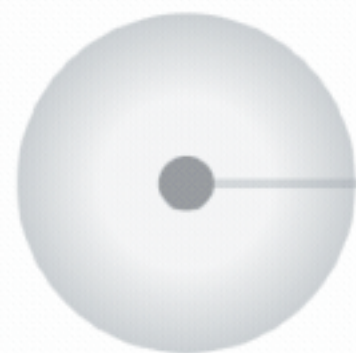


implementing the chart with

Customer



1

Active carts



0

Checking out



0

Purchased

✕ Close custom settings

Customer behavior settings

Page views settings

Total Sessions settings

Polling interval: 10000 (ms)



Increment range: 1-2




```
import { useSprings, animated, config } from "react-spring";
```

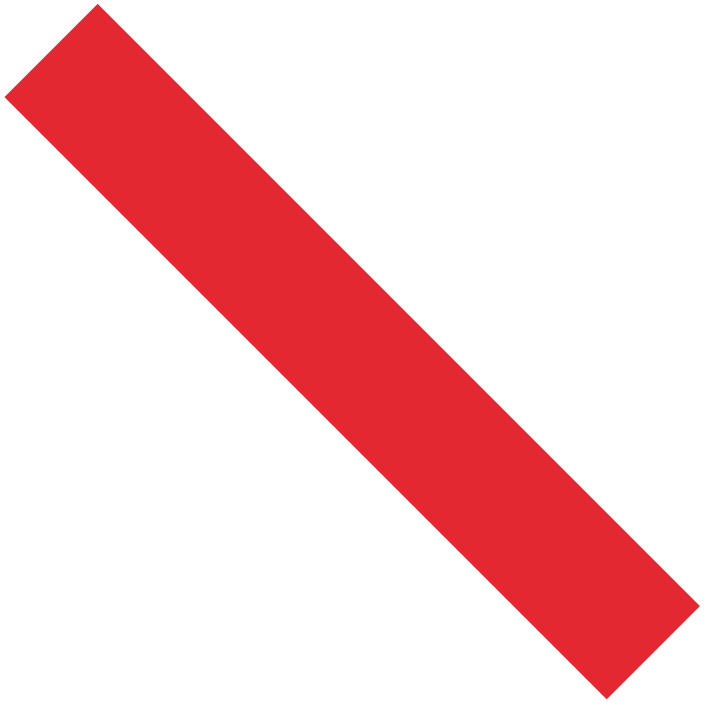
```
const radiusSprings = useSprings(  
  data.length,  
  data.map((dataPoint, index) => ({  
    radius: radiusScale(dataPoint.count),  
    config: config.molasses,  
  })))  
);
```



```
{data.map((dataPoint, index) => (  
  <circle  
    cx={xScale(index)}  
    r={radiusScale(dataPoint.value)}  
    // ...  
  >  
))}
```

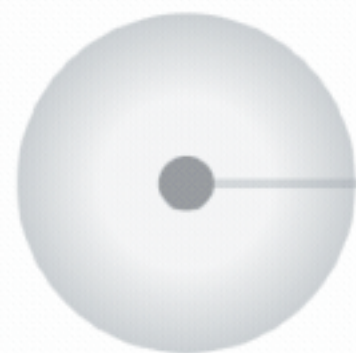


```
{data.map((dataPoint, index) => (  
  <animated.circle  
    cx={xScale(index)}  
    r={springs[index].radius}  
    // ...  
  />  
))}
```





Customer



1

Active carts



0

Checking out



0

Purchased

✕ Close custom settings

Customer behavior settings

Page views settings

Total Sessions settings

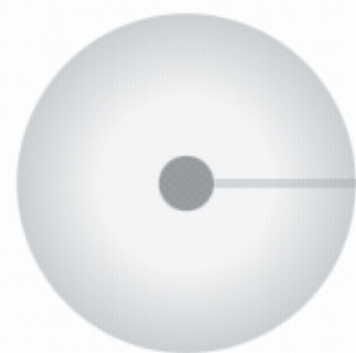
Polling interval: 10000 (ms)



Increment range: 1-2



Customer



1

Active carts



0

Checking out



0

Purchased

✕ Close custom settings

Customer behavior settings

Page views settings

Total Sessions settings

Polling interval: 10000 (ms)



Increment range: 1-2

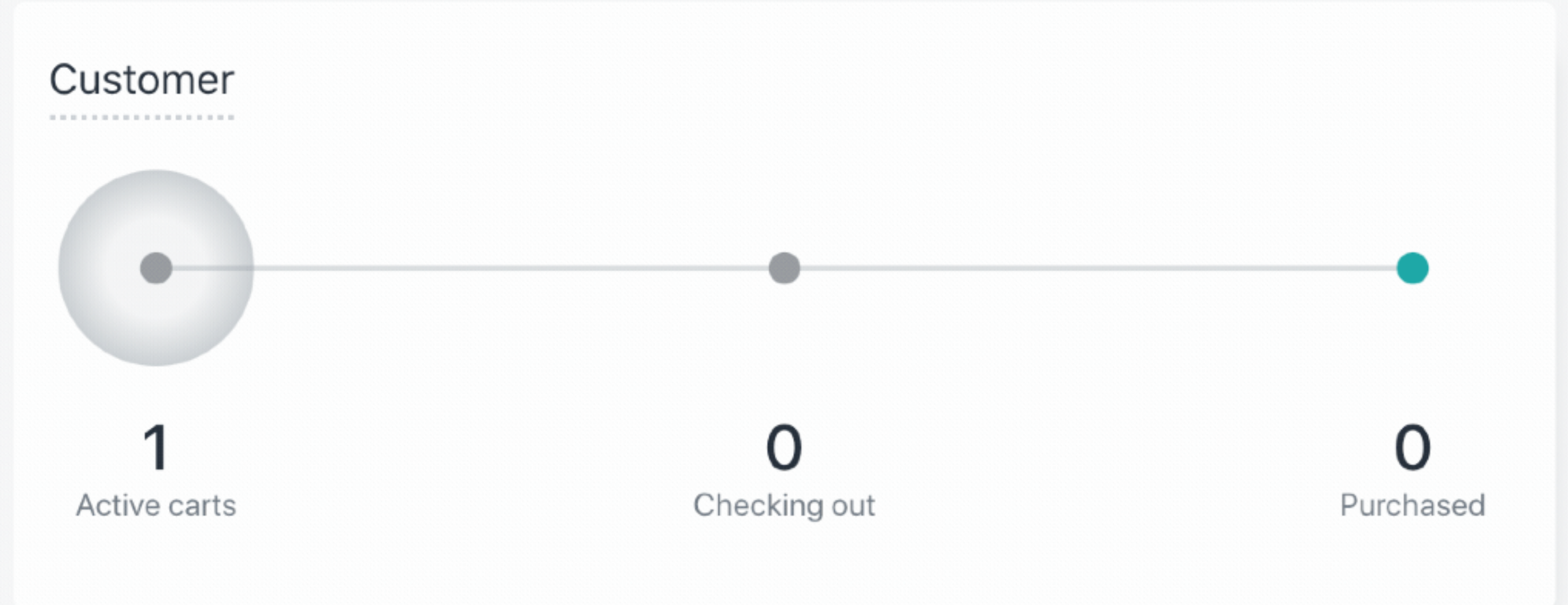


Re-implementing the chart with Canvas

```
import { useSprings, animated, config } from "react-spring";

const radiusSprings = useSprings(
  data.length,
  data.map((dataPoint, index) => ({
    radius: radiusScale(dataPoint.count),
    config: config.molasses,
  })))
);
```

```
{data.map((dataPoint, index) => (
  <animated.circle
    cx={xScale(index)}
    r={springs[index].radius}
    // ...
  />
))}
```



✕ Close custom settings

Customer behavior settings Page views settings Total Sessions settings

Polling interval: 10000 (ms)

Increment range: 1-2

Re-implementing the chart with Canvas



100.02