

1) Una nueva iteración del análisis de requisitos es necesario dependiendo de:

- A) Feedback de los developers.
- ☒ B) Feedback de los usuarios que hayan testeado el primer prototipo.
- C) Feedback del diseño del Sistema.
- D) La calidad de los Mock-ups.

2) Los casos de uso son modelos gráficos que se usan para representar:

- A) Requisitos detallados de la aplicación.
- ☒ B) Vistas de alto nivel de las aplicaciones de los usuarios, de los componentes y las funcionalidades.
- C) Un resumen de las interfaces de usuario.
- D) Un resumen de las interacciones entre usuarios.

3) ¿Cuál de estos ciclos de vida en software development es incorrecto?

- ☒ A) Casos de uso, documentación de requisitos, mock-up, diseño, programación, testing.
- B) Análisis de producto, requisitos, diseño, programación, testing.
- C) Documentación de requisitos, diseño, programación, test funcional, test de usabilidad.
- D) Identificación de usuario, análisis de requisitos, diseño, programación, testing.

4) ¿Que lenguaje se debería usar para la documentación de requisitos?

- A) Inglés para garantizar que todos los clientes/usuarios lo pueden entender.
- B) Lenguaje simple que todos puedan entender.
- ☒ C) Detallado y específico, sin ambigüedades.
- D) Lenguaje técnico del ámbito de la aplicación.

5) El proceso OnRestart() se inicia después de:

- ☒ A) OnStop()
- B) OnDestroy()
- ☒ C) OnPause() y OnStop()
- D) OnPause()

6) Una buena implementación del ciclo de vida puede ayudar a garantizar que tu aplicación:

- A) No falle si el usuario recibe una llamada telefónica o cambia a otra app mientras usa la tuya.
- B) No consuma recursos valiosos del sistema cuando el usuario no la use de forma activa.
- C) No pierda el progreso del usuario si este abandona tu app y regresa a ella posteriormente.
- ☒ D) Todas las anteriores

7) ¿Qué elemento se debe declarar solo en un intent explícito, a diferencia de un intent implícito?

- ☒ A) Component Name
- B) Data
- C) Action
- D) Filter ID

8) ¿Dónde se declaran las características necesarias para interpretar un inten?

- A) Filtro de intents en build.gradle de app
- B) Filtro de intents en Activity principal
- ☒ C) Filtro de intents en el Manifest.xml
- D) En ninguno de los anteriores

9) ¿Qué parámetro XML sirve para declarar el formato de datos que acepta un intent filter en su etiqueta <data>?

- ☒ A) android:mimeType
- B) android:layout_height
- ☒ C) android:id
- D) android:scheme

10) ¿Qué ventaja otorga el Frame Layout sobre otros layouts?

- A) Evita que los elementos que añades como hijos de layout se solapen
☒ B) Permite reservar un espacio en la pantalla para mostrar un único ítem
C) Es la opción preferible para añadir recursos de imagen
D) Permite distribuir los hijos en una estructura de cuadrícula rectangular
- 11) ¿Cuál de estos objetos pertenece al conjunto subclases de la clase View, también conocidos como widgets?

- A) Button
B) ImageView
C) EditText
☒ D) Todos

- 12) ¿Cuál de las siguientes características se deben declarar en el Manifest.xml?

- A) Estilos y temas.
B) Actividades de la app
☒ C) Todas son correctas
D) Permisos de acceso externos (Internet, etc.)

- 13) En el patrón Modelo - Vista - Controlador:

- A) Modelo y las Vistas no están conectadas.
☒ B) El Modelo, la Vista y el Controlador están conectados.
C) El Modelo y la Vista están separados por el Controlador.
D) La Vista puede enviar información al Modelo.

- 14) En el patrón Model-View-Viewmodel, la Vista contiene:

- A) Una instancia del Modelo.
B) Una instancia del Viewmodel
C) Una instancia del Model and Viewmodel
☒ D) Una instancia del Viewmodel y el Observer.

- 15) En un Proyecto de Android que sigue el patrón de diseño Model-View-Viewmodel, el Modelo puede contener:

- A) Objetos Android
B) Una instancia del ViewModel para comunicación.
☒ C) Sólo código puro Java sin objetos Android
D) Una instancia de la Vista para mostrar cualquier actualización en las interfaces.

- 16) El ViewModel en Android puede contener:

- A) Una instancia de la vista para actualizar las interfaces.
☒ B) La clase MutableLiveData para la comunicación con el Model.
C) Sólo código puro Java sin objetos Android.
D) La clase Observer.

- 17) ¿Cuándo es el mejor momento para planificar el paso de software testing?

- A) Después de terminar de implementar el código.
B) Durante la fase de implementación del código.
☒ C) Después de definir/reunir los requisitos.
D) Antes de definir/reunir los requisitos.

- 18) Cuándo es el mejor momento para desarrollar/implementar el paso de software testing?

- A) Después de las tareas de programación.
☒ B) Durante y después de las tareas de programación.
C) Antes de las tareas de programación.
D) Después del diseño de software.

- 19) ¿Cómo se desarrolla mejor la tarea de unit testing?

- A) Mirando el código.
B) Otra persona revise el código.
C) Testeando una serie de parámetros de entrada.
☒ D) Implementando funciones dedicadas de unit testing.

20) ¿Qué es integration testing?

A) Testear múltiples dispositivos.

B) Testear múltiples usuarios.

☒ C) Testear todos los elementos del código conjuntamente.

D) Testear usando varios grupos de datos.