

## Tasca 1

Link Github:

[https://github.com/cristianjimenezhernandezdev/Android\\_Programacio\\_DAM2b.git](https://github.com/cristianjimenezhernandezdev/Android_Programacio_DAM2b.git)

Link Drive per Campalans

[https://drive.google.com/drive/folders/1TfpE3NCBj8FdwE\\_DX6S87gykLIFmO-0I?usp=sharing](https://drive.google.com/drive/folders/1TfpE3NCBj8FdwE_DX6S87gykLIFmO-0I?usp=sharing)

### Preguntes:

1. Quin tipus de Layout has fet servir i perquè?

He fet servir el ConstraintLayout, és el que posa per defecte el android Studio en obrir el projecte, Permet anar posant elements diferents de layout amb el mode Disseny i poder editar parts del codi directament. Permet manipular els elements tant a nivell gràfic com a nivell de codi i te maneres automàtiques d'alinejar elements fàcilment.

2. Has pogut fer tot des de l'editor gràfic o has hagut de fer servir l'editor XML. Posa un exemple que t'hagis trobat?

He fet la majoria des de l'editor gràfic del Designer, però hi ha hagut elements que no podia fer, sobretot lògiques i crear funcions.

Parts que he hagut de fer amb codi:

- Elements menors dins del xml com pot ser canvis en les mides, posicions o introduir codi directament com el `layout_weight` o elements per després referenciar al `.kt` com és el `Top_toBottomOf="@id/rgSexe"`.
- Codi al `.kt` destinat a fer que funcionin certes parts, les principals han estat vincular les funcions del calendari amb la Caixa de text i els botons de neteja i enviar, que després de definir el nom de la funció per el menú de atributs del Designer s'ha hagut d'implementar la funció al `.kt`

3. Pensa com faries una matriu 3x3 fent servir només LinearLayouts

Faria servir la lògica que ja conec per fer matrius amb arrays. El LinearLayout te una orientació per atributs que és vertical i l'altre horitzontal. La vertical correspondria a les columnes i la horitzontal a les files. Així doncs posaria una vertical i dins, tres horitzontals amb les propietats que volguessis per cada part horitzontal.

4. Fan alguna acció o animació els elements que has posat a l'aplicació?

Fan accions bàsiques del component: el camp de data obre un DatePickerDialog en fer click, el TextInputLayout de contrasenya té l'ull per mostrar u ocultar, el RadioGroup fa selecció exclusiva, el Switch activa el ON/OFF i els botons executen la funció a onClick.

5. Què és `layout_width` i `layout_height`? Quines opcions tenen? Com afecta en cada cas?

Són les mides del widget. Els valors principals són `wrap_content` (ajusta automàticament al contingut), `match_parent` (ocupa tot l'espai del parent) i posar ho a `0dp` és útil per ajustar. Amb `ConstraintLayout`, `0dp` vol dir que s'adapti als constrains amb `LinearLayout`, `0dp` es fa servir amb `layout_weight` per repartir l'espai que ocupen.

6. Què és `match_parent` i `wrap_content`?

`match_parent`: el fill ocupa tot l'ample/alt del pare.

`wrap_content`: només l'espai necessari pel seu contingut. Jo faig servir `match_parent` quan vull que ompli tot, i `wrap_content` quan vull mida del propi contingut.

7. I el paràmetre `gravity`? I el paràmetre `layout_weight`?

`gravity` és per alinear el contingut intern d'un widget (ex.: text centrat dins d'un Button). `layout_gravity` (en `LinearLayout`) és per alinear el widget dins del seu pare.

`layout_weight` (dins el `LinearLayout`) reparteix espai.

Per exemple amb `width=0dp` i `weight=1` per tres parts, cada una ocupa la seva part proporcional, un terç de la fila.

Jo en el meu cas no he fet servir el `gravity`, ho he fet amb el `constraintLayout`. (tampoc coneixia res més he anat fent..)

8. (EXTRA) Busca com fer que quan premis el botó enviar surti un missatge flotant(`SnackBar`) que digui: "Informació Enviada"

Jo en el meu Layout ja ho he fet semblant però amb un Toast en comptes del SnackBar. Bàsicament fer-ho amb el Toast és gairebé el mateix però canviant una funció per una altre

Necessitem fer import per les funcionalitats

```
import com.google.android.material.snackbar.Snackbar
```

Fem un mètode:

```
fun enviarFormulari(view: View) {  
    Snackbar.make(view, "Informació Enviada",  
        Snackbar.LENGTH_SHORT).show()  
}
```