UNIVERSIDADE da MADEIRA

ESCOLA SUPERIOR DE TECNOLOGIAS E GESTÃO

Ano Letivo 2019/2020

Curso Técnico Superior Profissional em: Tecnologias e Programação de Sistemas de Informação

1 º Ano/2 º Semestre

Unidade Curricular: Desenvolvimento Web - Back-End **Data Entrega:** 08/06/2020

Docente: David Jardim **Época:** Normal

FREQUÊNCIA

Entrega

Entrega da resolução: através da plataforma moodle, na página da disciplina.

Data Limite de Entrega: 08 de junho de 2019, até às 19h30.

Nomenclatura da pasta com ficheiros: a pasta comprimida (zipada) deverá ter a seguinte estrutura quanto ao seu nome **FREQ_numero_aluno**

PARTE 1

1. Implementação de um servidor em Express.js respeitando a arquitetura REST e o padrão de desenho MVC utilizando o ficheiro de configuração **.env** (1 valor)
2. Utilize o ficheiro disponibilizado com o enunciado para importar o schema **"classicmodels"** para o seu servidor de MySQL. Efetue a ligação à base de dados utilizando o ORM *Sequelize* criando os modelos necessários (2 valores).
3. Tendo em conta a tabela 1, implemente os seguintes *endpoints* no servidor:
    a. Listar todos os clientes existentes na tabela ***customers*** e devolver a resposta no *body* (1 valor).
    b. Adicionar um novo cliente à tabela ***customers***. O ID do cliente adicionado deve ser devolvido na resposta (2 valores).
    c. Apagar um cliente da tabela ***customers*** pelo seu ID. O número de linhas afetadas deve ser devolvido na resposta. Caso o cliente a apagar não exista o erro deverá ser tratado de forma adequada (2 valores).
    d. Selecionar apenas um cliente pelo seu ID e devolver esse mesmo cliente na resposta. Caso o cliente a selecionar não exista, o erro deverá ser tratado de forma adequada (2 valores).
    e. Selecionar uma determinada encomenda referente a um determinado cliente. Caso não exista, o erro deverá ser tratado de forma adequada (2 valores).

Cofinanciado por:

MADEIRA 14-20 REGIÃO AUTÓNOMA DA MADEIRA PORTUGAL 2020 UNIÃO EUROPEIA Fundo Social Europeu

URI	Método HTTP	Body do POST	Resultado
/customers	GET	empty	Show list of all the customers.
/customers	POST	JSON String	Add details of new customer.
/customers/:id	DELETE	JSON String	Delete an existing customer.
/customers/:id	GET	empty	Show details of a customer.
/customers/:cid/orders/:oid	GET	empty	Show details of a single order from a single customer.
/log	GET	empty	Download log file

Tabela 1 - Lista de endpoints

4. Implemente uma função de *logging* como *middleware* para registar o histórico de todos os pedidos HTTP que forem efetuados ao servidor num ficheiro **log.txt**. Formato: **URI, Método HTTP, DIA:MÊS:ANO HORA:MINUTO** (3 valores).
5. Implemente um endpoint que permita efetuar download do ficheiro de log (2 valores).

PARTE 2

1. Crie um ficheiro denominado por algorithm.js, nesse ficheiro implemente uma função que dado o seguinte array:

```
[[9, 14], [1, 3], [5, 7], [6, 8]];
```

Devolve outro array transformado da seguinte forma (3 valores):

```
[[1, 3], [5, 8], [9, 14]];
```

Column	Type	Default Value	Nulla...
comments	text		YES
customerNumber	int		NO
orderDate	date		NO
orderNumber	int		NO
requiredDate	date		NO
shippedDate	date		YES
status	varchar(15)		NO

*Figura 1 - Tabela **orders***

Column	Type	Default Value	Nullable
customerNumber	int(11)		NO
customerName	varchar(50)		NO
contactLastName	varchar(50)		NO
contactFirstName	varchar(50)		NO
phone	varchar(50)		NO
addressLine1	varchar(50)		NO
addressLine2	varchar(50)		YES
city	varchar(50)		NO
state	varchar(50)		YES
postalCode	varchar(15)		YES
country	varchar(50)		NO
salesRepEmployeeNumber	int(11)		YES
creditLimit	decimal(10,2)		YES

*Figura 2 - Tabela **costumers***

Cofinanciado por:

MADEIRA 14-20 · REGIÃO AUTÓNOMA DA MADEIRA · PORTUGAL 2020 · UNIÃO EUROPEIA Fundo Social Europeu

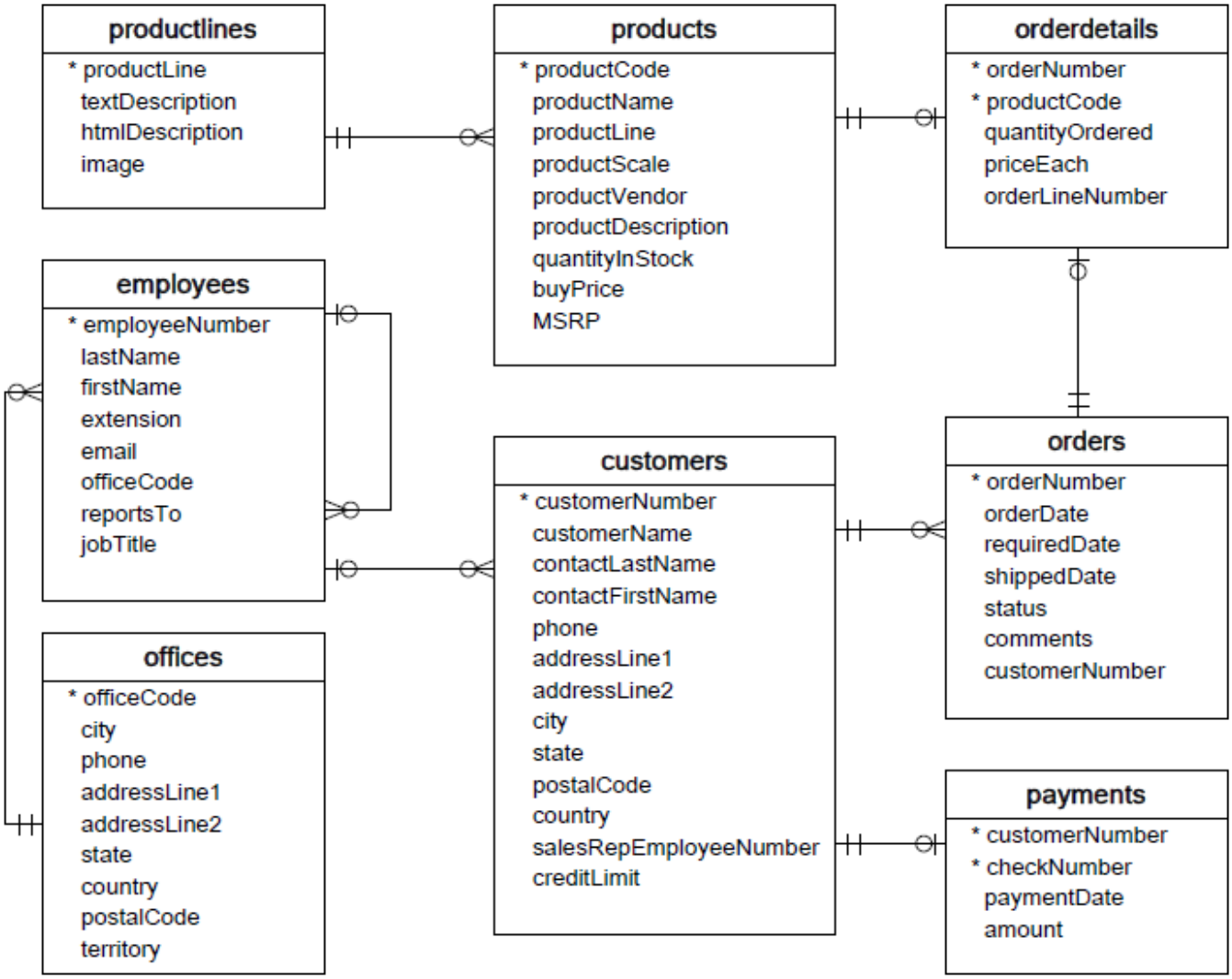

Figura 3 - Schema da Base de Dados

Cofinanciado por:

MADEIRA 14-20 REGIÃO AUTÓNOMA DA MADEIRA PORTUGAL 2020 UNIÃO EUROPEIA Fundo Social Europeu