Progress. Week 4

Type	Progress
Date	October 12, 2026
Purpose	Progression

Plan after the feedback:

After the feedback, I've realised image lacks more measurable and prolonged lifecycle, or at least, it was obscured through the prototype. The processes that the image undergoes within it's lifecycle could be transformed into simple but different interactions to highlight the process of care and nurturing. Simplifying all the interactions into simple variables would help in making the process more intentional and deliberate. All the things we do to it, how we edit and shape the image, focus on those relationships between the image and the person and make this connection visible and almost tangible.

The image that will be nurtured will be taken from a personal archive and sent via Web, as a way of transferring the image into a second vessel and, therefore, underscoring it's temporality and short lifespan. I decided to go away from the digital gardening concept, as it was perceived as something that could potentially lead away from the original idea and take away from it's complexity.

So, in that case the lifecycle would be an image's resolution, where the starting point is it's initial original resolution and the death is its loss. The interaction would be through touch(conductive tape and tilt sensor/maybe remove it), sound, buttons.

imagotchi?

Emotional aspect (emotional ones)

Sound to make user hold it

Attention button - click it to preserve resolution. (maybe somewhere in the middle)

Formal changes (physical processes)

Crop image -

Edit it’s values - brightness, like filter maybe

Share image - deteriorate it’s quality by expense of sharing it (is it formal change?)

Message will be communicated on top of the image.

Is the process happening within a day/week? Define a timeframe

How it’s deteriorating?

Core principle

The image has **one life variable: its resolution.** Everything that happens to it is expressed as pixels kept or pixels lost, so the lifecycle becomes measurable and visible at every moment. A small readout on top of the image always shows its current size, for example `238 × 317`.

Resolution never goes back up. Every loss is written into the stored file, so it's permanent, the same way a lossy save can't be undone. Care doesn't heal the image; it **preserves** it. That keeps the piece honest about finitude and makes neglect matter. (If you want the "+1 pixel" in your notes to mean actual recovery, that's a variant, but I think preservation is the stronger idea.)

Lifecycle

Stage	What happens
Arrival	The image is sent from a personal archive over the web, by scanning the QR code on the screen. The transfer is the first loss: the readout shows

Stage	What happens
	`arrived at 240 × 320 (from 4032 × 3024)`.
Life	It slowly loses resolution, one pixel of width at a time, getting blockier.
1 pixel	A single field of colour, the average of the whole image. It stays like this for one final day.
Death	The data is erased. The screen goes dark, and then the empty vessel asks for a new image.

Timeframe

- **Natural decay: −1 pixel per hour without care.** A completely neglected image dies in about **10 days** (240 hours). Careful attention can stretch that to weeks.
- **Time passes even when it's switched off**, using your DS3231 clock module. When you turn it on, the losses from the time away are applied at once: `you were gone 6 hours. I lost 6 pixels.`
- **Crit or demo mode**: one hour becomes 5 seconds, so a whole life plays out in about 20 minutes.

Interactions

Care (emotional): these preserve resolution

Action	Input	Effect
Holding	Touch zone on the frame	Time stops while it's held. Being with it is the most basic care.
Answering the call	It makes a sound about 3 times a day, at random hours. Hold it within 10 minutes.	Answered: no loss for the next 4 hours. Missed: **−10 pixels** at once.
Attention	Button, once a day	It sets the current resolution as a floor for 24 hours, so time can't take pixels below it. The image asks for this explicitly at the **middle of its life** (120 px wide): `I am half of what I was.`

Formal changes (physical processes): these cost resolution

These are choices the person makes to shape the image, and each one is paid for in pixels. Editing becomes wear.

Action	Input	Effect
Crop	Tilt towards an edge, then press the edit button	That edge is cut off. The pixels are gone, and the rest is shown larger and coarser.
Edit values	Long-press the edit button to cycle brightness, contrast or a filter	Each saved edit costs **−2 pixels**, like generation loss from re-saving a file.
Share	Someone opens the device's share page on their phone	They receive a copy at **half** the current resolution (the poor copy), and the original loses **−10 pixels**. Circulation wears the image down.

Sleep and waking

By default it sleeps. The screen is dark and the ESP32 is in deep sleep, using almost no power. The image exists, but you can't see it.

It wakes in two ways:

- **You wake it**: holding the touch zone or pressing a button. The ESP32 can wake from deep sleep on touch alone, so simply picking it up brings the image back.
- **It wakes itself to ask for attention**, about 3 times a day at random hours.

When it wakes itself:

1. The screen lights up **dimly**, showing the image at **half its current resolution**, a blurred version of itself, with a line such as `are you there?`
2. It makes a soft sound, then repeats it once a minute.
3. After 30 seconds the screen fades back to dark, but it keeps calling quietly for **10 minutes**.
4. **If you hold it within those 10 minutes**, it shows itself fully, sharp and bright: `you came.` No loss for the next 4 hours.
5. **If nobody comes**, it goes back to sleep and loses **10 pixels**. Next time you wake it: `I called. you didn't come.`

The image only shows itself fully when it's held. When it asks, it offers only a glimpse, so the call invites a gaze rather than giving the full image to anyone passing.

Whole lifecycle

0. The empty vessel

With no image inside, the screen shows only a QR code and one line: `give me an image`. It's waiting.

1. Arrival

Someone scans the QR code with any phone. Their phone joins the device's own WiFi network, and a page opens where they choose a photo from their album.

The transfer is the image's **first loss**. It's shrunk to fit its new vessel, and the screen says so:

`arrived at 240 × 320 (from 4032 × 3024)`

Before the image will settle, the page asks the person to delete it from their phone and confirm with **"I've deleted it"**. Letting go of the only other copy is the condition for care to begin.

2. Life

From now on, its only variable is **resolution**, and every loss is permanent: written into the file and never recovered. The readout `240 × 320` always sits on top of the image.

It sleeps by default. The screen is dark and the image can't be seen. It shows itself only when someone holds it or presses a button, or when it wakes up to ask for attention.

Time takes from it. It loses **1 pixel per hour** without care. This continues even when it's switched off; when it wakes, the time away is applied at once: `you were gone 6 hours. I lost 6 pixels.`

Care preserves it:

- **Holding**: time stops while it's held.
- **Answering its call**: about 3 times a day it wakes dimly, blurred to half its resolution, and asks `are you there?`, then keeps calling softly for 10 minutes. If it's held in time, it shows itself fully (`you came.`) and loses nothing for the next 4 hours. If not, it loses **10 pixels** and goes back to sleep. Next time: `I called. you didn't come.`
- **Attention**: one button press a day sets today's resolution as a floor that time can't go below for 24 hours.

Shaping it costs it:

- **Crop**: tilt towards an edge and press edit. That edge is gone for good.
- **Edit values**: brightness, contrast or a filter. Each saved edit costs **2 pixels**, like the loss from re-saving a file.
- **Share**: someone opens its share page and receives a copy at **half** its resolution, the poor copy, and the original loses **10 pixels**.

3. The middle

At half its original width (120 px), it wakes on its own, whatever the time: `I am half of what I was.`

This is where the attention button matters most. From here on it **calls more often**, gradually rising from 3 times a day to every couple of hours near the end. It needs you more as it fades.

4. One pixel

When the last step of resolution goes, the image becomes a **single field of colour**: the average of everything it once showed. It wakes one last time to say `I am one colour now`. It stays like this for **one final day**, still able to be held, but with nothing left to lose except itself.

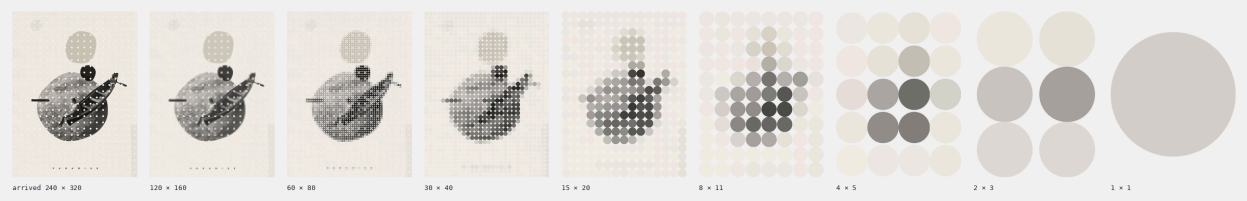
5. Death

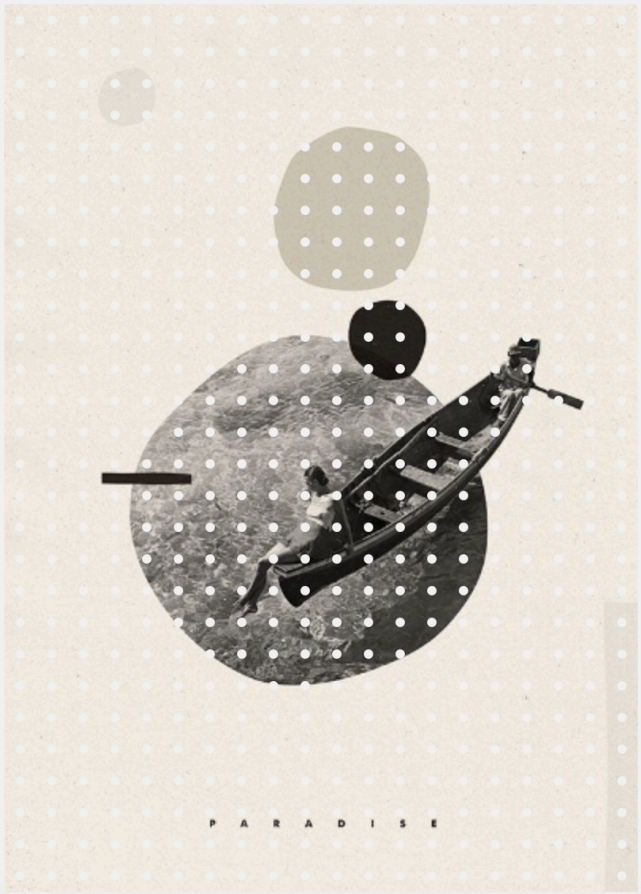
After that day, the data is **erased**. The screen goes dark, there is no copy anywhere, and then the QR code returns: `give me an image`. The vessel outlives what it held.

Timeframe

Life	Duration
Fully neglected	about **10 days** (240 hours, faster with missed calls)
Cared for daily	**weeks**: holding, answered calls and attention slow the losses almost to nothing
Crit or demo mode	about **20 minutes**: one hour becomes 5 seconds, with a call every minute or so and a 20-second window to answer

The care is what makes a long life possible, and it's measurable: anyone can read on the screen how well the image has been looked after.




PARADISE

