



Desenvolvimento de Sistemas de Software

LICENCIATURA EM ENGENHARIA INFORMÁTICA

URL: <https://github.com/LEI-DSS/trabalho-dss-grupo-11>

GRUPO11



a93320

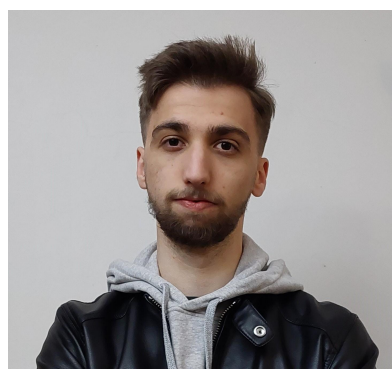
Augusto C.O. Campos



a93199

Carlos A.F.D. Silva

G11



a93285

Tiago P. Silva



a93182

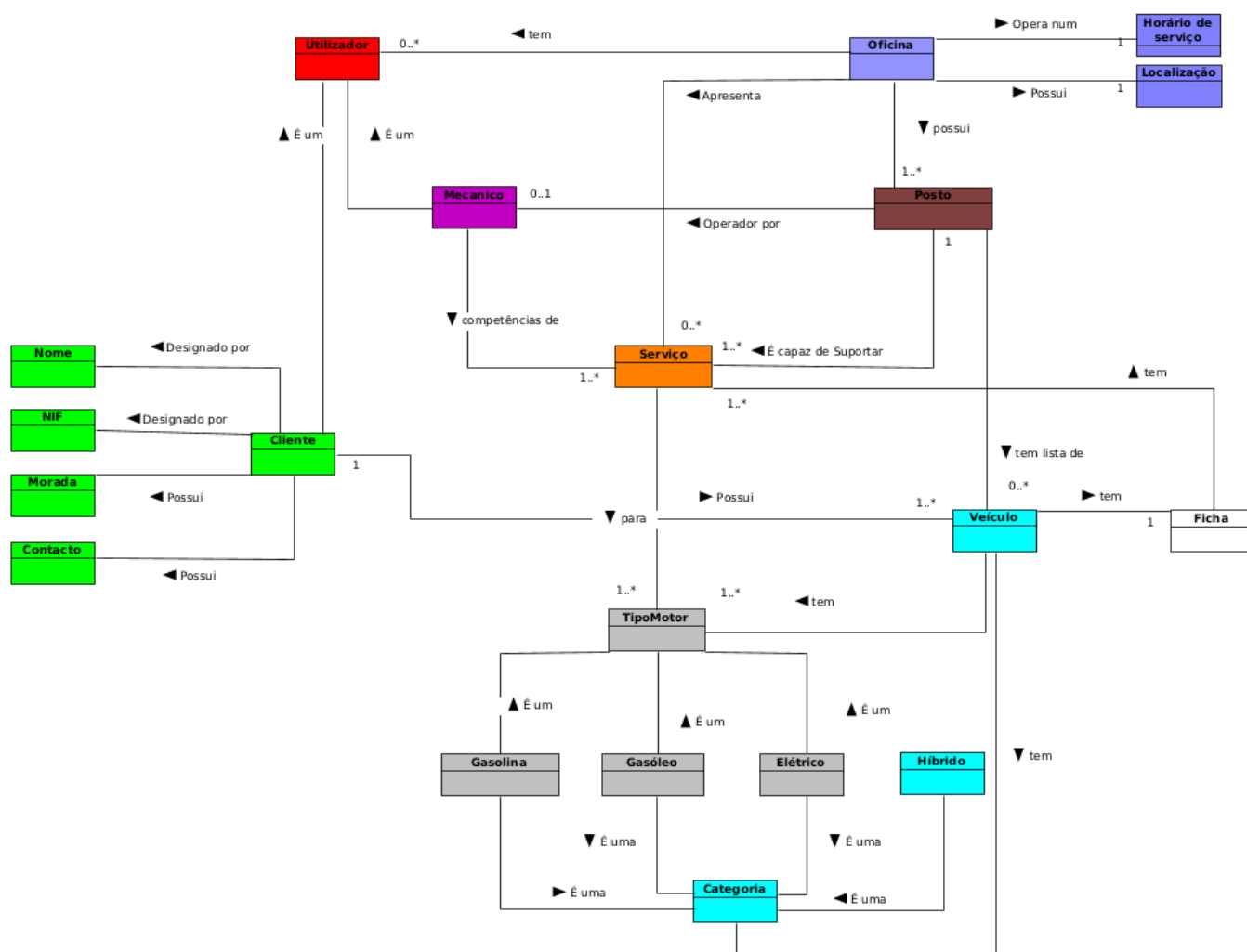
Tiago A.F. Silva

Índice

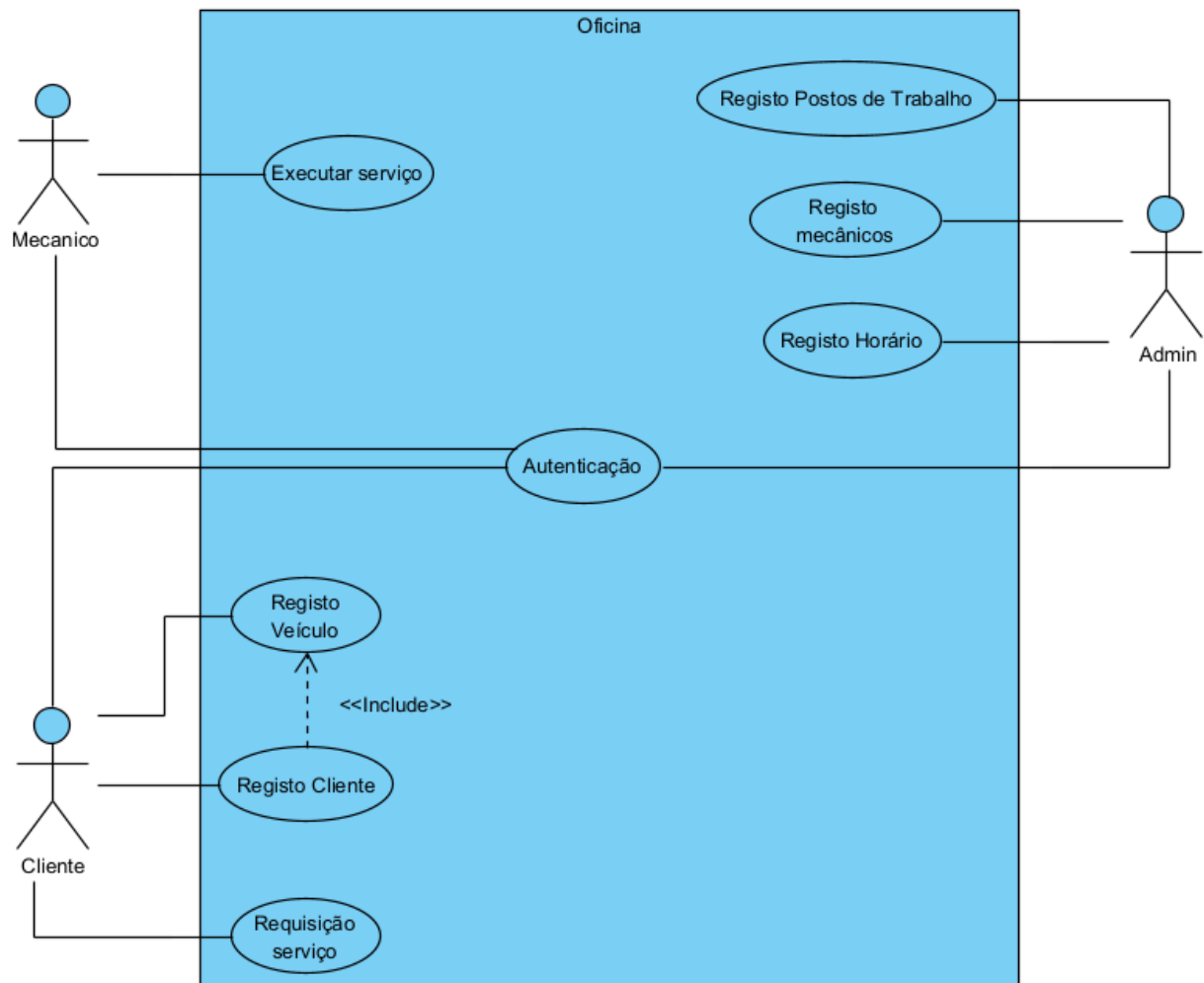
1. Diagramas de análise de requisitos	2
1.1 Modelo de Domínio	2
1.2 Diagramas de Casos de Uso	3
1.3 Descrições dos Casos de Usos	4
1.3.1 Autenticação	4
1.3.2 Registo Horário	4
1.3.3 Registo Cliente	5
1.3.4 Registo Posto Trabalho	5
1.3.5 Registo Veículo	6
1.3.6 Registo Mecânico	6
1.3.7 Requisitar Serviços	7
1.3.8 Executar Serviço	7
2. Diagramas de modelação conceptual	8
2.1 Diagrama de Classe	8
2.3 Diagramas de Sequência	9
2.3.1 Autenticar	9
2.3.2 Executar Serviço	9
2.3.3 Requisitar Serviço	10
3. Diagramas da Solução Efetivamente implementada	11
3.1.1 Diagrama de Classes (com os DAO's)	11
3.1.1.1 Diagrama de Classes (com os DAO's) Inicial	11
3.1.1.2 Diagrama de Classes (com os DAO's) Final	12
3.2 Diagramas de Sequência mudados depois da Implementação	13
3.2.1.1 Requisitar Serviço (com os DAO's) Inicial	13
3.2.1.2 Requisitar Serviço (com os DAO's) Final	14
3.2.2.1 Executar Serviço (com os DAO's) Inicial	15
3.2.2.2 Executar Serviço (com os DAO's) Final	16
3.2.3 Autenticar	17
3.3 Diagrama de packages	18
4. Resultados Obtidos	19

1. Diagramas de análise de requisitos

1.1 Modelo de Domínio



1.2 Diagramas de Casos de Uso



1.3 Descrições dos Casos de Usos

1.3.1 Autenticação

USE CASE:		Autenticar um Utilizador
DESCRIÇÃO:		O utilizador realiza a autenticação
CENÁRIOS:		1, 3 e 5
PRÉ-CONDIÇÃO:		
PÓS-CONDIÇÃO:		O sistema autentica o utilizador e dá-lhe acesso às informações e funcionalidades relevantes
FLUXO NORMAL:		
	1.	Utilizador escolhe a opção de autenticar
	2.	Utilizador fornece credenciais
	3.	Sistema verifica que as credenciais existem
	4.	Sistema verifica que as credenciais são validas no sistema
	5.	Sistema abre na conta associada às credenciais
FLUXO DE EXCEÇÃO	(1)	[credenciais não existem] (passo 3)
	3.1	Sistema informa o utilizador que não existe nenhuma conta com essas credenciais

1.3.2 Registo Horário

USE CASE:		Registo Horário
DESCRIÇÃO:		O Admin regista o horário de funcionamento de uma estação
CENÁRIOS:		1
PRÉ-CONDIÇÃO:		O Admin está autenticado.
PÓS-CONDIÇÃO:		O horário de funcionamento de uma estação é atualizado
FLUXO NORMAL:		
	1.	O Admin escolhe a opção atualizar o tempo de funcionamento da estação.
	2.	O Sistema pede o novo horário.
	3.	O Admin introduz o horário.
	4.	O Sistema valida o horário.
	5.	O Sistema atribui o horario à estação.
	6.	O Sistema apresenta ao admin que o horário foi atualizado com sucesso.
FLUXO DE EXCEÇÃO	(1)	[Horário Inválido] (passo 4)
	2.1	O Sistema informa que o horário inserido é inválido

1.3.3 Registo Cliente

USE CASE:		Registar um cliente
DESCRIÇÃO:		O cliente realiza um registo de cliente
CENÁRIOS:		2
PRÉ-CONDIÇÃO:		O Cliente está autenticado
PÓS-CONDIÇÃO:		Foi registado um cliente
FLUXO NORMAL:		
	1.	O Cliente escolhe a opção de registar cliente.
	2.	O Cliente fornece credenciais necessárias para o registo do cliente.
	3.	O Cliente indica estação de serviço pretendida.
	4.	O Sistema verifica que as credencias e a estação são válidas.
	5.	O Sistema cria um cliente novo na estação indicada.
	6.	Use case Registo de veículo passo 2
	7.	O Sistema apresenta ao cliente que foi registado com sucesso.
FLUXO DE EXCEÇÃO	(1)	[credenciais não válidas] (passo 4)
	4.1	Sistema informa o utilizador que não existe a possibilidade de registar um cliente com essas credenciais
FLUXO DE EXCEÇÃO	(2)	[estação não válida] (passo 4)
	4.1	Sistema informa o utilizador que a estação é inválida

1.3.4 Registo Posto Trabalho

USE CASE:		Registo posto de trabalho
DESCRIÇÃO:		Permite ao Admin registar um posto de trabalho
CENÁRIOS:		1
PRÉ-CONDIÇÃO:		O utilizador (Admin) está autenticado
PÓS-CONDIÇÃO:		O posto de trabalho fica registado no Sistema.
FLUXO NORMAL:		
	1.	Utilizador seleciona a opção "Registo posto de trabalho"
	2.	Utilizador indica tipo de serviço associado ao posto de trabalho
	3.	O Sistema valida o serviço escolhido.
	3.	O Sistema regista o posto de trabalho.
	4.	O Sistema apresenta ao admin que o posto foi registado com sucesso.
FLUXO DE EXCEÇÃO	(1)	[Serviço Inválido] (passo 3)
	3.1	O Sistema informa que o serviço inserido é inválido

1.3.5 Registo Veículo

USE CASE:		Registar um veículo
DESCRIÇÃO:		O Cliente realiza o registo de um veículo
CENÁRIOS:		2
PRÉ-CONDIÇÃO:		O Cliente está autenticado
PÓS-CONDIÇÃO:		Foi registado um veículo associado ao cliente
FLUXO NORMAL:		
	1.	O Cliente escolhe a opção de registar veículo.
	2.	O Cliente fornece credenciais necessárias para o registo do veículo.
	3.	Sistema verifica que as credenciais existem
	4.	Sistema verifica que as credenciais são validas.
	5.	Sistema criar um registo de veículo associada a conta autenticada.
	6.	O Sistema apresenta ao cliente que foi registado o veículo com sucesso.
FLUXO DE EXCEÇÃO	(1)	[credenciais válidas] (passo 3)
	3.1	Sistema informa o utilizador que as credencias introduzidas não são válidas

1.3.6 Registo Mecânico

USE CASE:		Registo Mecânico
DESCRIÇÃO:		O utilizador (Admin) regista um Mecânico.
CENÁRIOS:		1
PRÉ-CONDIÇÃO:		O Utilizador (Admin) está autenticado.
PÓS-CONDIÇÃO:		O Mecânico fica registado no Sistema.
FLUXO NORMAL:		
	1.	O Admin seleciona a opção de Registo de um mecânico
	2.	O Admin fornece credenciais necessárias para o registo de um Mecânico
	3.	Sistema verifica se as credenciais introduzidas são válidas
	4.	O Sistema regista o Mecânico no Sistema.
	5.	O Sistema apresenta ao admin que o Meacanico foi registado com sucesso.
FLUXO DE EXCEÇÃO	(1)	[credenciais não válidas] (passo 3)
	4.1	Sistema informa o admin que as credenciais não são válidas.

1.3.7 Requisitar Serviços

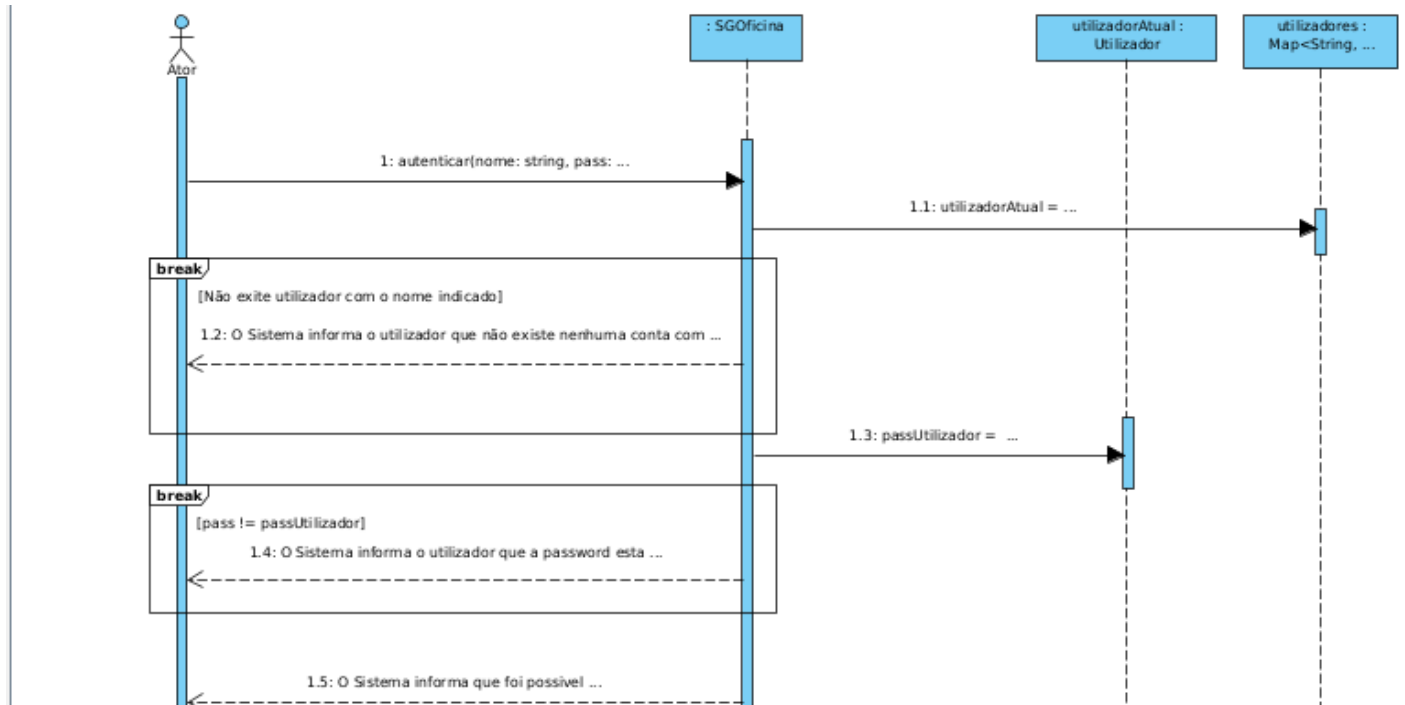
USE CASE:		Requisitar serviços
DESCRIÇÃO:		O cliente requisita um serviço à estação de serviço.
CENÁRIOS:		3 e 4
PRÉ-CONDIÇÃO:		O cliente e o veículo estão registrados.
PÓS-CONDIÇÃO:		O serviço é adicionado à lista de espera.
FLUXO NORMAL:		
	1.	O Cliente escolhe a opção de Requisitar serviço.
	2.	O Cliente indica o serviço que deseja que seja realizado.
	3.	O Sistema atualiza a ficha de veículo para a necessidade do serviço.
	4.	O Sistema verifica a disponibilidade dos postos de serviços.
	5.	O Sistema adiciona o serviço à lista de espera.
	6.	O Sistema informa que o serviço foi adicionado com sucesso.
FLUXO DE EXCEÇÃO	(1)	[O sistema verifica que não existe disponibilidade para realizar o serviço] (passo 4)
	4.1	O Sistema informa o cliente que não será possível realizar o serviço naquele dia.

1.3.8 Executar Serviço

USE CASE:		Executar Serviço
DESCRIÇÃO:		Um serviço irá ser executado
CENÁRIOS:		5
PRÉ-CONDIÇÃO:		O utilizador (Mecânico) está autenticado.
PÓS-CONDIÇÃO:		O serviço é executado com sucesso
FLUXO NORMAL:		
	1.	O Mecânico seleciona a opção de executar serviço.
	2.	O Sistema verifica se existem serviços a serem executados
	3.	O Serviço é executado
	4.	O Sistema atualiza o histórico de serviços realizados no histórico do veículo
	5.	O Sistema passa o próximo serviço a ser executado para o topo da lista de serviços.
	6.	O Sistema apresenta que o serviço foi executado ao mecânico.
FLUXO DE EXCEÇÃO	(1)	[Lista de serviços vazia] (passo 2)
	2.1	O Sistema informa o Mecânico que não existem serviços para serem executados.

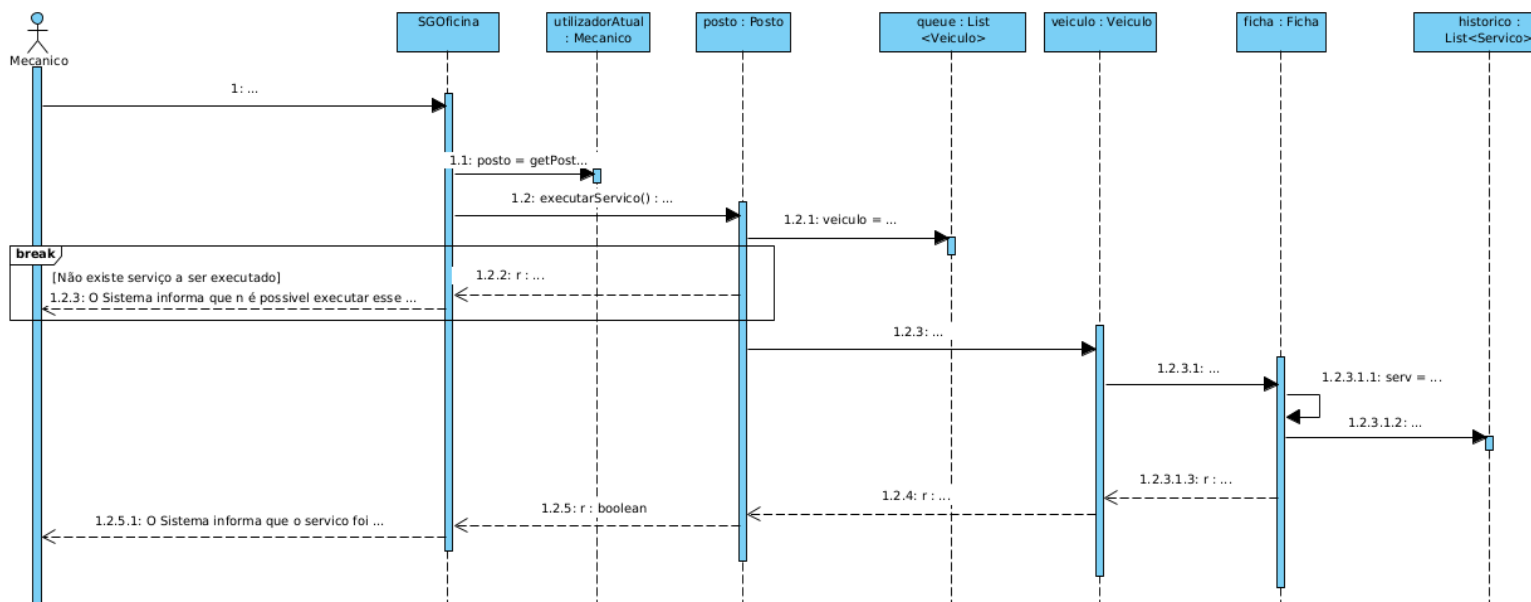
2.3 Diagramas de Sequência

2.3.1 Autenticar



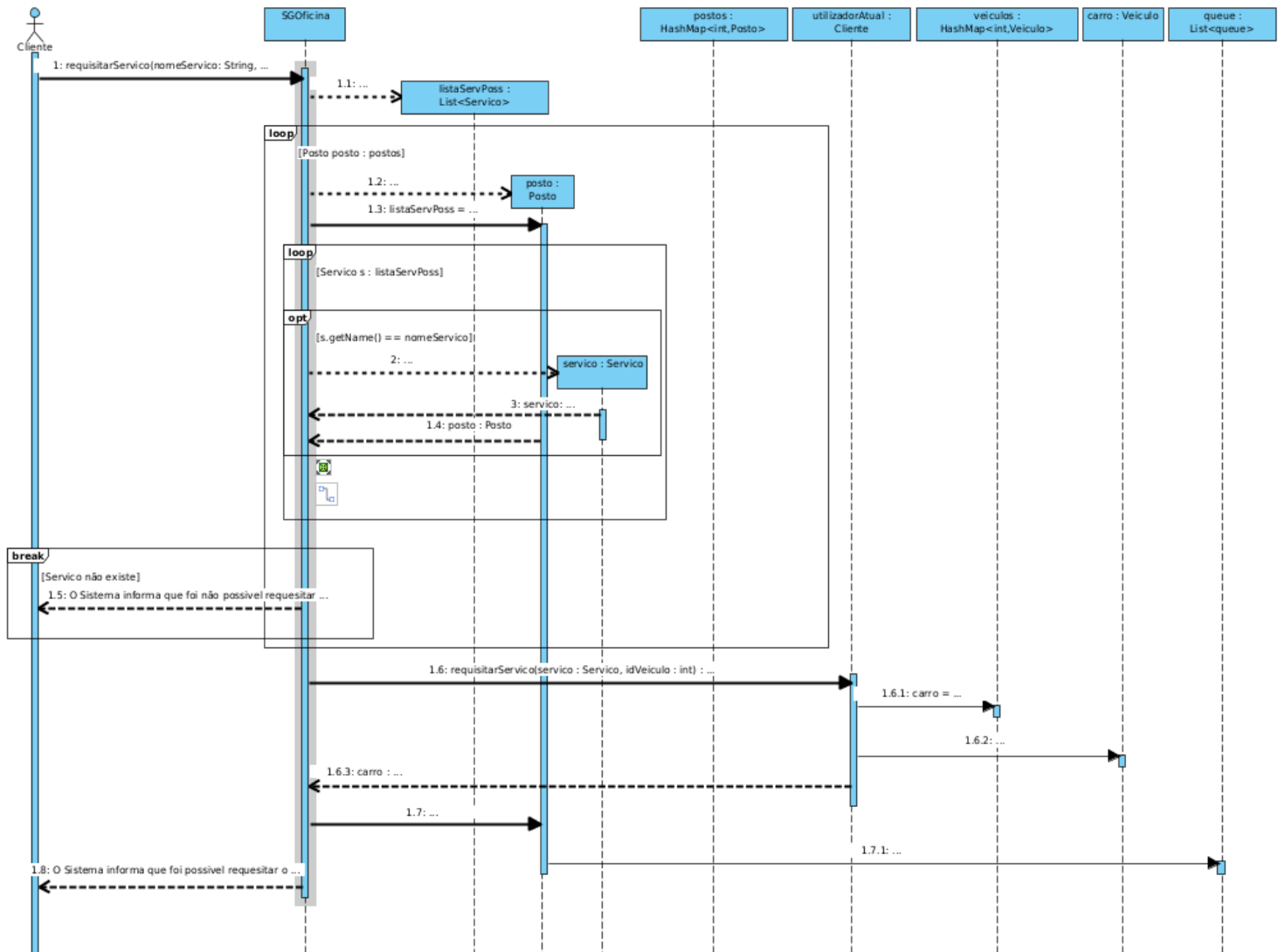
2.3.2 Executar Serviço

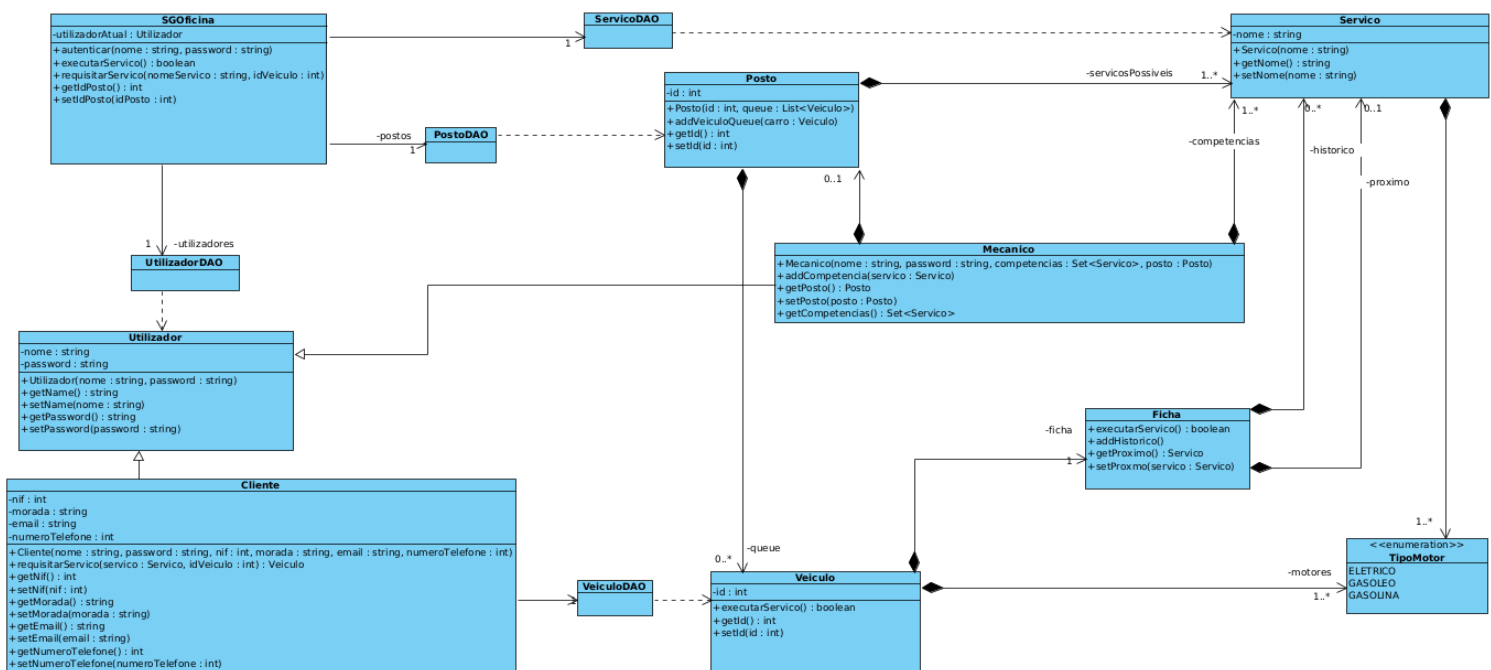
sd [executarServico]



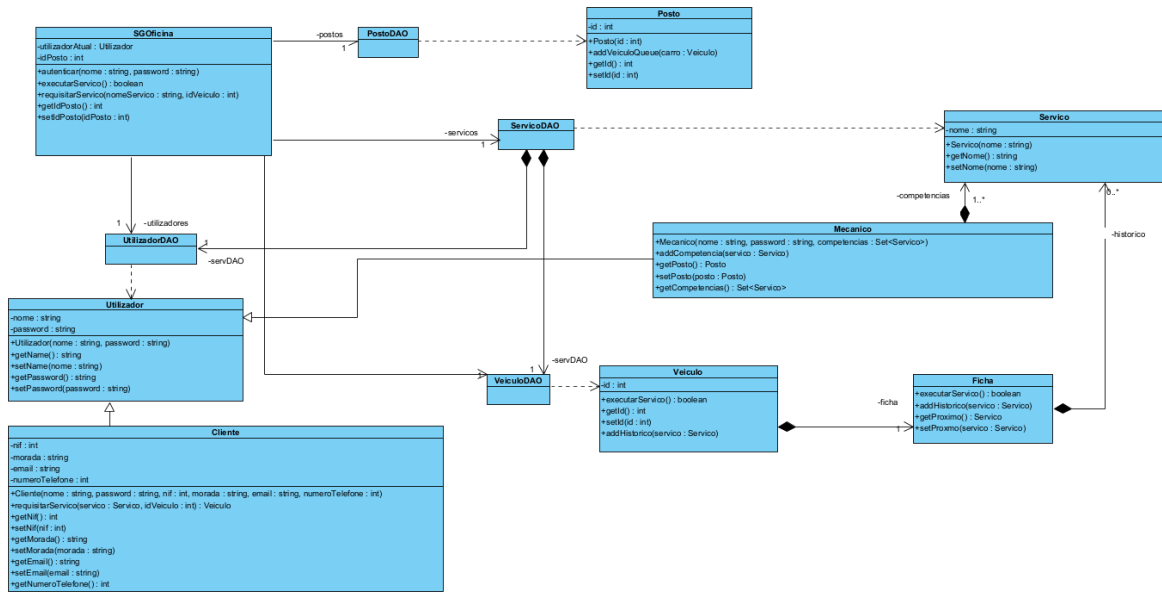
2.3.3 Requisitar Serviço

sd (requisitarServico)



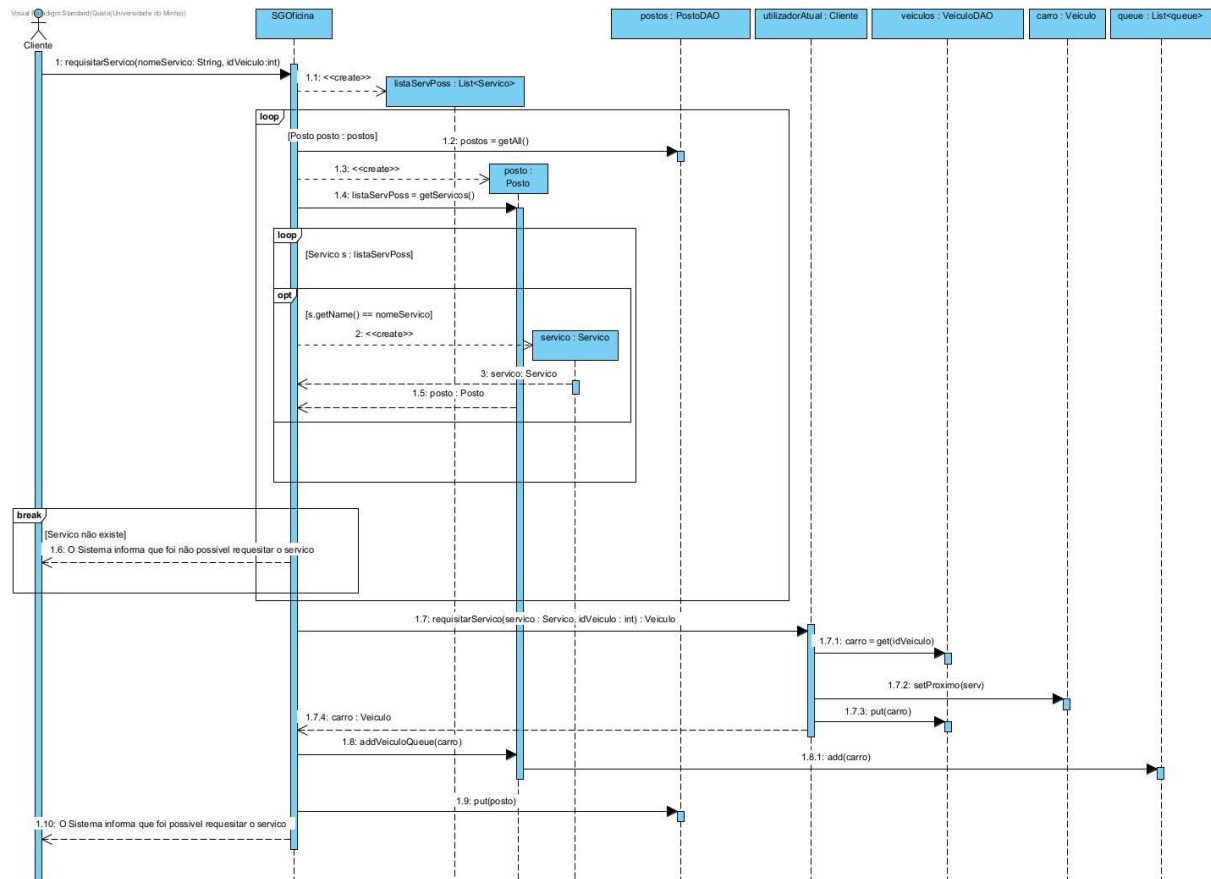


3.1.1.2 Diagrama de Classes (com os DAO's) Final

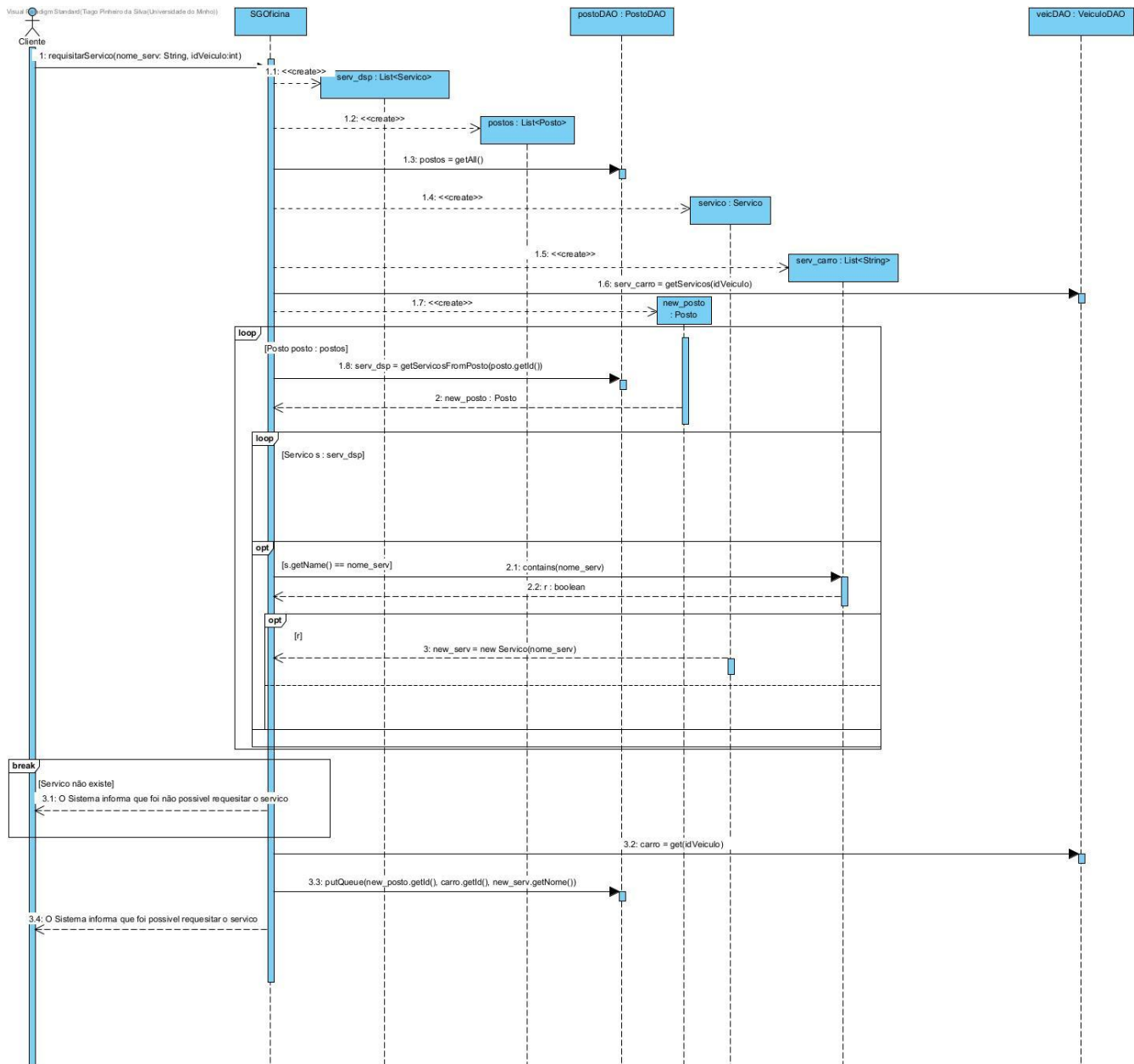


3.2 Diagramas de Sequência mudados depois da Implementação

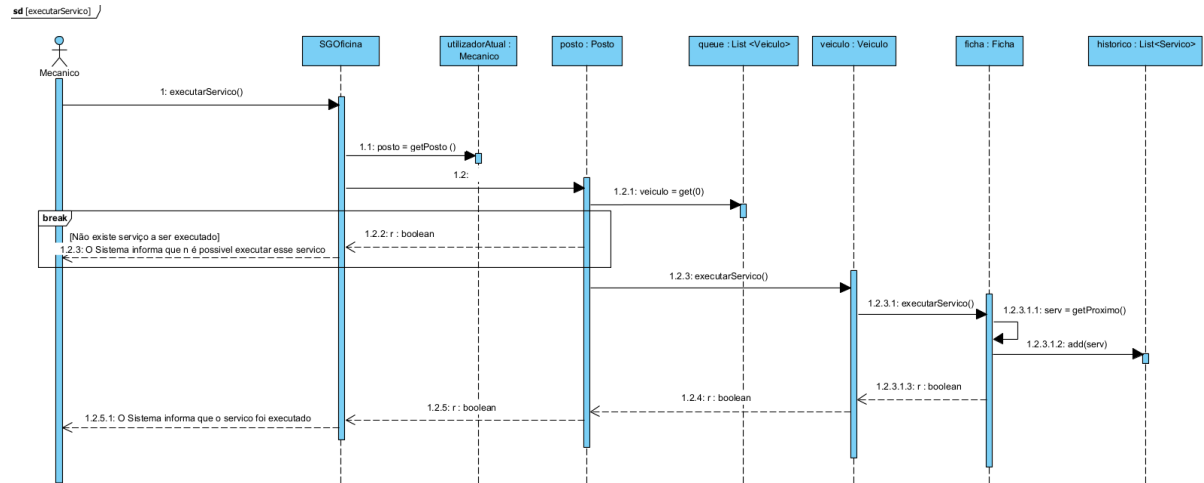
3.2.1.1 Requisitar Serviço (com os DAO's) Inicial



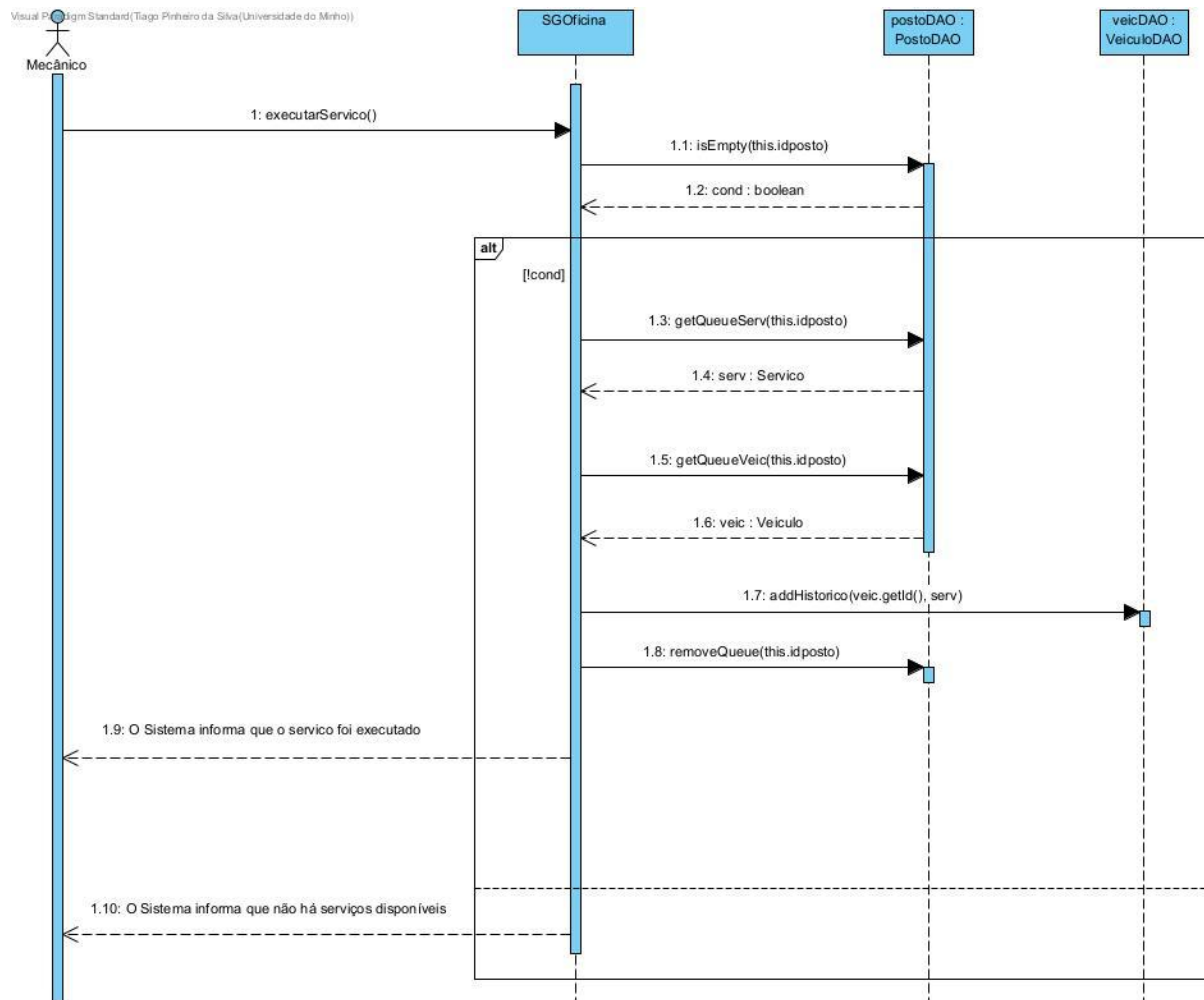
3.2.1.2 Requisitar Serviço (com os DAO's) Final



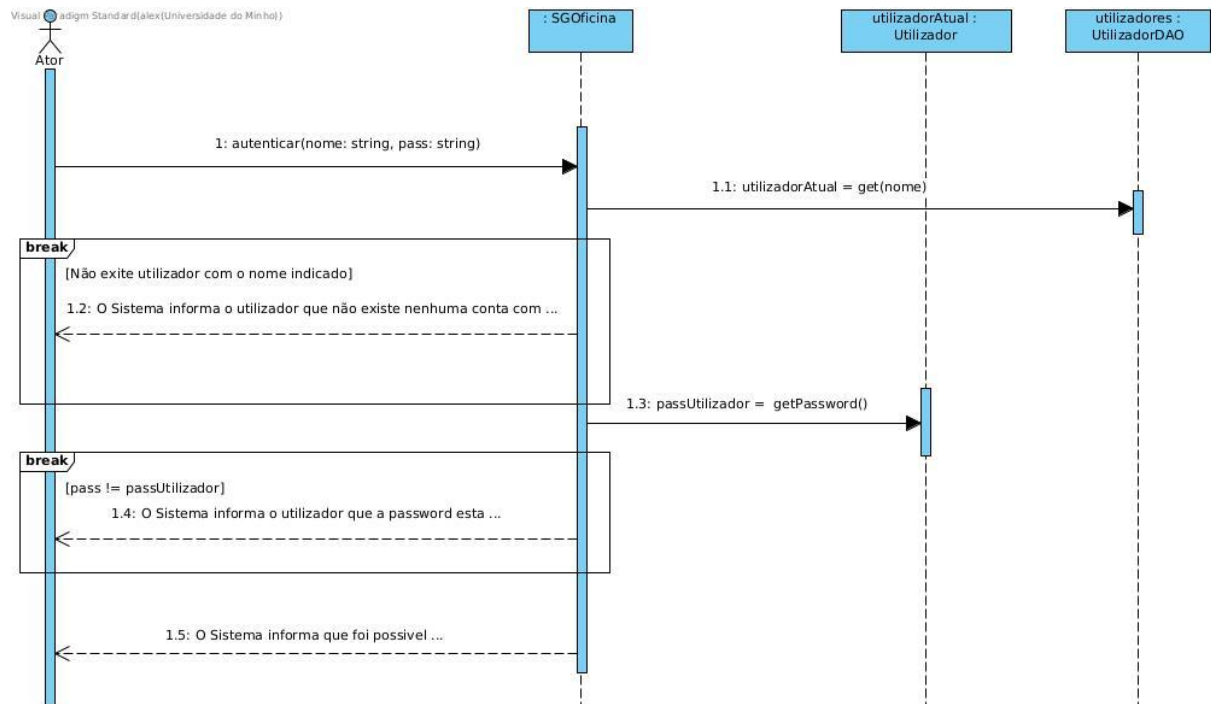
3.2.2.1 Executar Serviço (com os DAO's) Inicial



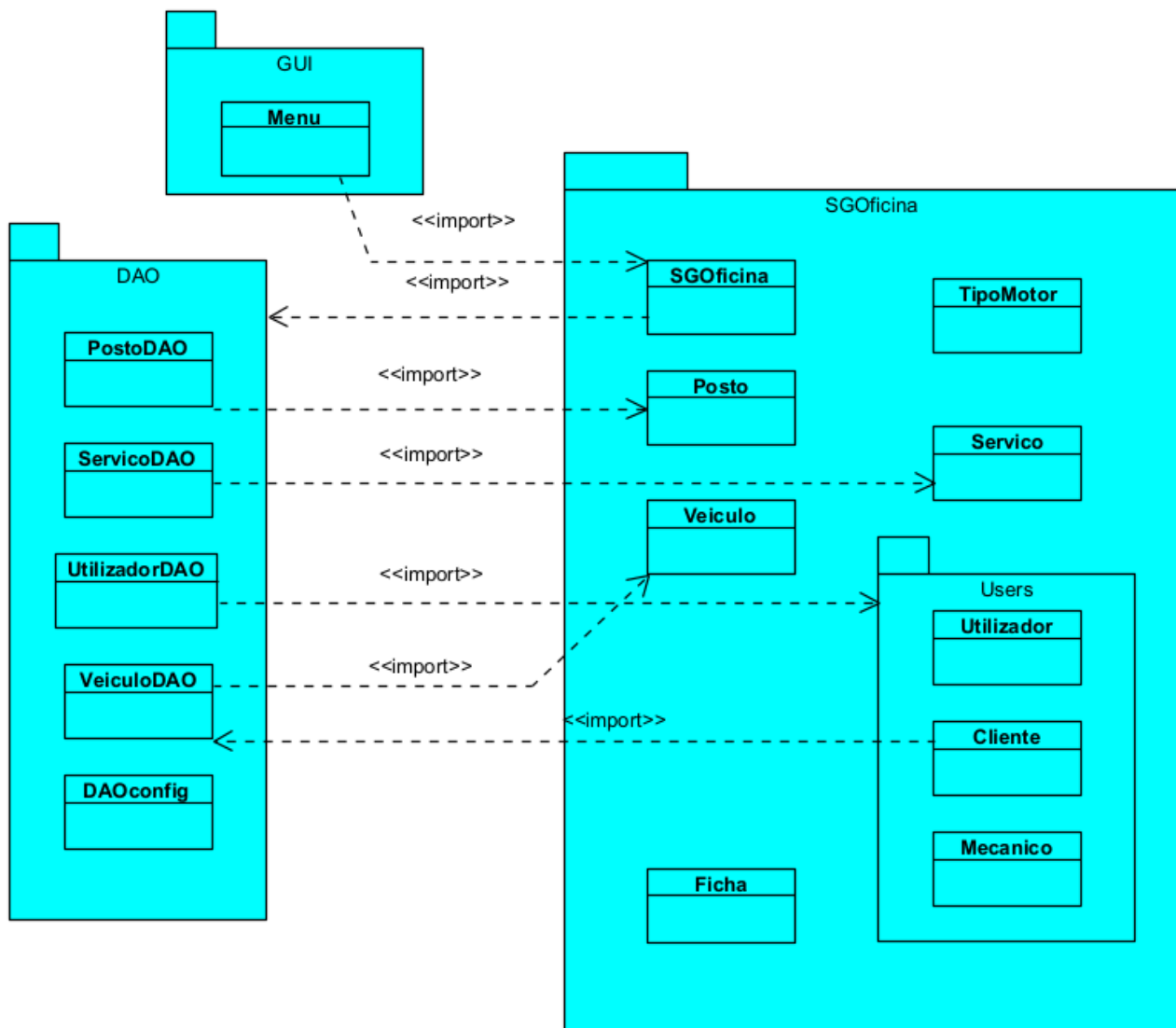
3.2.2.2 Executar Serviço (com os DAO's) Final



3.2.3 Autenticar



3.3 Diagrama de *packages*



4 Resultados Obtidos

Ao longo do Desenvolvimento do projeto percebemos a importância de uma boa análise do problema e de uma preparação de uma solução antes de uma fase de implementação, Apresentamos 4 grandes fases do projeto:

1. Realização do Modelo de Domínio e descrição dos *Use-Cases*

Através da análise dos cenários foi feito o levantamento de use-cases e do modelo de domínio, nesta fase tentamos ao máximo extrair a informação útil que previmos vir a precisar na implementação da camada de negócio e de dados.

2. Realização do Diagrama de Classes inicial com o uso de HashMaps

Após criarmos os respectivos Modelo de Domínio e Use-Cases analisamos de maneira a criar o Diagrama de Classes e os respectivos Diagramas de Sequência, nesta fase usamos HashMaps, Listas e Set's como maneira de armazenar informação, uma vez que não existe a necessidade de haver persistência dos dados.

3. Realização do Diagrama de Classes com o uso de *Data Access Object*

Nesta fase traduzimos a necessidade de *HashMaps*, Listas e *Set's* para o uso de tabelas numa base de dados que serão acessadas através de *DAO's*, estas classes têm a única responsabilidade de efetuar operações nas tabelas referidas, desta forma é garantida a persistência de dados e a integridade da aplicação.

4. Implementação final

Após a fase 3 passamos a implementação na linguagem *java* onde percebemos que algumas das classes que idealizamos tiveram de sofrer alterações, esta fase mostrou-nos que apesar de o planeamento ser importante e essencial para a melhor organização do processo de desenvolvimento de *software* não devemos ter medo de mudar e iterar sobre o planeamento feito previamente se isto levar a uma melhor entrega das funcionalidades iniciais descritas.

Em conclusão salientamos que a aplicação final obtida deveria ser iterada ainda mais, no enunciado foram referidos mais funcionalidades que ficaram fora do *scope* definido pelos docentes da UC, a adição destas funcionalidades poderia levar a novas mudanças na implementação, de modo que, o processo de desenvolvimento de *software* é iterativo e cumulativo sem haver na verdade uma versão que seja verdadeiramente final.