



Universidade do Minho

Licenciatura em Engenharia informática

Mestrado Integrado em Engenharia Informática

Unidade Curricular de Laboratórios de Informática IV

Ano Letivo 2021/2022

GoQuick

Daniel José Gonçalves Coutinho de Faria – a81667

Joana Maia Teixeira Alves – a93290

Maria Eugénia Bessa Cunha – a93264

Vicente Gonçalves Moreira – a93296

Novembro - 2021

LI4

Data de Receção	
Responsável	
Avaliação	
Observações	

GoQuick

Daniel José Gonçalves Coutinho de Faria – A81667

Joana Maia Teixeira Alves – A93290

Maria Eugénia Bessa Cunha – A93264

Vicente Gonçalves Moreira – A93296

Novembro - 2021

Resumo

Este relatório foi elaborado no âmbito do desenvolvimento de um *software* com o tema “guia de locais de interesse”, ou seja, uma aplicação que apresente os vários locais de interesse nas redondezas de uma dada localização.

Este projeto foi dividido em duas fases, na primeira, a equipa elabora todos os aspetos iniciais que foram tomados em consideração em relação à temática, assim como a definição dos requisitos, planeamentos e recursos necessários para o seu desenvolvimento. Na segunda fase, que está além do limiar deste relatório, irá elaborar a implementação do plano desenvolvido.

Para este projeto a equipa começou por diminuir a dimensão da aplicação, focando o seu uso numa cidade específica para a apresentação de locais, nomeadamente Braga, dado à familiaridade da equipa com esta cidade. Adicionalmente, optamos por modificar o comportamento da aplicação, de forma que apenas apresente as paragens de autocarro em redor do utilizador. Estas decisões foram feitas de modo a obter um produto final funcional e completo. Assim, os objetivos deste projeto tornar-se-iam exequíveis.

Neste relatório iremos apresentar as várias decisões, conceitos, planeamentos, requisitos e discussões que surgiram no âmbito do desenvolvimento da primeira parte deste projeto.

Área de Aplicação: Serviços de Transporte.

Palavras-Chave: Braga, Paragem, Transporte, Requisitos, Aplicação, Interface, API, Base de dados.

Índice

RESUMO	3
ÍNDICE DE FIGURAS	6
ÍNDICE DE TABELAS	7
1. INTRODUÇÃO	8
1.1. CONTEXTUALIZAÇÃO	8
1.2. MOTIVAÇÃO E OBJETIVOS	9
1.3. ESTRUTURA DO RELATÓRIO	10
1.4. JUSTIFICAÇÃO E UTILIDADE DO SISTEMA	11
1.5. ESTABELECIMENTO DA IDENTIDADE DO PROJETO	11
1.6. IDENTIFICAÇÃO DOS RECURSOS NECESSÁRIOS	12
1.7. MAQUETA DO SISTEMA	13
1.8. DEFINIÇÃO DE UM CONJUNTO DE MEDIDAS DE SUCESSO	14
1.9. DIVULGAÇÃO DA APLICAÇÃO	14
1.10. PLANO DE DESENVOLVIMENTO	15
2. LEVANTAMENTO E ANÁLISE DE REQUISITOS	16
2.1. APRESENTAÇÃO DA ESTRATÉGIA E MÉTODO	16
2.2. DESCRIÇÃO GERAL DOS REQUISITOS LEVANTADOS	18
2.3. REQUISITOS DO SISTEMA E A SUA BASE DE DADOS	18
2.3.1. <i>Requisitos das Paragens</i>	18
2.3.2. <i>Requisitos das Rotas</i>	19
2.3.3. <i>Requisitos da Base de Dados</i>	19
2.3.4. <i>Requisitos de Gestão da Aplicação</i>	19
2.3.5. <i>Requisitos de Rede</i>	19
2.4. REQUISITOS DO UTILIZADOR	19
2.5. VALIDAÇÃO DOS REQUISITOS ESTABELECIDOS	20
3. ESPECIFICAÇÃO E MODELAÇÃO DO SOFTWARE	21
3.1. APRESENTAÇÃO GERAL DA ESPECIFICAÇÃO	21
3.2. ASPETOS COMPORTAMENTAIS	22
3.2.1. <i>Modelo Use Case</i>	23
3.2.2. <i>Fluxos</i>	24
3.3. ASPETOS ESTRUTURAIS	25
3.3.1. <i>Interface da Camada de Apresentação</i>	26
3.3.2. <i>Interface da Camada de Negócio</i>	27
3.3.3. <i>Classes de Armazenamento de Dados em Memória</i>	28
4. CONCEÇÃO DO SISTEMA DE DADOS	29
4.1. APRESENTAÇÃO GERAL DA ESTRUTURA DO SISTEMA DE DADOS	29
4.1.1. <i>Modelo Conceptual</i>	30
4.1.2. <i>Modelo Lógico</i>	30
4.2. DESCRIÇÃO DOS ELEMENTOS DE DADOS E OS SEUS RELACIONAMENTOS	31
4.2.1. <i>Rota</i>	31
4.2.2. <i>Paragem_has_Rota</i>	31
4.2.3. <i>Paragem</i>	32
4.2.4. <i>Paragem_has_Hora</i>	32

4.2.5.	<i>Hora</i>	32
5.	ESBOÇO DAS INTERFACES GRÁFICAS DO SISTEMA	33
5.1.	ESTRUTURA GERAL DAS INTERFACES DO SISTEMA	33
5.2.	CARACTERIZAÇÃO DAS INTERFACES	34
6.	CONCLUSÕES E TRABALHO FUTURO	35
6.1.	APRECIAÇÃO CRÍTICA.....	35
6.2.	TRABALHO FUTURO	35
6.3.	CONCLUSÃO.....	35
7.	REFERÊNCIAS	36
8.	SIGLAS	37

Índice de Figuras

Figura 1 – Intensidade carbónica nos transportes	9
Figura 2 – Identidade do Projeto	11
Figura 3 – Maqueta de Utilização	13
Figura 4 – Maqueta da Base de dados	13
Figura 5 – Plano de Desenvolvimento do Projeto	15
Figura 6 – Grau de satisfação na divulgação de horários	16
Figura 7 – Interesse no desenvolvimento da aplicação	17
Figura 8 – Grau de interesse de uma aplicação de divulgação de horários	17
Figura 9 – Modelo de Domínio	22
Figura 10 – Modelo de Use Case	23
Figura 11 - Fluxos do Use Case “Obter o mapa da cidade com paragens”	24
Figura 12 – Fluxos do Use Case “Obter informação da paragem”	25
Figura 13 – Interface IVisualGoQuick	26
Figura 14 – Interface IGestGoQuick	27
Figura 15 – Diagrama de Classes	28
Figura 16 – Modelo Conceptual da BD	30
Figura 17 – Modelo lógico da BD	30
Figura 18 – Esboço das interfaces gráficas	33
Figura 19 – UI Menu Principal	34
Figura 20 – UI Recolha de Endereço	34
Figura 21 – UI Apresentação do Mapa	34
Figura 22 – UI Informação de uma paragem	34

Índice de Tabelas

Table 1 - Tabela Rota do Esquema Lógico	31
Table 2 - Tabela Paragem_has_Rota do Esquema Lógico	31
Table 3 - Tabela Paragem do Esquema Lógico	32
Table 4 - Tabela Paragem_has_Rota do Esquema Lógico	32
Table 5 - Tabela Hora do Esquema Lógico	32

1.Introdução

Este relatório tem como objetivo apresentar as várias fases de conceção, planeamento e desenvolvimento, assim como requerimentos e objetivos de uma aplicação focada no auxílio no uso de transportes públicos.

Para esta aplicação, decidimo-nos focar em aspetos que dizem respeito a transporte por autocarro, em específico, o mapeamento e disponibilização de informação acerca de paragens de autocarro.

Dado à relativa inexperiência da equipa no desenvolvimento de aplicações a serem usadas num ambiente real, foi decidido realizar um projeto de pequena escala, resultando assim, numa aplicação a ser utilizada por habitantes e visitantes de uma determinada cidade. A equipa, para este projeto, escolheu a cidade de Braga, em Portugal, dado à familiaridade dos membros com a mesma e à relativa abundância e uso dos transportes públicos na cidade.

De seguida, apresentamos aspetos de forma a justificar a decisão da equipa acerca do tema e utilidade final da aplicação, como também as razões e estudos que nos levaram a escolher a cidade de Braga. Sendo estes acompanhados, também, por considerações iniciais, objetivos, motivação, especificação e custos do projeto.

1.1. Contextualização

A cidade de Braga observou um grande crescimento populacional e urbanístico nos últimos anos. De acordo com o documento “CENSOS2021”, registou-se que na freguesia de Gualtar, houve um acréscimo populacional de 27,7% em relação a dados de 2011. Este crescimento levou a um grande aumento da procura de soluções de mobilidade urbana, com uma especial preocupação em aumentar a fluidez e facilidade no transporte da população.

Outro problema que surge desta rápida sobrepopulação é o aumento do uso de transportes individuais. No distrito de Braga verificou-se um aumento da sinistralidade na estrada, tendo aumentado cerca de 25,4% em relação ao ano passado, constituindo 10,3% da mortalidade na estrada em Portugal. Segundo o relatório de Sinistralidade e Fiscalização do mês de agosto de 2021, realizado pela Autoridade Nacional de Segurança Rodoviária, o veículo interveniente em 70,4% dos acidentes é um automóvel ligeiro.

Existe, também, uma vertente ecológica associada a este aumento de população, pois haverá uma maior utilização de recursos materiais e energéticos. Esta maior utilização é refletida no aumento do uso de transportes individuais.

A última edição do relatório do estado do ambiente realizado pela Agência Portuguesa do Ambiente (APA) aponta que o setor do transporte é responsável por uma grande parte das emissões de poluentes, como o dióxido de azoto e as partículas inaláveis, que contaminam o ar que respiramos, bem como de gases com efeito de estufa, indissociáveis das alterações climáticas. Este relatório também mostra como a intensidade carbónica nos transportes tem gradualmente aumentado em Portugal.

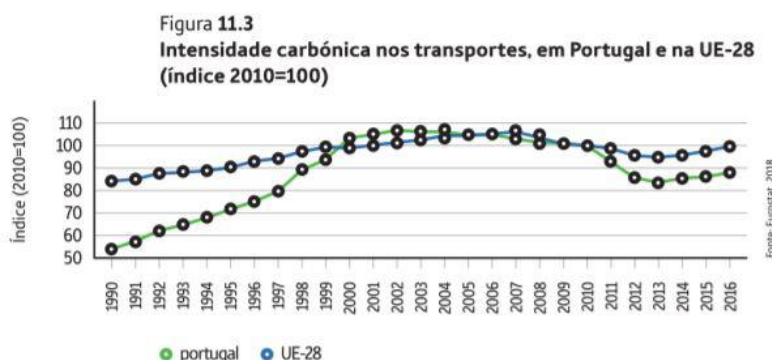


Figura 1 – Intensidade carbónica nos transportes

1.2. Motivação e Objetivos

Motivados a mitigar a sinistralidade, assim como proporcionar um meio ambiente mais ecológico e saudável, procuramos por estudos com possíveis soluções.

No estudo relacionado com transportes públicos “Public Transport Quality, Safety, and Perceived Accessibility” realizado pelo “CTF Service Research Center and Department of Social and Psychological Studies” da Universidade de *Karlstad*, na Suécia, aponta que um desenvolvimento sustentável de um bom sistema de transporte público está relacionado diretamente com a segurança e acessibilidade no transporte da população.

Não só isso, mas como um estudo realizado na cidade de *Medan*, na Indonésia, verificou que com um sistema de transporte público desenvolvido e fiável, a felicidade e a qualidade de vida tende a aumentar entre os seus utentes.

Já relacionado com o ambiente, o artigo “Why Transit Matters: The Environmental Benefits of Public Transportation” da ACOG (Association of Central Oklahoma Governments) indica várias vantagens no uso de transportes públicos em

relação a transportes individuais, como uma maior eficiência no uso de combustível, maior densidade no transporte de passageiros e uma redução na poluição do ar.

Face a estas estatísticas recolhidas, assim como os estudos com potenciais soluções, a equipa tem como objetivo incentivar e facilitar a utilização de transportes públicos de forma a reduzir o número de veículos particulares em circulação, alcançando assim um transporte mais seguro e confortável da população na cidade.

Para alcançar este objetivo a equipa decidiu desenvolver uma aplicação informática, a ser implementada num ambiente móvel, que comunique aos seus utilizadores a localização de várias paragens de autocarro ao seu redor, assim como fornecer informações críticas acerca desta como o seu horário.

1.3. Estrutura do Relatório

Este relatório é composto por **6 capítulos**.

No primeiro capítulo, **introduziram-se** as razões e necessidades para o desenvolvimento da aplicação, assim como a descrição de conceitos iniciais, recursos necessários e o plano de desenvolvimento.

No segundo capítulo, elaboraram-se os **requisitos** da aplicação, explicando a metodologia de análise de requisitos e a definição das funcionalidades pretendidas que a aplicação terá de ser capaz de cumprir.

No terceiro capítulo, produziu-se a **especificação** necessária para esta aplicação, exibindo vários modelos com o intuito de auxiliar na interpretação do problema, como também a definição das *API's* que a equipa acredita serem fundamentais.

No quarto capítulo, apresenta-se informação relativa à **base de dados**, como o seu modelo conceptual e lógico, assim como as suas especificações.

No quinto capítulo, exibem-se esboços da **interface gráfica** da aplicação, com alguns exemplos que acreditamos serem relevantes.

O último capítulo, **conclui** o relatório, acrescentando a apreciação crítica e a prospeção do trabalho futuro da equipa.

1.4. Justificação e Utilidade do Sistema

Um sistema que auxilie na utilização de transportes públicos é extremamente útil para incentivar os clientes a procurarem este modo de transporte. Entre um *software* móvel que lista e identifica num mapa todas as paragens e as suas informações de forma rápida e precisa, dada a localização do utente, ou uma pesquisa *web*, sem localização, de rotas e horários de cada autocarro que são consultados individualmente e lentamente, a escolha de um *software* rápido e centralizado é mais apelativa e vantajosa.

O programa a desenvolver será de grande utilidade no que diz respeito a facilitar a deteção e a visualização espacial de paragens (por zona ou por proximidade) e relativos horários tornando mais apelativa a utilização de transportes públicos na cidade.

1.5. Estabelecimento da Identidade do Projeto



Figura 2 – Identidade do Projeto

1.6. Identificação dos Recursos Necessários

Para este projeto, em termos de recursos humanos, será necessária uma equipa constituída por elementos especializados em engenharia de *software* (encarregues do planeamento do projeto), programação (construção do código da aplicação), *design* gráfico (desenvolvimento do aspeto das interfaces da aplicação) e *marketing* (responsável pela publicidade da aplicação). Resumindo, a mão-de-obra mínima necessária para este projeto será: um engenheiro de *software*, um programador, um *designer* gráfico e, por fim, um profissional de *marketing*.

Em recursos materiais, a equipa necessitará de vários *softwares* que permitam o planeamento, programação e teste da aplicação a ser desenvolvida.

Para este projeto, dado à sua pequena dimensão, necessitamos de um acesso à recolha de informação de bases de dados já existentes, tratando esta informação e inserindo-a numa base de dados própria. Para a recolha desta informação de terceiros decidiu recorrer-se à base de dados sobre locais disponibilizada através da *API* da *Google*. Optou-se por esta *API* dado a sua facilidade de implementação e pela extensividade dos dados fornecidos, visto que a empresa *Google* é uma das líderes de mercado na área de base de dados.

Também será necessário um sistema que permita recolher coordenadas *GPS* através de um endereço local. Esta necessidade proveio da falta de soluções satisfatórias em relação às pesquisas de paragens por endereços, optando assim por recorrer a um sistema externo de forma a traduzir o endereço em coordenadas, que depois serão interpretadas pela aplicação. Para este sistema exploramos novamente as *API*'s fornecidas pela *Google*, encontrando assim uma solução.

1.7. Maqueta do Sistema



Figura 3 – Maqueta de Utilização

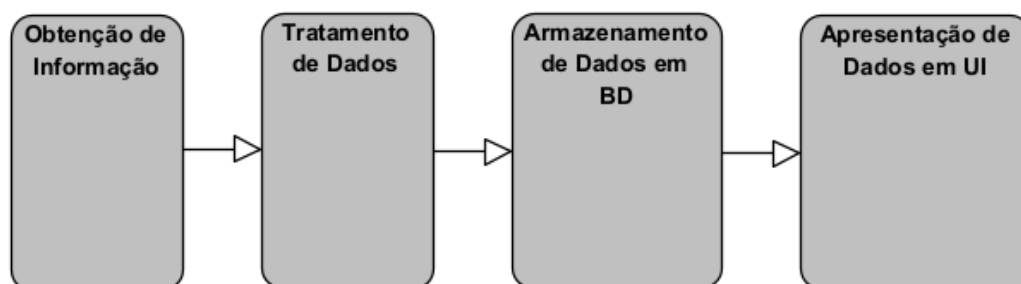


Figura 4 – Maqueta da Base de dados

Obtenção de Informação – a obtenção de informação deve ser feita com base na *API* da *Google Maps* que permita obter informações sobre a localização das paragens de autocarro.

Tratamento de Dados – o tratamento de dados visa filtrar a informação útil e organizar da forma mais adequada conforme for estabelecido.

Armazenamento de Dados em BD – uma vez tratada a informação esta deve ser armazenada numa base de dados

Apresentação de Dados em UI – A partir da base de dados existente, apresentamos ao utilizador a informação referente ao que ele desejar.

1.8. Definição de um Conjunto de Medidas de Sucesso

Para o conjunto de medidas de sucesso, a equipa está focada em obter um número de utilizadores ativos mensais suficiente para que haja um impacto positivo observável na cidade onde o projeto se realiza. Definiu-se então um valor mínimo de utilizadores correspondente a cerca de 20% da população de Braga (cerca de 27 mil pessoas) como a principal medida do sucesso do projeto.

Definimos como objetivo secundário a divulgação da aplicação por parte das empresas (públicas e privadas) encarregues da gestão e manutenção dos transportes públicos da cidade pois traria uma maior credibilidade à mesma.

1.9. Divulgação da Aplicação

Para atingir estas medidas de sucesso a equipa está empenhada em divulgar e publicitar a aplicação nos locais mais efetivos, ou seja, nas paragens de autocarro e centrais pois estes são os locais onde se situa a maior densidade de possíveis utilizadores da aplicação.

Outro local para a divulgação da aplicação seria por meios *online*, de forma a incentivar novos utilizadores que não são familiares a este meio de transporte. A equipa tenciona apostar em publicidade entre faixas etárias mais jovens através de redes sociais, escolas e universidades.

1.10.Plano de Desenvolvimento

Com os vários objetivos, conceitos e recursos reunidos, criou-se um plano de desenvolvimento, com a intenção de estimar a duração de cada fase deste projeto. Decidiu-se criar um diagrama de desenvolvimento *Gantt*, pois é um dos formatos mais reconhecidos no mercado.

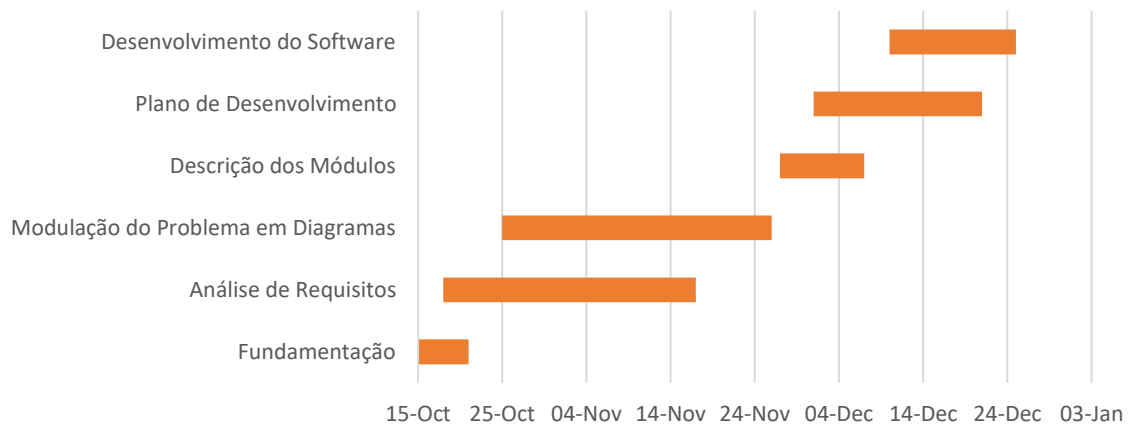


Figura 5 – Plano de Desenvolvimento do Projeto

2. Levantamento e Análise de Requisitos

2.1. Apresentação da Estratégia e Método

Para obtermos uma noção dos requisitos necessários, a equipa decidiu recolher opiniões via entrevistas e questionários à população de Braga.

As entrevistas foram realizadas de modo casual, inquirindo os transeuntes nas principais ruas de Braga. Nestas questionamos o entrevistado acerca do seu uso dos transportes utilizados na cidade, assim como perguntas relativas às razões que levam as pessoas a não optarem por transportes públicos como autocarros. Por fim, pedimos sugestões para melhorar o sistema de divulgação dos horários dos transportes.

O questionário realizado contém várias perguntas que visam recolher estatísticas acerca do uso de transportes públicos dos cidadãos, assim como níveis de satisfação e vários aspetos a melhorar de forma a incentivar o uso destes transportes, em específico, os autocarros. Este questionário foi partilhado via redes sociais, através de grupos e associações residentes na cidade de Braga.

Apresentamos aqui algumas das estatísticas que foram recolhidas através deste questionário:

Qual o grau de satisfação com a divulgação de horários e rotas de autocarros por parte das transportadoras?

89 responses

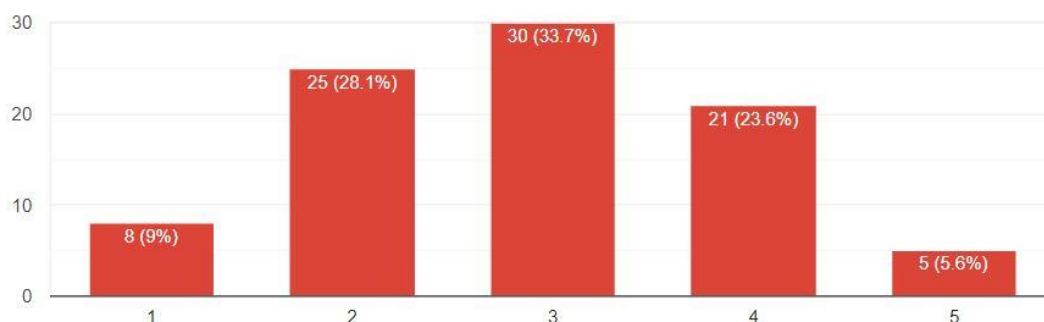


Figura 6 – Grau de satisfação na divulgação de horários

Com esta pergunta a equipa estava interessada em perceber melhor os níveis de satisfação dos cidadãos sobre as soluções já existentes acerca das informações divulgadas sobre horários e rotas dos autocarros.

Acharias interessante uma aplicação que te informasse das paragens de autocarro perto de ti?

91 responses

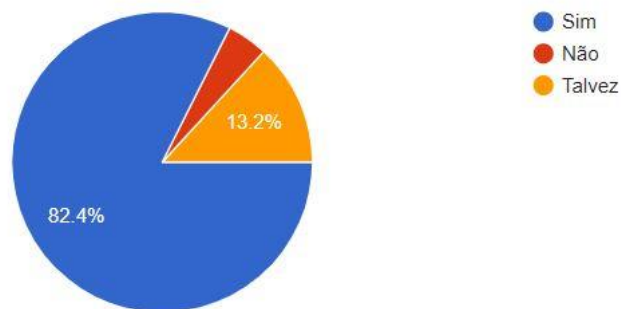


Figura 7 – Interesse no desenvolvimento da aplicação

Quão útil classificarias uma aplicação que te mostrasse as paragens de autocarro perto de ti, bem como os autocarros que por elas passam e as rotas que executam?

90 responses

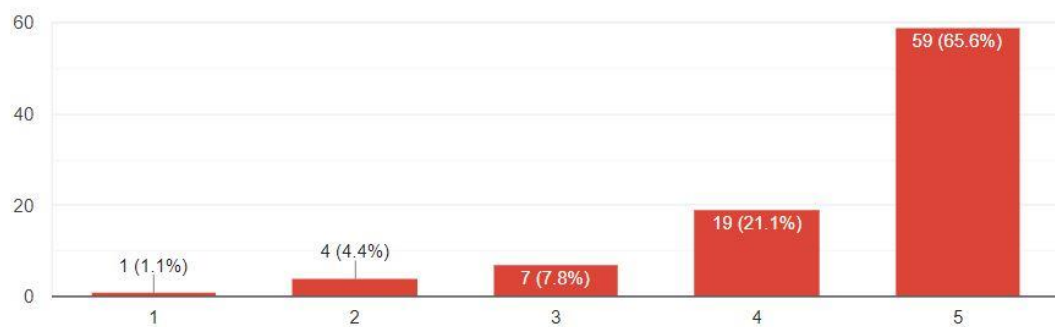


Figura 8 – Grau de interesse de uma aplicação de divulgação de horários

Com estas respostas a equipa desenvolveu uma melhor noção do potencial no mercado para uma aplicação que seja capaz de divulgar de forma eficaz informações relativas aos autocarros da cidade de Braga.

2.2. Descrição Geral dos Requisitos Levantados

Depois de se efetuarem várias entrevistas e recolherem as opiniões de alguns cidadãos de Braga, a equipa reuniu-se para discutir as várias capacidades e funcionalidades que a aplicação deveria cumprir, tendo em conta vários aspetos como requisitos de maior prioridade e a dimensão reduzida da aplicação.

Antes de elaborar, relembra-se que o primeiro requisito já definido anteriormente neste relatório indica que esta aplicação tem de funcionar num ambiente móvel para que a sua utilidade seja eficaz. Depois de definir este requisito fundamental, procedemos ao desenvolvimento dos restantes requisitos.

Dividiram-se estes requisitos em duas categorias, sendo uma destas os requisitos do sistema, delimitando as funcionalidades do *backend* da aplicação, ou seja, a recolha, tratamento e organização da informação. A outra categoria, requisitos do utilizador, é focada no *frontend* da aplicação, sendo esta responsável pela apresentação dos dados, ou seja, delimita as capacidades da interface de apresentação e interação com o utilizador.

2.3. Requisitos do Sistema e a sua Base de Dados

2.3.1. Requisitos das Paragens

- As paragens têm de ser identificáveis por uma chave única.
- As paragens têm que ter um nome associado.
- A paragem contém uma imagem para melhor representação da sua aparência e local.
- As paragens podem conter uma breve descrição para detalhes extra.
- As paragens têm que ter uma localização associada por *GPS*, ou seja, pelas coordenadas de latitude e longitude, e pelo endereço/rua.
- As paragens têm que conter uma lista de identificadores das rotas que por elas passam.
- As paragens têm que conter uma tabela de associação entre rotas e um horário, ou seja, uma lista de horas à qual dada rota passa na paragem identificada.

2.3.2.Requisitos das Rotas

- As rotas têm que ter um número identificador da rota.
- As rotas têm que ter uma lista de paragens que intersejam.

2.3.3.Requisitos da Base de Dados

- A base de dados tem de recolher a informação necessária de todas as paragens da cidade de Braga, através da *API* de locais da *Google*.
- A base de dados tem que ser capaz de comunicar com a aplicação.
- A base de dados tem de ser capaz de devolver a informação de todas as paragens num raio de 300 metros de uma dada localização *GPS*, fornecida pela aplicação.

2.3.4.Requisitos de Gestão da Aplicação

- A aplicação tem que ser capaz de processar e armazenar em memória a informação das paragens numa dada área geográfica definida, sendo esta informação fornecida pela base de dados através de uma *query*.
- A aplicação tem que ser capaz de devolver a informação de uma paragem dado o seu identificador.
- A aplicação tem que ser capaz de localizar as paragens geograficamente com assistência de um mapa.

2.3.5.Requisitos de Rede

- A aplicação tem de ser capaz de comunicar com a base de dados disponível *online*.
- A aplicação tem de ser capaz de obter a localização geográfica do dispositivo.
- A aplicação tem de ser capaz de obter a localização geográfica através de um endereço.

2.4. Requisitos do Utilizador

- O utilizador tem de ser capaz de visualizar as paragens num mapa fornecido.
- O utilizador tem de ser capaz de introduzir um endereço ou nome da rua para efetuar uma procura.

- O utilizador tem de ser capaz de fornecer a sua localização geográfica de forma automática.
- O utilizador tem de ser capaz de selecionar rapidamente, ou seja, sem utilização de escrita, uma paragem de autocarro.
- O utilizador, depois de selecionar uma paragem, tem de ser capaz de visualizar várias informações básicas como o nome da paragem, a sua descrição e imagem, assim como uma lista das rotas que intersejam nessa paragem.
- O utilizador tem de ser capaz de analisar, numa dada paragem, as horas a que várias rotas irão intersejar essa paragem.

2.5. Validação dos requisitos estabelecidos

Para confirmar os requisitos alcançados pela equipa, realizamos outras entrevistas entre os cidadãos de Braga para assegurar a praticabilidade e utilidade da aplicação.

Depois de efetuarmos algumas alterações nos requisitos, em específico, requisitos relativos à apresentação dos dados aos utilizadores e às informações contidas nas paragens, a equipa está confiante que possui todos os requisitos necessários que a aplicação a ser desenvolvida terá de ser capaz de efetuar.

3.Especificação e Modelação do Software

Com os requisitos estabelecidos, procedeu-se ao desenvolvimento dos primeiros conceitos para o funcionamento da aplicação. Para criar os vários modelos necessários à exposição do funcionamento da aplicação, recorreu-se à ferramenta de *software Visual Paradigm*. Esta tem bom suporte e permite de forma simples expor vários géneros de modelos do interesse da equipa, sendo muito intuitivo e eficaz.

3.1. Apresentação Geral da Especificação

Para desenvolver uma ideia inicial mais conceptual, optamos por desenvolver um modelo de Domínio, onde referimos as várias entidades presentes no nosso sistema, assim como algumas interações que estas irão realizar. Com este modelo não só desenvolvemos uma melhor ideia de comportamentos e interfaces a serem realizadas mais tarde, como serve também de fundação para o planeamento e desenvolvimento da futura base de dados.

Começou-se este modelo a partir da entidade Utente, sendo a entidade utilizadora da aplicação. Visto que esta aplicação será dependente da localização do dispositivo, acrescentou-se esta entidade e a sua relação com o utente, pois este será o responsável por fornecê-la da forma que achar mais adequado. Acrescentou-se, de seguida, a entidade Mapa que, naturalmente, terá que ter as várias paragens de autocarros para cumprir a sua função de visualização no local das paragens.

O desenvolvimento da entidade Paragem e as suas relações sofreram várias alterações de forma a representar da melhor forma possível a arquitetura geral da aplicação. A equipa optou por uma relação circular, onde a paragem contém uma lista de rotas, como requerido, sendo estas rotas descritas pelo ser percurso, ou seja, as várias paragens que esta rota tem assim como as horas a que os autocarros chegam a cada uma destas.

Apresenta-se assim o modelo de Domínio final alcançado. Acredita-se que este modelo, apesar de simples, poderá explicar de forma sucinta o funcionamento pretendido da aplicação.

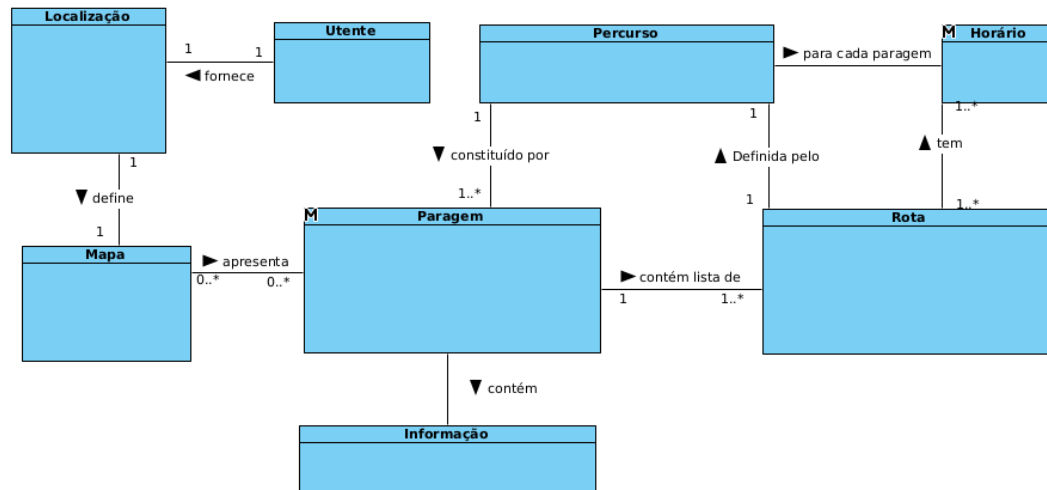


Figura 9 – Modelo de Domínio

3.2. Aspetos Comportamentais

Para definir o comportamento da aplicação a ser desenvolvida, concordou-se no desenvolvimento de um diagrama de *Use Cases*, assim como a descrição dos fluxos de cada um dos *Use Cases* que surgir do diagrama. Acredita-se que esta metodologia será a mais correta, pela facilidade de desenvolvimento e compreensão destes diagramas, mas também como pela profundidade e descrição das ações da aplicação de modo específico através do desenvolvimento dos fluxos.

3.2.1. Modelo Use Case

Para o diagrama de *Use Cases*, começou por se definir o seu único ator, o utente, e, de seguida, foi-se acrescentando a estes mais diagramas que a equipa acreditou serem necessários para o alcance dos requerimentos. No entanto, no desenvolvimento destes, começou-se por desenvolver vários *Use Cases*, principais e secundários, o que levou a uma fraturação destes, acabando por ter um diagrama complexo onde alguns *Use Cases* teriam uma dimensão extremamente reduzida. Por isso, para restaurar a simplicidade, desenvolveram-se apenas dois *Use Cases* distintos.

- *Use Case* “Obter mapa da cidade com paragens”, sendo este o *Use Case* principal, onde o utilizador opta pelo seu método de localização a usar, e, depois desta ser obtida, a aplicação realiza a comunicação com a base de dados para recolher a informação necessária e apresentá-la ao utilizador, sob a forma de um mapa localizado com as paragens que estiverem no seu redor.
- *Use Case* “Apresenta informação de uma paragem”, sendo um *Use Case* mais simples cujo único objetivo será apresentar de forma correta as várias informações acerca da paragem selecionada pelo utilizador.

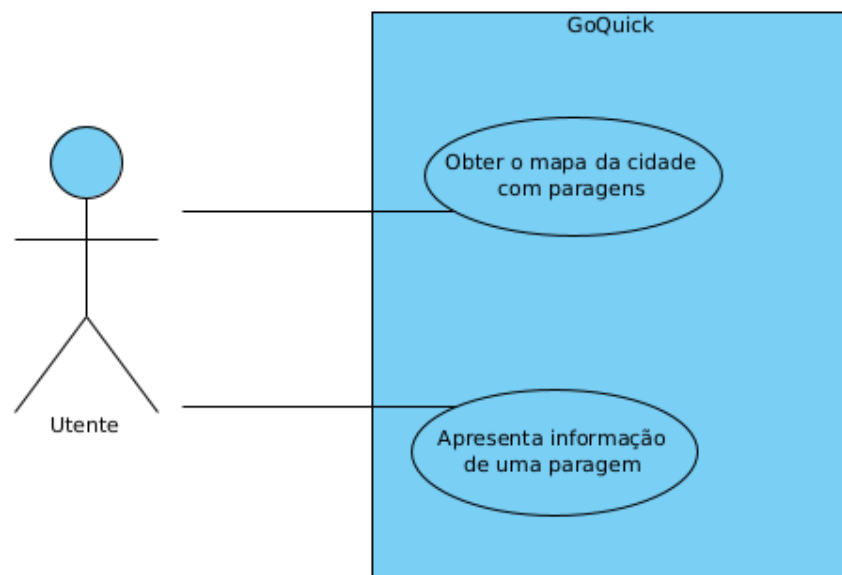


Figura 10 – Modelo de Use Case

3.2.2. Fluxos

Em conjunto com estes *Use Cases* desenvolvemos também a descrição de cada um dos Fluxos correspondentes. Estes não só auxiliam na definição concreta do comportamento da aplicação, como também no desenvolvimento inicial das várias *API's*, quer sejam *API's* da lógica de negócio ou da camada de interface de apresentação. Dado a reduzida capacidade da *API* da lógica de negócio, optamos por não separar esta em subsistemas, de forma a manter a interface de negócio centralizada.

O primeiro Fluxo realizado refere-se ao *Use Case* “Obter o mapa da cidade com paragens”. Este Fluxo será o principal da aplicação a desenvolver e, apesar de ser extensivo e conter vários fluxos alternativos, a equipa acredita que a divisão deste fluxo não seria benéfica na compreensão do seu comportamento. Neste fluxo também se pode observar a definição de alguns métodos a serem introduzidos na *API*. Estes serão especialmente importantes na próxima fase de planeamento.

Use case:	Obter o mapa da cidade com paragens	Dividir responsabilidade	Definir API
Cenários:	O Utente inicia a aplicação e escolhe qual método de localização a ser usado, de seguida a aplicação apresenta um mapa relativo à sua localização, com as paragens		
Pré-condições:	Há conexão com a base de dados		
Pós-condições:	O mapa local relativo à localização é apresentado, assim como as localizações das paragens		
Fluxo Normal:			
	1. O utente escolhe qual método de localização a usar	UI	apresentaMenuPrincipal()
	2. O dispositivo recolhe a informação do seu GPS	LN	getLocalizacao() : GPS
	3. A aplicação fornece à base de dados a sua localização.	—	—
	4. A base de dados recolhe todas as paragens num raio de 300m da localização fornecida.	LN	getParagensLocalizadas(localizacao : GPS) : Set<Paragem>
	5. A aplicação processa a resposta da query e adiciona as paragens na sua memória	LN	processaSetParagens(paragens : Set<Paragem>)
	6. A aplicação apresenta o mapa fixo na localização, com as paragens ao seu redor	UI	showMap(centro: GPS)
Fluxo Alternativo:			
	2. [O utente pretende fornecer o endereço]		
	2.1 O utente insere o endereço da rua a pesquisar	UI	InputString() : String
	2.2 A aplicação utiliza o sistema de pesquisa de endereços.	LN	localizacaoEndereco(endereco : String) : GPS
	2.3 A aplicação avalia se a localização pertence à cidade de Braga	LN	validaLocalizacao(localizacao : GPS) : Boolean
	2.4 Regressa ao passo 3		
Fluxo de Exceção:			
	2.3 [O endereço fornecido é inválido (Não foi possível calcular a localização ou esta está fora da cidade de Braga)]		
	2.3.1 A aplicação avisa o utente do endereço inválido	UI	falhaEndereco(msg : String)
Fluxo de Exceção:			
	2. [O dispositivo não tem meio para recolher a sua localização]		
	2.1 A aplicação avisa o utente sobre o erro de localização	UI	erroLocalizacao(msg : String)

Figura 11 - Fluxos do Use Case “Obter o mapa da cidade com paragens”

Para o *Use Case* “Obter informação de uma paragem” apenas se incluíram os passos necessários e suficientes para que a ação pretendida da aplicação possa ser realizada com sucesso. Esta apenas fará uma chamada à informação contida na memória e de seguida apresentá-la ao utente.

Use case:	Obter informação de uma paragem	Dividir responsabilidade	Definir API
Cenários:	O utente seleciona uma paragem para obter mais informação sobre esta		
Pré-condições:	Há conexão à base de dados.		
Pós-condições:	A informação da paragem é apresentada		
Fluxo Normal:			
	1. O utente seleciona uma paragem.	UI	selecionaParagem() : Int
	2. A aplicação recolhe a informação acerca da paragem	LN	getParagem(codParagem : Int) : Paragem
	3. A aplicação apresenta a informação da paragem ao utente.	UI	apresentaParagem(paragem : Paragem)

Figura 12 – Fluxos do *Use Case* “Obter informação da paragem”

3.3. Aspetos Estruturais

Depois da equipa desenvolver e concordar com os fluxos e comportamentos alcançados, avançou para a próxima fase de desenvolvimento, ou seja, a modelação da estrutura da aplicação. Para isso, decidiu esquematizar-se um Diagrama de Classes, visto que estes não só reúnem as *API*s desenvolvidas anteriormente, como também unificam a estrutura da aplicação de uma forma simples para a sua interpretação.

Iniciou-se este diagrama a partir das interfaces responsáveis pela lógica de negócio e interface da aplicação. Como referido, para estas utilizaram-se os vários métodos já definidos nos diagramas de fluxo.

3.3.1.Interface da Camada de Apresentação

Uma das primeiras partes desenvolvidas a partir dos fluxos é referente à interface da camada de apresentação ou interface da aplicação, esta contém os métodos necessários para a representação de toda a aplicação, contendo os métodos para apresentação dos menus iniciais, os mapas e a informação das paragens, assim como métodos de *input* e mensagens de erro.

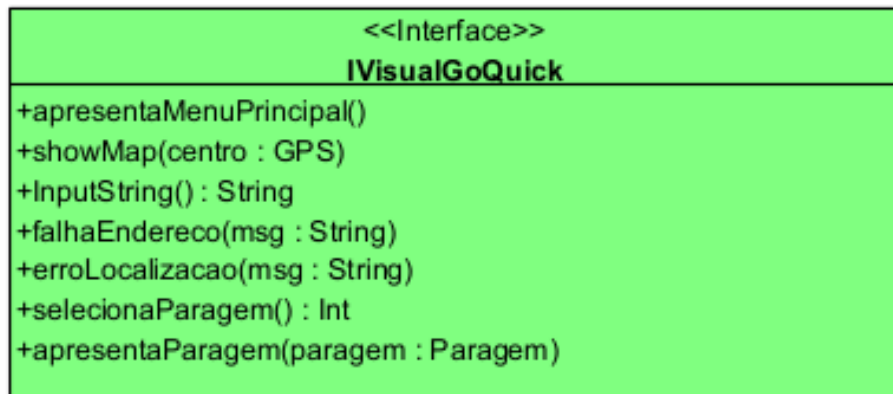


Figura 13 – Interface IVisualGoQuick

- **apresentaMenuPrincipal():** Método responsável pela execução inicial da aplicação, exibindo ao utilizador o Menu principal, assim como apresentar a escolha do método de localização a ser utilizado.
- **showMapa(GPS,Set<Paragem>):** Método cujo objetivo será apresentar um mapa fixo ao utilizador, centrado no *GPS* fornecido, nos quais as paragens contidas no Set irão ser representadas no mapa.
- **inputString():** Método simples para recolha de uma *String* inserida pelo utilizador.
- **falhaEndereco(String)** Método que dá um alerta ao utilizador a informar que houve um problema com o endereço inserido, ou que este é inválido.
- **erroLocalizacao(String)** Método que dá um alerta ao utilizador a informar que houve um problema com o sistema de localização automático.
- **selecionaParagem()** Método responsável por identificar a paragem selecionada pelo utilizador, retornando o identificador da paragem.
- **apresentaParagem(Paragem)** Método responsável pela apresentação de toda a informação contida na paragem selecionada, apresentando nome, imagem, descrição, localização e a lista de horários.

3.3.2.Interface da Camada de Negócio

Naturalmente, não vale a pena ter uma interface de apresentação sem ter forma de tratar e introduzir os dados a serem apresentados, por isso, desenvolveu-se a interface responsável pela lógica de negócio, utilizando os métodos já estabelecidos. Esta interface será responsável pelo tratamento dos dados recebidos através da base de dados, assim como armazenar estes dados na forma mais conveniente.

Tendo isto em conta, a equipa desenvolveu uma lógica de negócio simples, cuja função será cumprir os requisitos e operações da *API* definidas. Além disso, terá que armazenar, numa estrutura, paragens de autocarros sobre as quais já contém informação proveniente da base de dados.

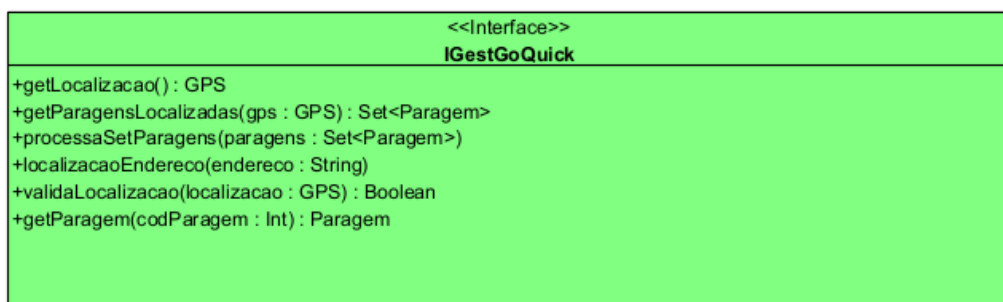


Figura 14 – Interface IGestGoQuick

- **getLocalizacao():** Método responsável por obter a localização do dispositivo, o objetivo deste método será executar uma função de localização que o sistema operativo do dispositivo disponibilize e tratar a informação que for devolvida, assim como inseri-la na sua variável de localização. Caso o dispositivo não contenha funções de localização, ou a permissão seja negada, esta deverá lançar uma exceção.
- **getParagensLocalizadas(GPS):** Este método é o único que irá efetuar a comunicação à *BD*. Este irá efetuar uma *query* definida à base de dados que deverá retornar o *Set* de paragens que se encontrem num raio de 300 metros da localização fornecida.
- **processaSetParagens(Set<Paragem>):** Este método é responsável pelo tratamento de dados resultantes da *query* realizada em “getParagensLocalizadas”, este irá fazer o *parsing* necessário e armazenar as paragens obtidas na sua lista ordenada e identificável.
- **localizacaoEndereco(String):** Este método tem como objetivo recolher a localização *GPS* de um endereço fornecido, para isto a aplicação recorrerá à *API* da *Google*.
- **validaLocalizacao():** Este método verifica se a localização obtida, tanto pelo *GPS* como pelo endereço fornecido, corresponde a uma localização dentro da cidade de Braga, fazendo uma verificação simples através das coordenadas.
- **getParagem(Int):** Este método tem como função recolher a informação de uma paragem selecionada, quando pedida pela camada de apresentação.

3.3.3. Classes de Armazenamento de Dados em Memória

Por fim, definiram-se as classes responsáveis para o armazenamento dos dados a serem recebidos pela base de dados. Estes irão ser fundamentais para estruturar a informação proveniente das *queries* executadas, garantindo um acesso rápido e eficaz à memória, com o objetivo de evitar acessos múltiplos à base de dados.

Dada a simplicidade das funções realizadas pela aplicação, a equipa não encontrou grande necessidade de um desenvolvimento profundo das classes, visto que a aplicação revolve apenas à volta da apresentação geográfica de paragens de autocarro num mapa e na exibição de informação adicional acerca destas.

Dado isto, desenvolveu-se a classe que irá conter toda a informação destas paragens, a classe “Paragem”, esta é uma das classes de maior extensão no projeto, no entanto a funcionalidade desta é reduzida, resumindo-se ao armazenamento da informação acerca da paragem e a devolução dos valores que contém, como o seu nome, a sua localização, a descrição e a lista de horários.

A paragem contém também na sua especificação os objetos da classe “GPS” e “Horário”, sendo este segundo guardado numa lista ordenada e identificável (por exemplo, um *Map*). Encontramos a necessidade de criar estas classes auxiliares pois estes dados contêm uma maior complexidade, pelo que a classe “GPS” tem como função guardar os valores das coordenadas de latitude e longitude da paragem, e a classe “Horário” é composta por uma lista de *Strings*, correspondentes às horas que os autocarros de uma rota passam nessa paragem.

Com estas interfaces e classes definidas, obteve-se este Diagrama de Classes final:

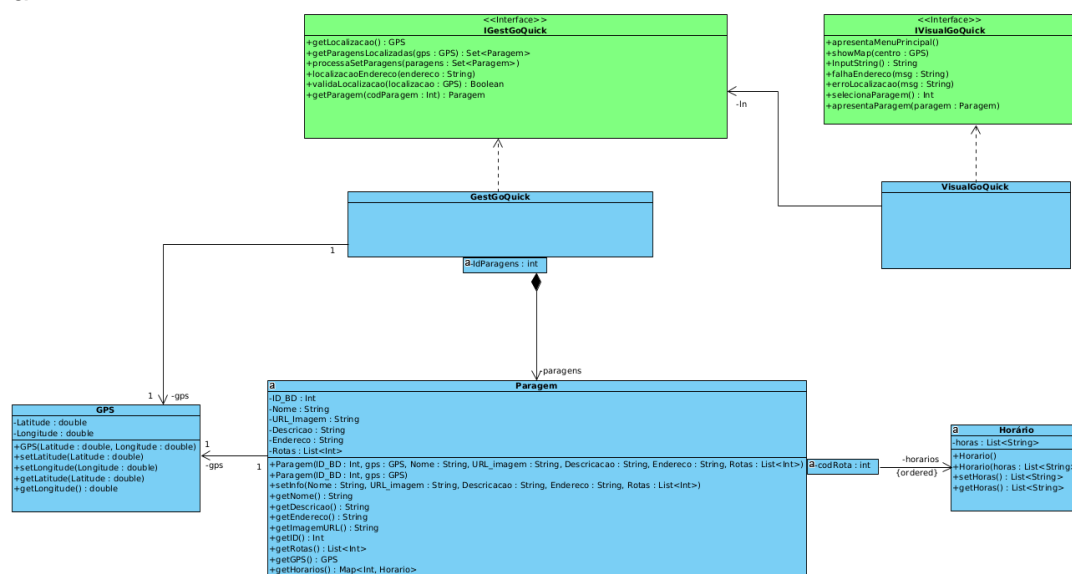


Figura 15 – Diagrama de Classes

4. Conceção do Sistema de Dados

Nesta parte do relatório, a equipa vai definir a estrutura, conceitos e modelos a serem implementados na base de dados da aplicação. Esta vai definir o funcionamento da base de dados presente *online* e como esta organiza os seus dados. Esta parte do projeto foi a que necessitou de um maior cuidado e discussão entre a equipa devido à in experiência no desenvolvimento de base de dados.

Para este desenvolvimento, começou por se estruturar um modelo conceptual da base de dados que satisfaz os requerimentos da aplicação. Este protótipo conceptual foi desenvolvido no *software brModelo*, visto ser uma ferramenta para este propósito. Esta decisão foi tomada devido à facilidade de uso, versatilidade e utilidade que este *software* apresenta.

Depois do modelo conceptual desenvolvido, escolhemos utilizar a ferramenta *MySQL* para a criação do modelo lógico da base de dados destinada à aplicação. Esta decisão foi tomada devida às várias qualidades deste *software* e a sua adoção geral, sendo um dos *softwares* mais populares. Este tem um bom desempenho e é facilmente adaptável a qualquer tipo de linguagem de programação, com uma margem de aprendizagem acessível.

4.1. Apresentação Geral da Estrutura do Sistema de Dados

Desde o início, a equipa pretendia ter as paragens como a entidade central na sua base de dados, visto que estas são o foco da aplicação. No entanto, com algumas das informações relativas a esta, como as rotas e o horário, as definições destas na base de dados foram sujeitas a várias discussões entre a equipa e sofreram várias mudanças no seu conceito inicial.

Parte deste problema proveio das várias interpretações reais, assim como definições do que são as rotas e horários, tendo a equipa debatido entre incluir as horas como parte exclusiva da rota, ou até ter uma duplicação de horários, dado que cada rota tem um horário, mas também se pode concluir que cada paragem tem o seu horário.

Por fim, decidiu-se incluir os horários como membro de uma paragem, visto que o foco nesta aplicação e a sua base de dados é relativo às informações das paragens. Definiu-se assim que um horário seria um conjunto de identificadores de rotas associados a uma hora, sendo estes pertencentes a uma paragem só.

4.1.1. Modelo Conceptual

Depois da equipa entrar em acordo com o funcionamento da base de dados da aplicação, desenvolveu-se um modelo conceptual. Este define a entidade paragem como o seu centro, sendo identificável com uma chave e contendo os vários campos relativos à sua informação. Também foram incluídas as relações entre as paragens e as suas rotas assim como as horas presentes nas paragens.

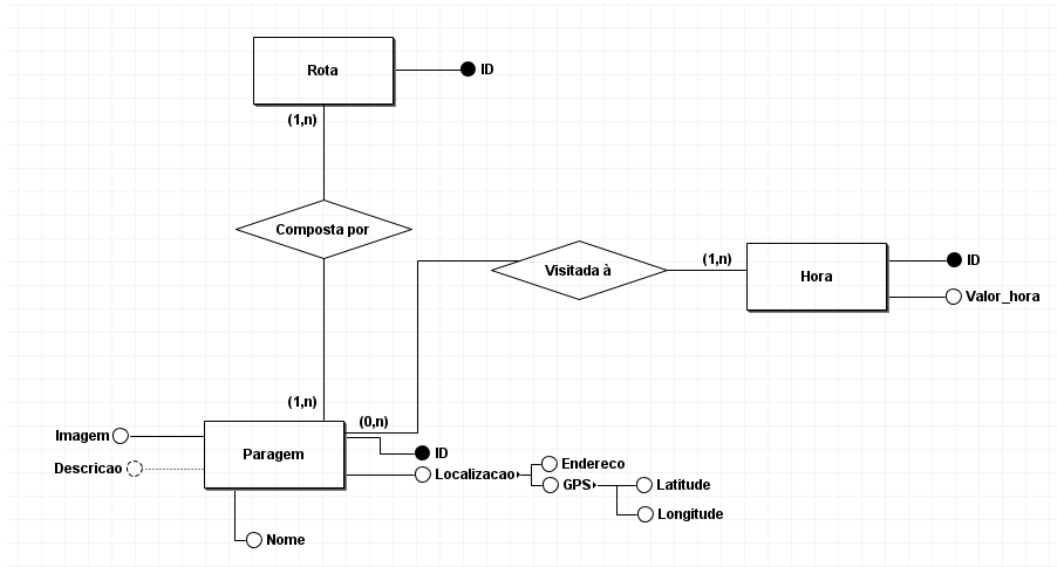


Figura 16 – Modelo Conceptual da BD

4.1.2. Modelo Lógico

De seguida, procedeu-se ao desenvolvimento do modelo lógico em *MySQL* utilizando o modelo conceptual como base. Este terá 5 tabelas na sua representação, uma vez que as relações entre as entidades **Rota** e **Paragem**, bem como entre **Hora** e **Paragem** são relações de cardinalidade n:m.

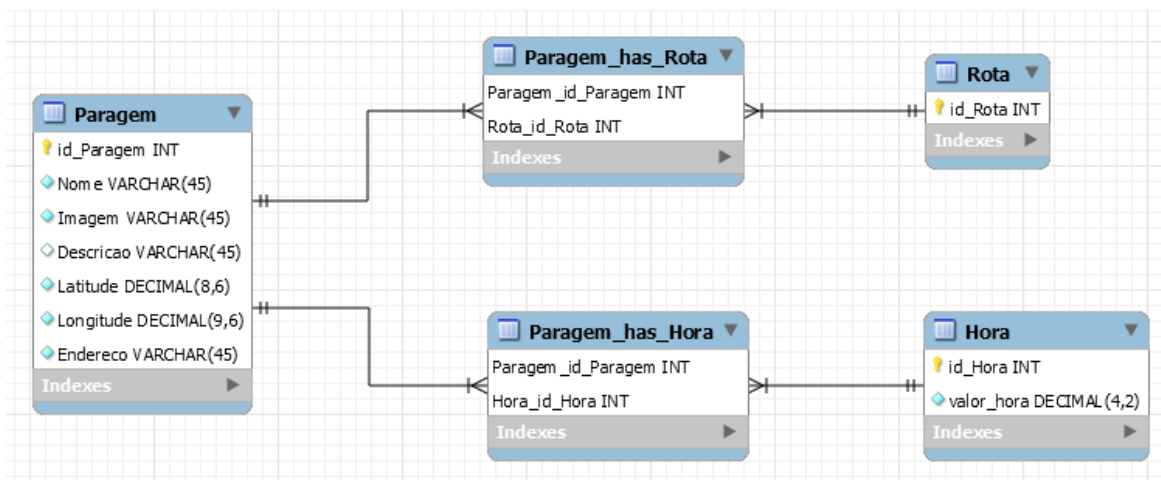


Figura 17 – Modelo lógico da BD

4.2. Descrição dos Elementos de Dados e os seus Relacionamentos

Neste subcapítulo, serão explorados os elementos de dados explícitos na base de dados originada, procedendo à explicação dos tipos de variáveis escolhidos e os diversos relacionamentos que estes têm entre si.

4.2.1. Rota

Atributo	Tipo	Chave Primária	Chave Estrangeira	Opcional
Id_Rota	INT	Sim	Não	Não

Table 1 - Tabela Rota do Esquema Lógico

Uma entidade Rota será identificada pelo seu ID, sendo este um número inteiro (63, por exemplo). Através deste, é possível saber quais as paragens que uma dada Rota interseja e a que horas tal se sucede em cada uma delas, devido às relações entre as entidades numeradas.

4.2.2. Paragem_has_Rota

Atributo	Tipo	Chave Primária	Chave Estrangeira	Opcional
Paragem_id_Paragem	INT	Sim	Sim	Não
Rota_id_Rota	INT	Sim	Sim	Não

Table 2 - Tabela Paragem_has_Rota do Esquema Lógico

Esta tabela é produto da relação entre as entidades Rota e Paragem que, sendo uma relação n:m, dá origem a uma nova tabela. A partir daqui é possível identificar que rotas passam em que paragens.

4.2.3.Paragem

Atributo	Tipo	Chave Primária	Chave Estrangeira	Opcional
Id_Paragem	INT	Sim	Não	Não
Nome	VARCHAR()	Não	Não	Não
Imagem	VARCHAR()	Não	Não	Não
Descricao	VARCHAR()	Não	Não	Sim
Latitude	DECIMAL(8,6)	Não	Não	Não
Longitude	DECIMAL(9,6)	Não	Não	Não
Endereco	VARCHAR()	Não	Não	Não

Table 3 - Tabela Paragem do Esquema Lógico

Nesta tabela encontra-se toda a informação relativa a uma Paragem. O tipo de dados 'DECIMAL(8,6)' e 'DECIMAL(9,6)', utilizados para a Latitude e Longitude, irão resultar, respetivamente, numa margem de erro de aproximadamente 10cm para a localização real. Além do mais, utilizamos o dado "VARCHAR()" para descrever o endereço *URL* da imagem pretendida. Estas devem ser guardadas no servidor responsável por responder às *queries*.

4.2.4.Paragem_has_Hora

Atributo	Tipo	Chave Primária	Chave Estrangeira	Opcional
Paragem_id_Paragem	INT	Sim	Sim	Não
Hora_id_Hora	INT	Sim	Sim	Não

Table 4 - Tabela Paragem_has_Rota do Esquema Lógico

Esta tabela resulta da relação n:m entre as entidades Paragem e Hora. A partir desta será possível descobrir a que horas uma Rota interseja uma dada Paragem.

4.2.5.Hora

Atributo	Tipo	Chave Primária	Chave Estrangeira	Opcional
Id_Hora	INT	Sim	Não	Não
Valor_hora	DECIMAL(4,2)	Não	Não	Não

Table 5 - Tabela Hora do Esquema Lógico

Na tabela que explicita as relações e características da entidade Hora existem apenas dois atributos: um que identifica a hora e outro que irá guardar esta. Isto acontece devido às várias rotas que podem ter a mesma hora associadas ao intersejar uma determinada Paragem.

5. Esboço das Interfaces Gráficas do Sistema

Para o desenvolvimento do aspeto das interfaces gráficas do sistema, começaram por se esboçar uma ideia inicial para várias instâncias da aplicação. Como tal, obtiveram-se esboços para o menu inicial, inserção de endereço (caso o utilizador selecione esse método de deteção de localização), apresentação do mapa com as várias paragens e por fim as informações de uma paragem selecionada pelo utilizador.

5.1. Estrutura Geral das Interfaces do Sistema

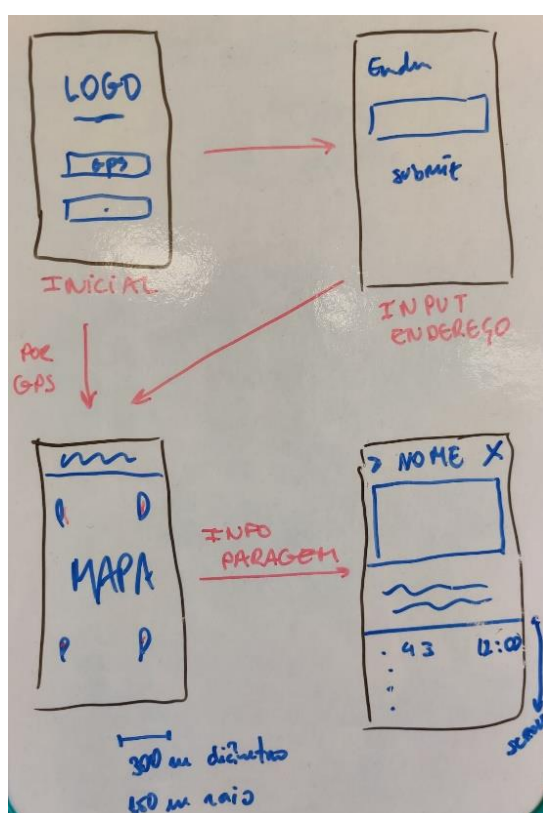


Figura 18 – Esboço das interfaces gráficas

de um raio de 300 metros a partir da localização do utilizador.

Por fim, quando o utilizador selecionar uma paragem do mapa, a informação da mesma é apresentada segundo o menu do canto inferior direito.

5.2. Caracterização das Interfaces

De seguida, utilizou-se a ferramenta de desenho *online Draw.io*, complementado com o *software Krita* para desenvolver esboços mais concretos. Optou-se por um tema de cores vermelho e azul, dado que estas não só foram as cores escolhidas para o logotipo, mas como também representam as cores da empresa *TUB*, responsável pelos transportes públicos em Braga.

A partir dos esboços da figura no ponto acima, construíram-se algumas ideias concretas de interfaces gráficas para os menus apresentados nela:



Figura 19 – UI Menu Principal



Figura 20 – UI Recolha de Endereço

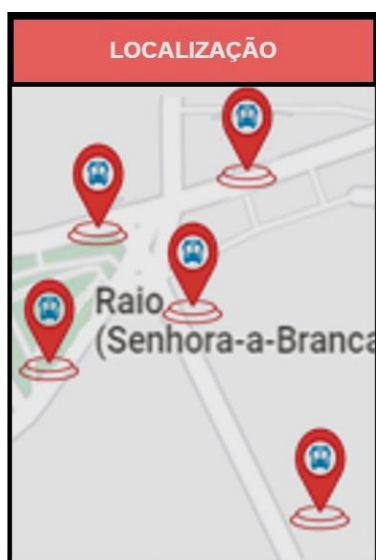


Figura 21 – UI Apresentação do Mapa



Rota 43	12:00
Rota 94	12:35
Rota 31	12:55
Rota 7	13:10
Rota 63	13:20

Figura 22 – UI Informação de uma paragem

6. Conclusões e Trabalho Futuro

6.1. Apreciação Crítica

A equipa encontra-se satisfeita com o desenvolvimento inicial deste projeto, acreditando no potencial sucesso da aplicação a ser desenvolvida. No entanto, dado à relativa inexperiência da equipa no uso desta metodologia no desenho e planificação de *software*, foram encontradas algumas dificuldades, especificamente no início do projeto e, mais tarde, entre aspetos de especificação da *BD*.

Apesar disso, sentimos que a equipa teve uma evolução positiva ao longo do projeto, desenvolvendo novos conhecimentos e capacidades.

6.2. Trabalho Futuro

Acreditamos que o progresso feito ao longo deste projeto se irá refletir positivamente no desenvolvimento de projetos futuros. No entanto, há certos aspetos que podem ser melhorados, e que, se forem resolvidos, podem aumentar a produtividade da equipa, como promover discussões mais a fundo acerca dos objetivos e expectativas de cada membro em relação ao projeto, assim como uma melhor exposição de ideias entre membros.

6.3. Conclusão

O desenvolvimento de *software* é complexo. Deste modo, o desenvolvimento de qualquer aplicação passa por fases em que este é maioritariamente teórico. Esta fase é significativamente mais importante que a própria implementação da aplicação pois define previamente vários aspetos como objetivos, requisitos, planeamento e funcionamento da aplicação, correndo-se menos riscos de erros, inconsistências e reescrita de código. Assim, o trabalho apresentado neste documento evita possíveis problemas que possam surgir na implementação do projeto.

Com esta fase terminada, e tendo em conta o trabalho cuidadoso e metódico da equipa em especificar e modelar o funcionamento da aplicação, acredita-se ter desenvolvido um planeamento necessário para a implementação de uma aplicação capaz não só ajudar os cidadãos de Braga no seu transporte diário, como também reduzir o congestionamento nas ruas e proporcionar um meio ambiente mais saudável.

7.Referências

José Paulo Silva, 2021. Braga urbana tem o maior crescimento populacional dos maiores municípios, Correio do Minho [online], 29 Agosto, Disponível em: <https://correiodominho.pt/noticias/braga-urbana-tem-o-maior-crescimento-populacional-dos-maiores-municipios/132312> [Acedido dia 26 de outubro de 2021]

CENSOS, 2021.INE – Censos, resultados preliminares 2021. Disponível em: https://www.ine.pt/scripts/db_censos_2021.html [Acedido dia 26 de outubro de 2021]

ANSR, 2021. Relatório de Sinistralidade e Fiscalização Agosto 2021. Disponível em: <http://www.ansr.pt/Estatisticas/RelatoriosDeSinistralidade/Documents/2021/Relat%C3%B3rioSinistralidadeAgosto2021.pdf> [Acedido dia 26 de outubro de 2021]

APA, 2019. Relatório do Estado do Ambiente. Disponível em: <https://sniambgeoviewer.apambiente.pt/GeoDocs/geoportaldocs/rea/REA2019/REA2019.pdf> [Acedido dia 16 de novembro de 2021]

Margareta Friman, Katrin Lättman, Lars E. Olsson, 2020. Public Transport Quality, Safety, and Perceived Accessibility. MDPI. [online] Disponível em: <https://www.mdpi.com/2071-1050/12/9/3563> [Acedido dia 9 de novembro de 2021]

Kaspan Eka Putra & Juwita Magdalena Sitanggang, 2006. The effect of public transport services on quality of life in Medan city. Elsevier. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1877042816315099> [Acedido dia 9 de novembro de 2021]

ACOG, 2013. Why Transit Matters: The Environmental Benefits of Public Transportation. Disponível em: <https://www.acogok.org/why-transit-matters-environment/> [Acedido dia 16 de novembro de 2021]

8. Siglas

GPS – *Global Positioning System*

BD – Base de datos

API – *Application Programming Interface*

UI – *User Interface*

TUB – Transportes Urbanos de Braga