



Universidade do Minho

Licenciatura em Engenharia Informática

UC de Desenvolvimento de Sistemas de Software

Ano Letivo de 2023/2024

Trabalho prático - Documento técnico

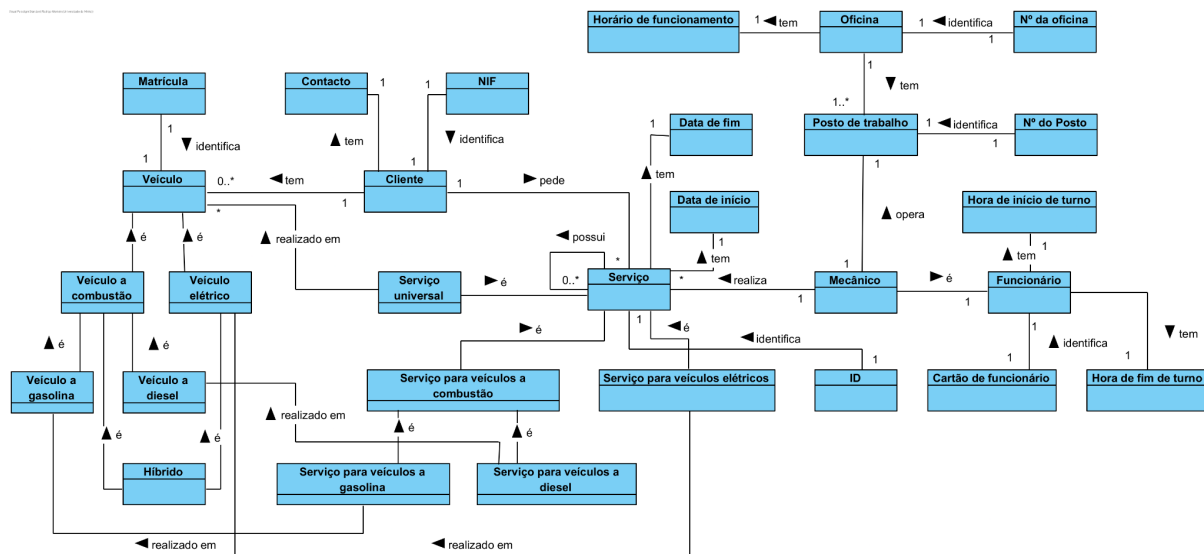
Rodrigo Monteiro, Diogo Abreu, e Luís Figueiredo

e-mail: {a100706,a100646,a100549}@alunos.uminho.pt

Índice

| | |
|---|----|
| 1. Modelo de domínio..... | 2 |
| 2. Descrições dos casos de uso..... | 2 |
| 3. Diagrama de casos de uso..... | 5 |
| 4. Diagrama de componentes..... | 6 |
| 5. Diagrama de classes sem DAOs (conceptual)..... | 7 |
| 6. Diagrama de classes com DAOs (conceptual)..... | 8 |
| 7. Modelo lógico da base de dados..... | 9 |
| 8. Diagrama de classes (implementação)..... | 10 |
| 9. Diagramas de sequência (implementação)..... | 11 |

1. Modelo de domínio



2. Descrições dos casos de uso

Caso de uso: Adicionar serviços

Descrição: Um funcionário adiciona um ou mais serviços ao sistema pedidos por um cliente.

Cenários: (4) Visita à estação de serviço 2; e (3) Visita à estação de serviço 1

Pré-condição: O cliente e o seu veículo estão registados no sistema. O sistema faz uma previsão de modo a que o serviço possa ser realizado o mais cedo possível.

Pós-condição: Os serviços são registados.

Fluxo normal:

1. Cliente pede que um ou mais serviços sejam efetuados.
2. Funcionário adiciona os serviços ao sistema.
3. Sistema analisa os serviços previstos.
4. Sistema fornece uma estimativa da hora a que os serviços estarão terminados.
5. Cliente concorda que os serviços sejam realizados.
6. Sistema fornece a opção de o cliente ser notificado quando os serviços terminarem.
7. Cliente indica a sua escolha.
8. Opção é registada.

Fluxo alternativo (1): [Parte dos serviços não podem ser realizados] (passo 3)

- 3.1. Sistema informa acerca dos serviços que podem ser realizados.
- 3.2. Avança para o passo 4.

Fluxo de exceção (1): [Nenhum serviço pode ser realizado] (passo 3)

- 3.1. O sistema termina o processo.

Fluxo de exceção (2): [Cliente não concorda que os serviços sejam realizados] (passo 5)

- 5.1. O sistema termina o processo.

Caso de uso: Check-up

Descrição: Um cliente pede para que seja efetuado um check-up — um serviço gratuito em que se faz a verificação do veículo e se identificam eventuais intervenções que sejam necessárias.

Cenários: (3) Visita à estação de serviço 1

Pré-condição: O cliente e o seu veículo estão registados.

Pós-condição: Check-up e os serviços resultantes são realizados.

Fluxo normal:

1. «include» Adicionar serviços.
2. Ficha do veículo é atualizada para assinalar a necessidade de um check-up.
3. Check-up é realizado.
4. Sistema informa quais são os serviços necessários a efetuar com base na ficha do veículo.
5. Sistema calcula a ordem de serviço necessária à execução dos trabalhos
6. Sistema fornece uma estimativa da hora a que o serviço estará terminado.
7. Cliente concorda que os serviços sejam efetuados.

Fluxo alternativo (1): [Existem serviços de um check-up anterior por realizar] (passo 1)

- 1.1. Cliente escolhe realizar apenas os serviços necessários do check-up anterior.
- 1.2. Avança para o passo 4.

Fluxo alternativo (2): [Cliente escolhe realizar novamente um check-up] (passo 1.1)

- 1.1.1. Avança para o passo 2.

Fluxo de exceção (1): [Cliente não concorda que os serviços sejam efetuados] (passo 7)

- 7.1. O sistema termina o processo.

Caso de uso: Comunicar conclusão de serviços

Descrição: O sistema envia atualizações automáticas por SMS para informar os clientes sobre o estado dos serviços realizados.

Cenários: (3) Visita à estação de serviço 1

Pré-condição: Cliente fez um pedido de um ou mais serviços, estes terminaram, e o cliente escolheu ser notificado.

Pós-condição: Cliente é notificado.

Fluxo normal:

1. Sistema notifica o cliente, informando que os serviços foram realizados com sucesso.

Fluxo alternativo (1): [Um ou mais serviços não puderam ser realizados] (passo 1)

- 1.1. Sistema informa acerca dos serviços que foram realizados e dos que não puderam ser realizados.

Caso de uso: Iniciar turno

Descrição: Um funcionário inicia o seu turno num posto de trabalho de uma oficina

Cenários: (5) Funcionários

Pré-condição: Funcionário está registado no sistema.

Pós-condição: Sistema regista início do turno.

Fluxo normal:

1. O funcionário passa o seu cartão de funcionário no posto de trabalho.
2. O sistema verifica que o funcionário tem competências para trabalhar no posto.
3. O sistema regista a hora de início de turno.

Fluxo de exceção (1): [Funcionário não tem competências] (passo 2)

- 2.1. O sistema verifica que o funcionário não tem competências para trabalhar no posto.
- 2.2. O sistema informa que o funcionário não pode trabalhar no posto.

Caso de uso: Realizar serviço

Descrição: Um mecânico realiza um serviço num posto de trabalho de uma oficina.

Cenários: (5) Funcionários

Pré-condição: O mecânico iniciou o turno no mesmo posto de trabalho.

Pós-condição: O serviço é concluído.

Fluxo normal:

1. Sistema mostra a lista de serviços.
2. Mecânico sinaliza o primeiro serviço da lista.
3. Sistema regista a data de início.
4. Mecânico indica a conclusão do serviço.
5. A ficha do veículo é atualizada.

Fluxo de exceção (1): [Serviço não pôde ser concluído] (passo 4)

- 4.1. O mecânico sinaliza que não foi possível concluir o serviço.
- 4.2. O mecânico indica o motivo.

Caso de uso: Finalizar turno

Descrição: Um funcionário finaliza o seu turno num posto de trabalho de uma oficina.

Cenários: (5) Funcionários

Pré-condição: O funcionário iniciou o turno no mesmo posto de trabalho. Já chegou a hora do fim de turno.

Pós-condição: Sistema regista fim de turno.

Fluxo normal:

1. O funcionário passa o seu cartão de funcionário no posto de trabalho.
2. O sistema regista a hora do fim do turno.

Caso de uso: Registo de um cliente

Descrição: Um utilizador é registado na aplicação de gestão de serviços de uma oficina.

Cenários: (2) Registo de um cliente.

Pré-condição: O utilizador tem de estar registado no website da E.S. Ideal.

Pós-condição: O utilizador é registado.

Fluxo normal:

1. O servidor da E.S.Ideal envia a informação recolhida no registo para o sistema da oficina correspondente.
2. O sistema da oficina guarda a informação do utilizador na base de dados.

Caso de uso: Configuração do sistema

Descrição: Um administrador da E.S Ideal instala o sistema numa oficina.

Cenários: (1) Configuração do sistema

Pré-condição: O administrador está registado com as permissões necessárias.

Pós-condição: O sistema está funcional.

Fluxo normal:

1. Administrador escolhe registar postos de trabalho.
2. «include» Registar posto de trabalho.
3. Administrador escolhe registar mecânicos.
4. «include» Registar mecânico.
5. Administrador escolhe registar horário de funcionamento.
6. Administrador indica o horário para cada dia.
7. Sistema regista horário de funcionamento.

Fluxo alternativo (1): [Administrador escolhe registar mais um posto de trabalho] (passo 3)

- 3.1. Regressa ao passo 2.

Fluxo alternativo (2): [Administrador escolhe registar mais um mecânico] (passo 5)

- 5.1. Regressa ao passo 4.

Caso de uso: Registrar posto de trabalho

Descrição: Um administrador registra um posto de trabalho no sistema.

Cenários: (1) Configuração do sistema

Pré-condição: O administrador está registrado com as permissões necessárias.

Pós-condição: Posto de trabalho fica registrado no sistema.

Fluxo normal:

1. Administrador adiciona um posto de trabalho, indicando o número do posto.
2. Administrador indica o tipo de serviço.
3. Sistema registra posto de trabalho.

Fluxo de exceção (1): [Número de posto já está registrado] (passo 1)

- 1.1. Sistema avisa que o número de posto indicado já está registrado no sistema.

Caso de uso: Registrar mecânico

Descrição: Um administrador registra um mecânico no sistema.

Cenários: (1) Configuração do sistema

Pré-condição: O administrador está registrado com as permissões necessárias.

Pós-condição: Mecânico fica registrado no sistema.

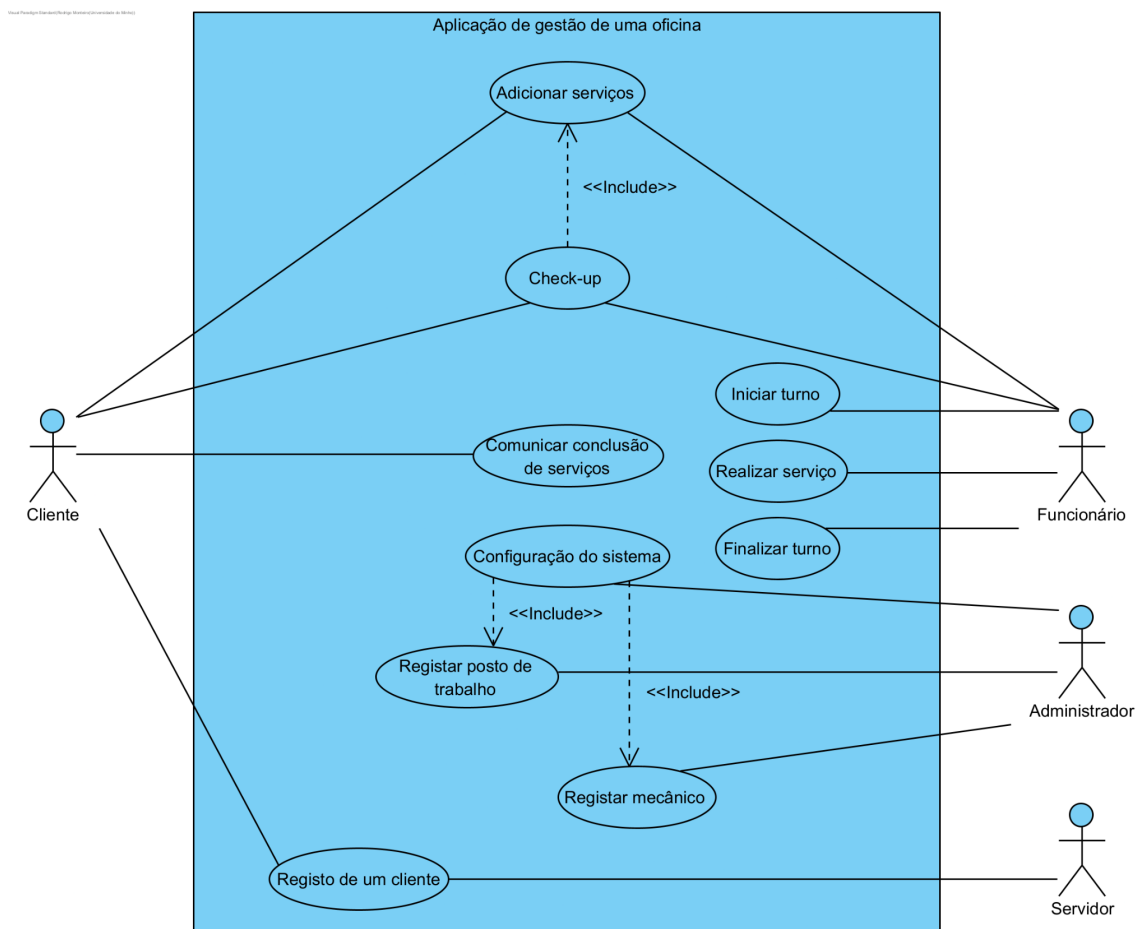
Fluxo normal:

1. Administrador indica o nome e número do mecânico.
2. Administrador indica a hora de entrada e hora de saída.
3. Sistema registra o mecânico.

Fluxo de exceção (1): [Número de mecânico já está registrado] (passo 1)

- 1.1. Sistema avisa que o número de mecânico indicado já está registrado no sistema.

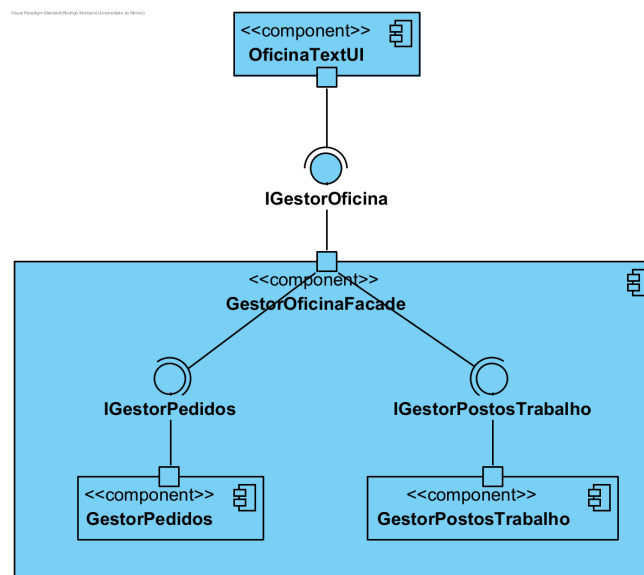
3. Diagrama de casos de uso

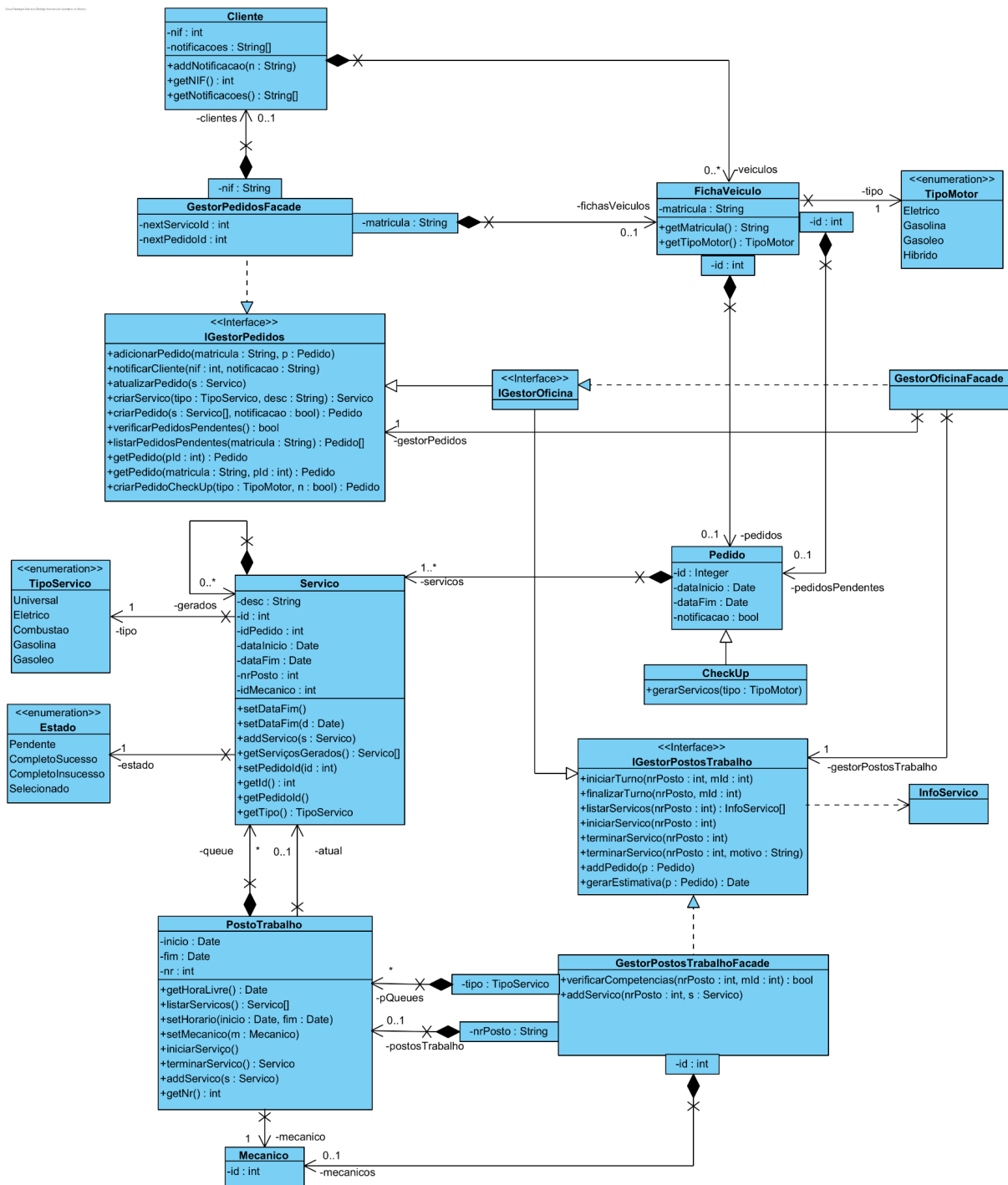


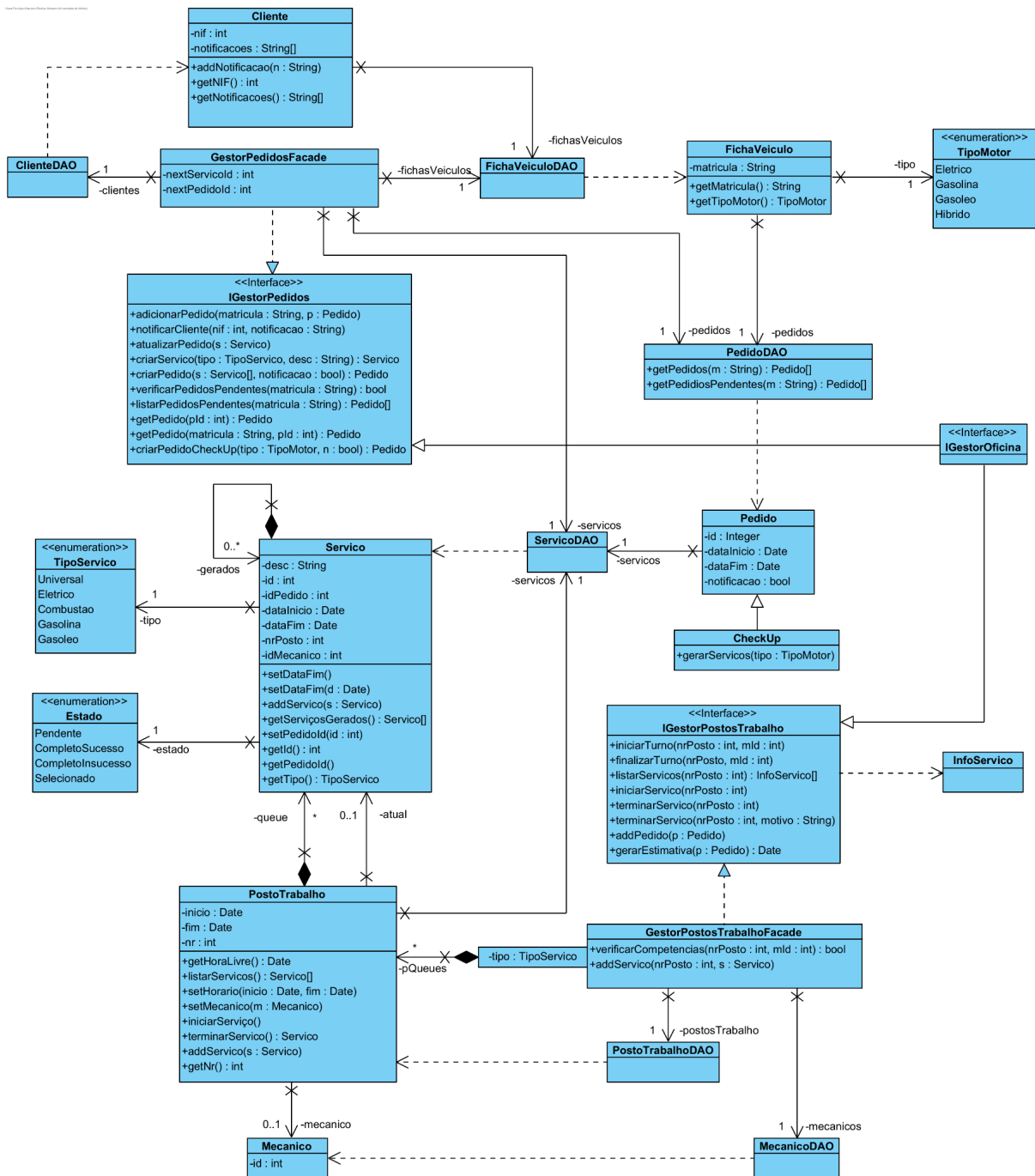
4. Diagrama de componentes

| <i>Caso de uso</i> | <i>Definição da API</i> | <i>Subsistemas</i> |
|---------------------------------|---|--|
| Adicionar serviços | adicionarPedido(matricula: String, pedido: Pedido) criarServico(tipo: TipoServico, desc: String): Servico criarPedido(s: Servico[], notificacao: bool): Pedido | GestorPedidos GestorPedidos GestorPedidos |
| Check-up | verificarPedidosPendentes(m: String): bool listarPedidosPendentes(m: String): Pedido[] criarPedidoCheckUp(s: Servico[], notificacao: bool): CheckUp gerarEstimativa(p: Pedido): Date | GestorPedidos GestorPedidos GestorPedidos GestorPostosTrabalho |
| Comunicar conclusão de serviços | notificarCliente(nif: int, notificacao: String) | GestorPedidos |
| Iniciar turno | iniciarTurno(nrPosto: int, mId: int) verificarCompetencias(nrPosto: int, mId: int) | GestorPostosTrabalho GestorPostosTrabalho |
| Finalizar turno | finalizarTurno(nrPosto, mId : int) | GestorPostosTrabalho |
| Realizar serviço | listarServicos(nrPosto: int): InfoServico[] iniciarServico(nrPosto: int) terminarServico(nrPosto: int, motivo: String): Servico termincarServico(nrPosto: int): Servico | GestorPostosTrabalho GestorPostosTrabalho GestorPostosTrabalho GestorPostosTrabalho |

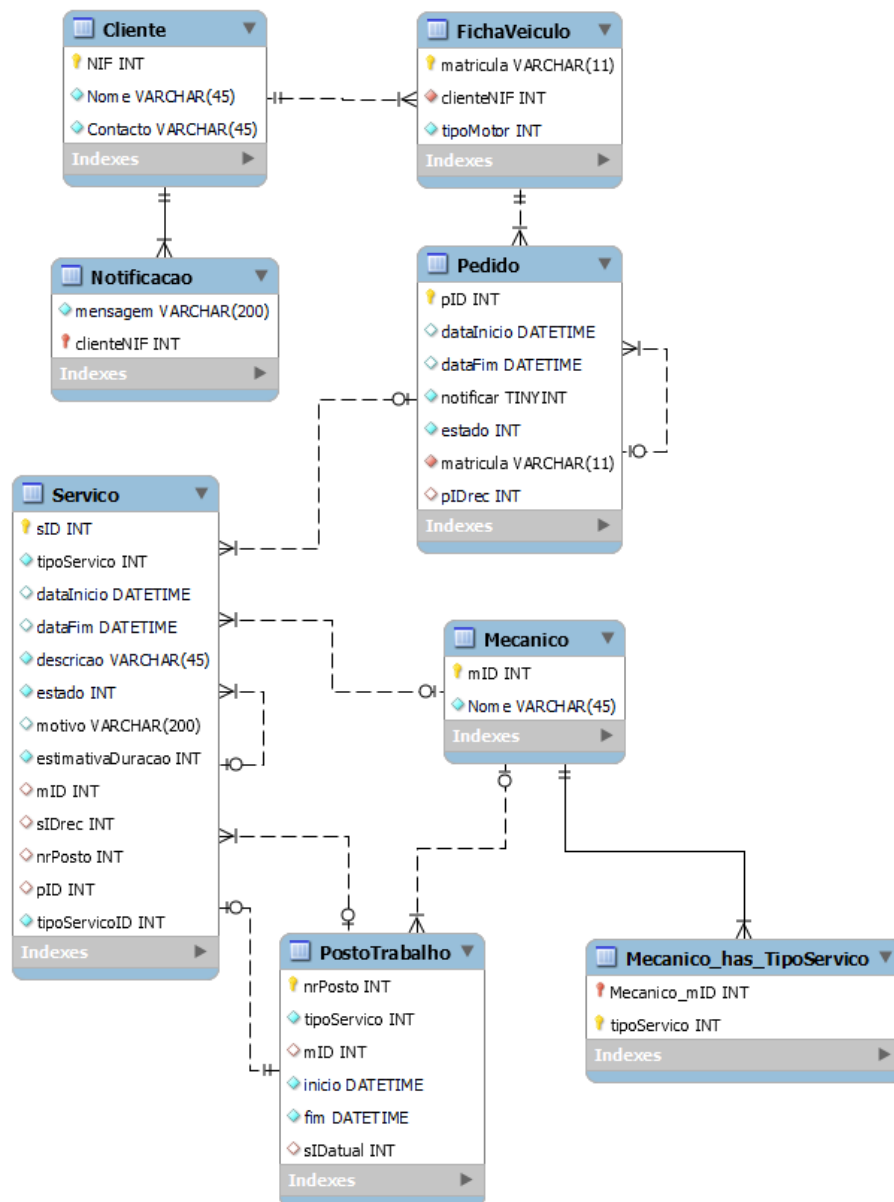
Nota: Clientes, mecânicos, e postos de trabalho já estarão presentes na base de dados.



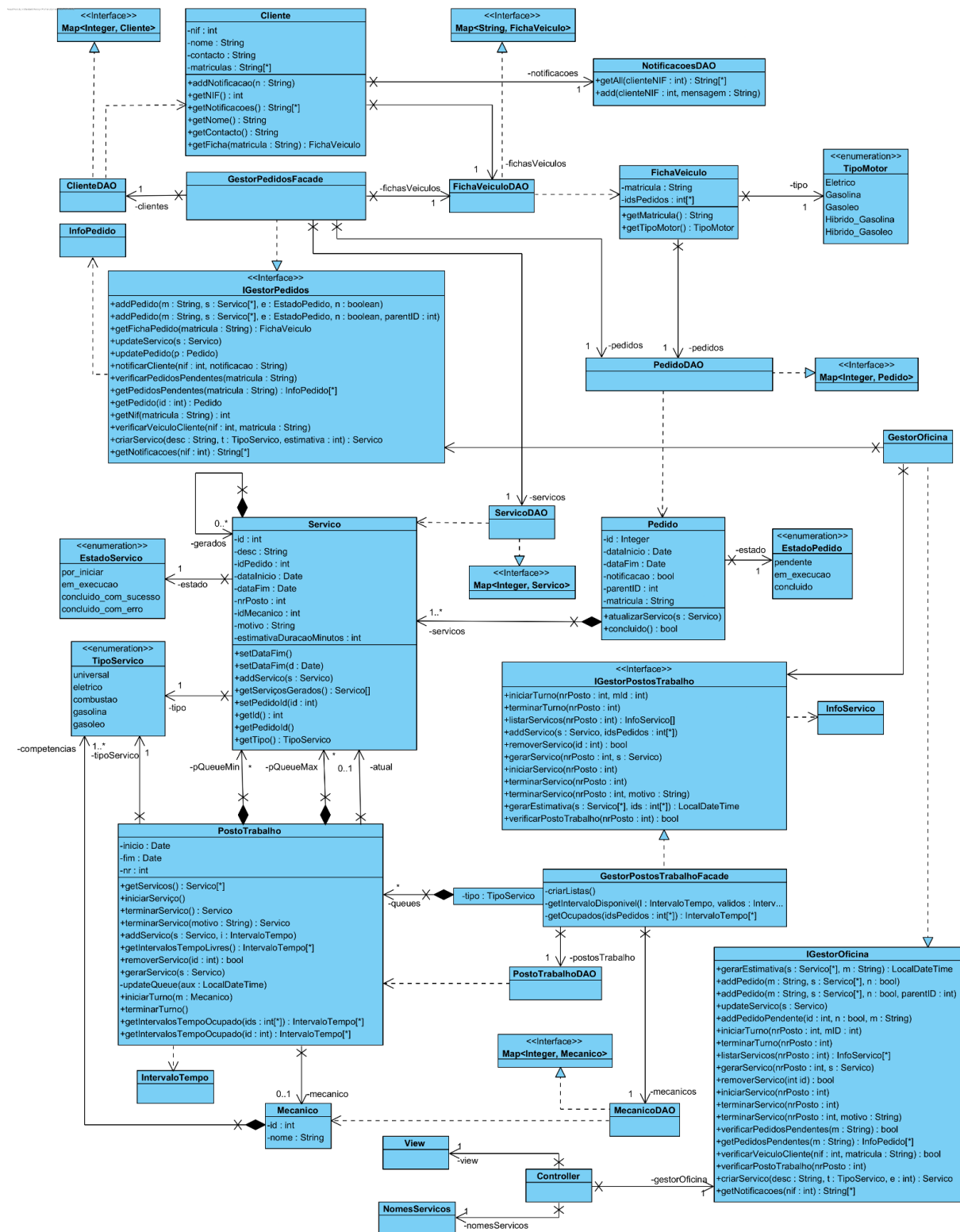




7. Modelo lógico da base de dados

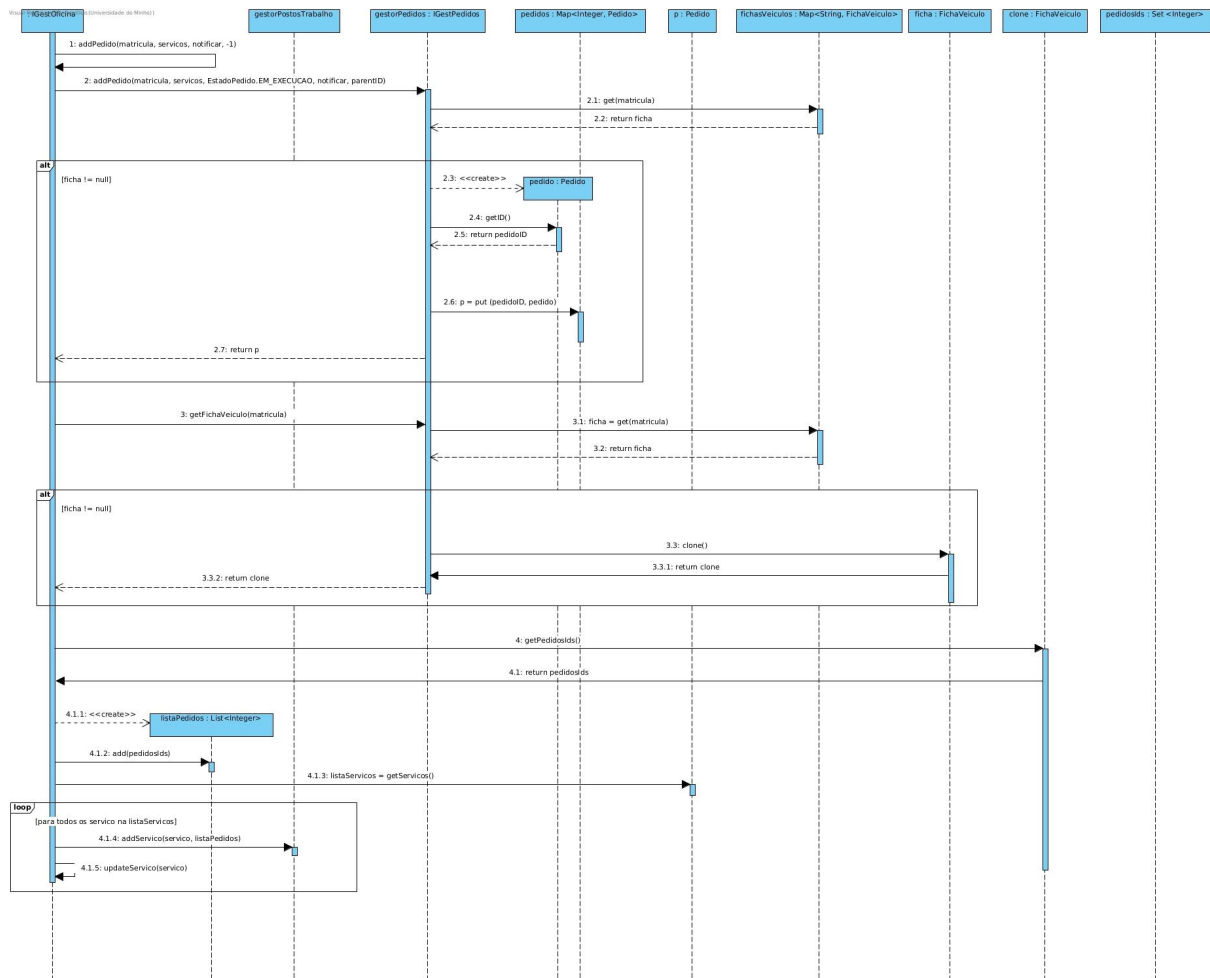


8. Diagrama de classes (implementação)



9. Diagramas de sequência (implementação)

função addPedido(String matricula, List<Servico> servicos, boolean notificar) da interface IGestorOficina:



função terminarServico(int nrPosto) da interface IGestorOficina:

UML Paradigm Standard (Universidade do Minho)

