

**Prova de Revisão****Tema:** Escolas, Docentes e Cursos

---

**1) Arquitetura base – Laravel + Auth + Bootstrap [2 valores]**Crie uma aplicação **Laravel 7** com autenticação de utilizadores e integração de **Bootstrap**.A navegação e acesso às páginas devem estar protegidos por login.

---

**2) Master Page [2 valores]**Crie uma **Master Page** que sirva como base para todas as páginas da aplicação.

A master page deve incluir:

- Navbar com links para **/schools**, **/teachers** e **/courses**
- 

**3) Migrações, modelos e relações [3 valores]**Crie todas as **migrações** e **models** necessários, com as relações Eloquent corretas.**Tabela: schools**

Campos mínimos:

- id (PK)
- name (único)
- city
- created\_at, updated\_at, deleted\_at

**Tabela: teachers**

Campos mínimos:

- id (PK)
- school\_id (FK → schools.id, *on delete cascade*)
- first\_name, last\_name
- hire\_date
- created\_at, updated\_at, deleted\_at

**Tabela: courses**

Campos mínimos:

- id (PK)
- teacher\_id (FK → teachers.id, *on delete cascade*)
- title
- category
- price (decimal)
- created\_at, updated\_at, deleted\_at

Relações esperadas:

- **School 1-N Teacher**
  - **Teacher 1-N Course**
-

**4) Seeder de Schools [2 valores]**

Crie uma *seed* para preencher a tabela *schools* com os seguintes registos:

- Escola de Lisboa
- Escola do Porto
- Escola de Coimbra
- Escola de Braga

**5) Factory de Teachers [2 valores]**

Crie uma *factory* para gerar **100 teachers**, garantindo que **cada teacher** pertence a **uma school existente**.

**6) Factory de Courses [2 valores]**

Crie uma *factory* para gerar **200 courses**, garantindo que **cada teacher** tem **pelo menos 2 courses** associados.

**7) CRUD de Schools [2 valores]**

Implemente as páginas CRUD acessíveis em **/schools**:

- Listagem com pesquisa por name e city
- Criar, editar, apagar (soft delete) e restaurar
- Validação (ex.: name obrigatório e único)

**8) CRUD de Teachers [2 valores]**

Implemente as páginas CRUD acessíveis em **/teachers**:

- Listagem com coluna da *school* e filtro por *school*
- Formulário com *dropdown* de *school*
- Validação (ex.: *first\_name*, *last\_name* obrigatórios; *hire\_date* data válida)

**9) CRUD de Courses [3 valores]**

Implemente as páginas CRUD acessíveis em **/courses**:

- Listagem com *teacher* associado e filtros por *category* e *teacher*
- Formulário (create/edit) com seleção do *teacher*
- Validação (ex.: *title* obrigatório; *price*  $\geq 0$ )

**Requisitos Transversais / Critérios de Avaliação**

- Uso consistente da Master Page e Bootstrap
- Rotas *resource* e *route model binding*
- Mensagens de validação em português
- Proteção por *Auth* nas rotas CRUD
- Uso de **Soft Deletes** em todas as tabelas
- Factories e Seeders integrados via DatabaseSeeder
- Clareza, organização e legibilidade do código