MOOD PAINT

FROM SIGNAL TO CANVAS

FABRICADEMY
textile and technology academy

FABLAB León

Interactive art generated from biometric data

Emma Yañez

MOOD PAINT
FROM SIGNAL TO CANVAS

INTERACTIVE ART GENERATED FROM BIOMETRIC DATA

Fabricademy Final Project 2025/2026

Project Concept & Research
Emma Yañez

Mentoring & Support

Nuria Robles
Anoush Arshakyan
Louise Massacrier
Emma Pareschi
Rachel Freire
Aristárco Cortés

Special thanks to Alfonso Yañez

INDEX

ABSTRACT

The project is a hybrid art-technology system that translates physiological data from the human body into generative visual environments. Using skin conductance (GSR) and ECG heartbeat data captured through wearable sensors, the system produces a unique, real-time audiovisual output, as biometric signals are distinct for every individual at any given moment.
Mood Paint is a painting system that translates emotional states into visual compositions in real time. It is not about expressing emotion intentionally—it bypasses intention.

This project explores:

- The body as a generator of art
- Emotion without language
- The translation of invisible states into visible form

A portrait—but not of your appearance. What you are seeing is not a picture of you. It is not your face or your body.
It is a portrait of your internal state in this exact moment.

Through this project, I explored the relationship between human emotions, biometric signals, and color, applying the knowledge I gained in Fabricademy in Wearables, E-textiles, and Computational couture.

KEYWORDS

WEREABLES | SENSORS | HEARTBEAT | GSR | GALVANIC SKIN RESPONSE | AI | VISUALS | POETRY | VOICE | IMAGE GENERATION | PAINTING | MOOD | EMOTIONS | COLOR | ELECTRONICS | PROGRAMMING | SHADERS | BIOMETRIC | ELECTRODES

INTRODUCTION

Throughout history, people have expressed their emotions through painting, dance, and music—forms of art that translate feelings into visible and audible experiences.
But what if your body could speak on its own, unintentionally becoming the medium of expression?
This project explores that possibility. It seeks to expand access to emotional expression by offering a system that allows anyone—regardless of artistic training—to transform their inner states into visual form.

MoodPaint is not about expressing emotion intentionally.
It bypasses intention.
The system captures subtle physiological signals (skin conductance and pulse variability) and translates them into dynamic audiovisual environments. The painting is not created by choice — it is generated by the body.

The MoodPaint experience begins the moment the participant puts on the bracelet or glove. Without needing artistic skill or conscious control, their physiological signals quietly shape the evolving painting in front of them. As colors shift, expand, or soften in response to subtle changes in their emotional state, participants witness their inner world becoming visible in real time. The result is both surprising and intimate — a living artwork that reflects a moment of being, transformed directly from body to canvas.

In gallery mode, the participant stands or moves naturally in front of a screen or projection while the painting continuously evolves based on their internal state. No instructions are required — the interaction is passive yet deeply responsive.
In personal mode, the glove allows for additional layers of interaction through gesture and movement, enabling more nuanced visual modulation. The system can generate ephemeral real-time visuals or capture and export a final composition as a digital artwork.
MoodPaint requires no artistic training — the body itself becomes the interface.

RESEARCH

Does your heart sense your emotional state?

Emotions and the heart

Recent HeartMath studies define a critical link between the heart and brain. The heart is in a constant two-way dialogue with the brain — our emotions change the signals the brain sends to the heart and the heart responds in complex ways. However, we now know that the heart sends more information to the brain than the brain sends to the heart. And the brain responds to the heart in many important ways. This research explains how the heart responds to emotional and mental reactions and why certain emotions stress the body and drain our energy. As we experience feelings like anger, frustration, anxiety and insecurity, our heart rhythm patterns become more erratic. These erratic patterns are sent to the emotional centers in the brain, which it recognizes as negative or stressful feelings. These signals create the actual feelings we experience in the heart area and the body. The erratic heart rhythms also block our ability to think clearly.

How Do Amusement, Anger and Fear Influence Heart Rate and Heart Rate Variability?

More than one hundred years ago, William James maintained that discrete emotional experiences could be identified with unique patterns of bodily changes, which are produced by autonomic nervous system activity. This notion was critiqued by Cannon (1927), who provided evidence that autonomic events are too slow, intensive, and undifferentiated to contribute to emotions. However, recent investigations have suggested that the patterns of autonomic activities could be related to discrete emotions, supporting James's original idea (Cacioppo et al., 2000). Findings from the early research on autonomic activity appeared to be heterogeneous. According to Levenson (1988), this inconsistency in prior studies was largely due to various methodological problems, including the failure to identify that an independent emotional state has been aroused and the unreliability of physiological recordings. To solve these problems, we selected the three most differentiated video clips for amused, fearful, and angry inducement, respectively, from the CAVS, which contains standardized video for specific emotional evoking. Further, the participants were required to evaluate their subjective emotional experience to each video clip immediately after watching it to ensure reliability of their appraisals. Additionally, autonomic analysis was done while participants viewed the video clips as the intensity of each emotion was expected to be the highest during this time. With these experimental manipulations, we expected that specific emotions elicited for the analysis of autonomic activity would be consistent with self-reported ratings.

In the present study, we investigated the influence of specific emotions on the pattern of cardiac activities, and found that amusement led to a decreased heart rate. This result is consistent with a meta-analysis that revealed that the heart rate is lower for amusement than that of fear, anger, and neutral emotions (Cacioppo et al., 1997; Kreibig, 2010). The current results also revealed that there was no significant difference in heart rate between anger (or fear) and the neutral category of emotion. In this study, we required the participants to watch affective movies to elicit emotions. Presumably, watching movies safely in the laboratory may have weakened the physiological responses of the participants in the anger and fear conditions. Consequently, this study did not find that anger and fear accelerate heart rate, as reported in previous studies (Cacioppo et al., 2000). Further research is therefore needed to investigate whether experienced anger and fear do in fact accelerate heart rate.

RESEARCH

The Psychology of Color: How the Shades Around You Impact Your Emotions

The Science Behind Color Perception

Colors are perceived through the interaction of light with our eyes and brain. When light hits an object, some wavelengths are absorbed while others are reflected. The reflected light enters our eyes, stimulating photoreceptor cells called cones. These cones interpret the light's wavelength, which the brain then translates into the colors we see.

Our perception of color is also influenced by context, lighting, and individual differences. For instance, someone with color blindness might perceive colors differently than someone with normal vision. Additionally, psychological associations with colors are shaped by both innate responses and learned experiences.

The Emotional Impact of Colors

1. Red

Red is a powerful and dynamic color that commands attention. It's often linked with:

- Energy and excitement: Think of red sports cars or the bold red of Coca-Cola's logo.
- Passion and love: Red roses and Valentine's hearts evoke romantic emotions.
- Danger and urgency: Warning signs and stoplights use red to signal caution.

Red can increase heart rate and adrenaline levels, making it both stimulating and overwhelming when overused.

2. Blue

Blue is universally associated with calmness and trust. Its psychological effects include:

- Relaxation: Blue skies and oceans inspire serenity.
- Trustworthiness: Many financial institutions use blue to convey stability.
- Sadness: The term "feeling blue" underscores its connection to melancholy.

Blue is ideal for creating a peaceful environment but can feel cold or distant if not balanced with warmer tones.

3. Yellow

Yellow is the color of sunshine and optimism. Its effects are:

- Happiness and warmth: Bright yellows evoke cheerfulness.
- Attention-grabbing: Used in caution signs and taxis.
- Overstimulation: Excessive yellow can lead to anxiety or agitation.

Yellow is most effective in small doses to add brightness and energy to a space or design.

4. Green

Green symbolizes nature, balance, and growth. Its psychological effects include:

- Harmony and peace: Green spaces promote relaxation and reduce stress.
- Health and renewal: Frequently used in wellness branding.
- Jealousy and greed: The phrase "green with envy" reflects its darker associations.

Green is a versatile color that works well in various settings, from homes to workplaces.

5. Purple

Purple blends the stability of blue and the energy of red, creating a sense of:

- Luxury and creativity: Historically linked with royalty and wealth.
- Spirituality: Often used in meditation spaces.
- Mystery: Its rarity in nature makes it intriguing.

Purple is a great choice for adding sophistication and depth to a design.

6. Black

Black is a color of contrasts—elegant yet intimidating. It's often associated with:

- Sophistication: Black tie events and luxury brands.
- Power and authority: Common in formal attire and high-end products.
- Mystery: Adds an element of the unknown.

While black is versatile, too much can feel oppressive or somber.

7. White

White symbolizes purity and simplicity. Its effects include:

- Cleanliness: Common in medical and minimalist spaces.
- Openness: Makes spaces feel larger and brighter.
- Sterility: Overuse can feel cold or impersonal.

RESEARCH

Galvanic Skin Response (GSR): The Complete Pocket Guide

What is Galvanic Skin Response (GSR) and Why Does It Matter?

The skin tells everything – our skin gives away a lot of information on how we feel when we're exposed to emotionally loaded images, videos, events, or other kinds of stimuli – both positive and negative. No matter whether we are stressed, nervous, fearful, psyched up, stoked, baffled, or surprised – whenever we are emotionally aroused, the electrical conductivity of our skin subtly changes.
One of the most sensitive measures for emotional arousal is Galvanic Skin Response (GSR), also referred to as Electrodermal Activity (EDA) or Skin Conductance (SC).
Galvanic Skin Response originates from the autonomic activation of sweat glands in the skin. The sweating on hands and feet is triggered by emotional stimulation: Whenever we are emotionally aroused, the GSR data shows distinctive patterns that are visible with bare eyes and that can be quantified statistically.

How Does Galvanic Skin Response (GSR) Work in Measuring Emotions?

To understand how GSR works, take a quick step back and have a look at the physiological characteristics of the largest organ of the human body – the skin.
Our skin functions as the principal interface between organism and environment. Together with other organs, it is responsible for bodily processes such as the immune system, thermo-regulation, and sensory-motor exploration:
1. Immune System
As a protective barrier, the skin separates our body from the environment and its threats – mechanical impacts and pressure, variations in temperature, micro-organisms, radiation, and chemical agents.
2. (Thermo-)Regulation
The skin controls body temperature by regulating sweat emission, piloerection ("goosebumps"), and peripheral blood circulation.
3. Sensing and Perception
The skin is an organ of perception. It contains an extensive network of nerve cells that detect and relay changes in the environment based on the activity of receptors for temperature, pressure, and pain.

GSR & emotional arousal

So far, so good: Emotional experiences trigger changes in autonomic arousal quite impressively. Now what does that mean exactly?
Exposure to fear-inducing stimuli (an angry face, the sight of a creepy spider etc.) induce emotional arousal, causing an increase in sweat secretion and, ultimately, measurable electrodermal activity.
In emotional situations, bodily processes are triggered automatically: The heart beats faster, the pulse rises, hands become sweaty. To put it bluntly: While we are physiologically or psychologically aroused (in fear, extreme joy or under stress), we start to sweat.
In case you were thinking sweat running down in streams, let's give the all-clear here: Actually, we don't need to be sweat-flooded in order to see differences in electrodermal activity (in fact, the sweating doesn't even need to be visible).
Besides emotional stimulus properties, recent findings indicate that skin conductance is also sensitive towards other aspects of a stimulus.
Are we familiar with the stimulus or do we encounter it for the first time? Is the stimulus threatening or rewarding? Do we associate the stimulus with wins or losses, love or hate, anticipation and outcome, memory recall or cognitive work? Against this backdrop, changes in skin conductance might also reflect motivational and attentional processing.

RESEARCH

ESP-NOW PROTOCOL

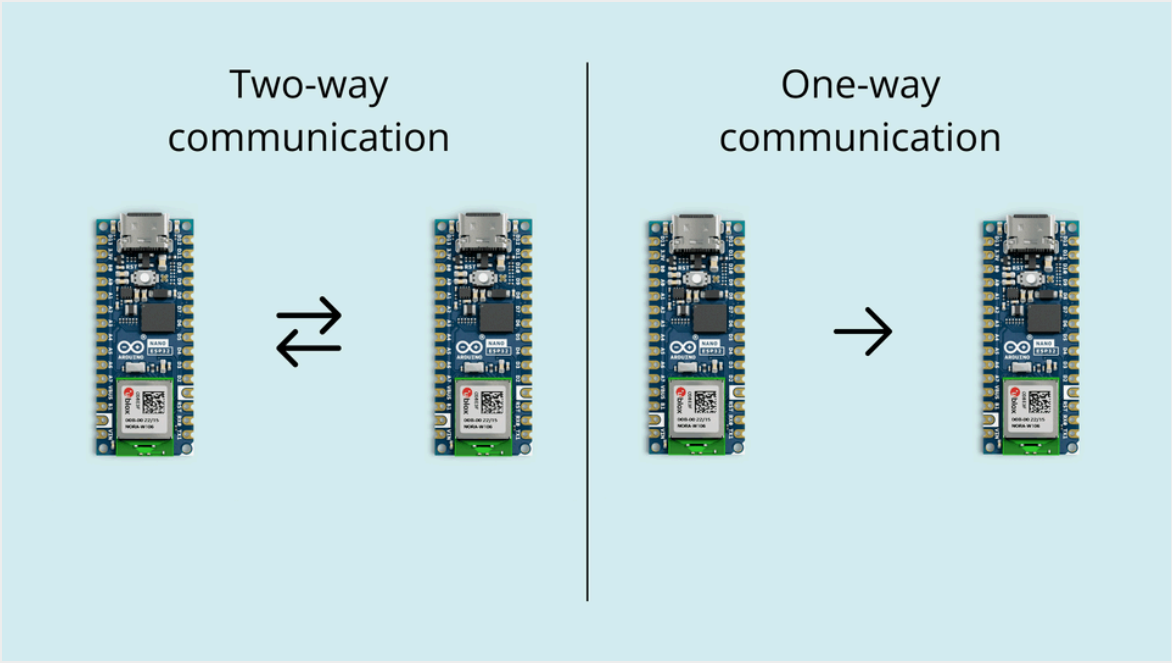
ESP-NOW is a wireless communication protocol defined by Espressif, which enables the direct, quick and low-power control of smart devices, without the need of a router. ESP-NOW can work with Wi-Fi and Bluetooth LE, and supports the ESP8266, ESP32, ESP32-S and ESP32-C series of SoCs. It's widely used in smart-home appliances, remote controlling, sensors, etc.

It has a fast and user-friendly pairing method that is suitable for connecting "one-to-many" and "many-to-many" devices, while also controlling them.

Communication Protocol
ESP-NOW operates as a peer-to-peer (P2P) protocol, meaning it allows direct communication between two ESP8266 or ESP32 devices without the need for a central server or access point, e.g. a Wi-Fi® router. Each ESP device has a unique **MAC address** which is used to identify the receiving board.
ESP-NOW can be set up in different ways:
One-way communication: In one-way communication mode, one device (the sender) can send data to another device (the receiver) without expecting a response. This mode is often used for scenarios where one device provides data or commands to another device, such as remote sensor readings or control commands.

Two-way communication: In two-way communication mode, both devices can exchange data bidirectionally. This mode enables a back-and-forth exchange of information between the devices, allowing for more interactive and responsive communication. It's suitable for applications where devices need to send and receive data from each other, such as remote control systems or interactive IoT devices.

Low Latency
As a result of the limited data quantities and its lightweight design ESP-NOW introduces a low latency, which means that devices can quickly exchange data making it suitable for remote controlling other devices, e.g. a radio car.



STATE OF THE ART

PULSE & BLOOM - AN INTERACTIVE BIOFEEDBACK INSTALLATION

Saba Ghole

20 interactive lotus flowers made out of steel and rowlux. Each lotus flower ranges from 8 to 18 feet tall, each of which lights up with your pulse. You and another person can put your hands on a couple of Hamsa hands at the base of the lotus flower and your respective heartbeats will light up the flower.

HEARTSYNC

Nino Basilashvili

Your live heartbeat data and the synchronization patterns of the group will be translated into dynamic visuals. The visuals are complemented by unique low-frequency sounds driven by the participants' heartbeats, blending into a collective symphony. Each person has their own unique heartbeat rhythm. Despite this uniqueness, people's heart rhythms can synchronize.

SIRO

Handuo Zhang.

Described as "A Mirror of the Mind," the project acts as an interactive, neuro-visual experience. It uses a Muse EEG headset to measure brain activity and translates it into generative art.

STATE OF THE ART

MIMU GLOVES

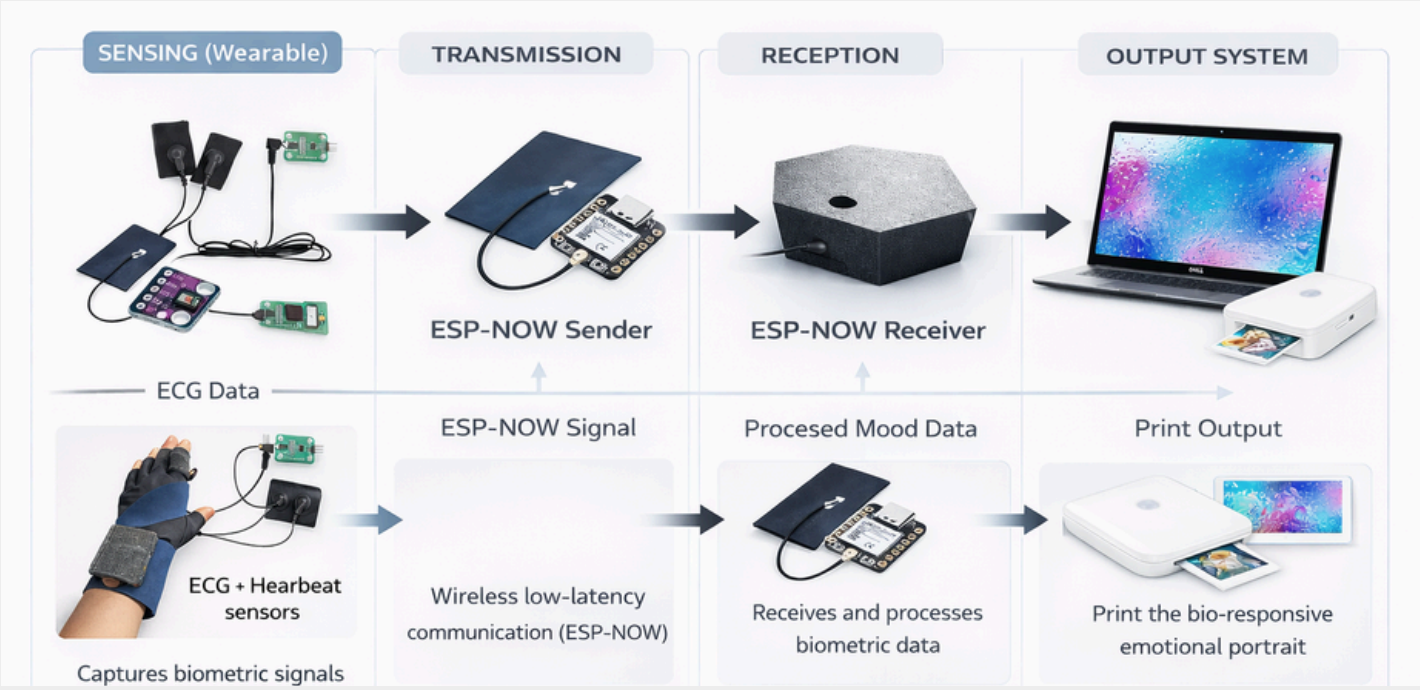
Rachel Freire

MURALS AS MOTION

sensors interactive walls that glow upon contact

PROCESS

System Architecture



THE SYSTEM WORKS IN THREE LAYERS:

Input Layer
- ESP32 (sender inside glove)
- Sensors:
- BPM (heart rate)
- GSR (galvanic skin response)
- Touch detection

Processing Layer
- Processing (Java-based)
- Serial communication (ESP-NOW receiver → USB)
- Emotion classification (received from microcontroller)

Output Layer
- Real-time shader visualization
- AI-generated image (OpenAI API)
- Voice synthesis (TTS API)
- PDF report generation
- Print output

Key Design Features

Real-Time Biofeedback
- Immediate visual response via shaders

Multi-Modal Output
- Visual (shader + AI image)
- Textual (poetry)
- Audio (voice)
- Physical (printed report)

Emotional Mapping
- Physiological → Emotional → Aesthetic

PROCESS

WEARABLE DEVICES

IDEATION

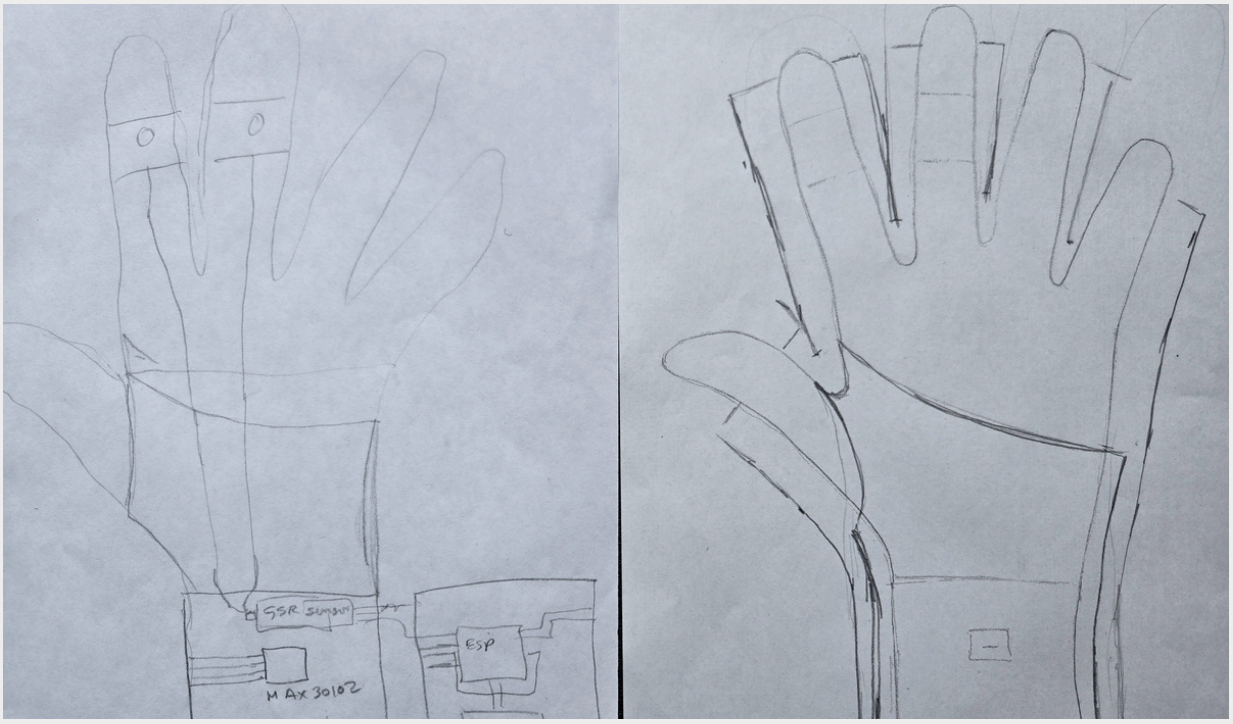
Two Modes of Interaction

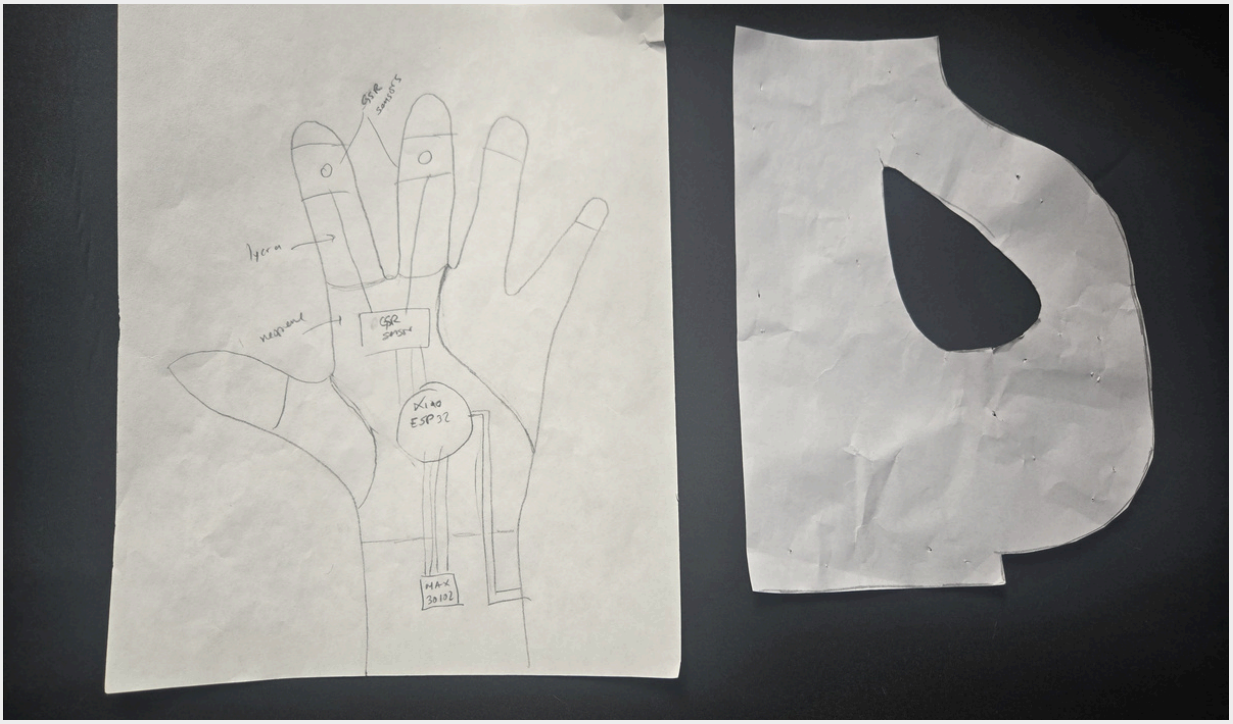
***Exhibition Mode — The Bracelet**
In galleries, participants wear a discreet bracelet that captures biometric signals.As they stand in front of a screen or projection, their internal state generates a unique visual composition in real time.

***Personal Mode — The Glove**
For intimate or customized use, the glove expands the system.

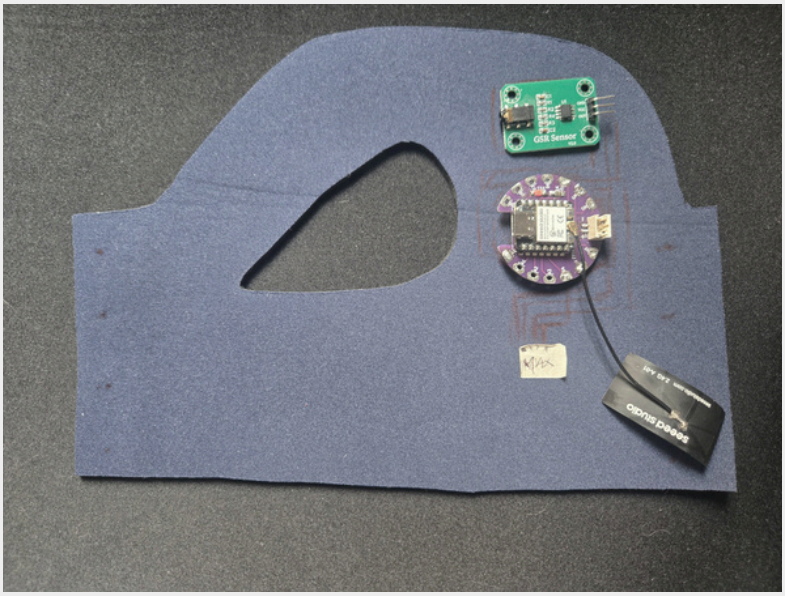
PROCESS

WEARABLE DEVICES
SKETCHES



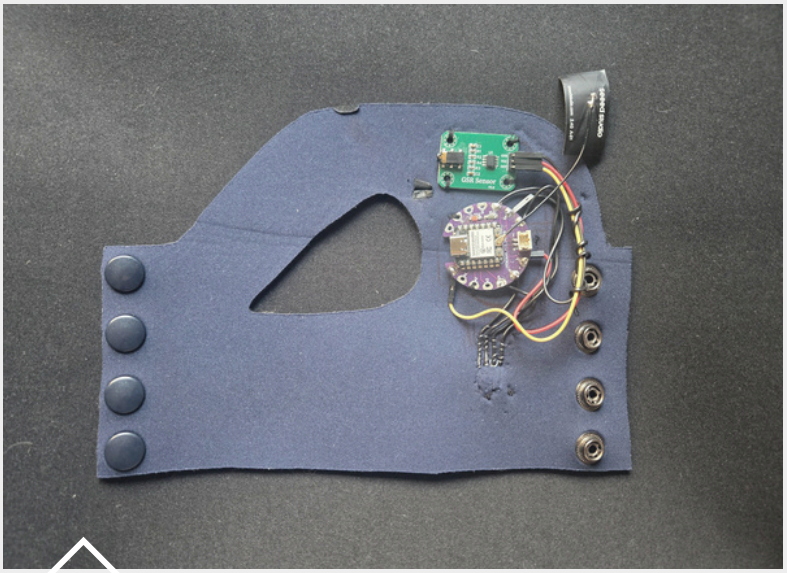


WEARABLE DEVICES-GLOVE



CUTTING THE PATTERN

Using a marker, I drew the pattern on the neoprene and cut it out. Then I also marked the areas where the Xiao ESP and the sensors would be placed. Aditionally, I added the snaps to close the glove



SEWING AND SOLDERING THE SENSORS

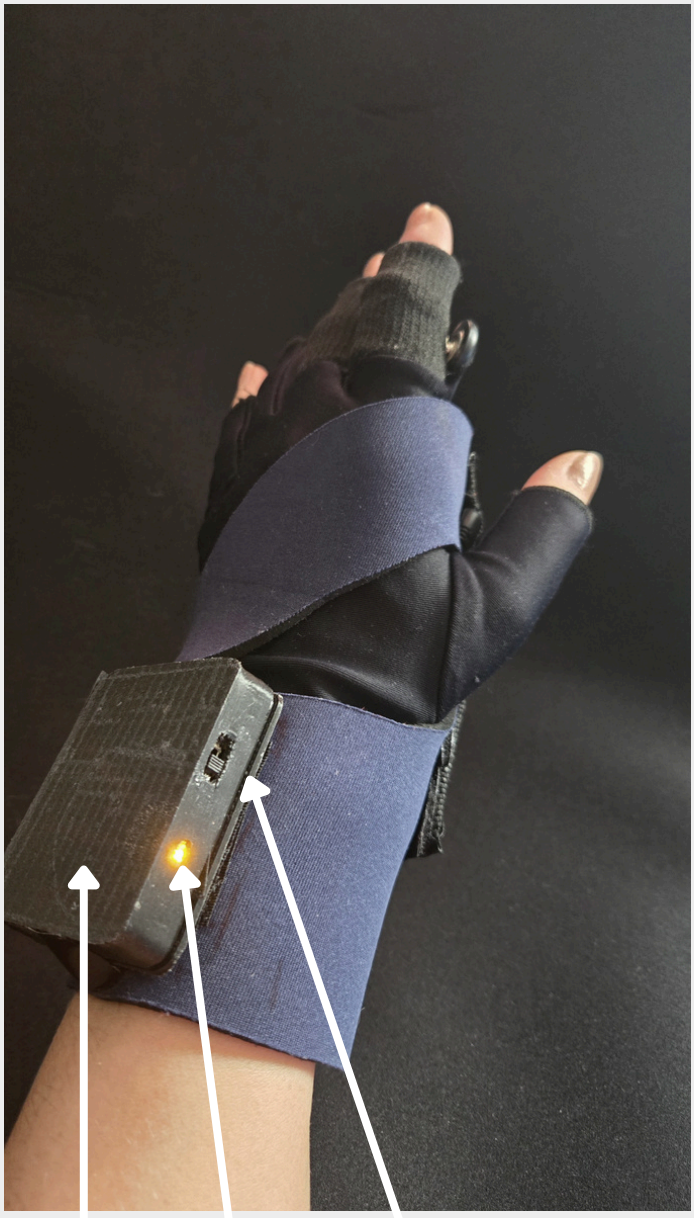
After attempting to sew the sensors to the Xiao ESP using conductive thread—which didn't work due to the small size of the MAX30102 holes—I decided to use 22-gauge wire, which is flexible and thin enough for the sensors. I sewed it onto the neoprene and soldered the ends after passing them through the holes.

SEWING ALL THE PIECES

The final step was to sew the mesh fabric over the Xiao ESP and the GSR sensor, while the heart rate sensor was placed on the inner part of the glove. I also attached the battery case with Velcro after gluing the switch and the LED inside. Then, I sewed the Lycra and neoprene parts of the glove together so they became a single piece. It is important to note that the Xiao antenna must remain outside.

WEARABLE DEVICES-GLOVE

INSIDE THE GLOVE



POWER ON/OFF SWITCH

POWER ON/OFF LED

LIPO BATTERY

GSR SENSOR CONNECTOR

ESP ANTENNA

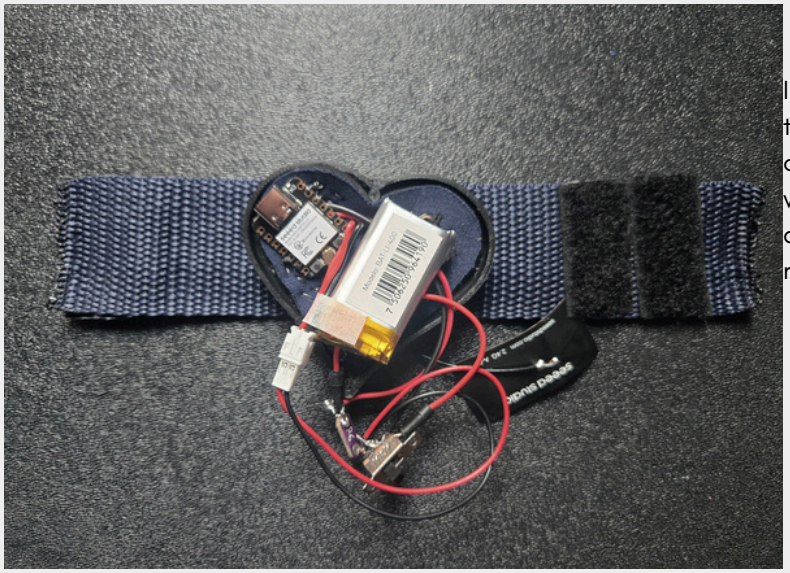
GSR SENSORS

FABRIXIAO BOARD

HEARTBEAT SENSOR

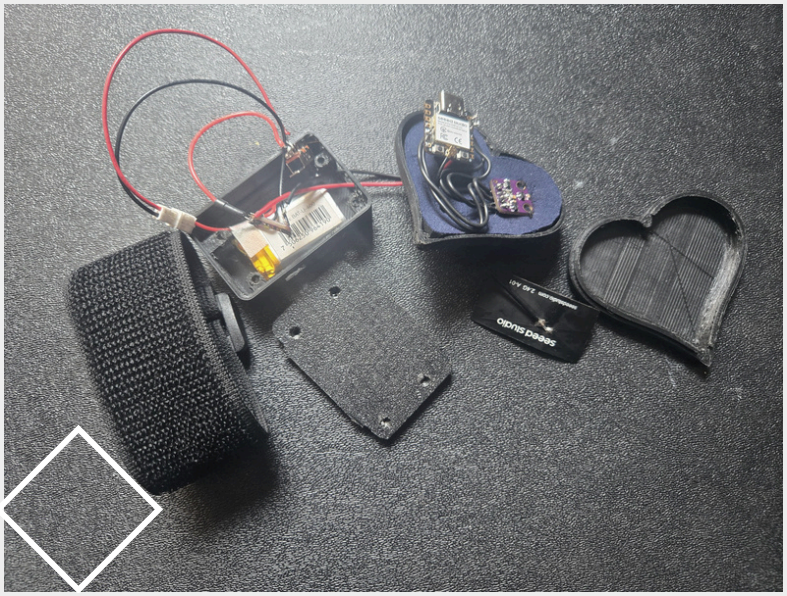
WEARABLE DEVICES

WRISTBAND



FIRST VERSION

I began by 3D printing a heart-shaped case for the ESP and the battery, while sewing the sensor onto the wristband. However, I soon realized there wasn't enough space to accommodate a switch and an LED inside the case, so I decided to redesign it



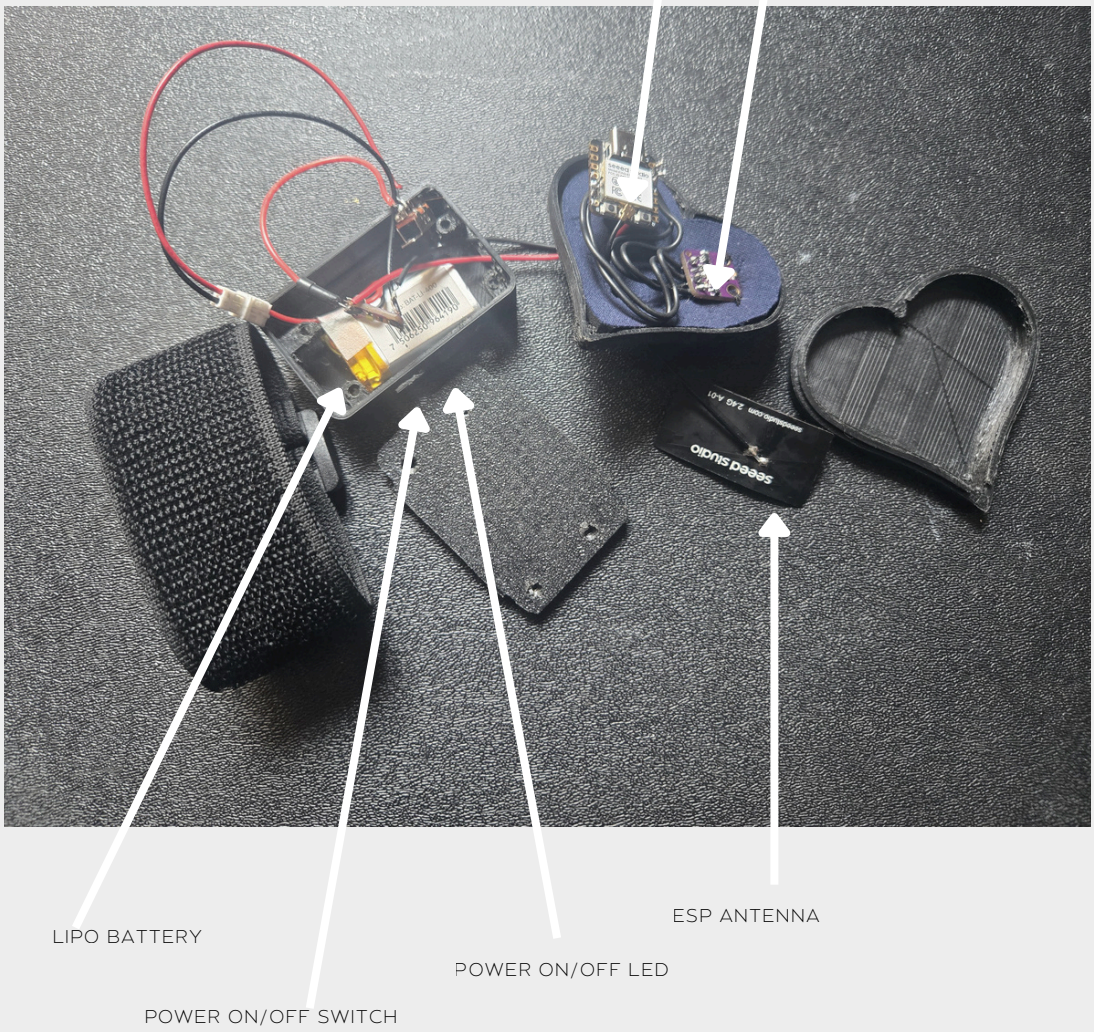
SECOND VERSION

In the second version, I 3D printed a separate case for the battery, LED, and switch, and sewed the ESP and the heart rate sensor onto the neoprene inside the heart-shaped case.

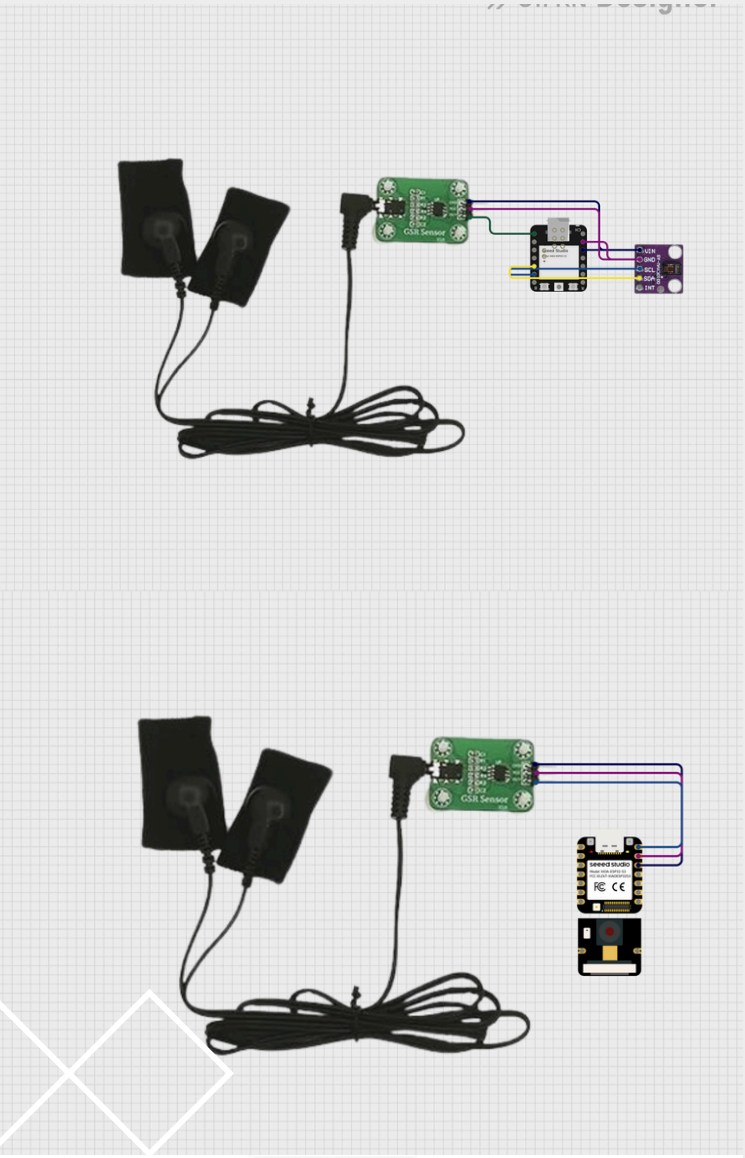
FINAL WRISTBAND

WEARABLE DEVICES

INSIDE THE WRISTBAND



ELECTRONICS

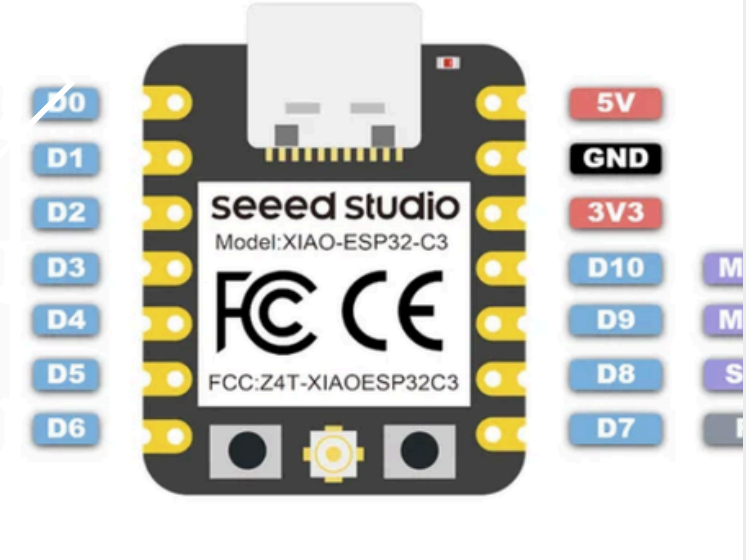


HEARTBEAT SENSOR

To connect the MAX30102 sensor involves an I2C interface, which allows the board to read oxygen saturation and heart rate data. The sensor is primarily designed to be used on parts of the body with thin skin and high blood flow, allowing the red and infrared light to effectively penetrate the tissue and reflect back to the photodetector, such as fingertip, wrist or earlob. It is required to use the SparkFun_MAX3010x_Sensor_Library in the Arduino IDE.

GALVANIC SKIN RESPONSE SENSOR

Connecting a Galvanic Skin Response (GSR) sensor to the Seeed Studio XIAO ESP32S3 Sense involves connecting the sensor's analog output to an available ADC pin on the XIAO board and providing power. Since the XIAO ESP32S3 Sense operates at 3.3V, it is important to power the sensor with 3.3V and read from the appropriate analog pin.

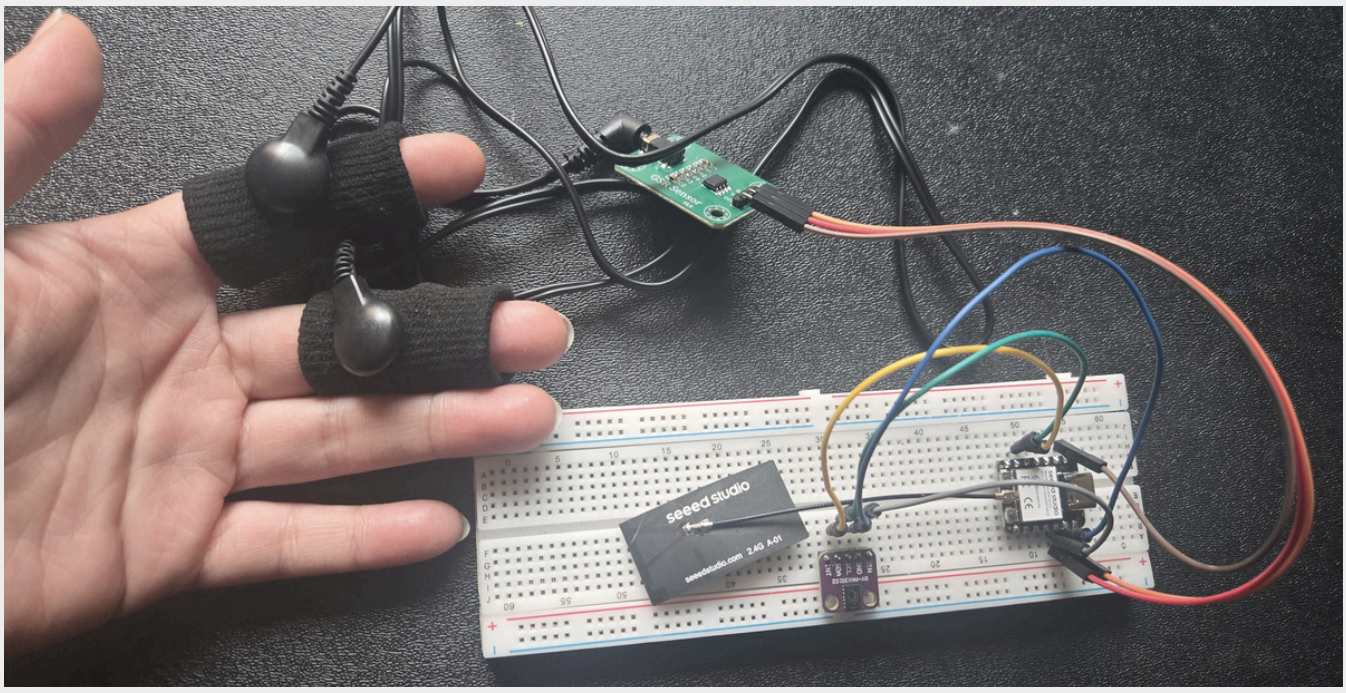


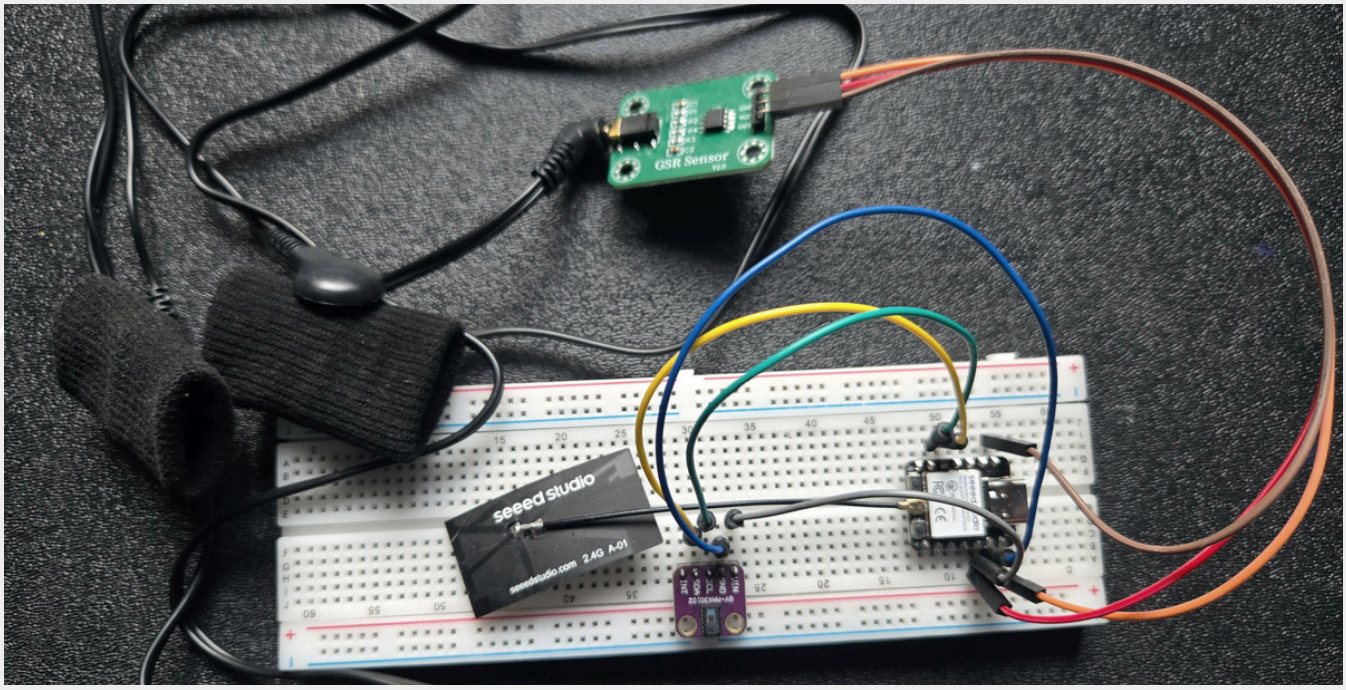
XIAO ESP32-C3

The ESP32-C3 was the best choice due to its integrated antenna for wireless communication in wearables and its strong processing capabilities for handling biometric signals.

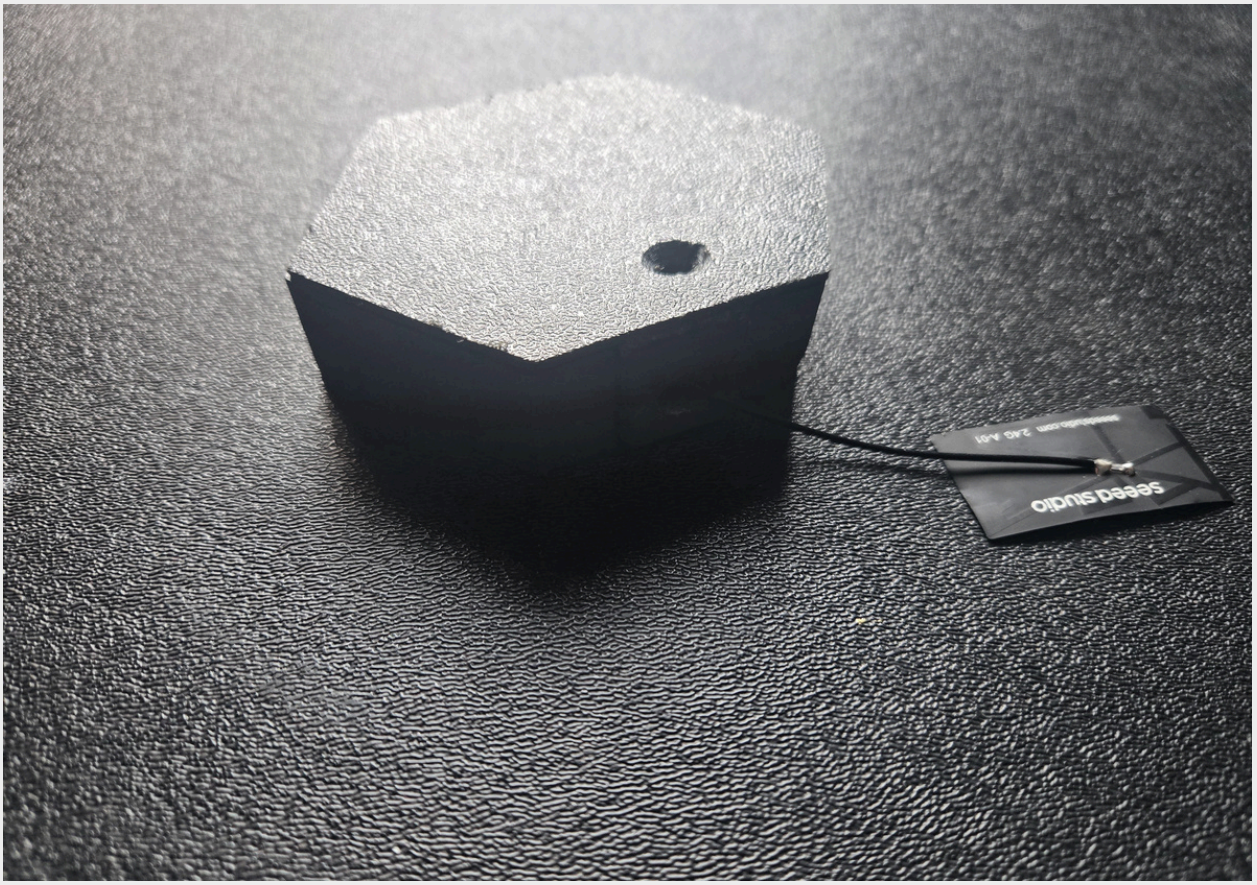
ELECTRONICS

WEARABLE CIRCUIT & SENSORS





ESP-NOW: XIAO ESP32 C3 AS A RECEIVER



PROCESS

Emotional State Model

Emotion	Visual Behavior	BPM Range
CALM	Soft colors, slow movement	BPM < 60
STABLE	Balanced tones, steady motion	60 ≤ BPM < 80
FOCUSED	Sharp, structured visuals	80 ≤ BPM < 95
ELEVATED	Warm colors, energetic motion	95 ≤ BPM ≤ 110
INTENSE	High distortion, fast motion	BPM > 110

AVERAGE ADULT RESTING BMP

55–65 → trained / athletic
60–75 → common adult
70–85 → slightly elevated baseline
80–95 → anxious baseline or caffeine

- BPM: BEATS PER MINUTE

PROCESS

VISUALS GENERATION

DYNAMIC IMAGES

The dynamic images are generated using shaders.
It computes the color of every single pixel on the screen, in real time, for every frame.
So instead of drawing shapes, they are defined by mathematical universe.
It used sine and cosine
Because they:

- oscillate smoothly
- create natural wave motion
- resemble organic systems (water, wind, breathing)

From signal to movement
Instead of showing numbers or graphs, the system transforms the signals into motion.

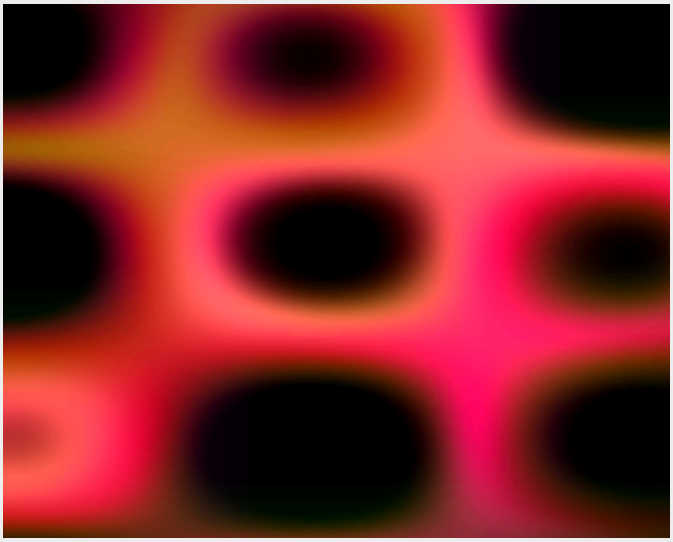
- A slower heartbeat creates gentle, flowing movement
- A faster heartbeat introduces energy and rhythm
-

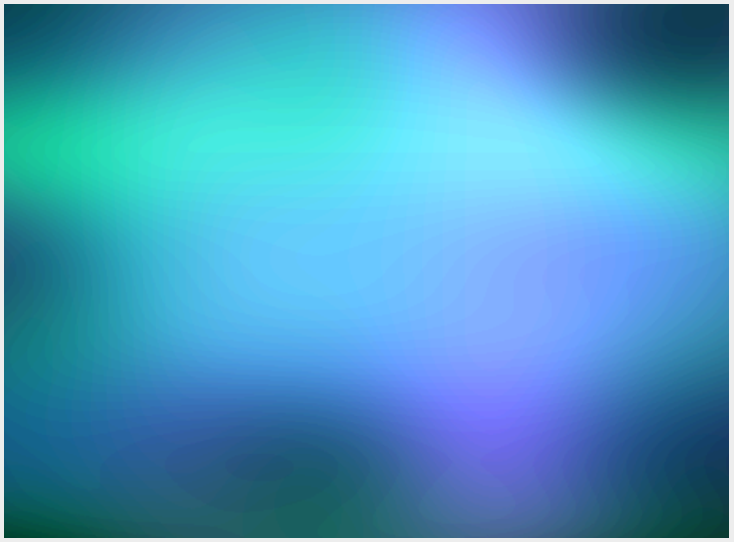
The shader transforms biometric signals into a continuously evolving field of color and motion, where emotion is expressed as distortion, rhythm, and chromatic behavior rather than representation.
Role of GSR:

- Low GSR → small distortion → stable image
- High GSR → strong distortion → chaotic flow

*GSR: Galvanic Skin Response: Skin Conductance

LIVE PAINTINGS





PROCESS

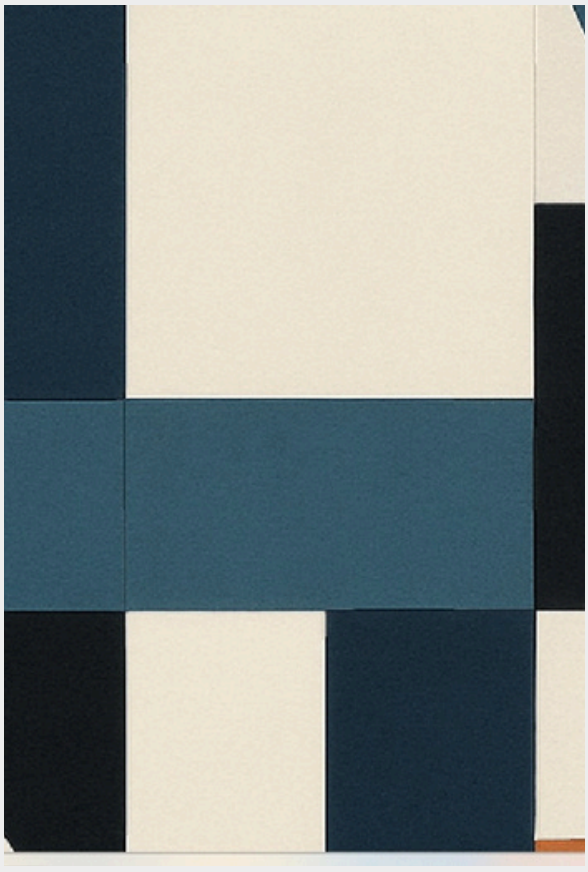
AI IMAGE GENERATION

Process

1. Build prompt from emotion
2. Call OpenAI Image API
3. Receive base64 image
4. Decode and save locally
5. Load into Processing

Prompt Mapping

```
if (emotion.equals("CALM"))
  return "Contemporary abstract painting. Soft translucent layers, muted teal and sand palette.";

if (emotion.equals("STABLE"))
  return "Balanced abstract artwork. Layered geometry, terracotta and deep teal.";

if (emotion.equals("FOCUSED"))
  return "Controlled abstract composition, crisp edges, minimal palette with strong lines.";

if (emotion.equals("ELEVATED"))
  return "Expressive abstract painting, coral and golden tones, dynamic shapes.";

if (emotion.equals("INTENSE"))
  return "Bold dramatic abstract artwork, deep crimson and black contrast, energetic strokes.";

return "Minimal contemporary abstract painting.";
```



PROCESS

POETIC REFLECTION

At the end of the experience, the system generates a short text.
It is not pre-written.
It is composed in that moment, using the biometric data.

During the interaction, the system quietly collects two key elements:

- How long the user stayed (duration)
- How the heart behaved (average rhythm)

From data to language

the system translates the data into poetic language.
For example:

- A slow, steady rhythm becomes calm, distance, or stillness
- A more active rhythm becomes tension, movement, or energy
- A high-intensity state becomes urgency, brightness, or force

A modular poem
The text is built in layers, almost like assembling a sentence from fragments.
Each reflection includes:
An opening
Describes the moment of contact
→ "When your skin met the surface…"
A transition
Suggests something invisible becoming visible
→ "…an unseen language began to unfold."
A duration reference
Anchors the experience in time
→ "For 42 seconds…"
A bodily metaphor
This is the most important part.
It reflects the physiological state

- Calm → "like a distant tide"
- Balanced → "a quiet current beneath the surface"
- Intense → "a signal flare in the dark"

A conceptual closing
Shifts from description to meaning
→ "This is not an image of your face…"
A final line
Leaves a lasting impression
→ "A fragment of you, translated into light."

Even though the system uses predefined phrases, the combination is always different.

It does not diagnose or interpret in a clinical sense
It transforms physiological signals into a poetic narrative.
It gives language to something that normally has none.

Instead of saying:

- "Your heart rate was 78 BPM"

The system says:

- "Your rhythm held tension and intention at once."

PROCESS

BIO-RESPONSIVE EMOTIONAL PORTRAIT

Date: 2026-03-06 18:13

Duration: 69 seconds

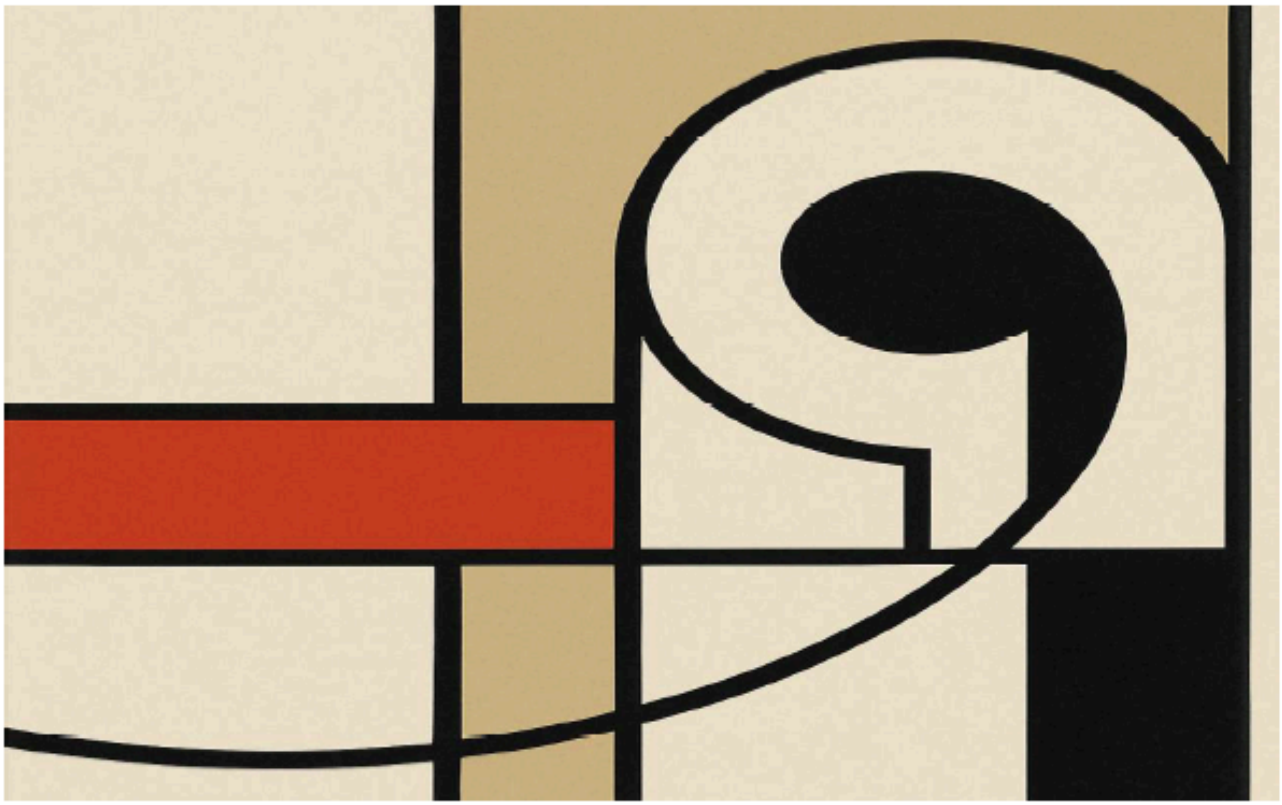
When your skin met the surface,
and the silence filled with rhythm.

For 69 seconds,
your internal rhythms surfaced into light.

Your rhythm held tension and intention at once.

The artwork does not show who you are,
but how you were — in this exact moment.

A fleeting self, captured in rhythm.



PROCESS

VOICE GENERATION

Process

1. Sends reflection text to TTS (Text-to-Speech) API
2. Receives WAV audio
3. Saves locally
4. Plays audio

It doesn't play a prerecorded audio.
It generates a voice in that moment.

- The poetic text is finalized
- The system sends that text to a voice engine
- AI generated voice reads it
- The audio is created instantly
- The system plays it.

PDF REPORT GENERATION

Contents

- Title
- Timestamp
- Duration
- Poetic reflection
- Generated artwork

Output

- A4 PDF (595×842)

Process

- The session ends
- All elements are already available:
- Image
- Text
- Timing
- The system arranges them into a page layout
- A PDF file is created instantly

This step transforms a temporary experience into something to keep.

PRINTING SYSTEM

Process

- Opens system print dialog
- It builds a system command (a Windows instruction)
- It sends that command to the OS (Windows)
- The OS opens the PDF using the default app (e.g., a PDF viewer)
- That app shows the print dialog

PROCESS

PROGRAMMING

Processing was used as the core development environment to handle communication between the hardware components and to coordinate the system's outputs, including real-time visual rendering, AI image generation, voice synthesis, and PDF report creation.

Serial Communication
Handles incoming data from ESP-NOW receiver.

SENSORS AND ESP CODE

The system was programmed in the Arduino IDE to read physiological sensor data, classify it into emotional states, and manage wireless communication via the ESP-NOW protocol, coordinating both the sender and receiver modules.

An ESP32 C3 acts as the sender node in the bio-responsive system.

Its responsibilities are:

- Read physiological data:
  - Heart rate (MAX30102)
  - Skin conductance (GSR)
- Detect user interaction (finger presence)
- Classify emotional states
- Transmit structured data wirelessly using ESP-NOW

Libraries and Dependencies
- WiFi.h → Required for ESP-NOW
- esp_now.h → Wireless peer-to-peer communication
- Wire.h → I2C communication for MAX30102
- MAX30105.h → Sensor control
- heartRate.h → Beat detection algorithm

Emotion Classification

if (bpm < 65 && gsrNorm < 0.25) → CALM
if (bpm < 80 && gsrNorm < 0.45) → STABLE
if (bpm < 95 && gsrNorm < 0.65) → FOCUSED
if (bpm > 110 || gsrNorm > 0.85) → INTENSE
else → ELEVATED

Another ESP32C3 acts as the receiver node in the bio-responsive system.

Its primary role is to:
- Receive biometric data wirelessly via ESP-NOW
- Decode the incoming data structure
- Format the data into readable serial messages
- Transmit the data to Processing via USB

Main Data Stream

```
Serial.print("BPM:");
Serial.print(data.bpm);

Serial.print(",GSR_RAW:");
Serial.print(data.gsrRaw);

Serial.print(",GSR_NORM:");
Serial.print(data.gsrNorm, 3);

Serial.print(",STATE:");
Serial.println(data.emotion);
```

Output Example
BPM:78,GSR_RAW:1320,GSR_NORM:0.45,STATE:FOCUSED

PROCESS

PROGRAMMING
SENSORS AND ESP CODECODE: SENDER

```
#include <WiFi.h>
#include <esp_now.h>
#include <Wire.h>
#include "MAX30105.h"
#include "heartRate.h"

MAX30105 particleSensor;

// ================= GSR =================
const int GSR_PIN = 2;       // safer than A0 on ESP32
const int CALIBRATION_TIME = 8000;

int gsrMin = 4095;
int gsrMax = 0;

bool calibrating = false;
unsigned long calibrationStart;

// ================= HEART RATE =================
float beatsPerMinute;
int beatAvg = 0;

byte rates[4];
byte rateSpot = 0;
long lastBeat = 0;

// ================= TOUCH =================
bool wasFingerDetected = false;

// ================= EMOTION =================
String currentEmotion = "";
String lastEmotion = "";

// ================= ESP-NOW STRUCT =================
typedef struct {
 int bpm;
 int gsrRaw;
 float gsrNorm;
 char emotion[16];
 int touch;
} SensorData;

SensorData data;

// ================= RECEIVER MAC =================
uint8_t receiverMAC[] = {0x80, 0xF1, 0xB2, 0x64, 0x33, 0xD0};

// ======================================================
// SETUP
// ======================================================

void setup() {
 Serial.begin(115200);
 delay(1000);

 Serial.println("Starting Sender...");

 // WIFI MODE
 WiFi.mode(WIFI_STA);
 WiFi.disconnect();

 Serial.print("Sender MAC: ");
 Serial.println(WiFi.macAddress());

 // ESP-NOW INIT
 if (esp_now_init() != ESP_OK) {
  Serial.println("ESP-NOW init failed");
  while (true);
 }

 // REGISTER RECEIVER
 esp_now_peer_info_t peerInfo = {};
 memcpy(peerInfo.peer_addr, receiverMAC, 6);
 peerInfo.channel = 0;
 peerInfo.encrypt = false;

 if (esp_now_add_peer(&peerInfo) != ESP_OK) {
  Serial.println("Failed to add peer");
  while (true);
 }

 Serial.println("ESP-NOW ready.");

 // SENSOR INIT
 if (!particleSensor.begin(Wire, I2C_SPEED_STANDARD)) {
  Serial.println("MAX30102 not found.");
  while (1);
 }

 particleSensor.setup(60, 4, 2, 100, 411, 4096);
 particleSensor.setPulseAmplitudeRed(0x3F);
 particleSensor.setPulseAmplitudeIR(0x3F);

 pinMode(GSR_PIN, INPUT);
 analogSetAttenuation(ADC_11db);

 Serial.println("System Ready. Place finger.");
}
```

PROCESS

PROGRAMMING
SENSORS AND ESP CODECODE: SENDER

```
// ================ HEART RATE ================
 if (checkForBeat(irValue)) {

  long delta = millis() - lastBeat;
  lastBeat = millis();

  beatsPerMinute = 60 / (delta / 1000.0);

  if (beatsPerMinute > 40 && beatsPerMinute < 180) {

   rates[rateSpot++] = (byte)beatsPerMinute;
   rateSpot %= 4;

   beatAvg = 0;
   for (byte i = 0; i < 4; i++)
    beatAvg += rates[i];

   beatAvg /= 4;
  }
 }

 // Optional stability improvement
 if (irValue < 50000) {
  beatAvg = 0;
 }

 // ================ EMOTION ================
 currentEmotion = classifyEmotion(beatAvg, gsrNormalized);

 if (currentEmotion != lastEmotion) {
  Serial.print("Emotion: ");
  Serial.println(currentEmotion);
  lastEmotion = currentEmotion;
 }

 // ================ FILL STRUCT ================
 data.bpm = beatAvg;
 data.gsrRaw = gsrValue;
 data.gsrNorm = gsrNormalized;
 data.touch = fingerDetected ? 1 : 0;

 strncpy(data.emotion, currentEmotion.c_str(), sizeof(data.emotion));

 // ================ SEND ================
 sendData();

 delay(30);
}

// ======================================================
// SEND FUNCTION
// ======================================================

void sendData() {

 esp_err_t result = esp_now_send(receiverMAC, (uint8_t *)&data, sizeof(data));

 if (result == ESP_OK) {
  Serial.println("Sent");
 } else {
  Serial.println("Send Error");
 }
}

// ======================================================
// EMOTION CLASSIFICATION
// ======================================================

String classifyEmotion(int bpm, float gsrNorm) {

 if (bpm < 65 && gsrNorm < 0.25)
  return "CALM";

 if (bpm < 80 && gsrNorm < 0.45)
  return "STABLE";

 if (bpm < 95 && gsrNorm < 0.65)
  return "FOCUSED";

 if (bpm > 110 || gsrNorm > 0.85)
  return "INTENSE";

 return "ELEVATED";
}
```

PROCESS

PROGRAMMING

SENSORS AND ESP CODECODE: RECEIVER

```
#include <WiFi.h>
#include <esp_now.h>

// ================ DATA STRUCT ================
typedef struct {
 int bpm;
 int gsrRaw;
 float gsrNorm;
 char emotion[16];
 int touch;
} SensorData;

SensorData data;


// ======================================================
void OnDataRecv(const esp_now_recv_info *info, const uint8_t *incomingData, int len) {

 memcpy(&data, incomingData, sizeof(data));

 // ================ SEND CLEAN DATA TO PROCESSING ================
 Serial.print("BPM:");
 Serial.print(data.bpm);

 Serial.print(",GSR_RAW:");
 Serial.print(data.gsrRaw);

 Serial.print(",GSR_NORM:");
 Serial.print(data.gsrNorm, 3);

 Serial.print(",STATE:");
 Serial.println(data.emotion);

 // TOUCH separately (important for Processing logic)
 Serial.print("TOUCH:");
 Serial.println(data.touch);
}

// ======================================================
// SETUP
// ======================================================
void setup() {

 Serial.begin(115200);
 delay(1000);

 Serial.println("Starting ESP-NOW Receiver...");

 // ================ WIFI MODE ================
 WiFi.mode(WIFI_STA);
 WiFi.disconnect();

 Serial.print("Receiver MAC: ");
 Serial.println(WiFi.macAddress());

 // ================ INIT ESP-NOW ================
 if (esp_now_init() != ESP_OK) {
  Serial.println("ESP-NOW init failed");
  while (true);
 }

 // ================ REGISTER CALLBACK ================
 esp_now_register_recv_cb(OnDataRecv);

 Serial.println("ESP-NOW Receiver Ready.");
}

// ======================================================
// LOOP
// ======================================================
void loop() {
 // Nothing needed here
}
```

PROCESS

FULL CODE

```
import processing.serial.*;
import processing.data.*;
import java.io.*;
import java.net.*;
import java.util.Base64;
import processing.sound.*;
import processing.video.*;
import processing.pdf.*;

// ------------------------------------
// SESSION TRACKING
// ------------------------------------

ArrayList<String> emotionTimeline = new ArrayList<String>();
int sessionStartTime = 0;
int sessionEndTime = 0;

int bpmSum = 0;
int bpmCount = 0;

// ------------------------------------------------
// SYSTEM STATE
// ------------------------------------------------

processing.serial.Serial myPort;

String OPENAI_API_KEY = "MyKEY";

// ------------------------------------------------
// INTRO VIDEO + SOUND
// ------------------------------------------------

Movie introVideo;
SoundFile heartbeat;

float introAlpha = 255;
boolean introFinished = false;
```

PROCESS

FULL CODE

```
// ------------------------------------------------
// PARTICIPANT DETECTION
// ------------------------------------------------

boolean isTouching = false;
boolean wasTouching = false;

// ------------------------------------------------
// IMAGE SYSTEM
// ------------------------------------------------

PImage generatedImage;
boolean loading = false;

// ================================================
// SETUP
// ================================================

void setup() {

 fullScreen(P2D);
 pixelDensity(1);

 introVideo = new Movie(this, "BrushStrokes.mp4");
 introVideo.loop();
 introVideo.volume(0);

 heartbeat = new SoundFile(this, "cinematic-heartbeat.wav");
 heartbeat.loop();
 heartbeat.amp(0.5);

 myPort = new processing.serial.Serial(this, "COM4", 115200);
 myPort.bufferUntil('\n');

 println("System ready.");
}
```

PROCESS

FULL CODE

```
// ================================================
// VIDEO EVENT
// ================================================

void movieEvent(Movie m) {
 m.read();
}

// ================================================
// DRAW LOOP
// ================================================

void draw() {

 background(0);
 imageMode(CENTER);

 if (generatedImage == null) {
  image(introVideo, width/2, height/2, width, height);
  return;
 }

 if (!introFinished) {

  if (introAlpha > 0) {

   introVideo.pause();

   tint(255, introAlpha);
   image(introVideo, width/2, height/2, width, height);

   tint(255, 255 - introAlpha);
   image(generatedImage, width/2, height/2, width, height);

   introAlpha -= 4;

  } else {

   introAlpha = 0;
   introFinished = true;

   introVideo.stop();
   heartbeat.stop();
  }

 } else {

  tint(255, 255);
  image(generatedImage, width/2, height/2, width, height);
 }
}
```

PROCESS

FULL CODE

```
// ---------------- BIO DATA ----------------

 if (incoming.startsWith("BPM:")) {

  try {

  String[] parts = split(incoming, ",");

  currentBPM = int(split(parts[0], ":")[1]);

  float newGSR = float(split(parts[2], ":")[1]);
  smoothGSR = lerp(smoothGSR, newGSR, 0.1);
  currentGSR = smoothGSR;

  String emotion = split(parts[3], ":")[1];

  bpmSum += currentBPM;
  bpmCount++;

  println("Parsed → BPM: " + currentBPM +
      " | GSR: " + currentGSR +
      " | STATE: " + emotion);

  if (!emotion.equals(lastEmotion) && !loading && isTouching) {

   lastEmotion = emotion;
   currentEmotion = emotion;

   introAlpha = 255;

   thread("generateImage");
  }

  } catch (Exception e) {

  println("Parse error.");
  e.printStackTrace();
  }
 }
}

// =======================================================
// IMAGE GENERATION
// =======================================================

void generateImage() {

 loading = true;

 try {

  println("Generating image for: " + currentEmotion);

  String prompt = buildPrompt(currentEmotion);

  JSONObject body = new JSONObject();
  body.setString("model", "gpt-image-1");
  body.setString("prompt", prompt);
  body.setString("size", "1024x1024");

  URL url = new URL("https://api.openai.com/v1/images/generations");
  HttpURLConnection conn = (HttpURLConnection) url.openConnection();

  conn.setRequestMethod("POST");
  conn.setRequestProperty("Content-Type", "application/json");
  conn.setRequestProperty("Authorization", "Bearer " + OPENAI_API_KEY);
  conn.setDoOutput(true);

  OutputStream os = conn.getOutputStream();
  os.write(body.toString().getBytes("UTF-8"));
  os.close();
```

PROCESS

FULL CODE

```
BufferedReader reader = new BufferedReader(
  new InputStreamReader(conn.getInputStream())
 );

 String response = "";
 String line;

 while ((line = reader.readLine()) != null) {
  response += line;
 }

 reader.close();

 JSONObject json = parseJSONObject(response);
 JSONArray data = json.getJSONArray("data");

 String base64Image = data.getJSONObject(0).getString("b64_json");
 byte[] imageBytes = Base64.getDecoder().decode(base64Image);

 String imagePath = sketchPath("generated.png");
 FileOutputStream fos = new FileOutputStream(imagePath);
 fos.write(imageBytes);
 fos.close();

 generatedImage = loadImage("generated.png");
 generatedImage.resize(width, height);

 println("Image ready.");

 } catch (Exception e) {
 e.printStackTrace();
 }

 loading = false;
}

// ======================================================
// EMOTION → PROMPT
// ======================================================

String buildPrompt(String emotion) {

 if (emotion.equals("CALM"))
  return "Contemporary abstract painting. Soft translucent layers, muted teal and sand palette.";

 if (emotion.equals("STABLE"))
  return "Balanced abstract artwork. Layered geometry, terracotta and deep teal.";

 if (emotion.equals("FOCUSED"))
  return "Controlled abstract composition, crisp edges, minimal palette with strong lines.";

 if (emotion.equals("ELEVATED"))
  return "Expressive abstract painting, coral and golden tones, dynamic shapes.";

 if (emotion.equals("INTENSE"))
  return "Bold dramatic abstract artwork, deep crimson and black contrast, energetic strokes.";

 return "Minimal contemporary abstract painting.";
}
```

PROCESS

FULL CODE

```
// =====================================================
// POETIC REFLECTION
// =====================================================

String buildPoeticReflection(int avgBPM, int duration) {

 // ----------------------------------
 // VARIATION POOLS
 // ----------------------------------

 String[] openings = {
  "You placed your hand upon the sensor,",
  "When your skin met the surface,",
  "At the moment of contact,",
  "Your touch awakened the system,"
 };

 String[] translations = {
  "and your body began to translate itself.",
  "and an unseen language started to unfold.",
  "and your internal signals rose into visibility.",
  "and the silence filled with rhythm."
 };

 String[] durationLines = {
  "For " + duration + " seconds,",
  "Across " + duration + " measured seconds,",
  "During " + duration + " suspended seconds,"
 };

 String[] calmMetaphors = {
  "Your pulse moved like a distant tide — steady and inward.",
  "Your rhythm settled like evening light over still water.",
  "Your heartbeat drifted, unhurried and reflective."
 };

 String[] midMetaphors = {
  "Your heart carried a quiet urgency — present and aware.",
  "A subtle current ran beneath the surface of your skin.",
  "Your rhythm held tension and intention at once."
 };

 String[] intenseMetaphors = {
  "Your pulse surged with intensity — electric and immediate.",
  "Energy rose sharply, bright beneath your skin.",
  "Your heartbeat struck like a signal flare in the dark."
 };

 String[] closings = {
  "This is not an image of your face,\nbut a portrait of your physiological moment.",
  "What emerged was not your likeness,\nbut the trace of your internal weather.",
  "The artwork does not show who you are,\nbut how you were — in this exact moment."
 };

 String[] finalLines = {
  "A record written in signal and breath.",
  "A fleeting self, captured in rhythm.",
  "A fragment of you, translated into light."
 };
```

PROCESS

FULL CODE

```
// ====================================================
// GENERATE PDF
// ====================================================

void generateSessionReport() {

 int durationSeconds = (sessionEndTime - sessionStartTime) / 1000;
 int avgBPM = bpmCount > 0 ? bpmSum / bpmCount : 0;

 String timestamp =
  year() + "-" + nf(month(),2) + "-" + nf(day(),2) +
  " " + nf(hour(),2) + ":" + nf(minute(),2);

 String reflection = buildPoeticReflection(avgBPM, durationSeconds);

 String filename = sketchPath("SessionReport.pdf");

 PGraphics pdf = createGraphics(595, 842, PDF, filename);
 pdf.beginDraw();

 pdf.background(255);
 pdf.fill(0);
 pdf.textAlign(LEFT);

 pdf.textSize(22);
 pdf.text("BIO-RESPONSIVE EMOTIONAL PORTRAIT", 50, 80);

 pdf.textSize(12);
 pdf.text("Date: " + timestamp, 50, 120);
 pdf.text("Duration: " + durationSeconds + " seconds", 50, 140);

 pdf.textSize(13);
 pdf.text(reflection, 50, 200, 495, 300);

 if (generatedImage != null) {

  float imgWidth = 495;
  float aspect = (float)generatedImage.height / generatedImage.width;
  float imgHeight = imgWidth * aspect;

  pdf.image(generatedImage, 50, 450, imgWidth, imgHeight);
 }

 pdf.endDraw();
 pdf.dispose();

 println("Poetic PDF generated.");
}

// ====================================================
// PRINT PDF
// ====================================================

void printSessionReport() {

 try {

  String pdfPath = sketchPath("SessionReport.pdf");
  String command = "rundll32 url.dll,FileProtocolHandler \"" + pdfPath + "\"";
  Runtime.getRuntime().exec(command);

  println("Print dialog opened.");

 } catch (Exception e) {
  e.printStackTrace();
 }
}
```

OUTCOME

GLOVE

OUTCOME

WRISTBAND

OUTCOME

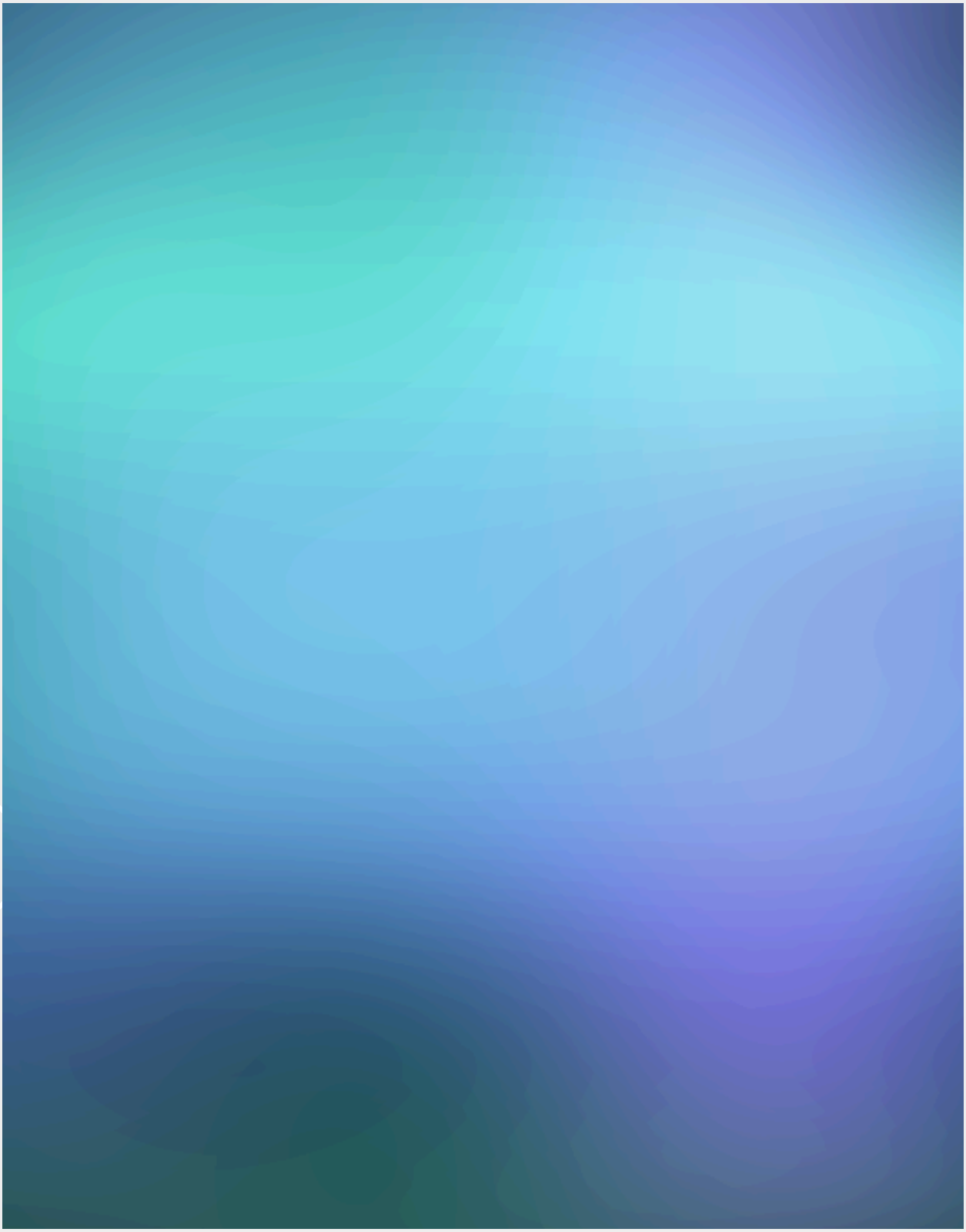
LIVE PAINTINGS

OUTCOME

LIVE PAINTINGS

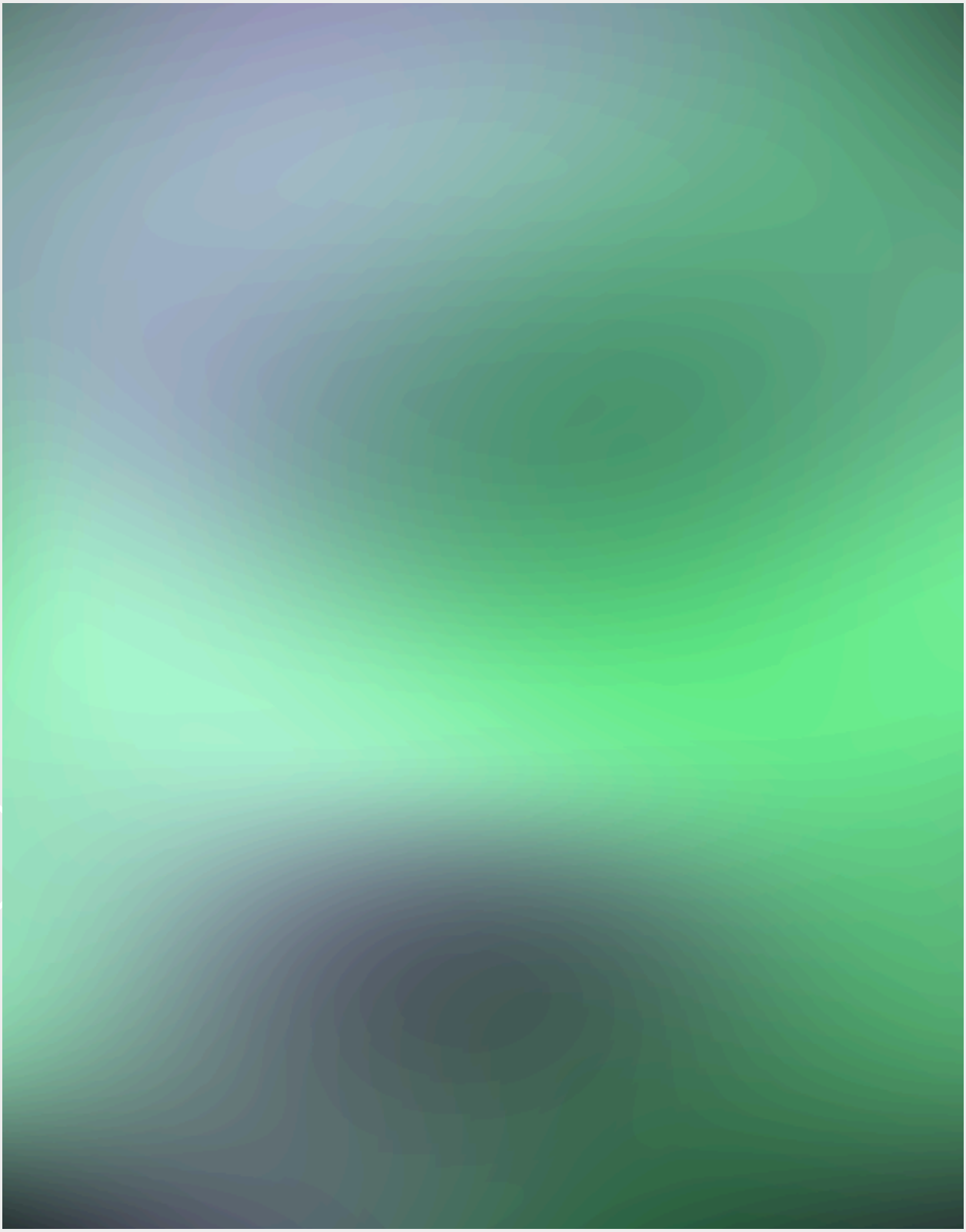
OUTCOME

LIVE PAINTINGS



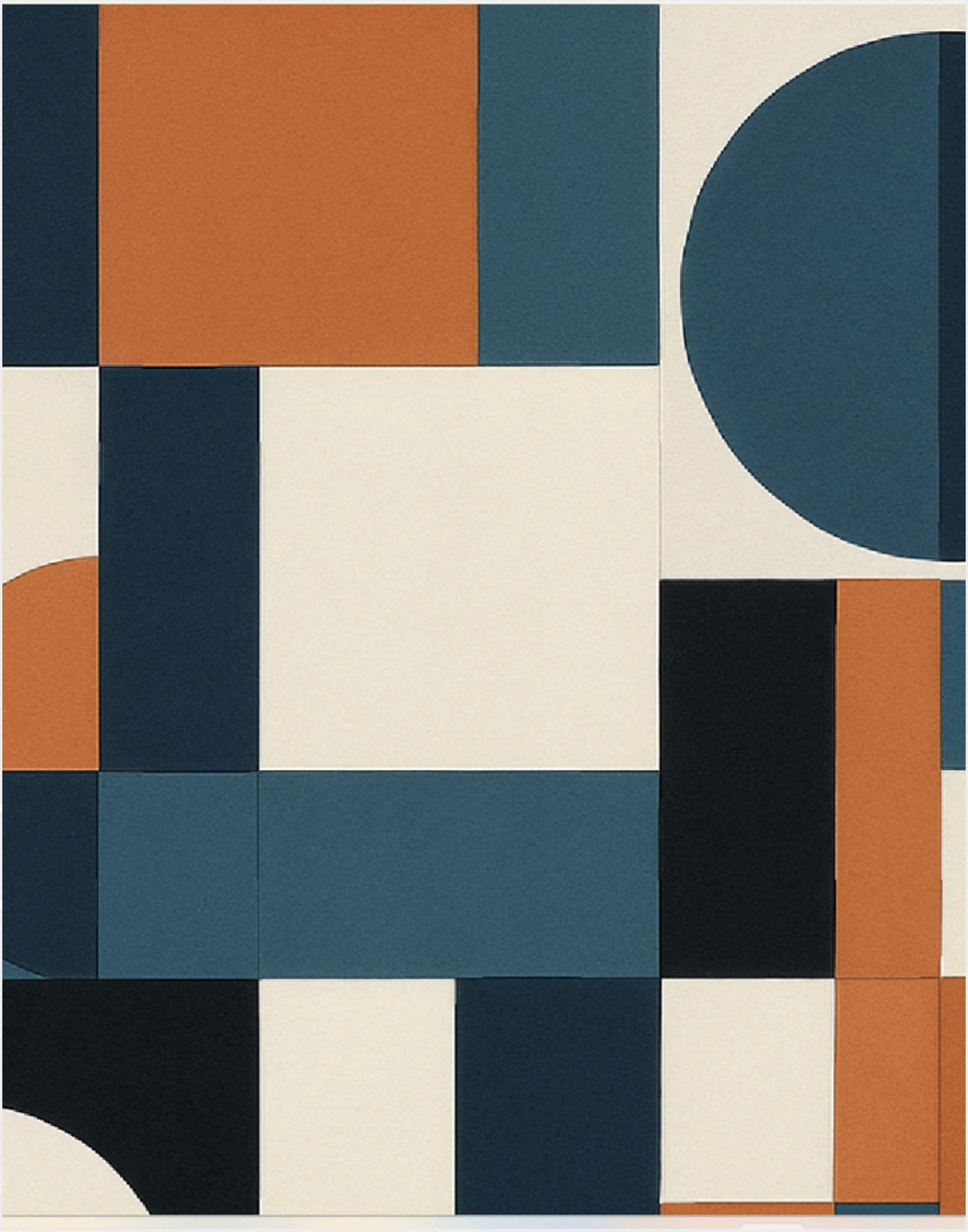
OUTCOME

LIVE PAINTINGS



OUTCOME

STILL IMAGES

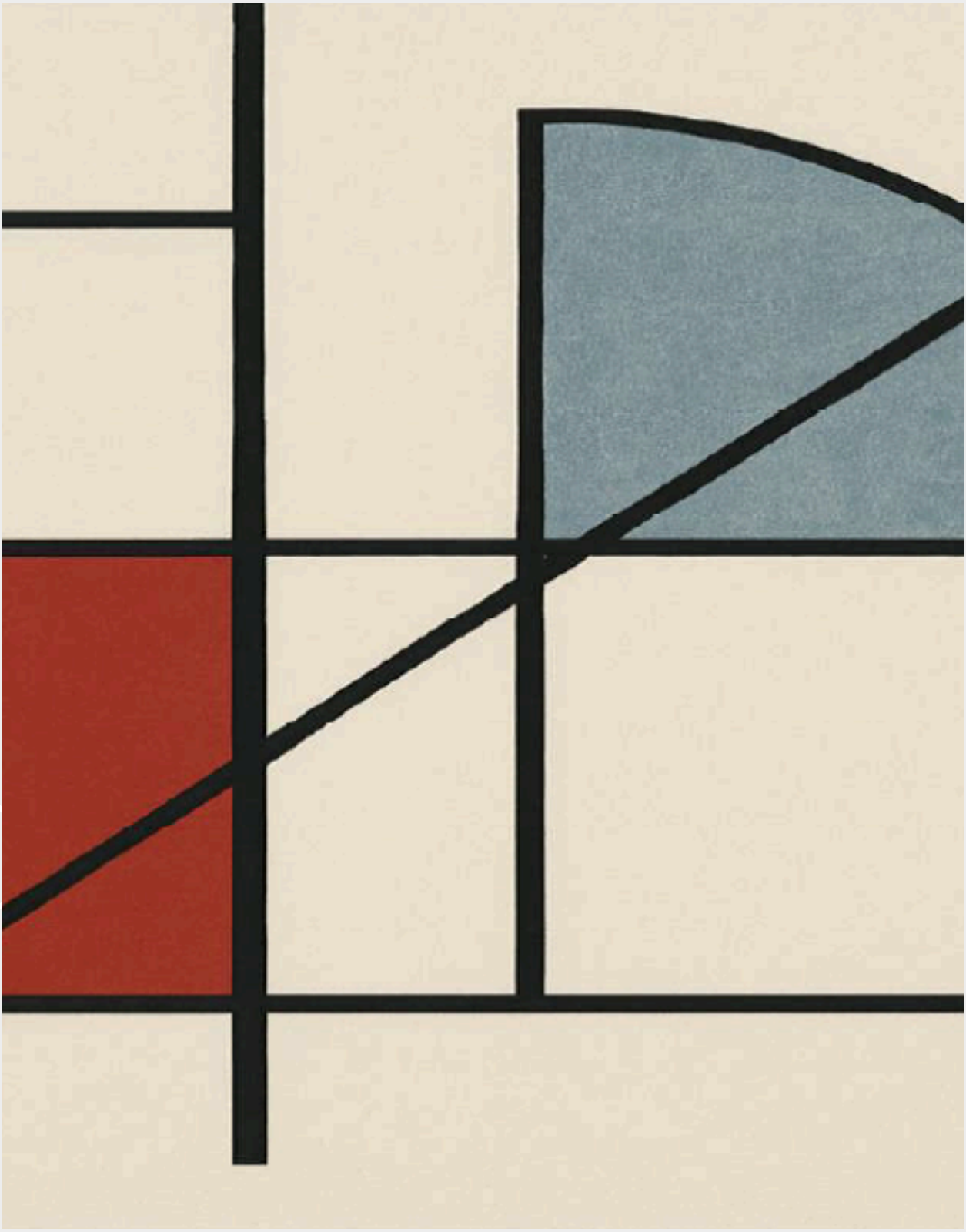


OUTCOME

STILL IMAGES

OUTCOME

STILL IMAGES



OUTCOME

STILL IMAGES

OUTCOME

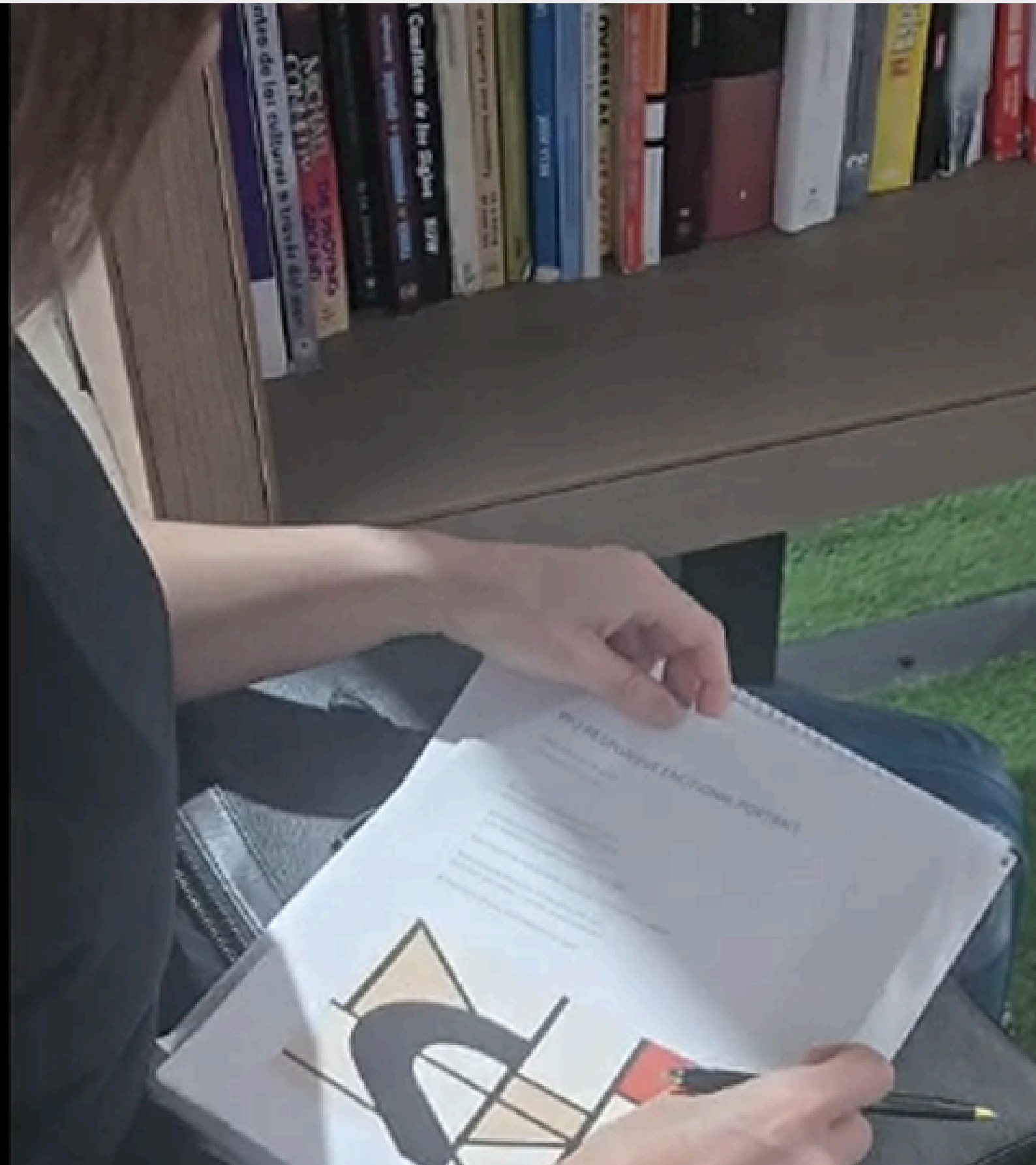
USER EXPERIENCE

OUTCOME

USER EXPERIENCE

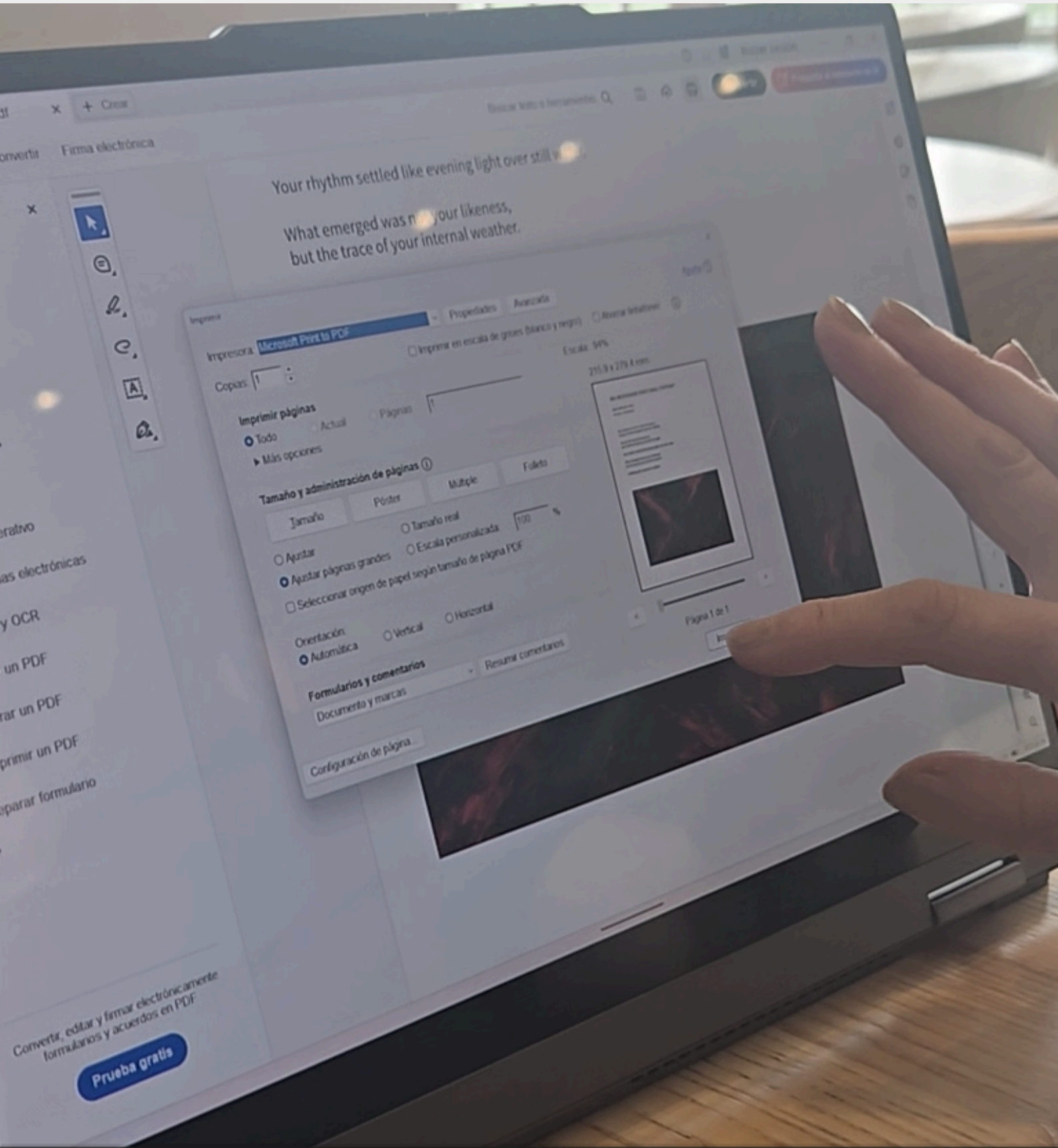
OUTCOME

USER EXPERIENCE



OUTCOME

USER EXPERIENCE



BIBLIOGRAPHY

01

EMOTIONAL STATES, HEART RATE AND GALVANIC SKIN RESPONSE

Does your heart sense your emotional state? https://www.today.com/health/does-your-heart-sense-your-emotional-state-2d80555354

Analysis of Heart Rate Variability and Implication of Different Factors on Heart Rate Variability. https://pmc.ncbi.nlm.nih.gov/articles/PMC8950456/#:~:text=3.5.&text=It%20has%20been%20observed%20that,of%20parasympathetic%20activation%20%5B69%5D.

How Do Amusement, Anger and Fear Influence Heart Rate and Heart Rate Variability? https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2019.01131/full

Galvanic Skin Response (GSR): The Complete Pocket Guide. https://imotions.com/blog/learning/research-fundamentals/galvanic-skin-response/?srsltid=AfmBOoo4DvbwF3QTWQUzzeiyHuaDqJhXfMOBwm3SoT0_vmxOHaOyi6DS

Could the Color You Wear Affect Your Heart Rate? https://medium.com/design-bootcamp/could-the-color-you-wear-affect-your-heart-rate-4801b4ede55c

Heartbeat Art. Claudia Arozqueta.

02

THEORY OF COLOR

The Psychology of Color: How the Shades Around You Impact Your Emotions. https://insightspsychology.org/psychology-of-color-emotional-impact/

Color me impressed! Psychology research links colors and emotions for over a century. https://featuredcontent.psychonomic.org/color-me-impressed-psychology-research-links-colors-and-emotions-for-over-a-century/

The Effect of Colour on Mood |How Colours Shape Our Emotions | MindTribe Live Workshop https://www.youtube.com/watch?v=wPlUrPQpMS4&t=521s

Feeling Colour: Chromatic Embodiment in Film Culture, 1950s-1960s. Bregt Lameris.

Exploring Color-Based Emotion Visualization for the Design of Affective Agents: A Pilot Study in a Special Needs School. Sunkyoung Kim

03

ESP-NOW PROTOCOL

ESP-NOW A wireless communication protocol. https://www.espressif.com/en/solutions/low-power-solutions/esp-now

Device to Device Communication with ESP-NOW. https://docs.arduino.cc/tutorials/nano-esp32/esp-now/

MOOD PAINT

FABRICADEMY APRIL 2026

EMMA YAÑEZ