PATTERNS EVERGREEN

Folk Interfaces

People reappropriating existing software to solve their own unique problems

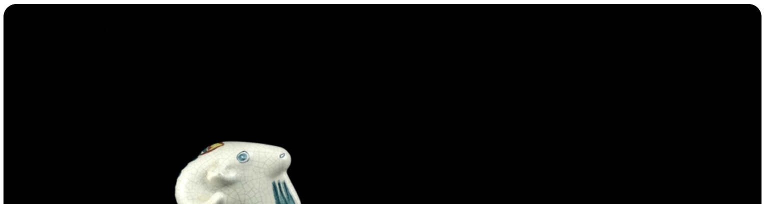
Planted 1 day ago

THE CONTEXT

Folk comes from the old English *folc* meaning “common people, laity; men; people, nation, tribe; multitude; troop, army.”

In modern lanuage we use it to mean “of the common people” or “from the people.” Anything that emerges from the ground-up, from the needs and wants of regular folk, rather than from large professional, political, or commerical interests.

Folklore is a collection of oral stories and cultural beliefs people pass through the grapevine. Folk art is what people make of their own accord, outside the institutions of Art, usually with an eye towards pragmatism.



Finnish folkart figurines – Source: Flickr

PRISPEJ AJ TY KU S
ÉHO PROSTREDIA
SMREČINA B. BYSTRICA ČSN-494705 CENA 20 HAL
POKLADNICE NÁRODA
Solo Lipník · ČSN 40 4105

Czech folkart matchbox covers – Source: Flickr

Folksonomies are informal taxonomies developed by users on social sharing platforms. Usually in the form of tags on sites like Flickr, Tumblr, or Instagram.

Designers talk of “folk practices” to describe the ways people creatively adapt and repurpose objects in their environment to meet their needs. Ledges used as tables, chairs as doorstoppers, cup handles as teabag holders, and stairs as seating.

An impromtu table – Source: Flickr

An impromtu cup holder – Source: Flickr

Folk creations fill a gap. They solve problems for individuals and small communities in a way that that centralised, top-down, industrial creations never can. They are informal, distributed practices that emerge from real world contexts. Contexts where individuals have little or no control over the "official" means of production – of furniture, urban architecture, crockery, artwork, media stories, or taxonomies. In response people develop their own unpolished, unofficial, and deeply practical creations.

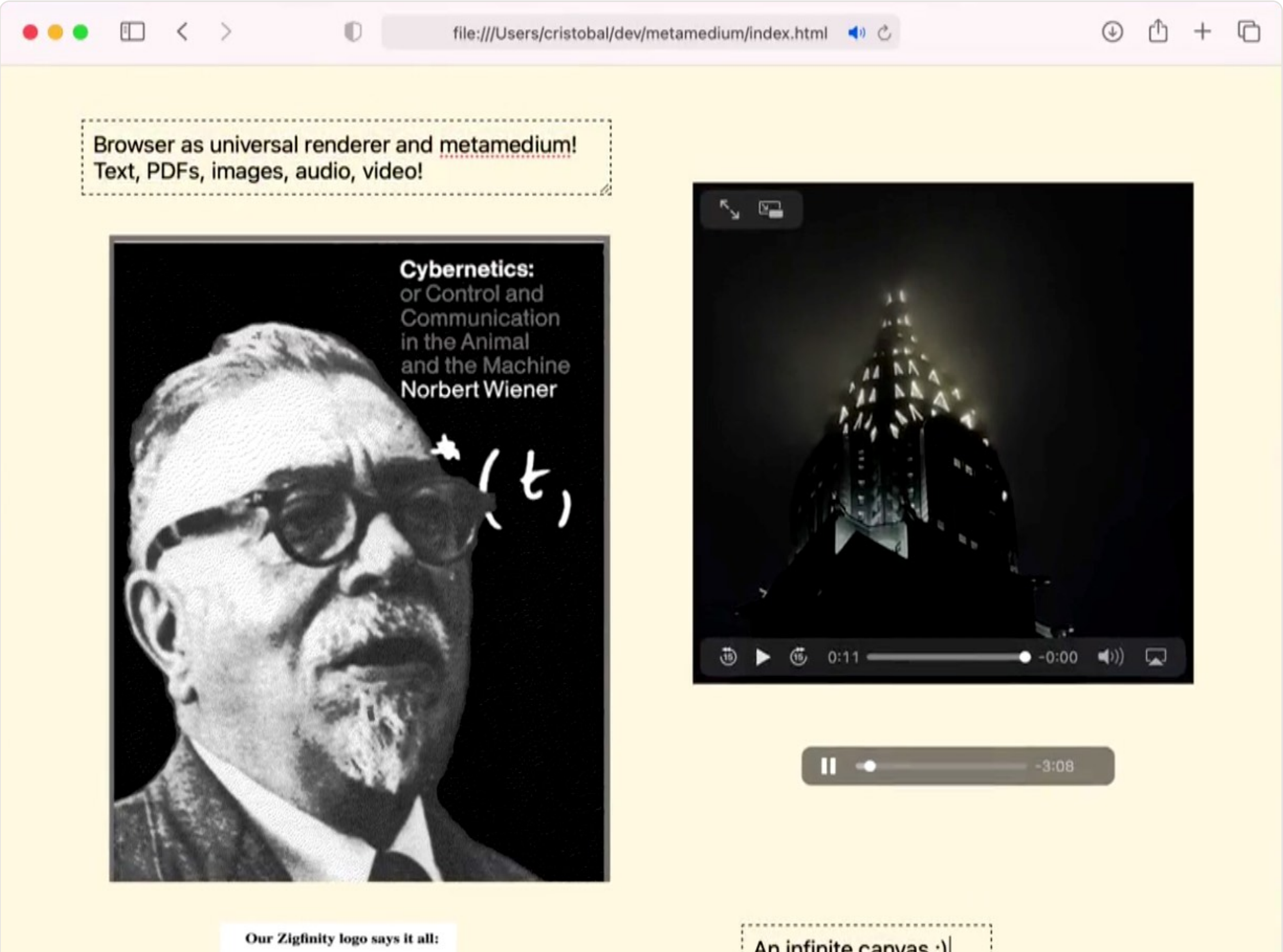
The world of software and interface design has become yet another place of centralised, mass production. Large California-based companies produce monolithic apps and interfaces for a global audience. They're designing for the broadest, generic use cases to make things work for the greatest number of people.

No matter how much well-intentioned user research these companies invest in, they'll never be able to produce software that fully meets the needs of individual users and culturally distant communities. A situation primed for folk creations.

THE PATTERN

Folk interfaces are when users reappropriate existing software to solve their own problems. Rather than using applications in the ways designers and developers intended, they creatively reconfigure them to do unexpected things.

Cristobal Scuitto coined the term **folk interfaces** while writing about their experiments using the browser as a media interface for local files.[1]


file:///Users/cristobal/dev/metamedium/index.html

Browser as universal renderer and metamedium!
Text, PDFs, images, audio, video!

Cybernetics:
or Control and Communication in the Animal and the Machine
Norbert Wiener
*(t)
0:11
-0:00
-3:08

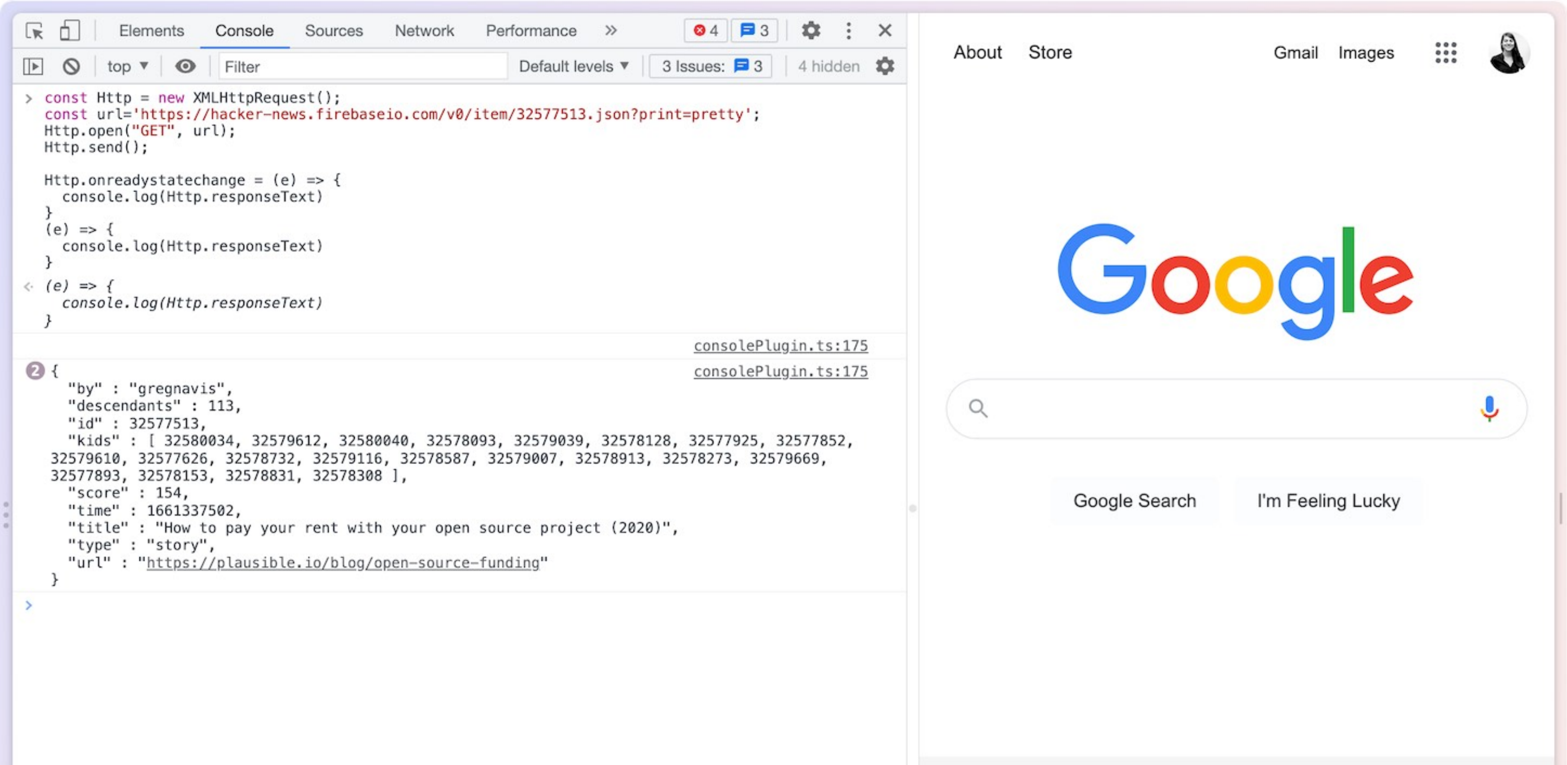
Our Zigfinity logo says it all:

An infinite canvas :)

Cristobal's browser-as-media-interface

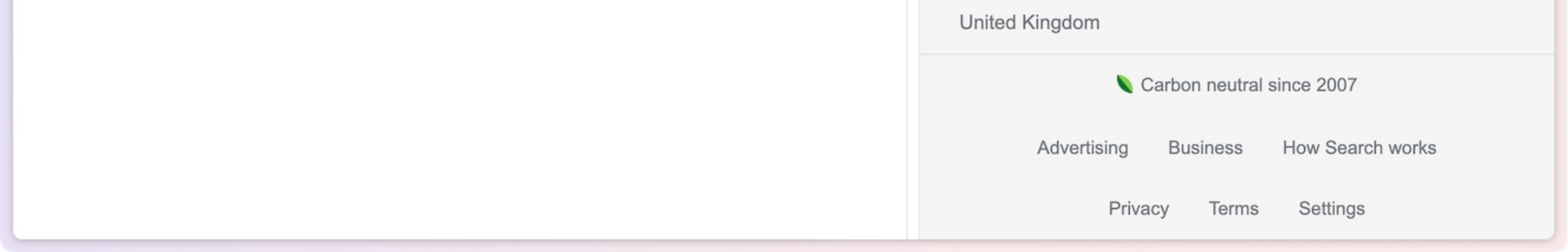
The browser is designed as a place for you visit and interact with webpages. But as a collection of affordances, it's a rendering engine that can display images, play animations, accept text, and register user clicks, drags, and gestures. The data it renders could all be local to your machine. There's no rule saying you *have* to browse the web with it.

We could also treat the browser as a JavaScript runtime environment. If you open up the console tab inside developer tools, you can execute JavaScript code just like in an IDE. You could check the date and time, write a fizz buzz test, or call the Hacker News API.



```
> const Http = new XMLHttpRequest();
  const url='https://hacker-news.firebaseio.com/v0/item/32577513.json?print=pretty';
  Http.open("GET", url);
  Http.send();

  Http.onreadystatechange = (e) => {
    console.log(Http.responseText)
  }
  (e) => {
    console.log(Http.responseText)
  }
< (e) => {
    console.log(Http.responseText)
  }
                                                              consolePlugin.ts:175
2 {                                                           consolePlugin.ts:175
    "by" : "gregnavis",
    "descendants" : 113,
    "id" : 32577513,
    "kids" : [ 32580034, 32579612, 32580040, 32578093, 32579039, 32578128, 32577925, 32577852,
  32579610, 32577626, 32578732, 32579116, 32578587, 32579007, 32578913, 32578273, 32579669,
  32577893, 32578153, 32578831, 32578308 ],
    "score" : 154,
    "time" : 1661337502,
    "title" : "How to pay your rent with your open source project (2020)",
    "type" : "story",
    "url" : "https://plausible.io/blog/open-source-funding"
  }
>
```



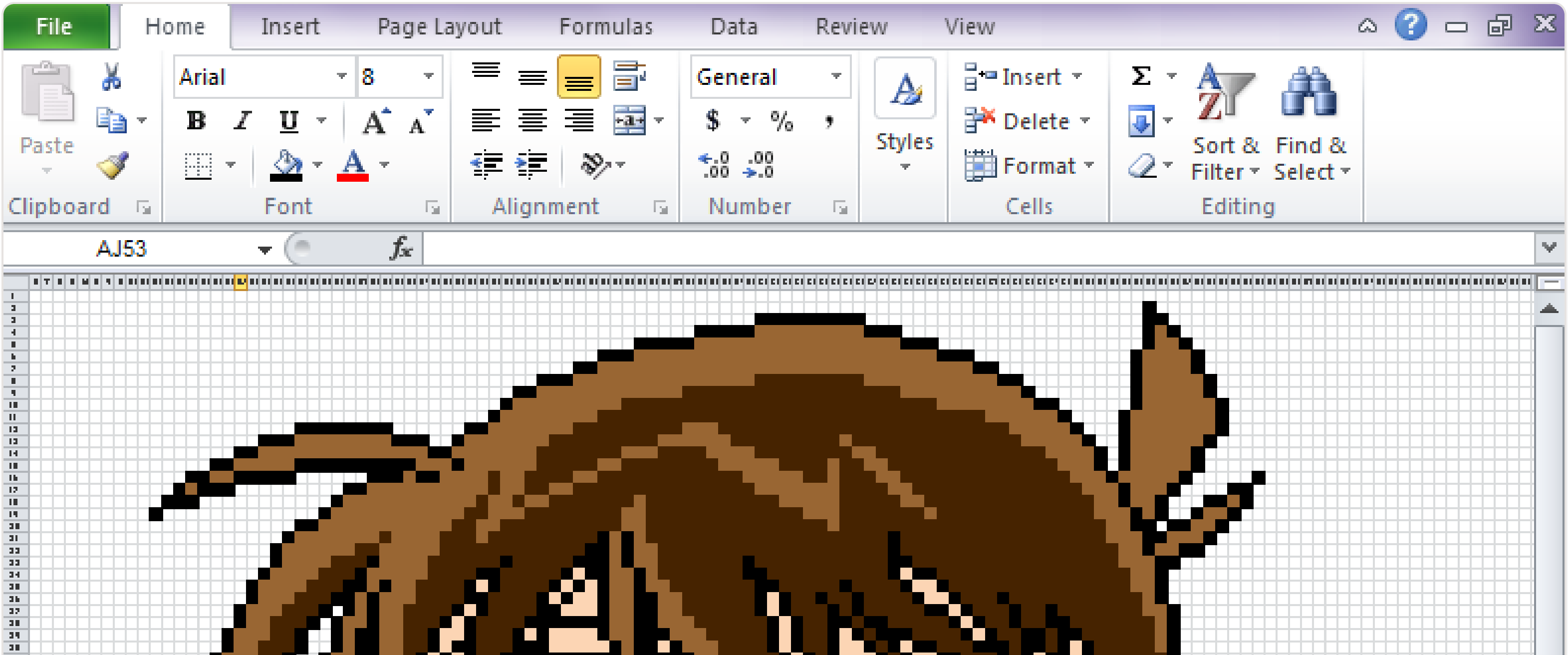
Using the built-in browser console as a JavaScript runtime environment

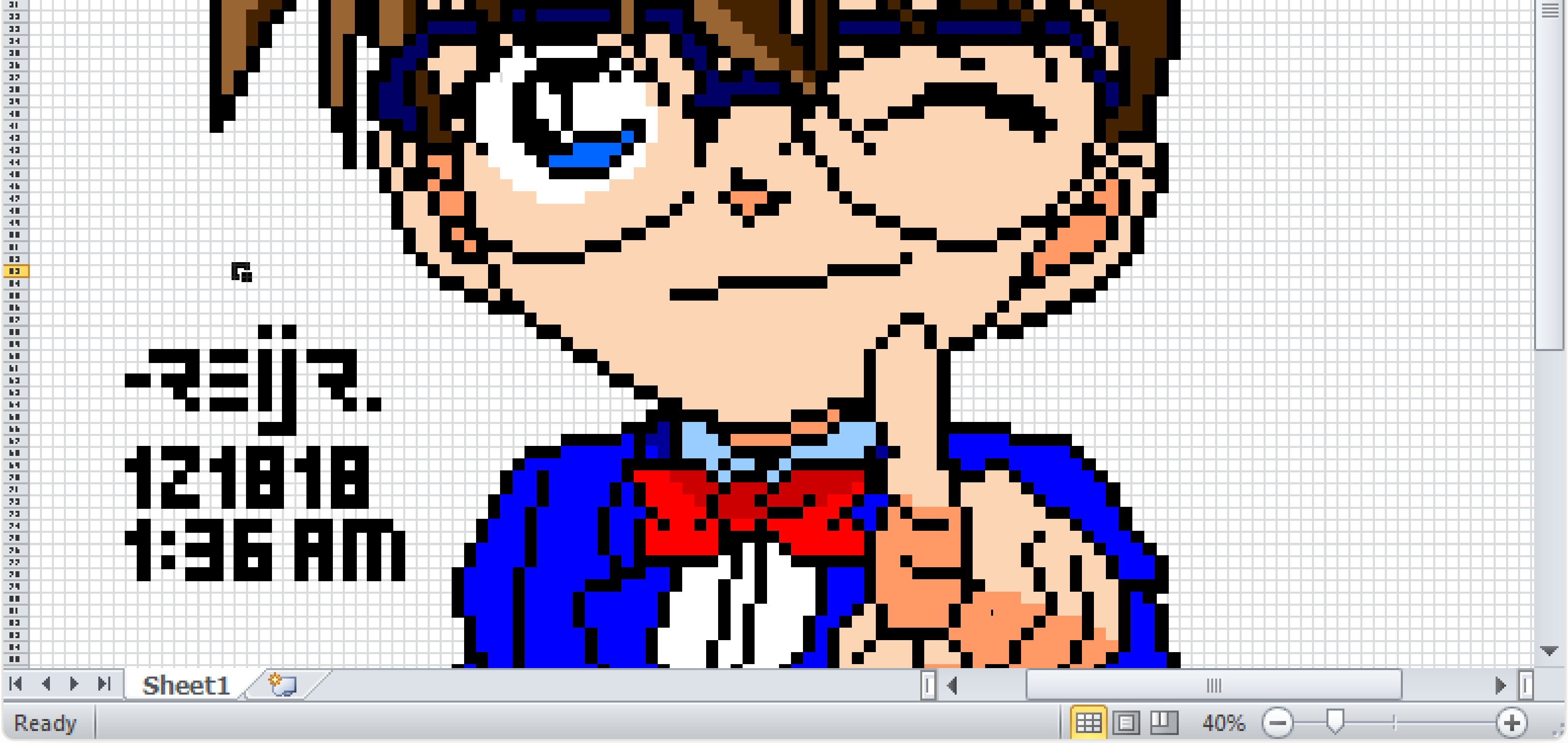
This requires looking at applications sideways. You can look at an interface and see it as a clearly signposted user journey you should follow. Or you can see it as a collection of functions and affordances to repurpose. As raw material, rather than a guided path.

Spreadsheets are another infamously malleable material for folk interfaces. The simple concepts of cells, rows, and formulas can be turned into all manner of things beyond financial accounting.

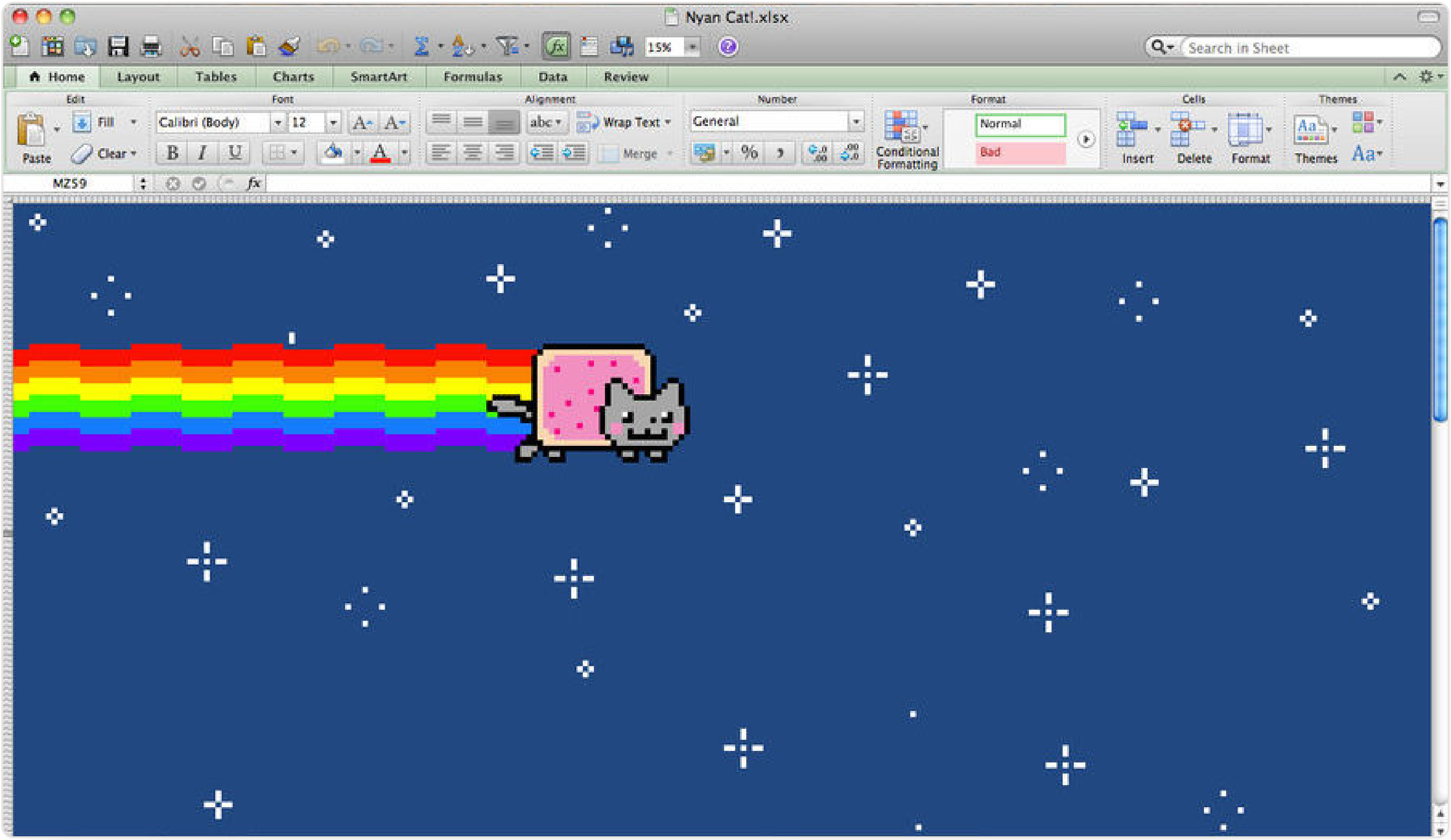
Here's an entire digital audio work station that Dylan Tallchief built in Excel:

Spreadsheet art is another reappropriation of the interface. People make elaborate pixel art images by colouring in the cells.



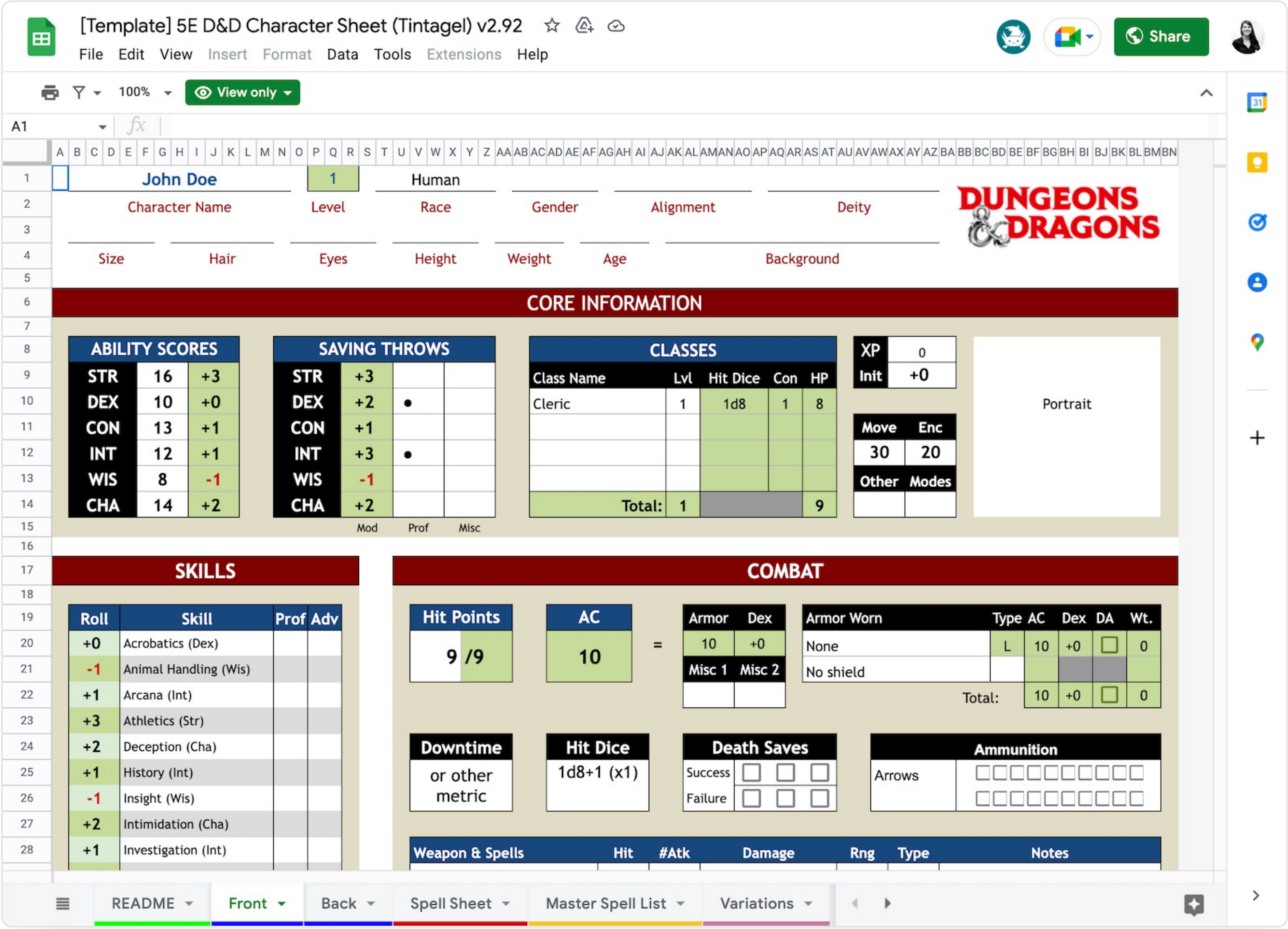


The manga character Conan Edogawa created in Microsoft Excel – Source: reijr



Nyan Cat art created in Microsoft Excel – Source: Oscarit07

Here's a character dashboard designed to faciliate a session of the role-playing game Dungeons and Dragons, entirely built in Google Sheets:



John Doe	1	Human			
Character Name	Level	Race	Gender	Alignment	Deity

Size	Hair	Eyes	Height	Weight	Age	Background

CORE INFORMATION

ABILITY SCORES		
STR	16	+3
DEX	10	+0
CON	13	+1
INT	12	+1
WIS	8	-1
CHA	14	+2

SAVING THROWS	Mod	Prof	Misc
STR	+3		
DEX	+2	●	
CON	+1		
INT	+3	●	
WIS	-1		
CHA	+2		

CLASSES

Class Name	Lvl	Hit Dice	Con	HP
Cleric	1	1d8	1	8
Total:	1			9

XP	0
Init	+0

Move	Enc
30	20
Other	Modes

Portrait

SKILLS

Roll	Skill	Prof	Adv
+0	Acrobatics (Dex)		
-1	Animal Handling (Wis)		
+1	Arcana (Int)		
+3	Athletics (Str)		
+2	Deception (Cha)		
+1	History (Int)		
-1	Insight (Wis)		
+2	Intimidation (Cha)		
+1	Investigation (Int)		

COMBAT

Hit Points	AC
9 /9	10

=

Armor	Dex
10	+0
Misc 1	Misc 2

Armor Worn	Type	AC	Dex	DA	Wt.
None	L	10	+0	□	0
No shield					
Total:		10	+0	□	0

Downtime	Hit Dice
or other metric	1d8+1 (x1)

Death Saves			
Success	□	□	□
Failure	□	□	□

Ammunition	
Arrows	□□□□□□□□□□ □□□□□□□□□□

Weapon & Spells	Hit	#Atk	Damage	Rng	Type	Notes

None of these uses cases could have been predicted or catered for by the Microsoft Excel product team. They're pure folk creations.

Historically, the term “folk” was used when talking about the rural poor and illiterate peasants. It's fitting that folk practices in software are often developed by people illiterate in programming. Without access to formal programming skills, they use what they have to make things work. In ways that demonstrate profound creativity and logical problem-solving.

This scavenging in the SaaS landscape is how people express agency and control in a computing environment that limits who has access to programmatic power. Only professional programmers and designers get to decide what buttons go on the interface, what features get prioritised, and what affordances users have access to. Subverting that dynamic is the only way people can get their needs met with the computational tools they have at hand.

Software isn't first discipline to struggle with this power dynamic between professionals and everyday people. Architecture and product design have been at it for centuries.

Victor Papanek is a widely venerated designer who developed the “design for the real world” philosophy in the 1970's. He advocated for an approach that moved away from aesthetics and toward solving practical problems for people with found, local materials. His most infamous design was the 'tin can radio' that could pick up a one-way signal in rural areas – enough to get a news broadcast. It could be made out of a single tin can, some copper wire, a rusty nail, and any avaliable fuel source, for a total of 9 cents in production costs.

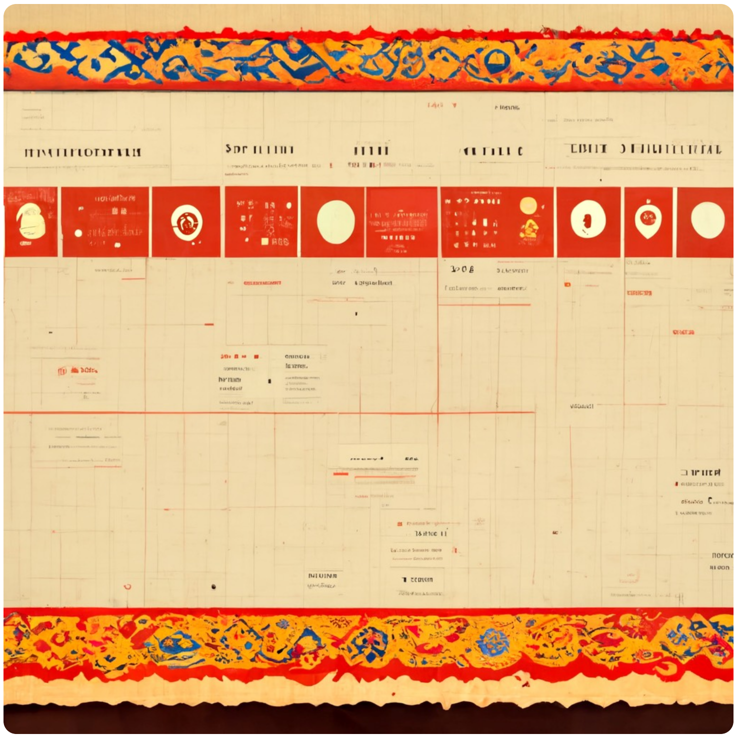
Victor Papnek's tin can radio

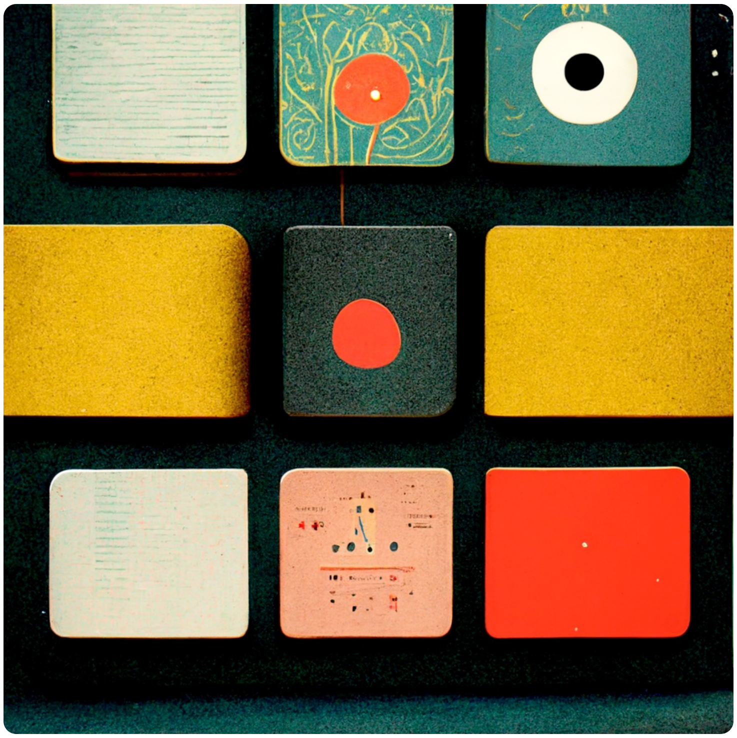
The Japanese philosophy of Mingei was another movement we should heed. Developed in the 1920's, it championed the beauty of everyday items made by “nameless craftsmen” with traditional techniques and local materials. This pushed against the prevailing cultural trends of moderinism that valued “high art” made by famous artists.

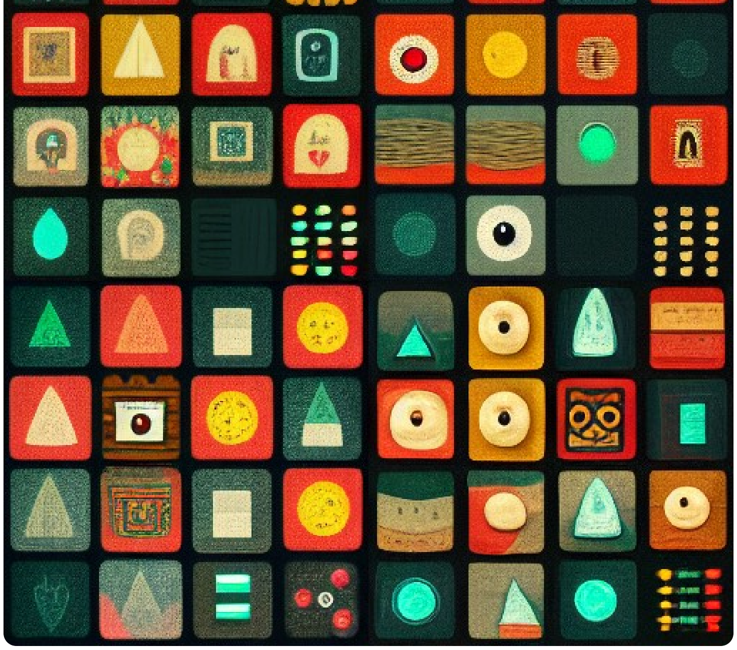
It's unclear what the full range of local materials and traditional techniques of interface design and programming are yet. Perhaps its simply spreadsheets, browsers, and more accessible end-user programming tools. We should be on the lookout for others. Giving people the raw materials to solve their own problems is the folkiest way forward.

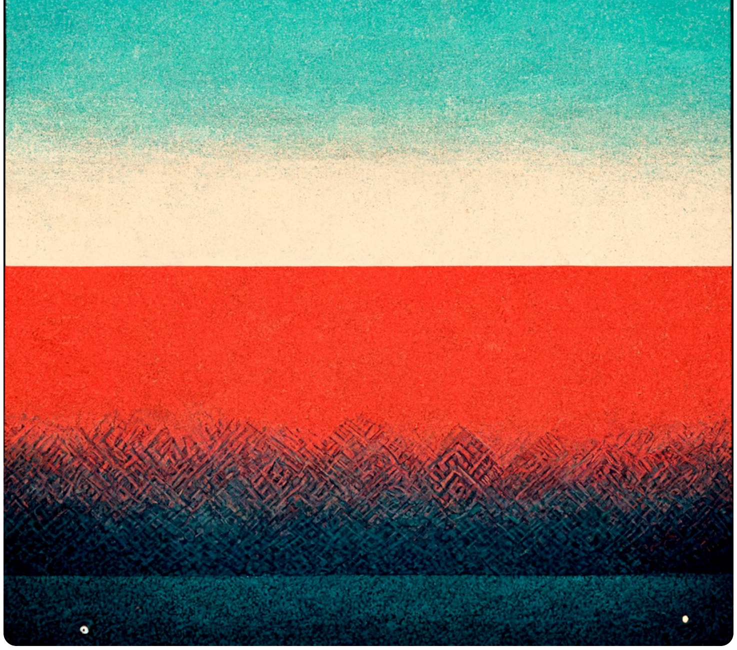
BONUS CONTENT

In the spirit of reimagining interfaces, please enjoy this collection of “folk art interfaces” dreamed up by Midjourney. Such vibes.









References and Further Reading

Folk (Browser) Interfaces →
Cristobal Scuitto

Painting the Tin Can – On Victor Papnek →
Alex Catanese

Design for the Real World →

Victor Papnek

The Beauty of Everyday Things (on Mingei) →

Soetsu Yanagi

End-user Programming →

Szynmon Kaliski & Adam Wiggins

Mingei – Wikipedia →

End-user Development – Wikipedia →

WANT TO SHARE? Tell Twitter About It