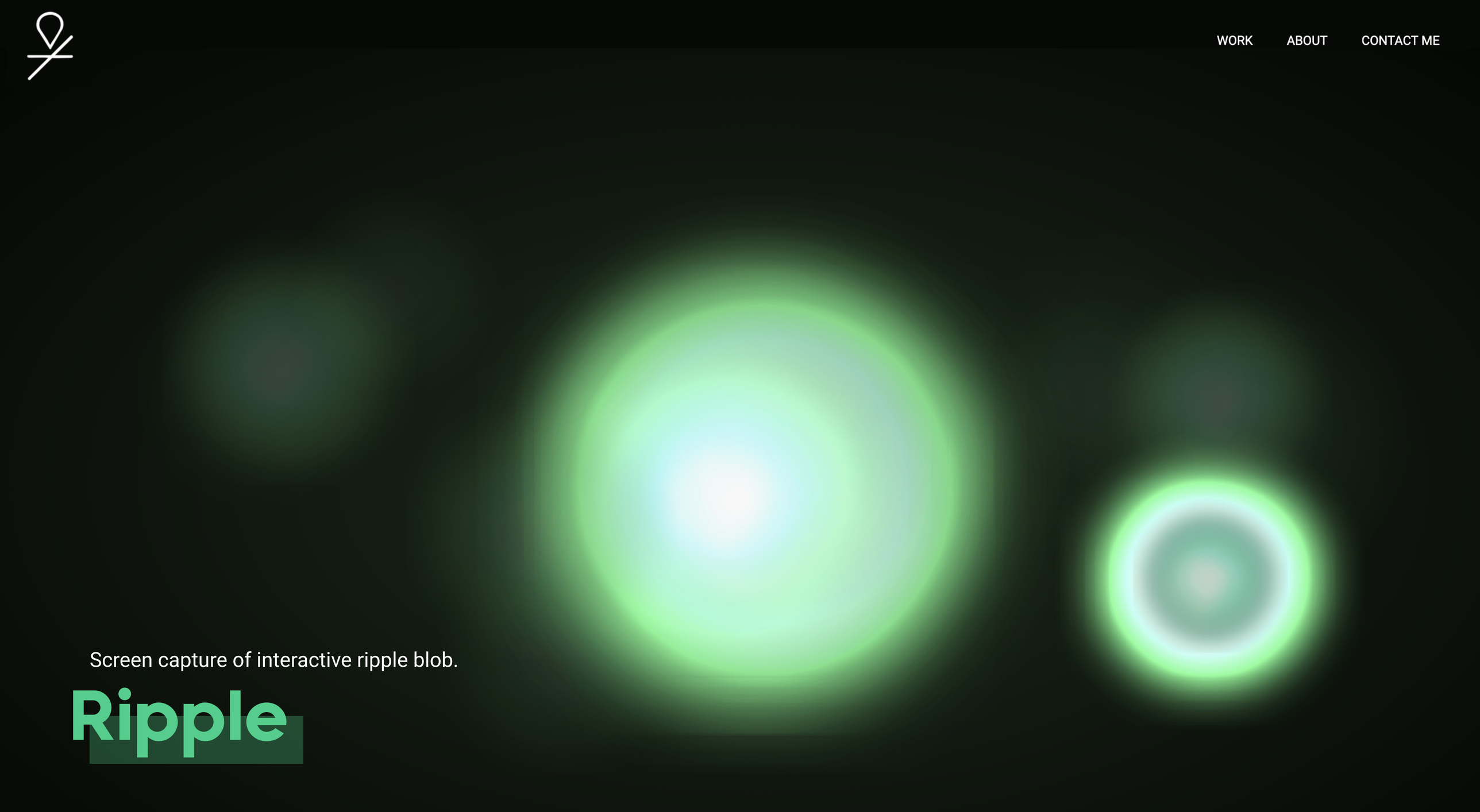
Blob Experiment

https://www.conniexu.com/blob.html

A series of web-based interactive blob experiments coded with JavaScript.

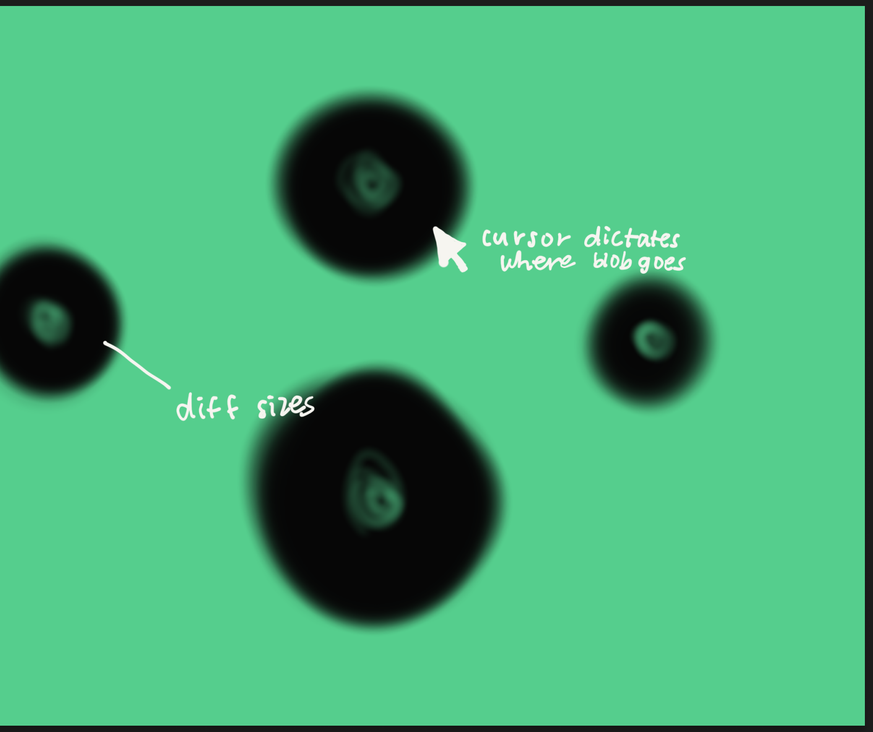
The following content are screenshots from the web pages that the experiment is housed on.

Ripple Growth Separation



Screen capture of interactive ripple blob.

Ripple



Ripple: Context

1. My design process started with **creating quick sketches** of what I wanted to create on **Procreate**.

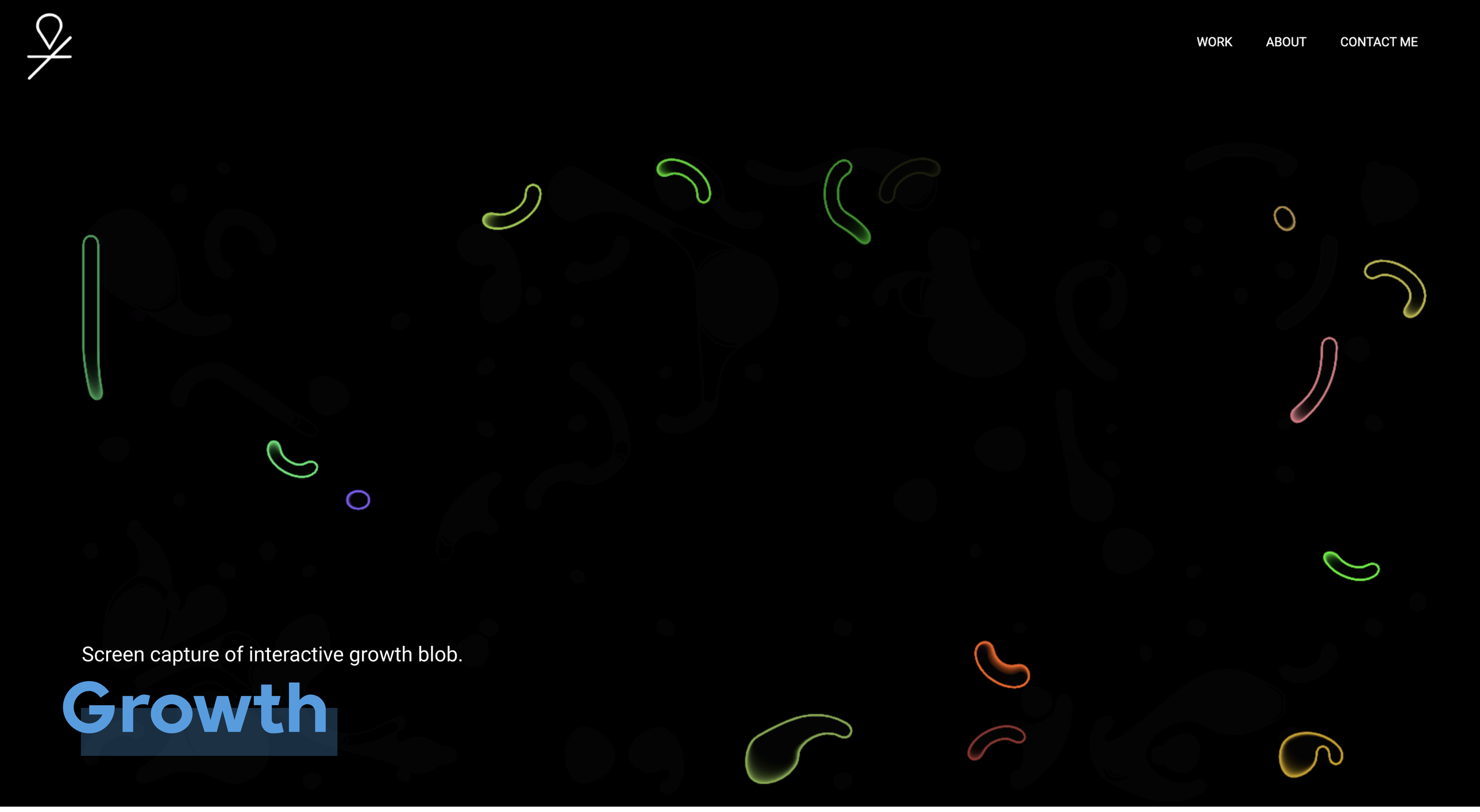
I drew inspiration from the core colors of my portfolio site which include this bold “Green Lantern” green.

2. I created the **glowing blob asset on Adobe Illustrator**.

These blobs will be different sizes and overlayed with one another when coded.

3. I used **JavaScript to code the web-based ripple effect**.

```
  var n = 100,
    speed = 10,
    startSize = rand(180,500);

var c = document.getElementById("c"),
    ctx = c.getContext("2d"),
    cw = (c.width = window.innerWidth),
    ch = (c.height = window.innerHeight),
    mousePos = {x:"", y:""},
    img = new Image(),
    particles = [],
    Particle = function(index) {
      this.index = index;
      this.dur = (100-rand(9, 90))/speed;
      this.draw = function() {
        ctx.translate( this.x, this.y );
        ctx.globalAlpha = this.alpha;
        ctx.globalCompositeOperation = 'lighter';
        if (index%2==0) ctx.globalCompositeOperation = 'xor';
        ctx.drawImage(img, -this.size/2, -this.size/2, this.size, this.size);
        ctx.translate( -this.x, -this.y );
      }
    };


c.onmousemove = function(e){ mousePos = { x:e.clientX, y:e.clientY }; }
document.onmouseleave = document.ontouchend = function(e){ mousePos = {x:"", y:""} }


function setParticle(p, firstRun) {
  var startProps = { x:cw/2+rand(0,60)-30, y:ch/2+rand(0,60)-30, size:startSize, alpha:0 };
  if (rand(0,1)>0.3 && mousePos.x!="") startProps = { x:mousePos.x, y:mousePos.y, size:startSiz
  var _tl = new TimelineMax().fromTo(p, p.dur, startProps, {
    size:'+='+String(rand(200,400)),
    bezier:[{alpha:rand(0.15,0.65)},{alpha:0}],
```



Screen capture of interactive growth blob.

Growth

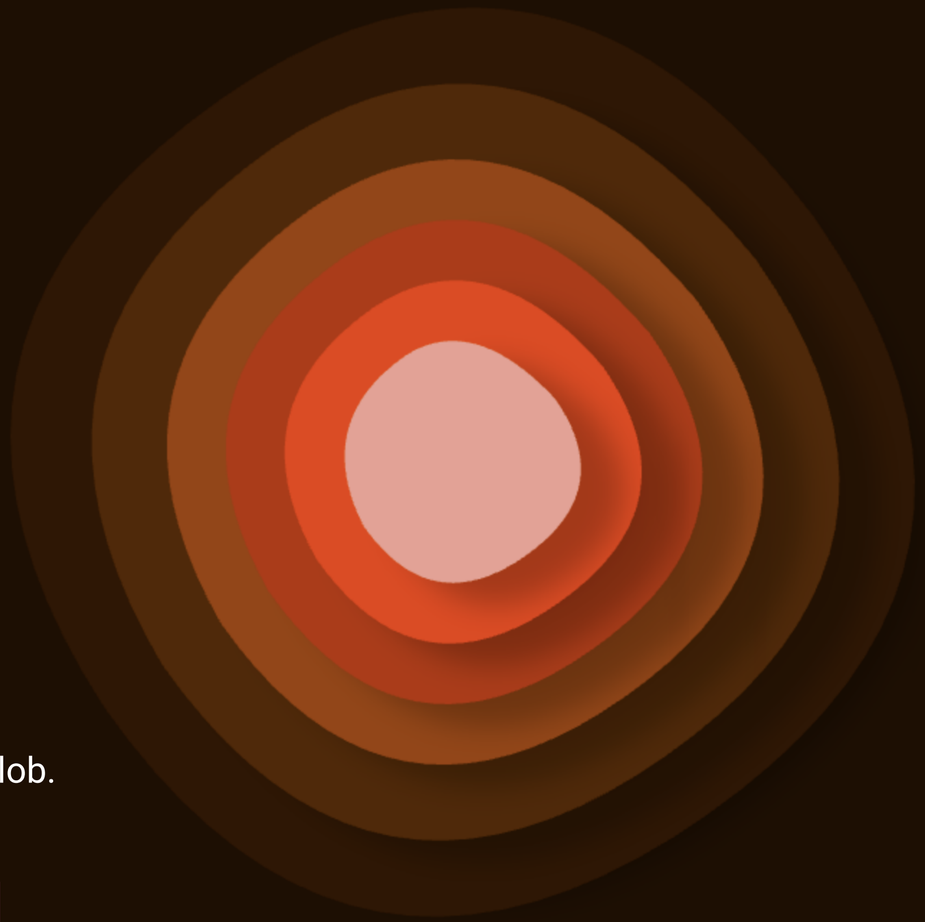
Growth: Context

1. My design process started when I was at my **research lab where we saw spirillum bacteria under a microscope**.

I wanted to recreate that distinct look of bacteria growing.

2. The entire project was created with **code in JavaScript**.

```
const mPI = Math.PI;
const mPIS2 = Math.PI / 2;
const m2PI = Math.PI * 2;
const msin = Math.sin;
const mcos = Math.cos;
const matan2 = Math.atan2;

const mhypot = Math.hypot;
const msqrt = Math.sqrt;

const rac3   = msqrt(3);
const rac3s2 = rac3 / 2;
const mPIS3 = Math.PI / 3;


function alea (min, max) {
  if (typeof max == 'undefined') return min * mrandom();
  return min + (max - min) * mrandom();
}



function intAlea (min, max) {
  if (typeof max == 'undefined') {
    max = min; min = 0;
  }
  return mfloor(min + (max - min) * mrandom());
```



Screen capture of interactive separation blob.

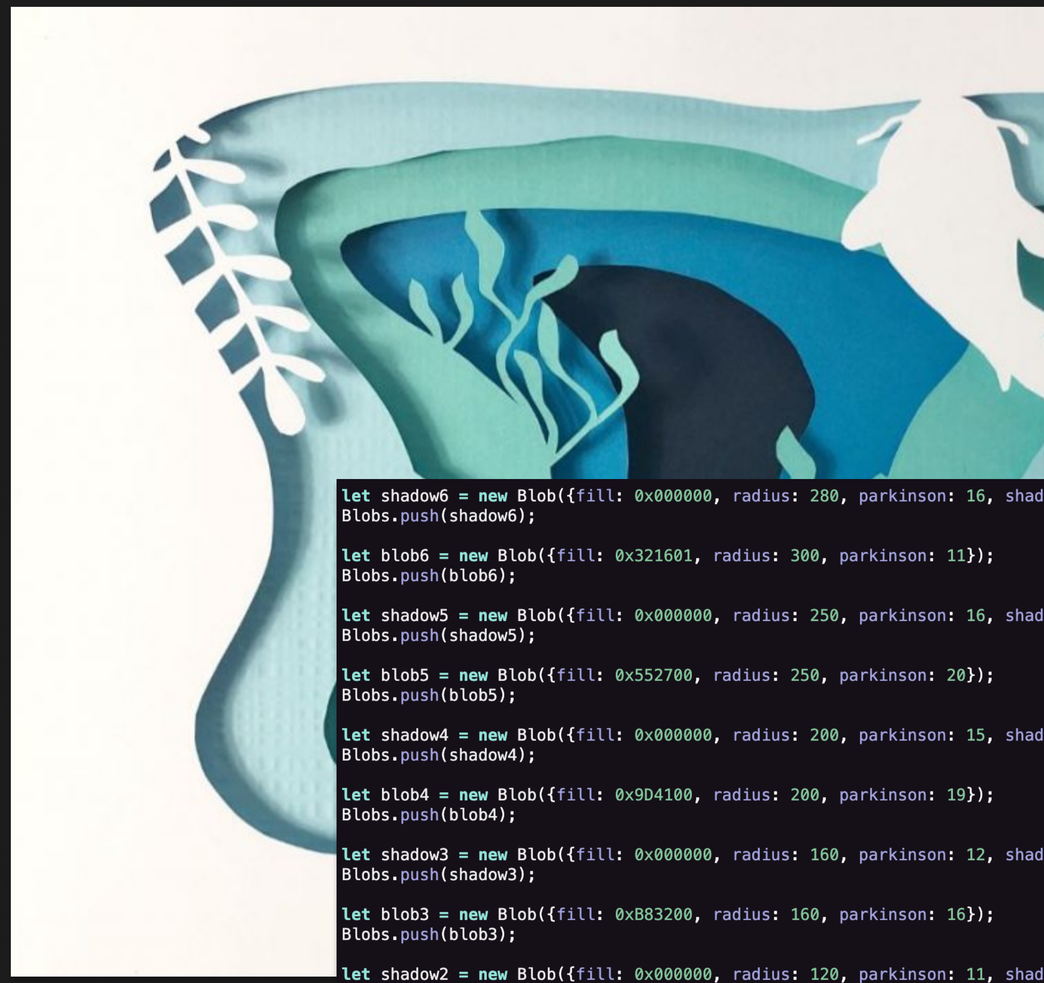
Separation

1. I drew inspiration from **paper layering art that I discovered when researching for a class project**.

I think there's something so beautiful t o the minimal monochromatic layered look.

I planned what I wanted my blob to look like on **Procreate** -- I mainly focused on the color palette.

2. The entire project was coded with **JavaScript**.



```
let shadow6 = new Blob({fill: 0x000000, radius: 280, parkinson: 16, shad
Blobs.push(shadow6);

let blob6 = new Blob({fill: 0x321601, radius: 300, parkinson: 11});
Blobs.push(blob6);

let shadow5 = new Blob({fill: 0x000000, radius: 250, parkinson: 16, shad
Blobs.push(shadow5);

let blob5 = new Blob({fill: 0x552700, radius: 250, parkinson: 20});
Blobs.push(blob5);

let shadow4 = new Blob({fill: 0x000000, radius: 200, parkinson: 15, shad
Blobs.push(shadow4);

let blob4 = new Blob({fill: 0x9D4100, radius: 200, parkinson: 19});
Blobs.push(blob4);

let shadow3 = new Blob({fill: 0x000000, radius: 160, parkinson: 12, shad
Blobs.push(shadow3);

let blob3 = new Blob({fill: 0xB83200, radius: 160, parkinson: 16});
Blobs.push(blob3);

let shadow2 = new Blob({fill: 0x000000, radius: 120, parkinson: 11, shad
Blobs.push(shadow2);

let blob2 = new Blob({fill: 0xEC3D02, radius: 120, parkinson: 13});
Blobs.push(blob2);

let shadow1 = new Blob({fill: 0x000000, radius: 80, parkinson: 13, shado
Blobs.push(shadow1);

let blob1 = new Blob({fill: 0xED9F93, radius: 80, parkinson: 12});
Blobs.push(blob1);
```

Separation: Context