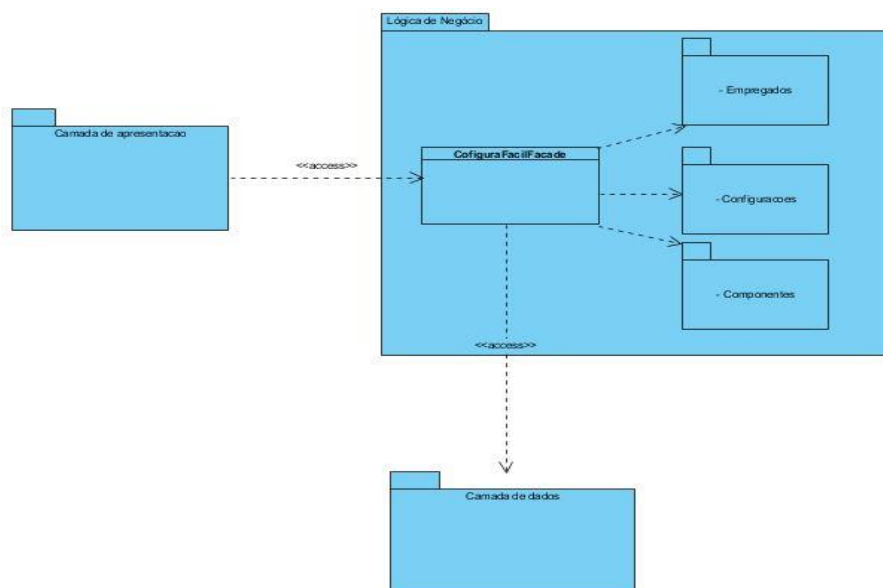
Diagrama de packages

Partindo de uma análise ao modelo de domínio, agrupamos as entidades em três packages principais:

- **“Empregados”** – no modelo de domínio desenvolvido, o atendedor e o mecânico são aqueles que fazem a gestão das configurações e a gestão da quantidade de componentes existentes, respetivamente, ou seja, é necessária a criação de um “package” que junte as duas entidades;
- **“Configuracoes”** – agrupamentos das partes mais básicas do sistema geral, ou seja, das várias componentes: as configurações podem ser do tipo “passo a passo” ou do tipo “escolha de pacote”;
- **“Componentes”** – unidades mais básicas do sistema e que compõem as mais variadas configurações: podem ser componentes essenciais ou opcionais.



Para testar a viabilidade do trabalho desenvolvido, foi necessário aplicar as nossas estruturas a uma camada de dados. Assim, o modelo de packages evoluiu, ficando com a forma final que se apresenta de seguida.



A camada de apresentação (User Interface) tem acesso ao facade do sistema. Este, por sua vez, tem dependências sobre os packages “Empregados”, “Configuracoes” e

“Componentes” e acede à camada de dados, com o intuito de garantir a persistência dos dados.

Diagrama de Classes

A seguir apresenta-se o diagrama de classes desenvolvido pelo grupo, com atributos e métodos definidos.

A classe principal é o *facade*, denominada por “ConfiguraFacil”, que contém os maps associados a configurações, pacotes, componentes e empregados.

A configuração tem identificadores de componentes (List<String>), que podem ser divididas em “ComponenteEssencial” e “ComponenteOpcional”. Assim, a classe Componente é abstrata.

O pacote tem como atributos a descrição, o preço e uma lista de componentes (apenas opcionais), sendo feita a distinção entre os seus vários tipos através do primeiro atributo referido anteriormente.

O que difere de “ComponenteOpcional” para “ComponenteEssencial” são duas listas que o “ComponenteOpcional” tem que o “ComponenteEssencial” não. Essas duas listas são: incompatibilidades (List<String>) e necessidades (List<String>).

O empregado tem como atributos o username e a password.

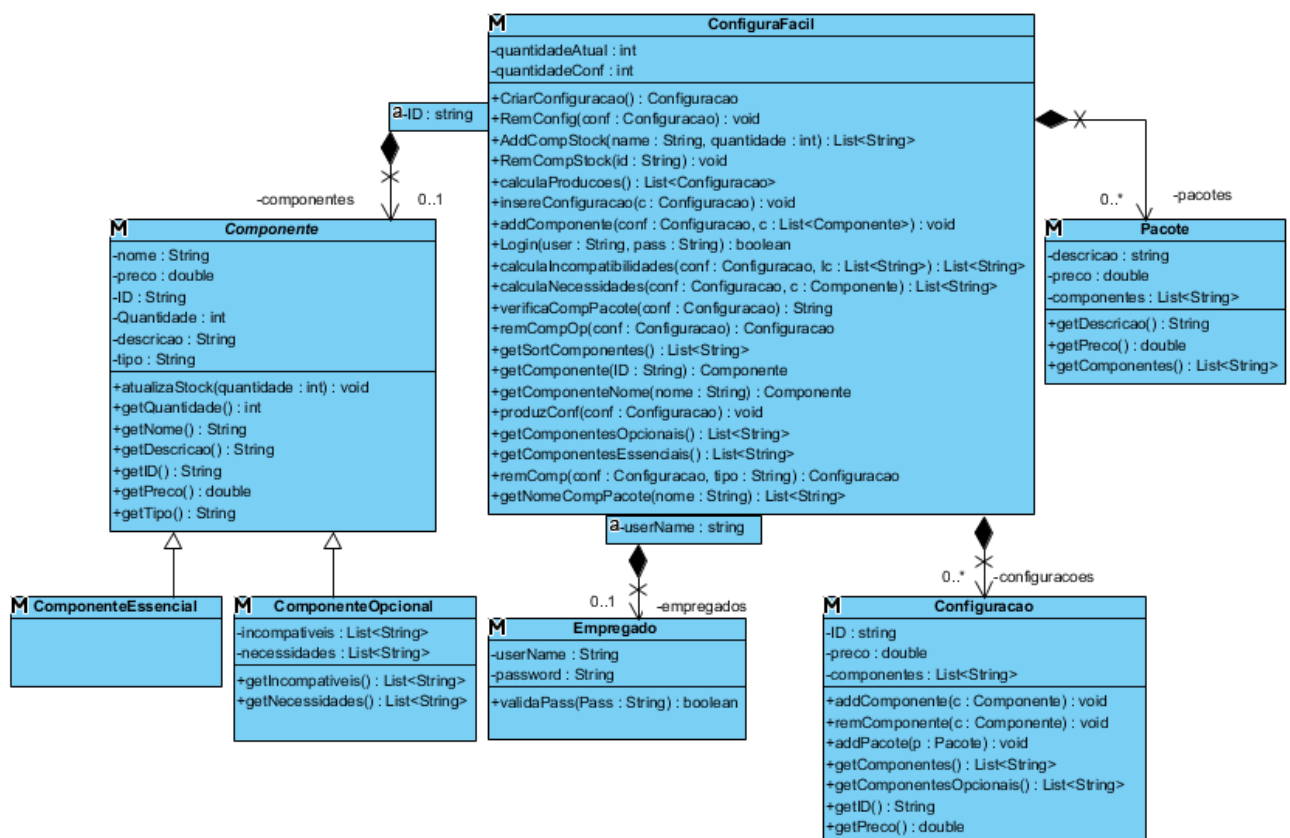
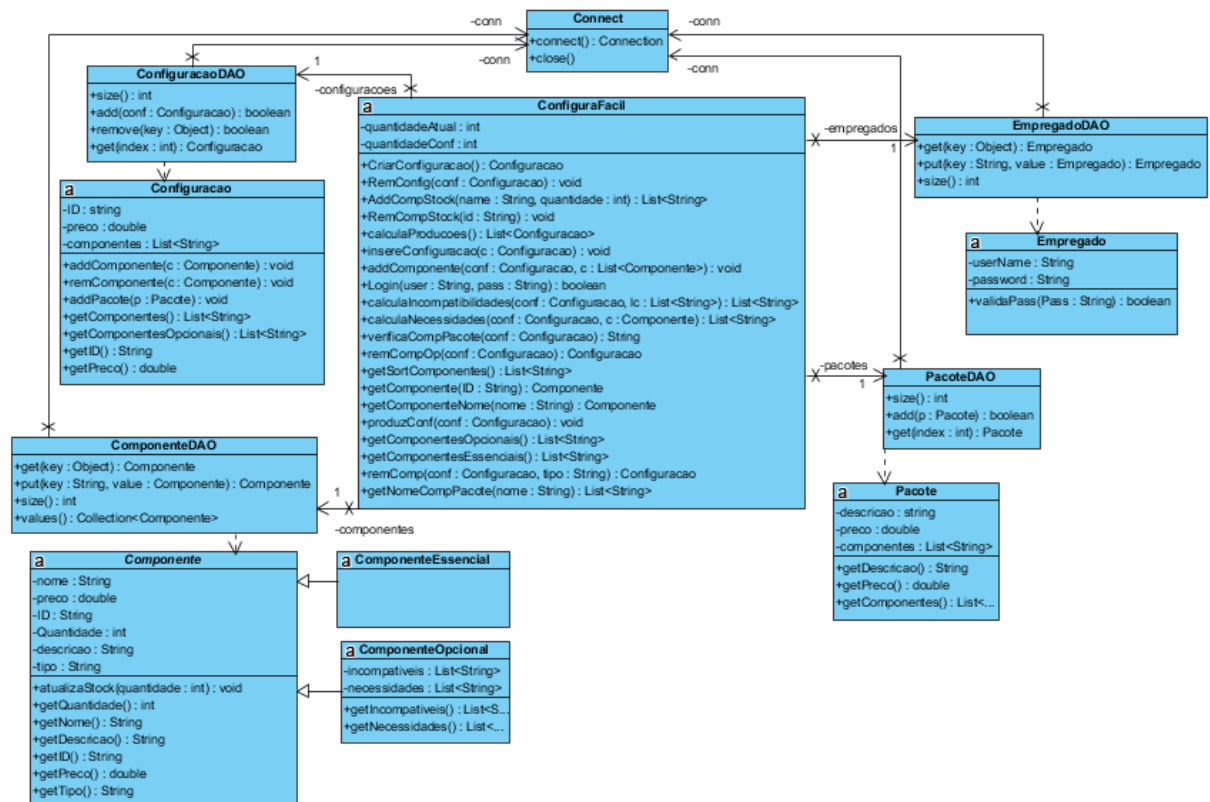


Diagrama de classes/DAO

Em relação ao diagrama anterior, as mudanças efetuadas relacionam-se com o facto de cada um dos maps associados a configurações, pacotes, componentes e empregados passarem a constituir, cada um deles, uma entidade DAO. Esta, por sua vez, tem uma relação de dependência com a entidade que representa cada um dos objetos do map inicial, por exemplo, “ComponenteDAO” depende de “Componente”, “ConfiguracaoDAO” depende de “Configuracao” ... Existe uma entidade extra, designada por Connect, que é responsável por fazer a ligação à camada de dados.

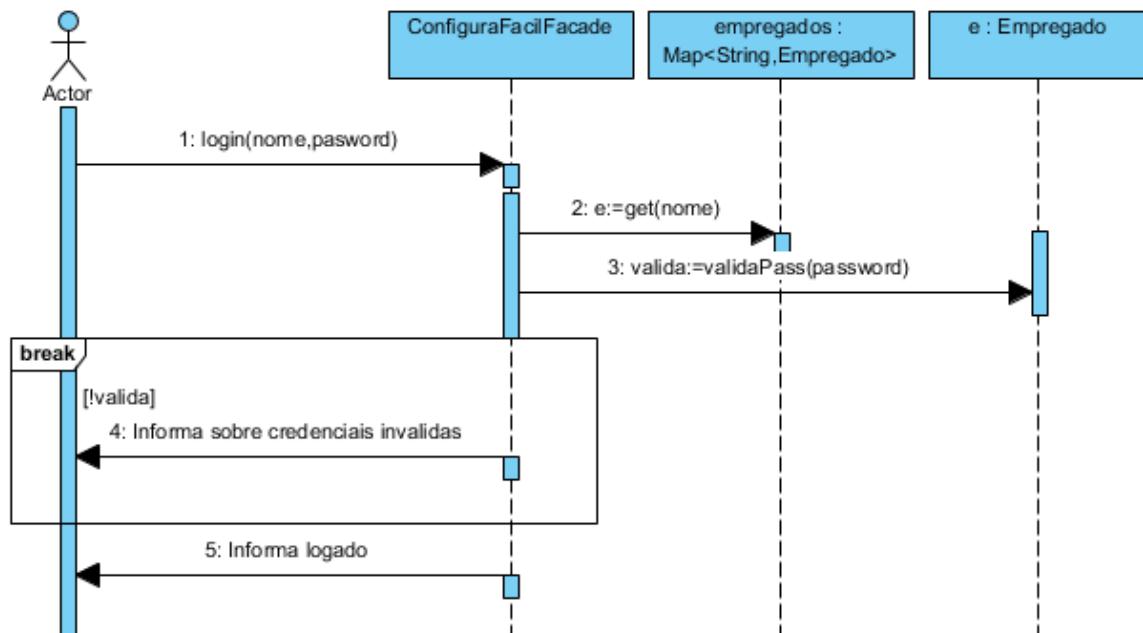


Diagramas de implementação

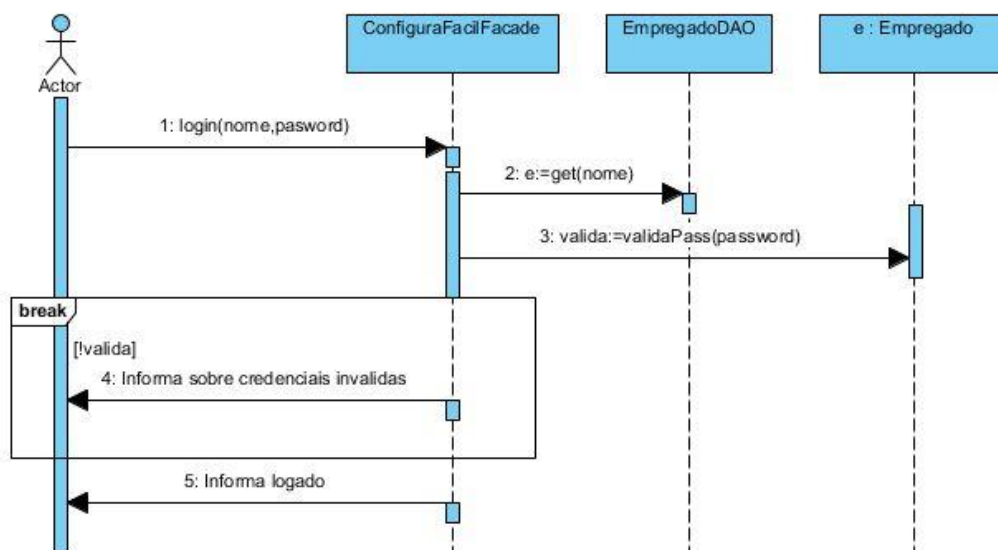
Passado a parte a modelação do projeto, são apresentados de seguida os diagramas de implementação, tanto para Use Cases como para métodos que o grupo considerou relevantes para o funcionamento da aplicação. São considerados dois tipos de diagramas: com classes presentes no diagrama de classes e com os DAO's que representam a camada de dados.

Login

Ator-Facade-Classe

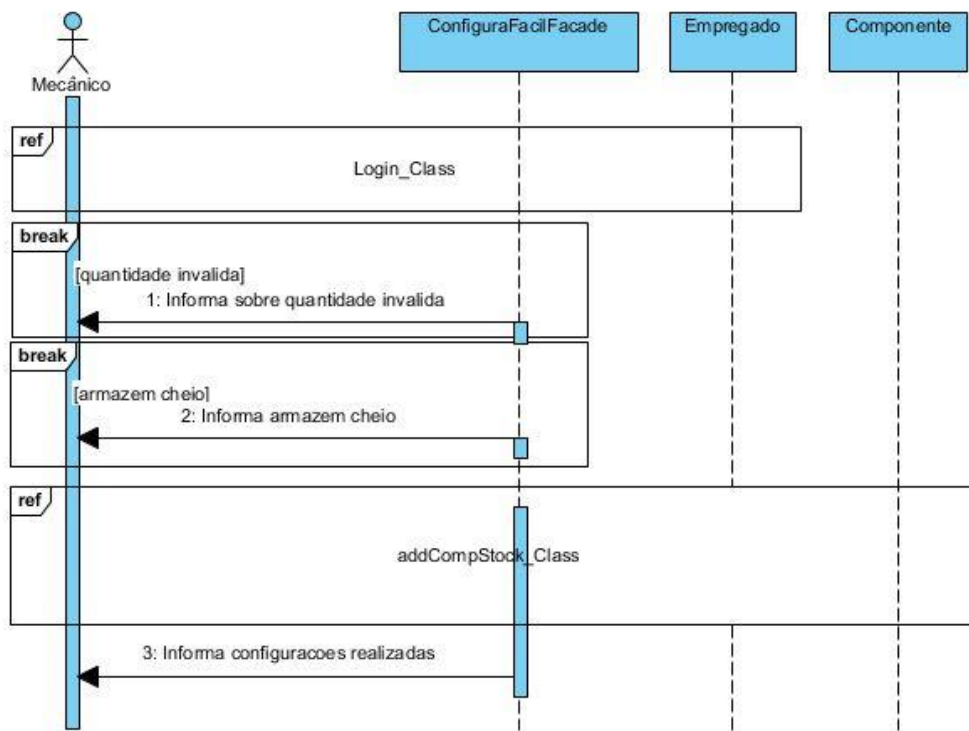


Ator-Facade-DAO

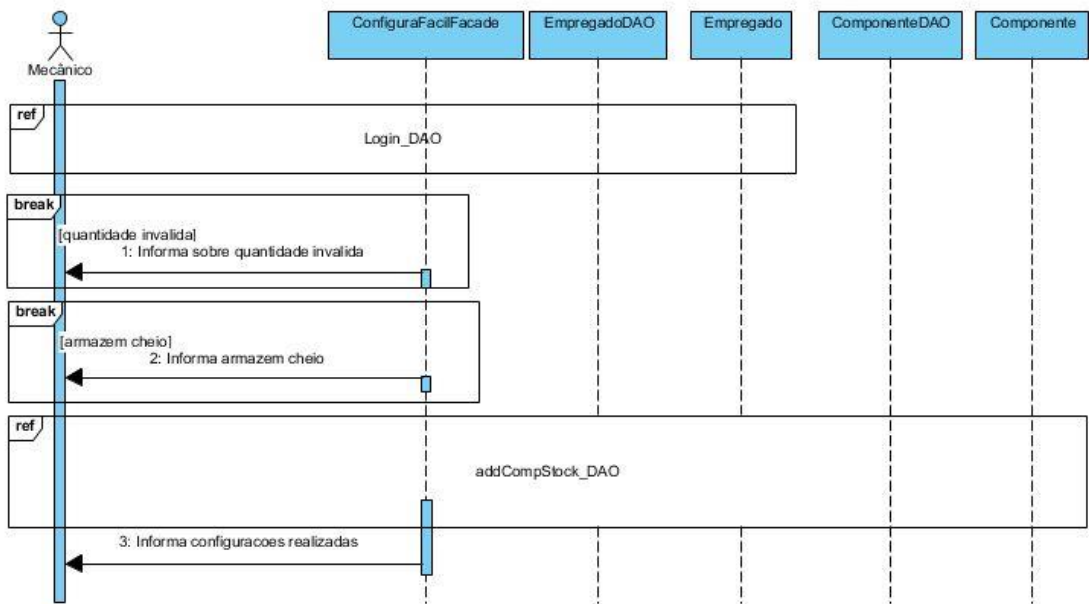


AtualizarStock

Ator-Facade-Classe

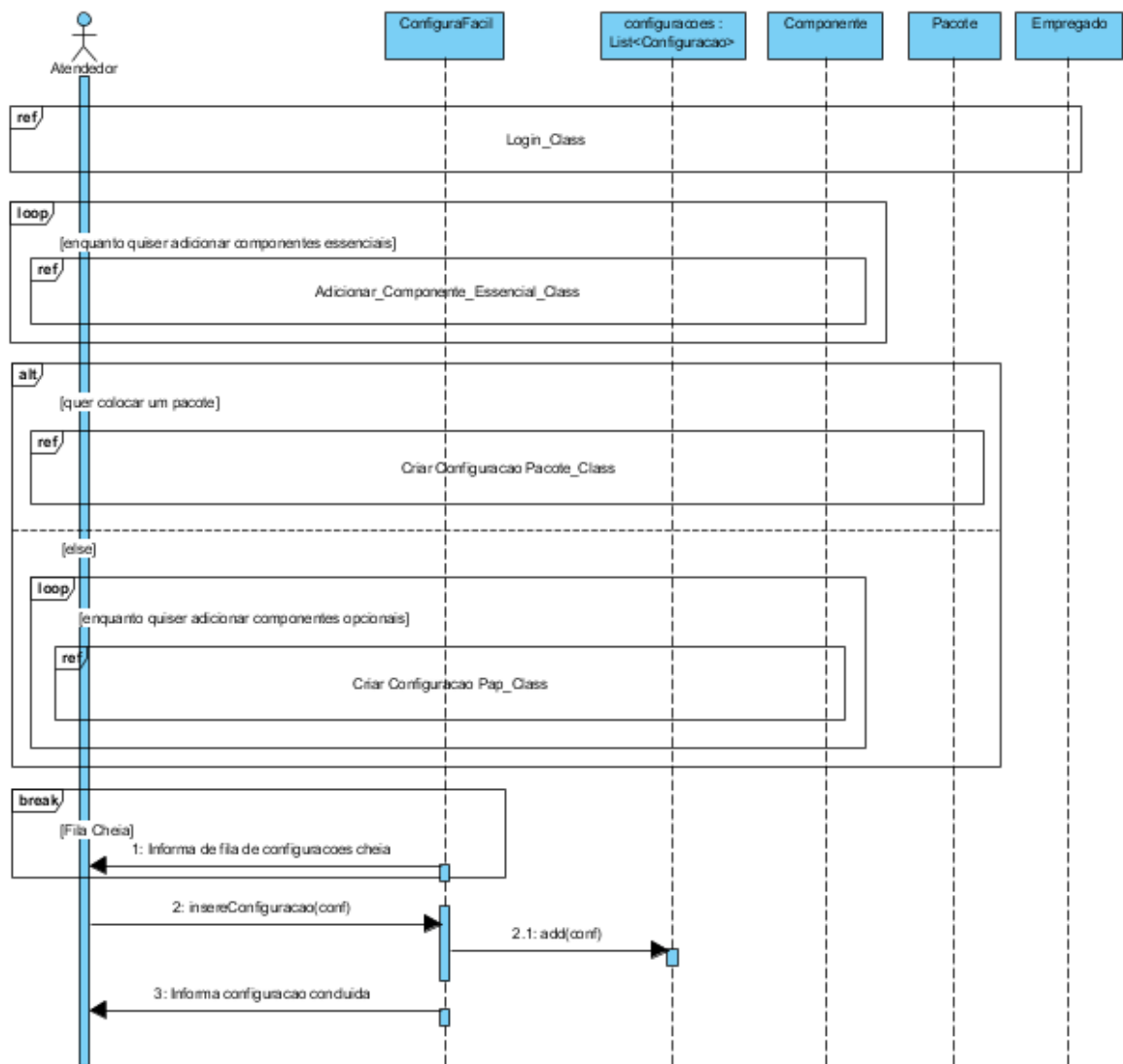


Ator-Facade-DAO

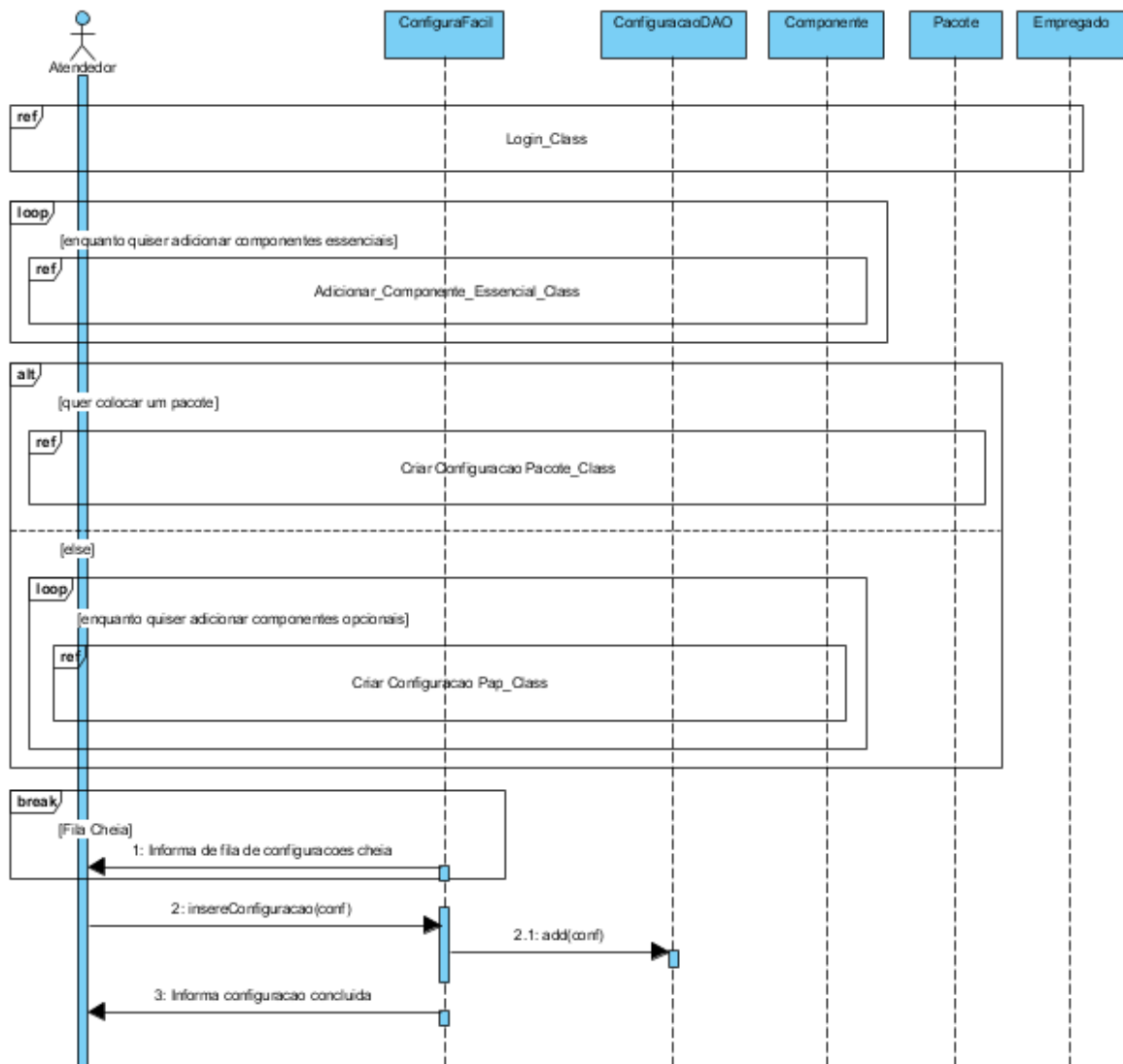


Criar Configuracao

Ator-Facade-Classe

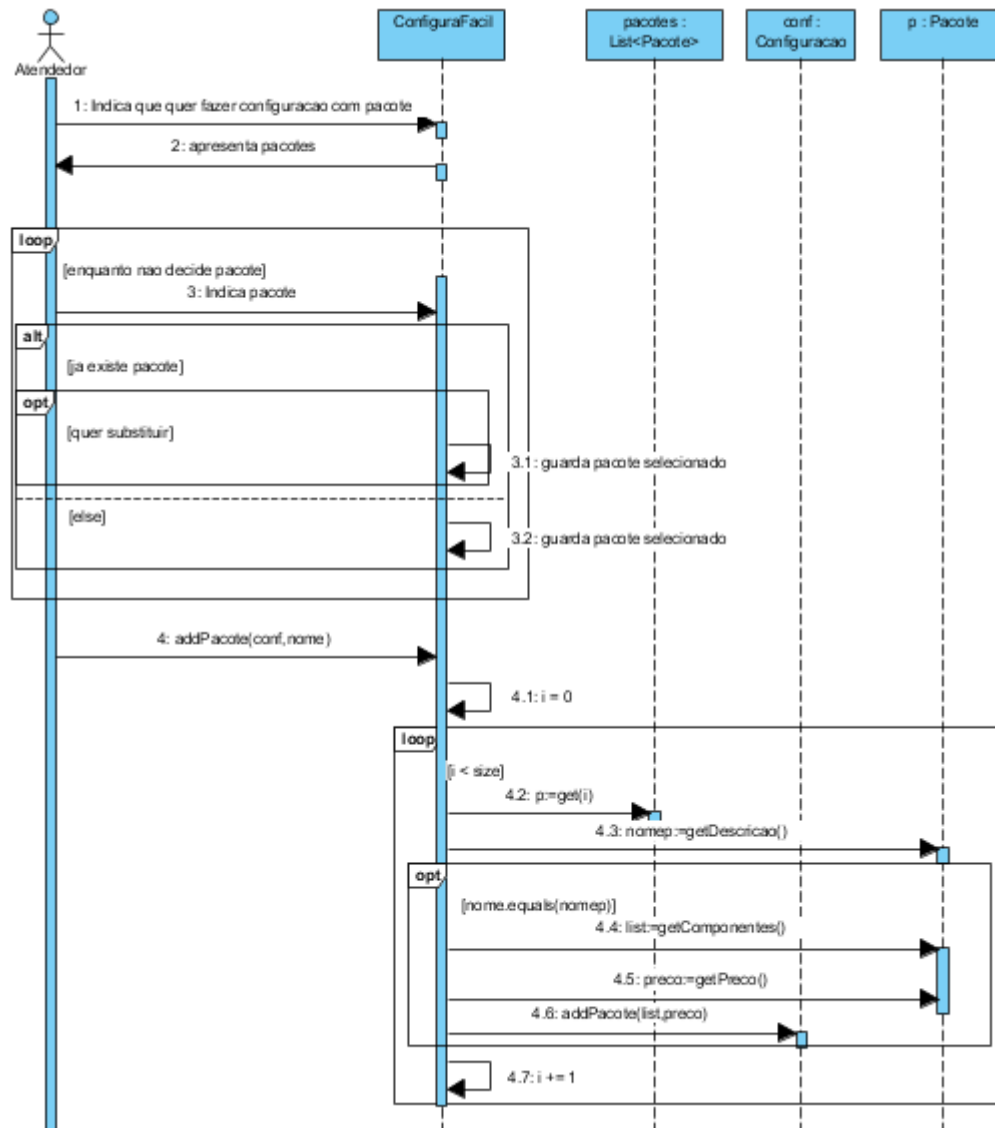


Ator-Facade-DAO

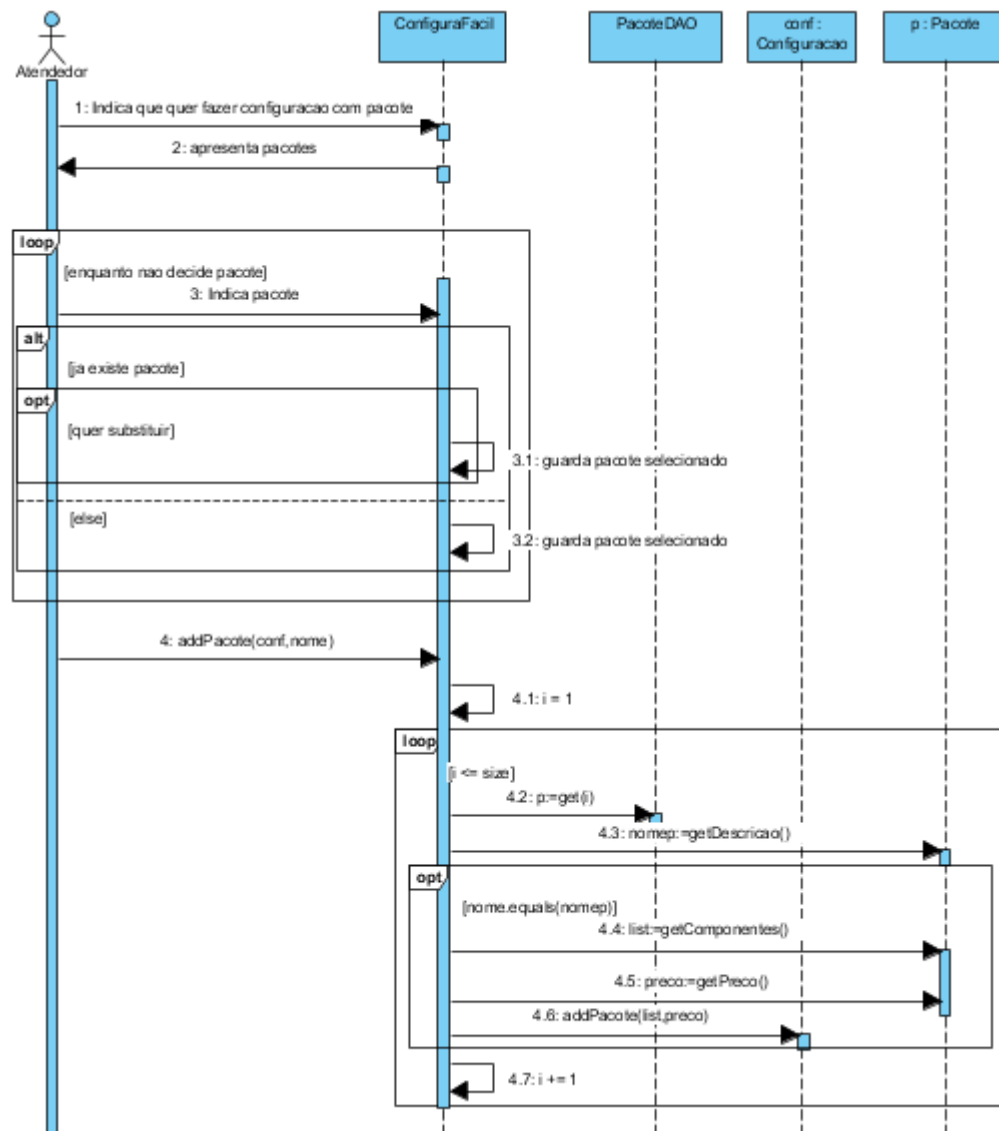


Pacote

Ator-Facade-Classe

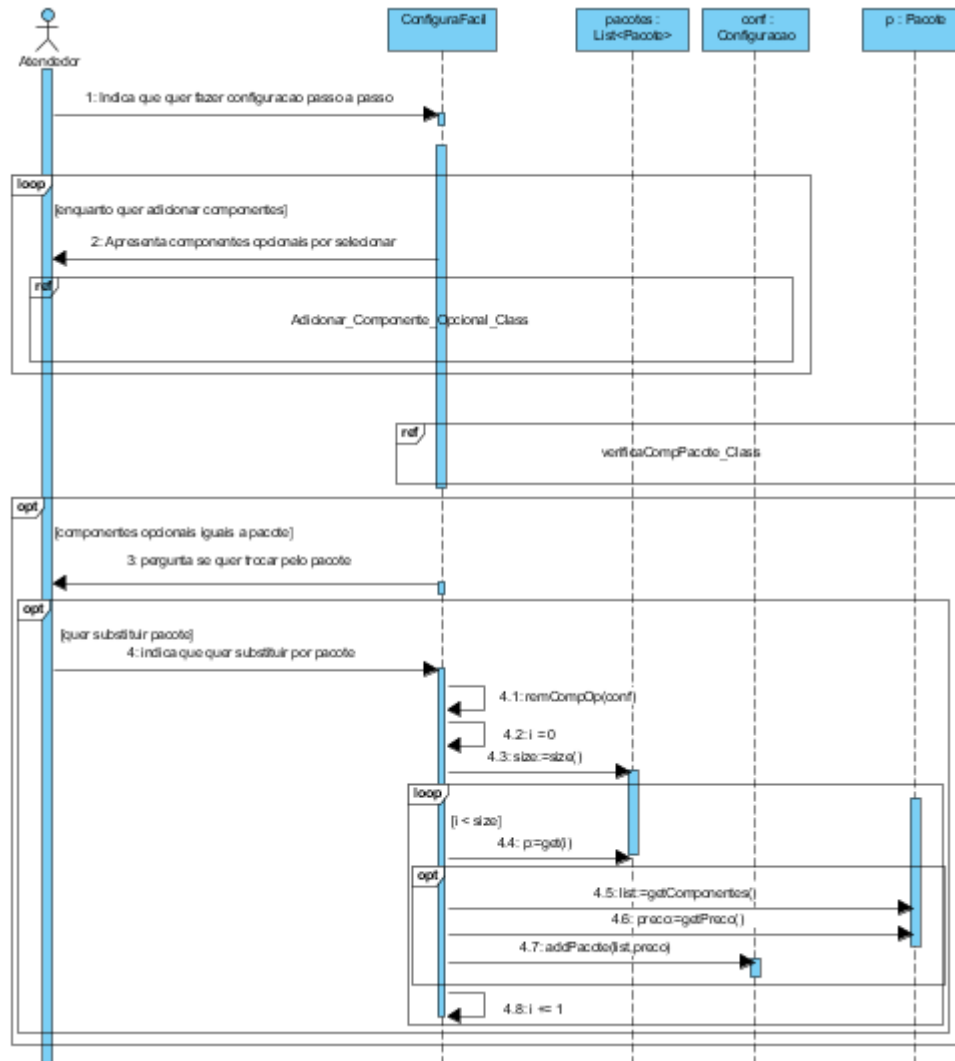


Ator-Facade-DAO

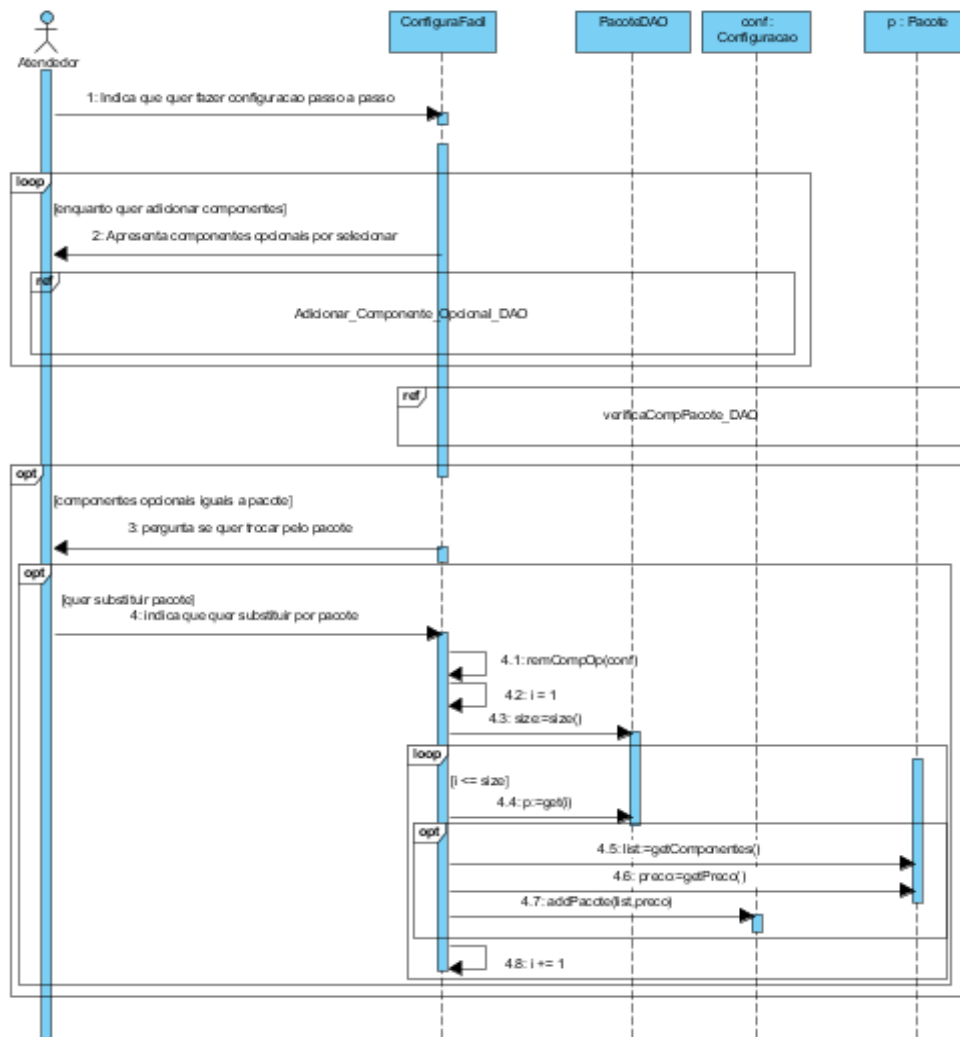


Passo a Passo

Ator-Facade-Classe

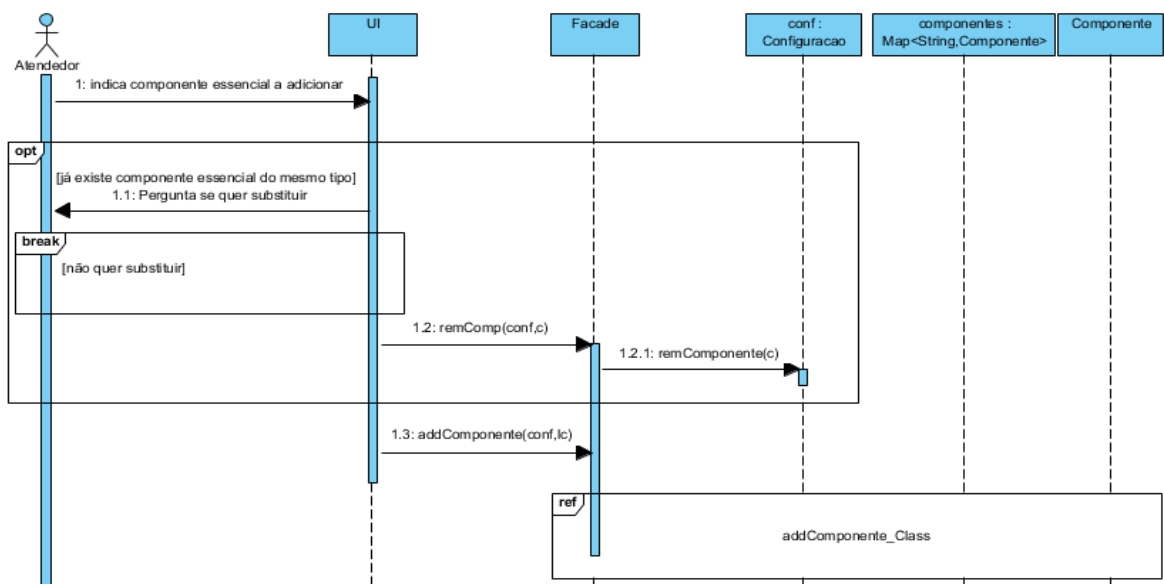


Ator-Facade-DAO

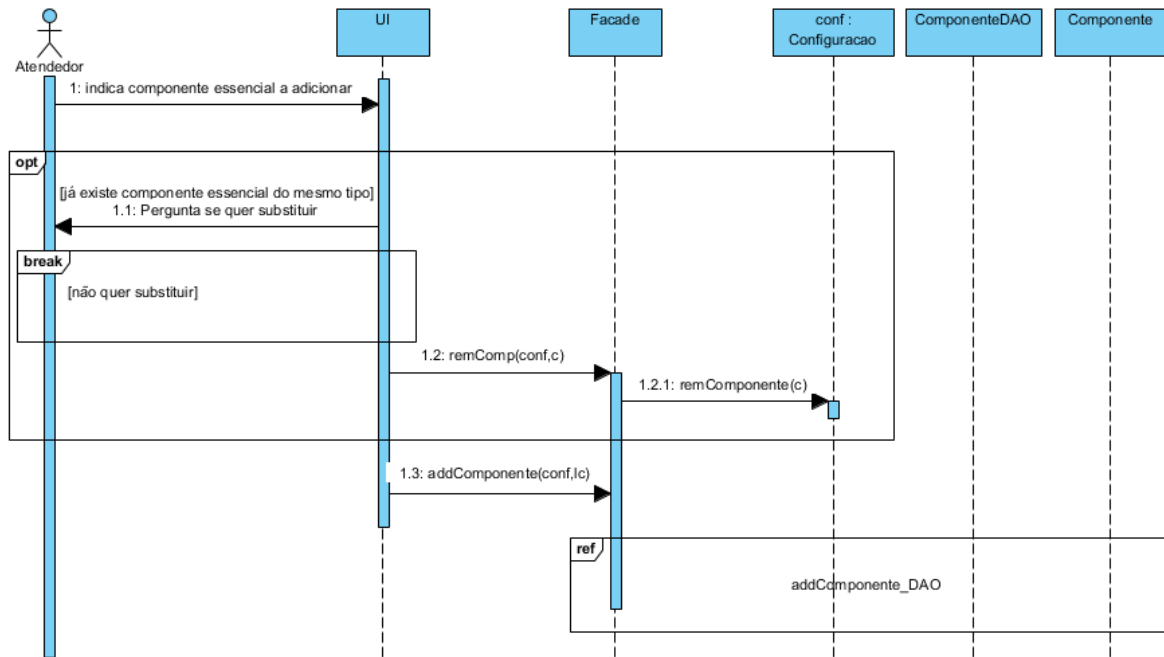


Adicionar Componente Essencial

Ator-Facade-Classe

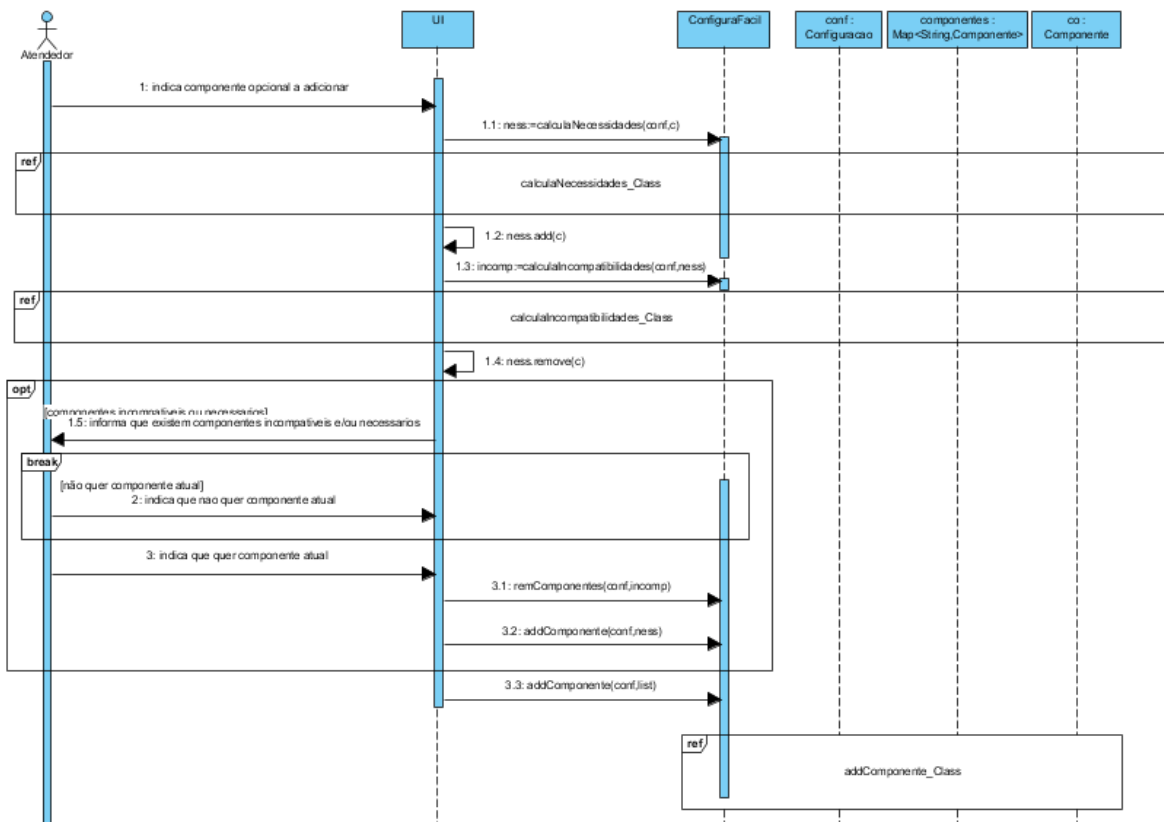


Ator-Facade-DAO

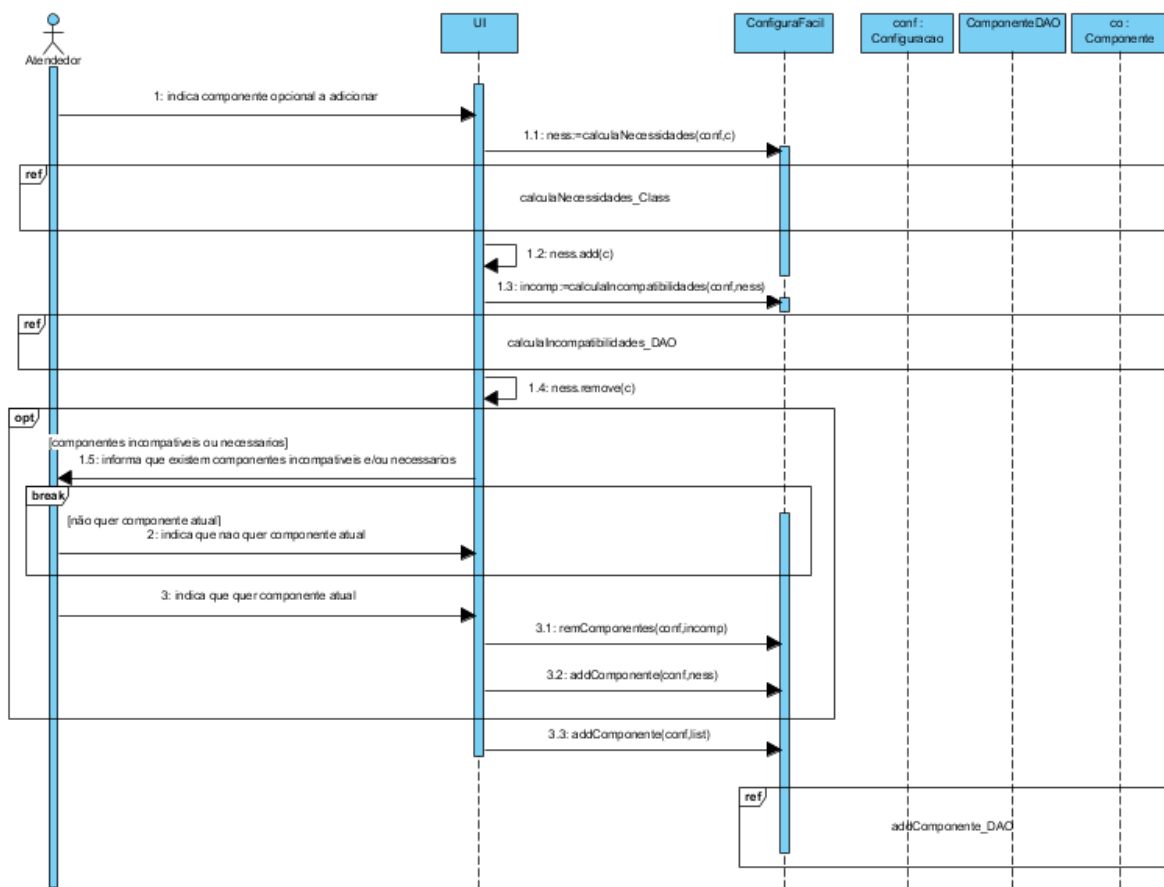


Adicionar Componente Opcional

Ator-Facade-Classe



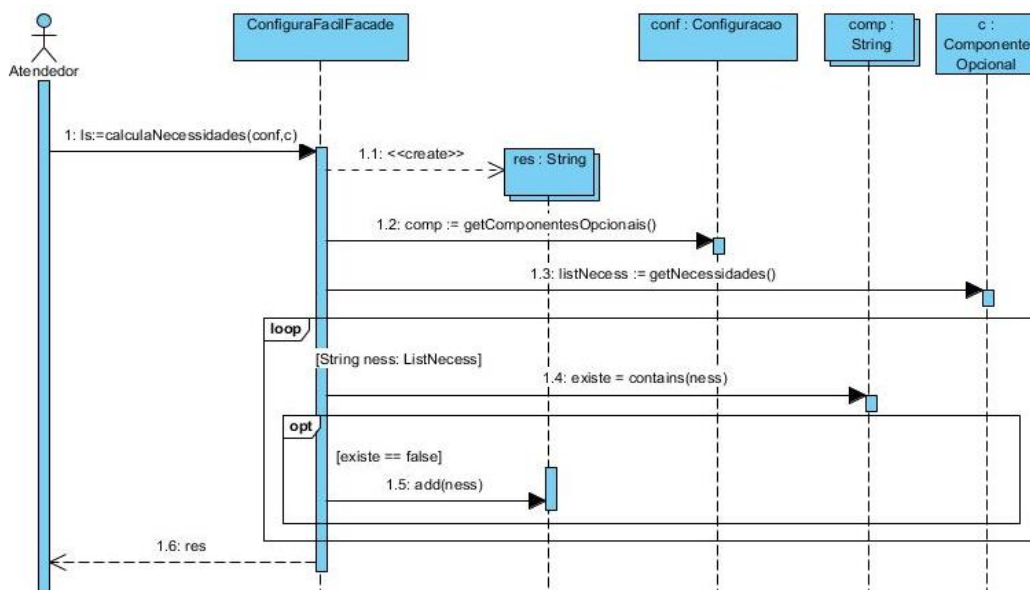
Ator-Facade-DAO



calculaNecessidades

Método utilizado na inserção de um elemento opcional numa configuração, informando o ator que outras componentes opcionais terão de estar presentes na configuração juntamente com o elemento que se deseja introduzir.

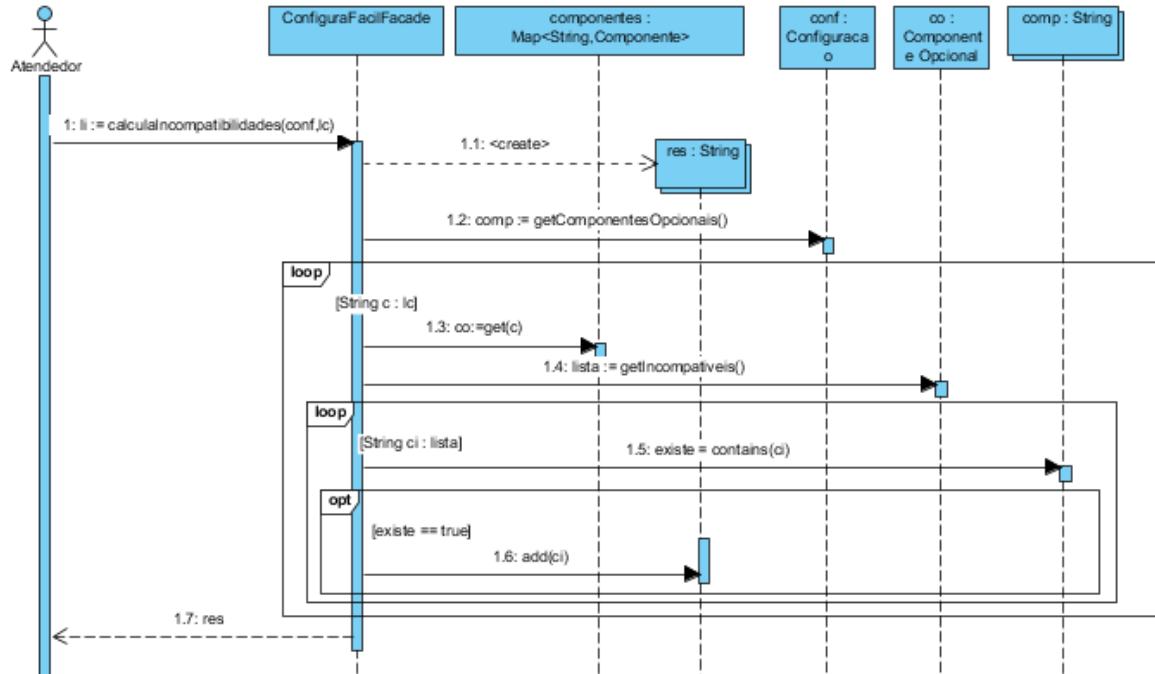
Ator-Facade-Classe



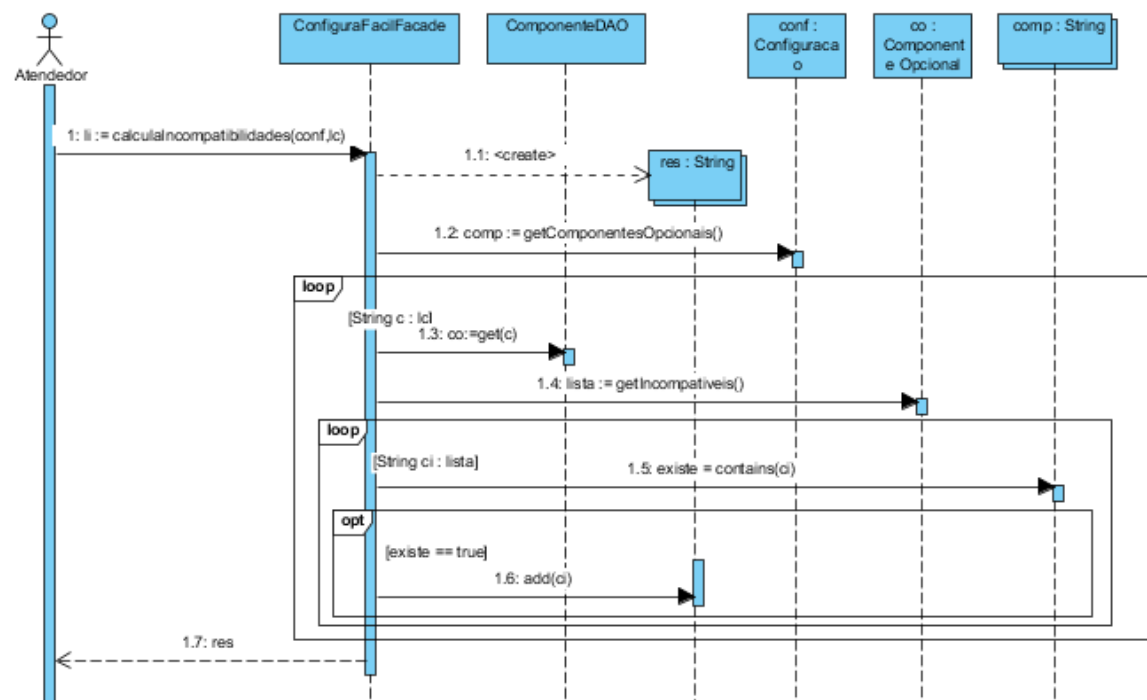
calculaIncompatibilidades

Método usado para indicar ao ator as incompatibilidades de componentes opcionais na configuração em relação ao elemento que se pretende introduzir.

Ator-Facade-Classe



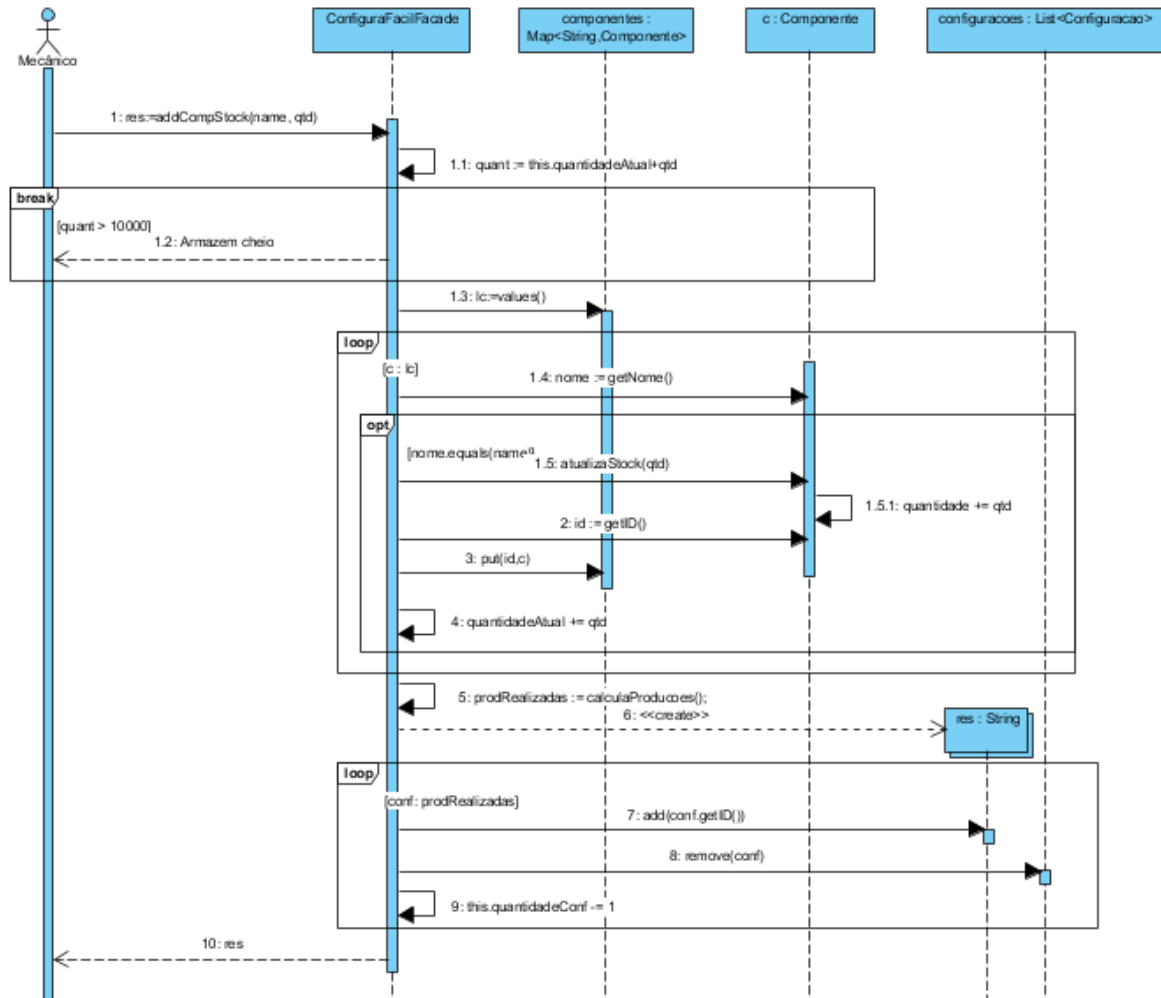
Ator-Facade-DAO



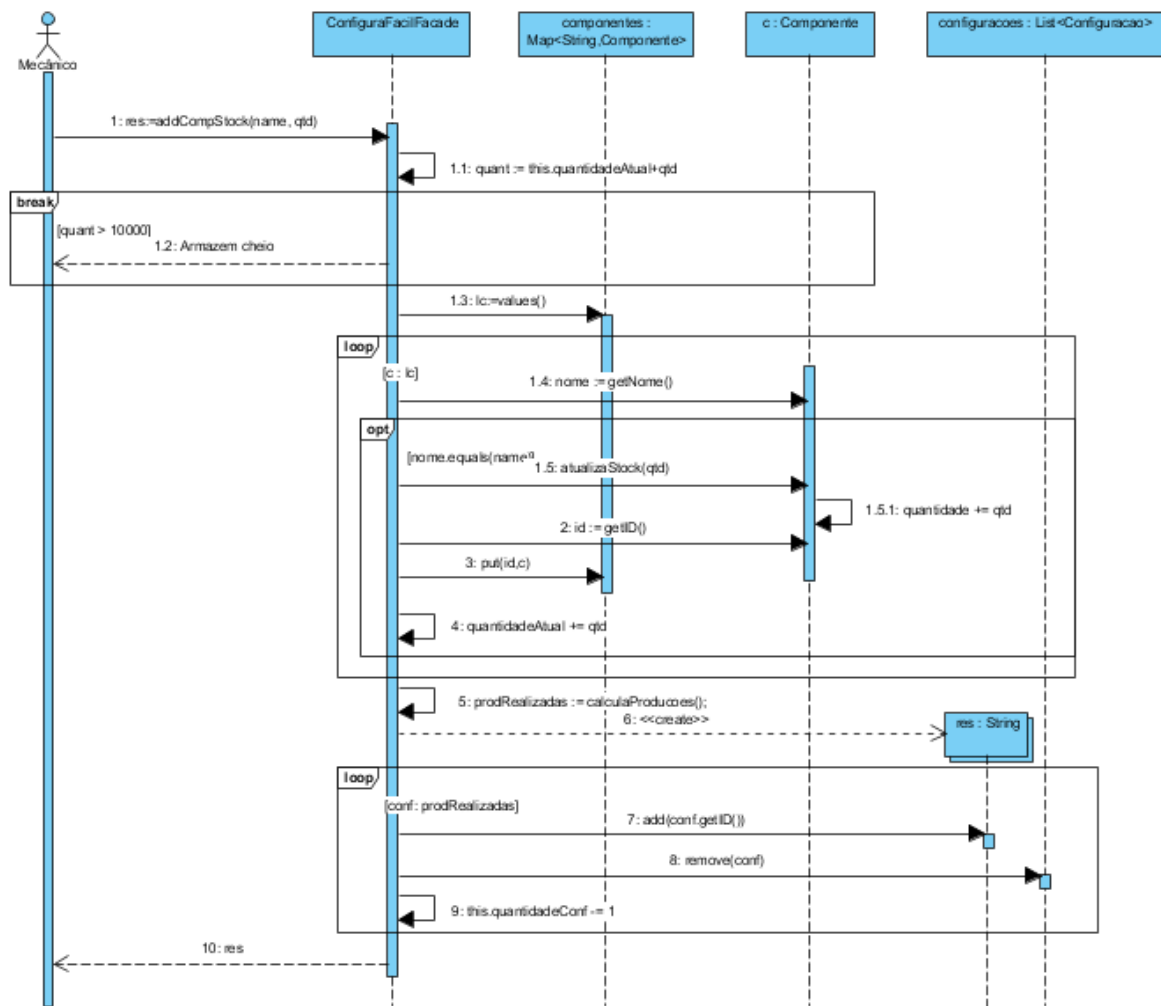
addCompStock

Método desenvolvido com o intuito de, para além de atualizar o stock de um dado componente, analisa posteriormente, que configurações podem ir para produção.

Ator-Facade-Classe



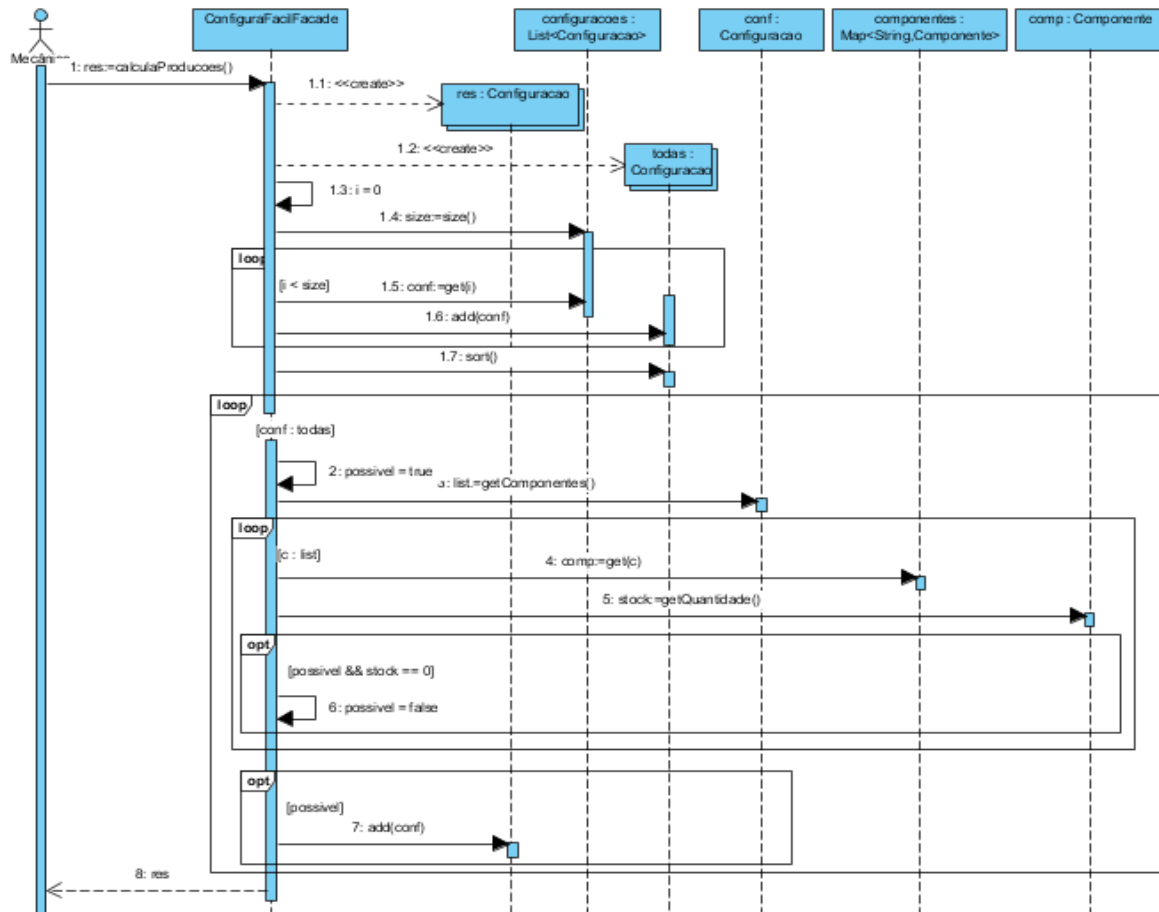
Ator-Facade-DAO



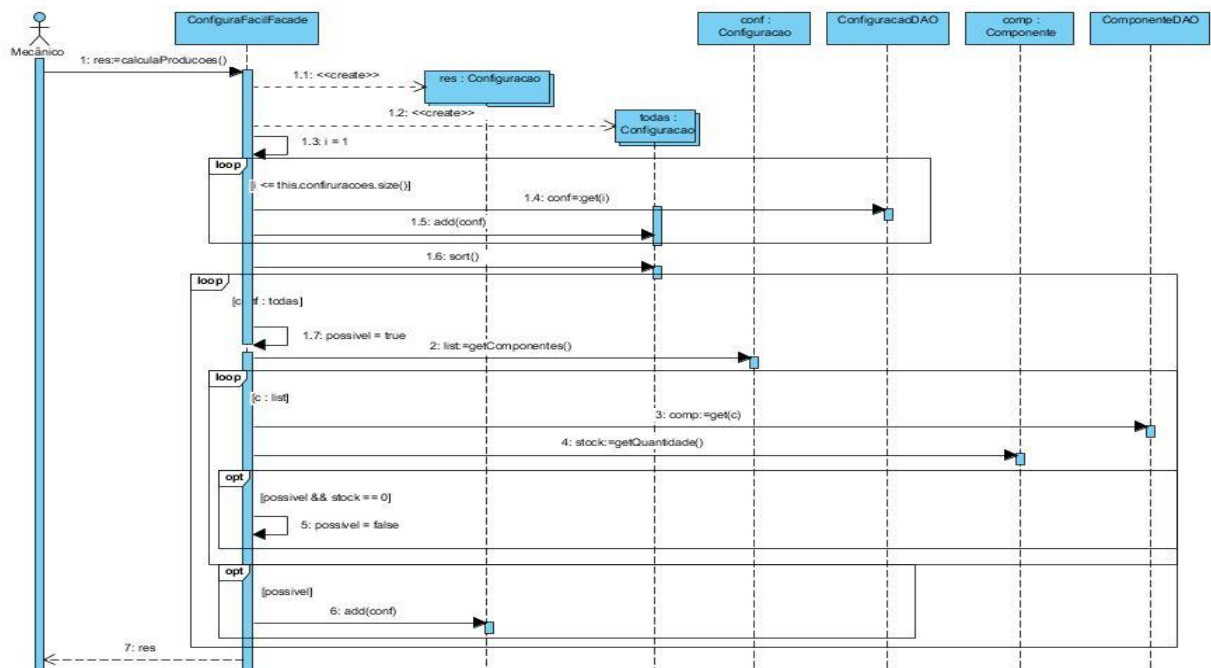
calculaProdicoes

Método usado para obter as configurações “aptas” para serem produzidas, sabendo que uma configuração só vai para produção quando todos os seus componentes tiverem quantidade suficiente de stock.

Ator-Facade-Classe



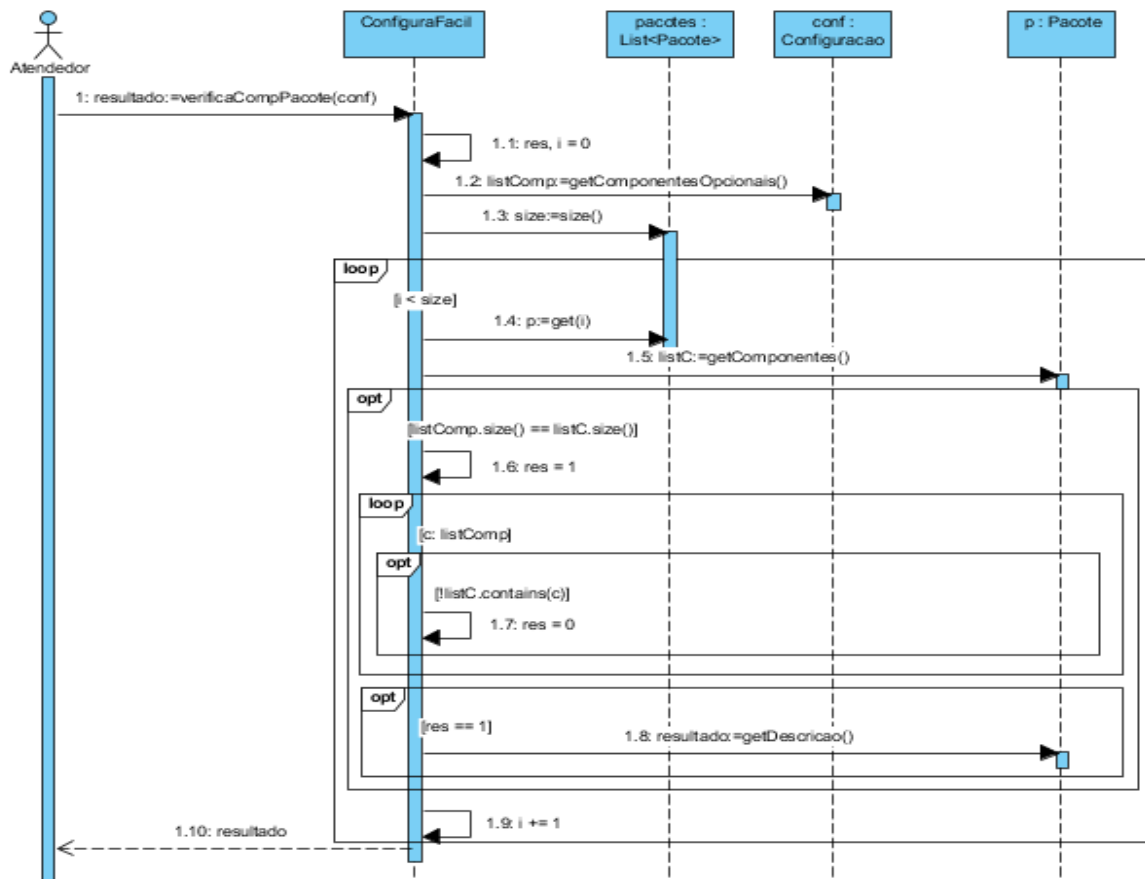
Ator-Facade-DAO



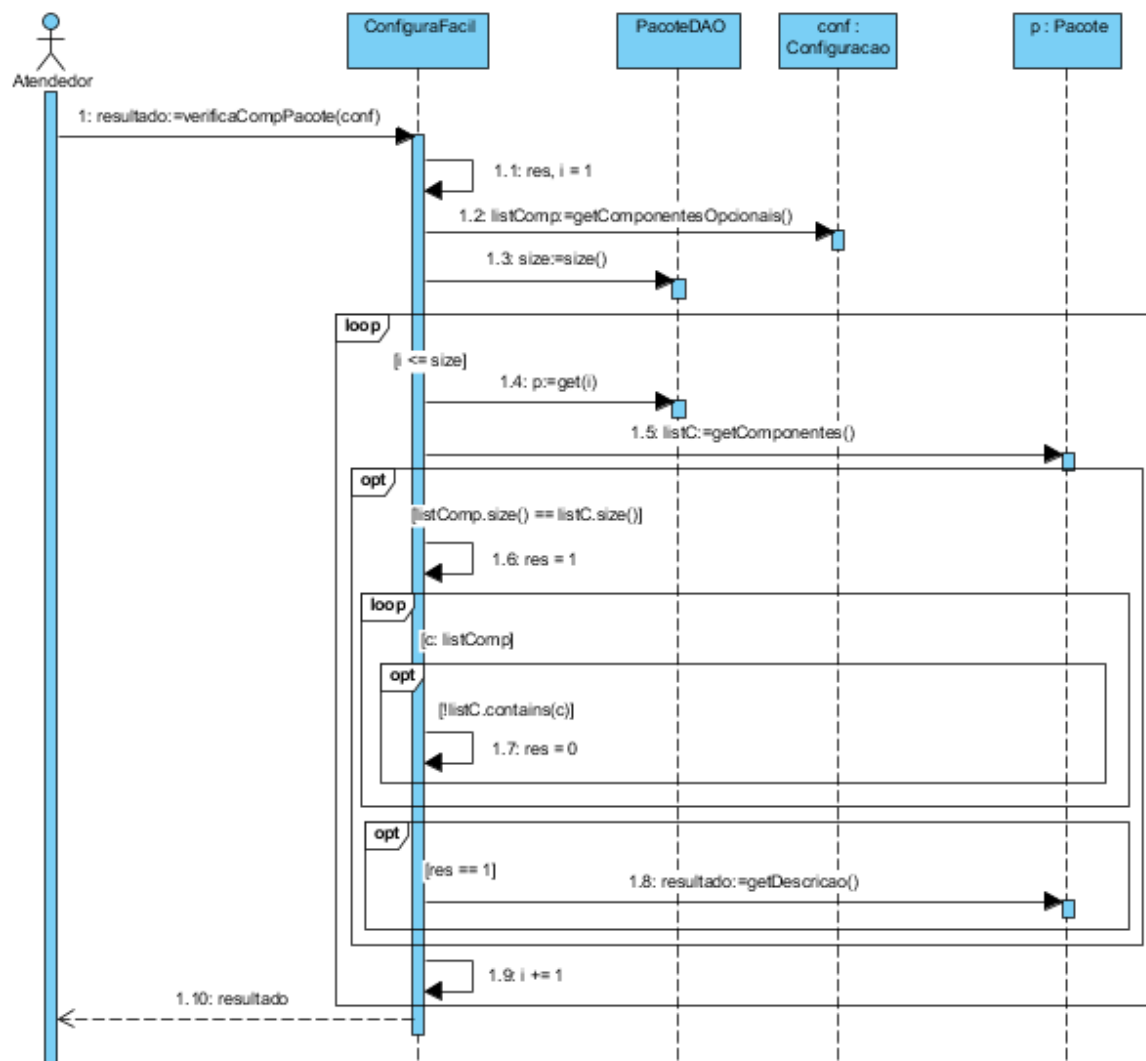
verificaCompPacote

Método que verifica se algum dos pacotes disponíveis corresponde ao conjunto de peças opcionais que foram introduzidas na configuração.

Ator-Facade-Classe



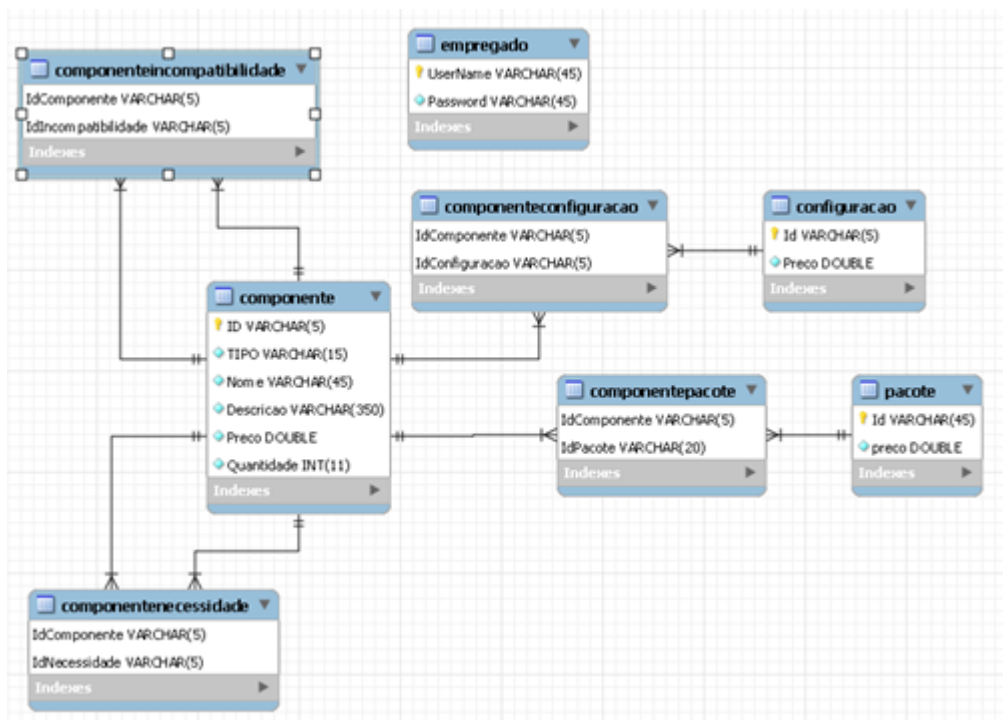
Ator-Facade-DAO



Base de dados

Com o intuito de formar e agrupar os dados num sistema de base de dados, o grupo formou o seguinte modelo lógico.

As tabelas “componenteIncompatibilidade”, “componenteNecessidade”, “componentePacote” e “componenteConfiguracao” foram criadas devido à necessidade de relacionar os componentes entre si e entre as configurações, uma vez que um componente (opcional) pode ter incompatibilidades e necessidades com outros componentes e uma configuração também está ligada a vários componentes, tal como um pacote é constituído por vários componentes.



Para o povoamento inicial da BD, usaram-se vários “INSERT” para colocar informação nas várias tabelas.

JDBC

Para se conseguir comunicar com a base de dados e atualizar constantemente a informação, é necessário que sejam estabelecidas conexões à mesma. Cada classe DAO tem uma variável do tipo *Connection*, que no início de cada método estabelece a ligação à BD e no final é fechada. Como um *Data Access Object* é um objeto persistido numa base de dados, é requerido que os métodos consigam fazer traduzir os parâmetros recebidos em “queries” funcionais em SQL.

A classe Connect cria a ligação a base de dados, usando as credenciais indicadas pelo utilizador, bem como o nome da “SCHEMA” sobre a qual vão ser feitas as atualizações de informação.

Conclusão

O grupo faz uma apreciação positiva sobre o trabalho realizado, porém existem alguns pontos do trabalho que poderiam ser melhorados.

Apesar de todos os diagramas estarem implementados o grupo acha que alguns poderiam ser modificados para uma melhor interpretação da implementação do programa “ConfiguraFacil” e o próprio programa poderia ter algumas modificações tanto em termos de segurança de acesso aos dados em memória tal como em termos de otimização para uma maior rapidez de execução dos métodos.