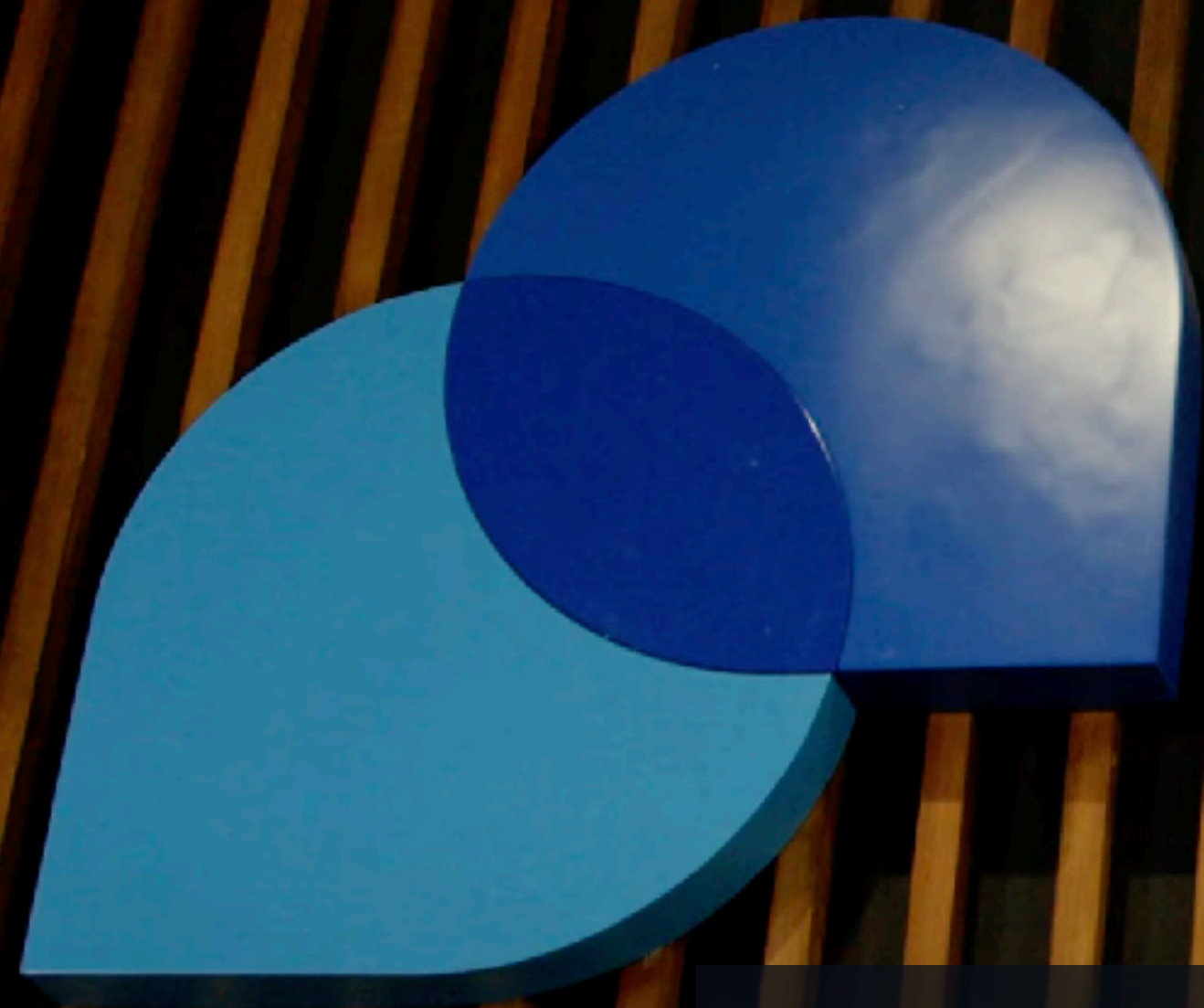


Jak pomalować ślonia

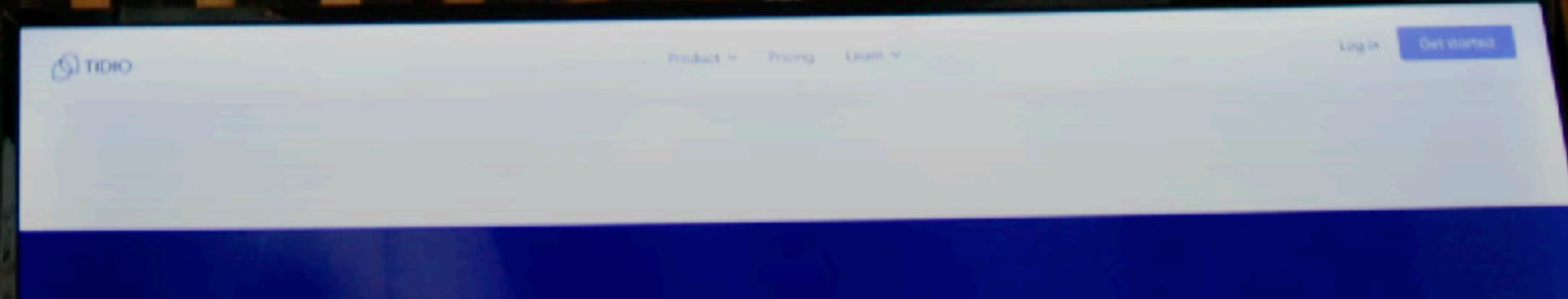
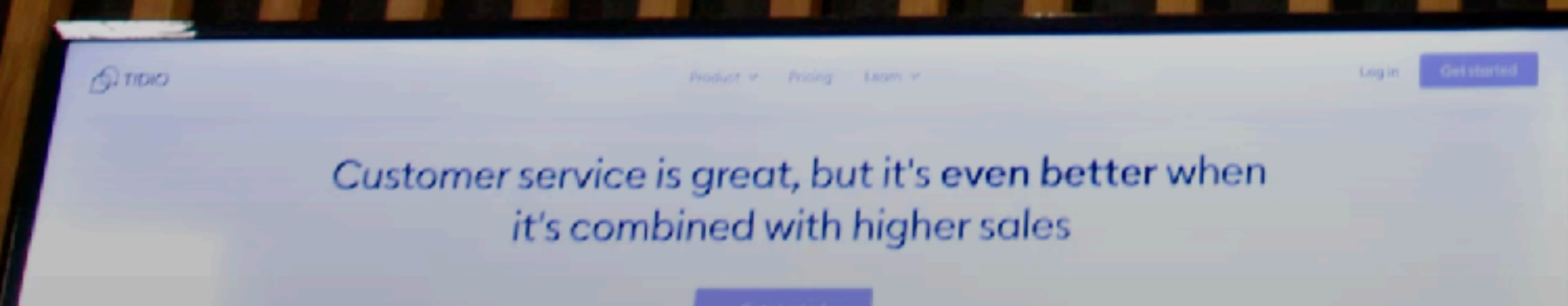
Design System w dojrzałym produkcie

Piotr Wcisłek



TIDIO

HISTORIA INSPIROWANA FAKTAMI





WYTYCZNE
ZASADY

- 1. Nie używać telefonu komórkowego
- 2. Nie używać komputera do innych celów
- 3. Nie używać internetu do innych celów
- 4. Nie używać sieci Wi-Fi do innych celów
- 5. Nie używać sieci Bluetooth do innych celów
- 6. Nie używać sieci NFC do innych celów
- 7. Nie używać sieci RFID do innych celów
- 8. Nie używać sieci ZigBee do innych celów
- 9. Nie używać sieci Z-Wave do innych celów
- 10. Nie używać sieci LoRa do innych celów

C







COPPER DOOR TAVERN
COPPER DOOR TAVERN
COPPER DOOR TAVERN

POST
HILLS

SPEED CHECKED
BY RADAR

160
EAST
22ND

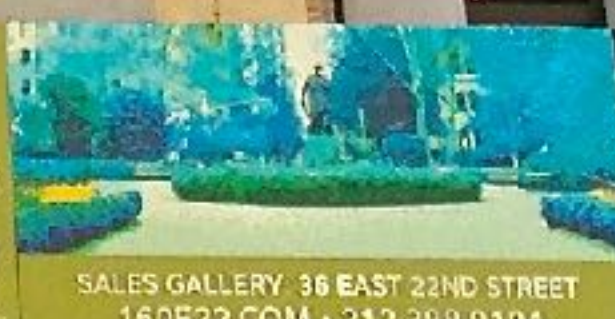


A LIMITED NEW CONDOMINIUM IN THE HEART
OF GRAMERCY WITH MODERN FINISHES
AND A CLASSIC SUITE OF AMENITIES.

160E22.COM

John Brothers
CORPORATION

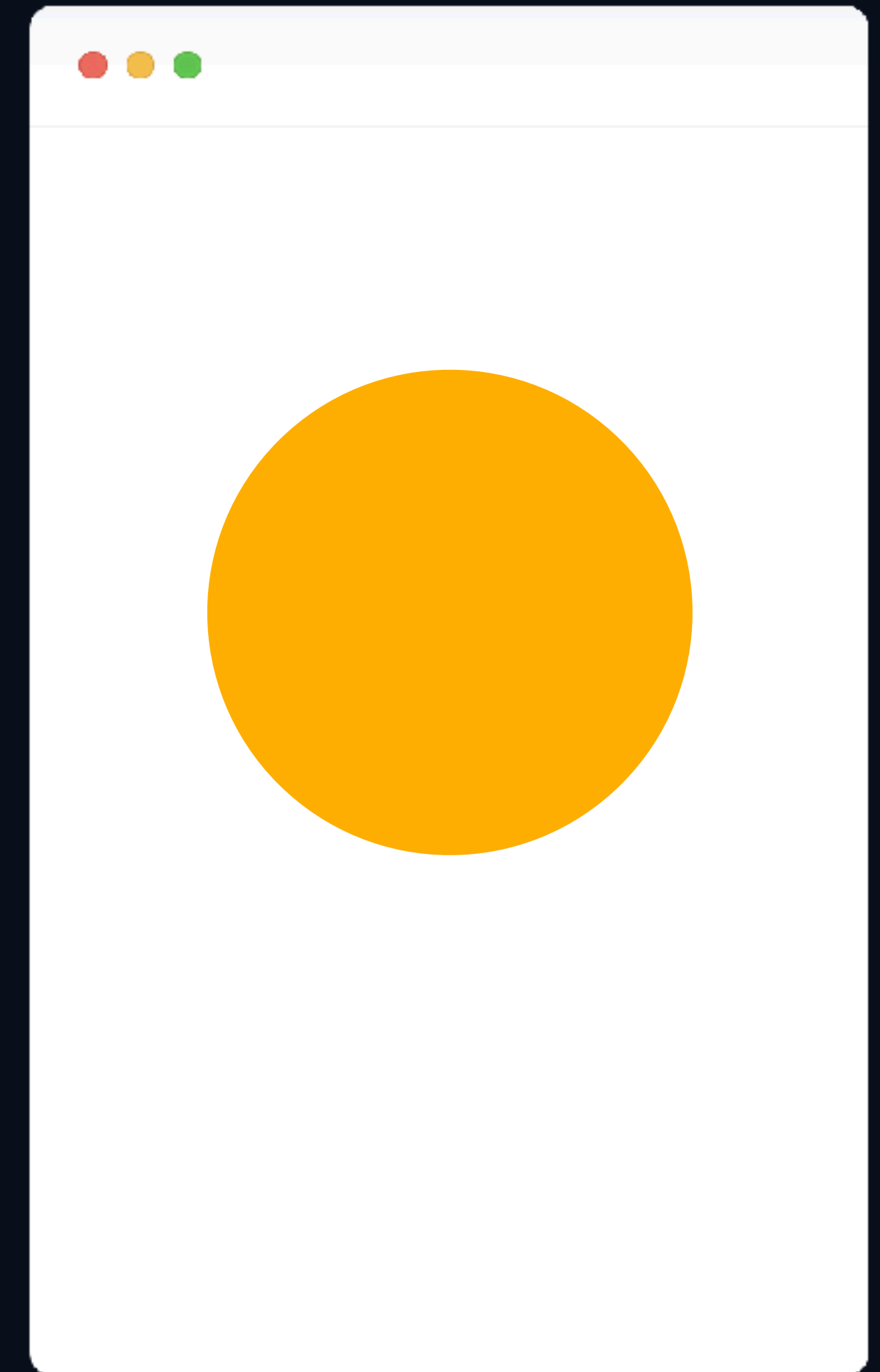
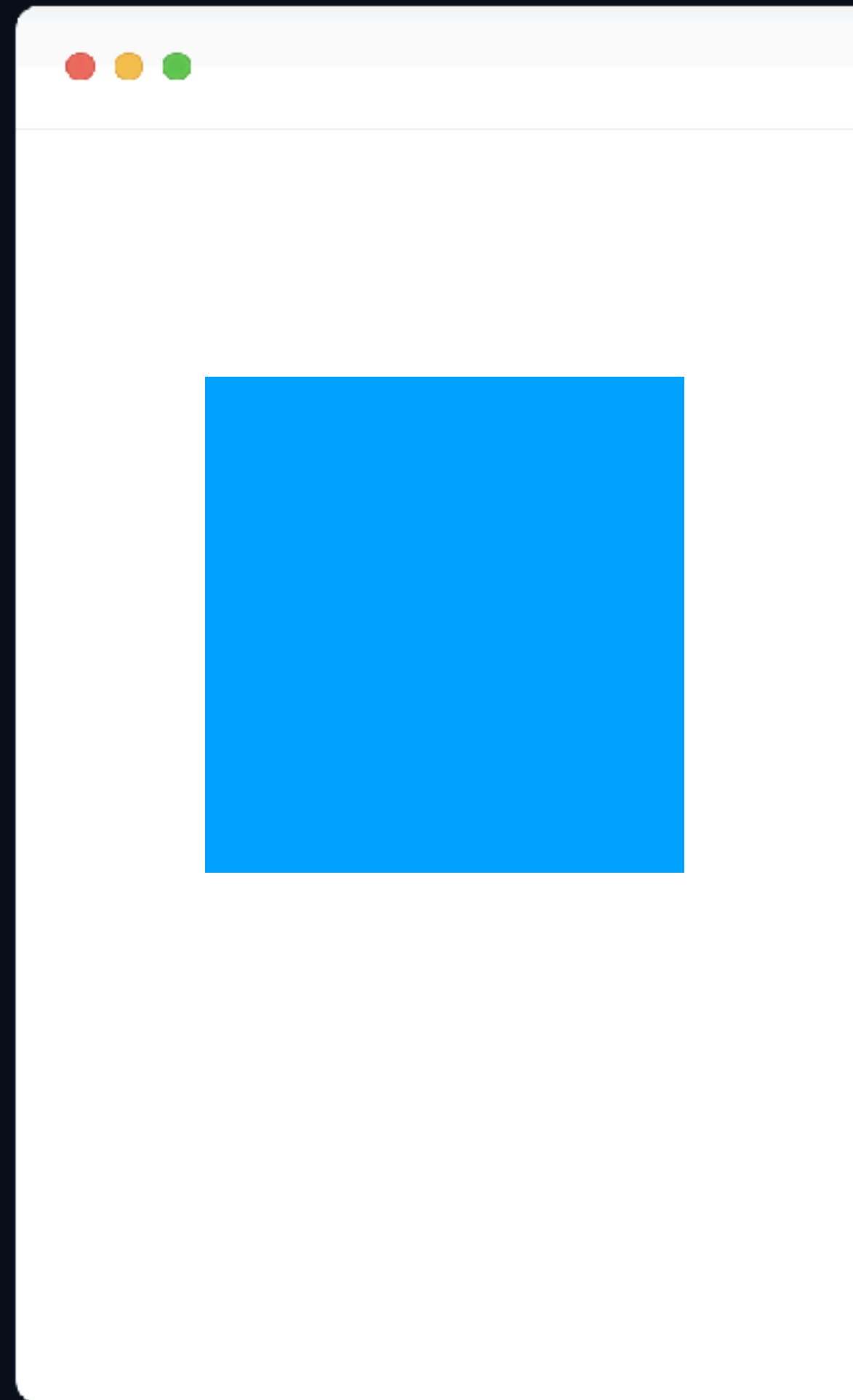
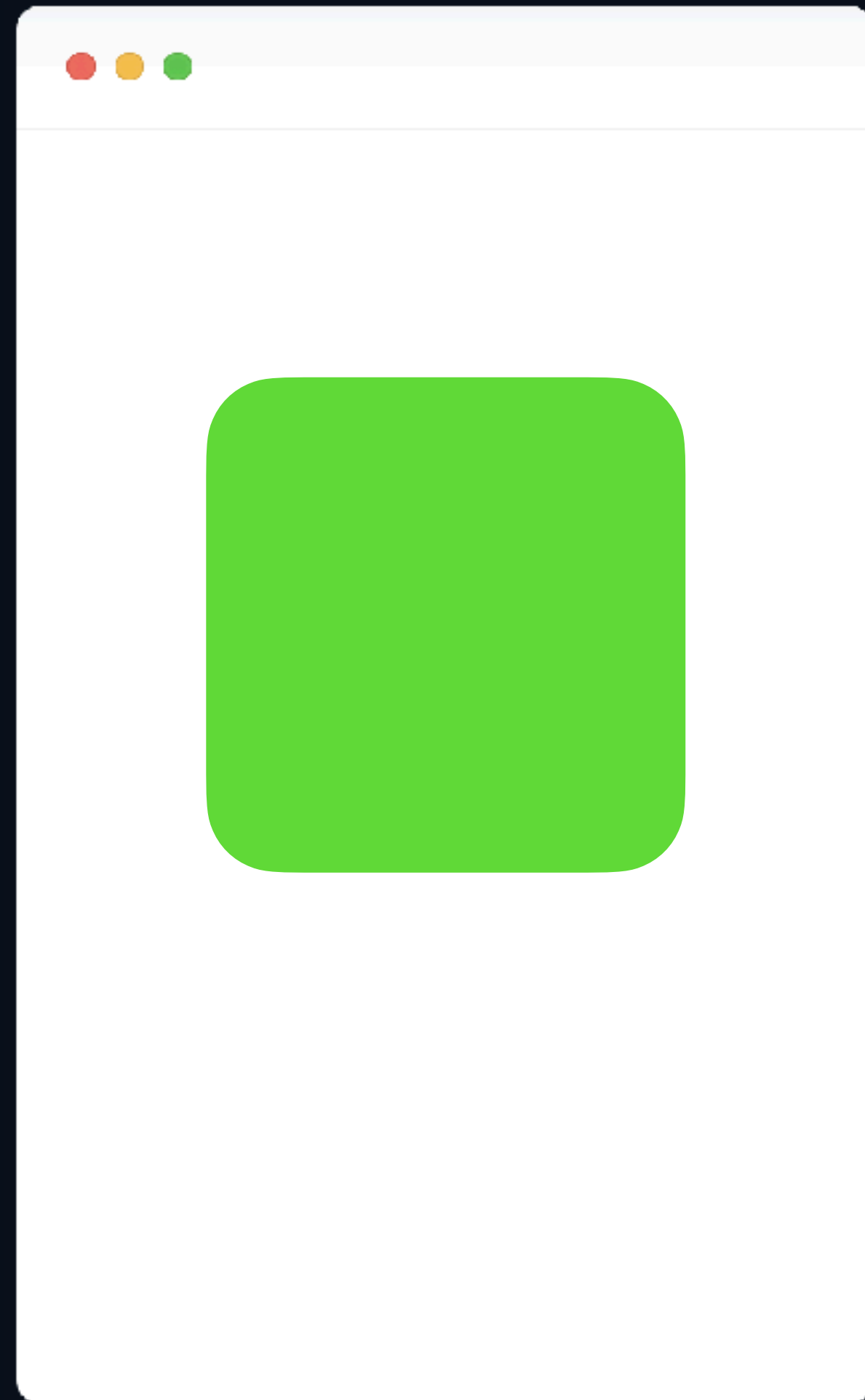
UNIQUELY GRAMERCY



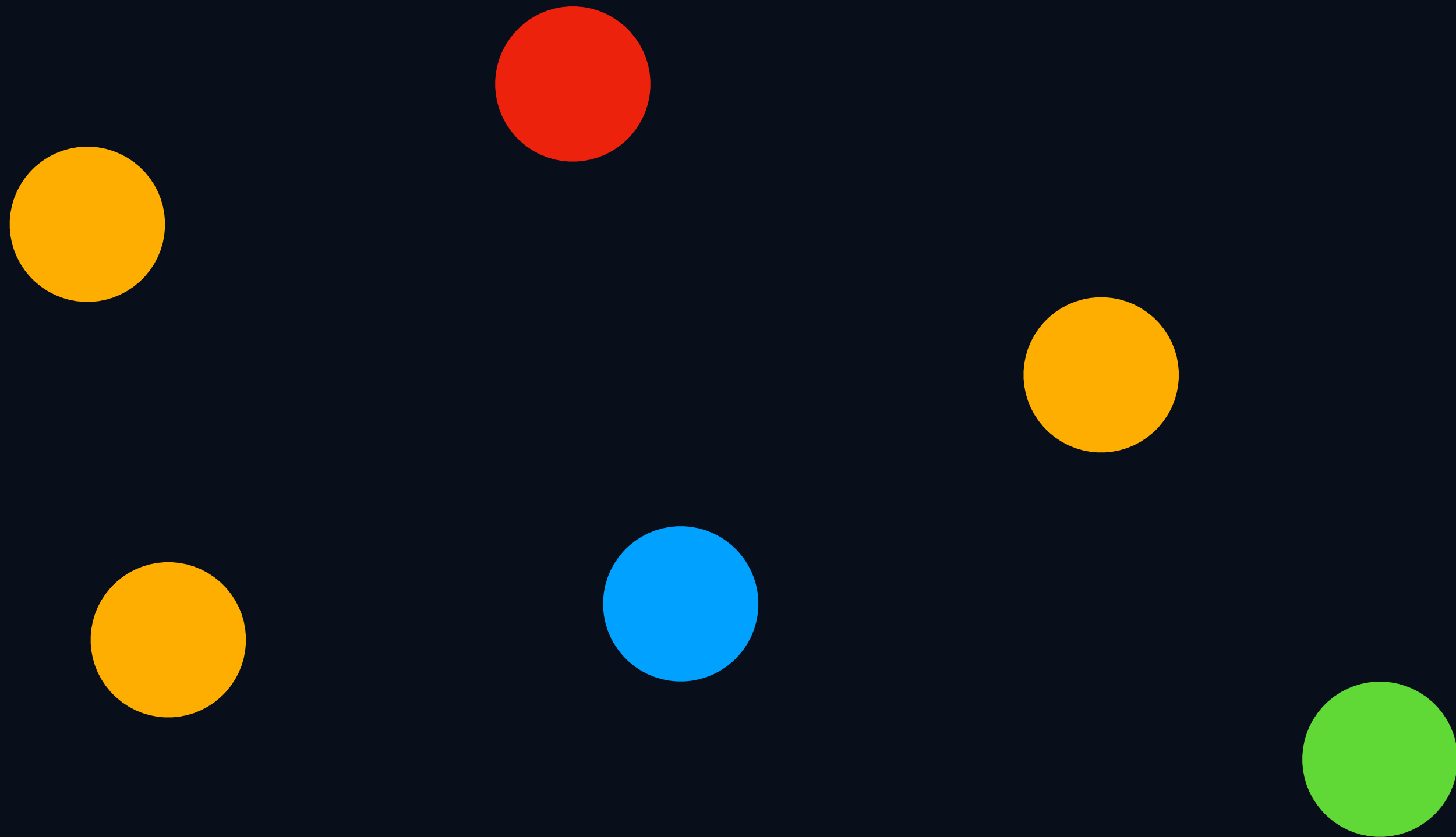
SALES GALLERY: 36 EAST 22ND STREET
160E22.COM • 212 388 9194



Implementacja w izolacji



Reimplementacja





DESIGN SYSTEM

**BIBLIOTEKA
KOMPONENTÓW**

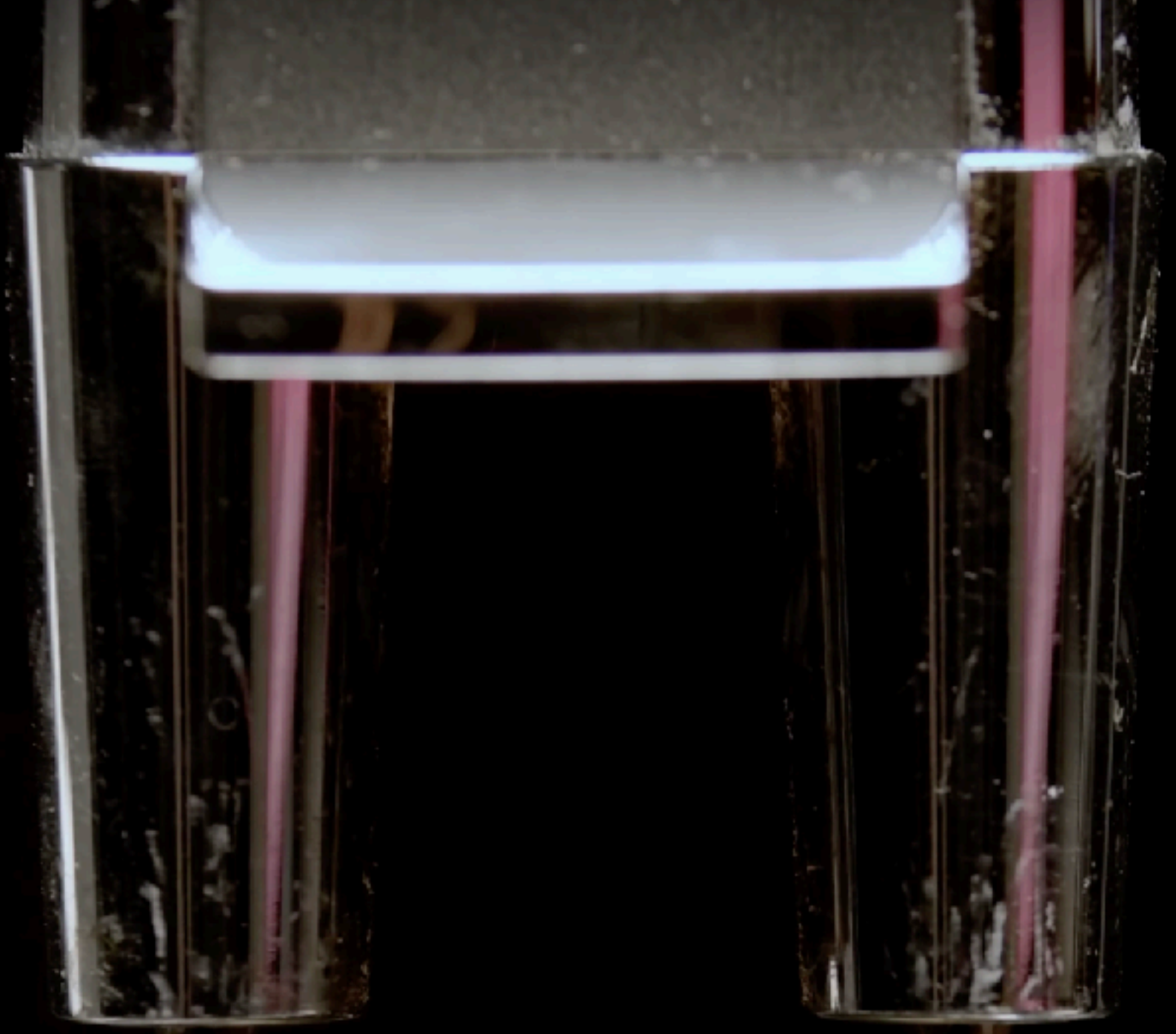
TOKENY

PROCESY

DOKUMENTACJA





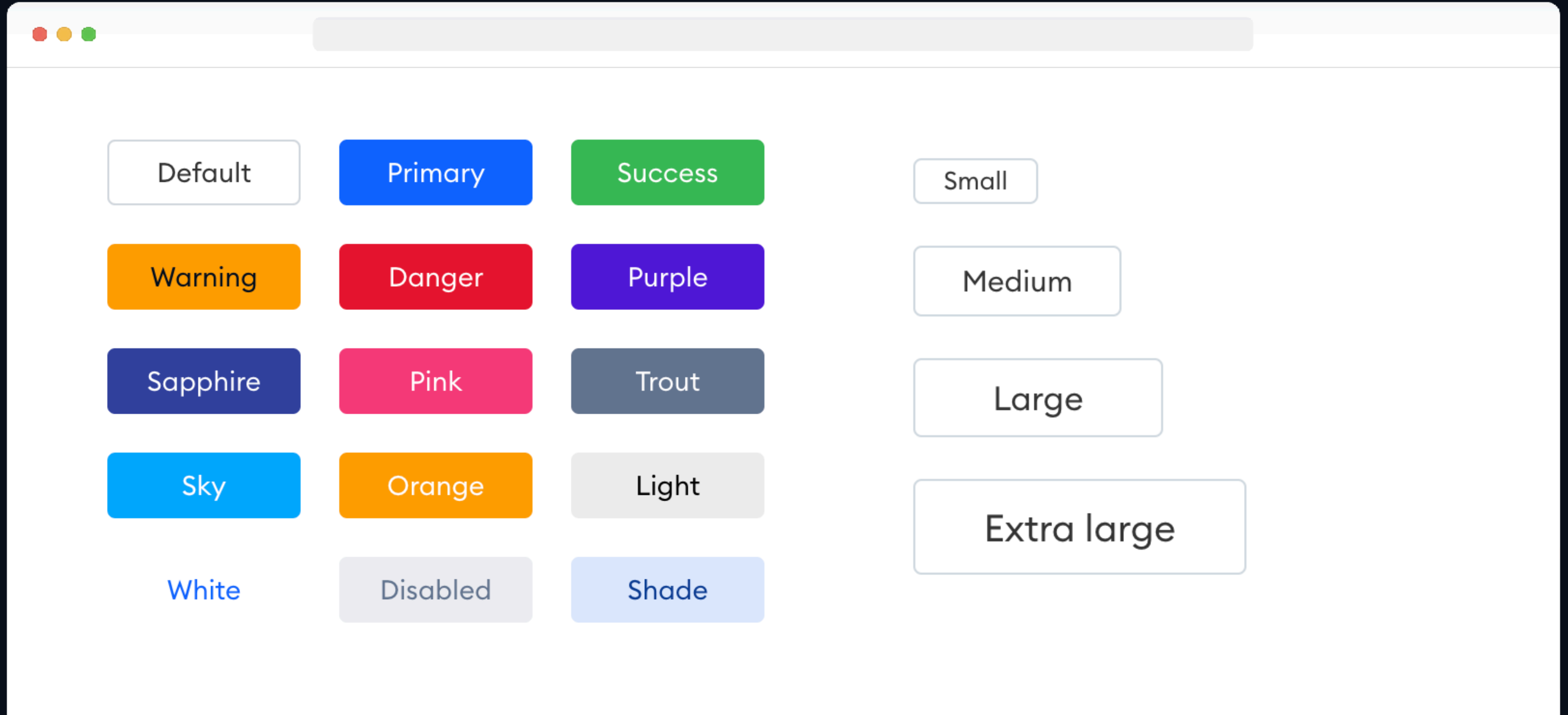


Buttony





Warianty i rozmiary



Kod przed & po

```
HTML [Settings] [Dropdown]  
1 <button type="button" className="btn btn-outline"
```

```
HTML [Settings] [Dropdown]  
1 <Button outline
```

Kod przed & po

```
HTML
1 <button className="btn btn-warning btn-xs"

* CSS
1 .btn-warning {
2   background-color: #f0ad4e;
3   border-color: #f0ad4e;
4   color: #080f1a;
5   &:hover {
6     background-color: darken(#f0ad4e, 5%);
7     border-color: darken(#f0ad4e, 5%);
8     color: #080f1a;
9   }
10 }
11
```

```
HTML
1 <Button type="warning" size="s"
```

Kod przed & po

```
HTML
1 <button type="button" className="btn btn-primary">
2   <IconPlus />
3 </button>
4 <button type="button" className="btn btn-outline"/>

CSS
1 .btn {
2   display: flex;
3   align-items: center;
4   justify-content: center;
5   letter-spacing: 0.3px;
6   margin-left: 20px;
7   line-height: 22px;
8   min-width: 120px;
9   &.btn-outline {
10    color: #080f1a;
11    border-color: #d1d9e0;
12    background-color: transparent;
13  }
14  &.btn-primary {
15    background-color: #3f88f6;
16    width: 170px;
17    &:hover {
18      background-color: #2d78e9;
19    }
20  }
21  svg {
22    width: 20px;
23    height: 20px;
24    margin-right: 4px;
25  }
26 }
```

```
HTML
1 <Button type="primary" icon={iconPlus} />
2 <Button outline />

CSS
1 .btn:first-of-type {
2   margin-left: 12px;
3 }
4
```

Dodatkowo



:hover

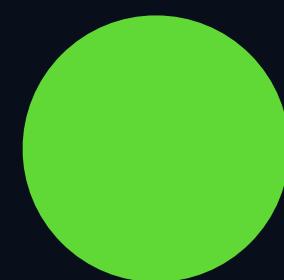
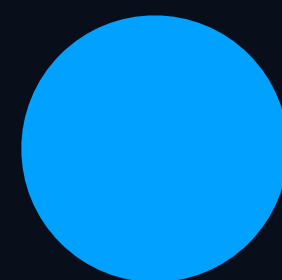
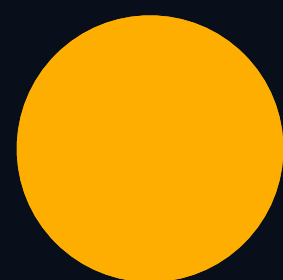
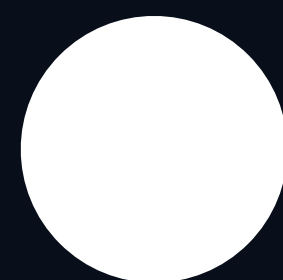


:active



:focus

Reużywalność







Typografia

Inwentaryzacja

8	16	24	32	44
9	17	25	34	46
10	18	26	36	48
11	19	27	37	50
12	20	28	38	52
13	21	29	40	54
14	22	30	42	56
15	23	31	43	64

Standaryzacja

8	16	24	32		
9					
	18				
11					
12	20	28		52	
			40		
14					

Headers

Używane dla nagłówków i tytułów



H 52 Header

H 40 Header

H 32 Header

H 28 Header

H 24 Header

H 20 Header

H 18 Header

H 16 Header

UI Text

Używany w UI



UI 18 Paragraph

UI 16 Paragraph

UI 14 Paragraph

UI 12 Paragraph

UI 11 Paragraph

UI 9 Paragraph

UI 8 Paragraph

Obrazki i tyle?

TypeScript

```
1  type HeadingTags = 'h1' | 'h2' | 'h3' | 'h4' | 'h5' | 'h6' | 'header';
2  type HeadingSizes = 'H16' | 'H18' | 'H20' | 'H24' | 'H28' | 'H32' | 'H40' | 'H52';
3
4  type TextTags = 'p' | 'span' | 'div';
5  type TextSizes = 'UI8' | 'UI9' | 'UI11' | 'UI12' | 'UI14' | 'UI16' | 'UI18';
6
7  export interface BaseProps<T, U> {
8      as?: T;
9      size?: U;
10 }
11
12 export type HeadingType = BaseProps<HeadingTags, HeadingSizes>;
13 export type TextType = BaseProps<TextTags, TextSizes>;
14
15
```

Obrazki i tyle?

TypeScript

```
return (  
  <ContextNavList>  
    <ContextNavHeader>  
      <Heading size="" weight="medium">  
        {trans('ti  
      </Heading>  
    </ContextNavHeader>  
    <InlineSVG src={en  
  </ContextNavList>  
);
```

↑ H20
↓ C24
↓ C40
H16
H18
H24
H28
H32
H40
H52

Press ↵ to insert, → to replace





Input

Schemat



identyfikacja

standaryzacja

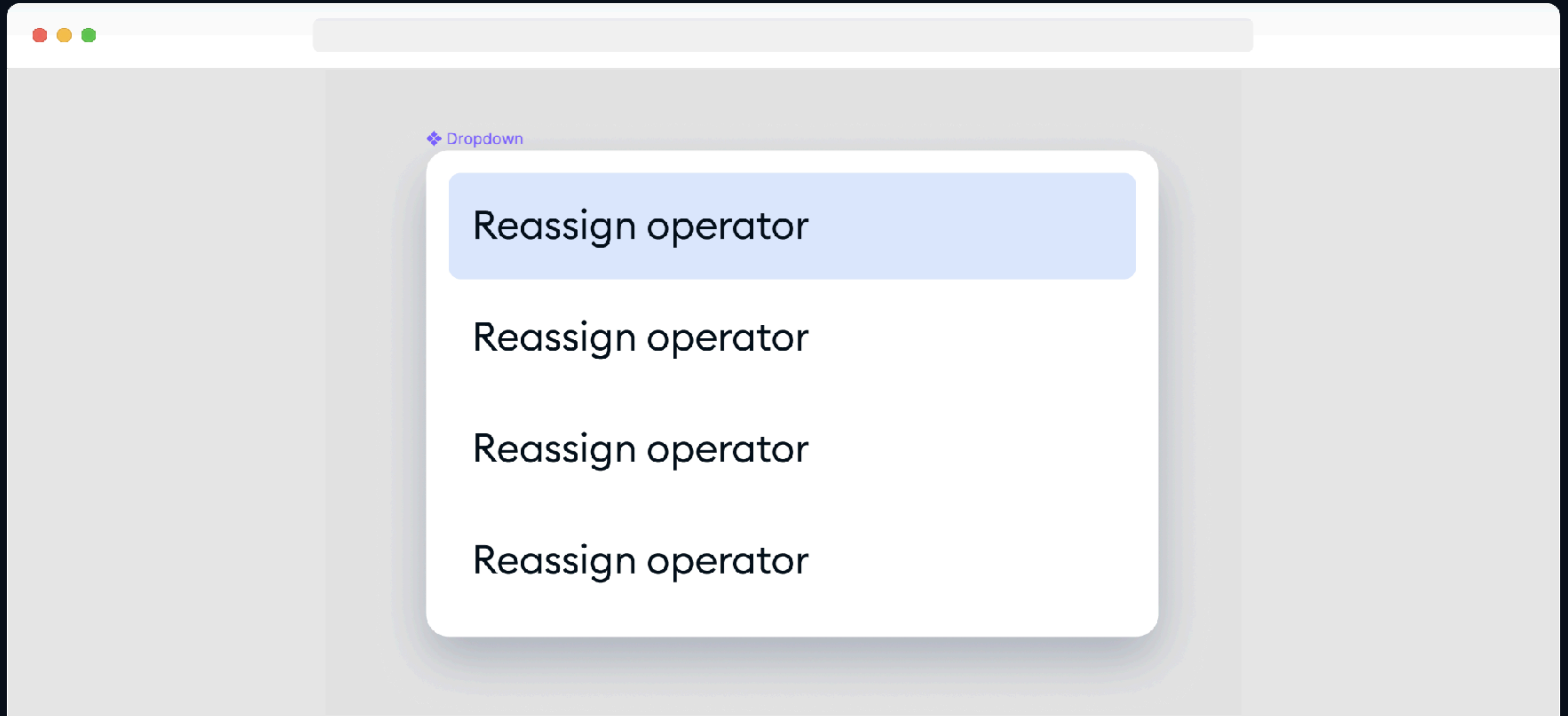
użycie

Spójność

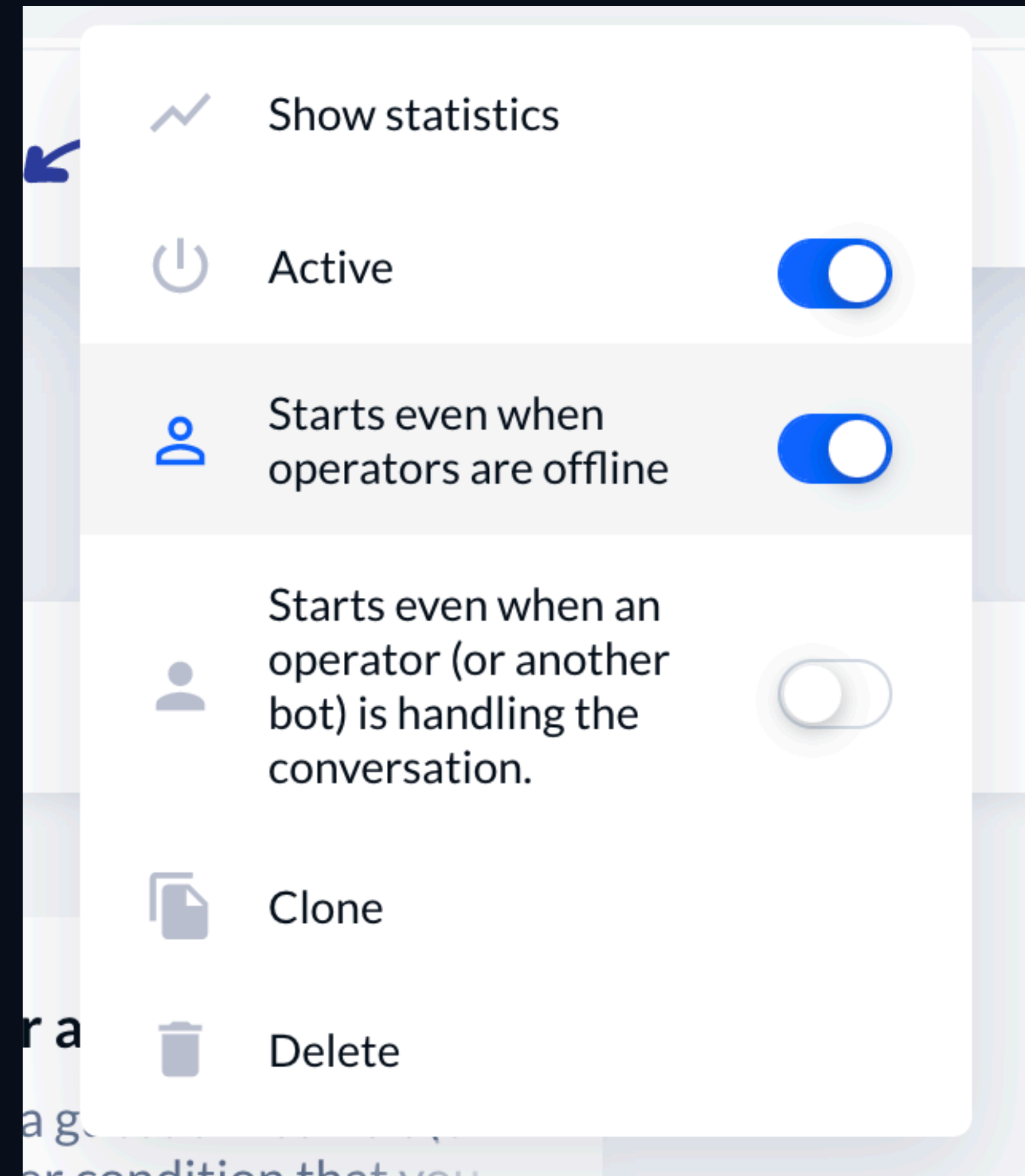
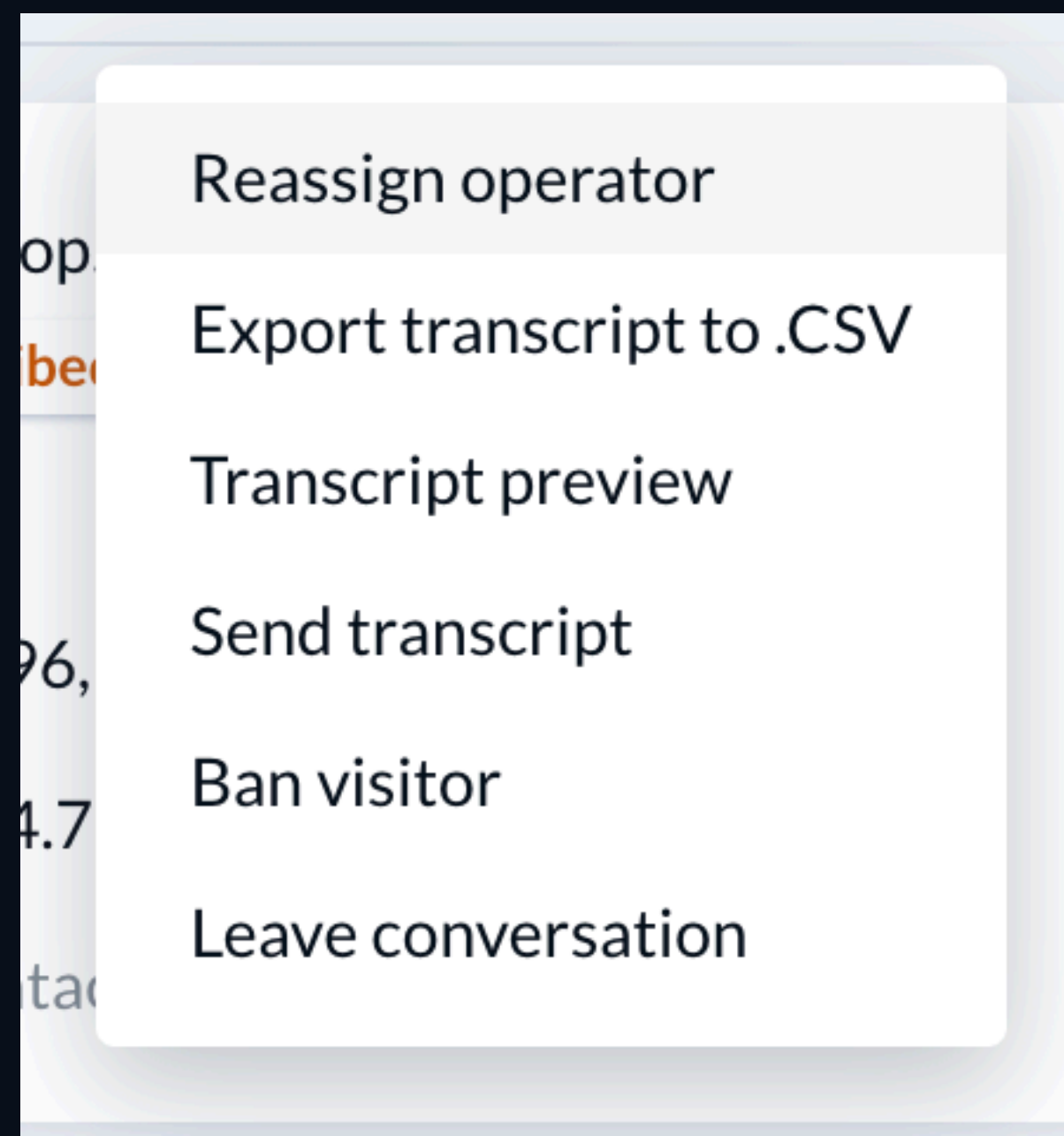
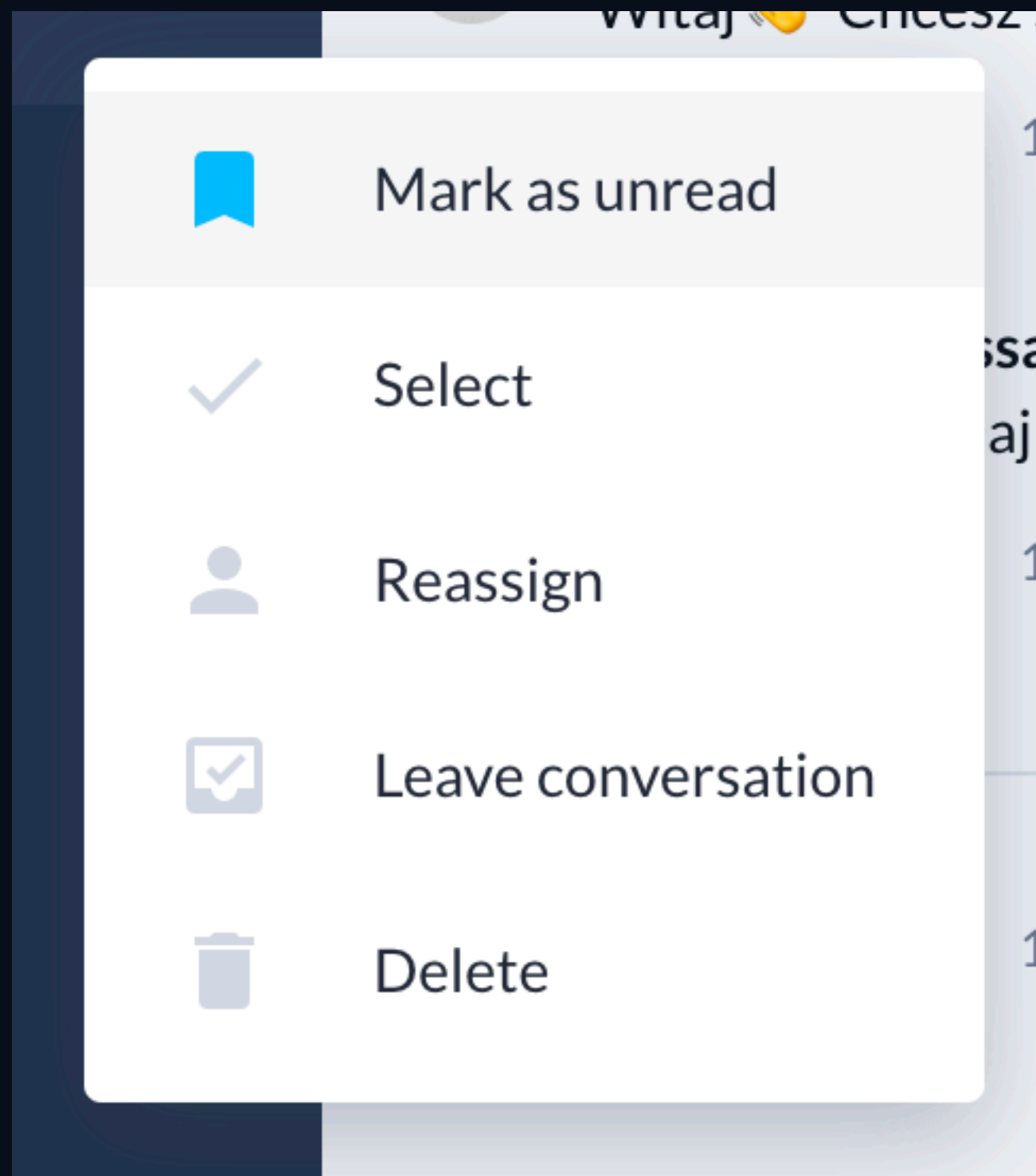


Spójność wizualna

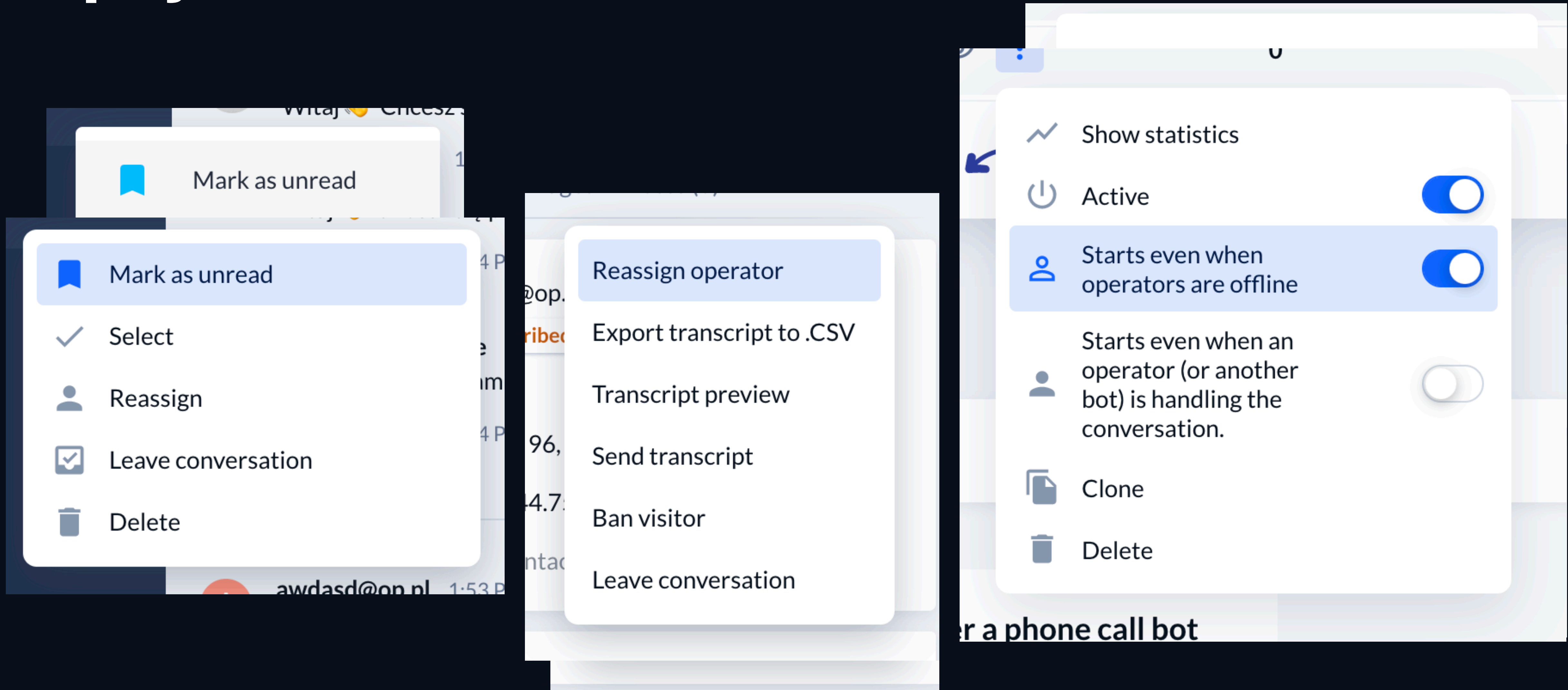
Figma



Spójność wizualna



Spójność wizualna



for a phone call bot

Ale

to tylko style

```
.tippy-box[data-theme~='dropdown'] {
  background-color: @white;
  border-radius: 4px;
  box-shadow: 0px 8px 24px 0 @oxford-blue32;
  color: #23bcf7;
  border-radius: 8px;
  box-shadow: 0 8px 24px 0 @oxford-blue32;
  padding-bottom: 8px;
  padding-top: 8px;
  border-radius: 8px;
  box-shadow: 0 8px 24px 0 @oxford-blue32;
  padding: 8px;
  margin-bottom: 10px;
}










.avatar {
  fill: @gray-200;
  margin-left: 10px;
  margin-right: 23px;
  width: 24px;
}

.dropdown__content {
  background-color: @white;
  border-radius: 4px;
  box-shadow: 0px 8px 24px 0 @oxford-blue32;
  padding-bottom: 8px;
  padding-top: 8px;
  border-radius: 8px;
  box-shadow: 0 8px 24px 0 @oxford-blue32;
  padding: 8px;
  margin-bottom: 10px;
}

&Text button,
&Image button {
  min-width: 90px;
  padding-left: 32px;
  min-width: 110px;
  padding-left: 38px;
  margin: 0;
  text-align: left;
  font-size: @font-size-l;
  line-height: 22px;
  font-size: @font-size-m;
  line-height: @line-height-m;
  border: 0;
  outline: none;
  display: flex;
  align-items: center;
  min-height: 36px;
  border-radius: 4px;
  &:hover {
    cursor: pointer;
    color: #23bcf7;
    background-color: @primary-blue-100;
    color: @primary-blue-900;
  }
}
```

Spójność w kodzie

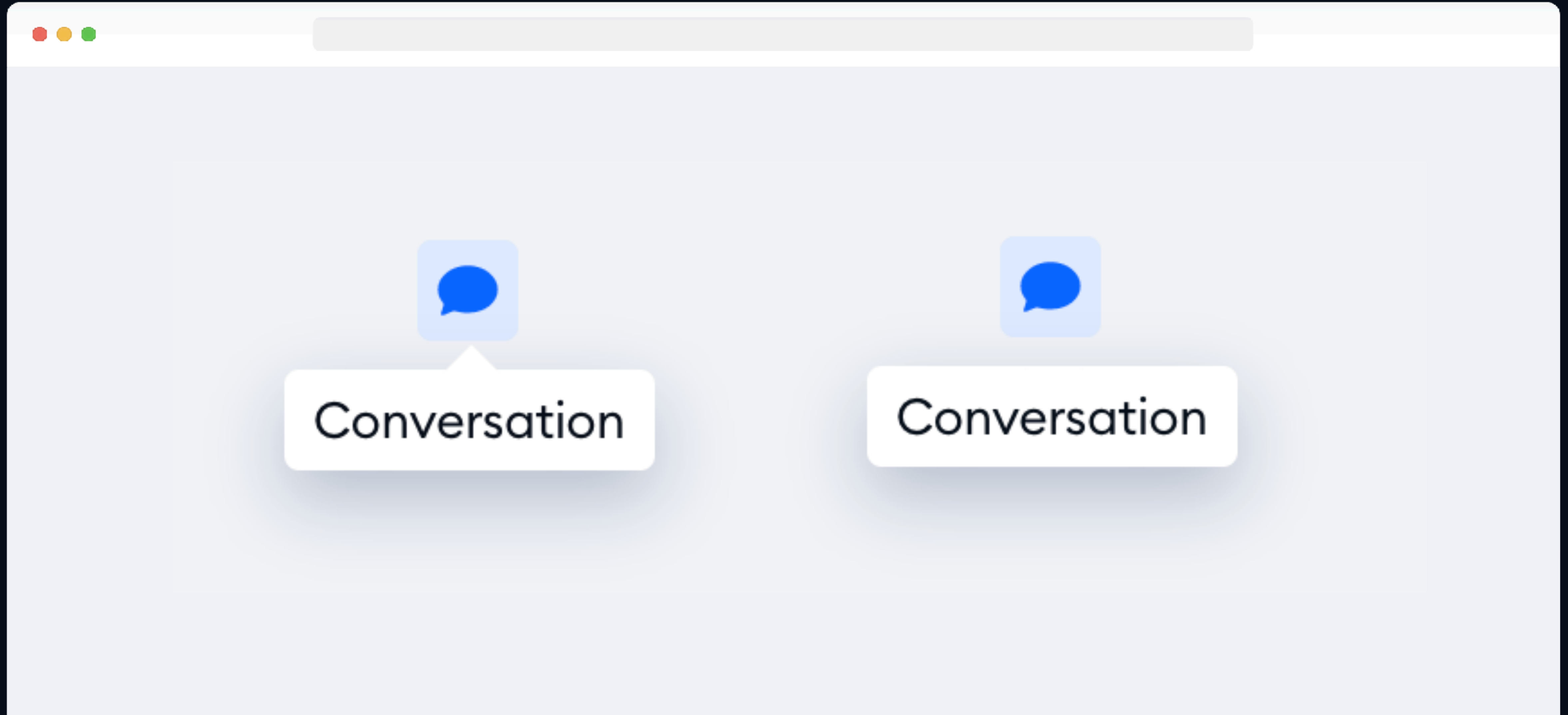
```
<DropDownList>
  {columns.entrySeq().map(
    ([key, column]: ReactElement => (
      <DropDownItem key={key}>
        <DropDownButton
          onClick={() => toggleColumn(key)}
          disabled={column.get('disabled')}
        >
          <DropDownIcon>{getIconForKey(key)}</DropDownIcon>
          <DropDownText>{getLabelForColumn(column, key)}</DropDownText>
          <Checkbox
            checked={column.get('visible', false)}
            disabled={column.get('disabled')}
            onChange={() => toggleColumn(key)}
          />
        </DropDownButton>
      </DropDownItem>
    )
  )
}</DropDownList>
```

	ID	<input type="checkbox"/>
	Name	<input checked="" type="checkbox"/>
	Email	<input checked="" type="checkbox"/>
	Email consent	<input checked="" type="checkbox"/>
	Created at	<input type="checkbox"/>
	Conversation rating	<input checked="" type="checkbox"/>
	Browser language	<input type="checkbox"/>
	City	<input type="checkbox"/>
	Country	<input checked="" type="checkbox"/>



Tooltipy

zostanie tylko jeden wariant



Tooltipy

domyślne wartości przyjazne developerom



```
<Tippy title={tooltip}>{label}</Tippy>  
<Tippy title={tooltip} arrow={false}>  
  {label}  
</Tippy>  
<Tippy title={tooltip} arrow={false}>  
  {label}  
</Tippy>
```



```
<Tippy title={tooltip} arrow>{label}</Tippy>  
<Tippy title={tooltip}>  
  {label}  
</Tippy>  
<Tippy title={tooltip}>  
  {label}  
</Tippy>
```



IconButton

szaleniestwo



```
<IconButton  
  onClick={handleClick}  
  icon={iconElement}  
  title={trans('visitor.quickResponses')}  
>
```

IconButton

szaleństwo



```
<IconButton
  onClick={handleClick}
  icon={iconElement}
  title={trans('visitor.quickResponses')}
  color={colors.GRAY_400}
```

IconButton

szaleństwo



```
<IconButton
  onClick={handleClick}
  icon={iconElement}
  title={trans('visitor.quickResponses')}
  color={colors.GRAY_400}
  hoverColor={colors.PRIMARY_BLUE_500}
```

IconButton

szaleństwo



```
<IconButton
  onClick={handleClick}
  icon={iconElement}
  title={trans('visitor.quickResponses')}
  color={colors.GRAY_400}
  hoverColor={colors.PRIMARY_BLUE_500}
  hoverBackgroundColor={colors.PRIMARY_BLUE_100}
```

IconButton

szaleństwo



```
<IconButton
  onClick={handleClick}
  icon={iconElement}
  title={trans('visitor.quickResponses')}
  color={colors.GRAY_400}
  hoverColor={colors.PRIMARY_BLUE_500}
  hoverBackgroundColor={colors.PRIMARY_BLUE_100}
  width={30}
  height={30}
```

IconButton

szaleństwo

```
<IconButton
  onClick={handleClick}
  icon={iconElement}
  title={trans('visitor.quickResponses')}
  color={colors.GRAY_400}
  hoverColor={colors.PRIMARY_BLUE_500}
  hoverBackgroundColor={colors.PRIMARY_BLUE_100}
  width={30}
  height={30}
  radius={4}
```

IconButton

szaleństwo

```
<IconButton
  onClick={handleClick}
  icon={iconElement}
  title={trans('visitor.quickResponses')}
  color={colors.GRAY_400}
  hoverColor={colors.PRIMARY_BLUE_500}
  hoverBackgroundColor={colors.PRIMARY_BLUE_100}
  width={30}
  height={30}
  radius={4}
  iconStyle={{
    width: 22,
    height: 22,
  }}
/>
```

IconButton

szaleństwo

```
<IconButton
  onClick={handleClick}
  icon={iconElement}
  title={trans('visitor.quickResponses')}
  color={colors.GRAY_400}
  hoverColor={colors.PRIMARY_BLUE_500}
  hoverBackgroundColor={colors.PRIMARY_BLUE_100}
  width={30}
  height={30}
  radius={4}
  iconStyle={{
    width: 22,
    height: 22,
  }}
  titleOffset={[0, 8]}
```

IconButton

szaleniestwo



```
<IconButton  
  onClick={handleClick}  
  icon={iconElement}  
  title={trans('visitor.quickResponses')}>
```

IconButton

szaleństwo



```
<IconButton
  onClick={handleClick}
  icon={iconElement}
  title={trans('visitor.quickResponses')}
  type={IconButtonTypes.MUTED}
  size="l"
```

















Meetup

TDO



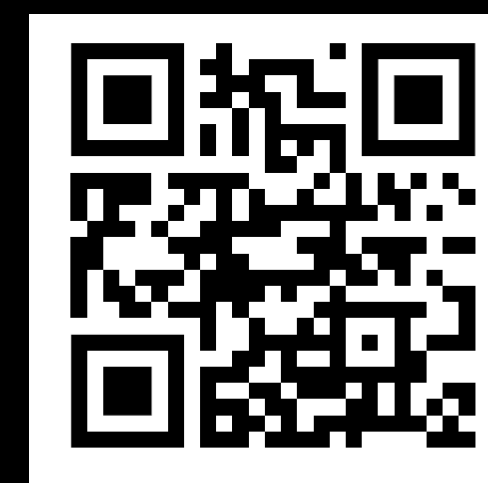
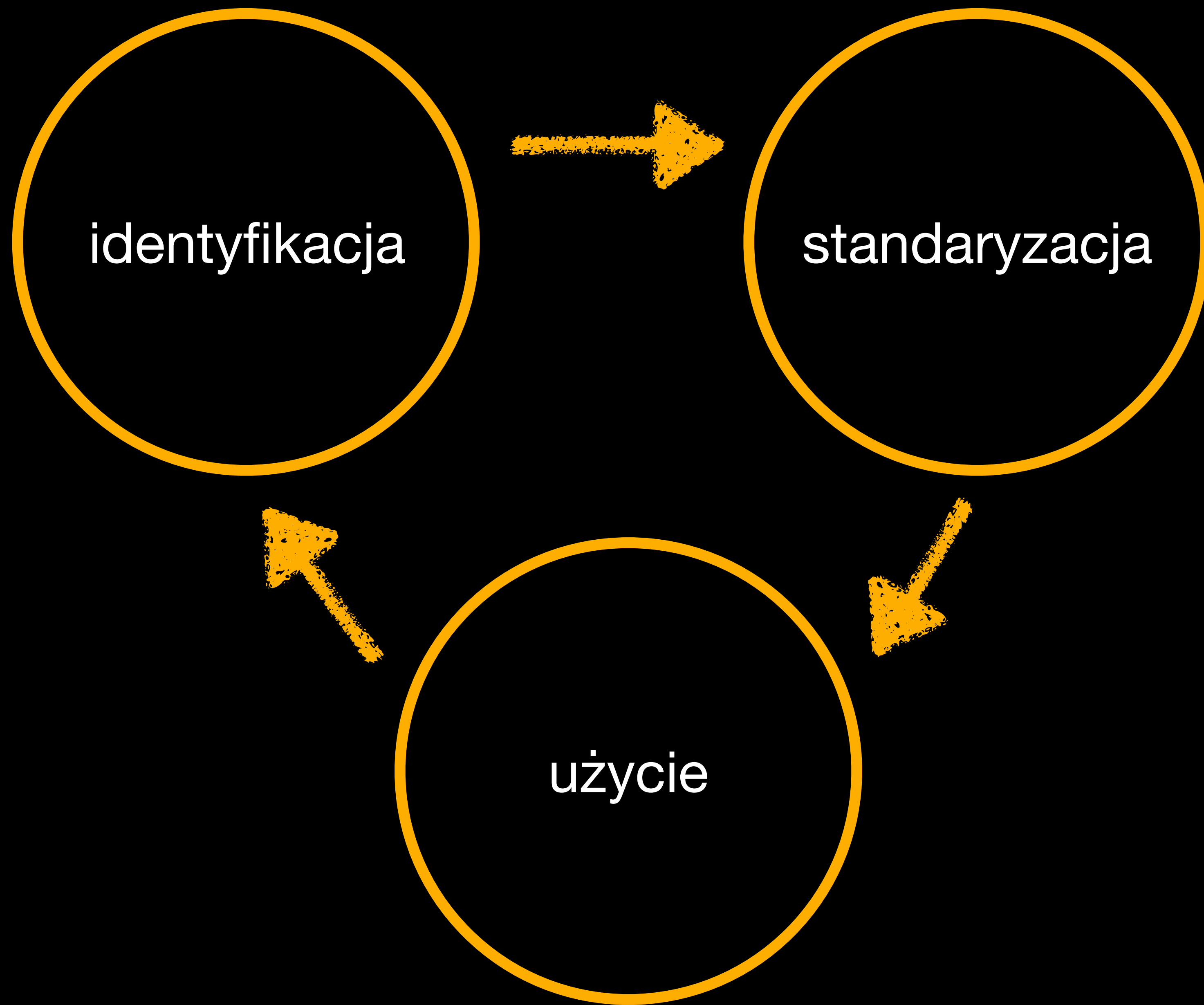
System







👉 careers.tidio.com

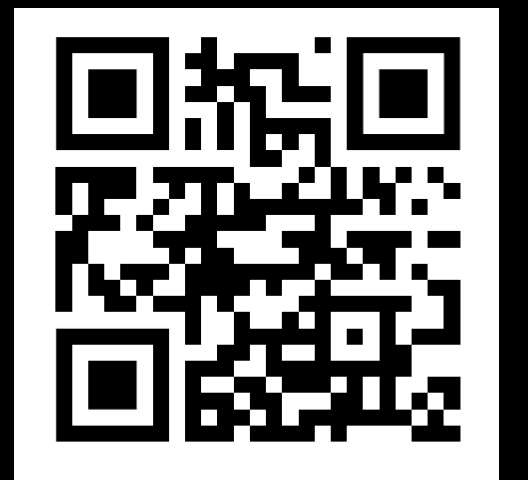


Dziękuję za uwagę!
Zapraszam do pytań 🖐️

Piotr Wcisłek

[linkedin.com/in/piotr-wcislek/](https://www.linkedin.com/in/piotr-wcislek/)

piotr.w@tidio.net



👉 careers.tidio.com