



Centro Universitario Felipe Carrillo Puerto

Licenciatura En Informática Y Sistemas De Seguridad

Desarrollo de aplicaciones móviles

6to. Cuatrimestre

Alumnos:

Miguel Antonio Rodríguez Cob

Examen parcial 1



CENTRO ESCOLAR FELIPE CARRILLO PUERTO
DIRECCIÓN GENERAL
LICENCIATURA EN INFORMÁTICA Y SISTEMAS Y SEGURIDAD
ASIGNATURA: DESARROLLO DE APLICACIONES MÓVILES II
1er EXAMEN PARCIAL
Periodo del 6 de mayo al 13 de agosto de 2024
Periodo de evaluación del 6 de mayo al 18 de junio de 2024

LOGO DE LA
LICENCIATURA

91

Nombre del alumno: Miguel Antonio Rodríguez Cob

Nombre del docente: M.I. ERIK EDUARDO LARA ROMERO

Turno: MATUTINO

Grado: Sexto

Grupo: Fecha: 10/06/24

Calificación: 100

INSTRUCCIONES:

- ESCRIBIR TU NOMBRE COMPLETO
- RESPONDER EL EXAMEN CON BOLIGRAFO NEGRO O AZUL
- NO SE ACEPTAN TACHADURAS
- SE ANULARÁ EL EXAMEN SI SE TE SORPRENDE COPIANDO
- PON EL CELULAR EN SILENCIO
- LEE CUIDADOSAMENTE LAS INSTRUCCIONES Y CONTESTA LO QUE SE TE PIDE

I. Relaciona ambas columnas y escribe la respuesta correcta en el paréntesis según corresponda. (1 pts. c/u Total: 15 pts.)

- 1- (C) Número total de píxeles físicos en una pantalla.
- 2- (E) Mecanismo que usa Android para aplicar un estilo
- 3- (F) Clase mediante la cual se pueden capturar eventos como el click.
- 4- (9) Colección de propiedades que especifican el aspecto y formato de una vista.
- 5- (b) Cantidad de píxeles en un área física de la pantalla (DPI).
- 6- (a) Colección principal de los elementos del menú para una actividad.
- 7- (h) Carpeta donde se guardan los recursos necesarios con las especificaciones requeridas.
- 8- (K) Control que permite al usuario escribir un texto.
- 9- (j) Control que se utiliza para mostrar al usuario un texto.
- 10- (d) Píxeles independientes de la densidad, es una unidad de pixel virtual que escala el tamaño físico de un pixel a 160dpi.
- 11- (l) Píxeles independientes de la densidad, es una unidad de pixel virtual que escala el tamaño físico de un pixel a 160 dpi pero escalada en función del tamaño de la letra configurada.
- 12- (i) Menú modal anclado a un objeto View, aparece debajo de la vista si existe espacio.
- 13- (m) Propiedad con la que se identificara de forma única la vista.

- a) View -
- b) Densidad de pantalla -
- c) Resolución -
- d) dp -
- e) Temas -
- f) Menú de opciones -
- g) Estilo -
- h) res -
- i) Menu PopUp -
- j) TextView -
- k) EditText -
- l) sp -
- m) android:id -
- n) RadioButton -
- ñ) android:hint -

13

Miguel Antonio Padilla Cob

- 14- (☒) Controles que permite al usuario seleccionar una opción de un conjunto.
15- (☒) Pre-definidos que permite poner un texto por default mientras el campo este vacío.

II. Explica que es y para qué sirve un adaptador? Referente a spinners y Listas. (15 pts. c/u Total 20 pts.)

13

III. Menciona los pasos a seguir en orden para generar una lista de elementos verticales: (10 pts. Total 10 pts.)

6

IV. Coloca el código de un estilo con nombre style6 que tenga las siguientes propiedades: color de fondo azul, tamaño de letra 12 sp, color de letra rojo, ancho 35dp y alto 45dp. (10 pts. Total 10 pts.)

9

V. Menciona y explica los pasos para generar un menú (Elige el tipo de menú que gustes) (10 pts. Total 10 pts.)

10

VI. Practica. (10 pts. Total 40 pts.)

Elabora un aplicación que muestre una lista con todos los planetas, cada elemento de la lista deberá tener una imagen; al menos es decir mostrara un imagen del planeta y su nombre, al dar click sobre algun elemento de la lista nos llevara a un apantalla donde nos mostrara una imagen mas amplia del planeta, mostrara tambien una descripción del planeta así como algunas de sus características.

Esta pantalla tambien debe contener un boton para regresar a la lista de planetas.

Se subira a GITHUB, y se compartiran el link para clonar el repositorio de su proyecto.

38

5. Para último paso para mostrar los cuentas cuando el usuario selecciona un elemento del menú, se establece un sistema utilizando el vector de menú item ClickListener(). Dentro de este sistema, ~~podemos~~ mostrar diferentes acciones según lo que se selecciona en el menú.

Continuación Pregunta V.

Continuación Pregunta III.

En la actividad se establecen los datos para cada elemento asociados al array y el audiotador para que se verifiquen en el menú como array los nombres de nuestro array para usarse la vista con la información correspondiente.

Miguel Antonio Rodríguez Cob

IV.

```
<style name="style6">
  <item name="android:background">@color/azul</item>
  <item name="android:textSize">12sp</item>
  <item name="android:textColor">@color/rojo</item>
  <item name="android:layout_width">35dp</item>
  <item name="android:layout_height">45dp</item>
```

</style>

resource)

V. Menu Popup Pasos

1. Creemos un archivo xml que va a representar el menu que queremos mostrar, asi definiremos los elementos del menu.
2. en la actividad donde mostraremos el menu definimos un metodo para iniciar y mostrar el menu, esto lo hacemos de las veces se hace en respuesta de un evento del usuario como un click en un boton o desplazamiento.
3. dentro del metodo que se cree utilizamos el objeto PopupMenu para iniciar el archivo de menu que se cree en el primer paso. Esto carga los elementos del menu en la memoria de la app.
4. despues de iniciar el menu llamamos al metodo 'show()' en el objeto PopupMenu para mostrar el menu pop-up en la ubicacion deseada. Esto se hace pasando el 'view' que desencadena la apertura del menu como el boton si se hizo click.

II. Listas: Nos sirve para llamar nuestra lista especial con un solo
y
IV. ^{SP} ^{primeros} text view para llenarlo con la información del adaptador
ya que en él establecemos toda la información que
contendrá la información un ejemplo en mi lista
de frutas. Establece lo que aparecerá en la lista
Foto, descripción y nombre.
En sí permite el acceso a los elementos del arreglo
y es el responsable de crear una vista para cada
elemento.

III. Pasos

1. tener un adaptador que solo contenga un text view para llenarlo
es decir, el primer xml llamará al adaptador los datos para
crear una vista a cada elemento.
2. se crea un adapter el cual permite el acceso a los elementos
del arreglo y crea las vistas para cada elemento.
podemos usar un array adapter y lo podemos añadir con el
método setAdapter
3. con esto nuestro primer archivo llama a la lista creada en el
adaptador, y creando un archivo más con la información de
cada elemento tendremos ya la lista completa usando
el array y el setAdapter
4. en nuestro archivo principal se declara el adaptador y el
array y se llama en el xml inicial el xml de la
lista item para que se refleje la lista.

