Nexus — Elemento: Chart

¿Qué es?

Chart es un componente de tipo **Data**. Muestra un gráfico de datos con múltiples tipos de visualización disponibles. Al arrastrarlo al lienzo viene con un ejemplo precargado (“Monthly Revenue”, gráfico de barras con datos de Enero a Mayo).

Propiedades

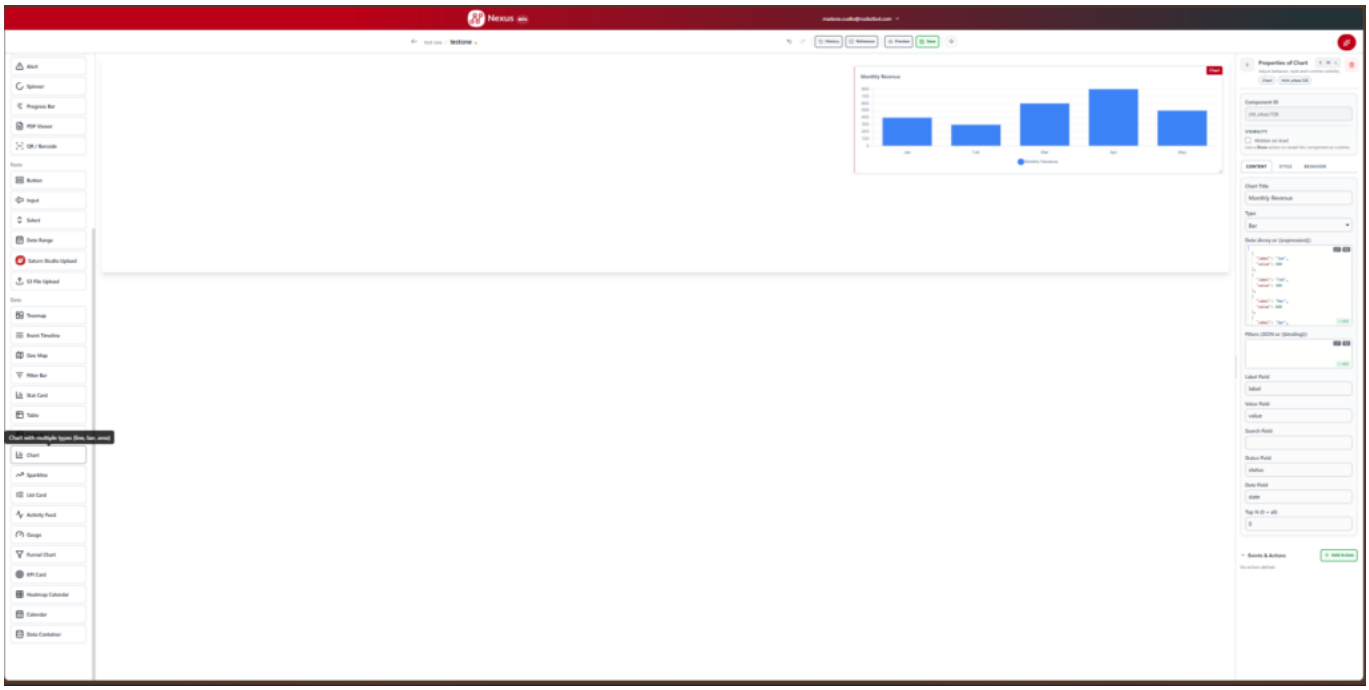
Al seleccionar Chart en el lienzo, el panel derecho muestra **“Properties of Chart”**, organizado en tres pestañas: **Content**, **Style** y **Behavior**.

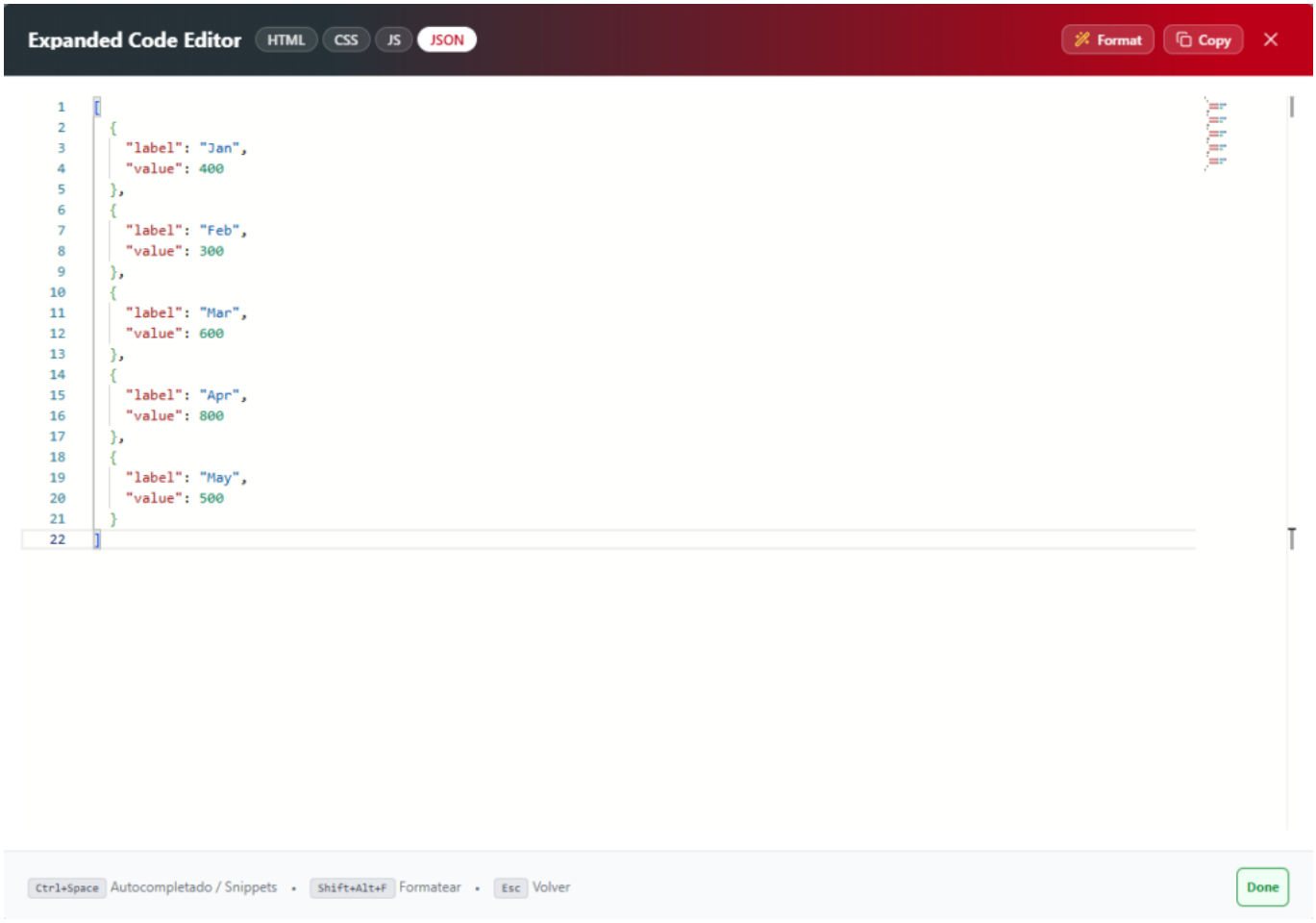
- **Component ID**: identificador único del componente, autogenerado (ej. cht_a4aec728).
- **Visibility → Hidden on load**: casilla para que el componente esté oculto al cargar la pantalla. Puede mostrarse luego mediante una acción de tipo **Show** (ver Events & Actions).

Pestaña Content:

- **Chart Title**: título del gráfico (ej. “Monthly Revenue”).
- **Type**: menú desplegable que define el tipo de gráfico. Opciones: **Bar**, **Line**, **Pie**, **Doughnut**.

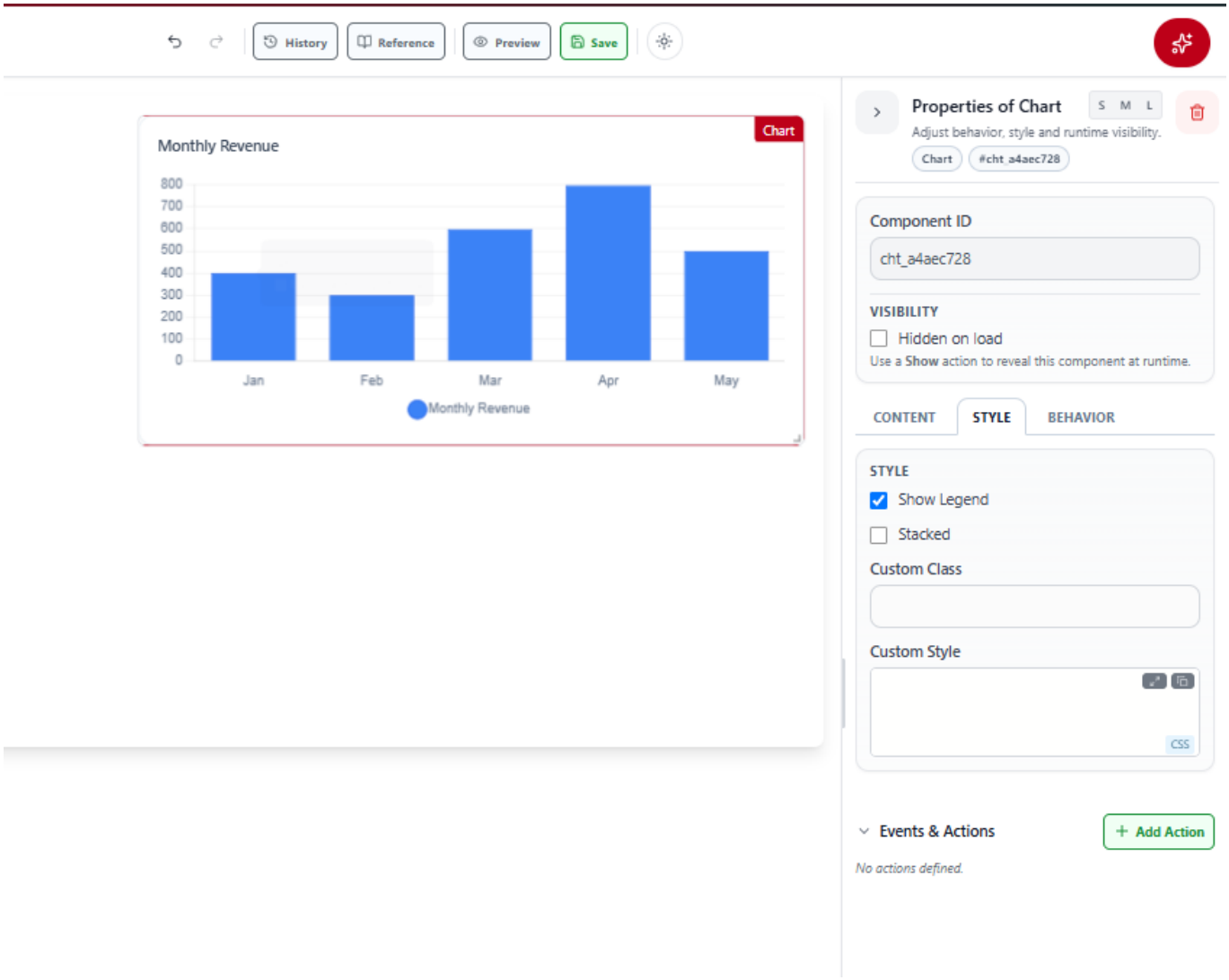
- **Data (Array or {{expression}})**: campo de código donde se cargan los datos del gráfico en formato JSON (array de objetos label / value), o alternativamente mediante una expresión/binding dinámico ({{...}}). Cuenta con el editor de código expandido.
- **Filters (JSON or {{binding}})**: campo para aplicar filtros a los datos mostrados, en JSON o mediante binding.
- **Label Field**: nombre del campo que se usa como etiqueta de cada dato (por defecto: label).
- **Value Field**: nombre del campo que se usa como valor de cada dato (por defecto: value).
- **Search Field**: nombre del campo usado para búsqueda/filtrado por texto (vacío por defecto).
- **Status Field**: nombre del campo usado para filtrar por estado (por defecto: status).
- **Date Field**: nombre del campo usado para filtrar por fecha (por defecto: date).
- **Top N (0 = all)**: limita el gráfico a mostrar solo los N valores más altos (0 = mostrar todos).





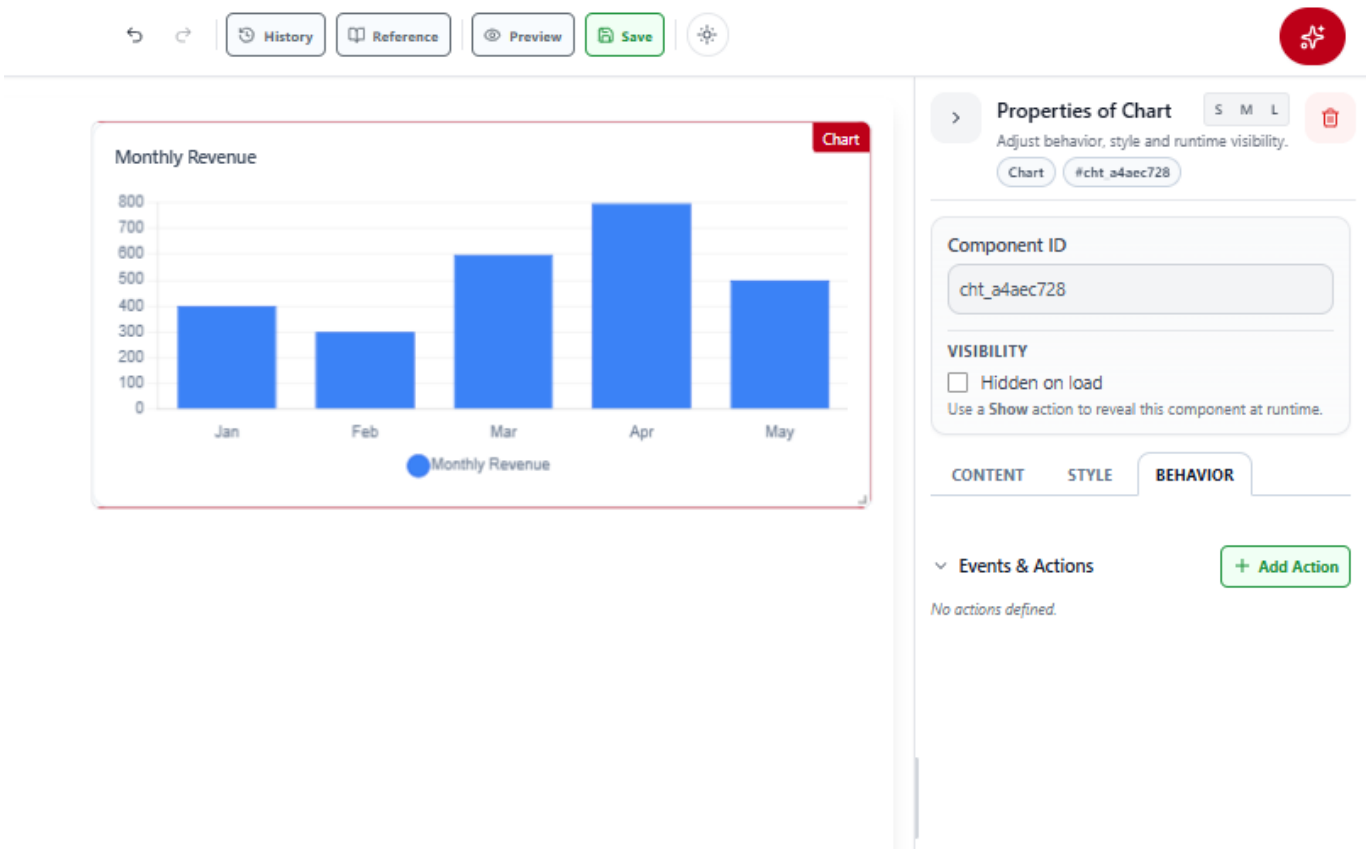
Pestaña Style:

- **Show Legend**: casilla que muestra u oculta la leyenda del gráfico.
- **Stacked**: casilla que apila las series de datos unas sobre otras (relevante en gráficos de barras/líneas con múltiples series).
- **Custom Class**: clase CSS personalizada.
- **Custom Style**: CSS personalizado aplicado directamente a este componente.



Pestaña Behavior:

- **Events & Actions**: sistema para definir comportamientos interactivos. Ver documentación aparte: **Nexus — Events & Actions** .
- **Delete Component** (ícono de papelera, arriba): elimina el componente del lienzo.

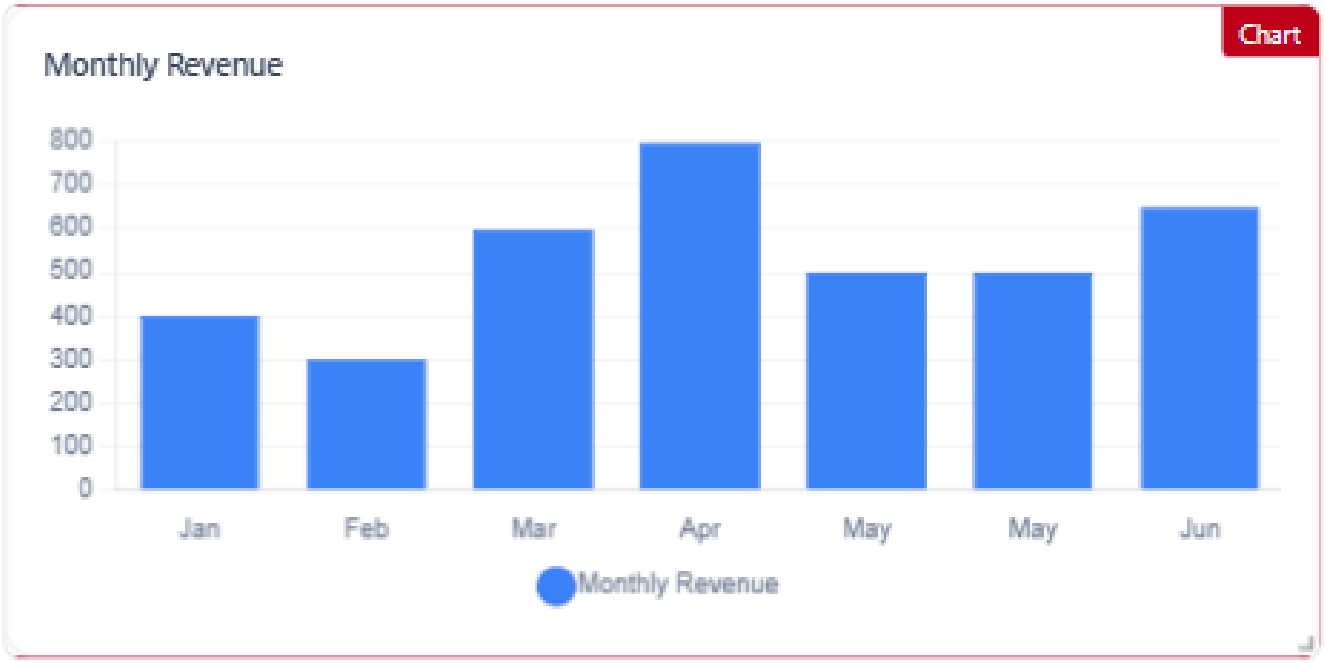


Ejemplo de uso

Para agregar un nuevo dato al gráfico, hay que ubicarse en la última llave } del bloque anterior dentro de **Data**, agregarle una coma, y pegar el nuevo bloque a continuación:

```
[
 {
    "label": "May",
    "value": 500
  },
  {
    "label": "Jun",
    "value": 650
  }
]
```

> ⚠ Importante: la coma va pegada a la llave } del bloque anterior, no al principio del bloque nuevo.



Con **Type** se elige la mejor forma de representar los datos: **Bar** o **Line** para tendencias en el tiempo, y **Pie** o **Doughnut** para mostrar proporciones dentro de un total. Los campos **Label Field** y **Value Field** permiten reutilizar el mismo componente con datos que usen claves distintas a label/value (por ejemplo, datos que vengan de una query con nombres de columna diferentes), sin tener que transformar el JSON. Con **Filters**, **Search Field**, **Status Field** y **Date Field**, el gráfico puede conectarse a un **Filter Bar** para actualizarse dinámicamente según lo que el usuario filtre.