

Instituto Superior de Engenharia de Lisboa  
Licenciatura em Engenharia Informática e de Computadores  
**Desenvolvimento de Aplicações Web**  
Teste Final, Época Especial, Semestre de Verão, 20/21  
**Duração: 2 horas**

---

1. (6) Para cada uma das seguintes questões, seleccione a opção de resposta correcta. A escolha de uma opção errada contribui negativamente para o resultado final com um terço da cotação da questão.
  - 1.1. No protocolo HTTP, o *header Authorization* tem semântica definida:
    - i. Apenas nas mensagens de pedido.
    - ii. Apenas nas mensagens de resposta.
    - iii. Nas mensagens de pedido e de resposta.
    - iv. Nenhuma das anteriores.
  - 1.2. A realização de um pedido de método GET para `https://example.com/projects/123/delete`, deve ser interpretado por um intermediário como sendo:
    - i. Equivalente ao pedido de método DELETE para `https://example.com/projects/123`.
    - ii. Um pedido idempotente mas não *safe*.
    - iii. Um pedido idempotente e *safe*.
    - iv. Nenhuma das anteriores.
  - 1.3. Segundo o RFC 8288 (*Web Linking*), um *link relation type* é
    - i. O identificador do recurso alvo do *link*.
    - ii. O identificador do recurso que define o contexto do *link*.
    - iii. O identificador que expressa a semântica da ligação entre dois recursos.
    - iv. Nenhuma das anteriores.
  - 1.4. No contexto da *framework* React, a utilização de `undefined` como segundo argumento da função `useEffect`, significa que:
    - i. O efeito vai ser chamado uma vez durante o tempo de vida da instância do componente.
    - ii. O efeito vai ser chamado uma vez durante o tempo de vida da aplicação.
    - iii. O efeito vai ser chamado sempre que a função que define o componente for chamada.
    - iv. Nenhuma das anteriores.
  - 1.5. No contexto de um *browser*, a avaliação da seguinte expressão resulta:

```
fetch('https://httpbin.org/status/404').status
```

    - i. No número 404.
    - ii. No número 200.
    - iii. No valor `null`.
    - iv. Nenhuma das anteriores.
  - 1.6. O resultado da avaliação da expressão JSX `<Div><div /></Div>` é:
    - i. Um elemento HTML, do mesmo tipo do obtido na avaliação da expressão `document.createElement('Div')`.
    - ii. O resultado da avaliação da expressão `React.createElement(Div, null, document.createElement('div'))`.
    - iii. O resultado da avaliação da expressão `React.createElement(Div, null, React.createElement('div', null))`.
    - iv. O resultado da avaliação da expressão `React.createElement(Div, null, React.createElement(div, null))`.

2. (2) Indique dois motivos para a utilização de *hypermedia* nas representações dos recursos de uma API HTTP.
3. (2) No contexto da plataforma Spring MVC, quais os critérios que se deve ter em conta para escolher um *servlet filter* ou um *handler interceptor* como ponto de extensibilidade a usar para criar um intermediário no processamento de pedidos.
4. (2) Tendo em consideração o modelo de construção de aplicações para a plataforma *browser* usado na unidade curricular, indique qual o propósito e forma de utilização da ferramenta *webpack*.
5. (4) Realize um ou mais componentes para a plataforma Spring MVC de forma a que todas as respostas produzidas tenham o *header* HTTP **Debug** contendo o tempo em milisegundos que a resposta demorou a ser processada, bem como o eventual *handler* envolvido nesse processamento. Valorizam-se soluções em que o cálculo do tempo de processamento inclua não só o tempo de execução do eventual *handler* mas também o da maioria dos intermediários envolvidos (e.g. filtros e interceptores).
6. (4) Realize um componente para a *framework* React que:
  - Apresenta o valor de um contador, iniciado com zero.
  - Apresenta uma caixa de texto e um botão. Sempre que o botão for premido, o valor presente na caixa de texto deve ser usado como o período, em milisegundos, com que o contador é incrementado. Inicialmente, e sempre que o botão seja premido com a caixa de texto vazia, o período de contagem deve ser infinito.

Tenha em consideração que a função de actualização de estado, retornada pelo *hook* `useState`, também pode receber uma função que recebe o valor de estado anterior e retorna o novo valor pretendido para o estado.