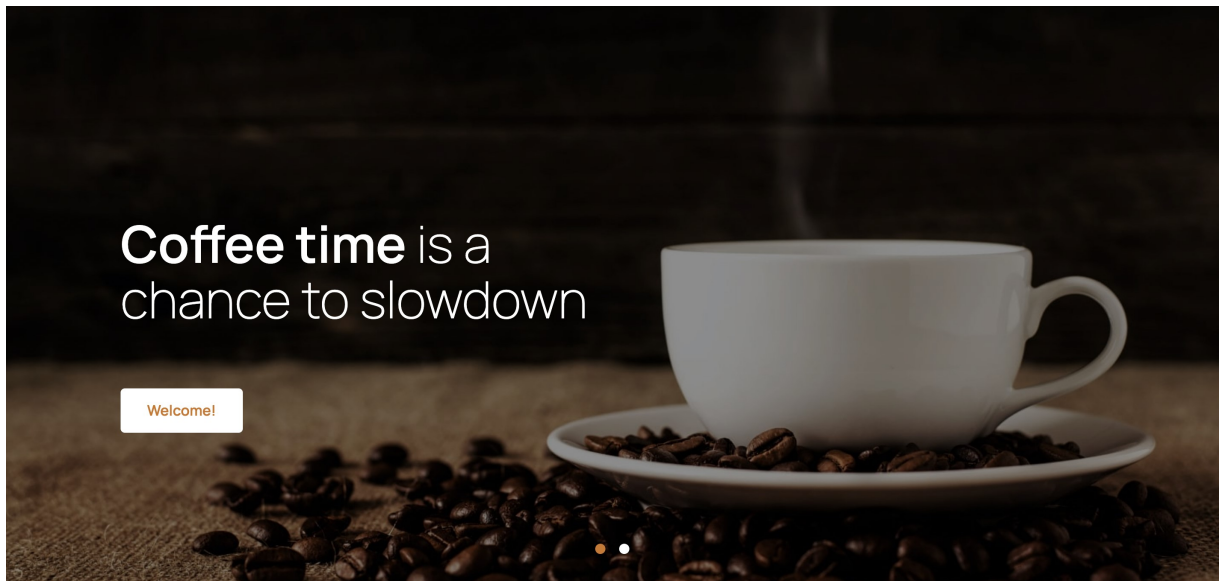


LER O ENUNCIADO ATÉ AO FIM ANTES DE COMEÇAR!

---

O objectivo do exame é construir 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 uma cafetaria que permite a utilizadores registados fazer reservas de mesas O site tem as seguintes páginas:

1. “**Home**” é a página de rosto do site;
2. “**Menu**” é a página principal do site;
3. “**Register**” para registo de utilizadores;
4. “**Login**” para o login de utilizadores;
5. “**Book**” permite ao utilizador registado reservar uma mesa;
6. “**MyBookings**” permite ao utilizador registado ver as suas reservas;
7. “**Message**” permite enviar mensagens;

É dado acesso a um conjunto de templates HTML construídos em *Bootstrap* fornecidos pela empresa de web design **w3layouts.com**.

Solicita-se ao aluno a realização do site web em *Symfony* e *Twig*, apenas das páginas acima descritas.

## PRELIMINARES

**A.** Faça login por ssh (com o PuTTY, por exemplo) no servidor com o IP 10.10.23.184<sup>1</sup>

```
a12345@daw2:~$
```

**B.** Actualize o ficheiro .env

Altere o ficheiro “.env” para utilizar as credenciais da sua base de dados.

```
a12345@daw2:~$ cd public_html/RECURSO2
```

```
a12345@daw2:~/public_html/RECURSO$ nano .env
```

```
DATABASE_URL=mysql://a12345:*****@10.10.23.184:3306/db_a12345?serverVersion=15.1
```

**C.** Crie a sua cópia da base de dados que é fornecida

```
a12345@daw2:~/public_html/RECURSO$
```

```
mysql -u a12345 --password=***** db_a12345 < database.SQL
```

Substitua “a12345” pelo seu login e “\*\*\*\*\*” pela password de acesso à sua base de dados

**D.** Teste a instalação do framework

A partir do seu browser preferido vá ao seguinte URL

```
http://daw.deei.fct.ualg.pt/~a12345/RECURSO/public/index.php/coffee
```

# Hello CoffeeController!

This friendly message is coming from:

- Your controller at [src/Controller/CoffeeController.php](#)
- Your template at [templates/coffee/index.html.twig](#)

Deverá receber uma página web de boas-vindas

---

<sup>1</sup> se não se encontra dentro da rede UALG terá que fazer primeiro login por ssh para ssh.deei.fct.ualg.pt

<sup>2</sup> a pasta RECURSO já existe na sua área pessoal

**E. OPCIONAL:** se prefere desenvolver o site no seu PC/portátil faça o download do repositório do exame com a aplicação git que tem instalada no seu PC/portátil:

```
user@meuPC:~$ git clone git://github.com/jmatbastos/RECURSO.git
user@meuPC:~/RECURSO$ composer install
```

Em alternativa, pode fazer o download do repositório num ficheiro "zip" apontando o browser no seu PC/portátil para o URL

<https://github.com/jmatbastos/RECURSO/archive/master.zip>

#### NOTAS:

- Se tiver dificuldade em criar a base de dados em linha de comando pode utilizar o acesso web *phpMyAdmin* e os comandos SQL descritos no APÊNDICE
- No caso de haver algum conflito com uma tabela já existente na sua base de dados, **mude o nome à tabela existente ou apague a tabela existente** (`DROP TABLE `nome_da_tabela``)
- A aplicação deve correr obrigatoriamente na área pessoal do aluno na pasta "RECURSO" no servidor web do departamento disponibilizado para o efeito: <http://daw.deei.fct.ualg.pt>
- O controlador principal tem que estar no ficheiro "CoffeeController.php"
- A classe com as funções de acesso à base de dados tem que estar no ficheiro "Coffee\_modelController.php"
- **Opcional!** Caso use o componente **Doctrine ORM** a restrição acima é levantada e pode utilizar um ficheiro para cada classe que representa uma tabela da base de dados
- **Opcional!** Pode utilizar **bin/console** para "scaffolding" das funcionalidades de registo e autenticação de utilizadores, bem como para a criação de formulários.
- **Opcional!** Pode utilizar **Doctrine SQL Query Builder** para o acesso à base de dados.

A página de entrada no site tem que ser

<http://daw.deei.fct.ualg.pt/~a12345/RECURSO/public/index.php/coffee>

(onde "a12345" é substituído pelo seu login)

Deve ser considerado uma **SUGESTÃO** o seguinte mapeamento entre URLs e controladores:

```
* @Route("/coffee", name="coffee")
* @Route("/menu", name="menu")
* @Route("/register", name="register")
* @Route("/login", name="login")
* @Route("/logout", name="logout")
* @Route("/book", name="book")
* @Route("/mybookings", name="myBookings")
* @Route("/message", name="message")
```

**Pode ver um demo do site no URL**

<http://daw.deei.fct.ualg.pt/~a12345/RECURSO/demo>

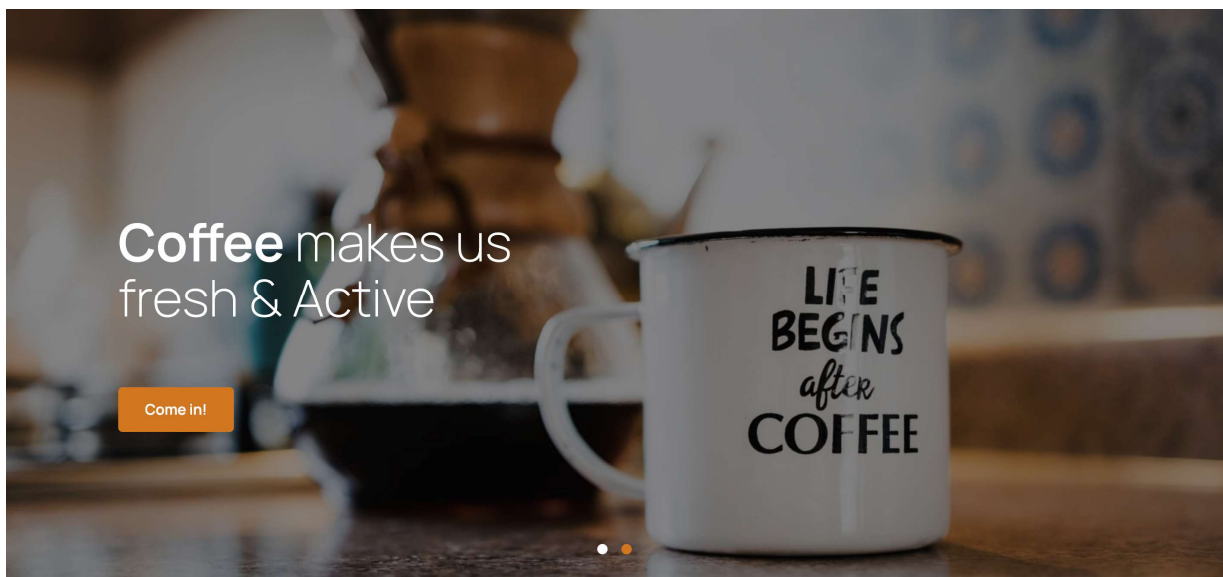
(Substitua "a12345" pelo seu login)

## 1. [2 valores] FUNCIONALIDADE “HOME”

A funcionalidade “Home” é página de rosto do site

Construa o template Twig para esta página.

SUGESTÃO: Adapte o template “index.html” fornecido pela empresa de web design. Encontra este template, bem como os componentes imagens, css, e javascript utilizados, na pasta “demo” (~ /public\_html/RECURSO/demo).



Construa o método no controlador `CoffeeController.php` responsável por gerar a página de rosto do site

“Welcome!/Come in!” é um hyperlink (“/menu”) que permite mostrar a página com o menu da cafetaria

### NOTA:

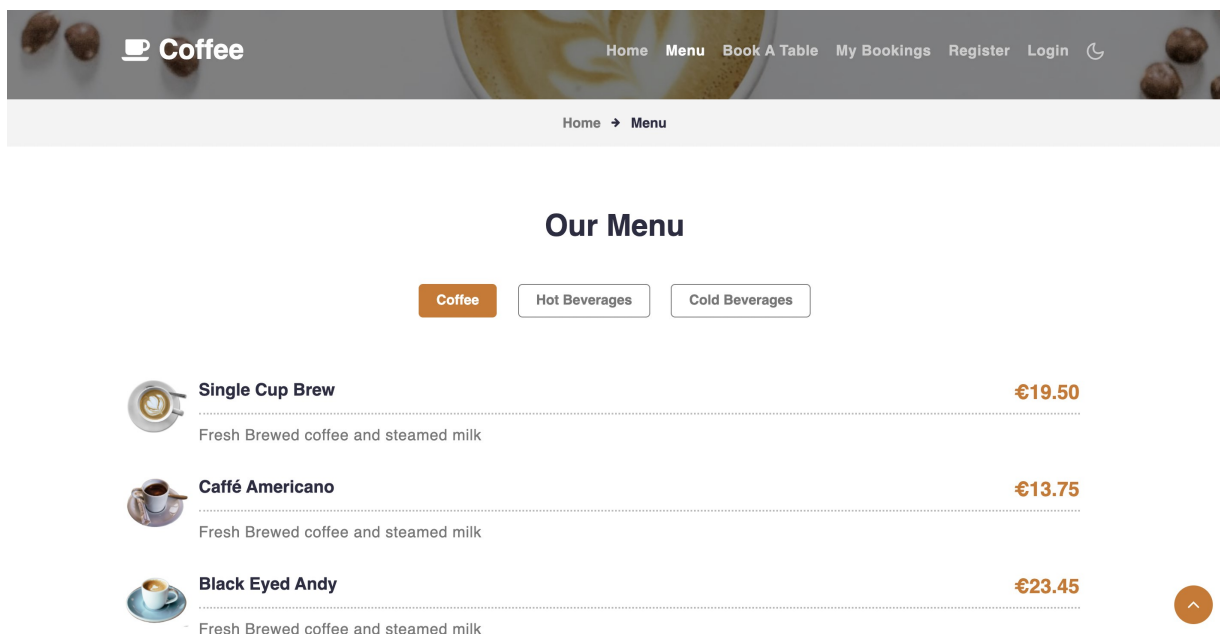
- É OBRIGATÓRIO o seu site ser portátil: os hyperlinks para recursos locais não podem ser absolutos! Recomenda-se que utilize as funções `path()` e `asset()` para gerar todos os hyperlinks locais.

## 2. [4 valores] FUNCIONALIDADE “MENU”

A funcionalidade “Menu” é página principal do site e contem um menu/lista dos cafés disponíveis nesta cafetaria.

Construa o template Twig para esta página.

SUGESTÃO: Adapte o template “menu.html” fornecido pela empresa de web design. Encontra este template na pasta “demo”



Construa o método no controlador `CoffeeController.php` e a query correspondente no controlador `CoffeeController_model.php` responsável por fazer uma lista dos cafés existentes na tabela “beverages” da base de dados

**Importante:** Note que o menu é controlado por três botões que permitem **mostrar à vez** cada uma das três categorias de bebidas existentes na tabela “beverages”. Estão claramente identificadas no template as três secções correspondentes às três categorias (`content1`, `content2`, `content3`). O código javascript responsável pela tabulação já se encontra implementado, apenas é necessário preencher o conteúdo das secções com as categorias de bebidas provenientes da base de dados.

- Pode considerar **estáticas** as três categorias de bebidas descritas nos botões.
- Para cada bebida mostre a imagem, o nome, a descrição, e o preço ("name", "image", "description", "price" existentes na tabela "beverages")
- "REGISTER", "LOGIN" são hyperlinks apenas visíveis caso o utilizador seja anónimo. Caso o utilizador tenha feito login, transformam-se em "LOGOUT", "BOOK" ("/logout", "/book"), fica visível um outro hyperlink "MY BOOKINGS" ("/mybookings"), bem como aparece o texto "Welcome user" onde *user* é o login do utilizador

## FUNCIONALIDADE “REGISTER”

A funcionalidade “Register” permite registar um utilizador.

### 3. [2 valores]

Construa o template Twig para esta página.

SUGESTÃO: Adapte o template “register.html” fornecido pela empresa de web design.

Encontra este template na pasta “demo”

**Name**

  
**Email Address**  
**Password**  
**Repeat Password**  

Construa os métodos responsáveis pelo formulário e pela acção deste formulário no controlador `CoffeeController.php` e a query correspondente no controlador `CoffeeController_model.php`

- Se o email ainda não existe na base de dados, regista o utilizador na tabela “users” utilizando a query correspondente no controlador `CoffeeController_model.php`. Re-direccione para a página “Login” (opcional: envia uma mensagem de sucesso ao utilizador)



A password é a *hash* utilizando o algoritmo MD5 da password:

```
substr(md5($password), 0, 32)
```

Pode utilizar a função MySQL `NOW()` ou a função PHP `date("Y-m-d H:i:s")` para actualizar os campos `created_at`, `updated_at` da tabela "users"

Se o registo falhar deve enviar uma mensagem de erro para o utilizador. Utilize uma variável de sessão para passar a mensagem entre controladores<sup>3</sup>

---

<sup>3</sup> a passagem de informação não é necessária se o método responsável pelo formulário e pela acção for o mesmo

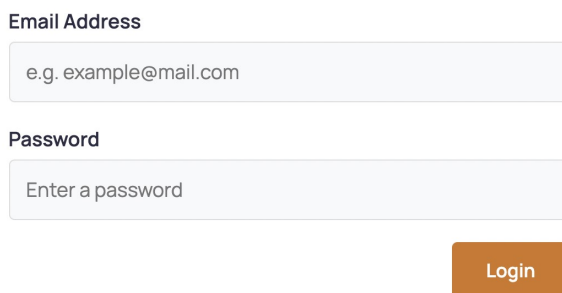
## FUNCIONALIDADE “LOGIN” & “LOGOUT”

A funcionalidade “Login” permite autenticar um utilizador.

### 4. [5 valores]

Construa o template Twig para esta página.

SUGESTÃO: Adapte o template “login.html” fornecido pela empresa de web design. Encontra este template na pasta “demo”



The image shows a login form with two input fields and a button. The first field is labeled 'Email Address' and contains the placeholder text 'e.g. example@mail.com'. The second field is labeled 'Password' and contains the placeholder text 'Enter a password'. Below the password field is an orange button labeled 'Login'.

Construa os métodos necessários no controlador `CoffeeController.php` e a query correspondente no controlador `CoffeeController_model.php`,

- em caso de sucesso no login:
  - regista no array da sessão os dados “id” e “name” do utilizador retirados da base de dados
  - re-direcciona para o portal principal do site “Menu”
  - No menu o hyperlink “Login” transforma-se no hyperlink “Logout” e
  - No menu o hyperlink “Register” desaparece
  - Aparece no menu o texto “Welcome user!” (onde “user” é o nome do utilizador registado)
  - No menu aparece um novo hyperlink “Book”
  - No menu aparece um novo hyperlink “My Bookings”
- em caso de insucesso no login
  - Envia uma mensagem de erro “Login failed”. Utilize uma variável de sessão para passar a mensagem entre controladores.

Construa o método “logout” que encerra a sessão no controlador `CoffeeController.php`

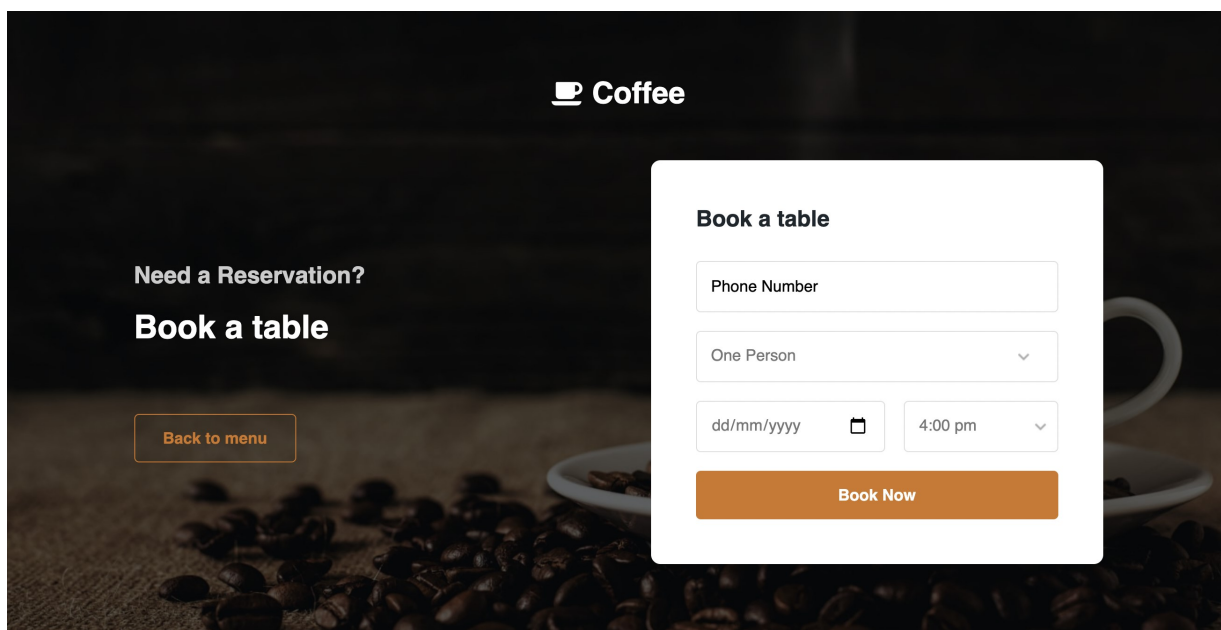
## 5. [4 valores] FUNCIONALIDADE “Book”

A funcionalidade “Book” permite ao utilizador registado fazer uma reserva de uma mesa na cafetaria. NOTA: Não é permitido ao utilizador fazer mais do que uma reserva por dia.

Construa o template Twig para esta página.

SUGESTÃO: Adapte o template “reservation.html” fornecido pela empresa de web design.

Encontra este template na pasta “demo”



Construa os métodos no controlador `CoffeeController.php` e a query correspondente no controlador `CoffeeController_model.php` responsável por fazer

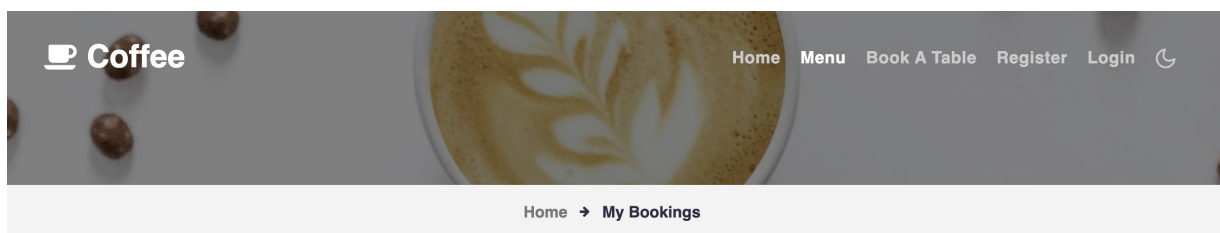
- a reserva da mesa na tabela “bookings”. Pode utilizar a função MySQL `NOW()` ou a função PHP `date("Y-m-d H:i:s")` para actualizar o campo `created_at`
- Envie a mensagem "Success, your booking is registered". Re-direccione para a página "My Bookings".

## 6. [3 valores] FUNCIONALIDADE “MY Bookings”

A funcionalidade “My Bookings” permite ao utilizador registado consultar todas as reservas de mesa que efectuou

Construa o template Twig para esta página.

SUGESTÃO: Adapte o template “mybookings.html” fornecido pela empresa de web design. Encontra este template na pasta “demo”.



### My Bookings

Table for one person

Date: 2022-01-01 Time: 4:00 pm

Reservation date: 2022-01-01

Table for 4 people

Date: 2022-01-01 Time: 4:00 pm

Reservation date: 2022-01-01

Construa o template Twig para esta página.

SUGESTÃO: Adapte o template “mybookings.html” fornecido pela empresa de web design. Encontra este template na pasta “demo”.

Construa o método no controlador `CoffeeController.php` e a query correspondente no controlador `Coffee_modelController.php` responsável por fazer uma lista das reservas existentes na tabela “bookings” da base de dados, cujo “id” da pessoa que fez a reserva corresponde à do utilizador que fez login.

# NOTAS:

- Caso tenha trabalhado no seu PC/portátil, **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) ou FileZilla (Windows e MAC) para fazer a cópia. As permissões dos ficheiros devem ser octal 640 (rw- r-- ---).

- Se precisar de copiar uma pasta faça-o no servidor com o comando

```
cp -R nome_da_pasta_original nome_da_pasta_destino
```

- 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
```

- Se receber uma mensagem de erro informando o acesso negado à pasta "**var/cache/dev**",

```
Warning: file_put_contents(/users/a12345/public_html/RECURSO/var/cache/dev/srcApp_KernelDevDebugContainerDeprecations.log): failed to open stream: Operation not permitted
```

terá que limpar o "cache" com o comando

```
a12345@daw2:~/public_html/RECURSO$ rm -rf var/cache/dev
```

OU (em alternativa) com o comando

```
a12345@daw2:~/public_html/RECURSO$ php bin/console cache:clear
```

## Verifique que o site fica operacional.

# ANEXO 1

Acesso à base de dados MySQL

- O acesso à base de dados MySQL pode ser feita em linha de comando (substitua "12345" pelo seu número de aluno)

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

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

- <http://daw.deei.fct.ualg.pt/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`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Table structure for table `categories`
--

CREATE TABLE `categories` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `description` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Dumping data for table `categories`
--

INSERT INTO `categories` VALUES (1,'Coffee'),(2,'Hot Beverages'),(3,'Cold Beverages');

--
-- Table structure for table `beverages`
--

CREATE TABLE `beverages` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(255) DEFAULT NULL,
  `price` smallint(6) DEFAULT NULL,
  `category_id` int(11) NOT NULL,
  `description` varchar(255) DEFAULT NULL,
  `image` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `category_id` (`category_id`),
```

```

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

--
-- Dumping data for table `beverages`
--

INSERT INTO `beverages` VALUES (1,'Single Cup Brew',20,1,'Fresh Brewed
coffee and steamed milk','middle.png'),(2,'Caffé Americano',14,1,'Fresh
Brewed coffee','middle1.png'),(3,'Caramel Macchiato',11,1,'Fresh
Caramel','middle2.png'),(4,'Standard black coffee',8,1,'Fresh black
','middle3.png'),(5,'Black Eyed Andy',23,1,'Fresh black
','middle2.png'),(6,'Coffee of the Day',22,1,'Fresh
coffee','middle3.png'),(7,'Cuban Shot',14,1,'Cream
cuban','middle.png'),(8,'Café Latte',11,1,'Fresh
latte','middle1.png'),(9,'Caramel Macchiato',20,2,'Fresh
macchiato','middle3.png'),(10,'Coffee of the Day',50,2,'Fresh
coffee','middle2.png'),(11,'Caffé Americano',23,2,'Fresh
americano','middle.png'),(12,'Filtered Coffee',50,2,'Fresh
filtered','middle1.png'),(13,'Cappuccino coffee',10,2,'Fresh
coffee','middle2.png'),(14,'Cafe latte',10,2,'Cream
latte','middle1.png'),(15,'Espresso coffee',15,2,'Cream
espresso','middle3.png'),(16,'Ice/Cold Coffee',18,2,'Super
iced','middle.png'),(17,'Single Cup Brew',20,3,'Single
cream','middle.png'),(18,'Caffé Americano',14,3,'Cream
coffee','http://daw.deei.fct.ualg.pt/~a666661/RECURSO/demo/assets/images/m
iddle1.png'),(19,'Caramel Macchiato',11,3,'Fresh
caramel','middle3.png'),(20,'Standard black coffee',8,3,'Standard
black','middle2.png'),(21,'Black Eyed Andy',23,3,'Super black
andy','middle1.png'),(22,'Coffee of the Day',22,3,'Cimbalino well
made!','middle3.png'),(23,'Cuban Shot',14,3,'Best bica in
town','middle2.png'),(24,'Café Latte',11,3,'Best \"meia de leite\" in
town!','middle.png');

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

CREATE TABLE `bookings` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `created_at` datetime NOT NULL,
  `date` date NOT NULL,
  `time` varchar(255) NOT NULL,
  `n_persons` varchar(255) NOT NULL,
  `user_id` int(11) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `user_id` (`user_id`),
  UNIQUE KEY `date` (`date`,`user_id`),
  CONSTRAINT `bookings_ibfk_2` FOREIGN KEY (`user_id`) REFERENCES `users`
(`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```