

1)

Na camada de regras de negócio. Em 1º lugar porque as classes `Utilizador` e `Produto` tem atributos para representar cada instância de um objecto de negócio. Em segundo lugar porque possuem métodos que permitem efectuar operações de negócio (ex., `GetUserByLoginAndPassword`) e não apenas encapsular os métodos de acesso a dados (ex., `Insert`, `Delete`, `Update`).

**OBSERVAÇÃO:** Estas duas classes pressupõem a existência de uma camada de mapeamento entre os `DataSet` usados na camada de acesso a dados (se a tecnologia de desenvolvimento fosse `.net`) e estes objectos de negócio à semelhança do que foi apresentado nas aulas para o projecto “Tarefas”.

2)

**NOTA PRÉVIA:** o âmbito da disciplina centra-se no modelo por camadas, tendo sido apresentado ao longo do semestre trabalhos de referência sobre como dividir a aplicação. No exemplo apresentado existem classes duas instâncias de classes com o mesmo nome (`Categoria` e `Produto`), que não são no entanto a mesma classe. Se recordarem o mecanismo de funcionamento do modelo 3 camadas concluíam que a 1ª instancia que aparece é da camada de regras de negócio e a 2ª é da camada de acesso a dados. Para as distinguir em termos de código teriam que as colocar em *packages* diferentes e em seguida definir cada *package* como sendo um *namespace* C#. Nada era dito sobre os parâmetros ou tipo de retorno dos métodos pelo que deviam assumir um tipo razoável, embora esse não fosse factor de classificação neste exercício, onde se pretendia que mostrassem claramente que perceberam a ligação entre modelo e código gerado.

```
namespace UI
{
    ...
    public class ListByCatForm
    {
        ...
        private void ListByCatForm_Load() // mensagem 1
        {
            ...
            // mensagem 2
            LoadCategorias();
            ...
        }
        private void LoadCategorias()
        {
```

```

    ...
    // mensagem 3
    BusRules.Categorias cat = new BusRules.Categorias();
    // mensagem 4
    cat.Load();
    ...
    // mensagem 8
    LoadProdutosByCategoria();
    ...
    // mensagem 16
    RefreshCategorias();
    ...
}
private void LoadProdutosByCategoria()
{
    ...
    // mensagem 9
    BusRules.Produto prod = new BusRules.Produto();
    // mensagem 10
    prod.Load();
    ...
    // mensagem 14
    prod = null;
    ...
    // mensagem 15
    RefreshProdutos();
    ...
}
}
...
}

namespace BusRules
{
    ...
    public class Categoria
    {
        ...
        public IList Load()
        {
            ...
            // mensagem 5
            DAL.Categoria cat = new DAL.Categoria();
            // mensagem 6
            cat.GetAll();
            ...
            // mensagem 7
            cat = null;
            ...
        }
        ...
    }

    public class Produto
    {
        ...

```

```

    public IList Load()
    {
        ...
        // mensagem 11
        DAL.Produto prod = new DAL.Produto();
        // mensagem 12
        prod.GetAll();
        ...
        // mensagem 13
        prod = null;
        ...
    }
    ...
}
...
}

namespace DAL
{
    ...
    public class Categoria
    {
        ...
        public IList GetAll()
        {
            ...
        }
        ...
    }

    public class Produto
    {
        ...
        public IList GetAll()
        {
            ...
        }
        ...
    }
    ...
}

```



