Terhi Marttila*
Interactive Technologies Institute
Pedro Cardoso**
Faculty of Fine Arts of the University of Porto, i2ADS

Pseudocode poetics: Inconvenient interactions for meaningful, dumb phone games

Abstract: Pseudocode is a way for programmers to sketch out the approximate structure and processes of their software before implementation. We explore the use of pseudocode to evoke imaginary software as a way of critically addressing some of the problematic elements of smartphone usage patterns. This paper presents work-in-progress in which we select six works in pseudocode that target different aspects of smartphone usage and discuss their critical contributions: the aspects they target, how this is done and whether such pseudocode could be implemented. Future work includes creating additional pseudocode to explore further aspects of smartphone usage and possibly implementing some of it.

Keywords: aesthetics, artistic research, friction, game design, interaction design, pseudocode poetics, smartphone, digital media

Resumo: O pseudocódigo é uma forma de os programadores esboçarem a estrutura aproximada e os processos do seu *software* antes da implementação. Exploramos a utilização do pseudocódigo para evocar *software* imaginário como forma de abordar criticamente alguns dos elementos problemáticos dos padrões de utilização de *smartphones*. Apresentamos um trabalho em curso no qual selecionamos seis peças em pseudocódigo que abordam diferentes aspetos da utilização de *smartphones* e discutimos as suas contribuições críticas: os aspetos que abordam, a forma como o fazem e se esse pseudocódigo poderia ser implementado. Trabalhos futuros incluem a criação de mais pseudocódigos para explorar outros aspetos da utilização de *smartphones* e, possivelmente, a sua implementação.

Palavras-chave: estética, investigação artística, fricção, design de jogos, design de interação, poética do pseudocódigo, *smartphone*, *media* digitais

1. In a world of frictionless aesthetics: Introduction

Look around you. At any random corner, you'll probably see someone, everyone, using a smartphone. Over the last two decades, mobile phones have increasingly claimed space in our everyday lives, to the extent that we are no longer able to clearly see all the changes this device has brought about (Greenfield 2017: 9-10). Consider this observation from the perspective of your own daily routine: "for many of us, they are the last thing we look at before sleep each night, and the first thing we reach for upon waking" (9-10). Whether this sounds familiar or not is, of course, an exercise in personal self-judgment. But in this sense, one might also wonder whether the smartphone is using us instead.

While the smartphone undeniably brings benefits to our daily lives, many of us may recognise harmful personal patterns in its usage. After all, much of the software it contains has been designed to lure us into spending time with it, and not always to our benefit, or at least not with our benefit in mind. As early as 2001 (and even before the advent of today's ubiquitous smartphones), attention was identified as a key resource in the information age (Davenport & Beck 2001). Today, social media corporations, for example, sell user attention to advertisers. This has given rise to an ongoing debate on the ethics of this business, given the harm that social media can cause to users (Bhargava & Velasquez 2021: 321). The design of social media platforms strives to entice us to spend as much time as possible engaging with the content they deliver. As such, much of today's software is developed using human-centred design methods and practices that heavily promote user-friendly interactions with end users (Kuang & Fabricant 2019), which over the last few decades have become mainstream in the industry. Such philosophies are conducive to the design of easy-to-use digital media that require little effort from users to interact with, in fact, that often require users to think very little at all (Krug 2014).

While not denying the undeniable breakthroughs and the fundamental importance of the design philosophies that have enabled us to use digital media with such ease, we can also observe that these same philosophies have been exploited to manipulate users into taking unreflective actions that do not serve their own benefit. Much of digital media commodifies Csíkszentmihályi's theory of flow (1994), and what was originally conceived to combat social alienation in the pursuit of happiness and self-fulfilment is now, in digital media, fuelling the very problems it sought to address in the first place (Soderman 2021). While in flow, users lose their reflective self-consciousness and ignore other needs, as the experience is so engaging. They therefore become easy to manipulate into spending more and more time with digital media, often performing thoughtless actions, and frequently acting against their own best interests. It is consequently no surprise that many corporations are eager to see the vast majority of people in the world willingly cherish the smartphone in their hands. After all, it is so easy to be captivated by what the phone offers.

It is time to change the way we relate to digital media. But we also need to consider new ways it might relate to us, and for that, we must begin designing alternative ways

eLyra 25, 06/2025: 99-115 – ISSN 2182-8954 | https://doi.org/10.21747/21828954/ely25a6

for these devices to interact with us in the first place. Instead of fostering constant flow, we now need something that makes us pause and reflect, something to moderate our relationship with digital media; we need some sort and some degree of friction to be designed into the system. In this work, we explore a way of relating with smartphones through a particular expression of what we call *aesthetic friction* (Cardoso *et al.* 2019; 2025; Carvalhais *et al.* 2019; Isabella *et al.* 2022; 2024; Malaquias & Cardoso 2025): *i.e.*, aesthetic disruptions that challenge users or players to stop and reassess their interactions with the system, seeking meaning rather than comfort and convenience. The speculative designs of the games we present explore *aesthetic friction* as a way of questioning the complacency fostered by current frictionless design philosophies.

In this creative-critical contribution we apply a specific arts-based research methodology to address the critical context of mobile phone usage patterns. One of the elements that motivated this work was the exploration of the evocative potential of pseudocode for imagining software that may not even be feasible, such as *Thrower* (see Section 3), which, in reality, would end up breaking users' phones. To what extent can pseudocode, as an instruction for use and as something that elucidates the inner workings of a software application, evoke a possible mental scenario in the reader? Can this playful imaginary software be used to critique our smartphone habits?

In this work, we introduce the concept of *pseudocode poetics* as a speculative artistic research practice in which we write pseudocode as a way of evoking imaginary scenarios, some of which might not even be possible to implement at all. We use pseudocode poetics to imagine subversive software, specifically for smartphones, designed to disrupt some of the usage or behaviour patterns we have identified. Finally, as part of the artistic research methodology, through the affordances of a critical practitioner reflection (Candy 2019) alongside an autoethnographic approach (Adams *et al.* 2022), we discuss how each of our pseudocode poems critically addresses smartphone usage patterns (through the tensions they create and the aspects they critique) and speculate on how the games could play out in real life.

2. Pseudocode poetics as speculative software design

For Roy, "pseudocode aims to fill the gap between the informal (spoken or written) description of the programming task and the final program (code) that can be executed (or at least automatically converted into an executable form)" (2007: 2). It consists of natural-language statements that describe computational processes, reserved words or symbols for common processes and actions, phrases that describe computational tasks, and often some graphical notation (*ibidem*). Pseudocode is thus a hybrid between the idea of the code and the actual executable code: a rough sketch of a functional software system.

Roy further defines the minimum requirements for pseudocode as containing syntactical elements to represent at least: "sequence, iteration, selection, and recursion"

(Roy 2006: 2). Additionally, pseudocode notation "can include elements to describe input/output operations, file operations, and some level of modularization through function and method/procedure definitions to allow for the reuse of code and for information hiding" (*ibid.*). Reading pseudocode allows one to begin to grasp what a software does, how it achieves this result, and how it could be implemented. Pseudocode poetics can thus be used both to invoke the effects of hypothetical software on the real world and on our social interactions, as well as to provide clues about how these events are orchestrated within the hypothetical software.

At the turn of the millennium, practitioners in the field of electronic literature experimented with entwining actual code, pseudocode and other code-inspired forms of writing into their (digital) literary practice, with Alan Sondheim coining the term *codeworks* to refer to these practices.[1] Some of the key artists who explored this approach include Mez Breeze (who developed her own language, *mezangelle*[2]), Alan Sondheim and John Cayley, among others. Some examples of early codework were included in the *Electronic Literature Collection I*[3] and others are archived and digitally preserved in *The NEXT*.[4] Sandy Baldwin describes this practice as one in which code appears in the text as a "kind of residue or catalyst of machinic processes. The text is 'contaminated' by code" (Baldwin 2001). Cayley describes codework as "a term for literature which uses, addresses, and incorporates code" (Cayley 2002). Rayley writes: "codework makes exterior the interior workings of the computer. One formal purpose is to bring the function and code of the computer to a kind of visibility" (2002). Raley also quotes Alan Sondheim[5] from his Introduction to an issue of the *American Book Review*, where Sondheim outlines a taxonomy of three types of codework: "'Works using the syntactical interplay of surface language'; 'Works in which submerged content has modified the surface language'; 'Works in which the submerged code is emergent content'" (*ibidem*).

Although the popularity of the codework genre may have waned after the turn of the millennium, what is interesting about this form of writing is its untroubled relationship with, and appropriation of, the language of code. The use of code can appear as either actual executable code with poetic intentions, code-inspired and code-infused writing, or the use of language and concepts from programming to evoke instructions or actions, such as *if-then* statements, *while* loops, *print*-functions and other concepts familiar from programming languages. Pseudocode poetics fits into this tradition through its untroubled use of programming concepts with the intention of invoking instructions for use. Pseudocode poetics does not aim for executable code but rather seeks to appropriate the notion of the algorithm and code concepts as a way to evoke hypothetical scenarios of human-computer interaction.

Raley observes that codework in the context of art has a socio-cultural history that extends beyond the turn of the millennium:

eLyra 25, 06/2025: 99-115 – ISSN 2182-8954 | https://doi.org/10.21747/21828954/ely25a6

> Oulipo's Algol code poems and the use of computer instructions in their texts; the long-term tradition of generative aesthetics and poetic programming, such as Tristan Tzara's 'algorithm' for Dadaist composition (including in a similar vein La Monte Young's and John Cage's instructional scores); ASCII art; the composition of Quines; and Perl poems. (Raley 2002)

The Surrealists were also known for foregrounding play, games and instructions in their practice as a form of subverting established modes of enquiry, in what Gooding describes as the "surrealist provocation of bourgeois normalities" (Brotchie & Gooding 1995: 10). The Surrealists devised algorithms that would produce text and images based on principles such as randomness and combinatorial approaches. "Chain games" like the 'exquisite corpse' invites three participants to draw a head, body and legs, each unaware of what the others had contributed to the whole (25). Another example is Tristan Tzara's "other ways of making texts": cut up an article word by word, shake it in a bag, and let a poem emerge in the order the scraps fall out of the bag (36). These practices use language as instruction to establish rules and orchestrate behaviour as a means of invoking the unanticipated.

Besides instructions for social, combinatory games, the Surrealists also explored the many dimensions and possibilities of objects: "the Object in its Surrealist manifestation is distinguished from others by its radical change of role. Wherever it comes from, whether it exists in the objective world, in reverie, or in dreams, it no longer maintains its identity within the banal categories of the ordinary and the everyday" (Brotchie & Gooding 1995: 105). What kinds of possible meanings could smartphones be stretched into, both as material objects and as non-material manifestations of behaviour and agency through their software? What kinds of games, instructions for use and play, would the Surrealists come up with in an era of smartphones? Take, for instance, this game:

> each participant has a watch with an alarm which is set to an identical time of which the players are ignorant. They carry on their usual activities, but at the instant the alarm goes off, each player notes down his location and what 'most strikingly impinges on his senses'. Players meet at an agreed time to compare notes. (121)

Here, the Surrealists use language for instruction alongside rudimentary technology to orchestrate collective behaviour and play, perpetuating a sense of wonder and the art of noticing.

In the vein of the Surrealist interventions, our aim is to disrupt the bourgeois normality of the information age and, in particular, the smartphone. After all, the smartphone promises to make our lives easier, while at the same time bringing a swarm of problems that may sometimes be difficult to pinpoint. Bridle writes: "computational thinking is an extension of what others have called solutionism: the belief that any given problem can be solved by the application of computation. Whatever the practical or social

problem we face, there is an app for it" (2018: 4). Our pseudocode poetics rages against technological solutionism by provoking the opposite: proposing impossible software that underscores and highlights some of the problematic aspects of software itself and the ways it shapes our behaviour and agency. Sicart argues that software performs actions that affect humans in areas such as politics and our social relations (2023: 22). Software is like an alien agency, and one of the great challenges of the information age is to learn to relate to such agencies (23-24). Sicart suggests that a key approach to making sense of this alien agency of software is to play with software (24). Pseudocode poetics invites play as a form of critique of the software designs that dominate, in this case, the domain of smartphone applications and the social and behavioural patterns that software agency imposes upon us.

Numerous digital artists have addressed the human-smartphone relationship through provocative, playful digital artworks for smartphones. Miguel Sicart's *ATTN* is "an attention manager that allows users to experience the pleasures of staring at the phone, without the burdens of content".[6] The app slowly fades the screen to black, allowing users to tap the screen to turn it back to white if they need more time in the presence of the phone's bright screen. Terhi Marttila's *Infinite Scroll*[7] is a digital poem about the capitalist processes at play behind the scenes that benefit from the movement of fingers drawn into the infinite scroll, which, much like Sicart's *ATTN*, plays with slowly fading the screen to black as a way of weaning the user from the lure of the screen. Ben Grosser's *The Endless Doomscroller*[8] caters to doomscrolling tendencies by delivering an unending stream of combinatorially generated "doom" content. Pippin Barr's *It is as if you were on your phone*[9] is a simple website for mobile phones in portrait mode, which invites the user to make different kinds of interaction gestures and to smile, laugh, scoff, etc., promising that to outsiders the behaviour will appear as authentic smartphone use. Moreover, having identified the existence of a market for personal devices that go against the grain, various corporate initiatives have emerged to create "dumbphones", phones "designed to be used as little as possible",[10] as the company behind the Light Phone claims. These interventions and initiatives seek to subvert, critique, and make visible the agency that software exercises over us, and they are important examples of subversive software design in and of themselves. However, in this contribution we focus on a semi-digital intervention as a means to the same end.

eLyra 25, 06/2025: 99-115 – ISSN 2182-8954 | https://doi.org/10.21747/21828954/ely25a6

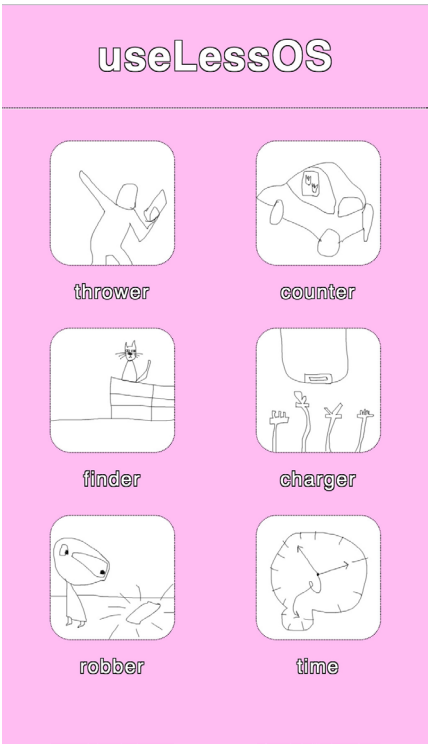

Figure 1. screenshot of *useLessOS* interface,
available at https://uselessos.github.io/useless-os/

Over the next few pages, we invite the reader to contemplate six works of pseudocode poetics, small provocations that evoke imaginary situations of play and take a critical stance on different aspects of the way that we relate to the smartphone as an object in our everyday lives. The pseudocode poems are also presented as a web page (Figure 1).

3. Dumb Phone Games

3.1 Thrower

```
open app
    >> plant your feet firmly on the ground
    >> continue?
    >> assume your best launch position
    >> continue?

    >> throw the phone as far as you can
```

```
while ( phone in air )
play sound: “wwwwwwhiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii”
[ for duration of throw ]

          if ( duration of throw > 3s )
               play reward sound
                                                                  “ TA - DAA ! “
                                                               [ reward sound ]

          if ( duration of throw > 5s )
               play very impressed -sound
                                                                    “ WOOOOUW ! “
                                                          [ very impressed -sound ]

multiplayer win: person whose phone no longer turns on
single player win: number of points equals duration of throw
```

3.2 Step Counter

```
open app
>> drive for one hour

multiplayer win: biggest step count
single player win: beat current high score
```

3.3 Robber

```
open app
     >> leave this phone unattended somewhere in the street or any other open space
     >> leave!

multiplayer win: whoever loses phone first
single player win: beat high score, shortest amount of time to lose phone

# DLC: Robber - Get the phone back
open app
app finds phone
>> get your phone back, you twit!
```


eLyra 25, 06/2025: 99-115 – ISSN 2182-8954 | https://doi.org/10.21747/21828954/ely25a6

```
multiplayer win: whoever gets phone back first
single player win: beat high score, longest amount of time to get phone back
```

3.4 Finder

```
attention: this game can only be played when battery is at 2%

mute all sounds
open app
ask someone to hide your phone
then look for the phone

while ( person is looking for phone )

    turn sounds back on, lowest possible volume
    play videos
    look for wi-fi
    look for bluetooth
    make screen brightness max
    spend as much energy as possible

    every 2 minutes
           increase volume, by very little, then:
                 play sound: “meow”

single player win: find phone
single player lose: battery drained, phone still hidden

multiplayer win: who finds phone first
```

3.5 Charger

```
attention:
this game can only be played at your workplace (minimum 10 colleagues present)
this game can only be played when battery is at 2%

open app
    if ( battery ≥ 2% )
```

```
            >> this game can only be played when battery is at 2%
            close app

    else
            >> ask around for a phone charger that is compatible with yours

multiplayer win: first player to find compatible charger
singleplayer win: compatible charger found before battery is drained
lose: battery is drained

double the stakes: reject first compatible charger, find other
win (double stakes): find other compatible charger, double your points
```

3.6 Timer

```
if ( user picks up phone )
        show time + ( random number )

if ( user picks up phone again )
        show time - ( another random number )

single/multiplayer win: never
single/multiplayer lose: always
```

4. Discussion

In this section, we critically reflect on the evocative potential of *pseudocode poetics* in the six pseudocode games included above. We outline our intended design considerations, discuss which aspects of our relationship with mobile phones these playful works target, and speculate on whether our creations could successfully evoke thought or provoke critical reflection on this relationship. When analysing the tensions that our games address, we employ an autoethnographic approach to examine smartphone use scenarios that we observe in our everyday lives. After all, these everyday situations and observations were the source of inspiration for drafting these games. In the *Critique and Implementation* sections, we again imagine and speculate on how these games might play out in our daily lives. We reflect critically on our own creative outputs to unpack how our pseudocode poetics/games address the tensions we have identified. This discussion is structured as follows and seeks to answer the following questions:

eLyra 25, 06/2025: 99-115 – ISSN 2182-8954 | https://doi.org/10.21747/21828954/ely25a6

1. **Tension:** What is the usage pattern that we target? Where is the tension?
2. **Critique:** How does the pseudocode disrupt or critique this aspect?
3. **Implementation:** Could this pseudocode be implemented? Why or why not?

4.1 Tension

The tension in *Thrower* arises from the perceived value of our personal smartphone. Often the smartphone is an expensive item that we may have coveted for some time before purchasing, or it may be at the cheaper end, but nevertheless something we do not want to lose, damage, or break. A fractured phone screen is a common sight, and the moment it cracks is invariably a source of great frustration. We go to great lengths to protect our smartphones, and *Thrower* creates tension precisely in this respect.

In *Step Counter*, the tension lies in the step counter built into many of today's phones: our smartphone monitoring us, counting our steps in order to instruct us regarding our health. The smartphone user examines the step count and draws conclusions about their own behaviour, perhaps vowing to walk more steps the next day or feeling guilty for not having achieved the goal set by the software. In Sicart's terms, the software is exercising agency over us (2023).

Finder taps into the unsettling feeling of not knowing where your phone is and uses this as motivation for trying to win the game (find the phone). The player knows that the battery is only at 2% to begin with, which adds further pressure to try to win the game as quickly as possible.

Robber puts players in the unreasonable position of intentionally leaving an often expensive and valued device unattended, waiting to be taken by someone unknown. Players can do anything to attract attention to the unattended phone in order to see whose phone is taken first. Afterwards (in the downloadable content or DLC – expansion pack), players are tasked with trying to get their phones back, but by taking the longest possible time to do so, an unreasonable attitude that only makes the task harder and even dangerous.

Charger stresses players into the undesirable but all too common situation of having to ask colleagues for a phone charger, knowing that various compatibility issues between chargers and phones are likely to arise during the quest. What aggravates the situation is that players may choose to reject the first offer to double their points, which not only risks losing the game but puts them in the awkward position of refusing the goodwill of a fellow colleague and potentially damaging their personal and/or professional relationship.

Timer targets the compulsive gesture of unlocking the screen to check the time, only to forget it a few moments later. It draws attention to the futility of actions in a game that players never win and always lose: the game (within the magic circle) and time itself (outside the magic circle).

4.2 Critique

Thrower seeks to disturb our protective attitude towards the smartphone by proposing a game in which the winner is the one who throws their phone the farthest. The pseudocode thus targets the care we feel for our phone, disrupting our protective instincts by asking us to do the exact opposite: to throw it as far as we can. At the same time, the pseudocode also critiques our submission to the reward mechanisms of software and digital games by playing the sounds "ta-daa!" and "woow!" when we manage to keep the phone in the air for a long time.

In *Step Counter*, we subvert the step counter by asking the player to drive a car, because driving a car would register as driving, not as steps. It plays with deception misdirecting the player into taking actions that are counterproductive to achieving the game's goal.

Imagine playing *Finder*: you know you'll only hear a faint "meow" every two minutes. You stand in absolute silence in the most central spot of your home, becoming intensely aware of your surroundings (after all, no phone to grab to kill time) and acutely attuned to every sound. You know the app is doing its best to drain the phone's battery, and you want to catch those scant meows before losing the game, and your phone.

Robber makes players connect with unknown people who are unaware they are part of a game – a kind of dark play (Sicart 2015). Players may even stalk these people, to take the longest time possible to get their phone back, following them into their own lives and daily routines. After all, the code imposes no maximum time limit for getting the phone back.

Charger ironises (and may even degrade at times) personal relationships when the player rejects someone's goodwill in order to double their score and gain an advantage in the game. It forces players to choose between preserving their personal or professional relationships and gaining an advantage in the game, as game actions within the game's magic circle (Huizinga 1949) and actions outside of it collide.

Timer plays with the user by displaying random times whenever players pick up their phone – after all, they will have forgotten this information by the next time they check, so what does it matter what time the phone shows? It ironises the relationship between game actions and 'real' world actions, that is, between what lies inside the magic circle and what falls outside of it, in such a way that players are always losing in both.

4.3 Implementation

The pseudocode game *Thrower* would be easy to implement using the gyroscopes to determine whether or not the phone is moving. Would anyone actually want to play this game? Perhaps on a grassy field, where no harm could actually be done to the device by throwing it.

Imagine playing *Step Counter*: "I want to win this game, and I think how can I get this damn device to count steps while driving a car? I seek out bumpy roads to coerce the

eLyra 25, 06/2025: 99-115 – ISSN 2182-8954 | https://doi.org/10.21747/21828954/ely25a6

phone into oscillations large enough to count as steps. I take sharp, dangerous turns and hope the phone will drop onto the floor of the car so I can see the step count go up by +1." Each of these games play with the thrill of competition (at least for those willing to engage with our rules of play), setting clear objectives for winning a single – or multiplayer game.

Finder is easy to implement (at least the meow every two minutes) and might make for quite a fun social game, especially if everyone's phone had the same meow sound and players were penalised for finding someone else's phone. *Charger* too falls into this category quite nicely.

Robber's implementation is questionable because, as soon as players lose their phone, or even leave their phones, they no longer have access to the software. They could have access to another phone that could track the first one, but that would go against the idea of being without the phone and trying not to lose track of it without any aid – which takes us to the critique of how unaware and lost we become without a smartphone in our hands.

Timer may not be possible to implement because of the limitations that operating systems set on tweaking the time, unfortunately. Indeed, in the age of alarm clocks, microwave clocks, wristwatches, wall clocks etc., it was, or is, by no means uncommon for people to manually set their clocks ahead of time in order to trick themselves into thinking of time and time management in different ways.

4.4 Pseudo-code poetics as useless instruction for friction aesthetics

So, after all, what have we done here? We identified a problem — worrisome patterns of smartphone usage – and devised a solution: pseudocode poetics as a form of critique of software-as-instruction that influences our behaviour, alongside a first version of an 'application' (*useLessOS*). Is this, ironically, an example of the (techno)solutionism that Bridle warns against? Looking deeper, Bridle states that computational thinking drives "the worst trends in our societies and interactions, and must be opposed by a real systemic literacy" (2018). With this in mind, is our creative approach of pseudocode poetics (and accompanying app) less a technological solution and more an effort towards the literacy (that Bridle calls for) in computational thinking?

Folding back onto its lineage of turn-of-the-millennium codework, pseudocode poetics engages with the idea of the computational system and its use of language (*while, for, if/else, print*), translating software, which normally presents as the frictionless apps that seem to use us, into their underlying logics, loops and sequences, giving visibility to the interior workings of the computer. Could this stripping-down of software into computational processes, through pseudocode poetics, allow us to see behind the scenes of the hedonistic loops that subjugate us to spending our time with them? We have engaged with some of these questions in three dimensions – Tension, Critique, and Implementation – as a way to explore the extent to which pseudocode, as instruction for use, can evoke a possible mental scenario and to consider whether, and how, this kind of playful idea of a possible software can be used to critique our smartphone habits.

Moreover, the idea of *useLessOS* (the digital implementation of our pseudocode poetics) is to expose the aforementioned frictionless aesthetics in human-computer interaction by making the uselessness of the designed media ironically evident: in this case, an app that only presents pseudocode for inconvenient interactions with the smartphone. Future work will expand on this digital implementation of the pseudocode.

5. Conclusions, limitations and future work towards a world with friction aesthetics

The work discussed in this article uses pseudocode poetics to critically address the status quo of frictionless aesthetics in smartphone application design. We present six games that explore pseudocode as a form of writing that describes instructions for algorithms of imaginary pieces of software. Our main contribution lies in exploring the creative-critical potential of this form of writing. We reflect on how such instructions evoke scenarios and events in one's mind, and we analyse how these imaginary scenarios offer a critical perspective on our relationship with our smartphones.

This creative-critical exploration of aspects of our entanglement with smartphones is written from the standpoint of our *situated knowledge*; we acknowledge that there are many other positions from which people engage with this material relationship. In this contribution, our perspective focuses solely on the detrimental impacts of this relationship with technology (particularly through the paradigms of frictionless design). Our work could benefit from including the lived experience of a broader group of people, or from drawing on research into aspects of smartphone use as starting points for the creative writing of the pseudocode poetics. *useLessOS*, the digital application, could be populated with co-created pseudocode poetics. We could also engage with other people's experience of our poetics rather than relying solely on an autoethnographic and critical analysis of our own creative artefacts and process.

For us, this artistic intervention and provocation through pseudocode poetics has been, as outlined in the introduction, about creating an aesthetic disruption that invites the reader to pause and reassess their interactions with their phone. Would I, for instance, be able to play *Robber* (section 3.3), and why or why not? What does this reveal about my relationship with this material object? Does the "game" outlined in *Charger* (section 3.5) sound familiar? Might I find myself sitting in the stillness of my home, trying to listen to that faint "meow" in *Finder* (section 3.4), and how would that separation from the device feel? Does the game *Thrower* (section 3.1) make me cringe or feel at ease, can I imagine playing it with my own device? And again, what do these observations say about my relationship with this material object? This contribution paves the way for further and deeper engagement with the poetic potential of pseudocode, as instruction and as a means of critically exploring the ways in which software shapes our agency.

eLyra 25, 06/2025: 99-115 – ISSN 2182-8954 | https://doi.org/10.21747/21828954/ely25a6

NOTES

* Terhi Marttila holds a PhD in digital media from the University of Porto and is an integrated member at the Interactive Technologies Institute (ITI/LARSyS) and currently a postdoctoral fellow in the eGames lab research project. Terhi is an artist and researcher who appropriates programming, language, voice and game engines to make things that meander at their crossroads. Her work has been published in the Electronic Literature Collection 4, at ICIDS 2024, ISEA 2024, ELO 2024, ELO 2023 and ELO 2022, Revista Saca, taper#11, the digital review #3, The New River, nokturno.fi and raum.pt as well as at other academic conferences. In this contribution her interest with critical making intersects with Pedro's interest in the notion of friction in design. https://terhimarttila.com.

** Pedro Cardoso is a Designer, an Assistant Professor at the Faculty of Fine Arts of the University of Porto, and a researcher at i2ADS: Research Institute in Art, Design and Society. His work, in the scope of Art, Design, Interaction and Experience, develops in multidisciplinary contexts, focusing on the exploration of poietic and aesthetic processes of games and computational systems. His interests lie in their study as artefacts for creative thinking and innovation, and critical work for affective experiences and social intervention. He holds a PhD in Art and Design (2016), a MA in Image Design (2008), and a BA in Communication Design (2006). https://pcardoso.tumblr.com

[1] See the ELMCIP knowledge base for the entry "codework": https://elmcip.net/node/8439 . Accessed on 18/03/2025.

[2] See interview with Mez Breeze on Rhizome: https://rhizome.org/editorial/2016/dec/15/mezangelle-an-online-language-for-codework-and-poetry/ . Accessed on 18/03/2025.

[3] See works listed under keyword "codework": <https://collection.eliterature.org/1/aux/keywords.html>. Accessed on 18/03/2025.

[4] See digitally archived works in The NEXT: <https://the-next.eliterature.org/search?q=codework>. Accessed on 18/03/2025.

[5] Unfortunately, we were not able to access the original article by Alan Sondheim and for this reason resort to citing Raley. The reference for the original article is: Sondheim, A. (2001), *Introduction: Codework*, American Book Review, 22:6.

[6] Find ATTN in the app store: <https://apps.apple.com/sg/app/attn/id1478659497>. Accessed on 18/03/2025.

[7] Experience the work on taper: <https://taper.badquar.to/11/infinite_scroll.html>. Accessed on 18/03/2025.

[8] Experience the work on Ben Grosser's site: <https://bengrosser.com/projects/endless-doomscroller/>. Accessed on 18/03/2025.

[9] Visit project on Pippin Barr's site: <https://pippinbarr.com/it-is-as-if-you-were-on-your-phone/info/>. Accessed on 02/04/2025.

[10] The Light Phone company website: <https://www.thelightphone.com/>. Accessed on 02/04/2025.

Bibliography

Adams, T. E., Holman Jones, S. L., & Ellis, C. (Eds.). (2022), *Handbook of autoethnography* (Second edition), Routledge. <https://doi.org/10.4324/9780429431760>.

Baldwin, S. (2003), "Process Window. Code Work, Code Aesthetics, Code Poetics", In Markku Eskelinen & Raine Koskimaa (Eds), *Cybertext Yearbook*, University of Jyväskylä.

Bhargava, V. R., & Velasquez, M. (2021), "Ethics of the Attention Economy: The Problem of Social Media Addiction". *Business Ethics Quarterly*, 31(3), 321–359. <https://doi.org/10.1017/beq.2020.32>.

Brotchie, A., & Gooding, M. (Eds.) (1995), *A book of surrealist games*. Boston, Shambhala Redstone Editions.

Bridle, J. (2018), *New dark age: Technology and the end of the future*, Verso Books.

Cayley, J. (2002), "The Code is not the Text (Unless It Is the Text)", *Electronic Book Review*. <https://electronicbookreview.com/essay/the-code-is-not-the-text-unless-it-is-the-text/>.

Candy, L. (2019), *The creative reflective practitioner: Research through making and practice*, Routledge.

Cardoso, P., Melo, R., Carvalhais, M. (2020), "Breaking the Hedonistic Loop: Meaning before fun in videogames". *In Proceedings of the 9th International Conference on Digital and Interactive Arts (ARTECH '19)*. Association for Computing Machinery, New York, NY, USA, Article 52, 1–4. <https://doi.org/10.1145/3359852.3359902>.

Carvalhais, M., Cardoso, P., Raimundo, J., Melo, R. (2019), "Designing games that prioritize meaning over fun". *Contemporânea-Revista do PPGART/UFSM*. Volume 2, Issue 4. DOI: <https://doi.org/10.5902/2595523341302>.

Csíkszentmihályi, M. (1990), *Flow: The Psychology of Optimal Experience*, Harper & Row.

Greenfield, A. (2017), *Radical technologies: The design of everyday life*, Verso Books.

Huizinga, J. (1949), *Homo Ludens: A Study of the Play-Element in Culture*, Routledge.

Kuang, C. & Fabricant, R. (2019), *User friendly: How the hidden rules of design are changing the way we live, work and play*. WH Allen.

Krug, S. (2014), *Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability*. New Riders.

Malaquias, R., Cardoso, P. (2025), "Deception in Video Games: Nine Game Design Patterns". In: Martins, N., Brandão, D. (eds), *Advances in Design and Digital Communication V*. DIGICOM 2024. Springer Series in Design and Innovation , vol 51. Springer, Cham. https://doi.org/10.1007/978-3-031-77566-6_8

Raley, R. (2002), "Interferences: [Net.Writing] and the Practice of Codework", *Electronic Book Review*. https://electronicbookreview.com/essay/interferences-net-writing-and-the-practice-of-codework/

Roy, G. G. (2006), "Designing and explaining programs with a literate pseudocode". *ACM Journal of Educational Resources in Computing*, 6, 1 (March 2006), 1–es. <https://doi.org/10.1145/1217862.1217863>.

Sicart, M. (2015), "Dark Play: The Aesthetics of Controversial Playfulness". In Linderoth, J., and Mortensen, T. E. (eds.), *The Dark Side of Game Play*, Routledge. <https://doi.org/10.4324/9781315738680>.

-- (2023), *Playing software: Homo ludens in computational culture*, MIT Press.

Silva, I., Cardoso, P., Giesteira, B. (2023), Strategies of Intentional Friction in the User Interface of Digital Games. In: Martins, N., Brandão, D. (eds) Advances in Design and Digital Communication III. DIGICOM 2022. Springer Series in Design and Innovation , vol 27. Springer, Cham. <https://doi.org/10.1007/978-3-031-20364-0_5>.

-- (2024), "Friction Firestarter: A Toolkit for Designing Meaningful Friction in Game User Interfaces". In: Martins, N., Brandão, D., Fernandes-Marcos, A. (eds), *Perspectives on Design and Digital Communication IV*. Springer Series in Design and Innovation , vol 33. Springer, Cham. <https://doi.org/10.1007/978-3-031-41770-2_14>.

Soderman, B. (2021), *Against flow: Video games and the flowing subject*, MIT Press.