



Universidade do Minho

Escola de Engenharia

Mestrado Integrado em Engenharia Informática

Unidade Curricular de Laboratórios de Informática IV

Ano Letivo de 2021/2022

<<GoOutBraga>>

Rita Gomes(87960) Duarte Moreira(93321) Lucas
Carvalho(93176) João Ferreira(93263) João Lourenço(93233)

19 de fevereiro de 2022

LI4

Data de Receção	
Responsável	
Avaliação	
Observações	

<<**GoOutBraga**>>

Rita Gomes(87960) Duarte Moreira(93321) Lucas Carvalho(93176) João
Ferreira(93263) João Lourenço(93233)

19 de fevereiro de 2022

Resumo

A primeira parte do projeto teve como objetivo desenvolver uma especificação de uma aplicação que seja um guia para locais de interesse.

O presente relatório descreve o processo de execução da segunda parte do projeto. Esta fase consistiu na implementação de um guia de locais de interesse com foco na área de restauração, sendo que a linguagem escolhida pelo grupo para concretizar este trabalho foi a linguagem Java.

A aplicação dispõe como finalidade permitir aos seus usuários conhecer restaurantes na cidade de Braga, de acordo com as suas preferências, como por exemplo: nome, preço e classificação. Para além disso, o sistema deverá permitir ao utilizador avaliar um estabelecimento e, de acordo com a frequência destas avaliações, o restaurante poderá disponibilizar promoções.

O trabalho realizado consistiu, numa primeira fase, na caracterização do trabalho a desenvolver e identificação dos recursos necessários, acompanhado do respetivo Diagrama de Gantt, que ajudou na organização das diversas tarefas intermédias para atingir um sistema capaz de responder a todos os critérios requeridos.

Numa segunda fase procedeu-se ao desenvolvimento da aplicação, passando este por várias etapas, tais como a identificação e caracterização da aplicação e dos seus serviços, explicação da arquitetura geral da aplicação e caracterização de todos os componentes que esta engloba.

Por fim, são elaboradas algumas conclusões sobre todo o trabalho desenvolvido, bem como sobre o processo em si.

Área de Aplicação: Desenvolvimento de um guia para locais de interesse.

Palavras-Chave: Restaurante, Java, Engenharia de Software, Aplicação Web.

Índice

1	Introdução	1
1.1	Enquadramento geral	1
1.2	Caracterização do trabalho desenvolvido	2
1.3	Identificação dos recursos necessários	2
1.4	Plano de desenvolvimento	3
2	Desenvolvimento da Aplicação	4
2.1	Apresentação da estratégia utilizada	4
2.2	Caracterização da aplicação	4
2.3	Arquitetura geral da aplicação	5
2.4	Caracterização dos componentes da aplicação	6
3	Considerações finais e Trabalho Futuro	13
	Lista de Siglas e Acrónimos	15
	Anexos	16
	Anexo 1	16
	Anexo 2	17

Lista de Figuras

1.1	Diagrama de Gantt.	3
2.1	Login.	6
2.2	Registo de um cliente.	7
2.3	Registo de um proprietário.	7
2.4	Menu de um cliente.	8
2.5	Menu de um proprietário.	8
2.6	Perfil de proprietário.	9
2.7	Perfil de utilizador.	9
2.8	Menu de filtragem.	10
2.9	Menu de um restaurante.	10
2.10	Menu de avaliação.	11
2.11	Reclamar cupão.	11
2.12	Criar e distribuir cupão.	12
2.13	Ver avaliações.	12
3.1	Modelo Lógico.	16
3.2	Script SQL da Base de Dados.	20

Lista de Tabelas

1 Introdução

O seguinte projeto realizou-se no âmbito da unidade curricular Laboratórios de Informática IV, presente no terceiro ano de LEI, e tem como objetivo principal o desenvolvimento de uma aplicação que deverá ser capaz de providenciar aos utilizadores uma forma intuitiva de encontrar o serviço que mais se adequa aos seus gostos, ao mesmo tempo que contribui para aumentar a exposição dos estabelecimentos locais.

Desta forma, a aplicação elaborada deverá sugerir estabelecimentos adequados às preferências do seu utilizador, tais como nome, preço ou classificação do estabelecimento. Esta aplicação terá o seu foco em restaurantes da cidade de Braga.

Ainda neste capítulo, é realizada a caracterização do trabalho desenvolvido, seguida de como este foi implementado, no capítulo posterior.

1.1 Enquadramento geral

Infelizmente, nos últimos dois anos temos vivido uma crise provocada pela COVID-19, sendo que a área da restauração foi das mais afetadas, o que levou a um decréscimo no consumo deste setor.

Esta aplicação surge como uma forma de recriar e revolucionar esta área, permitindo aos seus utilizadores um acesso simples e intuitivo a toda a informação sobre os restaurantes da cidade de Braga, ao mesmo tempo que ajuda a promover a imagem dos locais anunciados na aplicação, resultando numa nova fonte de rendimento destes mesmos.

Existe uma infinidade de variáveis que leva uma pessoa a escolher um determinado estabelecimento e, na área de restauração, esta escolha pode passar pelo tipo de estabelecimento (restaurante ou bar), qualidade e preço médio, localização ou ainda tipo de comida. Esta aplicação faz com que esta escolha seja muito mais simples e rápida.

Com o objetivo de aumentar a quantidade de utilizadores da aplicação e, consequentemente, a quantidade de clientes de um estabelecimento, a aplicação deverá oferecer um serviço de promoções associado a um sistema de avaliações, sendo que quanto mais o usuário avaliar um estabelecimento que visitou mais chances tem de ganhar cupões de desconto, o que cria uma situação benéfica tanto para os clientes como para os locais anunciados no sistema.

1.2 Caracterização do trabalho desenvolvido

Com base em tudo aquilo que já foi referido acima temos agora de analisar as funcionalidades que teremos de implementar para construir o sistema requerido.

Primeiramente, temos de ter em conta que qualquer pessoa que utilize o serviço terá que estar previamente registada no sistema, quer seja o dono de um estabelecimento quer seja um utilizador que pretenda utilizar a aplicação, ou caso isso não se verifique tenha a possibilidade de se registar e criar uma conta.

Cada estabelecimento apresenta uma imagem submetida pelo dono e apresenta como informação a classificação média com o número de avaliações, horário e preço médio. Para além disso, cada restaurante tem ainda associado o horário de funcionamento, nome, contacto e cardápio.

O utilizador poderá pesquisar por um estabelecimento específico, através do nome ou através de uma categoria (classificação média ou preço médio).

O utilizador tem ainda a possibilidade de avaliar o serviço de um restaurante que visitou podendo deixar uma classificação por estrelas entre 1 e 5, deixar um comentário ou ainda seleccionar uma avaliação do preço de acordo com as opções apresentadas (caro, barato, médio).

Este sistema deverá ainda dar a possibilidade de um estabelecimento oferecer um cupão de desconto aos utilizadores que avaliem o restaurante. A este cupão está associado um prazo para o utilizador o usufruir.

1.3 Identificação dos recursos necessários

De forma a que o desenvolvimento e implementação deste projeto seja bem sucedido serão necessários diversos recursos, quer a nível humano, quer a nível de software.

Para a implementação deste sistema o grupo decidiu utilizar a linguagem Java, visto ser a linguagem com que todos os elementos do grupo considera ter mais agilidade.

Quanto ao software de desenvolvimento e de suporte utilizado, o grupo optou por usar o Android Studio, devido ao facto de este ser um IDE direccionado especificamente, para o desenvolvimento de plataformas Android.

1.4 Plano de desenvolvimento

O plano de desenvolvimento deste projeto será dividido em fases distintas:

1. Relatório
2. Especificação
3. Back-End
4. Front-End
5. Finalização

De seguida, apresentamos o diagrama de Gantt que ilustra todo este planeamento rígido de todas as etapas do desenvolvimento do projeto.

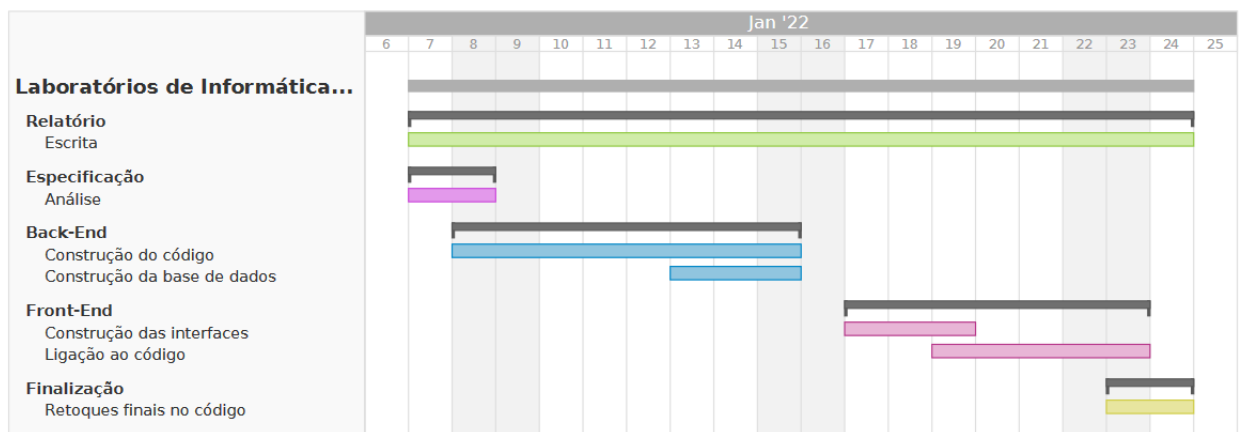


Figura 1.1: Diagrama de Gantt.

2 Desenvolvimento da Aplicação

2.1 Apresentação da estratégia utilizada

Para implementar o sistema solicitado, começamos por analisar detalhadamente o relatório do grupo que nos forneceu a especificação e procedemos à tomada de decisão de quais seriam as funcionalidades que iríamos desenvolver.

Depois disto e devido ao atraso do início da execução desta tarefa, decidimos adotar uma estratégia segura, na medida em que tentamos implementar um sistema apenas com os requisitos básicos.

Assim sendo, primeiramente decidimos dividir o sistema em Model, View, Controller e, a partir daí começar a implementar a aplicação sem nos preocuparmos com a parte da interface, ou seja, numa fase inicial a aplicação foi sendo corrida no terminal. Após termos o código funcional passamos à criação das interfaces da aplicação e posteriormente à ligação do código com as mesmas, o que nos levou ao resultado final.

2.2 Caracterização da aplicação

Com a especificação fornecida o grupo conseguiu implementar grande parte das funcionalidades propostas. Assim, a aplicação é capaz, neste momento, de:

Registar utilizadores, tanto clientes como proprietários.

Autenticar(login) utilizadores, tanto clientes como proprietários.

Consultar dados/perfil do utilizador, sendo este cliente ou proprietário.

Editar dados/perfil do utilizador, sendo este cliente ou proprietário.

Pesquisar por estabelecimentos, o que permite ao cliente procurar um restaurante com base no seu nome, classificação ou preço médio.

Listar estabelecimentos, que apresenta uma listagem de todos os restaurantes registados na aplicação.

Visualizar estabelecimento, que apresenta a interface de um restaurante, ou seja, dados do mesmo e algumas funcionalidades.

Adicionar/remover favorito, que permite ao cliente gerir a sua lista de restaurantes favoritos.

Avaliar restaurante, que permite ao cliente deixar uma crítica ao estabelecimento.

Reclamar cupão, que permite ao cliente escolher um cupão oferecido pelo restaurante.

Usar cupão, que permite ao cliente usufruir dos cupões que reclamou.

Consultar avaliações, que permite ao proprietário visualizar todas as críticas realizadas ao seu restaurante.

Distribuir cupão, que permite ao proprietário oferecer cupões aos clientes do seu estabelecimento.

Por outro lado, o grupo não foi capaz de satisfazer os seguintes requisitos:

Achievements, este tópico foi falado na especificação de forma muito breve, não especificando o que é e como se obtém, por isso o grupo acabou por não lhe dar muita atenção.

Localização, o sistema de localização tanto do cliente como dos restaurantes não se apresenta funcional uma vez que o grupo não arranhou maneira de o implementar.

Base de dados, apesar de termos feito uma base de dados, a ligação desta com o aplicação não foi realizada. Para podermos fazer testes foi feito um povoamento manual de restaurantes ou então o registo dos mesmos, assim como de clientes.

Validação, o sistema de validação de um cliente também foi um insucesso uma vez que o grupo, para além de não ter percebido completamente o cenário apresentado, não conseguiu pensar numa forma de aceder à base de dados do sistema de faturação de cada restaurante.

2.3 Arquitetura geral da aplicação

A arquitetura da aplicação segue o modelo **MVC**. Este modelo segue uma arquitetura de software que separa a lógica da interface do usuário. Isto é conseguido através da separação da aplicação em três partes distintas: **Model**, **View**, **Controller**. De forma resumida, o **Model** representa o comportamento lógico dos dados na aplicação, a **View** fornece a interface de usuário da aplicação e o **Controller** define o comportamento da aplicação.

2.4 Caracterização dos componentes da aplicação

Ao executar a aplicação somos direccionados para a tela de login, aí podemos fazer o login inserindo as credenciais 'username' e 'password' ou então criar uma conta como usuário ou como proprietário de um estabelecimento.

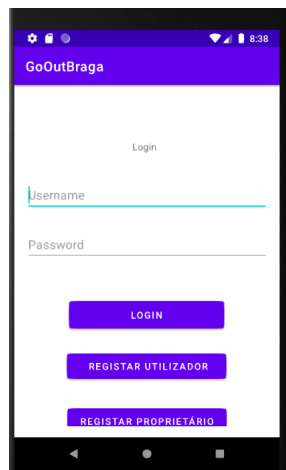
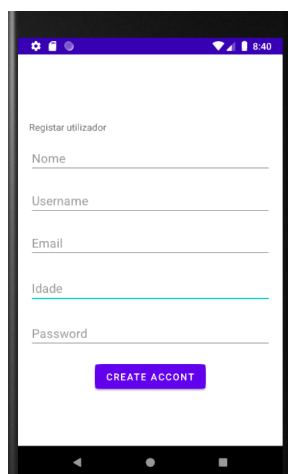


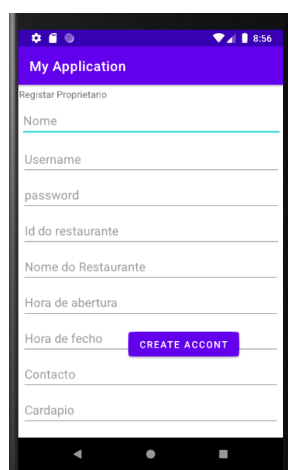
Figura 2.1: Login.

Se o usuário pretender registrar-se é levado para uma das interfaces seguintes, dependendo da conta que pretende criar.



A screenshot of a mobile application interface for registering a user. The title is "Registrar utilizador". Below the title are six input fields: "Nome", "Username", "Email", "Idade", and "Password". The "Idade" field is highlighted with a red border. At the bottom of the form is a red button labeled "CREATE ACCOUNT". The status bar at the top shows the time as 8:40.

Figura 2.2: **Registo de um cliente.**



A screenshot of a mobile application interface for registering a restaurant owner. The title is "Registrar Proprietario". Below the title are eight input fields: "Nome", "Username", "password", "Id do restaurante", "Nome do Restaurante", "Hora de abertura", "Hora de fecho", "Contacto", and "Cardapio". The "Hora de fecho" field is highlighted with a red border. At the bottom of the form is a red button labeled "CREATE ACCOUNT". The status bar at the top shows the time as 8:56.

Figura 2.3: **Registo de um proprietário.**

Uma vez registado como cliente o usuario pode executar 5 ações, 'ver perfil', 'procurar restaurante', 'listar restaurantes', 'favoritos' ou então fazer logout da sua conta.

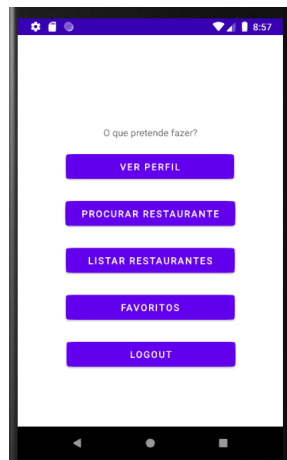


Figura 2.4: **Menu de um cliente.**

Uma vez registado como proprietario o usuario pode executar 4 ações, 'ver perfil', 'avaliacoes', 'distribuir cupao' ou então fazer logout da sua conta.



Figura 2.5: **Menu de um proprietário.**

Tanto o user como o proprietário podem verificar o seu perfil e mudar alguns campos de informação.

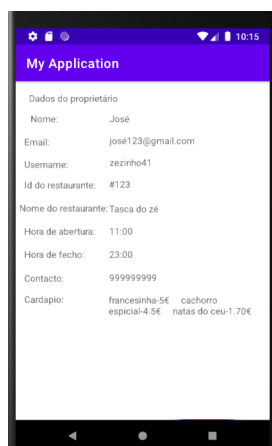


Figura 2.6: Perfil de proprietário.

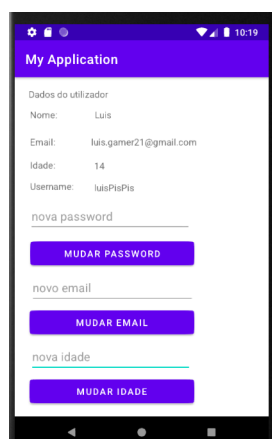


Figura 2.7: Perfil de utilizador.

Se o utilizador optar por procurar um restaurante ou restaurantes com certas características este pode filtra-lo de 3 formas, pesquisando o seu nome, pesquisando por preços(caro, medio, barato) ou por avaliação(de 1 a 5 estrelas).

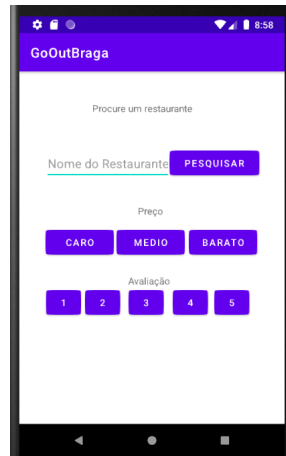


Figura 2.8: Menu de filtragem.

Sempre que for necessário listar restaurantes, a aplicação fá-lo sempre da mesma forma, mostra toda a informacao de um restaurante assim como todas as operações possíveis, com a opcao de mudar de página(proxima e anterior).

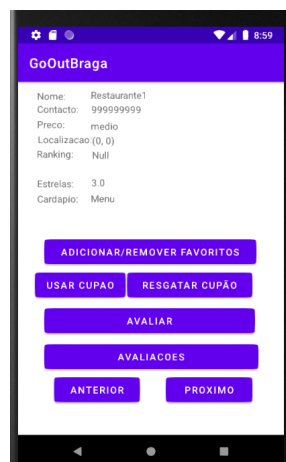


Figura 2.9: Menu de um restaurante.

No menu de avaliação é possível escrever um comentário, atribuir calssificação e atribuir um tipo de preço.



Figura 2.10: **Menu de avaliação.**

No menu de reclamar cupão o user escolhe o cupao que pretende resgatar baseando-se na sua descrição.

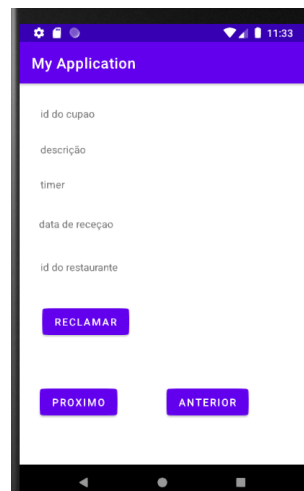
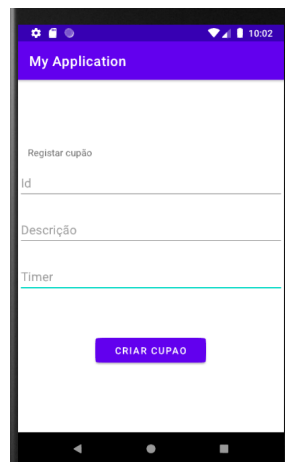


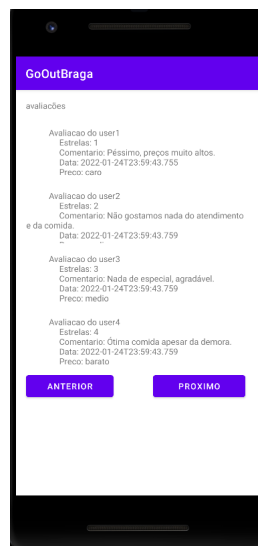
Figura 2.11: **Reclamar cupão.**

Se o proprietário optar por criar cupão basta dar-le um id, uma descrição e um timer(validade).



The screenshot shows a mobile application interface titled "My Application". It features a form with three input fields labeled "Id", "Descrição", and "Timer". Below the form is a purple button labeled "CRIAR CUPAO". The status bar at the top shows the time as 10:02.

Figura 2.12: **Criar e distribuir cupão.**



The screenshot shows a mobile application interface titled "GoOutBrage". It displays a list of reviews under the heading "avaliacoes". Each review entry includes the user name, star rating, comment, date, and price. At the bottom, there are two purple buttons labeled "ANTERIOR" and "PROXIMO".

user	Estrelas	Comentario	Data	Preço
user1	1	Péssimo, preços muito altos.	2022-01-24T23:59:43.755	caro
user2	2	Não gostamos nada do atendimento e da comida.	2022-01-24T23:59:43.759	
user3	3	Nada de especial, agradável.	2022-01-24T23:59:43.759	medio
user4	4	Ótima comida apesar da demora.	2022-01-24T23:59:43.759	barato

Figura 2.13: **Ver avaliações.**

3 Considerações finais e Trabalho Futuro

De uma forma geral, o grupo considera que podia ter feito um trabalho melhor visto que algumas funcionalidades ficaram por fazer assim como a ligação da base de dados ao sistema.

Apesar disso, o desenvolvimento deste trabalho permitiu-nos ter a noção do quão importante é a comunicação dentro do grupo para que um trabalho desta magnitude seja concebível. A maior dificuldade sentida assentou nisso mesmo e na falta de planeamento inicial. Para além disso, sentimos dificuldades em perceber algumas das funcionalidades requeridas pelo grupo que nos forneceu a especificação do sistema, como por exemplo o conceito de "achievement" e a "validação" dos clientes.

Em suma, o grupo reconhece que não teve a melhor prestação na implementação deste sistema, contudo consideramos ter implementado os básicos para garantir um bom funcionamento da aplicação.

Referências

«Thomas Connolly, Carolyn Begg, 2015, Database Systems. 6th ed. Scotland: University of the west of Scotland»

Lista de Siglas e Acrónimos

API Application Programming Interface

IDE Integrated Development Environment

Anexos

Anexo 1

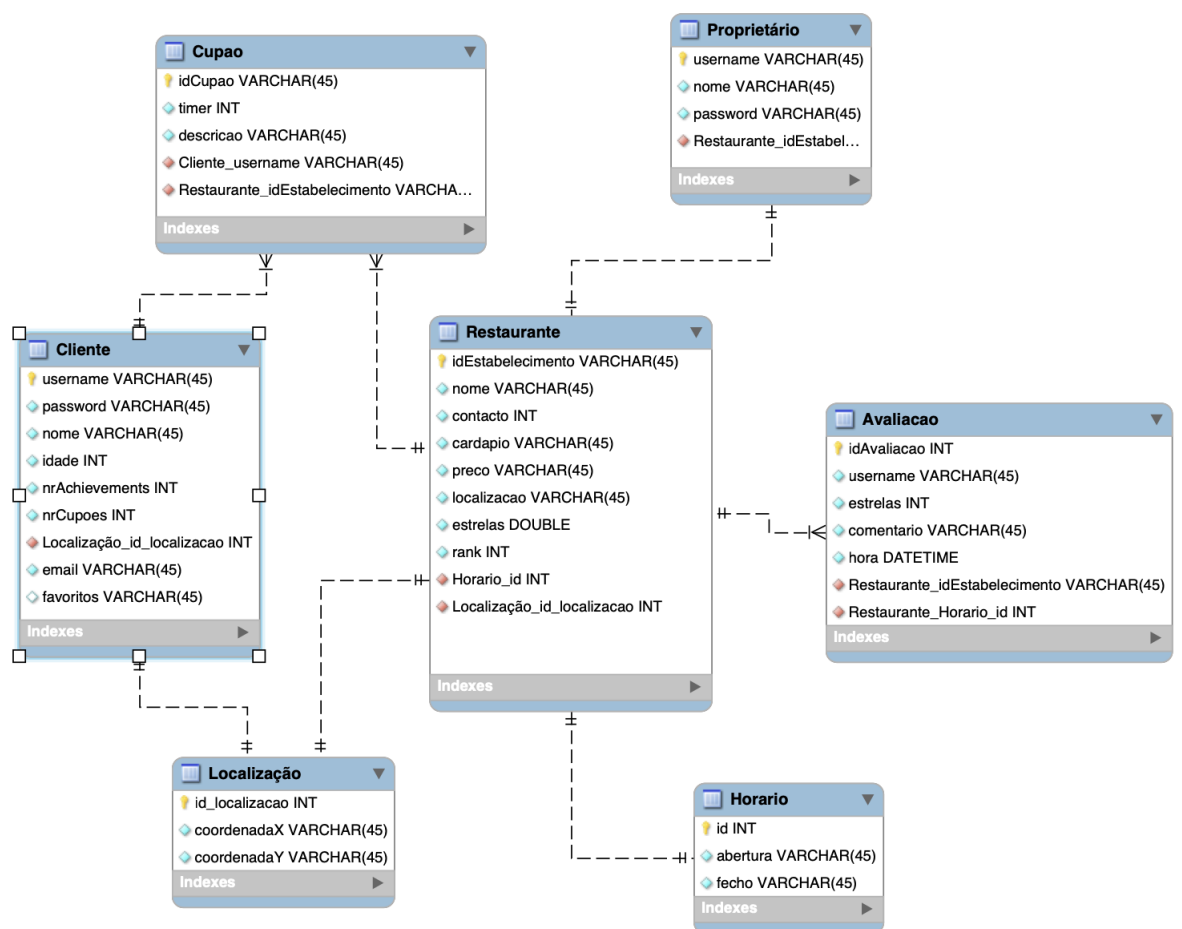


Figura 3.1: Modelo Lógico.

Anexo 2

```
1  -- MySQL Workbench Forward Engineering
2
3  • SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
4  • SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
5  • SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,
6      NO_ZERO_IN_DATE,NO_ZERO_DATE,
7      ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
8
9  -- -----
10 -- Schema mydb
11 -- -----
12
13 -- -----
14 -- Schema mydb
15 -- -----
16 • CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;
17 • USE `mydb` ;
18
19 -- -----
20 -- Table `mydb`.`Localização`
21 -- -----
22 • ⊖ CREATE TABLE IF NOT EXISTS `mydb`.`Localização` (
23     `id_localizacao` INT NOT NULL,
24     `coordenadaX` VARCHAR(45) NOT NULL,
25     `coordenadaY` VARCHAR(45) NOT NULL,
26     PRIMARY KEY (`id_localizacao`))
27 ENGINE = InnoDB;
28
29 -- -----
30 -- Table `mydb`.`Cliente`
31 -- -----
32
33 • ⊖ CREATE TABLE IF NOT EXISTS `mydb`.`Cliente` (
34     `username` VARCHAR(45) NOT NULL,
35     `password` VARCHAR(45) NOT NULL,
36     `nome` VARCHAR(45) NOT NULL,
37     `idade` INT NOT NULL,
38     `nrAchievements` INT NOT NULL,
39     `nrCupoes` INT NOT NULL,
```

```

40     `Localização_id_localizacao` INT NOT NULL,
41     `email` VARCHAR(45) NOT NULL,
42     `favoritos` VARCHAR(45) NULL,
43     PRIMARY KEY (`username`),
44     INDEX `fk_Cliente_Localização1_idx` (`Localização_id_localizacao` ASC) VISIBLE,
45     CONSTRAINT `fk_Cliente_Localização1`
46         FOREIGN KEY (`Localização_id_localizacao`)
47         REFERENCES `mydb`.`Localização` (`id_localizacao`)
48         ON DELETE NO ACTION
49         ON UPDATE NO ACTION)
50     ENGINE = InnoDB;
51
52
53     -----
54     -- Table `mydb`.`Horario`
55     -----
56     ● ⊖ CREATE TABLE IF NOT EXISTS `mydb`.`Horario` (
57         `id` INT NOT NULL,
58         `abertura` VARCHAR(45) NOT NULL,
59         `fecho` VARCHAR(45) NOT NULL,
60         PRIMARY KEY (`id`))
61     ENGINE = InnoDB;
62
63
64     -----
65     -- Table `mydb`.`Restaurante`
66     -----
67     ● ⊖ CREATE TABLE IF NOT EXISTS `mydb`.`Restaurante` (
68         `idEstabelecimento` VARCHAR(45) NOT NULL,
69         `nome` VARCHAR(45) NOT NULL,
70         `contacto` INT NOT NULL,
71         `cardapio` VARCHAR(45) NOT NULL,
72         `preco` VARCHAR(45) NOT NULL,
73         `localizacao` VARCHAR(45) NOT NULL,
74         `estrelas` DOUBLE NOT NULL,
75         `rank` INT NOT NULL,
76         `Horario_id` INT NOT NULL,
77         `Localização_id_localizacao` INT NOT NULL,
78         PRIMARY KEY (`idEstabelecimento`),

```



```

79 INDEX `fk_Restaurante_Horario1_idx` (`Horario_id` ASC) VISIBLE,
80 INDEX `fk_Restaurante_Localização1_idx` (`Localização_id_localizacao` ASC) VISIBLE,
81 CONSTRAINT `fk_Restaurante_Horario1`
82     FOREIGN KEY (`Horario_id`)
83     REFERENCES `mydb`.`Horario` (`id`)
84     ON DELETE NO ACTION
85     ON UPDATE NO ACTION,
86 CONSTRAINT `fk_Restaurante_Localização1`
87     FOREIGN KEY (`Localização_id_localizacao`)
88     REFERENCES `mydb`.`Localização` (`id_localizacao`)
89     ON DELETE NO ACTION
90     ON UPDATE NO ACTION)
91 ENGINE = InnoDB;
92
93
94 -----
95 -- Table `mydb`.`Cupao`
96 -----
97 ● ⊖ CREATE TABLE IF NOT EXISTS `mydb`.`Cupao` (
98     `idCupao` VARCHAR(45) NOT NULL,
99     `timer` INT NOT NULL,
100     `descricao` VARCHAR(45) NOT NULL,
101     `Cliente_username` VARCHAR(45) NOT NULL,
102     `Restaurante_idEstabelecimento` VARCHAR(45) NOT NULL,
103     PRIMARY KEY (`idCupao`),
104 ✖ INDEX `fk_Cupao_Cliente1_idx` (`Cliente_username` ASC) VISIBLE,
105 INDEX `fk_Cupao_Restaurante1_idx` (`Restaurante_idEstabelecimento` ASC) VISIBLE,
106 CONSTRAINT `fk_Cupao_Cliente1`
107     FOREIGN KEY (`Cliente_username`)
108     REFERENCES `mydb`.`Cliente` (`username`)
109     ON DELETE NO ACTION
110     ON UPDATE NO ACTION,
111 CONSTRAINT `fk_Cupao_Restaurante1`
112     FOREIGN KEY (`Restaurante_idEstabelecimento`)
113     REFERENCES `mydb`.`Restaurante` (`idEstabelecimento`)
114     ON DELETE NO ACTION
115     ON UPDATE NO ACTION)
116 ENGINE = InnoDB;
117

```

```

118
119  -- -----
120  -- Table `mydb`.`Proprietário`
121  -- -----
122  CREATE TABLE IF NOT EXISTS `mydb`.`Proprietário` (
123      `username` VARCHAR(45) NOT NULL,
124      `nome` VARCHAR(45) NOT NULL,
125      `password` VARCHAR(45) NOT NULL,
126      `Restaurante_idEstabelecimento` VARCHAR(45) NOT NULL,
127      PRIMARY KEY (`username`),
128      INDEX `fk_Proprietário_Restaurante1_idx` (`Restaurante_idEstabelecimento` ASC) VISIBLE,
129      CONSTRAINT `fk_Proprietário_Restaurante1`
130          FOREIGN KEY (`Restaurante_idEstabelecimento`)
131          REFERENCES `mydb`.`Restaurante` (`idEstabelecimento`)
132          ON DELETE NO ACTION
133          ON UPDATE NO ACTION)
134  ENGINE = InnoDB;
135
136
137  -- -----
138  -- Table `mydb`.`Avaliacao`
139  -- -----
140  CREATE TABLE IF NOT EXISTS `mydb`.`Avaliacao` (
141      `idAvaliacao` INT NOT NULL,
142      `username` VARCHAR(45) NOT NULL,
143      `estrelas` INT NOT NULL,
144      `comentario` VARCHAR(45) NOT NULL,
145      `hora` DATETIME NOT NULL,
146      `Restaurante_idEstabelecimento` VARCHAR(45) NOT NULL,
147      PRIMARY KEY (`idAvaliacao`),
148      INDEX `fk_Avaliacao_Restaurante1_idx` (`Restaurante_idEstabelecimento` ASC) VISIBLE,
149      CONSTRAINT `fk_Avaliacao_Restaurante1`
150          FOREIGN KEY (`Restaurante_idEstabelecimento`)
151          REFERENCES `mydb`.`Restaurante` (`idEstabelecimento`)
152          ON DELETE NO ACTION
153          ON UPDATE NO ACTION)
154  ENGINE = InnoDB;
155
156

```

Figura 3.2: Script SQL da Base de Dados.