



# Instituto Superior de Engenharia de Lisboa

## Engenharia Informática e de Computadores

### Tecnologias da Internet

Semestre de Inverno 2005/2006 - 1ª Época

11-01-2006

**Duração :** 2 h 30 min

**Os grupos 1 e 2 devem ser respondidos em folhas separadas e a folha com o grupo 3 do enunciado deve ser entregue com a restante resolução.**

### Grupo 1

1. [3] Responda, num máximo de 5 linhas, às seguintes questões:

- [1] Porque razão os controlos ASP.NET que recebem o seu estado através dos dados enviados na submissão do formulário, mantêm também esse estado em *View State*?
- [1] No Trace de um Web Form ASP.NET, verifica-se que o primeiro acesso é realizado através do comando HTTP GET e os subsequentes por POST. Justifique esta diferença.
- [1] Qual a forma recomendada para transferir informação no pipeline HTTP das ASP.NET?

2. [2] Considere o excerto de código de uma classe de Code Behind numa ASPX.

```
public class SomePage : System.Web.UI.Page {  
    protected override void OnLoad(EventArgs e) {  
        Response.Write("SLB");  
    }  
}
```

a) [1] Explique por que motivo o `Response.Write()` não deve ser usado aqui?

NOTA: Para ajudar na explicação, pode ser mostrado um excerto da página recebida pelo cliente.

b) [1] No contexto do processamento de um pedido qual o local indicado para chamar este método?

3. [2] Considere o seguinte excerto de uma página ASPX.

```
<script runat=server>  
protected void Page_Load(object src, EventArgs e) {  
    ...  
}  
</script>
```

a) [1] O que é necessário para que o código deste método seja executado.

b) [1] Descreva em pseudo-código o código gerado pela infra-estrutura ASP.NET, para que o código seja executado.

### Grupo 2

1. [10] Pretende-se implementar uma infra-estrutura Web para a publicação e visualização de notícias. Uma notícia tem título, data e o conteúdo.

a) [1] Defina o modelo de dados em UML para suportar esta infra-estrutura de notícias. Na descrição do modelo de dados defina e descreva a semântica (lógica) dos tipos que o constituem bem como da sua interface pública. Este modelo de dados deve ser pensado exclusivamente em *Object Oriented*. Os detalhes de onde os dados realmente residem serão da responsabilidade desta camada cuja implementação não é objectivo deste exercício.

b) [1] Implemente uma página ASP.NET com um formulário Web para inserção de uma notícia.

NOTA: Para implementação da lógica de inserção da notícia utilizar única e exclusivamente tipos e métodos descritos na alínea anterior.

c) [7] Implementar o CustomControl ASP.NET para visualização de notícias (NewsViewer), com um aspecto semelhante ao da figura seguinte.

Notícias	
Exame de TI - Salas 09/01/2005	O exame de TI vai ser realizado nas salas G0.18 e G0.19.
Exame de TI - Consulta 10/01/2005	
N 3 01/01/2005	
N 4 25/12/2005	
N 5 12/12/2005	

Este controlo tem as seguintes funcionalidades:

- apresenta o número de notícias, definido na sua propriedade NumNewsVisible (por omissão 5), de um máximo definida na propriedade MaxNews (por omissão 10);
- contém *script* de cliente de modo a que as notícias visíveis representem uma janela deslizante (de dimensão NumNewsVisible) do máximo definido (MaxNews); o ritmo a que as notícias vão rodando (em segundos) é definido pela propriedade NewsRotationPeriod (por omissão 1s); a rotação das notícias visíveis não deve provocar qualquer submissão da página onde o controlo se encontra inserido;
- o conteúdo da notícia seleccionada (através de um clique) é mostrado na zona direita do controlo (ver figura); se nenhuma notícia for seleccionada esta zona não mostra qualquer conteúdo;
- suporta *Data Binding* para fontes de dados do tipo IEnumerable de objectos que representam uma noticia (definidos na alínea a.) - assume-se que a fonte de dados associada já está ordenada pelo critério que se pretende escalar as notícias e que desta apenas são consideradas as primeiras MaxNews notícias, caso existam em número superior;
- mantém estado de visualização.

NOTAS sobre a implementação do controlo:

- Toda a lógica de cliente deve ser obtida através de JavaScript manipulando a API DOM do documento, utilizando CSS para criar os aspectos visuais que tornar visíveis e invisíveis determinados conteúdos do controlo.
- Assuma que só pode existir um controlo NewsViewer por página.

O desenvolvimento deste controlo está dividido em várias fases:

- [1] Apresentar o *Box Model*, indicando que informação fica em cada bloco e os identificadores dos elementos que necessitam de ser obtidos através da função getElementById(), da interface HTML gerada pelo controlo.
- [2] Implementar o ficheiro RotationNews.js que inclui toda a lógica de *script* de cliente para implementar a janela deslizante sobre as notícias apresentadas através do *Box Model* definido em i.
- [1,5] Implementar o render do CustomControl ASP.NET de modo a gerar o *Box Model* definido em i) e a incluir o *script* definido em ii).

NOTA: Para realizar esta alínea assumo que a fonte de dados é um IEnumerable e é fornecida ao controlo através de uma propriedade com o nome DataSource.

iv. [1] Acrescente o necessário para que o controlo suporte *Data Binding*.

v. [1,5] Acrescente o necessário para que o controlo mantenha estado de visualização.

d) [1] Implemente uma formulário Web (ShowNews.aspx) que inclui uma instância do controlo criado na alínea anterior e, utilizando o modelo de dados definido em a) contém a lista de notícias, e apresenta 20 notícias numa janela de 5.

## Grupo 3

1. [3] Para cada uma das questões seguintes existem 3 afirmações, das quais apenas uma é correcta. Indique-a.

ATENÇÃO: A escolha da opção correcta adiciona ao total o valor da cotação da pergunta. A ausência de resposta contabiliza zero valores. Uma resposta errada desconta metade da cotação da pergunta.

**a) [1]** Uma aplicação CGI

- i.** recebe sempre os dados do cliente através de variáveis de ambiente.
- ii.** tem que ser implementada numa linguagem que compilada.
- iii.** implica a criação de um processo para cada pedido.

**b) [1]** O acesso ao estado de aplicação (instância de `HttpApplicationState`)

- i.** deve ser sempre precedido da aquisição explícita, por parte do programador, da zona de exclusão através da chamada ao método `Lock()`.
- ii.** não necessita de aquisição de *lock* para leitura uma vez que tem semântica de múltiplos leitores.
- iii.** necessita sempre de obter o *lock*, para leitura e para escrita.

**c) [1]** Os controlos ASP.NET têm 3 propriedades que indicam o seu ID: `ID`, `ClientID` e `UniqueID`. Estas propriedades

- i.** têm sempre o mesmo valor.
- ii.** podem ter valores diferentes quando o controlo está contido noutra.
- iii.** têm valores diferentes quando o controlo implementa `INamingContainer`.

Os Docentes

(Luis Falcão e Carlos Guedes)