

# *The Rose*

\_ Lang Leav \_

Truong Nguyen Kieu My\_S4037623

## ***Project Brief***

“The Rose” is an interactive web-based poetry explores the journey of love—from the joy of falling in love, the hurt that sometimes follows, and the way love lingers in the heart, even when we try to let it go of a Australian poet, Lang Leav. In this project, I want to convey these feeling visually to people who are in love or having a break down to invites them think about their own experiences with love and its many sides, offering a heartfelt and meaningful reflection.

## ***The Rose***

Have you ever truly loved a rose,  
and watched her slowly bloom;  
and as her petals would unfold,  
you grew drunk on her sweet perfume.

Have you ever seen her gently dance,  
her leaves all wet with dew;  
and quivered with a new romance—  
the wind, he loved her too.

Have you ever longed for only her,  
on nights that go on and on;  
for now, her face is all a blur,  
like a memory kept too long.

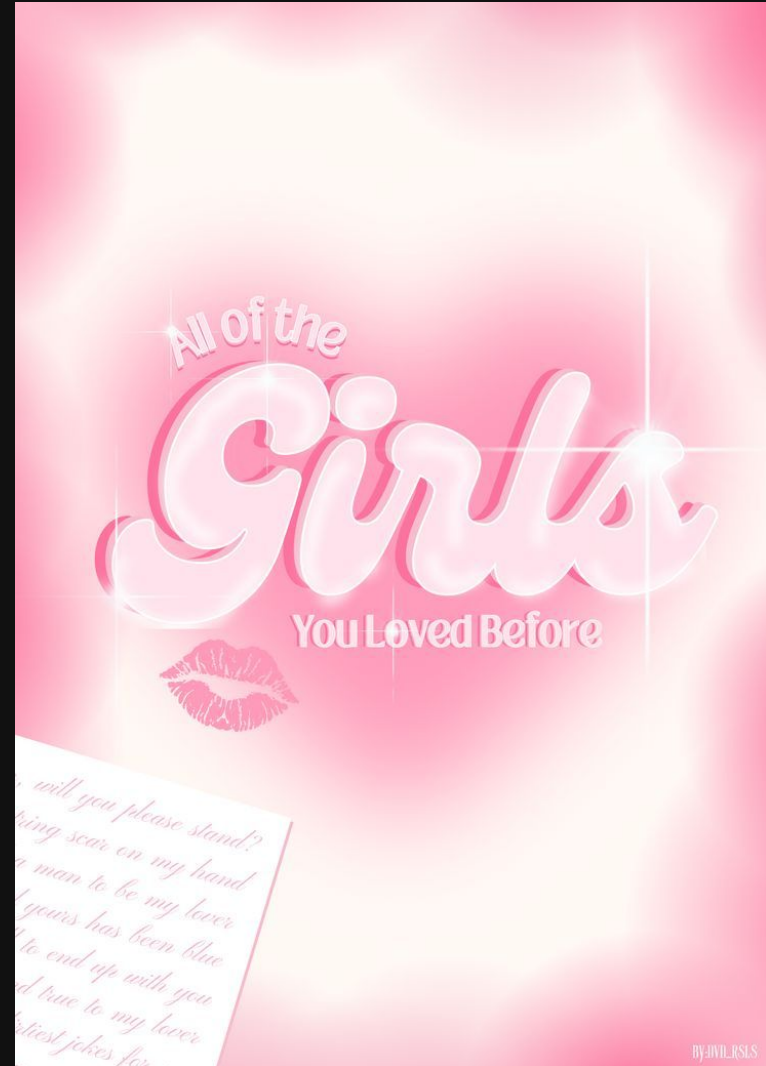
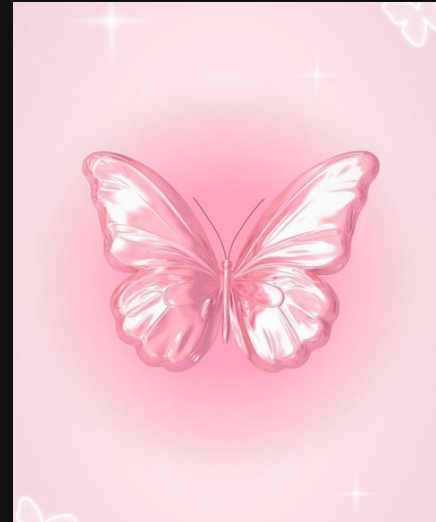
Have you ever truly loved a rose,  
and bled against her thorns;  
and swear each night to let her go,  
then love her more by dawn.”

by Lang Leav

## ***Concept***

In this project, I want combine the soft pastel aesthetic with the dark vibe, which are two contrast vibe together to trigger the change of feeling in phases of love from the joy of being loved to the pain and linger in love.

# Moodboard



These shiny dreamy vibe with pink gradient background triggers the in love feeling while the dark red vibe convey the misery, lingers in love.

# Design System

## Body

Garamond Premier Pro Display

The quick brown fox  
jumps over the lazy dog

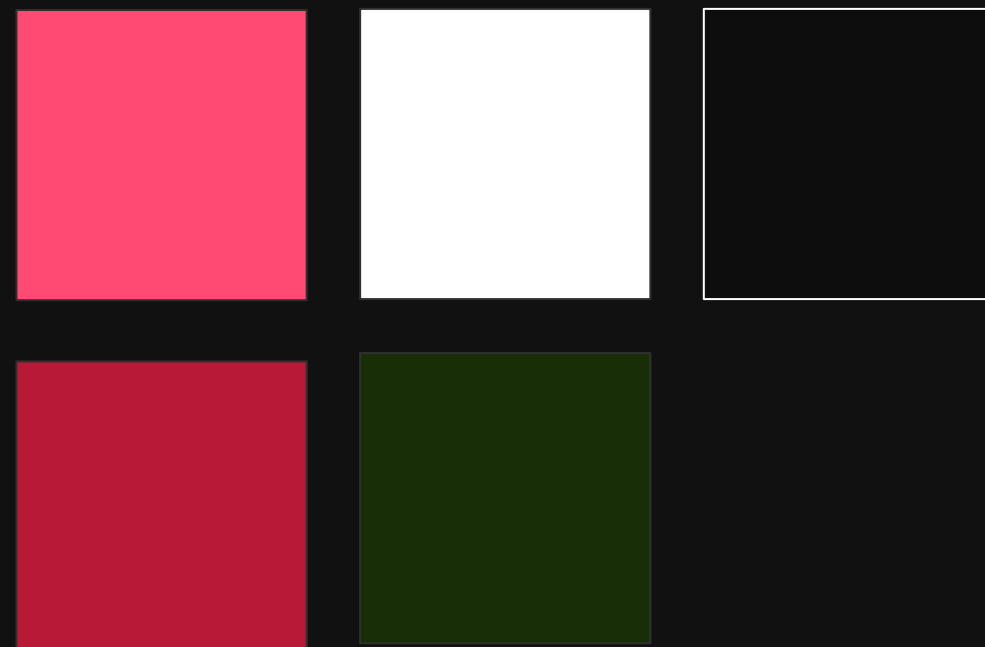
## Title & Highlight

Tangier Bold

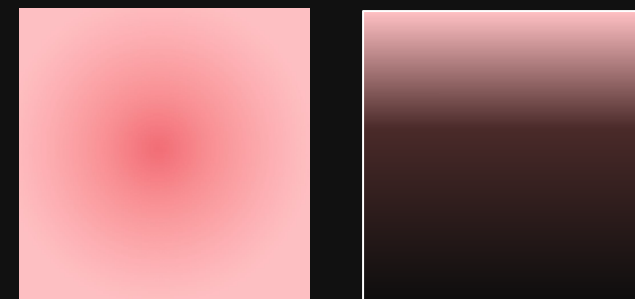
*The quick brown fox  
jumps over the lazy dog*

There is a high contrast between these two font about light and bold, regular and italic.

## Solid

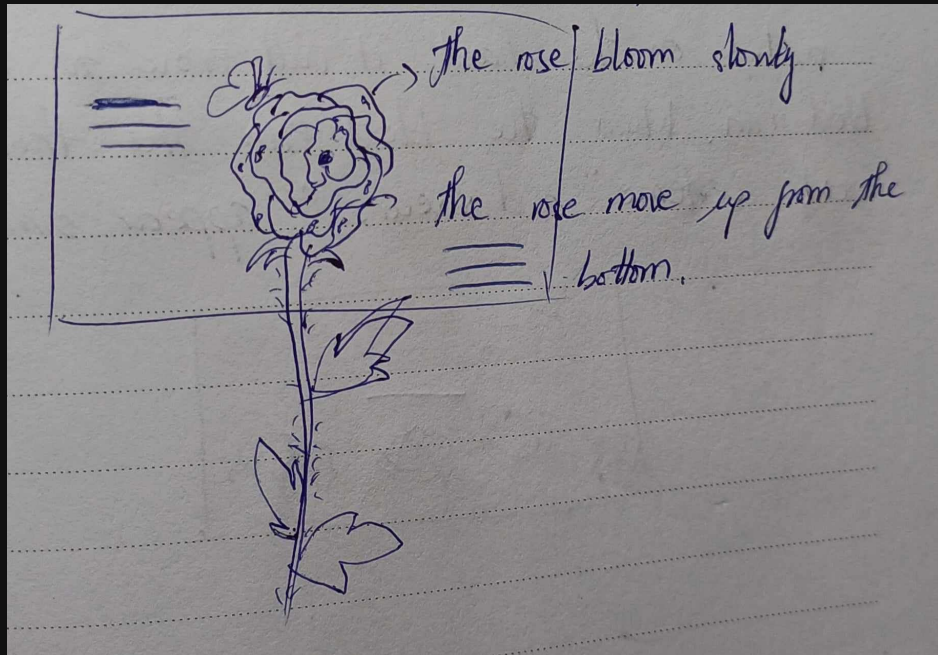


## Gradient

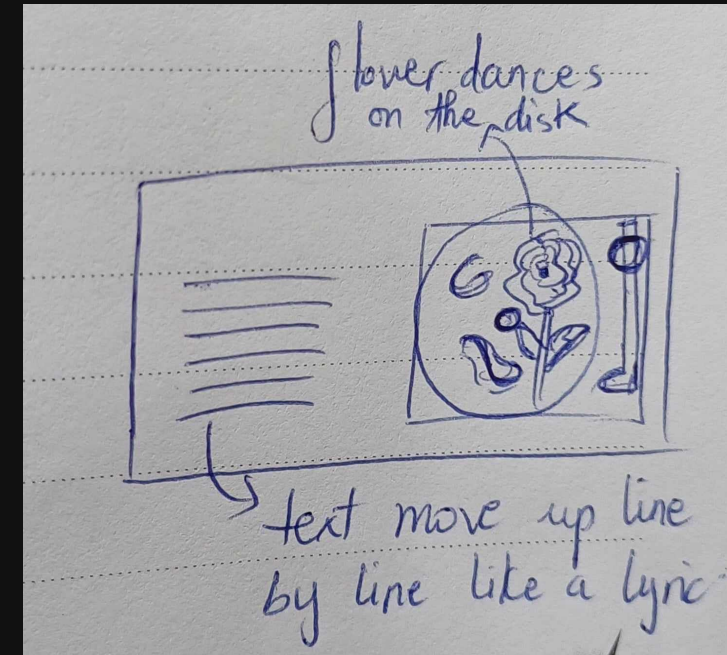


I used the radial gradient for each section to create the dreamy shiny vibe and the linear gradient for the whole project to have a smooth transition between 2 different vibe.

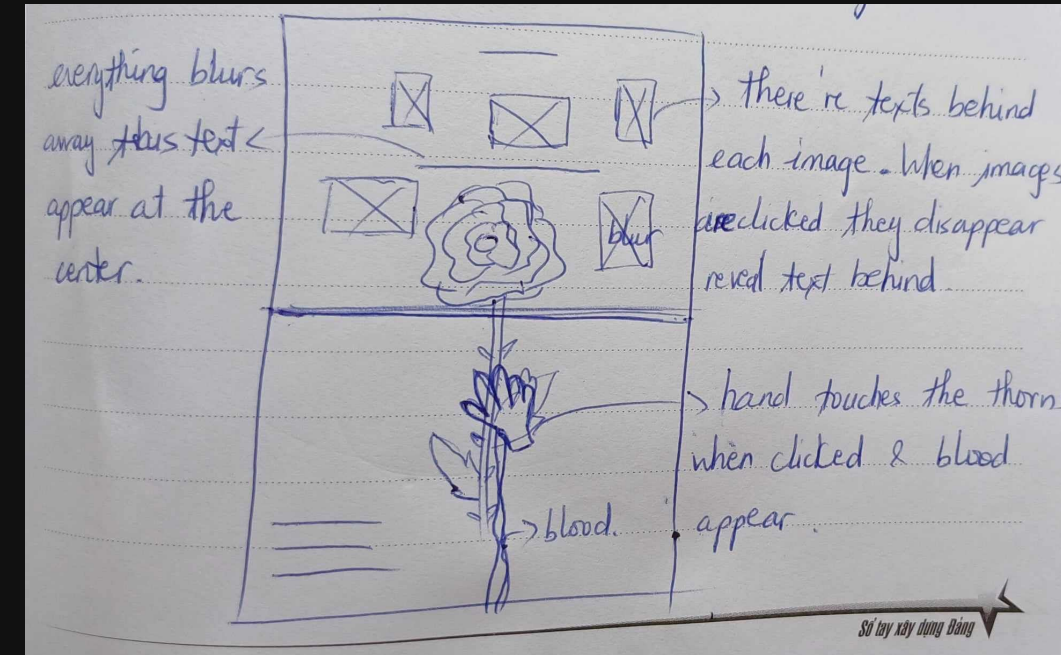
# Ideation & Sketches



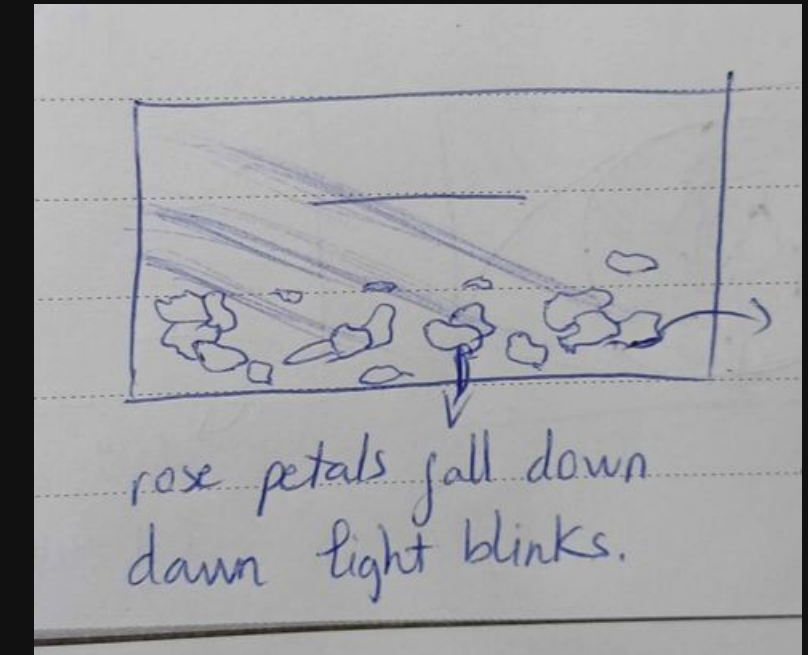
In the first section, I use the image of a blooming flower with a lots of sparkle and butterfly flying around stand for her sweet perfume.



The second paragraph tells about a rose dancing so I want to have a video of a rose dancing in a tender music, which makes me think about a vintage record.



There will have a huge rose place between section 3&4. The paragraph talks about the blur away of old memories so in section 3 I want to have some pictures capture beautiful moments when the rose danced with the wind before everything blurs away. Moreover, rose petals start to fall down. In section 4 is a scene of a hand bled against rose's thorns to signify for pains in love



In the last scene, I want to capture the scene the rose petals fall down on the floor in the dark floor to convey the words "let her go" then the dawn light shine on the floor to end the project.

# Design & Prototype

## CSS, HTML, JVS



When think of soft pastel vibes, I think that chrome text is the most appropriate type that can suit well with the blink and shiny vibe.

```

.author{
  position: relative;
  width: fit-content;
  font-family: "garamond-premier-pro-display", serif;
  font-weight: 400;
  font-style: normal;
  color: #ffffff;
  font-size: 60px;
  letter-spacing: -0.66px;
  line-height: 90px;
  white-space: nowrap;
  visibility: hidden;
}
.caret {
  display: inline-block;
  width: 2px;
  height: 1em;
  background-color: #ffffff;
  vertical-align: bottom;
  animation: blink 0.7s infinite;
}
@keyframes blink {
  0%, 50% { opacity: 1; }
  50%, 100% { opacity: 0; }
}
.title{
  width: fit-content;
  height: auto;
  position: relative;
  max-width: 80vw;
  object-fit: cover;
  animation: emphasize 2s cubic-bezier(0.39, 0.575, 0.565, 1) forwards;
}
@keyframes emphasize {
  0%{
    opacity: 0;
    transform: scale(0);
  }
  90%{
    transform: scale(1.2);
  }
  100%{
    opacity: 1;
    transform: scale(1);
  }
}
.butterfly-bottom {
  position: absolute;
  width: fit-content;
  height: fit-content;
  top: 70%;
  left: 20%;
  object-fit: cover;
  opacity: 0;
  animation: landing-page-opacity 1.5s linear forwards;
  animation-delay: 4.5s;
}
@keyframes landing-page-opacity {
  0%{
    opacity: 0;
  }
  20%{
    opacity: 1;
  }
  40%{
    opacity: 0.5;
  }
  60%{
    opacity: 1;
  }
  80%{
    opacity: 0.5;
  }
  100%{
    opacity: 1;
  }
}
<div class="landingPage">
  
  
  <div class="author" id="author-text">by Lang Leav</div>
  
  
</div>
document.addEventListener("DOMContentLoaded", function() {
  const textElement = document.getElementById("author-text");
  const text = "by Lang Leav";
  const typingSpeed = 100;
  const delayBeforeStart = 2500;
  let index = 0;
  function type() {
    if (index < text.length) {
      textElement.innerHTML = text.substring(0, index + 1) + '<span class="caret"></span>';
      index++;
      setTimeout(type, typingSpeed);
    } else {
      textElement.innerHTML = text;
    }
  }
  setTimeout(() => {
    textElement.style.visibility = 'visible';
    type();
  }, delayBeforeStart);
});

```

I want to have some title animation to attract impression at the first glance so I use keyframes in CSS to create zoom in and out effect combine with the typing effect for the author name. My plan was animate the butterfly then animate them moving by using CSS but I cannot remove the black background when I convert the video to WebM format so I changed my plan to make them blink only using keyframes in CSS.

# Design & Prototype



I used the image of blooming rose to with a lot of butterfly and sparkle flying around to show the sweet perfume of the flower.

## CSS

```
.hover-effect{
  transition: opacity 0.5s ease-in-out;
}
.small-rose{
  width: fit-content;
  height: fit-content;
  justify-self: center;
  display: grid;
  align-self: center;
  padding-left: 5%;
  position: absolute;
  left: 50%;
  bottom: 0;
  transform: translateX(-50%) translateY(50%);
  transition: transform 1.5s ease-in-out;
}
.small-rose.center {
  transform: translateX(-50%) translateY(0%);
}
```

```
.white-glow{
  text-shadow: 0 0 5px #ffffff,
    0 0 20px #ffffff,
    0 0 30px #ffffff,
    0 0 40px #ffffff;
  opacity: 0.5;
  animation: glow 1.5s linear infinite;
  animation-direction: alternate;
}
@keyframes glow {
  0%{
    opacity: 0.5;
  }
  100%{
    opacity: 1;
  }
}
.sparkle {
  position: absolute;
  width: fit-content;
  height: fit-content;
  top: 2%;
  left: 0%;
  object-fit: cover;
  transform: scale(0.8);
  transition: opacity 0.5s ease-in-out;
  opacity: 0;
  z-index: 0;
  animation: blink-animation 5s steps(2, start) infinite;
}
@keyframes blink-animation {
  50% {
    opacity: 0;
  }
}
.butterfly-top-2 {
  position: absolute;
  width: fit-content;
  height: fit-content;
  top: 15%;
  right: 30%;
  opacity: 0;
  animation: blink-animation 6s steps(2, start) infinite;
}
```

```
.container-1 {
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: flex-start;
  gap: 300px;
  height: 100vh;
  position: relative;
}
.cau-12 {
  display: flex;
  align-items: flex-start;
  align-self: stretch;
  flex-direction: column;
  position: relative;
  flex: 0 0 auto;
  overflow: hidden;
  padding-left: 5%;
}
.appear-effect{
  transition: opacity 2.5s ease-in-out;
  opacity: 0;
}
.fade-in{
  opacity: 1;
}
.cau-1 {
  display: flex;
  width: 439px;
  align-items: flex-start;
  flex-direction: column;
  position: relative;
  flex: 0 0 auto;
  gap: 7px;
}
```

## HTML & JVS

```
<div class="container-1">
  <div class="cau-12">
    <div class="text-wrapper-1">Have you ever</div>
    <div class="text-wrapper-2 appear-effect">truly loved a rose,</div>
  </div>
  <div class="cau-2">
    <div class="text-wrapper-3">and watched her</div>
    <div class="text-wrapper-4 appear-effect">slowly bloom;</div>
  </div>
  <div class="cau-3">
    <div class="p1">and as her petals would unfold,</p>
    <div class="u-grew-drunk-on">
      <div class="text-wrapper-5">you grew</div>
      <div class="text-wrapper-6">drunk on</div>
    </div>
  </div>
  <div class="her-sweet-perfume">
    <div class="her"><span class="span-1">her</span> <span class="text-wrapper-7">&nbsp;&nbsp;&nbsp;</span></div>
    <div class="text-wrapper-8 appear-effect white-glow id="sweet-perfume">sweet perfume.</div>
  </div>
  
  
  
  
  
  
</div>
```

```
document.addEventListener("DOMContentLoaded", function() {
  const hoverText = document.getElementById("sweet-perfume");
  const appearSparkle = document.querySelectorAll('.hover-effect');

  hoverText.addEventListener("mouseover", function() {
    appearSparkle.forEach(element => {
      element.style.opacity = '1';
    });
  });

  hoverText.addEventListener("mouseout", function() {
    appearSparkle.forEach(element => {
      element.style.opacity = '0';
    });
  });
});
```

```
document.addEventListener("DOMContentLoaded", function () {
  const container1 = document.querySelector('.container-1');
  const smallRoseElement = document.querySelector('.small-rose');

  const observer = new IntersectionObserver(
    (entries) => {
      entries.forEach((entry) => {
        if (entry.isIntersecting) {
          smallRoseElement.classList.add('center');
        }
      });
    },
    {
      root: null,
      threshold: 0.5,
    }
  );

  observer.observe(container1);
});
```

```
document.addEventListener("DOMContentLoaded", function() {
  const textElements = document.querySelectorAll('.appear-effect');
  const observerOptions = {
    root: null,
    rootMargin: '0px',
    threshold: 0.1
  };

  const observerCallback = (entries, observer) => {
    entries.forEach(entry => {
      if (entry.isIntersecting) {
        entry.target.classList.add('fade-in');
        observer.unobserve(entry.target); // Stop observing once the animation starts
      }
    });
  };

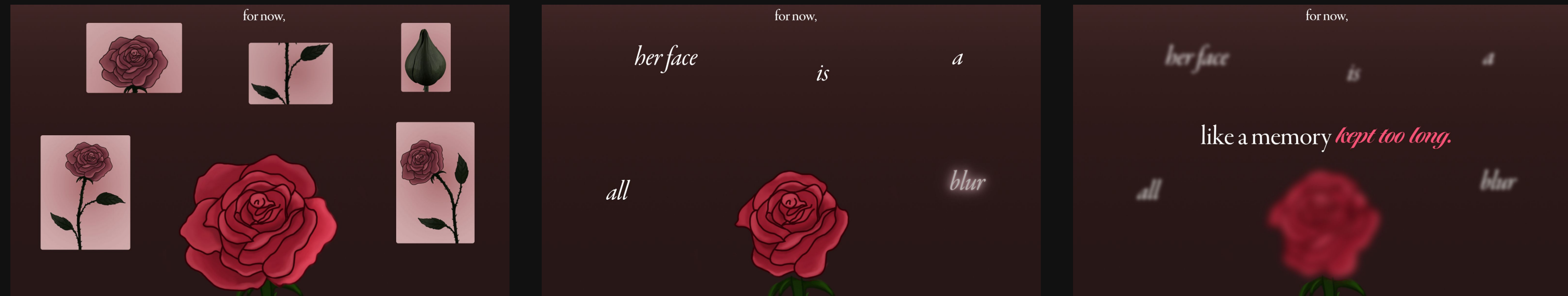
  const observer = new IntersectionObserver(observerCallback, observerOptions);

  textElements.forEach(element => {
    observer.observe(element);
  });
});
```

I use JVS for the moving up of the rose and the appear effect of highlight keywords to make sure that the animation only runs when the section is in view. About the hover effect I used to try using CSS keyframes but it doesn't work well because "Sweet perfume" isn't the parent of sparkle and butterflies so I changed to JVS to have a better control.



# Design & Prototype



For the background, it is transition between the pink and black because of the blur away of old memories, their love isn't "pink" as the beginning. There are also images of the rose beautifully dancing in the past with the "dark fade" filter on top to signify the vintage of the images. When each image is disappear, the petal also fallen down as it is also a part of the old memory and also conveys the message that by the time the love isn't as beautiful as the beginning.

# Design & Prototype

## HTML & JVS

```

<div class="container-4">
  <div class="a">
    
    <div class="text">a</div>
  </div>
  <div class="all">
    
    <div class="text">all</div>
  </div>
  <div class="blur">
    
    <div class="text pink-glow">blur</div>
  </div>
  <div class="her-face">
    
    <div class="text">her face</div>
  </div>
  <div class="for-now">
    <div class="for-now-text">for now,</div>
    <div class="like-a-memory-kept">
      <div class="text-wrapper-15">like a memory</div>
      <div class="text-wrapper-16">kept too long.</div>
    </div>
  </div>
  <div class="is">
    
    <div class="text">is</div>
  </div>
  <div class="font">
    <div class="cau-13">
      <div class="text-wrapper-17">Have you ever</div>
      <div class="truly-loved-a-rose appear-effect">truly loved a rose,</div>
    </div>
    <div class="cau-14">
      <p class="and-bleed-against-her">
        <span class="span">and </span>
        <span class="text-wrapper-18 appear-effect">bled</span>
        <span class="span">against her thorns.</span>
      </p>
    </div>
  </div>
</div>
<div class="container-4">
  <div class="font">
    </div>
</div>
<div class="rose">
  <div class="flower">
    
    
    
    
    
    
  </div>
  <div class="trunk">
    
    
    
    
    
  </div>
</div>
</div>

```

```

document.addEventListener("DOMContentLoaded", function() {
  const imageElements = document.querySelectorAll('.image');

  function changePetalPosition(event) {
    const petalClass = event.target.getAttribute('data-petal');
    const petalElement = document.querySelector(`.${petalClass}`);
    if (petalElement) {
      petalElement.classList.add('new-position');
    }
  }

  imageElements.forEach(image => {
    image.addEventListener('click', changePetalPosition);
  });
});

document.addEventListener("DOMContentLoaded", function() {
  const container4 = document.querySelector('.container-4');
  const elementsToAnimate = document.querySelectorAll('.is, .her-face, .blur, .all, .a');

  const observerOptions = {
    root: null,
    rootMargin: '0px',
    threshold: 0.1,
  };

  const observerCallback = (entries, observer) => {
    entries.forEach(entry => {
      if (entry.isIntersecting) {
        // Add the .animate class to each element
        elementsToAnimate.forEach((element, index) => {
          setTimeout(() => {
            element.classList.add('animate');
            // Stagger animations with a 500ms delay
            }, index * 500);
          });
        observer.unobserve(entry.target); // Stop observing once triggered
      }
    });
  };

  const observer = new IntersectionObserver(observerCallback, observerOptions);
  observer.observe(container4); // Observe container-4
});

document.addEventListener("DOMContentLoaded", function() {
  const blurImage = document.querySelector('.blur .image');
  const herFaceImage = document.querySelector('.her-face .image');
  const aImage = document.querySelector('.a .image');
  const isImage = document.querySelector('.is .image');
  const allImage = document.querySelector('.all .image');

  function hideImage(event) {
    event.target.classList.add('hidden');
  }

  blurImage.addEventListener('click', hideImage);
  herFaceImage.addEventListener('click', hideImage);
  aImage.addEventListener('click', hideImage);
  allImage.addEventListener('click', hideImage);
  isImage.addEventListener('click', hideImage);
});

document.addEventListener("DOMContentLoaded", function() {
  const blurText = document.querySelector('.blur .text');
  const textElements = document.querySelectorAll('.text');
  const pinkGlowElement = document.querySelector('.pink-glow');
  const flowerElement = document.querySelector('.flower-only');
  const memoryElement = document.querySelector('.like-a-memory-kept');

  blurText.addEventListener('click', function() {
    textElements.forEach(element => {
      element.classList.add('blur-away');
    });
  });

  if (flowerElement) {
    flowerElement.classList.add('blur-away');
  }

  if (pinkGlowElement) {
    pinkGlowElement.classList.remove('pink-glow');
  }

  setTimeout(() => {
    if (memoryElement) {
      memoryElement.classList.add('appear');
    }
  }, 2000);
});

```

## CSS

```

@keyframes her-face-move {
  0% {
    top: 25%;
    left: 50%;
    transform: scale(0);
    opacity: 0;
  }
  100% {
    top: 3%;
    left: 15%;
    transform: scale(1);
    opacity: 1;
  }
}

@keyframes blur-move {
  0% {
    top: 25%;
    left: 50%;
    transform: scale(0);
    opacity: 0;
  }
  100% {
    top: 18%;
    left: 76%;
    transform: scale(1);
    opacity: 1;
  }
}

@keyframes all-move {
  0% {
    top: 25%;
    left: 50%;
    transform: scale(0);
    opacity: 0;
  }
  100% {
    top: 20%;
    left: 6%;
    transform: scale(1);
    opacity: 1;
  }
}

```

```

@keyframes a-move {
  0% {
    top: 25%;
    left: 50%;
    transform: scale(0);
    opacity: 0;
  }
  100% {
    top: 3%;
    left: 77%;
    transform: scale(1);
    opacity: 1;
  }
}

.image {
  position: relative;
  width: auto;
  height: auto;
  object-fit: cover;
  border-radius: 6px;
  z-index: 1;
  transition: opacity 1.5s ease-in-out;
}

.image.hidden {
  opacity: 0;
  visibility: hidden;
  transition: opacity 1.5s ease-in-out, visibility 0s 1.5s;
}

.like-a-memory-kept {
  display: inline-flex;
  align-items: flex-end;
  justify-content: center;
  gap: 15px;
  position: relative;
  flex: 0 0 auto;
  opacity: 0;
  transition: opacity 2s ease-in-out;
  transform: translateY(20px);
}

.like-a-memory-kept.appear {
  opacity: 1;
  transform: translateY(-5px)
}

```

```

.is .animation {
  animation: is-move 2s ease-in-out forwards;
  animation-delay: 1.5s;
}

.her-face .animation {
  animation: her-face-move 2s ease-in-out forwards;
  animation-delay: 1s;
}

.blur .animation {
  animation: blur-move 2s ease-in-out forwards;
  animation-delay: 0.5s;
}

.all .animation {
  animation: all-move 2s ease-in-out forwards;
}

.a .animation {
  animation: a-move 3s ease-in-out forwards;
}

@keyframes is-move {
  0% {
    top: 25%;
    left: 50%;
    transform: scale(0);
    opacity: 0;
  }
  100% {
    top: 6%;
    left: 47%;
    transform: scale(1);
    opacity: 1;
  }
}

.text, .flower-only {
  transition: filter 2s ease-in-out;
}

.text.blur-away {
  filter: blur(5px);
}

.flower-only.blur-away {
  filter: blur(10px);
}

.text, .flower-only {
  filter: blur(0);
}

.pink-glow {
  text-shadow: 0 0 5px #ff4d4d,
    0 0 20px #ff4d4d,
    0 0 30px #ff4d4d,
    0 0 40px #ff4d4d;
  opacity: 0.5;
  cursor: pointer;
  animation: glow 1.5s linear infinite;
  animation-direction: alternate;
}

.pink-glow.stop-glow {
  animation: none;
  opacity: 0no;
}

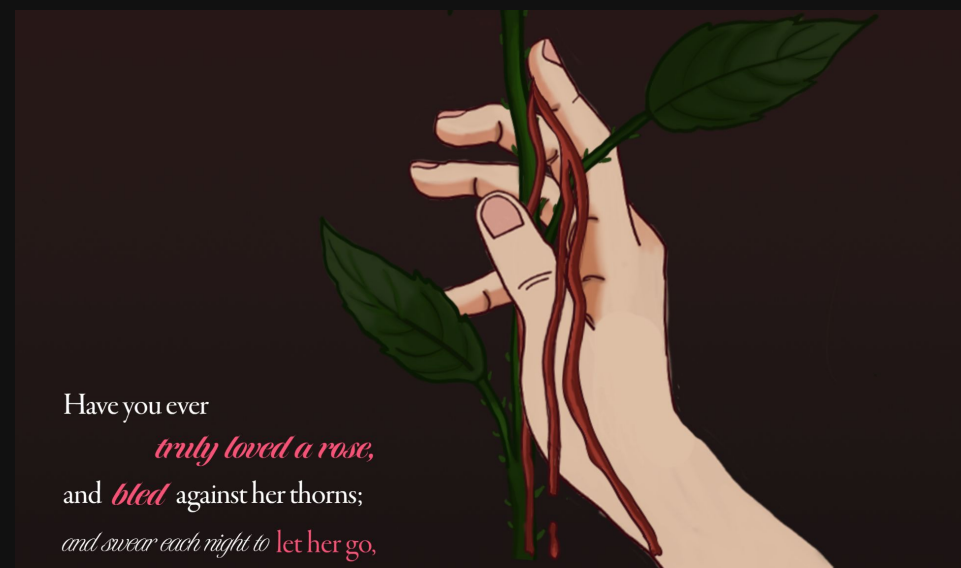
.is, .her-face, .blur, .all, .a {
  position: absolute;
  transform: scale(0);
  opacity: 0;
  transition: opacity 2s ease-in-out, transform 2s ease-in-out;
  display: grid;
  width: fit-content;
  height: fit-content;
  align-items: center;
  justify-content: center;
}

```

For the moving of the image I keyframes each image with different position and time delay on CSS to have a random moving. For the falling down of petal, I associate the image with appropriate data-petal and use JVS to animate them. I also use JVS for the blur away effect and the appear of the last sentence.

# Design & Prototype

## JVS



```
document.addEventListener("DOMContentLoaded", function() {
  const frontHandElement = document.querySelector('.front-hand');
  const backHandElement = document.querySelector('.back-hand');
  const frontBloodElement = document.querySelector('.front-blood');
  const backBloodElement = document.querySelector('.back-blood');

  function moveElements(event) {
    frontHandElement.classList.add('move');
    backHandElement.classList.add('move');
  }

  function handleAnimationEnd() {
    frontBloodElement.classList.add('appear');
    backBloodElement.classList.add('appear');
  }

  if (frontHandElement) {
    frontHandElement.addEventListener('click', moveElements);
    frontHandElement.addEventListener('transitionend', handleAnimationEnd);
  }

  if (backHandElement) {
    backHandElement.addEventListener('click', moveElements);
    backHandElement.addEventListener('transitionend', handleAnimationEnd);
  }
});
```

## CSS

```
.back-blood {
  position: absolute;
  width: fit-content;
  height: fit-content;
  top: 21%;
  left: 54%;
  object-fit: cover;
  transform: scale(1.7) rotate(4.2deg);
  opacity: 1;
  z-index: 3;
  transform-origin: top left;
  transition: clip-path 3s ease-in-out;
  clip-path: polygon(0% 0%, 100% 0%, 100% 0%, 0% 0%);
}

.front-blood {
  position: absolute;
  width: fit-content;
  height: fit-content;
  top: 22%;
  left: 51.5%;
  object-fit: cover;
  transform: scale(1.45);
  opacity: 1;
  z-index: 2;
  clip-path: polygon(50% 0%, 100% 0%, 100% 0%, 50% 0%);
  transition: clip-path 3s ease-in-out;
  transform-origin: top;
}

.back-blood.appear {
  clip-path: polygon(0% 0%, 100% 0%, 100% 100%, 0% 100%);
}

.front-blood.appear {
  clip-path: polygon(0% 0%, 100% 0%, 100% 100%, 0% 100%);
}

.back-hand {
  position: absolute;
  width: 846px;
  height: 819px;
  top: 120px;
  left: 346px;
  z-index: 2;
  object-fit: cover;
  transition: transform 1s ease-in-out;
  transform: translate(0, 0);
  cursor: pointer;
}

.front-hand {
  position: absolute;
  width: 377px;
  height: 511px;
  top: 521px;
  left: 760px;
  z-index: 3;
  object-fit: cover;
  transition: transform 1s ease-in-out;
  transform: scale(1.35);
  cursor: pointer;
}

.front-hand.move {
  transform: translate(-300px, -124px) scale(1.35);
}

.back-hand.move {
  transform: translate(-300px, -124px);
}
```

To illustrate the “bled against her thorns” I use the image of a hand holding the trunk with lots of thorns.

For the text I used the same appear effect as above. For the hand I cut them into 2 parts and layer them upon an down the trunk. All the hand and blood are position absolute to the trunk so it won't move apart from each other when I change to another computer. I use clip path in CSS to animate the blood and JVS to have more control on the moving of the hand.

# Design & Prototype



```
document.addEventListener("DOMContentLoaded", function() {
  const container5 = document.querySelector('.container-5');
  const lastScenePetals = document.querySelectorAll('.last-scene-petal');
  const lightElement = document.querySelector('.light');

  const observerOptions = {
    root: null,
    rootMargin: '0px',
    threshold: 0.1
  };

  const observerCallback = (entries, observer) => {
    entries.forEach(entry => {
      if (entry.isIntersecting) {
        lastScenePetals.forEach((element, index) => {
          setTimeout(() => {
            element.classList.add('move');
            if (index === lastScenePetals.length - 1) {
              element.addEventListener("transitionend", () => {
                lightElement.classList.add('blink');
              });
            }, index * 500); // Stagger animations with a 500ms delay
          });
        });
        observer.unobserve(entry.target); // Stop observing once the animation starts
      }
    });
  };

  const observer = new IntersectionObserver(observerCallback, observerOptions);
  observer.observe(container5);
});
```

```
.last-scene-petal {
  justify-self: center;
  bottom: 0px;
  left: 0px;
  object-fit: cover;
  z-index: 1;
  transform: translate(-100%) scale(0);
  transition: transform 4s ease-in-out, opacity 4s ease-in-out;
}
.last-scene-petal.move {
  transform: translate(0%) scale(1);
}
.light {
  position: absolute;
  width: fit-content;
  height: fit-content;
  bottom: 0px;
  left: 0;
  object-fit: cover;
  transition: opacity 4s linear;
  opacity: 0;
}
.light.blink {
  animation: light-opacity 4s linear forwards;
}
@keyframes light-opacity {
  0% {
    opacity: 0;
  }
  50% {
    opacity: 0.7;
  }
  75% {
    opacity: 0;
  }
  100% {
    opacity: 1;
  }
}
```

For the last scene, it is a like the end for section 4, where rose petals start to fall down and for the project as well with the dawn light start to shine in the dark space, indicate there maybe the beginning of a new love cycle.



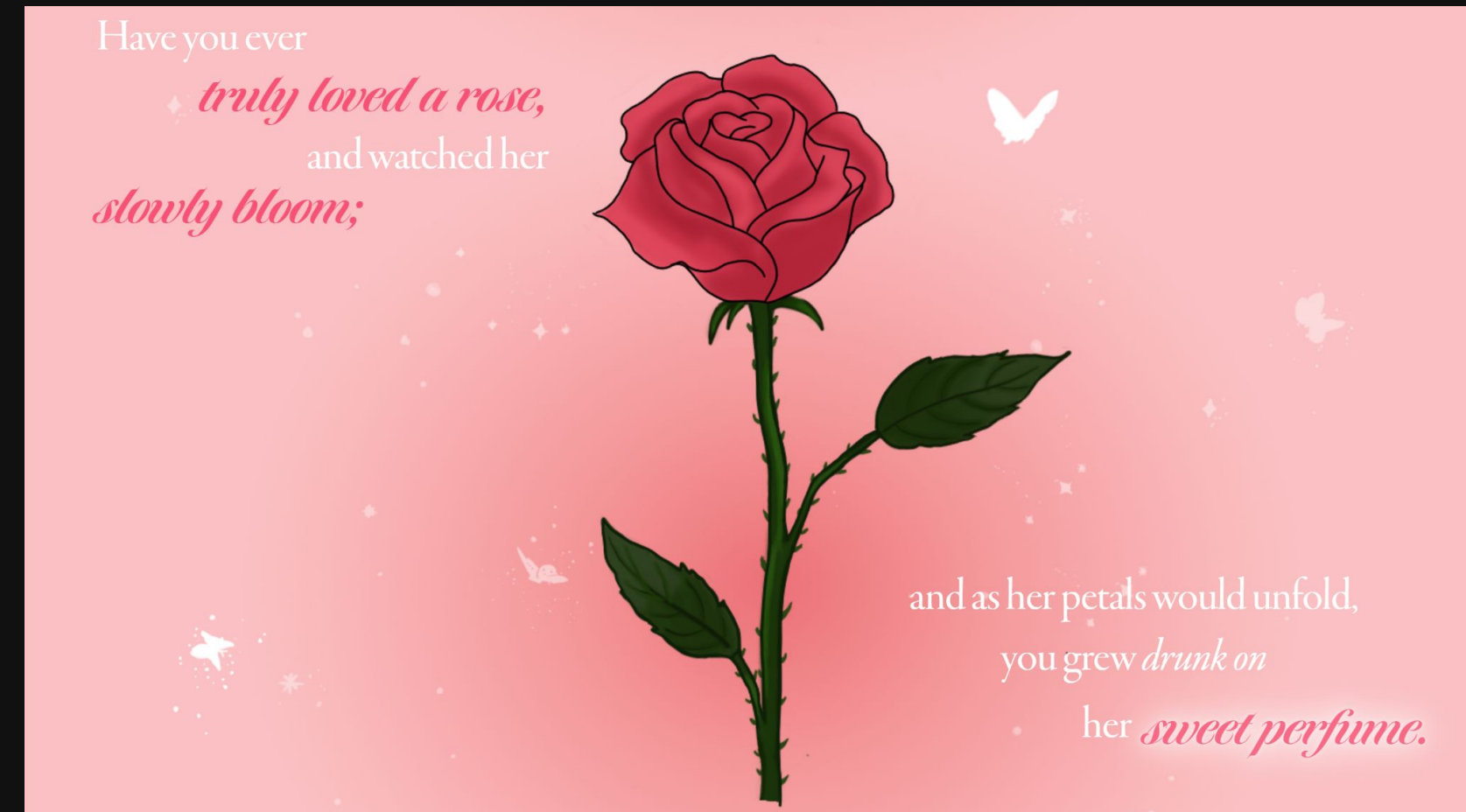
For the code. I use math to have a random fallen down duration time ranging from 2-4s and blink animation of the light in CSS

# Design Iteration

## 1. Inconsistent art style:



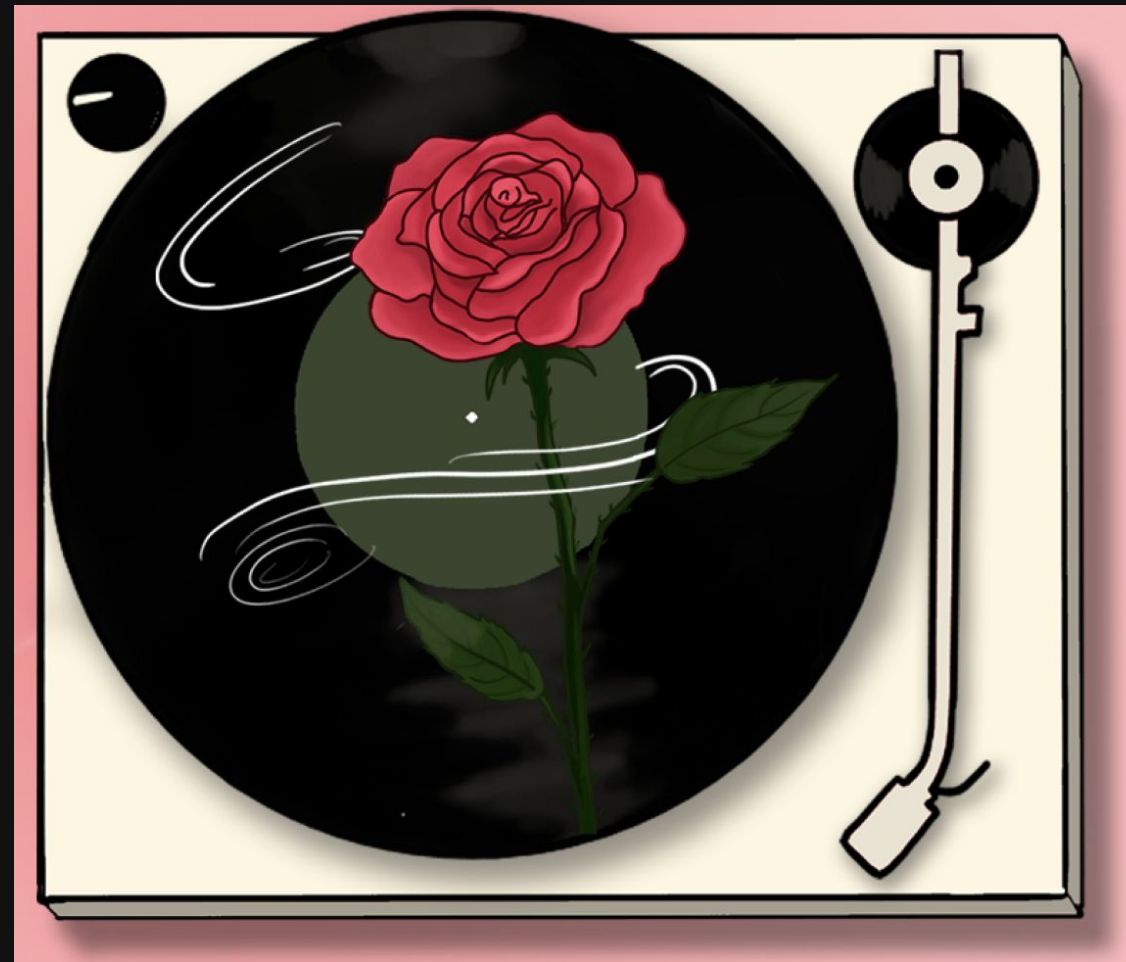
Only the rose is 3D and others surrounding things and others illustration are 2D made the project become inconsistent in art style



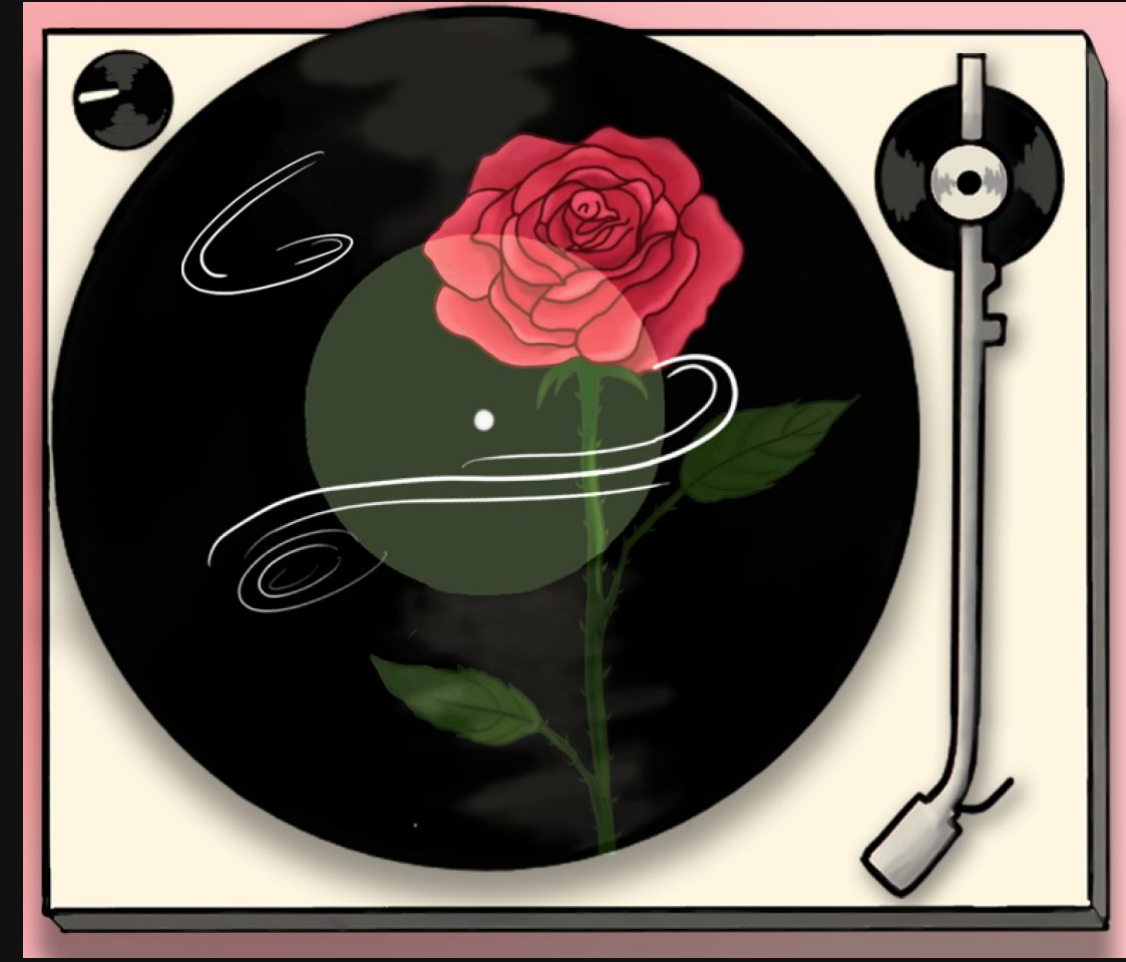
Change the rose to 2D with shade to add more depth to the rose.

# Design Iteration

## 1. Inconsistent art style:



Only the rose has shade and depth, others surrounding things are kinda flat.



Adding shade to all surrounding things such as record, record handle, and make the shade more visible to make them all consistent.

# Design Iteration

## 1. Inconsistent art style:



Because other previous sections have gradient background, so I change the background of this section to gradient as well instead of black to have a more consistent throughout the project as well as more shade to the blood and change the images to make them all consistent.

# Design Iteration

## 2. Blood animation

```
.back-blood.appear {
  opacity: 1;
  transform: scale(1.7) rotate(4.2deg);
}
.front-blood.appear {
  opacity: 1;
  transform: scale(1.45);
}
.front-hand {
  position: absolute;
  width: 377px;
  height: 511px;
  top: 521px;
  left: 760px;
  z-index: 3;
  object-fit: cover;
  transition: transform 1s ease-in-out;
  transform: scale(1.35);
  cursor: pointer;
}
.front-hand.move {
  transform: translate(-300px, -124px) scale(1.35);
}
.back-hand.move {
  transform: translate(-300px, -124px);
}
```


```
.back-blood {
  position: absolute;
  width: fit-content;
  height: fit-content;
  top: 21%;
  left: 54%;
  object-fit: cover;
  transform: scale(1.7) rotate(4.2deg);
  opacity: 1;
  z-index: 3;
  transform-origin: top left;
  transition: clip-path 3s ease-in-out;
  clip-path: polygon(0% 0%, 100% 0%, 100% 0%, 0% 0%);
}
.front-blood {
  position: absolute;
  width: fit-content;
  height: fit-content;
  top: 22%;
  left: 51.5%;
  object-fit: cover;
  transform: scale(1.45);
  opacity: 1;
  z-index: 2;
  clip-path: polygon(50% 0%, 100% 0%, 100% 0%, 50% 0%);
  transition: clip-path 3s ease-in-out;
  transform-origin: top;
}
.back-blood.appear {
  clip-path: polygon(0% 0%, 100% 0%, 100% 100%, 0% 100%);
}
.front-blood.appear {
  clip-path: polygon(0% 0%, 100% 0%, 100% 100%, 0% 100%);
}
```

I used to use scale and set the origin to top to animate the blood but it looked weird and doesn't look like the blood flowing so I changed to use clip path instead to have a better result.

# Final Design

*The Rose*  
by Lang Leav

Have you ever  
*truly loved a rose,*  
and watched her  
*slowly bloom;*



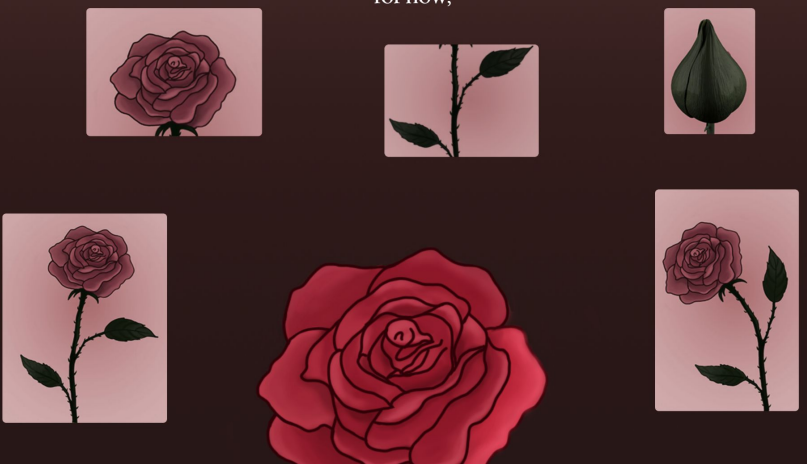
and as her petals would unfold,  
you grew *drunk on*  
her *sweet perfume.*

Have you ever seen  
*her gently dance,*  
her leaves all wet with dew;  
and quivered with  
*a new romance —*  
*the wind,*  
he *loved* her too.




Have you ever *longed for only her;*  
on nights that go on and on;

for now,



Have you ever  
*truly loved a rose,*  
and *bled* against her thorns;  
*and swear each night to let her go,*



*then love her more by dawn.*



# References

Get Back Into Drawing. (n.d.). [Image] *Pinterest*. Available at: <https://www.pinterest.com/pin/903956956454210256/> [Accessed 26 Nov. 2024].

lena (n.d.). 'Half moonshine, full eclipse'. [Poster] *Pinterest*. Available at: <https://www.pinterest.com/pin/903956956454210144/> [Accessed 26 Nov. 2024].

Love Cuts. (n.d.). [Image] *Pinterest*. Available at: <https://www.pinterest.com/pin/903956956454210753/> [Accessed 26 Nov. 2024].

*Metamorphosis*. (n.d.). [Image] *Pinterest*. Available at: <https://www.pinterest.com/pin/903956956454210825/> [Accessed 26 Nov. 2024].

Pastel Pink Butterfly Wallpaper. (n.d.). [Wallpaper] *Pinterest*. Available at: <https://www.pinterest.com/pin/903956956454209630/> [Accessed 26 Nov. 2024].

Pinterest. (2017). 'half moonshine, full eclipse' – taylor swift [poster]. [online] Available at: <https://www.pinterest.com/pin/903956956454210212/> [Accessed 26 Nov. 2024].

st.martin, cesar (2019). *Tumblr*. Available at: <https://cesar-stmartin.tumblr.com/> [Accessed 26 Nov. 2024].

this love – taylor swift [poster]. (n.d.). [Poster] *Pinterest*. Available at: <https://www.pinterest.com/pin/903956956454210142/> [Accessed 26 Nov. 2024].