



Software de Comunicaciones  
I.T.T. Especialidad Telemática  
Escuela Politécnica Superior  
Universidad Carlos III de Madrid

**Duración del test: 1 hora.**

Duración total del examen: 2 horas 30 minutos.

**NORMAS:**

No se permiten ni libros ni apuntes durante el examen.

**Nombre:** .....

**Apellidos:** .....

**Test (5 puntos)**

Cada respuesta **correcta** puntúa **0,25**, mientras que una incorrecta de las de múltiple opción **resta 0,083** (1/3 de una correcta). Las preguntas de rellenar (5-8, 16-20) **no** tendrán ningún tipo de penalización.

1. Señale cuál de las siguientes afirmaciones es **correcta** en referencia a las aplicaciones empresariales.
  - a. J2EE no es una tecnología basada en componentes
  - b. Una arquitectura multicapa (*multi-tier*) es menos escalable que una cliente-servidor
  - c. J2EE no soporta un servicio de persistencia
  - d. El servidor de aplicaciones hospeda aplicaciones y les ofrece ciertos servicios, permitiendo y facilitando su gestión
2. ¿Cuál de los siguientes servicios **no** es ofrecido por un contenedor, a un *Enterprise Java Bean*?
  - a. Servicio de nombres y directorio
  - b. Transacciones
  - c. Administración de instancias
  - d. Copia de seguridad
3. Señale la afirmación **correcta** con respecto a un **bean de sesión**
  - a. Los sin estado permiten trabajar con transacciones
  - b. Los con estado permiten compartir estado entre varios clientes
  - c. Los sin estado pueden almacenar estado de un único cliente en el servidor
  - d. Los con estado están respaldados por una base de datos

4. Señale la afirmación **falsa** con respecto a un **bean de entidad**
  - a. Los hay con persistencia gestionada por bean (BMP) y por contenedor (CMP)
  - b. Su persistencia se puede apoyar en bases de datos
  - c. Carecen de modelo de seguridad, pudiendo ser compartidos por múltiples clientes
  - d. Por lo general tienen soporte transaccional

**Complete** las siguientes afirmaciones (1 punto):

5. Un bean de \_\_\_\_\_ en la arquitectura J2EE ofrece asincronismo.
6. La \_\_\_\_\_ de un EJB permite acceder a la lógica de negocio de un EJB, independientemente de su localización (local o remota).
7. El atributo transaccional \_\_\_\_\_, aplicado sobre un método remoto obliga a que dicho método sea llamado desde un *contexto transaccional no nulo*, lanzando una excepción en caso de que no se realice desde este tipo de contexto.
8. Al utilizar el asistente proporcionado por el `deploytool`, para crear el descriptor de despliegue y empaquetar un EJB en un componente JAR hay que seleccionar un nuevo objeto de tipo \_\_\_\_\_.
9. ¿Cuál de los siguientes retos está asociado a la programación para dispositivos limitados?
  - a. interfaz de usuario mono-modal y con programación limitada
  - b. capacidad de proceso y de memoria limitada
  - c. conectividad intermitente con una única interfaz de red inalámbrica
  - d. memoria y almacenamiento únicamente disponible para aplicaciones nativas
10. ¿Cuál de las siguientes afirmaciones respecto a los sistemas operativos para dispositivos móviles es **falsa**?
  - a. En Android, las aplicaciones pueden acceder a librerías en código nativo utilizando el NDK
  - b. En RIM las aplicaciones se ejecutan en una máquina virtual denominada Dalvik
  - c. Symbian soporta desarrollo de aplicaciones en código nativo C/C++, y en otros lenguajes como JME, Python, Visual Basic, Perl y Simkin
  - d. Windows Phone 7, también conocido como WM7, es la última versión comercialmente disponible del SO basado en Windows CE
11. En el nivel **KVM/CLDC** se introduce el *Generic Connection Framework* (GCF) principalmente para:
  - a. complementar el soporte de Java SE, añadiendo nuevos protocolos a la API
  - b. implementar las diferentes formas de comunicación posibles que pueden ser extendidas en el perfil correspondiente
  - c. soportar diferentes tipos de protocolos de red a través de un mecanismo general de conexión
  - d. implementar mecanismos de E/S de información persistente independientemente de la localización

12. En el perfil MIDP (*Mobile Information Device Profile*),
- un MIDlet Suite es la unidad básica de ejecución con un ciclo de vida bien definido
  - un MIDlet es la unidad básica de ejecución que tiene un ciclo de vida bien definido
  - un MIDlet es un conjunto de aplicaciones que comparten recursos en el contexto de una única máquina virtual
  - un Servlet Suite es un conjunto de aplicaciones que comparten recursos en el contexto de una única máquina virtual
13. Diga cuál de las siguientes afirmaciones acerca del sistema de almacenamiento persistente en MIDP 2.0 **no** es correcta:
- La API es independiente del dispositivo y se define en el paquete `javax.microedition.rms`
  - Está orientado a registros que se guardan en almacenes de registro (`RecordStore`)
  - Para recorrerlos se proporciona la clase `RecordEnumeration`, que es una lista doblemente enlazada
  - Proporciona interfaces para agrupar y combinar registros de uno o varios almacenes de registro
14. En el modelo de seguridad **a nivel de aplicación** de MIDP 2.0:
- los atributos `MIDlet-Permissions` y `MIDlet-Permission-opt` permiten al programador definir el conjunto de permisos indispensables u opcionales requeridos por el MIDlet suite.
  - no se soporta JNI (*Java Native Interface*), reflexión, finalización de instancias de clases, ni la definición de cargadores de clase para evitar posibles ataques
  - la arquitectura se basa en un modelo “sandbox” de modo que no se pueden sobrescribir clases del sistema ni acceder a las clases nativas, sólo se tiene acceso a un conjunto limitado de APIs.
  - la gestión de los dominios de seguridad y de las políticas para los permisos asociados a los mismos se realiza a nivel del programador en cada MIDlet.
15. Diga cuál de las siguientes afirmaciones es **falsa** respecto a Android:
- Utiliza Linux para los drivers de su dispositivo, gestión de memoria y procesos, y red
  - Es un sistema multi-proceso, en el cual cada aplicación se ejecuta en su propio proceso
  - Dalvik VM ejecuta archivos ejecutables `.dex` y `.jar` ya que ambos formatos están optimizados
  - La API del *Activity Manager* permite gestionar el ciclo de vida de las aplicaciones

**Complete** las siguientes afirmaciones (1 punto):

16. En la arquitectura de Java ME, \_\_\_\_\_ define el conjunto mínimo de clases disponibles para un grupo de dispositivos. Los grupos se establecen según requisitos similares de memoria y procesamiento.
17. En MIDP 1.0 la API para juegos, la clase `Canvas`, utilizaba tres hilos distintos para: \_\_\_\_\_, \_\_\_\_\_, y \_\_\_\_\_.

18. En la fase de \_\_\_\_\_ se comprueba que el MIDlet no vulnera las políticas de seguridad del móvil.
19. La arquitectura del sistema operativo iOS incluye cuatro capas: núcleo del sistema operativo, servicios del núcleo, media y \_\_\_\_\_.
20. Una aplicación en Android puede disponer de almacenamiento persistente a través del componente \_\_\_\_\_.