

# Jonny Jackson CSS Component Library

## Nav bar



WorldRecipeace.com

About us

Services

Pricing

Contact

### HTML

```
<!--icon-->
<link rel="stylesheet" href="https://fonts.googleapis.com/css2?family=Material+Symbols+Outlined:opsz,wght,FILL,GRAD@24,400,0,0" />
<!--font-->
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Noto+Sans:ital,wght@0,100..900;1,100..900&family=Roboto:ital,wght@0,100;0,300;0,400;0,500;0,700;0,900" rel="stylesheet">

<div class="myNavBar">
<span class="material-symbols-outlined menu-icon">
  menu
</span>
<a href="" class="companyName">WorldRecipeace.com</a>
<ul>
  <li><a href="">About us</a></li>
  <li><a href="">Services</a></li>
  <li><a href="">Pricing</a></li>
  <li><a href="">Contact</a></li>
</ul>
</div>
```


### CSS

```
.myNavBar {
  display: flex;
  flex-direction: row;
  align-items: center;
  background-color: royalblue;
  border-bottom: 1px solid gray;
  width: 100%;
  top: 0;
  left: 0;
  position: fixed;
  z-index: 1000; /*on top of everything*/
}
.menu-icon {
  margin-right: 10px;
  margin-left: 10px;
  color: white;
}
.myNavBar > a {
  color: white;
  text-decoration: none;
  transition: all .3s ease;
  font-family: "Roboto", sans-serif;
  font-weight: 400;
}
.myNavBar > a: hover {
  color: lightskyblue;
  cursor: pointer;
}
.myNavBar > ul > li {
  display: inline;
  margin-right: 10px;
  margin-left: 10px;
}
.myNavBar > ul > li > a {
  color: white;
  text-decoration: none;
  transition: all .3s ease;
  font-family: "Roboto", sans-serif;
  font-weight: 400;
}
.myNavBar > ul > li > a: hover {
  color: lightskyblue;
  cursor: pointer;
  text-decoration: none;
}
.myNavBar > ul {
  margin-left: auto;
}
```

### Javascript

```
document.querySelector(".material-symbols-outlined").addEventListener('click',()=> {
  console.log("menu button clicked");
});
```

## Nav drawer



### HTML

```
<div class="myDrawer">
<div class="myNavBar">
  <span id="menuOpen" class="material-symbols-outlined menu-icon">
    menu
  </span>
</div>
<div class="popup ">
  <ul>
    <li><a href="">About us</a></li>
    <li><a href="">Services</a></li>
    <li><a href="">Pricing</a></li>
    <li><a href="">Contact</a></li>
  </ul>
</div>
</div>
```

CSS

```
.myDrawer {
  border:1px solid black;
}

.myNavBar {
  display:flex;
  flex-direction: row;
  align-items: center;
  background-color: royalblue;
  border-bottom: 1px solid gray;
  width:100%;
  z-index: 1000; /*on top of everything*/
}

.menu-icon {
  margin-right: 10px;
  margin-left:10px;
  color:white;
}
.menu-icon:hover {
  margin-right: 10px;
  margin-left:10px;
  color:lightblue;
}

.popup {
  background-color: cornflowerblue;
  border-right:1px solid gray;
  padding:10px;
  transition: opacity .3s ease;
  opacity:0;
  width:20%;
  z-index: 1000; /*on top of everything*/
}
.showing {
  opacity:1;
}
.popup > ul > li > a {
  color:white;
  text-decoration: none;
  transition:all .3s ease;
  font-family: "Roboto", sans-serif;
  font-weight: 400;
}
.popup > ul {
  list-style-type:none;
}
.popup > ul > li {
  margin-top:20px;
}
.popup > ul > li > a:hover {
  color:lightblue;
  cursor: pointer;
  text-decoration: none;
}
}
```

Javascript

```
document.getElementById('menuOpen').addEventListener('click',() => {
  const popup = document.querySelector('.popup');
  popup.classList.toggle('showing');

});
```

Loading spinners



HTML

```
<div id="container-one">
<div class="circle" id="circle-one"></div>
<div class="circle" id="circle-two"></div>
<div class="circle" id="circle-three"></div>
</div>
```

CSS

```
.circle {
  width:50px;
  height:50px;
  background-color: chartreuse;
  border-radius: 50%;
  border:none;
  margin:20px;
}
#container-one {
```

```
        display: flex;
        flex-direction: row;
    }

    @keyframes bounce {
        0%, 100% {
            transform: translateY(0);
        }
        50% {
            transform: translateY(-50px);
        }
    }

    #circle-one {
        animation: bounce 3s ease 0s infinite;
    }
    #circle-two {
        animation: bounce 3s ease .5s infinite;
    }
    #circle-three {
        animation: bounce 3s ease 1s infinite;
    }
}
```

Javascript

```
no javascript
```

Lo

HTML

```
<div>
<div class="loading-text">Loading...</div>
</div>
```

CSS

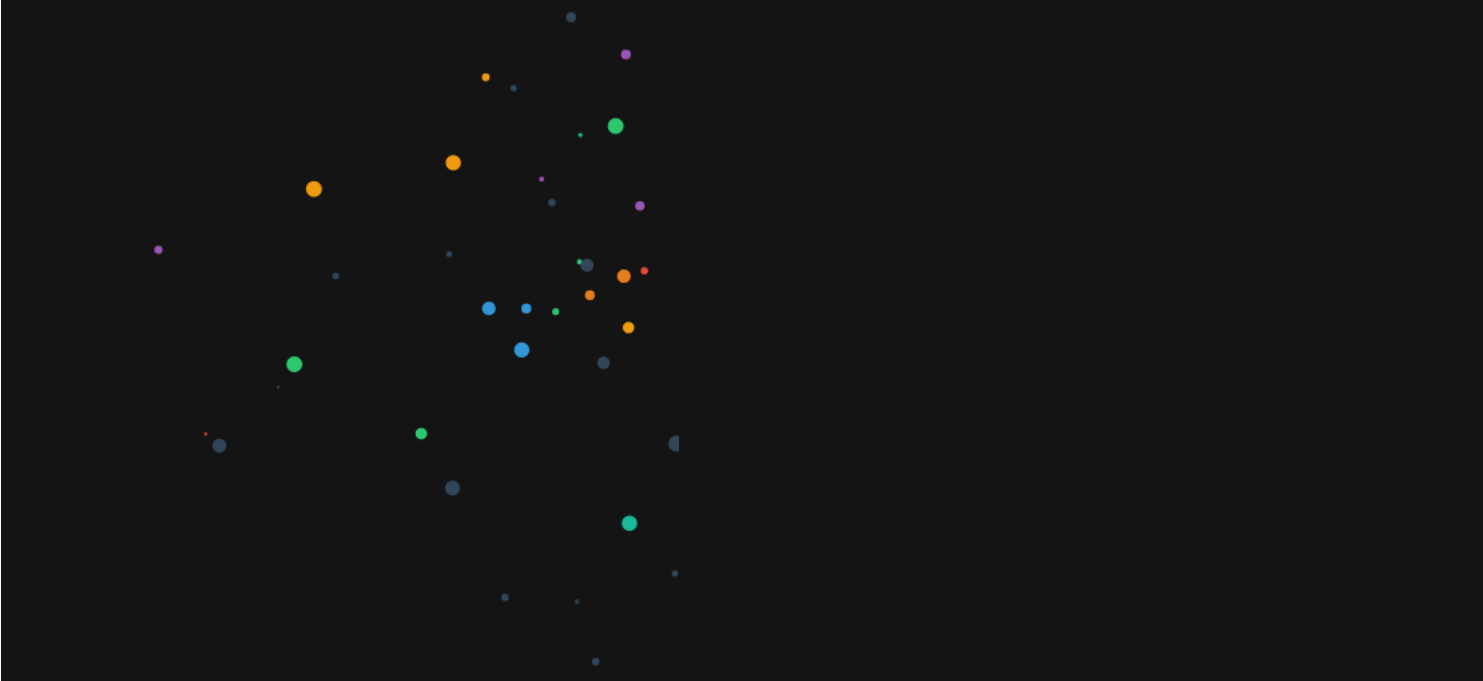
```
.loading-text {
    font-size: 24px;
    font-weight: bold;
    color: #333;
    animation: bounceOpacity 1.5s ease infinite;
}

@keyframes bounceOpacity {
    0%, 20%, 50%, 80%, 100% {
        opacity: 0.5;
        transform: translateY(0);
    }
    40% {
        opacity: 1;
        transform: translateY(-10px);
    }
    60% {
        opacity: 1;
        transform: translateY(-5px);
    }
}
```

Javascript

```
const loadingText = document.querySelector('.loading-text');
let text = '';
let currentIndex = 0;
const fullText = 'Loading...';

setInterval(() => {
    if (currentIndex <= fullText.length) {
        text = fullText.substring(0, currentIndex);
        currentIndex++;
    } else {
        text = '';
        currentIndex = 0;
    }
    loadingText.textContent = text;
}, 136);
```



HTML

```
<canvas id="loading-canvas" width="500px" height="500px" ></canvas>
```

CSS

```
no css
```

Javascript

```
const mouse = {
  x: undefined,
  y: undefined
};

const canvas = document.getElementById('loading-canvas');
const ctx = canvas.getContext('2d');
const particles = [];

class Particle {
  constructor(x, y, size,color) {
    this.x = x;
    this.y = y;
    this.size = size;
    this.color = color;
    this.baseX = this.x;
    this.baseY = this.y;
    this.density = (Math.random() * 30) + 1;
  }

  draw() {
    ctx.fillStyle = this.color;
    ctx.beginPath();
    ctx.arc(this.x, this.y, this.size, 0, Math.PI * 2);
    ctx.closePath();
    ctx.fill();
  }

  update() {
    const dx = mouse.x - this.x;
    const dy = mouse.y - this.y;
    const distance = Math.sqrt(dx * dx + dy * dy);
    const speed = 1;

    if (distance > 0) {
      const moveX = (dx / distance) * speed;
      const moveY = (dy / distance) * speed;

      this.x += moveX;
      this.y += moveY;
    }
  }
}

const init = () => {
  for (let i = 0; i < 1000; i++) {
    createNewParticle()
  }
};

function createNewParticle() {
  const size = (Math.random() * 5) + 1;
  const x = (Math.random() * (window.innerWidth - size * 2)) + size;
  const y = (Math.random() * (window.innerHeight - size * 2)) + size;
  particles.push(new Particle(x, y, size,getRandomColor()));
}

const animate = () => {
  ctx.clearRect(0, 0, canvas.width, canvas.height);

  for (const particle of particles) {
    particle.update();
    particle.draw();
  }

  createNewParticle();
  particles.shift(); //removes first element
  console.log(particles.length);
  requestAnimationFrame(animate);
};

init();
animate();

window.addEventListener('mousemove', (e) => {
  mouse.x = e.x;
  mouse.y = e.y;
});

function getRandomColor() {
  const colors = [
    "#3498db", // Blue
    "#e74c3c", // Red
    "#2ecc71", // Green
    "#f39c12", // Orange
    "#9b59b6", // Purple
    "#1abc9c", // Teal
    "#e67e22", // Brown
    "#34495e"  // Dark Gray
  ];

  const randomIndex = Math.floor(Math.random() * colors.length);
  return colors[randomIndex];
}
```

Floating Action button



HTML

```
<!--add button-->
<link rel="stylesheet" href="https://fonts.googleapis.com/css2?family=Material+Symbols+Outlined:opsz,wght,FILL,GRAD@24,400,0,0" />
<button id="fab">
<span class="material-symbols-outlined">add</span>
</button>
```

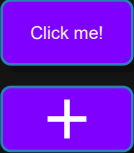
CSS

```
#fab {
  border-radius: 50%;
  width:75px;
  height:75px;
  border:none;
  background-color: #3498db;
  bottom:50px;
  right:50px;
  position:fixed;
  box-shadow: 5px 5px rgba(10,10,10,.5);
  color:white;
  cursor: pointer;
  transition: all 0.3s ease;
  display:flex;
  align-items: center; /* Center vertically */
  justify-content: center; /* Center horizontally */
}
#fab:hover {
  background-color: #2980b9;
  box-shadow: 8px 8px rgba(10, 10, 10, 0.3);
}
.material-symbols-outlined {
  font-size:50px !important;
}
```

Javascript

```
document.getElementById('fab').addEventListener('click',()=> {
  console.log("javascript works");
});
```

Button



HTML

```
<link rel="stylesheet" href="https://fonts.googleapis.com/css2?family=Material+Symbols+Outlined:opsz,wght,FILL,GRAD@24,400,0,0" />
<button id="myButton">Click me!</button>

<button id="myButton">
<span class="material-symbols-outlined">add</span>
</button>
```

CSS

```
#myButton {
  color: #fff;
  width:100px;
  height:50px;
  background-color: #8000ff;
  border-radius: 8px;
  border: 2px solid #2980b9;
  cursor: pointer;
  transition: color 0.3s ease, box-shadow 0.3s ease, background-color 1s ease;
  box-shadow: 5px 5px rgba(10,10,10,.5);
  display: flex;
  justify-content: center;
  align-items: center;
  margin:15px;
}

#myButton:hover {
  color: lightgoldenrodyellow;
  background-color: #3498db;
  box-shadow: 8px 8px rgba(10, 10, 10, 0.3);
}
```

Javascript

```
document.getElementById('myButton').addEventListener('click',()=> {
  console.log("javascript works");
});
```