

## TAREA 4.4 - Organización de informes con Datapane

Hasta ahora, al crear informes con Datapane, hemos visto que los componentes que añadimos (tablas, gráficos, texto, imágenes, etc.) se colocan **uno debajo de otro**, en vertical.

Este comportamiento es el más sencillo y, en muchos casos, puede ser suficiente.

Sin embargo, cuando un informe empieza a crecer, **no siempre es la mejor opción desde el punto de vista del diseño de interfaces**.

Por este motivo, Datapane ofrece **diferentes mecanismos para organizar los componentes** y mejorar la claridad y la experiencia del usuario.

En este apartado vamos a ver las tres opciones principales:

- Grupos de componentes
- Páginas
- Selectores

### Grupos de componentes

Los **grupos de componentes** permiten organizar varios elementos del informe **en forma de cuadrícula**, es decir, en filas y columnas.

Esto es útil cuando:

- Queremos mostrar varios elementos relacionados al mismo tiempo.
- Queremos aprovechar mejor el espacio horizontal.
- Los componentes tienen un peso visual similar.

### Cómo funcionan los grupos

Al crear un grupo, indicamos:

- Qué componentes forman parte del grupo.

- Cuántas **filas** y **columnas** tendrá la cuadrícula.

### Ejemplo:

```
1 dp.Report(  
2     dp.Group(  
3         componente1, componente2, componente3, componente4,  
4         columns=2, rows=2))
```

En este caso:

- El informe contiene un único grupo.
- El grupo tiene **4 componentes**.
- Se organizan en **2 filas y 2 columnas**.

Visualmente, el resultado sería una cuadrícula con cuatro bloques del mismo tamaño.

Para organizaciones más complejas, Datapane permite:

- Incluir **grupos dentro de otros grupos**.

Esto permite crear diseños más elaborados, aunque conviene no abusar para no complicar la lectura del informe.

## Páginas

Otra forma de organizar un informe es mediante **páginas**.

Las páginas permiten dividir un informe en **secciones independientes**, de forma que:

- Cada página contiene su propio conjunto de componentes.
- El usuario puede cambiar de página desde la parte superior del informe.

La idea es similar a:

- Las **hojas de una hoja de cálculo**.
- O las **pestañas de una aplicación**.

## Cuándo usar páginas

Las páginas son adecuadas cuando:

- El informe contiene mucha información.
- Cada conjunto de gráficos o tablas responde a un objetivo distinto.
- No queremos que todo aparezca en una sola vista.

### Ejemplo:

```

1 dp.Report(
2     dp.Page(
3         title="Página 1",
4         blocks=[componente2, componente3]
5     ),
6     dp.Page(
7         title="Página 2",
8         blocks=[componente3, componente4]
9     )
10 )

```

En este ejemplo:

- El informe tiene **dos páginas**.
- Cada página tiene:
  - Un título (`title`), que es lo que verá el usuario para navegar.
  - Una lista de componentes (`blocks`) que forman esa página.

### Restricción importante

Hay dos normas clave que conviene recordar:

- No se pueden anidar páginas dentro de otras páginas.
- Las páginas deben estar **directamente dentro de `dp.Report`**.

Esto evita estructuras confusas y mantiene una navegación clara.

## Selectores

Los **selectores** se utilizan cuando queremos mostrar **varios componentes alternativos**, pero **no todos a la vez**.

En lugar de saturar la pantalla:

- El usuario elige qué componente quiere ver.
- El resto permanece oculto.

## Cuándo usar selectores

Los selectores son especialmente útiles cuando:

- Tenemos distintas vistas de los mismos datos.
- Queremos alternar entre una tabla y un gráfico.
- Los componentes cumplen una función similar.

Ejemplo:

```
1 dp.Report(  
2     dp.Select(blocks=[componente1, componente2, componente3])  
3 )
```

En este caso:

- El informe incluye un selector con tres componentes.
- El usuario puede elegir cuál visualizar.

Cómo se muestra el selector

Datapane adapta la forma del selector automáticamente:

- Si hay **menos de cinco opciones** → se usan **pestañas**.
- Si hay **cinco o más opciones** → se usa un **desplegable**.

La etiqueta que aparece para cada opción se toma de la propiedad `label` de cada componente.

Desde el punto de vista del diseño de interfaces (DI)

Elegir bien cómo organizar los componentes es tan importante como elegir el gráfico adecuado.

Buenas prácticas:

- No mostrar más información de la necesaria a la vez.
- Agrupar elementos relacionados.
- Facilitar que el usuario decida qué quiere ver.

Un informe bien organizado:

- Se entiende más rápido.
- Reduce errores de interpretación.

- Mejora la experiencia del usuario.

## Enunciado de la tarea

En esta tarea tendrás que diseñar un informe con Datapane que use las distintas posibilidades de organización de componentes comentadas (grupos, páginas y selectores). Como origen de datos utiliza este fichero CSV utilizados en el tema:



DI\_U05\_A02\_PP\_E\_01.csv  
Text File  
1.6 KB

## Entrega

### 1. Archivo Python (`apellido_nombre_T4.4.py`)

Un archivo con el programa completo que:

- Cargue un fichero CSV utilizado en el tema.
- Genere un informe con **Datapane**.
- Utilice **al menos una vez** cada uno de los siguientes elementos:
  - `dp.Group`
  - `dp.Page`
  - `dp.Select`
- Cree correctamente el informe y lo guarde como archivo HTML.
- Esté comentado de forma básica para explicar qué hace cada parte.

### 2. Informe Datapane generado (`apellido_nombre_T4.4.html`)

El archivo HTML resultante del programa:

- Debe haberse generado mediante `dp.Report`.
- Debe abrirse correctamente en un navegador.
- Debe reflejar la organización de componentes:
  - grupos,
  - páginas,

- y selectores.

⚠ El contenido del informe debe corresponder con los datos reales del CSV utilizado.

### 3. Documento pdf (`apellido_nombre_T4.4.pdf`)

Un documento (o comentario al inicio del código) donde se explique:

- Cómo se han organizado los componentes del informe:
  - qué se ha puesto en grupos,
  - qué se ha separado en páginas,
  - y qué se ha incluido en selectores.
- Por qué se ha elegido esa organización (1-2 frases por elemento).
- Una webgrafía con **al menos una fuente consultada**, por ejemplo:
  - Documentación oficial de Datapane.
  - Tutoriales utilizados.
  - Recursos online sobre gráficos o diseño de informes.

## Rúbrica



Criterio evaluado	Excelente	Adequado	Básico
<b>Estructura general del informe</b>	El informe tiene una estructura clara, lógica y bien organizada. El orden de los componentes facilita la lectura.	La estructura es correcta y comprensible, aunque podría organizarse mejor.	La estructura es poco clara y mejorable.
<b>Uso de grupos (dp.Group)</b>	Se utilizan grupos correctamente, con una organización adecuada en filas y columnas.	Se utilizan grupos, pero la organización podría mejorarse.	Uso muy poco justo de los grupos.
<b>Uso de páginas (dp.Page)</b>	Las páginas están bien definidas, con títulos claros y contenido coherente en cada una.	Las páginas están correctamente creadas, aunque podrían organizarse mejor.	Uso mínimo de páginas, con falta de coherencia.
<b>Uso de selectores (dp.Select)</b>	El selector se utiliza de forma adecuada y mejora la experiencia del usuario.	El selector funciona correctamente, aunque su uso es mejorable.	Uso muy poco justo del selector.
<b>Funcionamiento técnico del informe</b>	El informe se genera sin errores, el HTML se abre correctamente y los datos son coherentes con el CSV.	El informe funciona, con pequeños detalles técnicos mejorables.	El informe genera errores menores o incoherencias.
<b>Explicación y razonamiento (DI)</b>	La explicación es clara y justifica correctamente la organización del informe.	La explicación es correcta, aunque poco detallada.	Explicación breve o poco precisa.
<b>Webgrafía</b>	Se incluye	Se incluye	Webgrafía

weogratiá	weogratiá mínima.	ciara o ma presentac
adecuada y relacionada con la tarea.		

## Criterios de evaluación

### RA5. Crea informes evaluando y utilizando herramientas gráficas:

- a) Se ha establecido la estructura del informe.
- b) Se han generado informes básicos a partir de diferentes fuentes de datos mediante asistentes.
- d) Se han incluido valores calculados, recuentos y totales.
- e) Se han incluido gráficos generados a partir de los datos.
- f) Se han utilizado herramientas para generar el código correspondiente a los informes de una aplicación.
- g) Se ha modificado el código correspondiente a los informes.