**Faculdade de Engenharia da Universidade do Porto**

A black and white sign

Description automatically generated with low confidence

**Base de Dados: Rede de Transportes**

**Bases de Dados 2ºAno – Licenciatura em Engenharia Informática e Computação**

Professor Supervisor: Michel Celestino Paiva Ferreira

Estudantes:

José Maria Borges Pires do Couto e Castro - [up202006963@edu.fe.up.pt](mailto:up202006963@edu.fe.up.pt)

Miguel Lourenço Pregal de Mesquita Montes - [up202007516@edu.fe.up.pt](mailto:up202007516@edu.fe.up.pt)

Pedro Miguel Magalhães Nunes - [up202004714@edu.fe.up.pt](mailto:up202004714@edu.fe.up.pt)

Grupo 902 – Turma 9

**Índice**

[Contexto 3](#_Toc524428597)

[Diagrama UML 4](#_Toc524428598)

[Diagrama UML Revisto 5](#_Toc524428598)

[Modelo Relacional 6](#_Toc524428598)

[Análise de Dependências Funcionais e Formas Normais 7](#_Toc524428598)

[Restrições 9](#_Toc524428598)

[Interrogações 12](#_Toc524428598)

[Gatilhos 13](#_Toc524428598)

**Contexto**

A base de dados do serviço da STCP terá o objetivo de armazenar os diversos elementos de funcionamento da rede de transportes de autocarros.

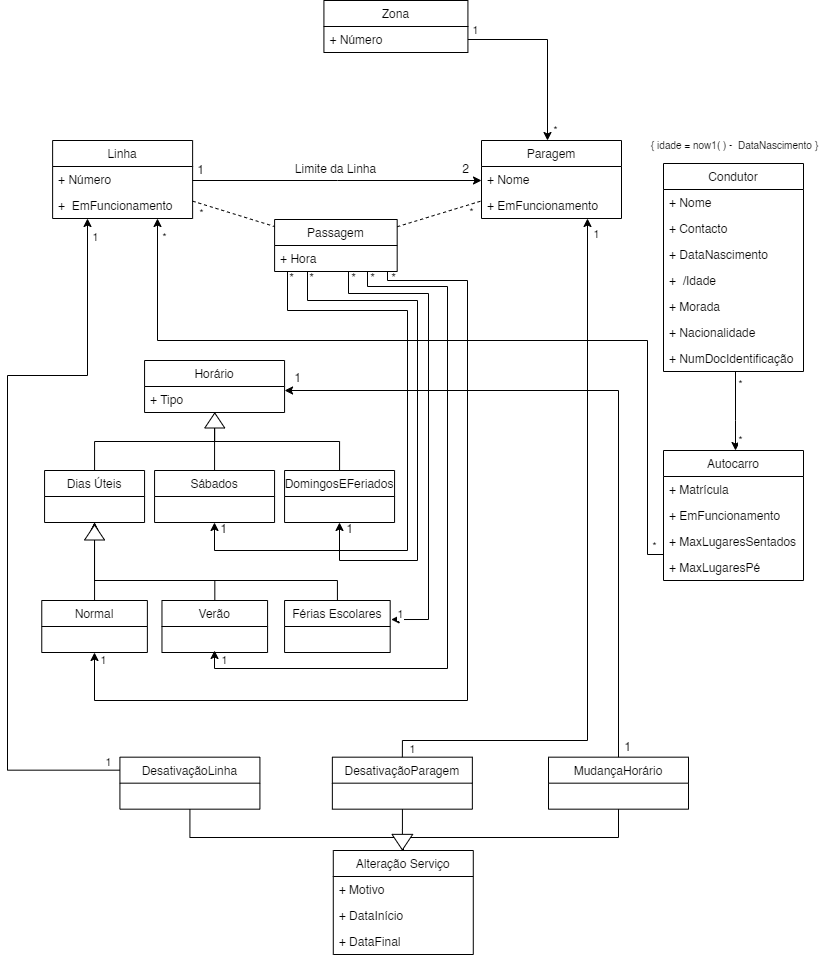
Será necessário organizar a base de dados a partir de zonas compostas por muitas paragens que, juntas, formam linhas que serão percorridas por diferentes autocarros. Cada autocarro pode ser chamado para percorrer qualquer uma das linhas, invertendo o sentido quando chega a um dos limites desta.

A cada autocarro estão associados muitos condutores, mas estes podem continuar em posto de trabalho sem estar, de momento, a conduzir. Os condutores, por si só, podem estar registados para conduzir muitos autocarros. Faria sentido, assim, especificar os autocarros que se encontram em funcionamento.

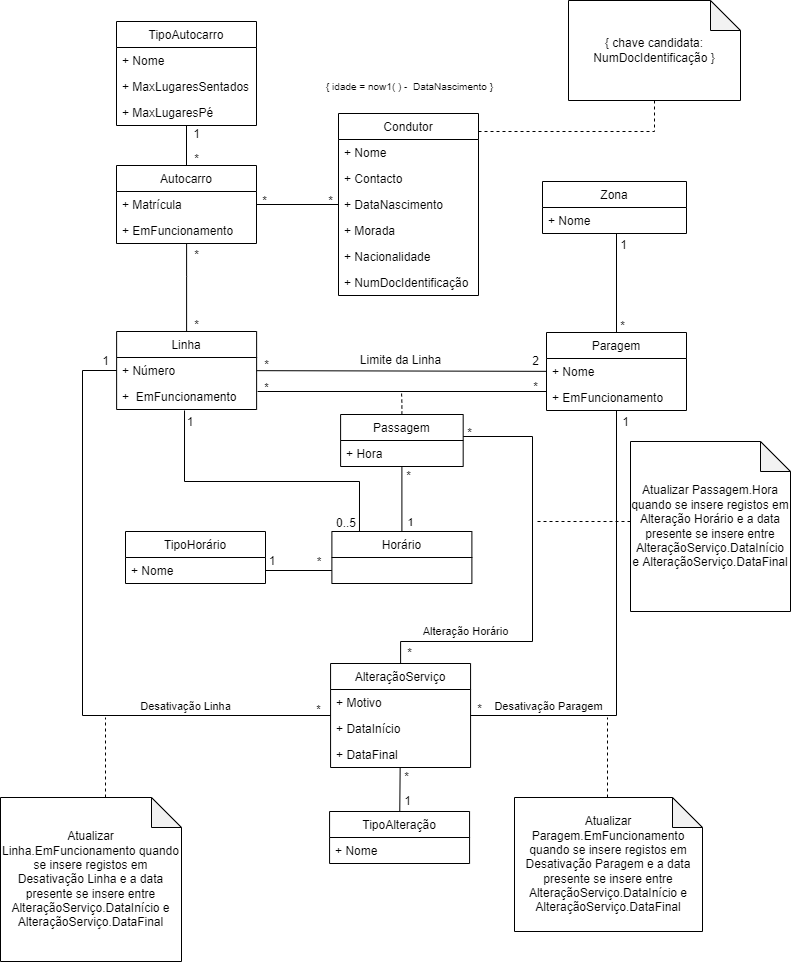
Em termos de horários é obrigatório diferenciar três tipos de horários diários: o dos Dias Úteis, o dos Sábados e o dos Domingos/Feriados. É importante diferenciar, dentro deste primeiro, três tipos de horários sazonais: o normal, o de verão e o de férias escolares. Os horários são compostos por passagens, que são instâncias de horas em que está programado um autocarro passar.

A qualquer momento, deverá ser permitido a alteração de serviço por fatores externos: isto deverá ser especificado através da referência do motivo da mudança de serviço, do tipo da mudança a ser efetuada (desativação de linha, desativação de paragem, mudança de horário) e o período de tempo durante a qual ficará em vigor.

**Diagrama UML**

****

**Diagrama UML Revisto**



**Modelo Relacional**

**Zona(idZona, nome)**

**Paragem(idParagem, nome, emFuncionamento, idZona->Zona)**

**Linha(idLinha, número, emFuncionamento, limitedalinha1->Paragem, limitedalinha2->Paragem)**

**Condutor(idCondutor, nome, contacto, dataNascimento, morada, nacionalidade, numDocIdentificação)**

**Autocarro(idAutocarro, matrícula, emFuncionamento, idTipoAutocarro->TipoAutocarro)**

**TipoAutocarro(idTipoAutocarro, nome, maxLugaresSentados, maxLugaresPé)**

**CondutorDoAutocarro(idAutocarro->Autocarro, idCondutor->Condutor)**

**LinhaDoAutocarro(idAutocarro->Autocarro, idLinha->Linha)**

**Passagem(idPassagem, idLinha->Linha, idParagem->Paragem, idHorário-> Horário, hora)**

**Horário(idHorário, idLinha->Linha, idTipo->TipoHorário)**

**TipoHorário(idTipoHorário, nome)**

**AlteraçãoServiço(idAlteraçãoServiço, motivo, dataInício, dataFinal, idTipoAlteração->TipoAlteração)**

**TipoAlteração(idTipoAlteração, nome)**

**AlteraçãoHorário(idAlteraçãoServiço, idPassagem->Passagem)**

**DesativaçãoLinha(idAlteraçãoServiço->AlteraçãoServiço, idLinha->Linha)**

**DesativaçãoParagem(idAlteraçãoServiço->AlteraçãoServiço, idParagem->Paragem)**

**Análise de Dependências Funcionais e Formas Normais**

**Zona:**

**{ idZona -> nome }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF.**

**Paragem:**

**{ idParagem } -> { nome, idZona }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF.**

**Linha:**

**{ idLinha } -> { número, limitedalinha1, limitedalinha2 }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF.**

**Condutor:**

**{ idCondutor } -> { nome, contacto, dataNascimento, morada, nacionalidade, numDocIdentificação }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF.**

**Autocarro:**

**{ idAutocarro } -> { matrícula, emFuncionamento }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF.**

**CondutorDoAutocarro:**

**Não existe nenhuma FD.**

**LinhaDoAutocarro:**

**Não existe nenhuma FD.**

**Passagem:**

**{ idPassagem } -> { hora }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF.**

**Horário:**

**{ idHorário } -> { idLinha, idTipo }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF.**

**TipoHorário:**

**{ idTipoHorário } -> { nome }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF**

**AlteraçãoServiço:**

**{ idAlteraçãoServiço } -> { motivo, dataInício, dataFinal }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF.**

**TipoAlteração:**

**{ idTipoAlteração } -> { nome }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF.**

**AlteraçãoHorário:**

**{ idAlteraçãoServiço } -> { idPassagem }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF.**

**DesativaçãoLinha:**

**{ idAlteraçãoServiço } -> { idLinha }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF.**

**DesativaçãoParagem:**

**{ idAlteraçãoServiço } -> { idParagem }**

**Não existe nenhuma FD que não contenha as chaves à esquerda.**

**Encontra-se na BCNF.**

**Restrições**

**Zona:**

**O ID da zona determinará sempre o resto dos atributos desta, pois não existem zonas repetidas (*PrimaryKey*).**

**Uma zona tem sempre um nome único que a distingue (*not-null* & *unique*).**

**Paragem:**

**O ID da paragem determinará sempre o resto dos atributos desta, pois não existem paragens repetidas (*PrimaryKey*).**

**Uma paragem tem sempre um nome único que a distingue (*not-null & unique*).**

**Uma paragem apenas pode estar inativa ou ativa (*attribute-based check & not-null*).**

**Cada paragem está ligada a uma zona obrigatoriamente (*not-null*).**

**Linha:**

**O ID da linha determinará sempre o resto dos atributos desta, pois não existem linhas repetidas (*PrimaryKey*).**

**Uma linha apresenta sempre um número que a identifica somente a ela (*not-null &* *unique*).**

**Tal como a paragem, uma linha tem de estar ativa ou inativa (*attribute-based check & not-null*).**

**Cada linha apresenta sempre duas paragens que servem de limites da linha: estas ditam o início e o começo do percurso (*not-null*).**

**Condutor:**

**O ID do condutor determinará sempre o nome deste, pois não existem condutores repetidos (*PrimaryKey*).**

**Cada condutor apresenta obrigatoriamente um nome próprio, uma data de nascimento, uma morada e uma nacionalidade (*not-null*).**

**Para maior identificação de cada condutor, estes dispõem também de contacto telefónico e número de documento de identificação (CC ou BI) (*not-null* & *unique*).**

**Autocarro:**

**O ID do autocarro determinará sempre o resto dos atributos deste, pois não existem autocarros repetidos (*PrimaryKey*).**

**Cada autocarro apresenta obrigatoriamente uma matrícula exclusiva, pois não existem matrículas repetidas (*not-null* & *unique*).**

**Um autocarro está sempre ou parado ou em funcionamento (*attribute-based check & not-null*) e corresponde obrigatoriamente a um tipo de autocarro (*not-null*).**

**CondutorDoAutocarro:**

**Um autocarro está sempre ligado aos condutores que o operam e um condutor está sempre ligado aos autocarros que este mesmo conduz (*PrimaryKey & not-null*).**

**LinhaDoAutocarro:**

**Um autocarro está sempre ligado às linhas que este percorre e uma linha está sempre ligada aos autocarros que a fazem (*PrimaryKey & not-null*).**

**Horário:**

**O ID do autocarro determinará sempre a linha e o tipo a que este se associa (*PrimaryKey*).**

**Um horário existe sempre em conjunto com uma linha e corresponde obrigatoriamente a um tipo de horário específico (*non-tull*).**

**TipoHorário:**

**O ID de cada tipo de horário infere sempre o nome deste (*PrimaryKey*), restringindo-se sem falha a 1 de 5 tipos: o de Dias Úteis, o de Sábados, o de Domingos e Feriados, o de Verão e, finalmente, o de Férias Escolares (*atribute-based check & not-null*).**

**Passagem:**

**A hora de cada passagem, bem como o horário, a linha e a paragem a que esta se associa é inferida a partir do ID de cada instância de passagem (*PrimaryKey*).**

**Não existe nenhuma passagem que não referencie a própria hora prescrita (*not-null*).**

**AlteraçãoServiço:**

**O ID de cada alteração de serviço automaticamente especifica o motivo da alteração e o período de tempo que esta abrange (*PrimaryKey*).**

**Cada alteração de serviço necessita de apresentar uma data de início e uma data de fim que ditam o tempo em que estarão em vigor e o tipo de alteração a ser feita (*not-null*).**

**TipoAlteração:**

**O ID de cada tipo de alteração de serviço automaticamente especifica o nome do tipo (*PrimaryKey*), sendo que este tipo se diferencia sempre entre alteração numa Linha, numa Paragem ou num Horário (*attribute-based check & not-null*).**

**AlteraçãoHorário:**

**O ID de cada alteração de serviço é necessária para saber que passagem é afetada pela alteração num caso de alteração de horário (*PrimaryKey & not-null*). Não existe nenhuma alteração deste tipo que não se associe a alguma passagem (*not-null*).**

**DesativaçãoLinha:**

**O ID de cada alteração de serviço é necessária para saber que linha é afetada pela alteração num caso de desativação de linha (*PrimaryKey & not-null*). Não existe nenhuma alteração deste tipo que não se associe a alguma linha (*not-null*).**

**DesativaçãoParagem:**

**O ID de cada alteração de serviço é necessária para saber que linha é afetada pela alteração num caso de desativação de paragem (*PrimaryKey & not-null*). Não existe nenhuma alteração deste tipo que não se associe a alguma paragem (*not-null*).**

**Interrogações**

**1 – Nome de todos os condutores, ordenados alfabeticamente**

**2 – Matrículas dos autocarros**

**3 – Nome das paragens pertencentes a cada zona**

**4 – Linhas que percorrem paragens de uma zona, e quantas paragens dessa zona percorrem atualmente (baseado nas passagens)**

**5 – Passagens das paragens pertencente à linha 202 em horário de dias úteis (horário da linha 202)**

**6 –**

**7 – Nome dos condutores que não conduzem nenhum autocarro do tipo “Articulado”**

**8 – Linhas afetadas por uma alteração de serviço do tipo “DesativaçãoLinha”**

**9 – Primeira e última passagem em cada paragem**

**10 – Linhas chamadas circulares (limites da linha iguais)**

**Gatilhos**

**Os três gatilhos servem a mesma função de efetivar as alterações de serviço inseridas na base de dados. Cada um destes dispara dependendo se a data do dia em que se insere uma alteração de serviço (desativação de linha, desativação de paragem, alteração de horário) se encontra dentro do período de duração da própria alteração.**

**Para o primeiro e segundo gatilho, o seu disparo apenas muda o estado da linha/paragem a ser desativada: transforma o atributo ‘EmFuncionamento’ de 1 para 0.**

**No caso do terceiro, o seu disparo muda a hora da passagem afetada pela alteração de serviço, adicionando 20 minutos à hora previamente inserida.**