

UF1. [PAC04] Sensores

Para la realización de las siguientes actividades deberéis ayudaros del libro de texto, capítulos 20, 21, 22, 24, 25, 26, 27, 28 y 29, y consultar, si lo creéis necesario, internet.

Debéis subir un único archivo comprimido que contenga UN ÚNICO DOCUMENTO PDF y los proyectos creados para la parte práctica.

Los ejercicios tienen la siguiente puntuación: los ejercicios de la parte teórica tienen un valor de 0,5 puntos, mientras que el primer ejercicio práctico vale 1,5 puntos y el segundo 2 puntos.

Se valorará positivamente el correcto uso del nombre en los nombres de variables, métodos y clases.

Recordad que la fecha límite para la entrega de esta PAC es el 25 de NOVIEMBRE.

Sugerencia: Para la realización de estos ejercicios es necesario utilizar Android Studio.

Información: Cualquier PAC copiada y/o en la que se haya utilizado “copy-paste” de código ya escrito será puntuada con un 0.

Actividades

PARTE TEÓRICA

1. ¿Cómo y cuándo se termina una operación de arrastre?

Si bien para comenzarla se llama a la función `startDrag()`, se termina de forma automática por parte del SSOO al dejar caer el elemento en alguna vista.

2. ¿Qué es la sombra y por qué es obligatorio crear una?

La sombra es el mecanismo que tiene el usuario para determinar por dónde está el elemento que se está arrastrando.

3. ¿Puede cualquier vista recibir un objeto arrastrado?

Verdadero, pero se tiene que registrar para ello y al recibir el primer evento comunicar al SSOO al dejar caer el elemento en alguna vista.

4. Describe los tipos de servicios.

Servicios que corren durante todo el tiempo (Ejecutados) y servicios que se lanzan cuando se solicitan (Enlazados). El primer caso los lanza el SSOO o la aplicación y se deben parar ellos mismos o una actividad, mientras que los segundos se terminan en el momento de finalizar el trabajo.

5. Mecanismos de comunicación entre un servicio y el usuario.

Toast y mensajes en la barra.

6. ¿Qué es un proveedor de contenido?

Un mecanismo para exponer nuestros datos internos al resto de aplicaciones, no son necesarios si solo se prevé hacer un uso interno de los datos de la aplicación.

7. ¿Qué es una clase contractual?

Es una clase abstracta que se distribuirá con nuestra aplicación con los metadatos necesarios para acceder al proveedor de contenido por parte de otras aplicaciones. No es obligatoria pero facilitará la labor de programación a los demás mucho.

8. ¿Qué es un broadcast receiver?

Es un componente que puede recoger mensajes desde cualquier aplicación.

9. Tipos de mensajes que puede recoger un broadcast receiver.

Síncronos y asíncronos.

10. Verdadero o falso: Los sensores siempre estarán presentes si existen en el dispositivo.

Falso, el usuario tiene la posibilidad de activar o no un sensor, por lo que es obligatorio comprobar su disponibilidad.

11. Tipos de sistemas de posicionamiento.

Basados en satélites, basados en redes, bien wi-fi o bien de telefonía.

12. ¿Qué es el Google Maps API?

Una interface de programación de Google que proporciona localización.

13. Tipos de alertas de proximidad.

De entrada en un radio y posición, o de salida.