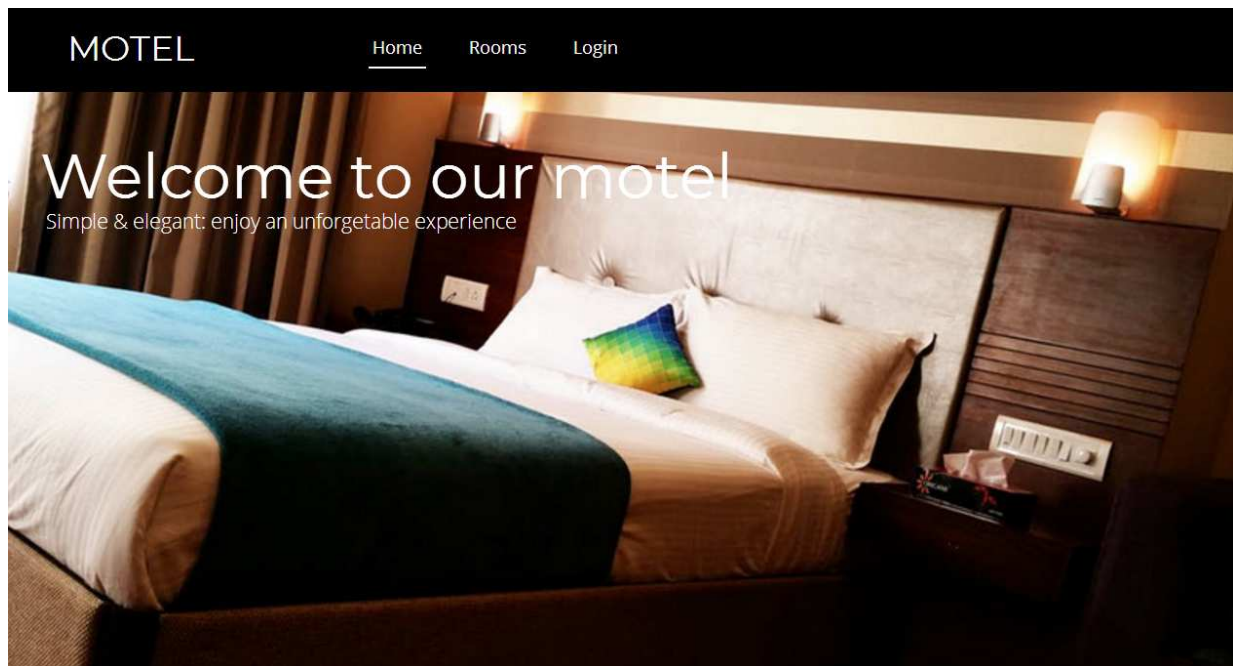


LER O ENUNCIADO ATÉ AO FIM ANTES DE COMEÇAR!

O objectivo do exame é adicionar funcionalidades a um site web suportado por uma base de dados relacional. Informação sobre a base de dados encontra-se em ANEXO.



O site web consiste no portal de um motel que permite fazer reservas online de quartos a utilizadores registados. O site tem as seguintes páginas:

1. “**Home**” é página de entrada no site;
2. “**Rooms**” apresenta os quartos do motel;
3. “**Details**” apresenta as características de um quarto em particular, e permite a sua reserva a utilizadores registados;
4. “**Register**” para registo de utilizadores;
5. “**Login**” para o login de utilizadores;
6. “**My Bookings**” permite aos utilizadores registados consultar uma listagem das suas reservas

É dado acesso a uma implementação funcional do site web, realizado em *CodeIgniter* e *Smarty*, e apenas se solicita ao aluno a programação das funcionalidades em falta.

PRELIMINARES

A. Faça login por ssh (com o PuTTY, por exemplo) no servidor com o IP 10.10.23.183

```
a12345@daw:~$
```

B. Faça download para a pasta “RECURSO”*, do código do site web arquivado em “Github” com o comando “git”

```
a12345@daw:~$
```

```
git clone https://github.com/jmatbastos/RECURSO.git ~/public_html/RECURSO
```

C. Ponha as permissões de escrita na pasta `template_c`:

```
chmod -R g+w ~/public_html/RECURSO/application/views/templates_c
```

D. Actualize a base URL in `config.php`

```
a12345@daw:~$ nano ~/public_html/RECURSO/application/config/config.php
```

```
$config['base_url'] = '~/a12345/RECURSO';
```

NOTAS:

- **substitua “12345” pelo seu número de aluno!**
- O seu site deve ser portátil; utilize sempre as funções `base_url()` e `site_url()` para gerar hyperlinks!

E. Crie a sua cópia da base de dados e actualize as credenciais em `database.php`

```
a12345@daw:~$
```

```
mysql -u a12345 -p db_a12345 < ~/public_html/RECURSO/database.SQL
```

```
a12345@daw:~$ nano ~/public_html/RECURSO/application/config/database.php
```

```
$db['default'] = array(  
    'dsn' => '',  
    'hostname' => '10.10.23.183',  
    'username' => 'a12345',  
    'password' => '*****',  
    'database' => 'db_a12345',  
    'dbdriver' => 'mysqli',
```

* a pasta “RECURSO” é criada automaticamente pelo comando `git clone`

NOTAS:

- substitua “12345” pelo seu número de aluno!
- Se **não** se recorda da password da sua base de dados, recupere-a com o comando

```
a12345@daw:~$ /usr/local/bin/mysql-db
```

- Se tiver **dificuldade** em criar a base de dados em linha de commando pode utilizar o acesso web **phpMyAdmin** e os comandos SQL descritos no APÊNDICE 2

F. Verifique que o site é visível no URL

```
http://10.10.23.183/~a12345/RECURSO/index.php/motel
```

NOTA: coloque o seu código no final dos ficheiros a seguir ao comentário
//PONHA O SEU CODIGO ABAIXO DESTA LINHA

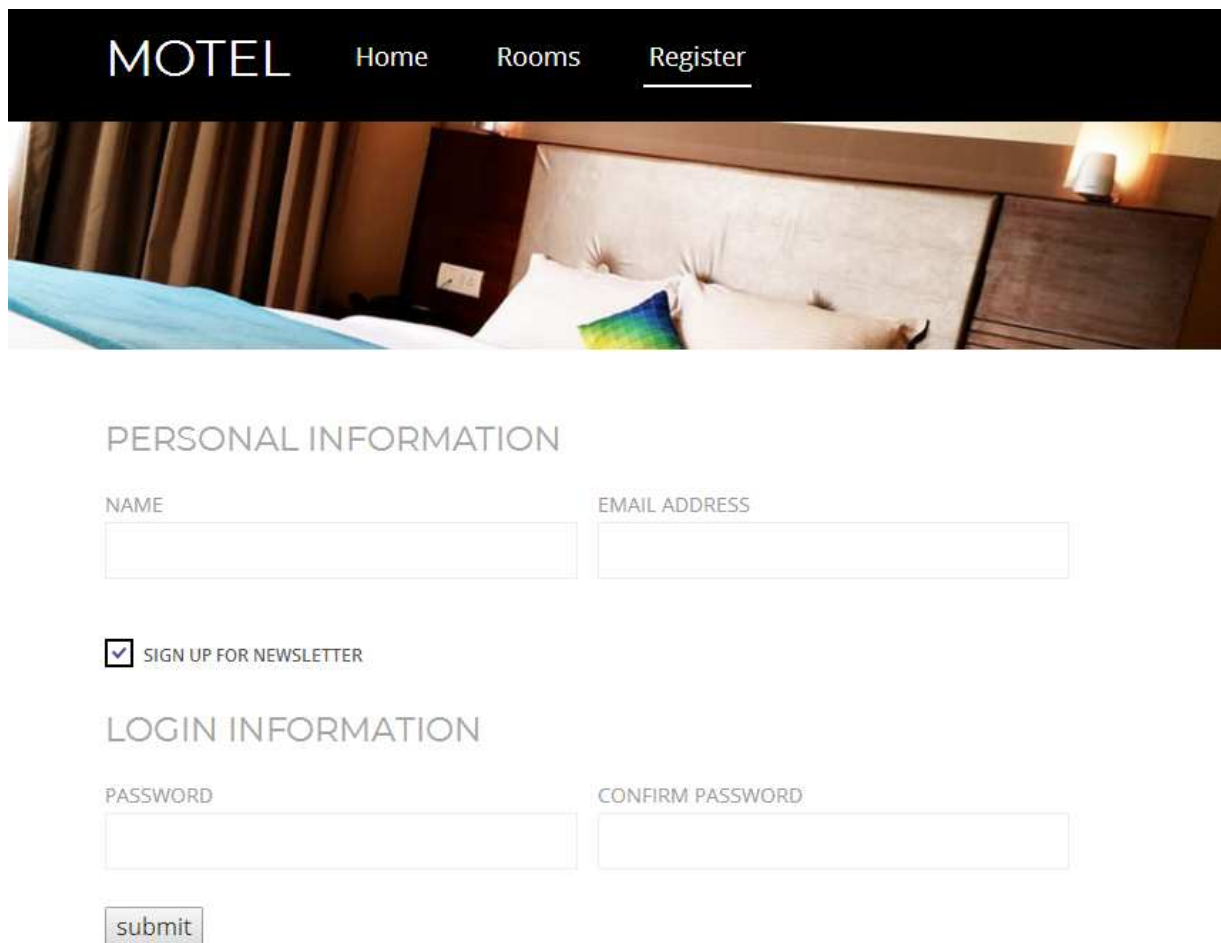
FUNCIONALIDADE “REGISTER”

A funcionalidade “Register” permite registar um utilizador .

1. [5 valores]

Construa o template Smarty “register.tpl”.

SUGESTÃO: Adapte o template “register.html” fornecido pela empresa de web design. Encontra este template na pasta “views/templates”



MOTEL Home Rooms Register

PERSONAL INFORMATION

NAME EMAIL ADDRESS

SIGN UP FOR NEWSLETTER

LOGIN INFORMATION

PASSWORD CONFIRM PASSWORD

submit

Construa o método `register()` no controlador `Motel.php`

- Se todos os campos obrigatórios não estão preenchidos, redirecciona novamente para o formulário com uma mensagem de erro
- Se todos os campos obrigatórios estão correctamente preenchidos, e o email ainda não existe na base de dados, regista o utilizador na tabela “users” utilizando a query correspondente no controlador `Motel_model.php`.

FUNCIONALIDADE “BOOK”

A funcionalidade “Book”, associada à página “Details” permite a um utilizador registado reservar um quarto.

Reservations
November 2018

today < >

Sun	Mon	Tue	Wed	Thu	Fri	Sat	
28	29	30	31	1	2	3	
4	5	6	7	8	9	10	
11	12	13 RESERVED	14	15	16	17	
18	19	20	21 RESERVED	22	23	24	
25	26	27	28	29	30	1	
2	3	4	5	6	7	8	

CHECK IN
2018-11-28

NUMBER OF NIGHTS
01

SPECIAL REQUESTS

RESERVE

2. [2 valores]

Construa o método `book()` no controlador `Motel.php` e a query correspondente em `Motel_model.php`

- Se os campos obrigatórios “Check in” e “Number of nights” não estão preenchidos, redirecciona novamente para o formulário com uma mensagem de erro
- Se os campos obrigatórios estão correctamente preenchidos, regista a reserva na tabela “bookings” (Nota: o calendario de reservas à esquerda do formulário é actualizado automaticamente por uma função AJAX)

3. [4 valores]

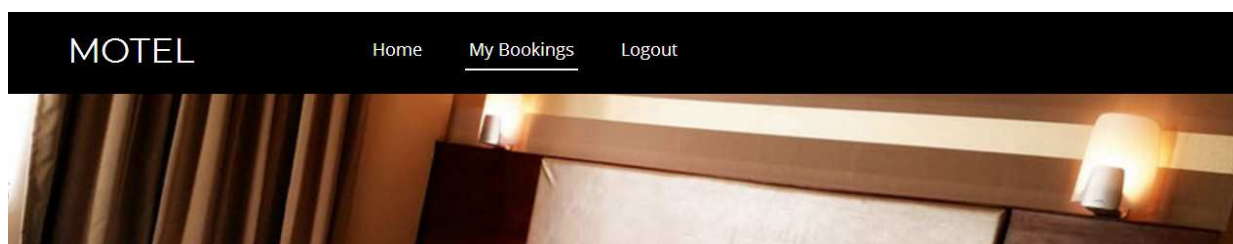
Na alinea anterior não está considerada a possibilidade de “overbooking”. Melhore o método `book()` no controlador `Motel.php` de forma a garantir a **impossibilidade** de haver mais do que uma reserva no mesmo dia para o mesmo quarto.

NOTA: basta haver sobreposição de um dia na tentativa de reserva para que esta não se efectue na totalidade.

- No caso de impossibilidade de reserva, deve aparecer uma mensagem de aviso por cima do campo “Check in”

FUNCIONALIDADE “MY BOOKINGS”

A página “My Bookings” permite a um utilizador registado consultar uma listagem com as suas reservas



My Bookings

#	Booking date	Check in	Check out	Nº of Nights	Room	Requests	Total Price
1	2018-11-23 11:52:27	2018-11-29	2018-12-01	2	Presidential Room	baby bed	600
2	2018-11-23 16:52:51	2018-11-20	2018-11-21	1	Presidential Room		300
3	2018-11-23 17:52:51	2018-11-13	2018-11-14	1	Bachelor Room		50
4	2018-11-23 18:52:51	2018-12-11	2018-12-13	2	Family Room		200
5	2018-11-23 19:52:51	2018-11-21	2018-11-22	1	Bachelor Room	big bed	50

4. [7 valores]

Construa o template Smarty “mybookings.tpl”.

SUGESTÃO: Tem à sua disposição o template “shortcodes.html” fornecido pela empresa de web design com o exemplo de uma tabela. Encontra este template na pasta “views/templates”

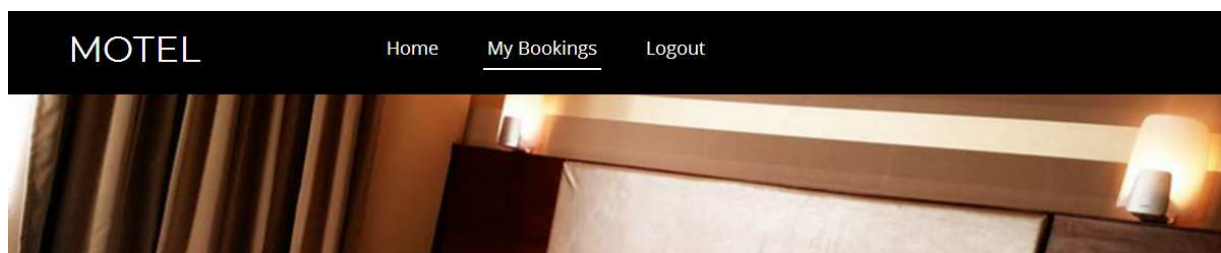
Construa o método mybookings() no controlador Motel.php e a query associada no controlador Motel_model.php

FUNCIONALIDADE “ON MOUSE OVER”

5. [2 valores]

Adicione ao template Smarty “mybookings.tpl” uma funcionalidade em Javascript de forma que se o “rato” passar em cima do tipo de quarto reservado “Bachelor Room, Family Room, etc) este texto é substituído pelas características do tipo de quarto descritas na tabela “room_categories”.

Se o “rato” sair de cima deste texto deve aparecer novamente o texto indicativo do tipo de quarto reservado.



My Bookings

#	Booking date	Check in	Check out	Nº of Nights	Room	Requests	Total Price
1	2018-11-23 11:52:27	2018-11-29	2018-12-01	2	Size: 40m ² Facilities: Closet with hangers, HD flat-screen TV, Telephone Adults: 4 Bed type: water bed	baby bed	600
2	2018-11-23 16:52:51	2018-11-20	2018-11-21	1	Presidential Room		300

NOTAS:

- Caso tenha trabalhado no seu portatil, **é obrigatório fazer o upload de todos os ficheiros*** para a pasta “RECURSO” no seu site web pessoal

```
/users/a12345/public_html/RECURSO
```

(onde 12345 é o seu número de aluno). Utilize scp (Linux) ou WinSCP (Windows) para fazer a cópia. As permissões dos ficheiros devem ser octal 640 (rw- r-- ---).

Verifique que o site fica operacional.

- Caso tenha problemas como seu código, lembre-se que pode consultar o log do servidor web com o comando

```
a12345@daw:~$ tail -f /var/log/apache2/error.log
```

* NÃO faça o upload de pastas!

ANEXO 1 Acesso à base de dados MySQL

- O acesso à base de dados MySQL pode ser feita utilizando um cliente gráfico à sua escolha (por exemplo <http://www.heidisql.com/>),

ou em linha de comando (substitua “12345” pelo seu número de aluno)

```
a12345@daw:~$mysql -u a12345 -p -h 10.10.23.183 db_a12345
```

ou ainda utilizando o software **phpMyAdmin** disponível no URL

- <http://all.deei.fct.ualg.pt/phpMyAdmin>
- <http://10.10.23.183/phpMyAdmin>

ANEXO 2 : estrutura da base de dados

```
--
-- Table structure for table `users`
--

CREATE TABLE IF NOT EXISTS `users` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(255) DEFAULT NULL,
  `email` varchar(255) DEFAULT NULL,
  `created_at` datetime NOT NULL,
  `updated_at` datetime NOT NULL,
  `password_digest` varchar(255) DEFAULT NULL,
  `remember_digest` varchar(255) DEFAULT NULL,
  `admin` tinyint(1) DEFAULT NULL,
  `activation_digest` varchar(255) DEFAULT NULL,
  `activated` tinyint(1) DEFAULT NULL,
  `activated_at` datetime DEFAULT NULL,
  `reset_digest` varchar(255) DEFAULT NULL,
  `reset_sent_at` datetime DEFAULT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `email` (`email`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Table structure for table `room_categories`
--

CREATE TABLE IF NOT EXISTS `room_categories` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `description` varchar(255) DEFAULT NULL,
  `adults` int(2) DEFAULT NULL,
  `bed_type` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Dumping data for table `room_categories`
--
INSERT INTO `room_categories` VALUES (1,'single',1,'single
bed'),(2,'double',2,'double bed'),(3,'suite',4,'water bed');

--
-- Table structure for table `rooms`
--

CREATE TABLE IF NOT EXISTS `rooms` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(255) DEFAULT NULL,
  `price` smallint(6) DEFAULT NULL,
  `category_id` int(11) NOT NULL,
  `facilities` varchar(255) DEFAULT NULL,
  `size` int(11) DEFAULT NULL,
  `image` varchar(255) DEFAULT NULL,
```

```

    PRIMARY KEY (`id`),
    KEY `category_id` (`category_id`),
    CONSTRAINT `rooms_ibfk_1` FOREIGN KEY (`category_id`) REFERENCES
`room_categories` (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Dumping data for table `rooms`
--
INSERT INTO `rooms` VALUES (1,'Bachelor Room',50,1,'Closet with hangers,
HD flat-screen TV, Telephone',10,'1.jpg'),(2,'Family Room',100,2,'Closet
with hangers, HD flat-screen TV, Telephone',20,'4.jpg'),(3,'Presidential
Room',300,3,'Closet with hangers, HD flat-screen TV,
Telephone',40,'2.jpg');

--
-- Table structure for table `bookings`
--

CREATE TABLE IF NOT EXISTS `bookings` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `title` varchar(255) NOT NULL DEFAULT 'RESERVED',
  `created_at` datetime DEFAULT NULL,
  `start` date NOT NULL,
  `end` date NOT NULL,
  `n_nights` int(11) DEFAULT NULL,
  `allDay` int(1) NOT NULL DEFAULT '1',
  `requests` varchar(255) DEFAULT NULL,
  `room_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `room_id` (`room_id`),
  KEY `user_id` (`user_id`),
  CONSTRAINT `bookings_ibfk_1` FOREIGN KEY (`room_id`) REFERENCES `rooms`
(`id`),
  CONSTRAINT `bookings_ibfk_2` FOREIGN KEY (`user_id`) REFERENCES `users`
(`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```