



ADEETC – Área Departamental de Engenharia Eletrónica e  
Telecomunicações e de Computadores

LEIM -Licenciatura Engenharia informática e multimédia

## **Sistemas Multimédia Para a Internet**

### **Trabalho prático**

**Turma:**

LEIM-61D

**Trabalho realizado por:**

Pedro Ferreira N°43747

Miguel Silvestre N°45101

Duarte Domingues N°45140

**Docente:**

Carlos Gonçalves

04/07/2021



## Índice de conteúdos

Introdução.....	5
1ªParte .....	7
Descrição do sistema a desenvolver .....	7
Análise comparativa em relação a um outro SGC .....	8
Instagram .....	8
Zomato .....	9
2ª Parte .....	11
Mockups.....	11
Página Principal:.....	11
Entrar/Registo:.....	11
Procura por tipo: .....	12
Página do restaurante:.....	13
About us e footer: .....	14
Modelo EA.....	15
Entidade relativas ao utilizador .....	16
Entidades relativas ao restaurante .....	17
Entidade relativas ao tipo de restaurante .....	18
Entidades relativas à associação utilizador – restaurante .....	18
3ª Parte .....	19
Implementação do Registo/Login e Logout .....	19
Registo.....	19
Login.....	21
Logout .....	22
Upload de conteúdos .....	22
Conteúdos com visibilidade pública e/ou privada .....	25
.....	26
Pesquisa de conteúdos .....	26
Adicionar tipos de comida .....	27
Visualização de conteúdo .....	27

Visualizar e editar perfil .....	29
Gerir utilizadores.....	30
Conclusões .....	31
Bibliografia .....	32

## Índice de figuras

Figura 1-Instagram .....	8
Figura 2-Zomato.....	9
Figura 3-Mockup Página Principal.....	11
Figura 4-Mockup Página de Login/Registo .....	11
Figura 5-Mockup Pesquisa Por Tipo.....	12
Figura 6-Mockup Página do Restaurante.....	13
Figura 7-Mockup Abouts Us e Footer .....	14
Figura 8-Modelo EA.....	15
Figura 9-Entidades user e auth_role.....	16
Figura 10-Entidades restaurante e restaurante_location.....	17
Figura 11- Entidades menu e restaurant_pictures .....	17
Figura 12- Entidades restaurant_type e type .....	18
Figura 13-Entidades user_favorite_res e comment.....	18
Figura 14-Tabela Utilizador .....	19
Figura 15-Validação Por Email .....	19
Figura 16-Exemplo Expressão Regular .....	20
Figura 17-Exemplo de Verificação .....	20
Figura 18-Página de Registo.....	20
Figura 19-Username ou Password inválidos.....	21
Figura 20-Conta inativa ou banida .....	21
Figura 21-Página de Login .....	21
Figura 22- Sessão criada após login .....	22
Figura 23-Sessão destruída após logout .....	22
Figura 24-Tabela para Admins e Simpatizantes.....	22
Figura 25-Tabela para localização do restaurante.....	23
Figura 26-Página Para Adicionar um Restaurante .....	24
Figura 27-Página para Adicionar Imagens do Restaurante.....	24
Figura 28-Tabela para os ID do Restaurante.....	25
Figura 29-Página Para Colocar Restaurantes em Privado ou Público.....	25
Figura 30- Conteúdo Bloqueado Visto Por um Guest.....	26
Figura 31-Pesquisa por Localização e Tipo de Comida .....	26
Figura 32-Inserir Novos Tipos de Comida .....	27
Figura 33-Descrição do Restaurante.....	27
Figura 34- Imagens do Restaurante .....	28
Figura 35-Localização e Contactos do Restaurante .....	28
Figura 36-Tabela para Comentários.....	29
Figura 37-Página para o Perfil do User .....	29
Figura 38-Restaurantes Favoritos do User.....	30
Figura 39- Gerenciamento de Users .....	30



## Introdução

Um Sistema de Gestão de Conteúdos é uma ferramenta utilizada para gerir o conteúdo de sítios web, permitindo que vários colaboradores criem, editem e publiquem conteúdos. O conteúdo de uma SGC é normalmente armazenado num banco de dados e exibido numa camada de apresentação com base num conjunto de modelos.

O sistema de gestão de conteúdos tem a vantagem de permitir utilizadores sem conhecimento de linguagens de programação conseguirem criar e gerir facilmente o seu próprio conteúdo da web. No contexto de uma empresa a autonomia sobre a presença dos conteúdos na internet é total, contudo não dispensa os profissionais de internet.

Um sistema de gerenciamento de conteúdos tem duas componentes principais, respetivamente:

**Um aplicativo de gestão de conteúdo (CMA)** – Componente “front end”, permite os utilizadores criar, editar ou gerir conteúdo na sua aplicação web.

**Um aplicativo de apresentação de conteúdo (CDA)** – Componente “back end”, permite o correto armazenamento a visualização conteúdo inserido no CMA.

Existem sistemas de gestão de conteúdo específicos que abordam diferentes funcionalidades, como por exemplo, a construção de blogs e suporte de redes sociais. Os sistemas de gestão de conteúdo podem abordar diferentes tecnologias utilizadas, como por exemplo, JAVA, MySQL e PHP.





## 1ªParte

### Descrição do sistema a desenvolver

É necessário escolher um conjunto de SGC que permitam implementar um sistema equivalente ao que irá ser desenvolvido na segunda parte do trabalho prático.

O SGC que pretendemos desenvolver na 2ªparte do trabalho consiste num sistema que permite consulta de restaurantes baseado numa localização e categorias de comida/nome de restaurante dados. Os restaurantes seriam inseridos na aplicação pelos simpatizantes/administradores com imagens (espaço, menu, etc.), descrição e localização associada. A informação do conteúdo dos restaurantes é guardada na base de dados e as imagens ficarão guardadas no sistema de ficheiro.

Para guardar informação na base de dados vamos utilizar *phpMyAdmin*. O *phpMyAdmin* é essencialmente uma aplicação web gratuita que permite a administração da base dados *MySQL* na internet.

O nosso sistema admite 4 perfis de utilizadores:

- **Convidado** - este utilizador pode apenas pesquisar/consultar restaurantes baseadas na categoria e localização.
- **Utilizador** - este perfil inclui as funcionalidades do perfil anterior e acrescenta as seguintes:
  - Permite comentar e classificar o restaurante.
- **Simpatizante** - este perfil inclui as funcionalidades do perfil anterior e acrescenta as seguintes:
  - Inserir conteúdos no sistema.
  - Gerir a visibilidade dos conteúdos.
- **Administrador** - este perfil inclui as funcionalidades do perfil anterior e acrescenta as seguintes:
  - Gestão das categorias de conteúdos.
  - Gestão de utilizadores.

## Análise comparativa em relação a um outro SGC

**Instagram** - nesta SGC é focada no perfil dos utilizadores e na interação entre estes. Os conteúdos são inseridos no perfil, sendo o acesso deste conteúdo feito com o acesso a esse mesmo perfil.

Assemelha-se ao nosso trabalho no sentido em que é um sistema de criação de conteúdos pelos utilizadores, utiliza a localização dos utilizadores para sugerir perfis e permite comentários a cada conteúdo.

Como maiores diferenças temos que o Instagram é uma rede social, tendo como foco na interação entre utilizadores enquanto na nossa tem uma funcionalidade de serviço ao utilizador.

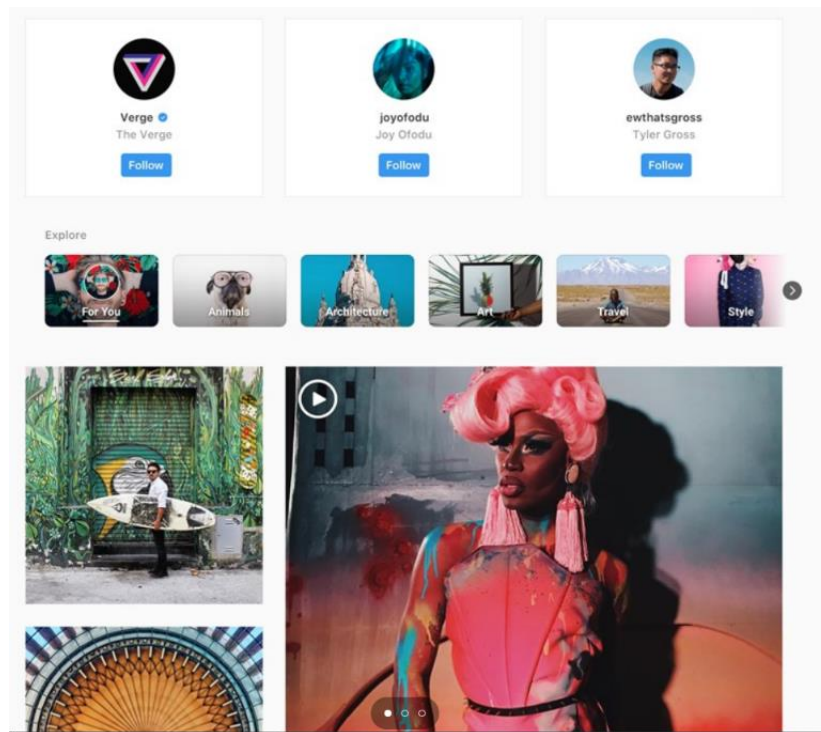


Figura 1-Instagram

Zomato - Este *web service* é bastante semelhante ao nosso, têm o mesmo conceito de pesquisa de serviços de restauração com base nas localizações de restaurantes. O conteúdo também é introduzido.


**zomato** Lisboa  Entrar Registrar

[Início](#) / [Portugal](#) / [Grande Lisboa](#) / [Lisboa](#) / [Restaurantes Bifes](#)


**Restaurantes** Beber um copo Takeaway Entrega ao domicílio

[Filtros](#) [Steak](#) [Ofertas Pro](#) [Distância](#) [Esplanada](#) [Rating: 4.0+](#) [Cafés e pastelarias](#) [Aberto agora](#) [Refeições de luxo](#)


### Bifes - Restaurantes - Lisboa



**Atalho Real**  
★★★★★ 4.6 (3.899 Opiniões)  
Bifes, Hamburgueria  
821 m • Príncipe Real, Lisboa  
45 € para dois  
[Ver todos os estabelecimentos](#)



**K.O.B by Olivier**  
★★★★★ 4.6 (945 Opiniões)  
Bifes  
12 km • Avenida da Liberdade, L...  
80 € para dois • Abre às 19:30



**Brut By Natábua**  
★★★★★ 4.1 (164 Opiniões)  
Bifes, Só bebidas  
8 km • Parque das Nações, Lisb...  
40 € para dois • Abre às 19:00  
[Reservar mesa](#)

Figura 2-Zomato



## 2º Parte

### Mockups

Página Principal:

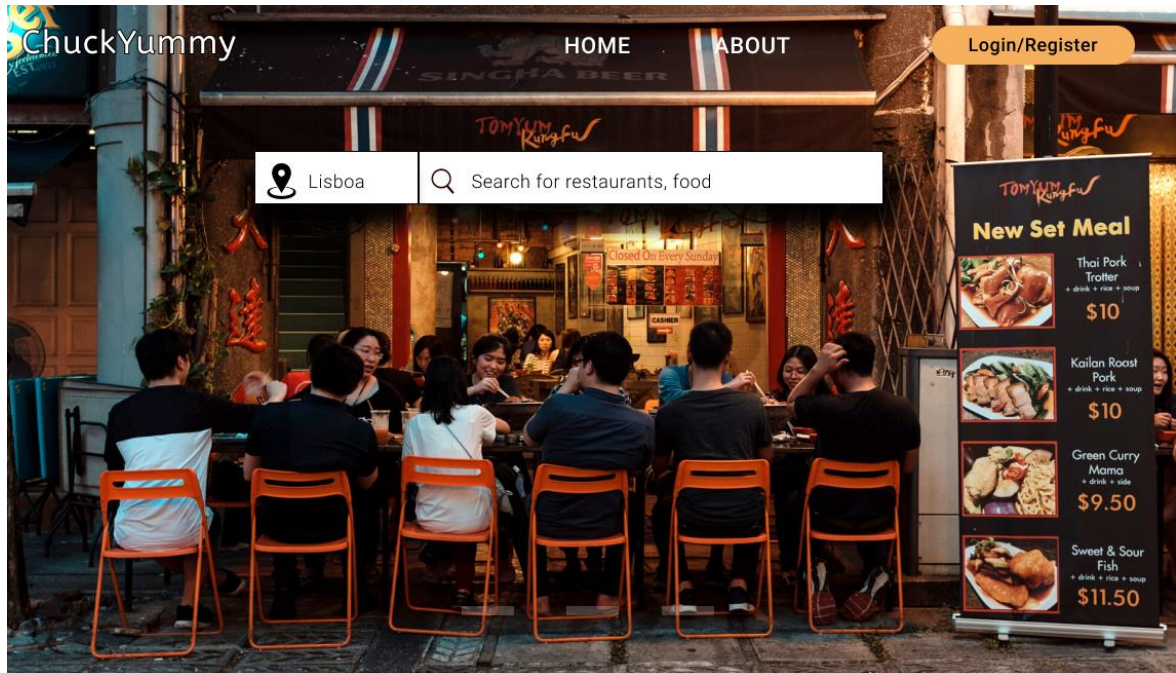




Figura 3-Mockup Página Principal

Entrar/Registo:

**SIGN IN**





or

[New? Create Account](#)




Figura 4-Mockup Página de Login/Registo

Procura por tipo:

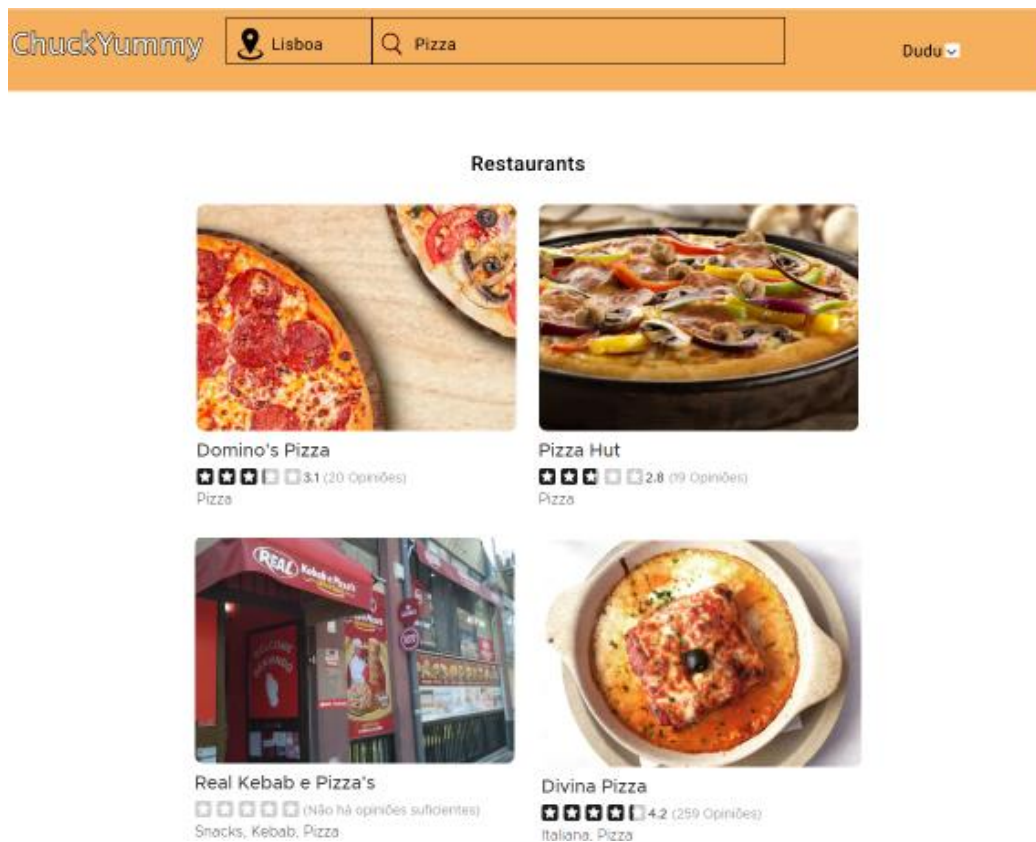






Figura 5-Mockup Pesquisa Por Tipo



Página do restaurante:

**ChuckYummy**  Lisboa    Dudu

### Divine Pizza




#### Pizza meals

Cascais, Madorna  
Open from 10:00 to 23:00 every day

### Menu

<b>Pizza Napolitana</b>	\$7.95
Queijo, tomate, enxofres	
<b>Pizza Fiambre</b>	\$6.95
Queijo, tomate, fiambre	


### Directions



R. Dia Mundial da Criança 21A, 2775-169 Parede

### Rate this restaurant

★★★★★

**Dr. Taveira**

Vim cá na semana especial e ainda bem, caso contrário teria pago demasiado pela refeição. O sabor da sopa é mais de uma moqueca do que da sopa de peixe a que estamos habituados, mas é muito saborosa. A comida é boa (o polvo muito bom, o peixe com as migas nem por isso), mas quase 20 euros por cada prato é um exagero. Assim como também o é pedir 1,70 por cada café. Não voltaria sem ser com estes preços especiais

Figura 6-Mockup Página do Restaurante

About us e footer:

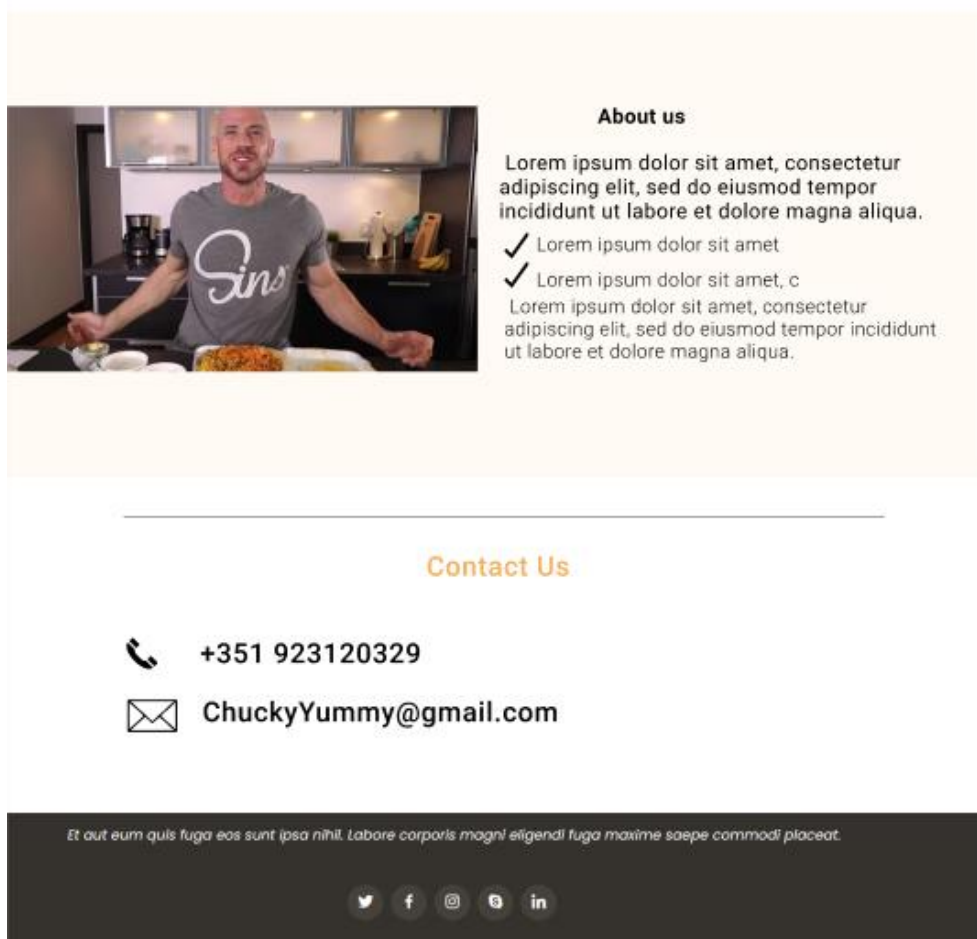


Figura 7-Mockup Abouts Us e Footer



## Modelo EA

Na figura seguinte está representado o modelo entidade associação do nosso trabalho, mais abaixo as entidades e diferentes atributos são explicados em pormenor.

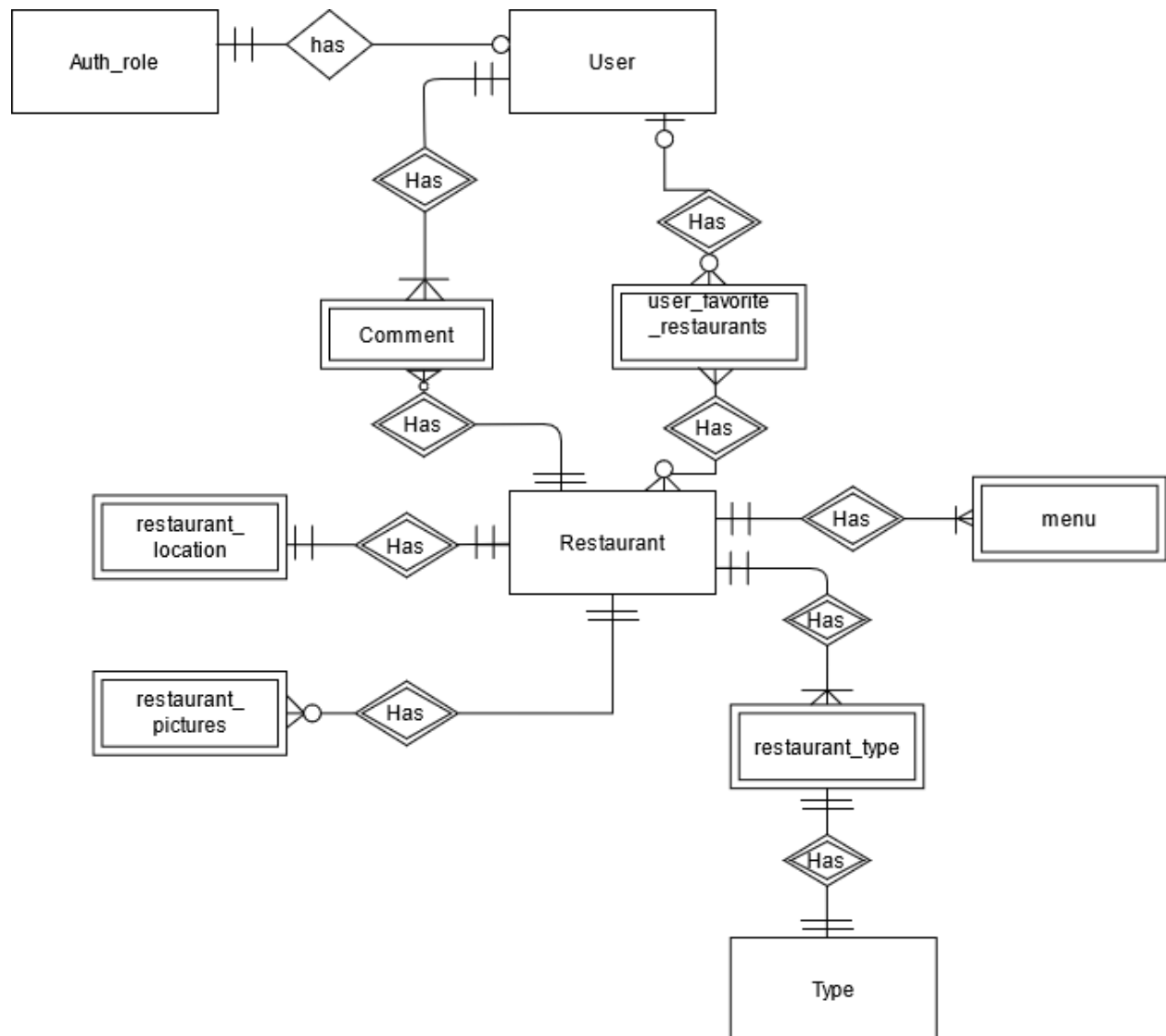


Figura 8-Modelo EA

## Entidade relativas ao utilizador

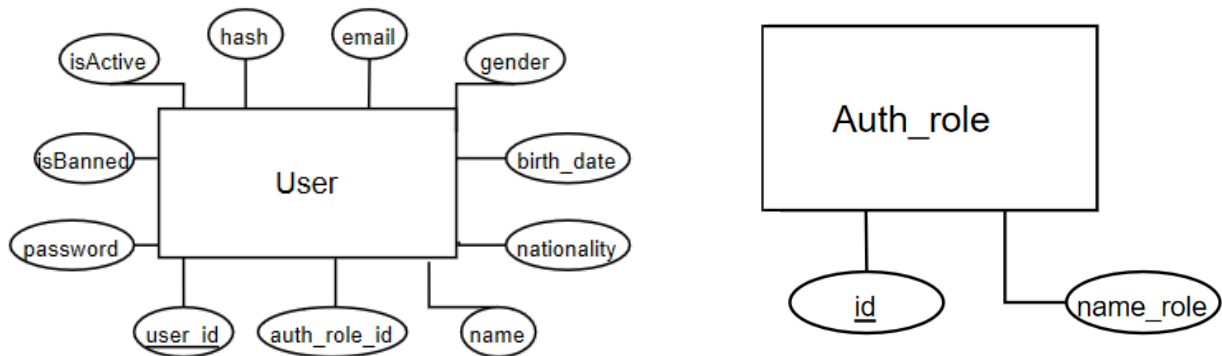


Figura 9-Entidades user e auth\_role

Nas figuras acima estão representadas as entidades “user” e “auth\_role”.

Na entidade *user* está representada a tabela do utilizador, onde estão guardadas as diferentes informações associadas a um utilizador. De notar que o atributo “auth\_role\_id” é uma *foreign key* para a entidade “auth\_role”, o atributo “isActive” serve para definir se a conta do utilizador está ou não ativa, “isBanned” define o estado da conta do utilizador e o “hash” é utilizado no processo de validação de e-mail, tendo cada um valor único para o atributo “hash”.

A entidade “auth\_role” representa o tipo de utilizador, consoante este tipo vai ser definido as diferentes atividades que o utilizador tem acesso no website.

Entidades relativas ao restaurante

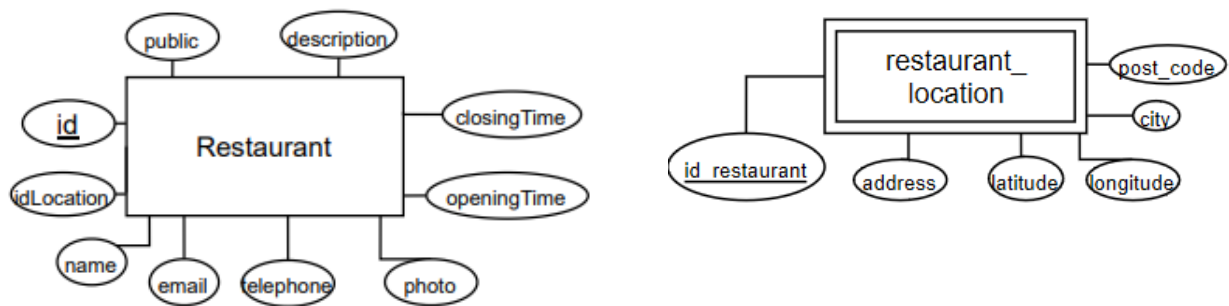


Figura 10-Entidades restaurante e restaurant\_location

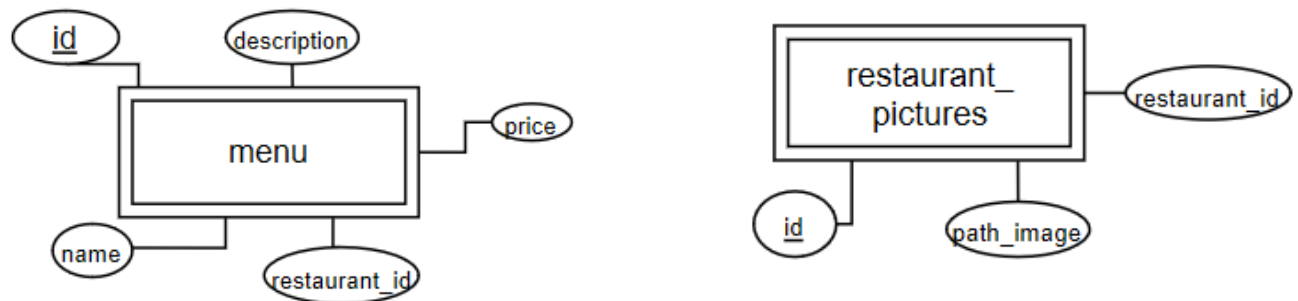


Figura 11- Entidades menu e restaurantPictures

Nas figuras acima estão representadas as entidades relativas aos diferentes dados associados a um restaurante.

Na tabela “*restaurant*” estão representados os campos base relativos a um restaurante, com a adição de uma *foreign key* “*id\_location*” e o atributo “*public*” que serve para definir a visibilidade do restaurante.

Na tabela “*restaurant\_location*” a chave primária é uma chave estrangeira para a tabela do restaurante, e os outros atributos são utilizados para representar a localização real do restaurante.

A tabela menu contém uma chave estrangeira para a tabela do restaurante, e os seguintes campos servem para representar um item da ementa.

Na tabela “*restaurantPictures*” o atributo “*path\_image*” armazena o caminho relativo para uma imagem do restaurante, tendo uma chave estrangeira para relacionar a imagem com o respectivo restaurante.

Entidade relativas ao tipo de restaurante

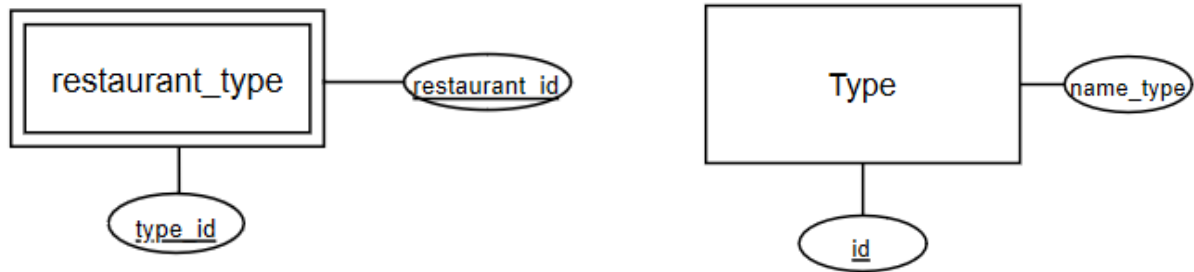


Figura 12- Entidades restaurant\_type e type

Nas figuras acima estão representadas as entidades responsáveis por definir a categoria de comida associada a um restaurante.

Entidades relativas à associação utilizador – restaurante

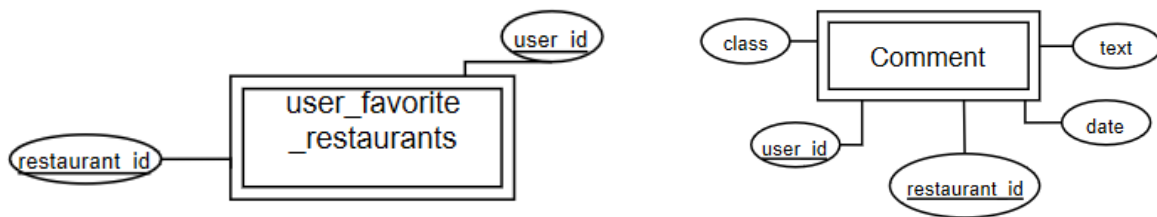


Figura 13-Entidades user\_favorite\_res e comment

O utilizador pode comentar ou adicionar restaurantes a uma lista de favoritos, estas tabelas servem para representar os dados associados com estas ações. O comentário tem associado uma classificação entre 0 e 5. O utilizador só pode comentar uma vez num restaurante de forma a evitar *spam* ou negatividade no website.

Ambas as tabelas têm uma chave primária composta por o identificador do utilizador e do restaurante.

## 3ª Parte

### Implementação do Registo/Login e Logout

#### Registo

Começamos por desenvolver um sistema de registo e login, com validação por e-mail e validação com *capthca*, inspirado nos exemplos fornecidos pelo docente [3].

No registo é pedido o *username*, password, email, sexo e data de nascimento, para isso foi criado a seguinte tabela [Figura 14]:

userid	auth_role_id	name	nationality	birthdate	gender	email	user_hash	password	isBanned	isActive
1	1	Chucky	Portugal	2021-06-13	1	pedrorodrigo1998@gmail.com	b05aef36dfef34e845fa2648ade30c1b	Pppppppppp	0	1
2	1	dudu	Brasil	1999-10-21	1	pedroferreiranunzcrypto@gmail.com	00f8d4370e6a604dd008344d96a79703	Pppppppp	0	1
3	1	Nuno	Belarus	1970-01-01	1	chuckynuno1998@gmail.com	4bdecc2706fb33e53da7e32b7c43c792	P12345678	0	1

Figura 14-Tabela Utilizador

O *user\_hash* serve para efeitos de validação de email [Figura 10], onde cada *user* tem um único *hash*:

Thanks for signing up!  
Your account has been created, you can login with the following credentials after you have activated your account by pressing the url below.

-----  
Username: Chucky  
Password: Pppppppppp  
-----

Please click this link to activate your account:  
<http://localhost/examples-smi/Projecto/verifyRegist.php?hash=b05aef36dfef34e845fa2648ade30c1b>

Figura 15-Validação Por Email

O *isActive* fica a 1 quando o *user* valida o email, podendo apenas dar login nesta situação.

O *isBanned* serve para impedir o login do user caso este seja banido.

O *auth\_role\_id* referencia o id da tabela *auth\_role* onde é representado as roles diferentes que existem:

id	name_role
4	Administrator
1	Guest
3	Sympathizer
2	User

No momento do registo existem duas validações, uma da parte do cliente na linguagem programação *javascript* e outra do lado do servidor em PHP.

A validação do lado do cliente é feita a partir de expressões regulares onde as duas principais são, uma expressão regular para o email e outra para o password. Também existe uma verificação do campo da *password* com o *confirmed password*.

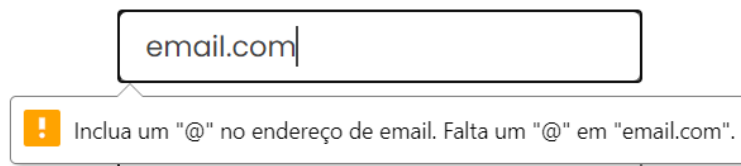


Figura 16-Exemplo Expressão Regular

A validação do lado do servidor é igual à validação do lado do cliente, com a diferença de ser no servidor, verificação de um utilizador com o mesmo *username* ou email e também existe uma validação extra de prevenção contra robots. A validação do lado do servidor é fundamental devido aos utilizadores mal-intencionados que querem sobrelotar o website.

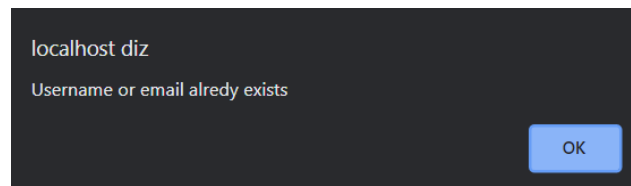


Figura 17-Exemplo de Verificação

A screenshot of a web registration page for a site called "ChuckYummy". The page has a dark header with the site name and navigation links for "Home", "About", and a "LOGIN" button. The main heading is "Create account". Below this, there is a registration form with the following fields: "Username", "Your Email", "Password", "Password Confirm", a "Gender" dropdown menu (set to "Male"), an "Age" field with a date input format "dd/mm/aaaa", a CAPTCHA image showing the number "a319545b", and a "Type your captcha" text input. At the bottom of the form is a "Sign up" button. Below the button, there is a link: "Already have an account ? [Login here](#)".

Figura 18-Página de Registo

## Login

A validação do login é feita pelo servidor onde se verifica se os dados correspondem aos existentes na base de dados e se a conta esta ativada e não esta banida [Figura 19] [Figura 20].

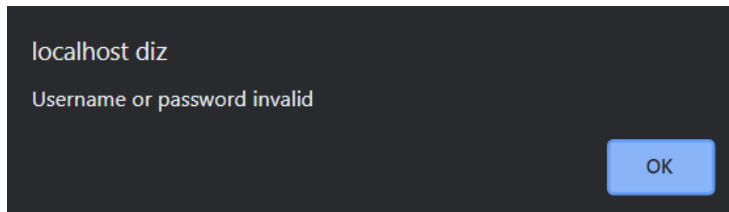


Figura 19-Username ou Password inválidos

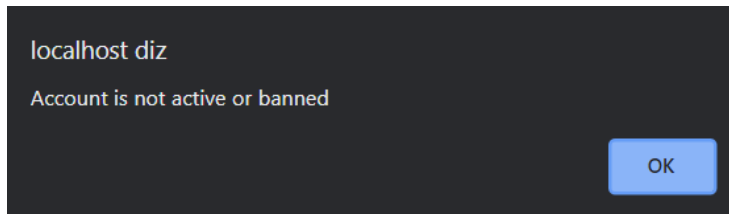


Figura 20-Conta inativa ou banida

Na seguinte figura [Figura 21] o utilizador preenche os campos para realizar login, baseado em forms para submeter informação. [2]

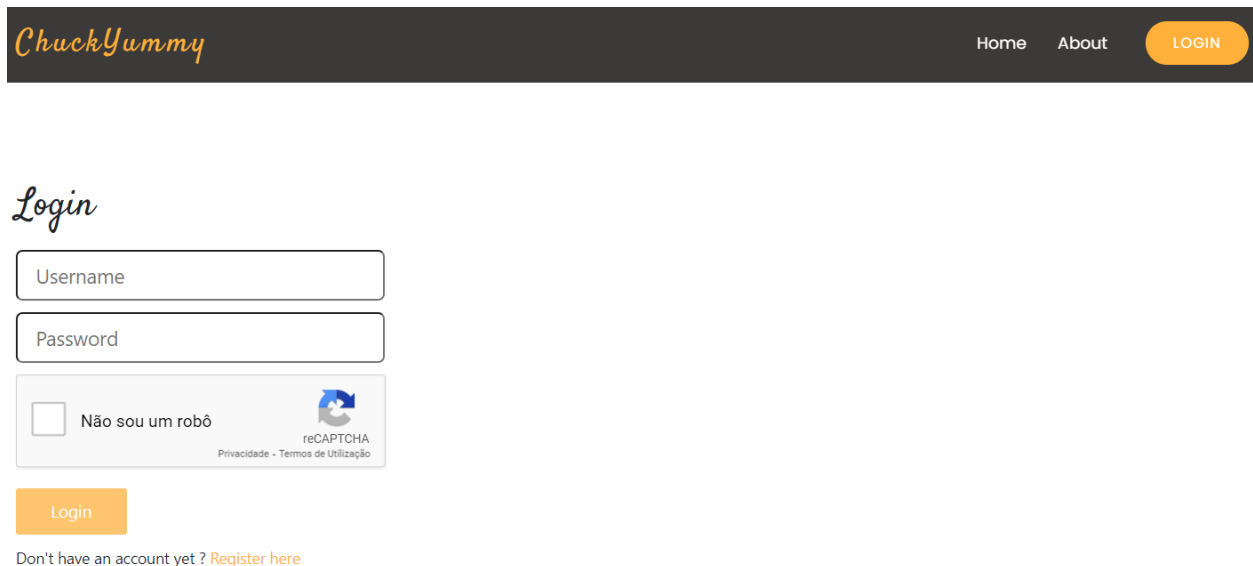
The image shows the login page of a website called "ChuckYummy". At the top, there is a dark navigation bar with the site's logo on the left and links for "Home", "About", and a yellow "LOGIN" button on the right. Below the navigation bar, the word "Login" is written in a large, elegant script font. Underneath, there are two white input fields for "Username" and "Password". Below these fields is a reCAPTCHA widget with the text "Não sou um robô" and a checkbox. At the bottom of the form area is a yellow "Login" button. Below the button, there is a link that says "Don't have an account yet ? Register here".

Figura 21-Página de Login

Quando o login é realizado com sucesso é criada uma sessão [Figura 23], com informação relativa ao *user*:

```

if (session_status() !== PHP_SESSION_ACTIVE) {session_start()}

$_SESSION["userId"] = $row["userId"];
$_SESSION["authRoleId"] = $row["auth_role_id"];
$_SESSION["name"] = $row["name"];

```

Figura 22- Sessão criada após login

## Logout

A ideia por de trás do logout é o término de sessão no PHP, que resulta na destruição dos valores de *username*, *email* e *password* [Figura 18].

```

if( isset($_POST["submitLogoutForm"])) {
    session_destroy();
    header("Location: http://localhost/examples-smi/Projecto/index.php");
    exit();
}

```

Figura 23-Sessão destruída após logout

## Upload de conteúdos

Quando um utilizador é simpatizante ou administrador, este consegue introduzir restaurantes na base de dados, para estes poderem ser visualizados no site.

Para isso foi necessário criar a seguinte tabela [Figura 24]:

id	name	email	telephone	photo	location_id	public	opening_time	closing_time	description
5	fifities	fif@email.com	219999999	dudu/back3.jfif	1	0	09:30	20:30	Prazer e Diversão sem limites. Venha conhecer o no...
6	duduChurrasco	churracos@email.com	219999999	dudu/back3.jfif	1	1	09:00	10:00	Prazer e Diversão sem limites. Venha conhecer o no...

Figura 24-Tabela para Admins e Simpatizantes

O atributo *location\_id* referencia o id da tabela *restaurant\_location* [Figura 25] onde é representada a localização:



idRestaurant	address	latitude	longitude	City	post_code
5	R. Catarina Eufémia G98	38.838917	-9.101075	Santa Iria de Azoia	2690
6	R. da Carochia 113	38.809036	-9.190130	Ramada	2620-206

Figura 25-Tabela para localização do restaurante

## Add Restaurant

ChurrascariaManel

chumanel@email.com

210000000

Os melhores frangos do mundo.

### Select restaurant picture cover:

Escolher ficheiro back5.jfif

Add new item to menu +

1/2 frango	5	barato	
1 frango	price 2	descrip 2	Delete

Choose food types:

- ☒ BBQ
- ☒ Bebida
- ☒ Carne
- ☐ Massa
- ☐ Peixe
- ☐ Pizza
- ☐ Sushi
- ☐ Vegan

Choose opening time:

11:30 ⌚

Choose closing time:

23:00 ⌚



Figura 26-Página Para Adicionar um Restaurante

O utilizador tem que introduzir informações do restaurante como: [2] nome, email, telefone, descrição, imagem principal, menu, tipos de comida, horário e localização.

De notar que quando é feita a seleção da imagem principal, é criado um ficheiro com o nome do *user* e dentro deste é onde se ira encontrar a imagem selecionado, é este o *path* que é guardado na base de dados. [4]

Os tipos de comida são pré-definidos e são mostrados dinamicamente através da base de dados.

Na localização, o utilizador realiza um clique no mapa onde pretende registar o restaurante e automaticamente é guardado na base de dados, a morada, latitude, longitude e código postal.

De seguida o utilizar é redirecionado para a página de adicionar imagens representativas do restaurante (pratos, esplanada, ...)[6] [Figura 27]:

Select more restaurants pictures:

Nenhum fiche...o selecionado



Figura 27-Página para Adicionar Imagens do Restaurante

O *path* destas imagens são inseridos na tabela *restaurant\_pictures* [Figura 28]:

restaurant_id	path_image
9	dudu/back1.jpg
9	dudu/back2.jfif
9	dudu/back3.jfif
9	dudu/back1.jpg
9	dudu/back2.jfif
9	dudu/back5.jfif

Figura 28-Tabela para os ID do Restaurante

## Conteúdos com visibilidade pública e/ou privada

Quando um simpatizante ou administrador visualiza os restaurantes este tem a opção de o colocar privado ou público [Figura 29]:

### See Restaurants









fifities	duduChurrasco	cac	Pedro Ferreira
			
Santa Iria de Azoia	Ramada	Santa Ant3nio dos Cavaleiros	Madorna
Carne, BBQ	Carne, BBQ	Carne, BBQ	Carne, BBQ
☆☆☆☆☆	☆☆☆☆☆	☆☆☆☆☆	☆☆☆
			

Figura 29-Página Para Colocar Restaurantes em Privado ou Público

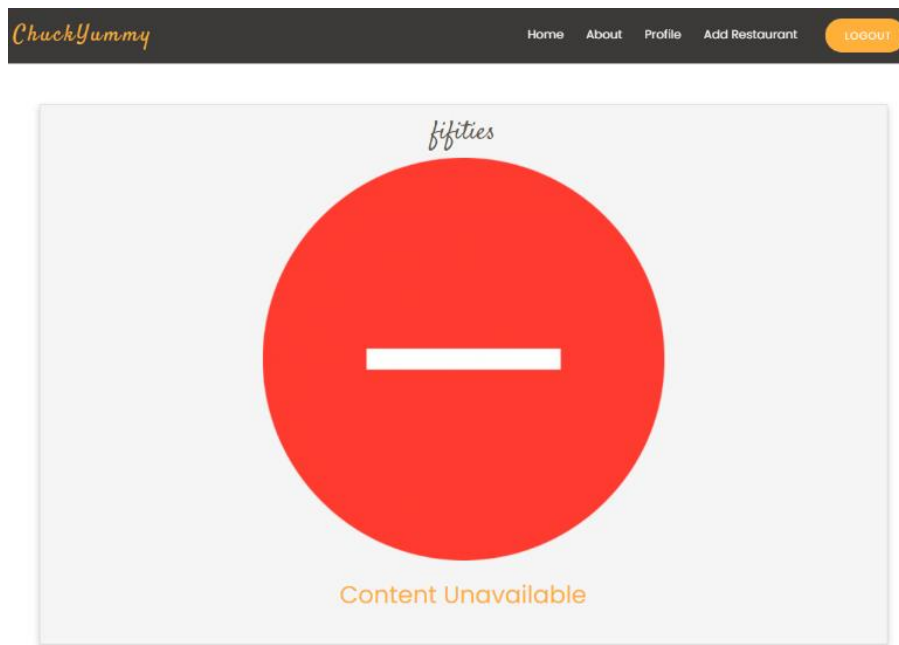


Figura 30- Conteúdo Bloqueado Visto Por um Guest

## Pesquisa de conteúdos

A pesquisa de restaurantes é feita com um auxílio de um *auto-complete* das localizações, nomes e tipos de comida dos restaurantes que existem na base de dados.

Quando se seleciona uma localização, apenas aparecem nomes e tipos de comida de restaurantes relativas a essa localização. Isto é feito através da possibilidade de guardar a localização, e só depois é que são sugeridos os restaurantes. Uma alternativa seria usar AJAX, o que simplifica a interação de pesquisa pois não necessitaríamos de um botão para guardar a localização introduzida. [5]

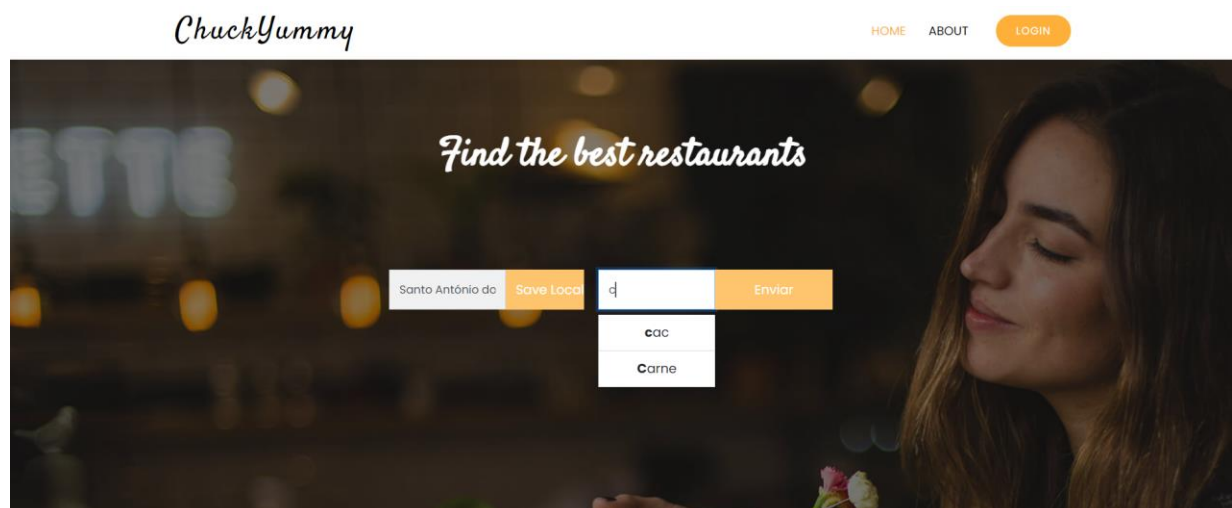


Figura 31-Pesquisa por Localização e Tipo de Comida

## Adicionar tipos de comida

Um administrador pode visualizar os tipos de comida existentes e adicionar novos [Figura 32]:

### Add Types

BBQ

Bebida

Carne

Massa

Peixe

Pizza

Sushi

Vegan

type 1

Delete

type 2

Delete

Add

Figura 32-Inserir Novos Tipos de Comida

## Visualização de conteúdo

Aqui é possível visualizar o nome do restaurante, a imagem principal, morada, tipos de comida, horário e classificação [Figura 33].

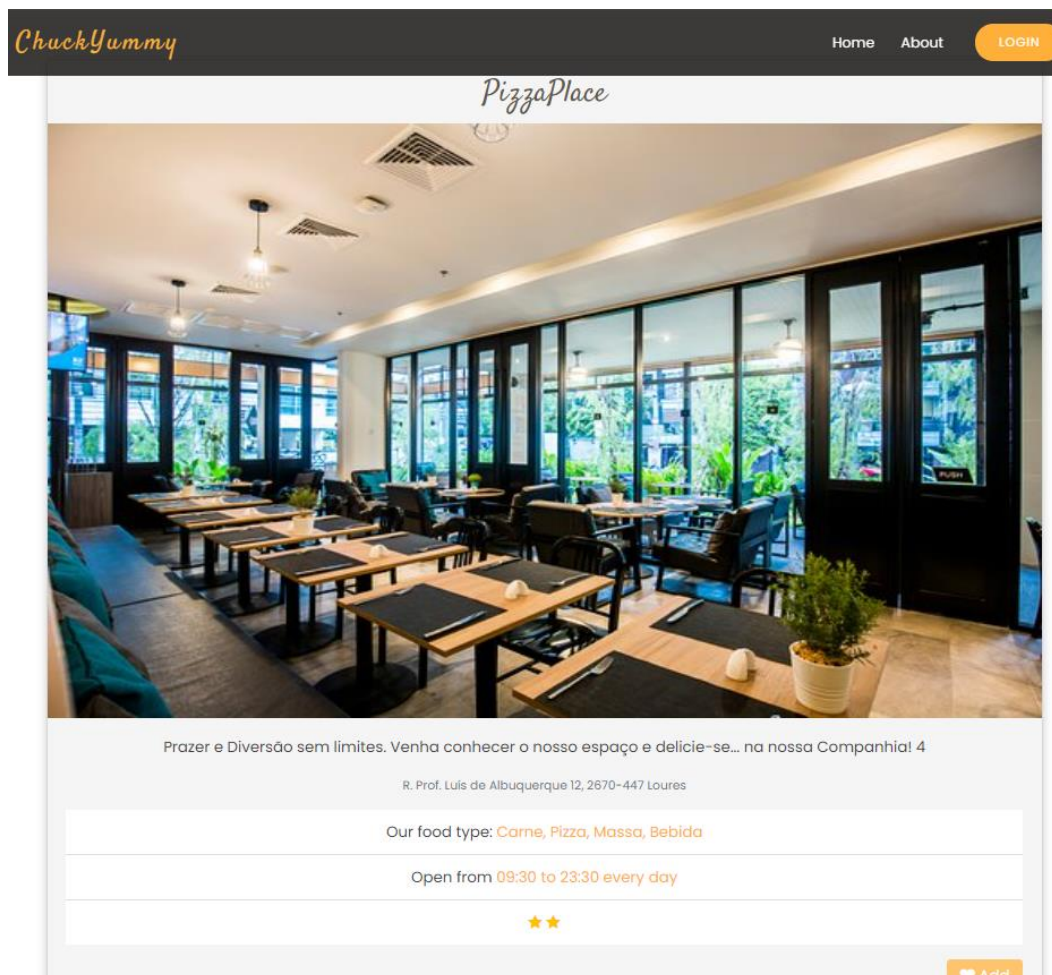


Figura 33-Descrição do Restaurante



Aqui encontram se as imagens relativas ao restaurante [Figura 34].

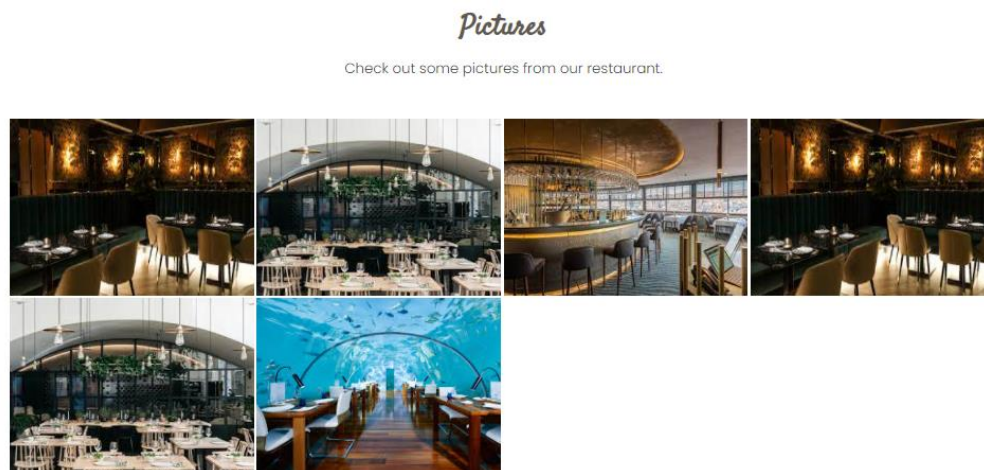


Figura 34- Imagens do Restaurante

Nesta parte são representados o menu, a localização e os contactos [Figura 35].

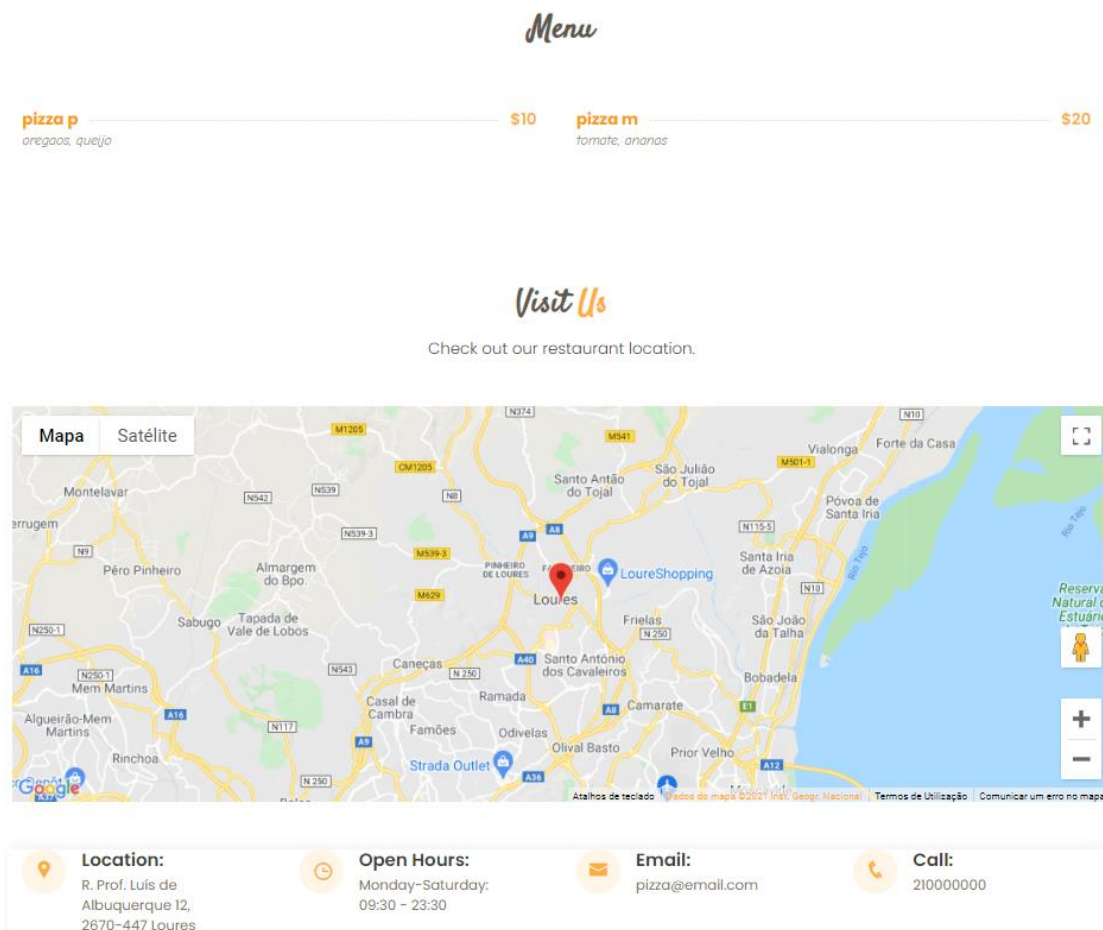


Figura 35-Localização e Contactos do Restaurante

Quando é feito um comentário este é guardado na tabela *comment* [Figura 36]:

text	user_id	restaurant_id	date_comment	class
ola	2	8	2021-06-30	2

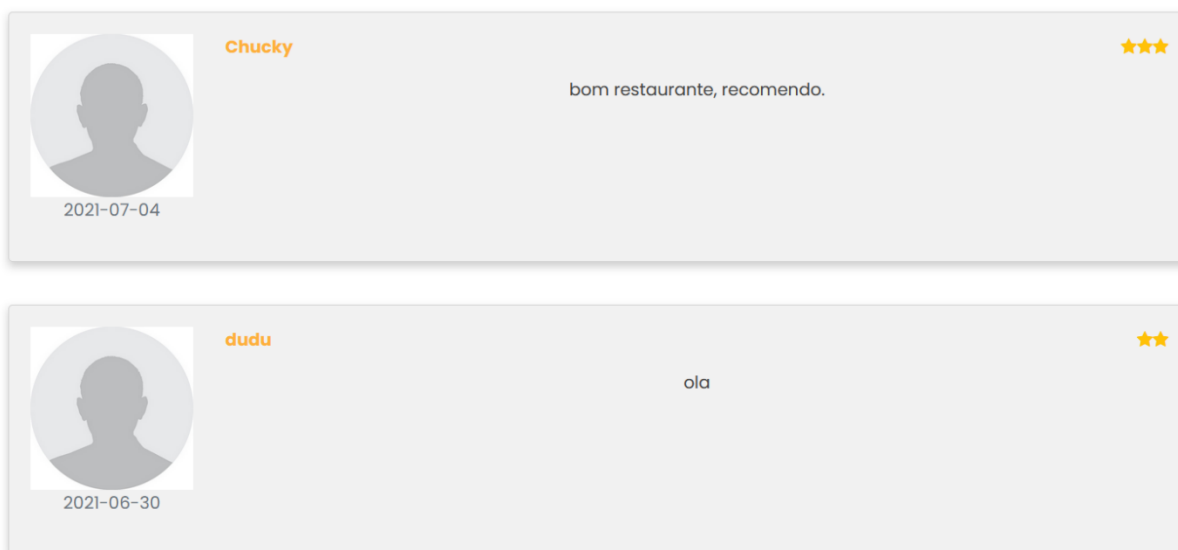


Figura 36-Tabela para Comentários

## Visualizar e editar perfil

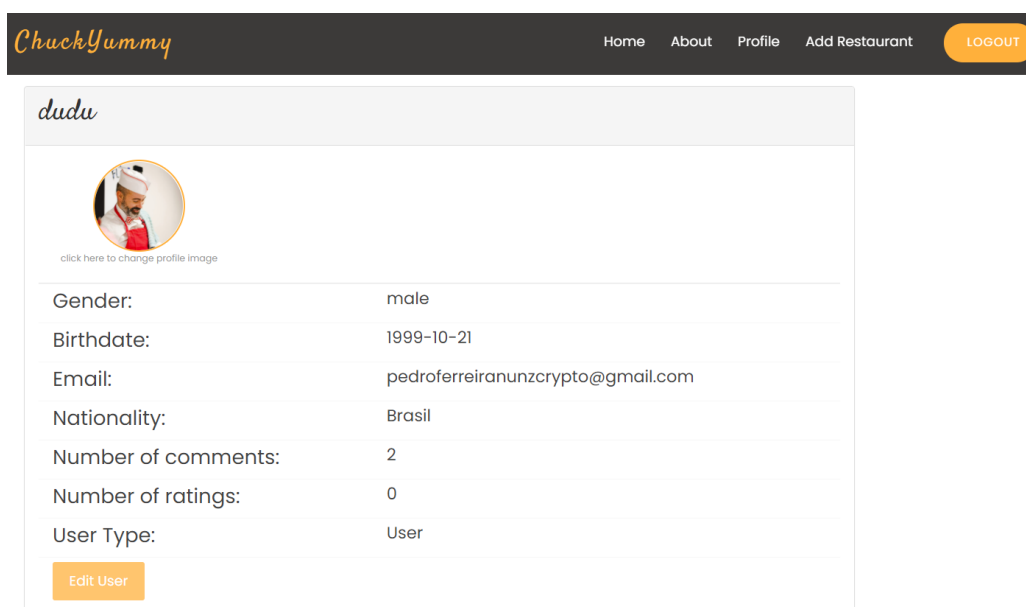


Figura 37-Página para o Perfil do User

Quando um *user* dentro de um restaurante adiciona aos favoritos, todos os restaurantes favoritos do *user* são apresentados no perfil [Figura 38]:

### *Favorite Restaurants*



Figura 38-Restaurantes Favoritos do User

## Gerir utilizadores

Nesta página o administrador pode gerir os utilizadores, nomeadamente as suas roles e bloquear ou desbloquear um *user* [Figura 39]:

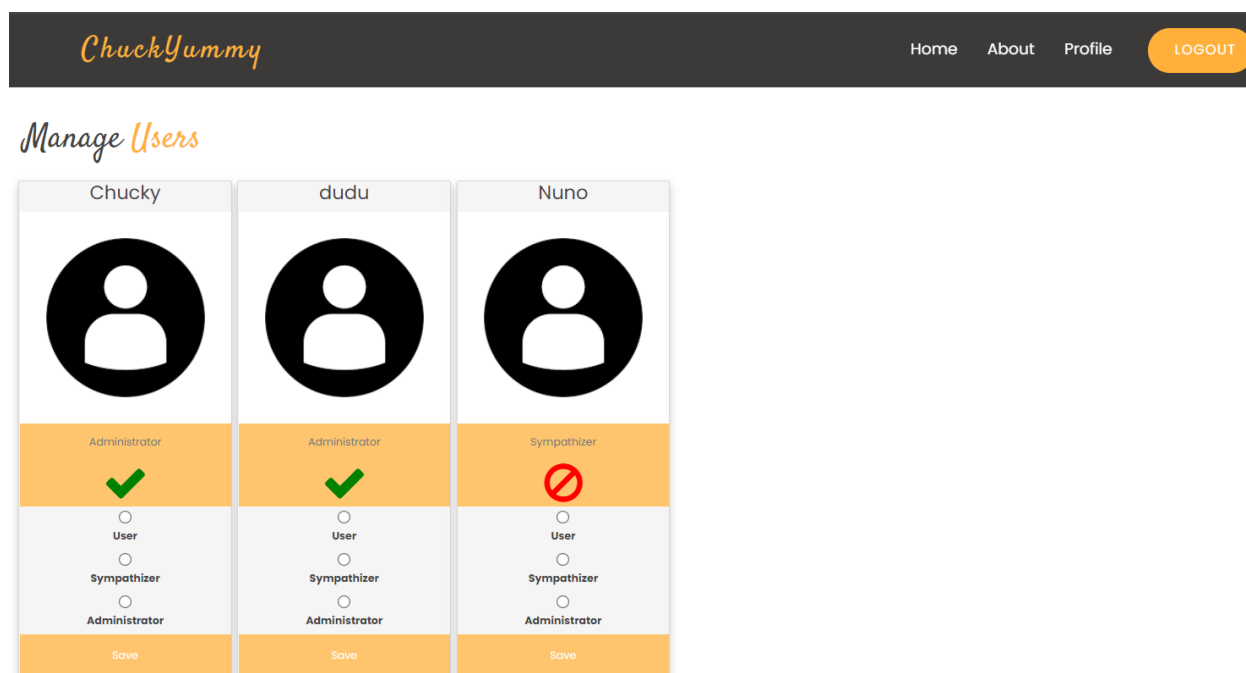


Figura 39- Gerenciamento de Users



## Conclusões

Com este trabalho de sistemas multimédia para a internet o grupo desenvolveu competências de: consolidação de conceitos de *Web Services*, as vantagens de criação de páginas dinâmicas na web, interligação de aplicações Web com sistemas externos como base de dados, estruturação de aplicações distribuídas segundo o modelo cliente/servidor e estruturação de aplicações Web dinâmicas e Sistemas de Gestão de Conteúdos bem como a consolidação dos conceitos da linguagem PHP.

Foi possível criar um local de partilhas de conteúdos que neste caso é referente a restaurantes. Achámos que conseguimos cumprir os principais objetivos do trabalho inicialmente estudados na 1ª parte.

Cumprimos com sucesso os seguintes objetivos:

- Existe um registo automático com validação de email de utilizadores na base de dados do *phpMyAdmin*.
- Login/logout.
- Validação "Não sou um robot" no registo de utilizadores.
- Upload e visualização de conteúdos.
- Utilização de serviços externos (Google Maps), para guardar a localização de um conteúdo.
- Os conteúdos possuem visibilidade publica ou privada, onde somente utilizadores logados têm acesso.
- Validação de dados no lado do cliente.
- Validação de dados no lado do servidor.
- Possibilidade de modificar a visibilidade dos conteúdos.
- Gestão de utilizadores.
- Gestão de categorias.
- Utilização de CSS.
- Grafismo apelativo.
- Possibilidade de efetuar pesquisas sobre os conteúdos e/ou eventos com sistema de auto complete complexo.
- Utilização de *Bootstrap*.
- Apresentação de conteúdos com base em tipos de comida e favoritos.

Pontos em falta ou que poderiam ser melhorados no futuro:

- Utilização de AJAX para melhorar a experiência do utilizador nomeadamente quando é realizada a pesquisa dum conteúdo.
- Download e upload de conteúdos por lote.
- Notificação de eventos por email com base em tipos de comida favoritos.
- Criação de categorias de comida secundarias.
- Foto de perfil de *user*.
- Possibilidade de um alterar um conteúdo.

## Bibliografia

- [1] – Carlos Gonçalves, Sistemas Multimédia para a Internet, 03-DistributedSystems, 2021
- [2] – Carlos Gonçalves, Sistemas Multimédia para a Internet, 08-FormsHTML, 2021
- [3] – Carlos Gonçalves, Sistemas Multimédia para a Internet, 09-Authentication-DataBase, 2021
- [4] – Carlos Gonçalves, Sistemas Multimédia para a Internet, 10-Images-DataBase, 2021
- [5] – *Submit page without page refresh*, disponível em: <https://tecadmin.net/submit-form-without-page-refresh-php-jquery/>
- [6] - PHP 8 Multiple Files/Images Upload in MySQL Database: <https://www.positronx.io/php-multiple-files-images-upload-in-mysql-database/>