

Positions through essaying

Transcript:

[Background]

It was an ordinary day.

I finally overcame my laziness, opened my laptop, and got ready to work—when an error message popped up.

I stared at it for 30 seconds, feeling frustrated and helpless, while it silently stared back.

It felt like a conversation, but an unfair one.

[Enquiry1]

--“How do system error messages use language and interface to look like a dialogue, but actually give users no real response or control?”

I want to reveal all of this...

[Research1]

The system gives us buttons, but are they really choices?

I want “Help”, it opens a useless webpage.

I want to know what actually happened, but I can’t understand a thing.

Is there really a difference between “OK” and “Cancel”?

[Iteration1]

So, I started intervening.

I rewrote the language and structure of the error messages.

“Help” doesn’t actually help me.

No matter what I choose, it always leads to waiting.

“Viewing details” only leaves me more confused.

□ In this dialogue, we usually have no space or right to respond.

So I began inserting imagined user replies.

I wanted to express my confusion, frustration, and helplessness when facing these messages.

□ In this dialogue, the system often uses design strategies to guide our decisions.

Some buttons are highlighted, some are dimmed or hidden.

So I removed all the text — leaving only buttons.

What happens when there’s no instruction at all?

Will we still be guided by color, by position, by visual hierarchy?

□ In this dialogue, we are always forced to wait.

But we never know how long the wait will be.

What does “a moment” even mean?

[Enquiry2]

So I decided to start a counterattack beginning with the time wasted in waiting.

**--“If there’s a moment when we’re forced into a waiting state,
what can we do to make that time feel more meaningful?”**

[Iteration2]

□ I started experimenting with prompts on the waiting screen,
so users would have something to do while they wait.

Do you want to learn how to say ‘hello’ in another language?

(in Chinese, French, Spanish, Hindi...)

[Reference]

What else could be done?

Jenny Holzer’s project – placing truisms in everyday public spaces – inspired me.

Could the waiting screen become a kind of micro information transmission platform?

[Iteration2]

□ I began embedding readable content into buttons like “Help”:

facts from today in history, weather updates...

And then I thought –

what if this space could host more meaningful content that needs to be seen?

I added missing person notice.

Clicking “Help” could lead to a missing person platform.

□ Then I wondered –

what if this space invited interaction?

Maybe it could be a to-do list,

or a place to express frustration with the system.

And maybe...

we’re not alone in waiting.

Think about it –

while you’re stuck watching a loading screen,

there are probably thousands of others around the world

doing the same thing.

□ So I started thinking about creating a sense of togetherness –

making waiting feel less lonely.

I drew inspiration from game interfaces and built a virtual waiting room.

I added system announcements,

a chat window,

even achievements,

and a waiting-time leaderboard.

[Enquiry3]

But that still wasn't enough –
I wanted an equal dialogue with the system.
If the system has its own language structure,
then **I want to build a human one, like a way to speak back.**

[Reference]

In Mirror Ritual, the system reads users' faces and writes a poem using AI.

[Iteration3]

I wanted to reverse that.
So I began breaking down error messages,
rewriting them with imagery,
and translating them into poetic expressions.
I analyzed their structure by dividing them into three parts:
Who(subject); What(action); For what(subject)

I started creating a poetic language toolkit.

I built a vocabulary list
and set translation principles:
first, understand the word in its original error context;
then, translate it using imagery, metaphor, and emotional association.
I also designed poetic translation templates,
drawing from haiku, Song lyrics, Romantic poetry, visual poetry, etc.,
to make poetic responses easier to generate.
So users could turn cold, technical error messages into emotional, human
responses.
By selecting original system vocabulary and applying it to poetic templates,
they can reconstruct each sentence –
transforming system commands into something metaphorical, reflective, and
deeply personal.

Now,
Would you like to try?

Link:<https://tinyurl.com/t68ma3jz>.