UNIVERSITY OF
WEST ATTICA
ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

Department of Graphic Design and Visual Communication

Graphic Arts Technology

Bachelor's thesis – "From skeuomorphism to Material Design – an evolution of user interfaces"

John Steven Farrington

Overseeing professor: Dr. Chrysoula Gatsou

Athens, 2023

Three-party examination committee

- **Chrysoula Gatsou**

Chrysoula Gatsou Digitally signed by Chrysoula Gatsou
Date: 2023.08.08 10:28:20 +03'00'

- **Philippa Athimaritou**

FILIPPA ATHYMARITOU Digitally signed by FILIPPA ATHYMARITOU
Date: 2024.10.03 10:38:15 +03'00'

- **Antigoni Karamani**

Antigoni Karamani Digitally signed by Antigoni Karamani
Date: 2024.10.01 18:49:52 +03'00'

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Ο κάτωθι υπογεγραμμένος Φάρρινγκτον Ιωάννης – Στέφανος του Άντριου, με αριθμό μητρώου 13006, φοιτητής του Πανεπιστημίου Δυτικής Αττικής της Σχολής Γραφιστικής και Οπτικής Επικοινωνίας του Τμήματος Τεχνολογίας Γραφικών Τεχνών, δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της πτυχιακής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολο τους με την πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από εμένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας μου, όσο και του Ιδρύματος.

Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου».

Ο Δηλών

Ιωάννης – Στέφανος Φάρρινγκτον

Aknowledgements

I would like to personally thank the members of the examination committee for their help and support during the completion of this thesis. I would also like to extend my heartfelt gratitude to Dr. Chrysoula Gatsou for overseeing this thesis to the end, despite any issues and missed deadlines that arose during its completion. I would also like to thank my family and friends for their help and support.

Abstract

User interfaces have been an ever evolving concept, with their roots going all the way back to the early 20th and late 19th century. In their earliest stages, user interfaces were much more about being able to control the mechanical parts of the then large and complicated computers of the time. As time went on, computers would change from mechanical to electronic, and a simpler and more intuitive user interface would be needed. Advancements in science and technology would give us the first examples of a command line interface. In order to make computers more approachable and usable for more people, the idea of abstracting applications and commands through the use of icons would take place in the 80s. This, along with the falling prices of computers, meant that the barrier to entry would come down for users. The increase in computer usage and the rise of internet websites, would cause the World Wide Web to evolve into what is known today as “web 2.0”. Design would become unruly and eye-catching in order to attract a user’s attention as much as possible. As technology matured, aesthetics would come to play a larger role in design. The new millennium would bring with it a refined and futuristic style, full of curves and clean surfaces. This clean style would then evolve into “skeuomorphism”, mainly due to the rising complexity and power of computers. Users wanted an interface that would mimic real-world items as much as possible while also offering a plethora of functions and selections constantly on the application’s desktop. This would prove tiring and taxing however, and simpler, “flatter” interfaces would start to dominate modern technology, with their look and feel being able to easily adapt from one form factor to another. The chief complaint of flatter interfaces, of course was the lack of information and feedback toward the user. This would eventually be addressed through the new style of interface known as “Material Design”, which would treat interfaces like a more three-dimensional piece of paper instead of a flat plane, while also using sound and animation to guide the user organically.

Key Words

Interface, command line, web 2.0, Y2K aesthetic, skeuomorphism, Material design, Flat design, usability, information density

Περίληψη

Οι διεπιφάνειες εργασίας είναι μια ιδέα με τις ρίζες της στα τέλη του 19ου και τις αρχές του 20ου αιώνα. Στις απαρχές των ηλεκτρονικών υπολογιστών, οι διεπιφάνειες εστίαζαν στην ικανότητα του χρήστη να μπορεί να ελέγχει τα μέλη των μεγάλων και σύνθετων υπολογιστών της τότε εποχής. Με την πάροδο του χρόνου, οι διεπιφάνειες εργασίας έπρεπε να γίνουν απλούστερες και χρηστικότερες, αντανακλώντας την μετάβαση από ηλεκτρικούς και μηχανικούς σε ηλεκτρονικούς υπολογιστές. Η τεχνολογία εξελίχθηκε στο σημείο οπού μπορούσαν να χρησιμοποιηθούν γραμμές εντολών. Στην συνέχεια, χρησιμοποιήθηκαν εικονίδια και σύμβολα ως μεταφορές για να μπορούν όλο και περισσότεροι χρήστες να χρησιμοποιήσουν εύκολα ηλεκτρονικούς υπολογιστές, προσιτοί πλέον λόγω τεχνολογίας και χαμηλών τιμών. Οι περισσότεροι πλέον χρήστες θα έφτιαχναν τις δικές τους εφαρμογές και ιστοσελίδες στο διαδίκτυο, με άτακτες και ιδιόρρυθμες συνθέσεις, για να τραβήξουν την προσοχή του χρήστη. Οι αλλαγές αυτές έλαβαν το όνομα “web 2.0”. Μια ωρίμανση κατά την διάρκεια της επόμενης χιλιετίας έδωσε στις επιφάνειες εργασίας τις χαρακτηριστικές τους καθαρές λευκές επιφάνειες και καμπύλες. Η απαίτηση πολλών χρηστών να έχουν πρόσβαση σε περισσότερες λειτουργείες και η επιθυμία τους οι επιφάνειες να αντανακλούν αντικείμενα της καθημερινότητας έδωσε προβάδισμα στον «σκευομορφισμό». Η πυκνή παρουσίαση του σε συνδυασμό με την τεχνολογική εξέλιξη, σήμαινε πως οι σύγχρονες, «επίπεδες» επιφάνειες εργασίας θα κυριαρχούσαν, με την ικανότητα τους να ανακατατάσσονται χωρίς πρόβλημα από συσκευή σε συσκευή. Η έλλειψη πληροφορίας σε συνδυασμό με απούσα πολλές φορές ανατροφοδότηση για τον χρήστη, οδήγησε στην δημιουργία του “Material Design”, ένα νέο είδος διεπιφάνειας εργασίας, που χρησιμοποιούσε ήχο, εικόνα και κίνηση για να δώσει στοιχεία στον χρήστη, ενώ παράλληλα διαχειρίζονταν την επιφάνεια της ως ένα τρισδιάστατο φύλλο χαρτιού. Μέσω αυτών των ιδιοτήτων ο χρήστης καθοδηγείται οργανικά στον στόχο του.

Λέξεις Κλειδιά

Διεπιφάνεια, γραμμή εντολών, δίκτυο 2.0, αισθητική του 2000, Σκευομορφισμός, Material design, Flat design- Επίπεδη Σχεδίαση, χρηστικότητα, πυκνότητα πληροφορίας

Index

- The Y2K aesthetic in the computing mainstream.

Chapter 6 – Skeuomorphism

- What is skeuomorphism?
- A brief history of skeuomorphism
- “feel” in user interfaces
- Skeuomorphism in user interfaces
- An example in WinAMP
- Why did skeuomorphism fall out of favor?
- Skeuomorphism today

Chapter 7 – Flat interfaces

- What is a flat style interface?
- Responsive design for user interfaces.
- Flat interfaces.
- Elements of flat design.

Chapter 8 – Material Design

- What is “Material Design”?
- A quick background.
- The principles and innovations of Material Design.
- Material Design is not a revolution, but rather a refinement of existing technology.

Conclusions

The aim of this thesis

The aim of this thesis is to illustrate the evolution of user interfaces used in computing from the inception of the computer to the modern era. It begins from the earliest electronic computers from the 50s and 60s. It will then move on to the user interfaces of the 70s and 80s, right when computing moved from the sterile and professional laboratories down to consumer grade and consumer level hardware. From that point on this thesis will move to the 90s and early 00s, at which point computers had become a staple and part of every home and office, with their technology being revolutionized and democratized. Personal styles and touches were also beginning to be used, as computers began to be used more and more as a medium to express personal taste and styles, along with the so-called “Y2K aesthetic” which is fondly remembered by many to this day. Further on, as technology moved from the computer to handheld devices, we will see how user interfaces evolved during the 00s and 10s, when the concept of the smartphone became common knowledge and commonplace to the average user. The final evolutionary step of the user interface will be the modern takes of material design and flat design. A user interface design aesthetic that takes advantage of modern large high resolution screens and smart devices, while at the same time keeping distractions to a minimum, exposing the user to only the needed information through a very clean and filtered user experience. Finally, this thesis will end with a case study in the evolution of a well-known application which is used to this very day, Microsoft Word. The case study will look at its very first iterations as an application on DOS to its most recent evolution on Windows computers and as an application on smartphones.

Introduction and background

The earliest computer user interfaces would make an appearance in the early 50s and 60s, as computers were just beginning to emerge. They were less user interfaces and much more control panels, using devices such as levers, buttons and punch cards to perform simple tasks. A user interface that would project updates and be visible by the user would take many years to develop further. Computers back then were still the realm of scientists and engineers, and required specialized training that would take years to fully materialize. They would only be used sparingly, and most of the time due to their size, which was often as large as multiple rooms, in conjunction with their mechanical and electronic complexity, they would often require full teams of scientists in order to operate them properly.

As computers increased in complexity and decreased in size, they became a viable option to perform tasks and would slowly become incorporated into many companies and organizations. The rapid evolution of technology would allow for more functions to be performed by these machines, and as such a different method of control would have to be developed. The earliest modern computers would use a control method dubbed the “command line interface” which consisted of lines of code, or “commands” being typed through a keyboard on the computer’s display. This allowed the user full control of these commands, while also providing immediate feedback to them. At the same time, the user would be able to view exactly what they commanded to the machine, and as such, they would be able to remedy any mistakes and catch errors before the execution of the command. These command line interfaces would form the backbone of modern computing as we know it, and are still used in low level application control and legacy systems to this very day.

Command line user interfaces were used and are still used to this day, and despite their simplicity in comparison to the mechanical controls of the past, still posed a considerable learning curve for the average user. A new way of interacting with the machine was the graphical user interface or GUI. GUIs would use iconography in order to illustrate the functions of the computer more as a desk or work area than a list of commands. Through icons and colors and symbols as well as the desktop metaphor, the barrier to entry was significantly lowered and more users could accomplish more in a relatively smaller timeframe with much less training. The first graphical user interface would find itself in the Xerox Alto system in 1973, which would further innovate buy using a human- computer interface to control its function. This device would be more commonly known as the “mouse”.

Graphical user interfaces would further develop in the coming years. As technology progressed, more elements, higher resolutions and a wider gamut of colors could be displayed on computer monitors. Technology would eventually become powerful enough to display multiple colors and shapes and even gradients of the same color. At that point in time, user interfaces would incorporate the use of colors and symbols. By today they would be bright and garish, using colors such as magenta, cyan, greens and yellows. Their main purpose was to direct the user’s attention to the interactive parts of the display, while simultaneously having to make do with the best they could, due to the inherent limitations of the technology of their time. It would take a few years for user interfaces to be able to be designed more freely, without much concern with the performance impact of the design on the computer’s hardware.

Computers would eventually become a staple of both offices as well as homes. The computer would become known less as a work machine, and more as a personal computer (PC). The more users would receive and use computers, the more evolved and eclectic the styles became. As computers moved away from a centralized and rigid structure, the technology would become accessible and democratized to more people. Along with the advent of the World Wide Web and its adoption by millions in the coming years, the new aesthetic of user interfaces would become known as "Web 2.0". A much less restrained and free type of design, which would place aesthetics and eye catching design as its main goal. Usability was less of a concern. The aim behind that aesthetic was to make it as eye catching as possible

Streamlining this sort of design aesthetic would lead to the well-known "Y2K design aesthetic" which is fondly remembered to this very day. It mostly consisted of clean solid colors with glossy or mirrored effects made to emphasize order, lack of clutter and a look to the possible future of technology. Transparency became a large part of this design aesthetic, with interfaces becoming much more evidently three-dimensional and interactive in comparison to the relatively simplistic and barebones two-dimensional interfaces of the recent past. These interfaces were cleaner, more orderly and often reflected the thinking that computers were much less machines and much more a part of everyday society and life. Their three-dimensional nature was an attempt at further removing the two-dimensional boundary between the user and the user interface, allowing them to fully experience the benefits and advanced technology of their personal device. This user interface style would not be solely bound to computers, but also found in the then budding video game console market.

During this emergence of clean user interfaces, the exact opposite aesthetic was also beginning to flourish in many applications. Instead of going the way of clean and shiny surfaces, many user interfaces would incorporate sliders, knobs and buttons into their design. This aesthetic would come to be known as skeuomorphism, and would be found in a large amount of enthusiast as well as more commonly used applications. This style of user interface, while not necessarily being more in depth or complicated than other user interfaces, would offer a much more “hands-on” aesthetic than other user interfaces. The so called “feel” of the user interface would be what set it apart, and allowed it to become so popular. It was often favored by many users who required a lot of control on their technology. The entire idea of skeuomorphism was not modern, but was rather based on ideas of the distant and not so distant past. It would last for many years and even find a home in many creative applications (Evans, 2013).

Skeuomorphism would eventually give way to the cleaner user interfaces we use today. A large complaint in regards to its design would be its cluttered appearance. Technological advancement and the universal adoption of smart devices and touch screens meant that not only less people would come to use desktop computers for their applications, but the fact that skeuomorphism produced interfaces with inherently small touch targets, made that interface increasingly less usable and more obsolete. The answer would come in the form of the now well-known flat user interface.

A flat user interface is a user interface which consists of mostly flat shaded or single color interfaces, tabs and buttons which scale easily to multiple devices, resolutions and even can be easily interacted with using touch, as well as devices like styli and mice. They would be often found in smart devices. The mantra behind their design was clarity, usability and simplicity. In stark contrast to the former user interfaces, these interfaces would instead be as simple and understandable as possible, only showing to the end user what they are meant to interact with, while hiding any sort of unnecessary information. Large and simple icons, bold and simple fonts and sparse use of color would come to characterize these user interfaces. Since the majority of modern users would come to own some sort of smart device with a capacitive touchscreen, the focus would shift in UI design. Instead of focusing on machines with a separate keyboard, mouse and display, touch interactions became the priority. The big icons, large spacing and clear touch targets would allow for little confusion when the user would need to interact with any part of the screen. On top of that, the large and clean touch targets would help in avoiding the so called “fat finger error” in which the user makes a mistake while interacting with the touch screen interface.

Touch screens and touch devices would eventually become larger and much more usable and user friendly. The larger size would allow for larger, clearer and more aesthetically pleasing user interfaces. Around this time, touch interactions apart from pressing and holding would steadily trickle down in most user interfaces. Two-finger scrolling, as well as pinching and spreading to zoom in and out, respectively. The touch device would take over as the primary method of interacting. Keyboards and styli for resistive touch would remain, but be used less in day-to day usage. That isn’t to say of course that they were used less in a professional environment. Most professional or power users would -and still do- use these to input almost all of their commands, preferring their tactile nature.

This clean and flat user interface aesthetic would allow for very little experimentation and creativity. This did not mean, however that certain styles would not emerge. Many companies would create their own design language during this time. They would keep the positives of the simple flat user interfaces, while also adding their own design flourishes. This would also allow their applications to develop an identity of their own, while still being user friendly and scalable to many devices and operating systems. Due to their popularity, certain elements would eventually evolve into design staples that we use to this day, and will probably continue to use for the foreseeable future.

One of these examples is the material design language used by Google. During the second half of the 2010s, a heavy emphasis was placed into creating a solid, identifiable and aesthetically pleasing user interface. As Google was the creator of the most popular and widely used smartphone operating system, Android, their style would be promoted on almost all smart devices. Using texture, bright, complementary colors, subtle shadows and the now popular CTA (call- to- action button), they were able to carve out an identity of their own. Perhaps the most important part of this user interface was not the aesthetic itself, but rather their new interaction buttons. The three dots or “meatballs”, the “hamburger menu” would find themselves in most modern user interfaces. On top of that, clear guidelines in regards to the creation of applications with said user interface would be freely available for users to download, allowing many an aspiring designer and programmer to become acquainted with modern design sensibilities.

Chapter 1- introduction to user interfaces

Punched cards, or why user interfaces were so needed.

Before we move on the earliest introduction and usage of user interfaces, we have to talk about how computers and commands worked before they were introduced. Along with the mechanical inputs of mechanical computers of the late 19th and early 20th century, punched cards would come to be used as some of the earliest forms of input as well as representations of code (Maxfield, 2011). They would be divided into columns and rows, and each element in these columns and rows would be punched through. This would in turn be "fed" into the machine, which would read it and perform its given task.

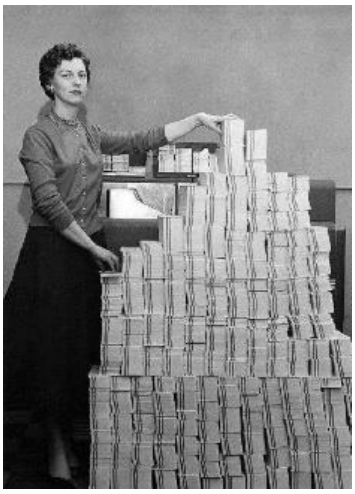


Figure 1, Woman standing in front of the equivalent of 4.5 megabytes of data in 62,500 punched cards, 1955

A very difficult and obtuse method, especially for people who were just beginning their careers with the use of computers. Since each and every machine had many mechanical components, each one was also slightly different from each other, making these slight differences a learning requirement for proper usage (Rounds,2016). While small changes from machine to machine may not have been noticeable at first, once they added up, they would increase workload and training time without offering increased productivity in return. As such, simplification and standardization would be some of the most important issues that would need to be solved.

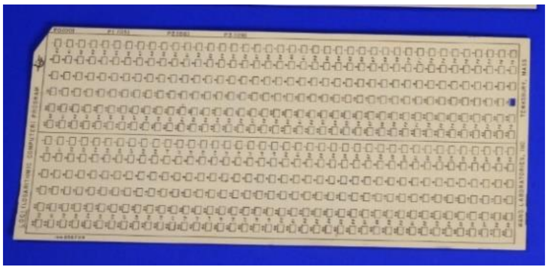


Figure 2, LOCI-2 punch card, Smithsonian Institute, 1965

The need for standardized ways of working.

As mentioned above, each machine was slightly different from each other back then. Each computer had slightly different instructions, a slightly different control panel, and slightly different ways of functioning. Punch cards and mechanical input from experienced users and operators would be enough in the beginning, but as time went on and computing became a real and viable option to more traditional work, more people would have to be trained in order to increase the available workforce, output and amount of machines that could be operational at the same time. This shift would necessitate that machines had to stop being a boutique cottage industry, and instead move to a more standardized and streamlined way of operating, one which would also reduce training for future operators.

The greatest obstacle against this shift would be the available technology. Each and every command would take very long not only to be input, but also to be completed. At the same time, not being able to see the command that was being input, made it vulnerable to many mistakes. An errant character or letter, a spelling mistake would need many hours to rectify and perform the task from the beginning, correctly, this time. It would often feel akin to stumbling in the dark for many users.

The shift to electronics and electronically operated user interfaces.

Technology would progress, and computers with it. Displays and networking would slowly enter the field and slowly start to solve some of these problems. A full display would allow the user more control. Around this period, an invention of the past would find itself in the forefront of technological evolution.

The “Teletype” was a machine designed in the mid-19th century. It was one of the first examples of electric engineering, and was used to transfer short messages through point-to point and multipoint configurations. In the beginning it would transfer signals, akin to 0s and 1s. These signals would then be printed on a piece of paper and in turn deciphered into words and sentences. It would take a long time for the teletype to be able to transfer full texts from point to point, but once it became able to do so it was used in many offices and companies, many of them both public and private. It would be an invaluable tool and would be used to issue commands to workers from long distances Akin to a much more efficient form of telegraphy (The Teletype Corporation, 2021).

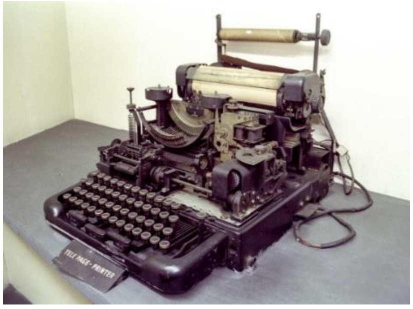


Figure 3, Teleprinter. Image by Biswarup Ganguly September 15, 2018

The logic behind the teletype would inevitably transfer to the computers of the era. Commands would be issued from operator to operator and machine to machine using a system of similar logic. The messages would be displayed on a screen and would be visible to the operator. The result of the command being typed would be visible in real time. A first for technology of the era.

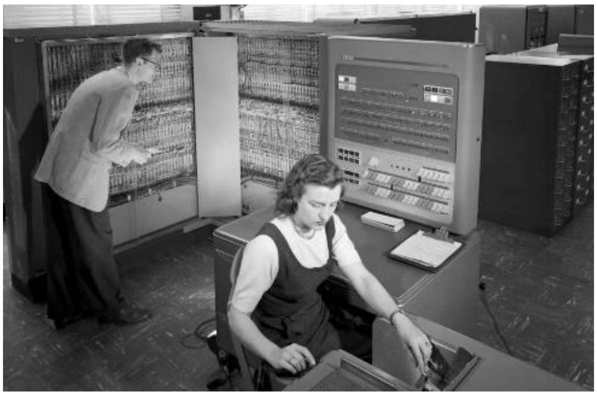


Figure 4, Conducting aeronautical research with an IBM 704 in 1957. NASA

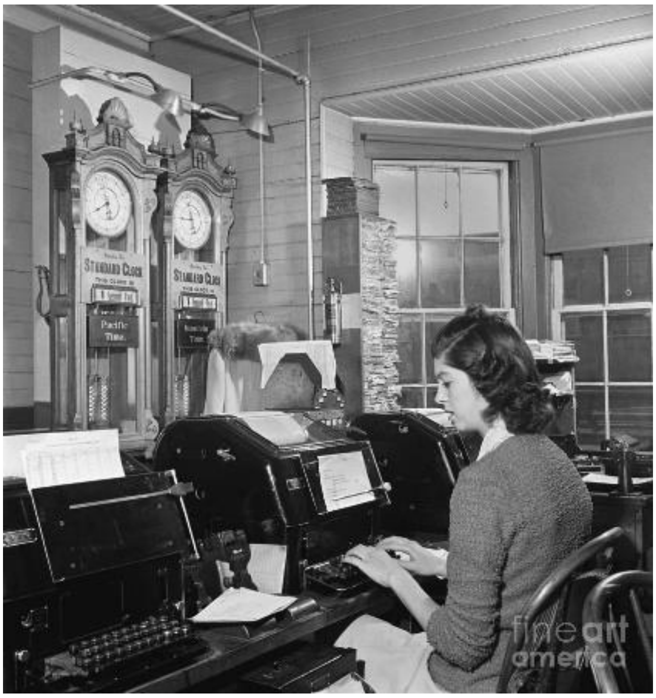


Figure 5, Teletype Operator.

This leap in logic and computing would move us to the first step of user interfaces, **the command line user interface.**

Chapter 2 - Command line interfaces

The introduction of command line interfaces.

More than half a decade would come to pass before the implementation of graphical user interfaces. Before they would become the norm, Computer terminals were interacted with through the use of command line interfaces, or CLIs. Command lines were used almost exclusively up until the 1980s, when graphical user interfaces would slowly find their way into more and more computers, thus enabling more users to interact with said machines.

Command line interfaces would come about in the late 40s. They served as a replacement to the Teleprinter system (TTY) which was in use from the later years of the 19th century. As technology improved, they would output on a display. The appearance of the command line is very simplistic. It is comprised of nothing more than a white string of text on a dark background. All of the commands are input by the user, using standardized vocabulary.

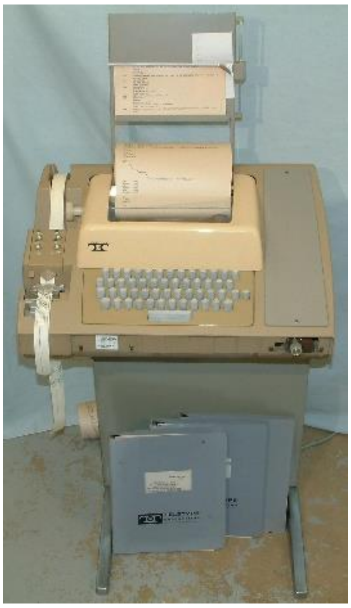


Figure 6, ASR Teletype machine used to interact with IBM mainframes. One of the first command line interfaces, displaying on a sheet of paper

A command line interface is a shell interface. A shell is a computer program that is used as a means to gain access to the core of an operating system's functionality. This method is used as a way to bypass the heaviness and complexities of a graphical user interface, allowing for only the bare essentials to be loaded in the machine's memory. This was invaluable, especially in the earlier days of computing, when performance was often limited and any processes would mean a large investment of both time and system resources.

Due to their rather obtuse nature, command line interfaces were phased out in the following years, allowing more users to perform more tasks without extensive background knowledge on informatics or computer science. The revolution of the desktop metaphor made command lines redundant to the masses, and as such they are very rarely used as a primary mode of interaction between humans and computers today.

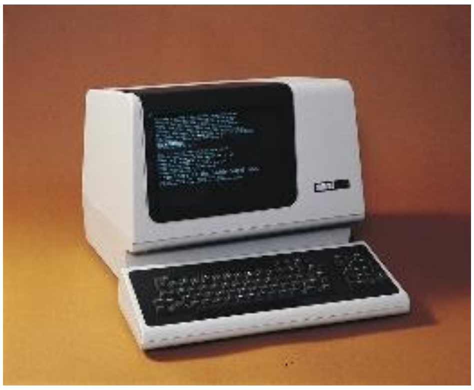
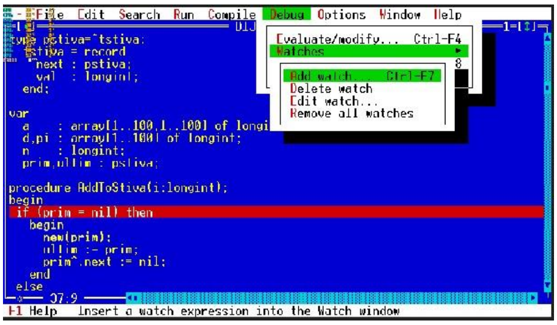


Figure 7, The VT100, terminal, introduced in 1978, displaying characters and text on its interface



```
type pstiva=^tstiva;
  tstiva = record
    next : pstiva;
    val  : longint;
  end;

var
  a    : array[1..100,1..100] of longi
  d,pi : array[1..100] of longint;
  n    : longint;
  prim,ultim : pstiva;

procedure AddToStiva(i:longint);
begin
 if (prim = nil) then
   begin
     new(prim);
     ultim := prim;
     prim^.next := nil;
   end
 else
```

Insert a watch expression into the Watch window

Figure 8, A typical command line interface, (CLI)

Command lines are still today, in one form or another. Many advanced computer users prefer them over the simplified and streamlined experience of today. Notable examples of the command line still in use today are the Windows PowerShell interface, Apple's Terminal and even the adb debugging tool that is used in devices running android.

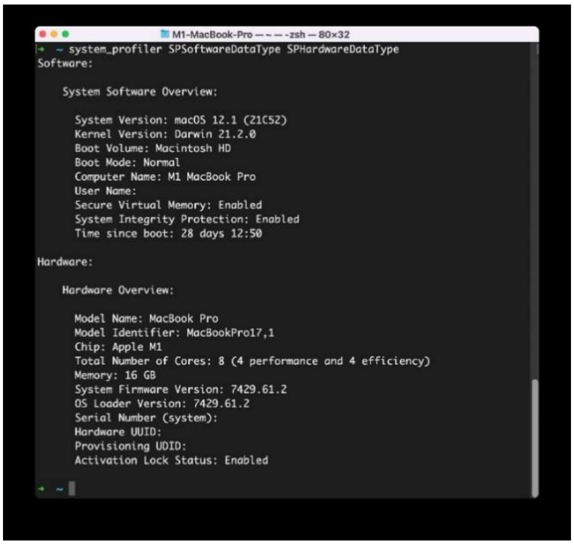
What is a command line interface? How does it work?

As mentioned before, a command line interface is essentially a blank screen with some characters on its top left side. These characters reflect the commands that the user inputs in said interface. These commands allow the user to perform a variety of processes, thus allowing him to control the machine directly. In order to control the command line interface, a keyboard is used as the input device. Almost all of the keys are used in different commands, and often special keys exist in order to insert special characters or special symbols.

The screen of the machine which is operating via the command line interface is very spartan. It's almost always displayed in monochrome, and unless an application is actively running or a file system is being accessed, the screen is blank in its entirety, save for a small flashing cursor. The cursor is used to indicate the line that can be edited by the user in order to perform any actions. The characters are often bright white and displayed in a low resolution (for today's standard) font.

Once a user has input their command, the user interface will usually scroll downward and display any sort of feedback toward the user in text form. This serves as an indication to the user of what they have done or whether what they have done has been successful or not. Alternatively, it may show an outline of the computer's file system and any sort of data relating to directories or sub files. These are often separated with the backslash symbol, which shows the file path.

Once the user has chosen which application they wish to launch, all they have to do is type the corresponding command and begin their work. Once an application is launched, the user interface changes to reflect this. The app's name is usually displayed on the top right of the screen, with a name, date and publisher. Very rarely the screen's color may shift in order to display a different one, in accordance with the launched app. This is a feature that was not always available, and when available, was often available to few computers of the time (something that we shall analyze further down the line). Depending on the program, some graphical elements may also show, however they are rare, and would have required rather powerful hardware in order to display properly back then.



```
→  ~ system_profiler SPSoftwareDataType SPHardwareDataType
Software:

    System Software Overview:

      System Version: macOS 12.1 (21C52)
      Kernel Version: Darwin 21.2.0
      Boot Volume: Macintosh HD
      Boot Mode: Normal
      Computer Name: M1 MacBook Pro
      User Name:
      Secure Virtual Memory: Enabled
      System Integrity Protection: Enabled
      Time since boot: 28 days 12:50

Hardware:

    Hardware Overview:

      Model Name: MacBook Pro
      Model Identifier: MacBookPro17,1
      Chip: Apple M1
      Total Number of Cores: 8 (4 performance and 4 efficiency)
      Memory: 16 GB
      System Firmware Version: 7429.61.2
      OS Loader Version: 7429.61.2
      Serial Number (system):
      Hardware UUID:
      Provisioning UDID:
      Activation Lock Status: Enabled

→  ~ 
```

Figure 9, Mac Terminal program used today, Apple computers, 2023

The downsides of a command line user interface.

As often as it is romanticized and used even today, the command line user interface was very much a product of its time and as such served a very particular purpose. It was designed to be as light and efficient as possible, while allowing full and complete access to the system by any user and simultaneously being able to display as much info as possible.

During the height of the command line user interface screen real estate was at a premium. As large as they may have been back then, computer monitors had small displays relative to their size and weight, and the low resolution meant that the information displayed would have to be as tightly packed as possible. Dedicated graphics processing units didn’t exist back then, and the simple act of scrolling the screen was a big deal. As such, these interfaces solved this problem by displaying as much info as possible on the screen at the same time. More often than not, the display looked more like a densely printed book page instead of what we may recognize today as a computer program.

Command lines were also characterized by lightness. As mentioned earlier, computer performance was at a premium and could not afford to be wasted through the display of intricate pictures or user interfaces. A command line user interface could be refreshed easily and be adapted for a multitude of applications.

Part of the problems of command lines though came from their simplicity. It was a deliberately simple exercise with form following function. Make it as light, and simple as possible, so that it could suit many needs and many different applications.

Chapter 3 - Graphical User Interfaces. The revolution of icons and iconography.

As mentioned above, command line interfaces, while bringing with them a great many positives, would eventually become difficult to learn and more importantly, to teach to younger professionals. Not only would they need to learn quickly, they would have to learn a great many commands, all while having to input them in through a relatively opaque and obtuse system.

Icons in user interfaces

Figure 10, Ferdinand de Sassure

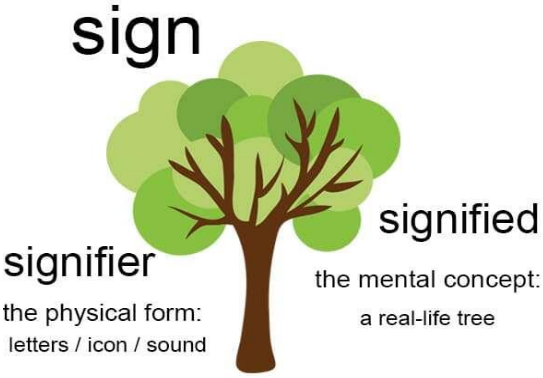


Figure 11, Signifier and signified, de Sassure

Before we move on to icons in user interfaces, one must first understand exactly what an icon is. In general, when referring to **icons in computing, we refer to any simple graphic or shape that is used to represent an application or a function of the operating system**. These icons are used through the process of semiotics in order to explain a function to the user (Atkin, 2005), (Ron, 2014).

Icons, in a broader sense, work through the idea of the signifier and the signified. Through a signifier (in this case the icon that the user clicks on), a signified process will occur (in this case launching an application or performing any other function on any operating system or computer (Gatsou et al., 2012). This metaphor has persisted to the present day, massively lowering the skill ceiling of user interfaces, and allowing them to be used easily and understood quickly, something that would be completely unheard of just mere years ago (Norman, 1999). From the oldest computer to the most bleeding edge smartphone of today, Icons are used and will still be used for the foreseeable future, as the metaphor which they project is the most simple and user – friendly way of performing tasks.

Icons have also made themselves part of the cultural zeitgeist. Examples of icons being part of our everyday lives can be found in the idea of using a diskette or a CD to perform the file save function, or a trashcan icon in order to indicate deletion of a file. Even though these icons are not necessarily part of our modern lives, especially with the advent of solid state media as a way of reading and writing files to.

They still persist as ubiquitous and always used signifiers of interaction.

The Xerox Alto

In March of 1973, Xerox would introduce its newest machine to professionals and consumers. It was a large machine which cost thousands of dollars to buy and operate, despite being available as a tool to the average buyer (Clement, 2002). It operated through the use of a keyboard and a radical new input device, whose task was to point and translate two-dimensional motion relative to a surface into computer language. These days it is known as a mouse.

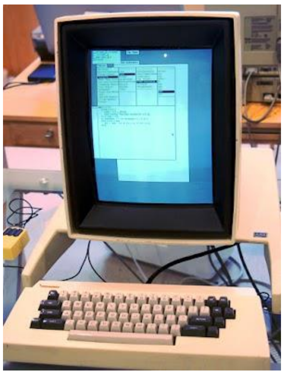


Figure 12, Xerox Alto running the Smalltalk-76 software

What really stood out about the Xerox Alto however, was not its size or price or shape, or even the mouse, what really stood out about this machine was its user interface, or rather, lack thereof (McJones, 2014). A command line user interface was not used, but rather a series of large rectangles, which had many labels and small pictures inside of them. The user could use the mouse and its buttons to move a large pointing arrow on the screen and then select them through clicking. These rectangles and pictures would earn a new name as time went on. They would be known as "windows" and "icons". The large pointer on the screen would be known as a "cursor"

Through the use of these graphics, this interface would drop the "command line" name. Instead it would be known as a GUI, or *"graphical user interface"*.

As Xerox developed this user interface, they would also develop another type of technology. **"WYSIWYG" or "what you see is what you get", is any type of editing software that allows the user to see and edit content as it appears when**

displayed on the interface. This would be a fundamental change for user interfaces and computing, as functions that would be executed through applications would be instantly visible and could be manipulated by the user as they happened. It would allow the operator to "see" what would happen, instead of having to wait for it to happen.

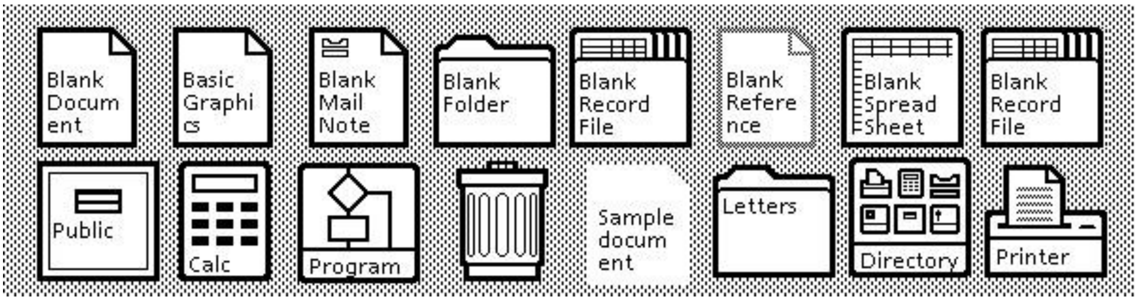


Figure 13, 1981 Xerox 8010 Star icons

The Xerox Alto never sold incredibly well, and due to its exorbitant price (over 100.000$ in today's money, with an accompanying printer costing almost three times as much) it could only be bought by research industries, laboratories or universities. In order for the graphical user interface to break into the casual user, the cost and complexity would need to be scaled down and made much cheaper (Wadlow, 1981).

The Apple LISA

In the late 70s, California's Silicon Valley was well aware of the revolution that graphical user interfaces would bring to the table. One of the first companies to try and bring them to the consumer would be Apple. The GUI was so important to Apple in fact, that in order to try and understand it better and receive instructions and documentations for its engineers, Apple would offer options from its stock to Xerox. After multiple visits to Xerox's PARC, Apple's engineers and the CEO Steve Jobs, would use the concepts to fuel their own development of their first GUI-driven computer, the Apple LISA.

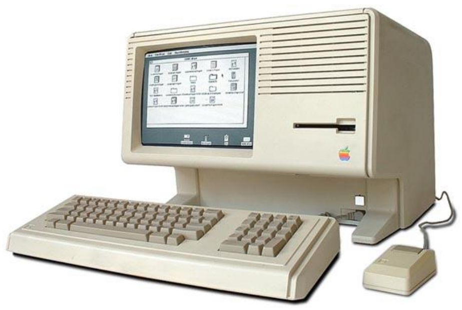


Figure 14, The Apple LISA with its GUI and computer mouse

Launched in 1983, the LISA would scale down the complexity of the user interface of Xerox and avail it to the average user. It would use a combination of a mouse and keyboard to allow the user to click on icons in a graphical user interface. It would be one of the first machines to do so, and it would be both a work and entertainment machine. It would perform all of these tasks for less than a third of the Xerox Alto's Price (Smith, 1988).

The LISA never sold well, however. So big was its failure in fact, that it would cause Jobs to be ousted from Apple. The complaints would mostly be about its price and performance, though. Retroactively, the LISA would come to be known as a bold, albeit unsuccessful experiment, with its GUI being regarded as one of the most influential developments in personal computing.

Graphical User interfaces of the era.

The LISA and the Alto would not be the only machines with user interfaces, of course. As the idea of the GUI slowly took over the computing world, many user interfaces would start to show in more and more operating systems and applications.

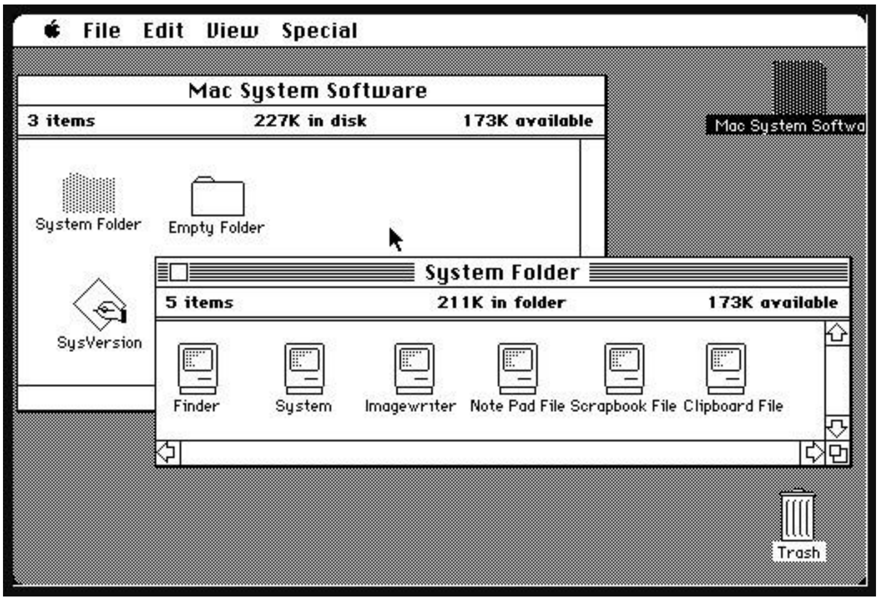


Figure 15, Apple's monochrome GUI in 1984

GUIs of the time would remain mostly similar to the original ideas of the time. Desktop environments would consist of large floating rectangles nicknamed windows. These windows would show expanded views of folders or of applications that were running. Within those windows would be other icons, not just for running applications, but also for basic navigation. Arrows and sliders would begin to make their appearance then, along with buttons for expanding, selecting, minimizing and closing windows and terminating processes. As computers increased in power and complexity, multitasking became more prevalent. While the earliest user interfaces with windows would not allow a user to switch between multiple running applications, but rather allow him to have them open on a desktop and remain static, these moveable windows could be moved freely and scaled on the screen. As computer RAM and ROM increased, the user could have multiple instances of applications running, and could freely switch between them (McJones, 2014).

In regards to looks, the GUI was a time of experimentation, albeit a rather limited one. In the beginning of their implementation, computers were often underpowered and could not display high resolution images, much less images with color. The earliest GUIs would be monochrome and use elements of shading such as monochromatic gradients or patterns to draw the user's interest.

As computers became more powerful however, color became a viable solution. In the earliest days of the GUI, developers often had to make the choice between colors or resolution. A computer's CPU couldn't handle displaying colors at a high resolution. Graphic cards, or GPUs, or CGAs (as they were called back then) would begin to tackle this problem. They were dedicated hardware which could output either an array of colors, or in a high resolution monochromatic mode. The monochromatic mode would mostly be used for office work.

Due to the experimental nature of many of these CGAs (color graphic adapters) there would be no standard system. Different color palettes would be used to display images, often with variable levels of success and accuracy. Often the user could switch between these different display modes. In the late 80s and early 90s, the modern additive color display model of RGB would be universally adopted by almost all manufacturers, leading to a simpler and more streamlined output.

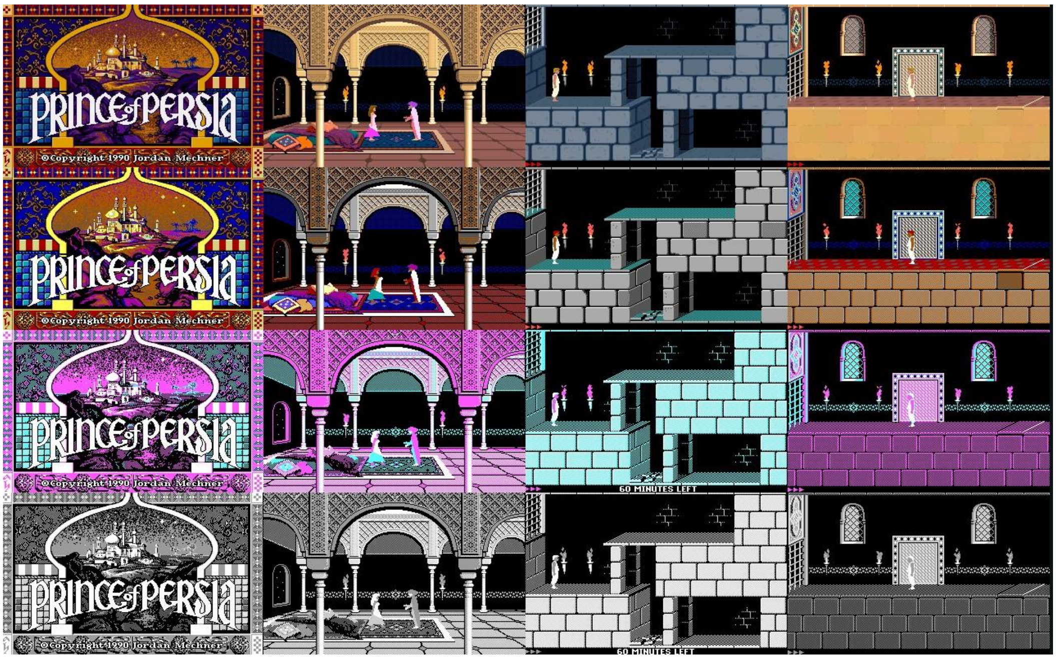


Figure 16, CGA, VGA, EGA, RGB color in Jordan Mechner's Prince of Persia, 1990

Chapter 4 – Web 2.0.

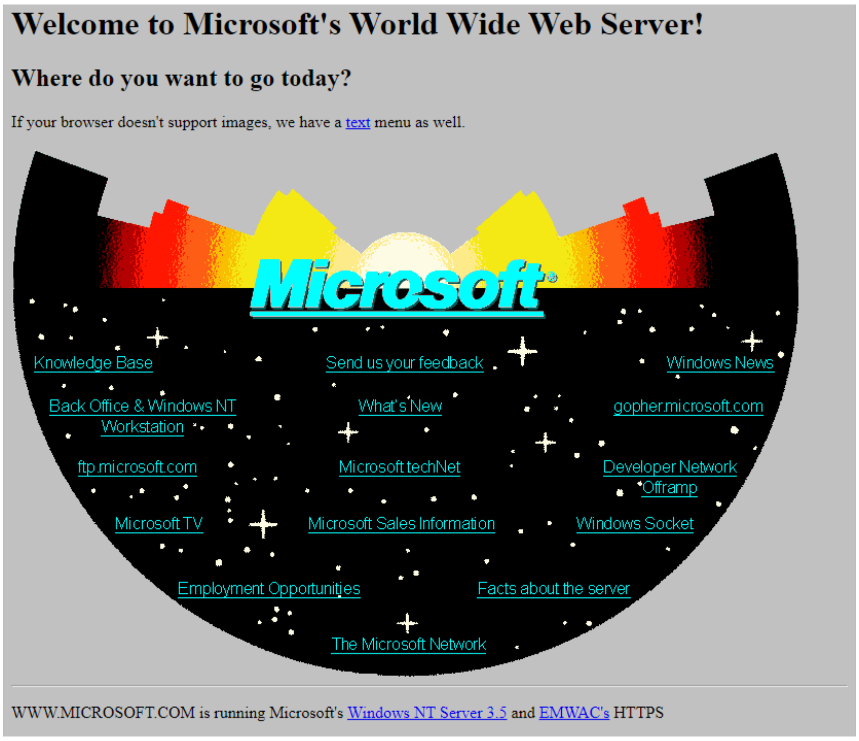
A shift in logic

Web 2.0 interfaces started showing up as a logical evolution of The GUI. They are often cited as the first step toward the modernization of applications. Computers would slowly but surely take on a new form and become heavily oriented toward consumers. Computers would not be the sole beneficiaries of the turn toward this more modern type of interface. The rapid technological evolution of the first half of the 2010s meant that more and more devices would make their way to the hands of the average consumer. Whether it may be an MP3 player, an early generation smartphone or even a gaming console.

After the utilization of computers and terminals as strictly professional pieces of equipment, Web 2.0 interfaces began appearing as the evolution of previous user interfaces. They are often cited as the first step toward the modernization of applications. Computers would slowly but surely take on a new form and become heavily oriented toward consumers.

The term “web 2.0” is often used when describing the state of the internet and the World Wide Web from the late 90s and early 00s until the late 00s and the beginning of the 2010s. When talking about web 2.0, we usually refer to websites that feature user-generated content, ease of use, participatory culture and interoperability. Web 2.0 marks a great shift from the sterile and utilitarian user interfaces of the 90s and 80s and into a new era of colors, shapes and graphics.

Web 2.0 interfaces would be characterized by a plethora of colors, shapes and sounds. Using all of the available tools, the then budding internet user community of the era would in turn create a multitude of various and stylistically different user interfaces.



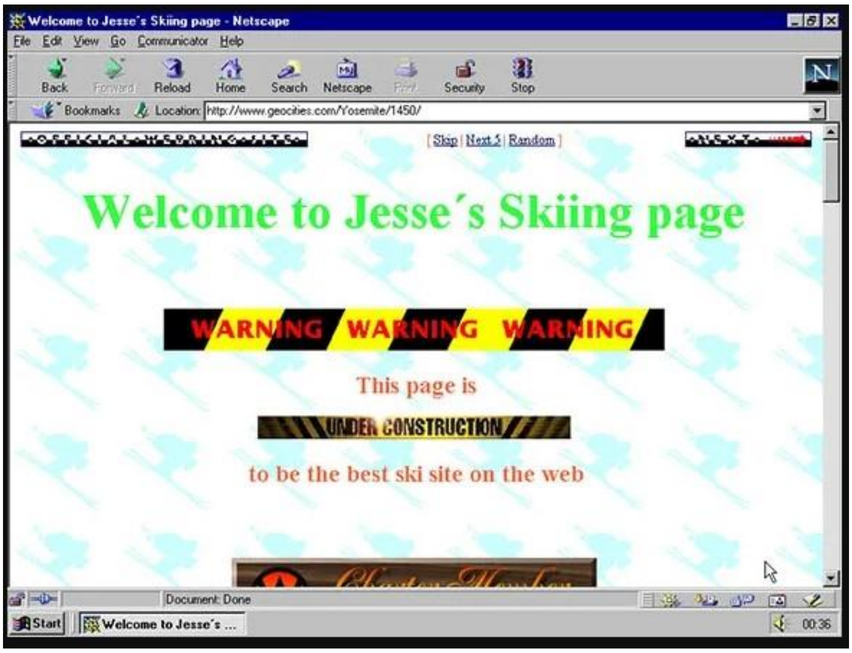
Welcome to Microsoft's World Wide Web Server!

Where do you want to go today?

If your browser doesn't support images, we have a text menu as well.

WWW.MICROSOFT.COM is running Microsoft's Windows NT Server 3.5 and EMWAC's HTTPS

Figure 17, Microsoft's early attempts at a homepage on the internet.


Welcome to Jesse's Skiing page - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Stop
Bookmarks Location: http://www.geocities.com/Yosemite/1450/
[Skip | Next 5 | Random]
Document: Done
Start
Welcome to Jesse's ...
00:36

Welcome to Jesse´s Skiing page

WARNING WARNING WARNING

This page is

UNDER CONSTRUCTION

to be the best ski site on the web

Figure 18, A user generated webpage on Netscape Navigator

The advent of web 2.0 also marked the first step of user generated content. This influx of user generated content would necessitate the creation and maintenance of platforms that would be capable of hosting and having the ability to edit and create user generated content. This is a very important part of web 2.0, which we will analyze below in detail.

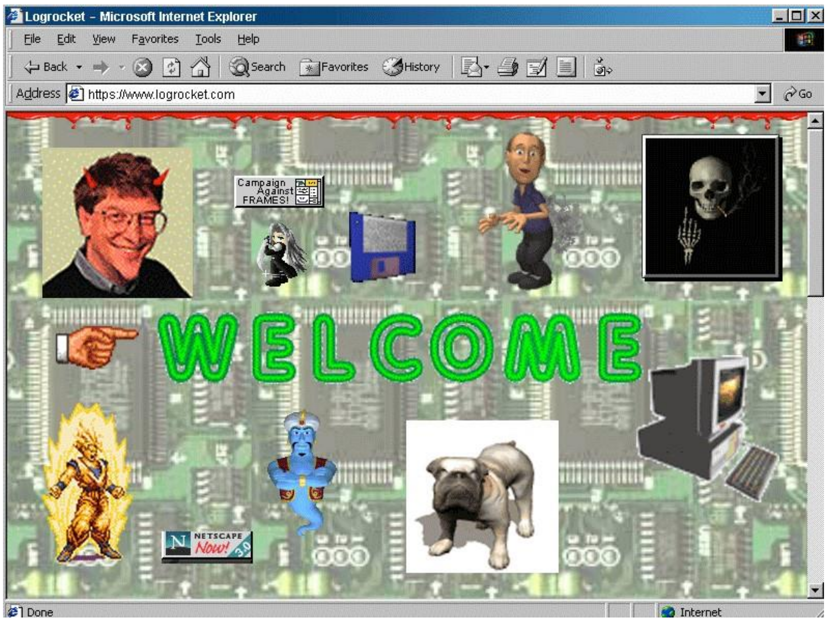

Logrocket – Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address https://www.logrocket.com
Go
Campaign Against FRAMES!
WELCOME
NETSCAPE Now!
Done
Internet

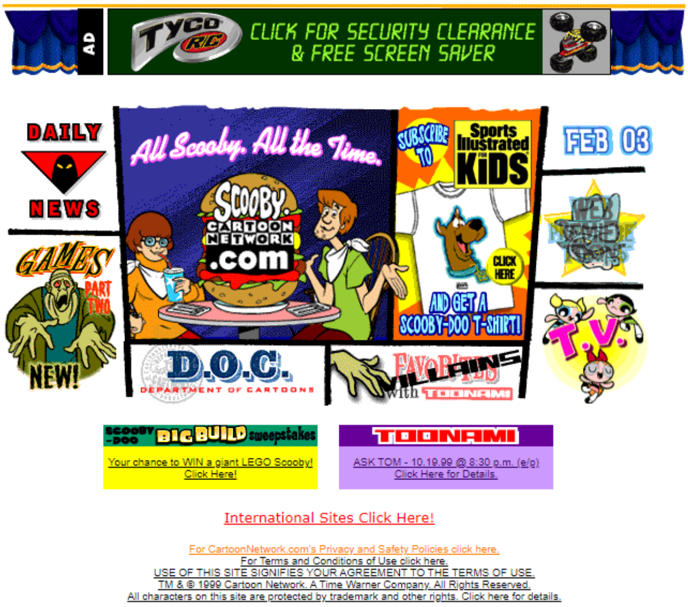
Figure 19, Typical playful 90s webpages

The participatory web

Web 2.0 is also widely known as the "**participatory web**" in stark contrast to user interfaces and websites of the past which were heavily centralized and controlled by their creators, web 2.0 would be a much more decentralized and democratized version of the web. This web 2.0 was widely acknowledged as the beginning of user generated content. Thanks to the ever expanding number of people with access to the internet, many users with little to no experience in programming would interact and cooperate with each other, thus creating new and different user-generated content for each other to enjoy.

Unlike the passive consumption and usage of websites in the past, Users would now actively participate in the creation and forming of these websites through cooperation and collaboration. Moving from the sterile and monitored websites of the past users could form blogs, wikis or even social networking sites. Social networking especially, would prove extremely influential and useful not just for the looks and functions of websites back then, but also decades later, even in the present day.

The reason that web 2.0 earned this particular moniker, was the fact that is slowly but surely moved away from the classic idea that many people had in regards to the internet. The internet would evolve from a heavily professional and corporatized background, toward a much more forward looking and decentralized form.


AD
TYCO R/C
CLICK FOR SECURITY CLEARANCE & FREE SCREEN SAVER
DAILY NEWS
GAMES PART TWO NEW!
All Scooby. All the Time.
SCOOBY CARTOON NETWORK .com
SUBSCRIBE TO Sports Illustrated KIDS
CLICK HERE
AND GET A SCOOBY-DOO T-SHIRT!
FEB 03
T.V.
D.O.C. DEPARTMENT OF CARTOONS
FAVORITE VILLAINS with TOONAMI
SCOOBY-DOO BIG BUILD sweepstakes
TOONAMI

Your chance to WIN a giant LEGO Scooby!
Click Here!

ASK TOM - 10.19.99 @ 8:30 p.m. (e/p)
Click Here for Details.

International Sites Click Here!

For CartoonNetwork.com's Privacy and Safety Policies click here.
For Terms and Conditions of Use click here.
USE OF THIS SITE SIGNIFIES YOUR AGREEMENT TO THE TERMS OF USE.
TM & © 1999 Cartoon Network. A Time Warner Company. All Rights Reserved.
All characters on this site are protected by trademark and other rights. Click here for details.

Figure 20, Cartoon Network's website in 1999

In regards to the creation of the webpages as well as services would steadily begin to be hosted on the machines of several different users instead of the servers of large companies. Technological improvements both on the side of software as well as hardware, would allow many users the ability to author and tailor their webpages to their liking, with an often bold and striking design to go with them. There were no more design constraints, and most people were able to design freely, inserting almost anything they desired into their webpages.

The new wave of user generated content and interface design.

This new style of interface would most often appear prominently in user generated webpages. Since computers were getting faster and cheaper, it was only a matter of time until they became available to most people. These people would often create their own websites. The cheap barrier to entry and the ability to generate a very striking webpage would make these websites a very attractive solution, especially to people who did not have enough capital for large and expensive advertising campaigns, a lot of the time these websites would belong to programming students. Very often they would be used as a way to promote their products and even amateur games they had developed.

Very similarly to the early adopters of social media as a marketing and promotional tool of today, the early adopters and users of these interfaces would go on to advance and promote their businesses and websites. Sometimes these pages would be created just for fun, with the developers wanting to create something interesting and intricate just for the fun of it.

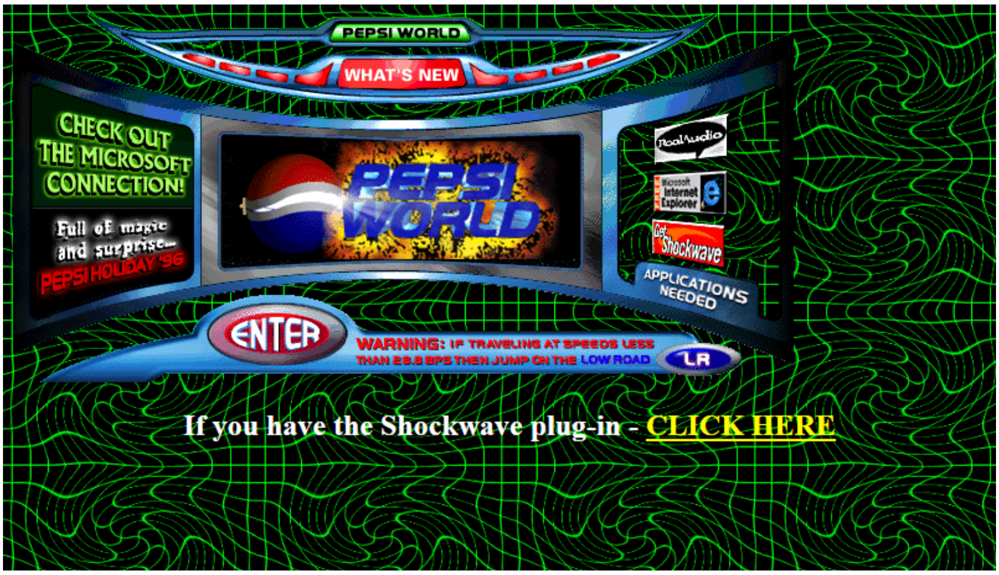

PEPSI WORLD
WHAT'S NEW
CHECK OUT THE MICROSOFT CONNECTION!
Full of magic and surprise...
PEPSI HOLIDAY '96
PEPSI WORLD
RealAudio
Microsoft Internet Explorer
Get Shockwave
APPLICATIONS NEEDED
ENTER
WARNING: IF TRAVELING AT SPEEDS LESS THAN 28.8 BPS THEN JUMP ON THE LOW ROAD
LR
If you have the Shockwave plug-in - CLICK HERE

Figure 21, Pepsi's 1996 website. Note the annotation for the Adobe Shockwave middleware

While these days, we often look back at this so called “wild west” of interface design more as a relic than anything else, it would affect interface design in the years to come. The design of an interface would become a legitimate profession, and developers would now be familiar with the need to draw one’s attention to parts of a user interface. As website design would slowly and surely mature, the wilder ideas and aberrations of this style would slowly be phased out, and a more deliberate and directed style would be adopted in its stead.

Chapter 5 – "The Y2K aesthetic"

Aesthetics start playing an active role in interfaces.

Y2K is often used as a shorthand for the term "year 2000". When referring to user interfaces as having the "Y2K aesthetic", we often describe them as having **an over-the-top futuristic style. With organic curving lines and smooth gradients**, it was a design language that embraced the new millennium with an optimistic spirit.

This design aesthetic would become popular with the turn of the millennium, and would be popularized through home computing as well as the budding videogame console industry. This look and style would represent machines not as boxy and old fashioned pieces of technology, but rather a part of everyday life, almost organic. No longer would they be represented as appliances, but a piece of every user's daily routine, advanced in technology and sleek in presentation.

Higher resolution screens and more powerful graphics processors would allow this shift in presentation to happen. As part of the idea that computers were organic rather than machines, animations would play a big role.

Figure 22, The futuristic and shiny look of the early 2000s

User interfaces move away from the personal computer.

The mid- and late 90s had become the decade in which electronics had finally gotten cheap and cost effective enough to become available to the average consumer. Scaling down the capabilities of a computer to a closed system with a lower price that was available to buy off the shelf, was a very attractive solution to both vendors and consumers. This closed system was the videogame console. A personalized computer designed for gaming. Consoles had gotten their start in the 80s, but had become profitable enough by then, that they had entered the mainstream.

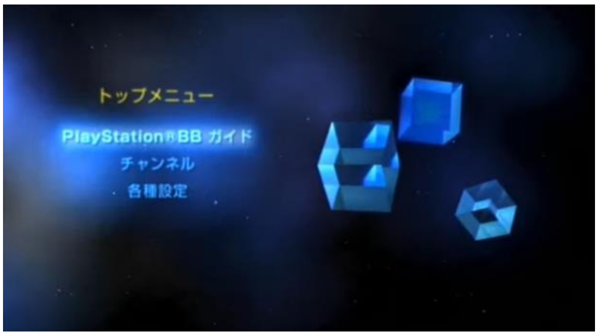


Figure 23, Sony PlayStation 2 user interface

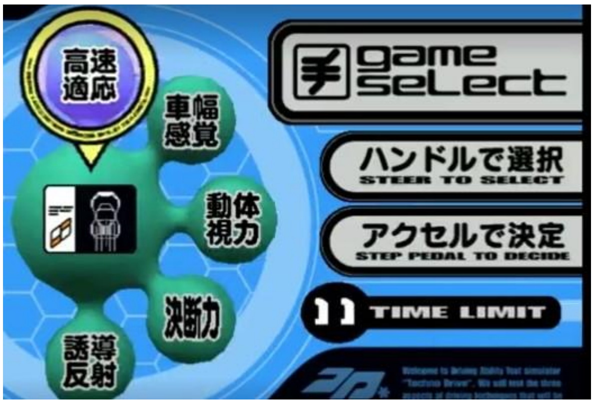


Figure 24, The UI to Namco's Techno Drive, designed by Japanese designer Minoru Sashida

Since these consoles were oriented mainly toward teens and young adults, the design behind their user interfaces would be much more experimental. Most of these consoles would also be imported from overseas, specifically Japan (Sony's PlayStation). The playful and futuristic style of the Asian country would influence both these consoles, as well as personal computers.

Looking toward an idealized future.

The new millennium had brought with it a renewed sense of optimism to the world at large. Arguably the greatest invention of the past years, the computer, would also follow in this path. Instead of a style that would swing wildly from interface to interface, a more homogenous solution would be favored. Clean, sleek lines and curves with smooth shading would be the new look, leaning heavily into the "computers as a partner" or "computers as a living thing" idea.

Kai Krause and Kai's Power Tools.

Kai Krause was born in 1957 in Dortmund, Germany. He is a graphical user interface and software designer, and in 2005 was named as one of the most influential innovators of the past 15 years.

He is known as the creator of one of the most influential styles of user interfaces, and a lot of people trace the entire Y2K design aesthetic revolution back to him and his software. Some of his most well-known applications are "*Kai's Power Tools", "Kai's Power Goo",* and "*Bryce 3D".* They were mostly video and image editing software, or 3d generative tools as well as plugins for other existing Applications, such as Adobe's Photoshop.

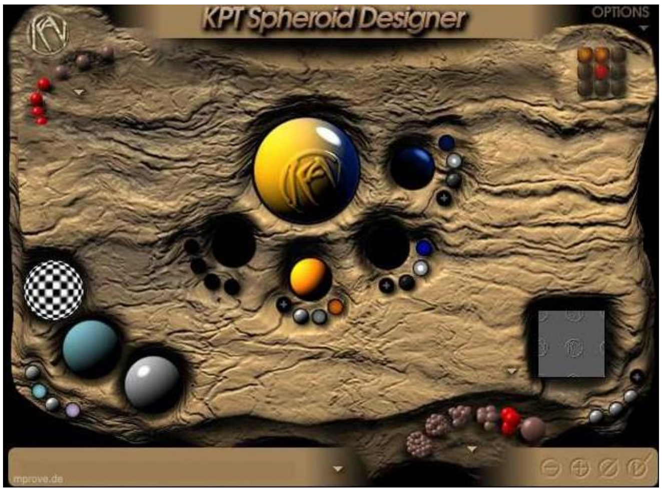


Figure 25, Kai's Power Tools user interface

Kai Krause's visionary designs would incorporate smooth shading and rounded corners into almost everything. Most of his user interfaces would be compared more to interactive entertainment applications today, than anything else. Textures and patterns would also be used in most of the interface . In regards to text, Krause would either use large colored speech bubbles, or forego text at all, electing to use detailed symbols or pictures instead of text commands. The user interface would take up most of the screen's real estate, but could be made smaller in order to focus on the picture or video that was being edited (Müller-Prove, 1999).

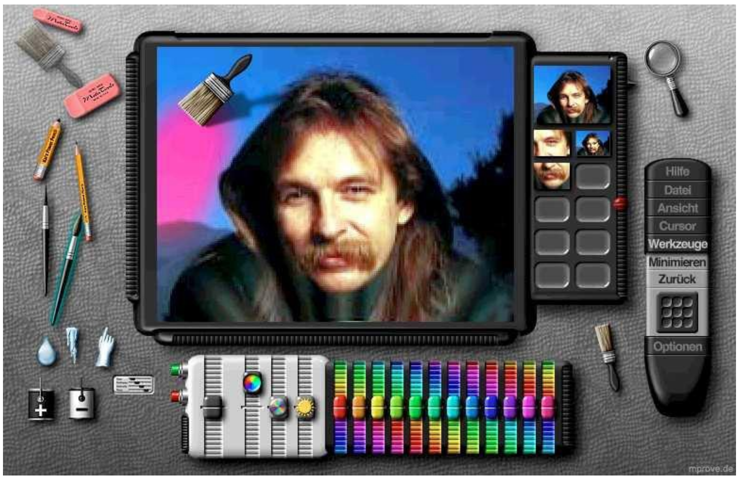


Figure 26, Kai's Power Tools. Note the smoothly shaded icons and elements. also note the skeuomorphic style tools

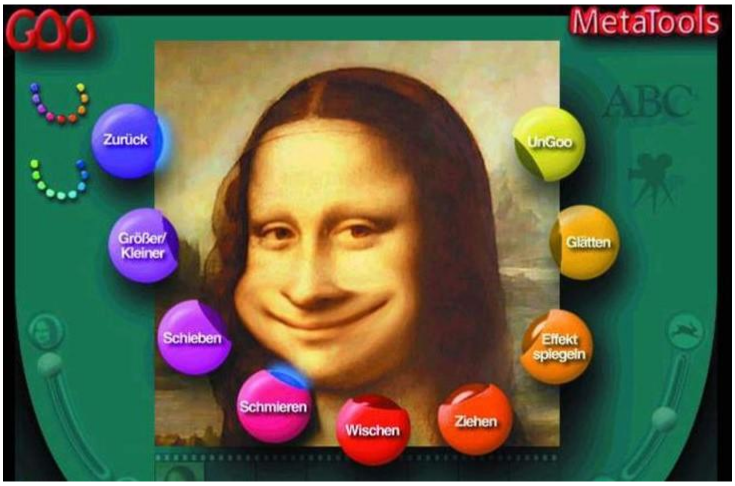


Figure 27, Kai's Power Goo

Another very important part of his user interfaces was sound. Leaning heavily into the idea of computers as more than machines, the user interfaces would incorporate clicks, scrolling sounds, beeps and other various organic and inorganic audio into its UI, giving it a very tactile feel. This was an effort to "remove the barrier of the computer's screen between the user and the program" and making the user interface as "non-computery" as possible.

Kai's Power Tools and their tactile nature are often also regarded as the prototypical version of skeuomorphism due to their similarity to real world objects, something that will be analyzed further down.

The Y2K aesthetic in the computing mainstream.

One of the first companies to adopt this new form of design in their mainstream computing would be Apple with their emblematic Mac OS X operating system (Souppouris, A). The new GUI would we named "Aqua" and consist of smooth clean lines and gradients and the color blue (or "aqua").

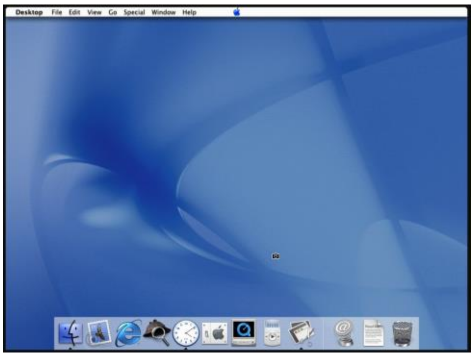


Figure 28, Mac OSX Aqua user interface

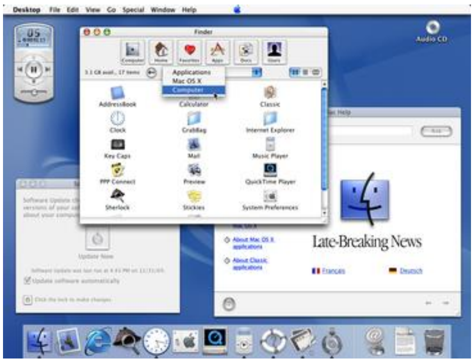
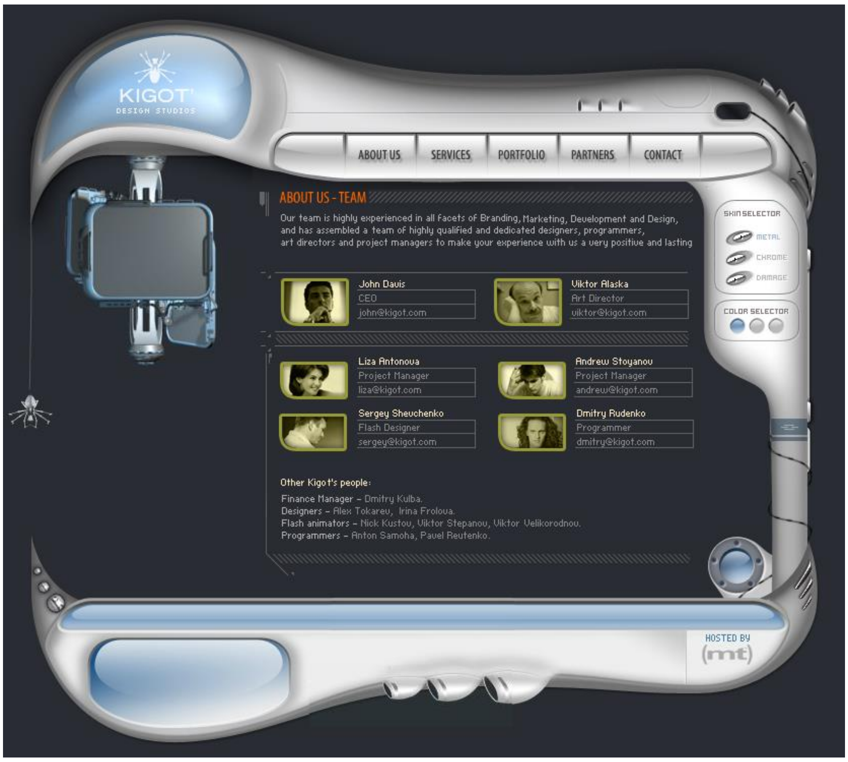


Figure 29, the clean white windows of Aqua

The windows would be large and detailed, and the text would be crisp and clean, while also being very easy to read. Electronic devices would benefit with the introduction of “sans-serif” fonts into their user interfaces.

Due to the increased screen real estate and detail that could be resolved from modern graphics cards, Icons became much more high resolution, featuring lots of details that would make them both easier to understand, as well as make them look more aesthetically pleasing. The aforementioned shadows would be dropped behind many layers and parts of the user interface, giving and almost three-dimensional effect to the user. This three dimensional effect would further the “lack of barriers” between man and machine idea. This concept would also be taken further with more advanced effects such as transparencies in future iterations.

Colors would often be a vibrant cool blue or green, offering a relaxing contrast to the plethora of icons. Various buttons and functions would often have a colored and monochrome version, denoting as to whether they were active or not.



ABOUT US - TEAM

Our team is highly experienced in all facets of Branding, Marketing, Development and Design, and has assembled a team of highly qualified and dedicated designers, programmers, art directors and project managers to make your experience with us a very positive and lasting

John Davis
CEO
john@kigot.com

Viktor Alaska
Art Director
viktor@kigot.com

Liza Antonova
Project Manager
liza@kigot.com

Andrew Stoyanov
Project Manager
andrew@kigot.com

Sergey Shevchenko
Flash Designer
sergey@kigot.com

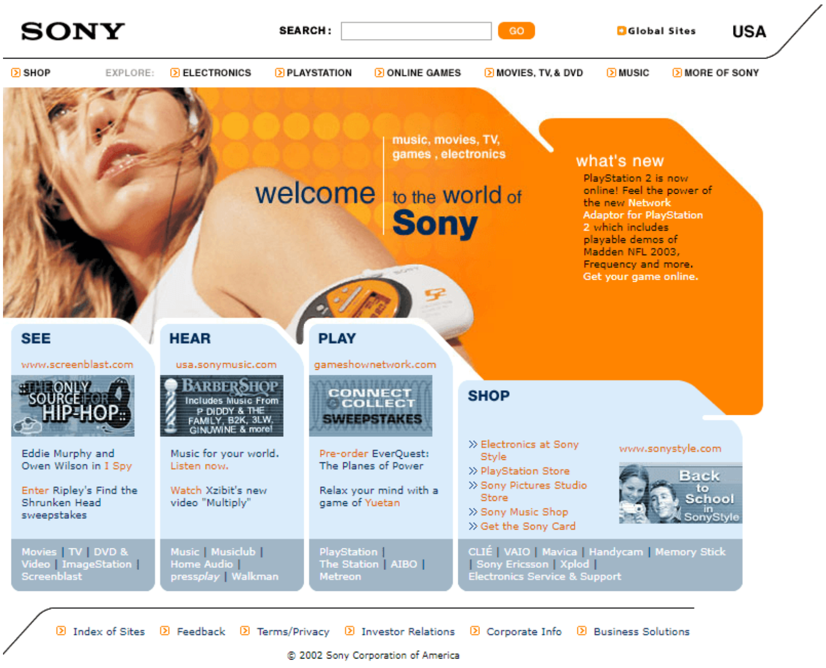
Dmitry Rudenko
Programmer
dmitry@kigot.com

Other Kigot's people:
Finance Manager – Dmitry Kulba.
Designers – Alex Tokarev, Irina Frolova.
Flash animators – Nick Kustov, Viktor Stepanov, Viktor Velikorodnov.
Programmers – Anton Samoha, Pavel Reutenko.

Figure 30, Kigot's design studios interface in 2004

Finally, animations would play an equally important role for the user interface. Powerful new hardware allowed for smooth scrolling and panning across large surfaces without sacrificing performance or resolution. Windows could be moved around easily without any lag. Animations would also be played when minimizing and maximizing them, allowing the user to more clearly visualize they were interacting with and what was non-interactive. Sounds would also be used both in operating systems as well as applications, which would denote the status of the currently used application. Iconic sounds which are used up to this day for many applications.

The Y2K aesthetic would further evolve in the later parts of the 00s, transforming into a more transparent, almost "ethereal" layer to user interfaces. It would be fondly remembered not just as a bold new experiment in the design of user interfaces, but also as a part of many a people's first memories of interacting with a computer. Today it stands as a reminder of a more optimistic and hopeful vision of the future, an artifact of a time when design was much freer and much less standardized.



PlayStation 2 is now online! Feel the power of the new Network Adaptor for PlayStation 2 which includes playable demos of Madden NFL 2003, Frequency and more. Get your game online.

Eddie Murphy and Owen Wilson in I Spy

Enter Ripley's Find the Shrunken Head sweepstakes

Movies | TV | DVD & Video | ImageStation | Screenblast

Music for your world. Listen now.

Watch Xzibit's new video "Multiply"

Music | Musiclub | Home Audio | press*play* | Walkman

Pre-order EverQuest: The Planes of Power

Relax your mind with a game of Yuetan

PlayStation | The Station | AIBO | Metreon

» Electronics at Sony Style
» PlayStation Store
» Sony Pictures Studio Store
» Sony Music Shop
» Get the Sony Card

CLIÉ | VAIO | Mavica | Handycam | Memory Stick | Sony Ericsson | Xplod | Electronics Service & Support

Index of Sites | Feedback | Terms/Privacy | Investor Relations | Corporate Info | Business Solutions

© 2002 Sony Corporation of America

Figure 31, Sony's website interface in 2002

Chapter 6 – Skeuomorphism.

What is skeuomorphism?

Skeuomorphism is the design concept of making items represented resemble their real-world counterparts. Skeuomorphism and skeuomorphs are not exclusive to user interfaces. Rather, they are used in a variety of different fields. From computers to architecture, to interior decorating.

The word itself is derived from the ancient Greek word "skeuos" (σκεύος) and "morphe" (μορφή).

When talking about skeuomorphism, abstract ideas and definitions such as "look" and "feel" often come into the discussion. While no one can define with absolute precision what the look and feel of a certain object or interface is, these ideas are the ones that are chiefly used in skeuomorphic design. As such, they are typically used in order to make something that is new and unfamiliar to a user or observer appear as something much more familiar.

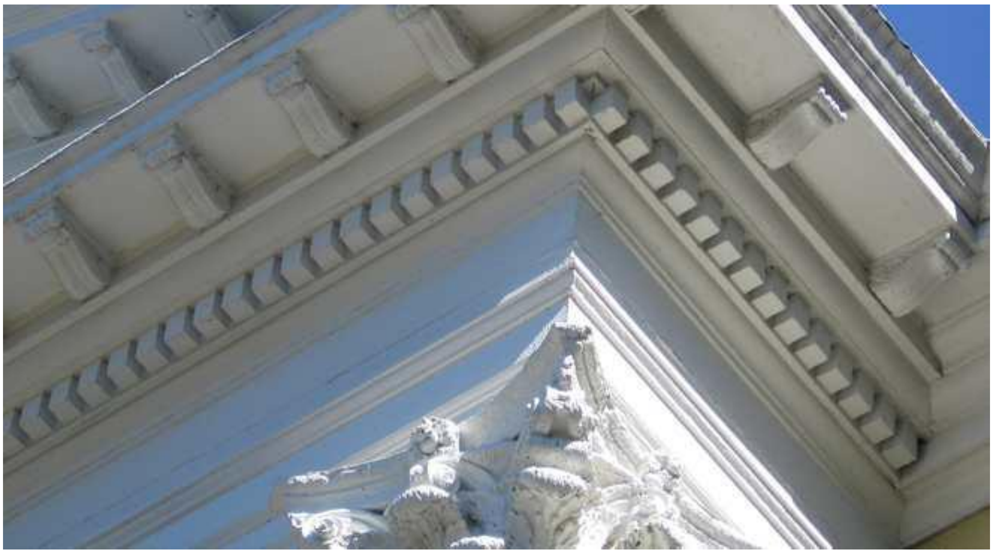


Figure 32, Skeuomorphic pillars in an ancient Greek temple. Note the decorative support beams.

A brief history of skeuomorphism.

As mentioned before, skeuomorphic design isn’t relegated just to user interfaces. Examples are often found in many different areas. Skeuomorphism is also not recent. On the contrary, it has been around for a very long time.

Some of the very first examples of skeuomorphic design date all the way back to the classical Greek era, and often, even well before then. An example would be the cups used by the ancient Minoans. Their designs often have decorative elements such as small rivets or screws, sometimes even painted on. These elements were used on more expensive cups and chalices which were made of metal. The skeuomorphic approach was used in order to make these objects seem more premium.

Later on, the ancient Greeks would use skeuomorphism in the design and construction of their temples. The temples would often have elements such as guttae or modillions purely as decoration. These elements were used in the past as parts of the design and structure of a temple. Before the usage of marble, they were required as a means of repelling water from the then thatched or wooden roofs or as a means of additional structural integrity for pillars. One of the most well-known examples of Greek skeuomorphism is the Corinthian column, whose capital was designed to represent a roof tile set upon a hedge.

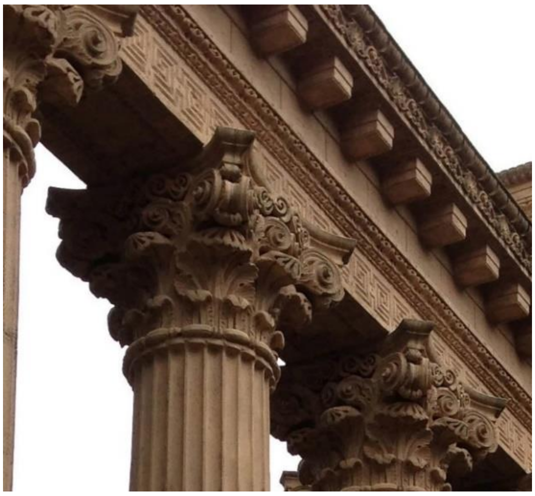


Figure 33, Corinthian Pillars. It was said that they were inspired by placing a roof tile on a thyme bush

During the Middle Ages, pipe clay or alabaster were used instead of more expensive and difficult to use materials such as stone or marble. They would then be painted to imitate the real material. The painting on of detail or texture would also be used for paintings and illustrations, sometimes even clothing. It would even be used for illustrations in scientific texts and many more manuscripts. Gold leaf or golden paint would be used in order to illustrate sunlight. Shadows would be painted on, for example.

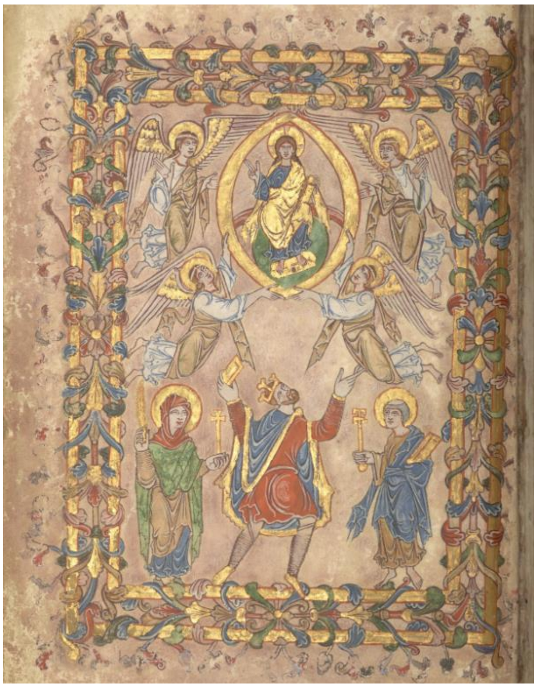


Figure 34, The illustration of a Medieval manuscript. The divine light is illustrated through golden paint on the parchment.

Later on, in the Renaissance times, skeuomorphism would appear a lot more prevalently in art and architecture. Using the knowledge of the era, large and complex structures could be erected without the use of materials or structural elements, which were often used as decoration. Interiors of buildings would use plaster and carvings in order to give the effect of a different material. Interior pillars with wooden textures are good examples.

As technology improved and materials became both better and cheaper, Skeuomorphism was used more often. In contemporary houses, skeuomorphism is still used in order to convey luxury and quality without sacrificing the practicality and lower price of modern materials. A vinyl tile floor made to look like reclaimed wooden planks or brick walls made of plastic are readily available and can be seen in many a modern interior .

In conclusion, skeuomorphism was an attempt to replicate the look and feel of a certain material type of design flourish using contemporary methods. This was mostly done for aesthetic reasons, but sometimes would be done in order to convey a certain message, or encourage familiarity between the observer and the subject.

“Feel” in user interfaces.

Very often when describing a various electronic medium, the term “feel” is used. When talking about how a user interface feels or looks, we don’t describe the actual tactile feeling of user interfaces (Smith, 2021). Rather, we describe certain aspects of the interface’s design and how various dynamic elements of the interface make us react to them. Are they logically placed? Are they responsive? Do they convey the message that they are meant to convey, and more importantly, are they laid out in such a way that makes sense to the user? Ease of use is a very important factor which contributes greatly to the look and feel of a user interface. This look and feel is often so important, that entire lawsuits would be laid out, alleging copyright infringements from software vendor to software vendor (Sharp et al., 2007).

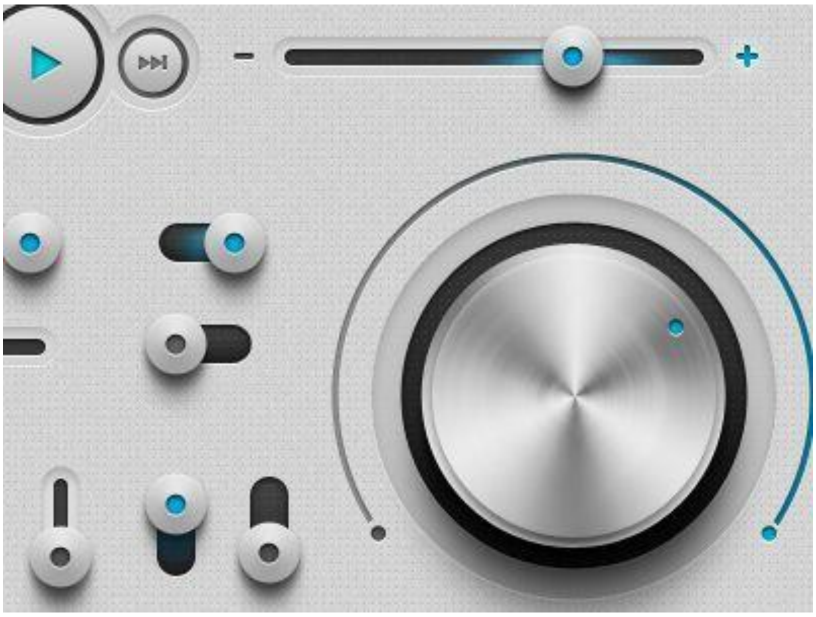


Figure 35, typical skeuomorphic design elements

Skeuomorphism in user interfaces.

The main argument in favor of user interfaces was that the signifiers used (icons, buttons, etc.) were much more familiar to users who had already existing knowledge of the actual, physical counterpart of the application. This would allow them to come to terms and learn the program faster. Skeuomorphism would **"give comfort and make learning easier"** to newer users, and give them a required background of knowledge which could be used even when future iterations of the application no longer resemble the physical counterpart. Another argument in support of skeuomorphism would be its ease of use. Large and clear components tied to each and every function of the application are inherently easier to navigate and use compared to similar looking flat and minimalistic icons and interfaces.

When skeuomorphism began to be used in user interfaces, the reason behind its application remained mostly the same as with the rest of the branches which utilized said design language. Skeuomorphism in user interfaces aimed to make the user feel more at home with the program that they were using, while also displaying various design flourishes that were emblematic of that time period (Janusheske, 2012).

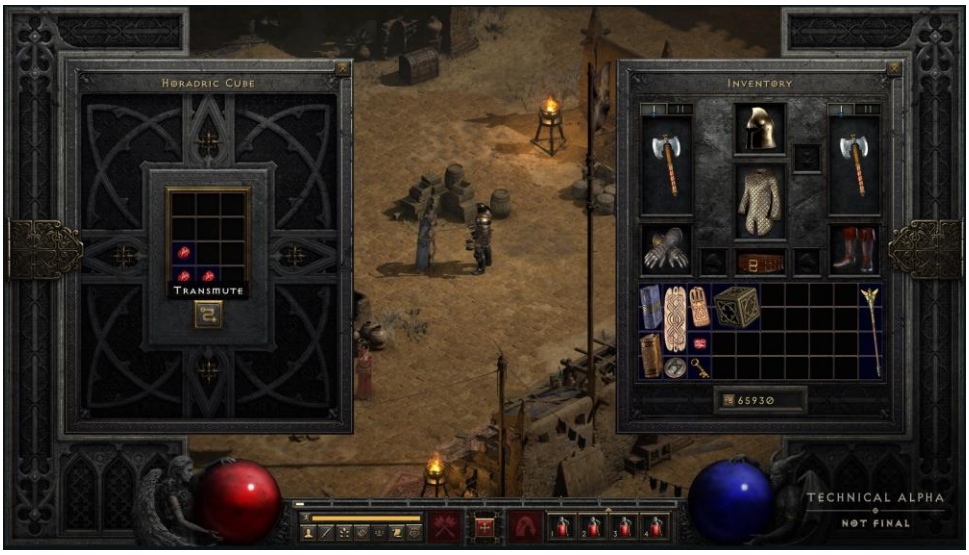
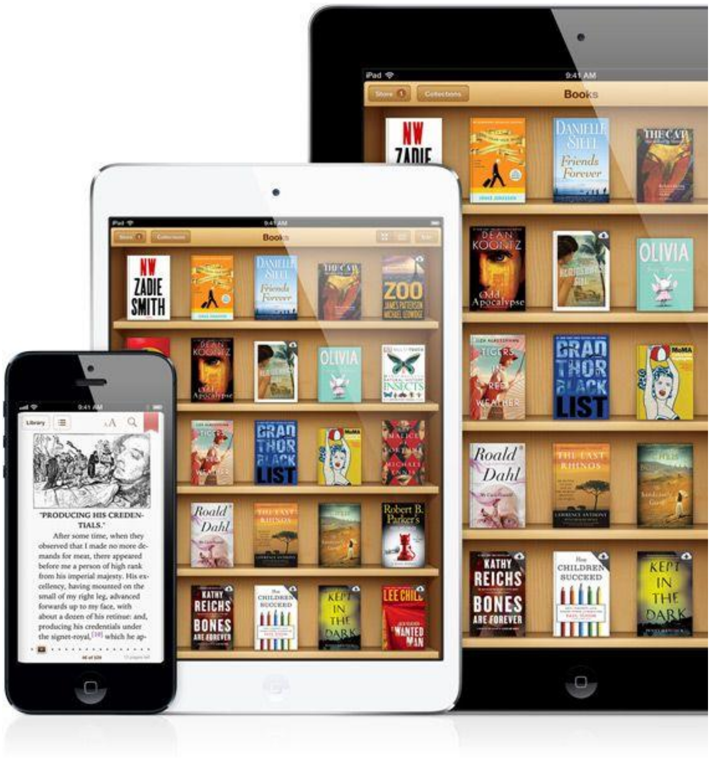


Figure 36, Diablo 2 Remastered's UI. Note the inventory represented as metal chests and the hit points and magic below represented by the orbs full of liquid.

As time would go on, skeuomorphism would develop its own user interface design principles that would be in turn replicated in many other programs. These design elements would endure for many years, and are sometimes found even today, mostly in eclectic and obscure or enthusiast applications.

What is a skeuomorphic user interface like? Much like skeuomorphism in fields other than computing, skeuomorphic user interfaces attempt to replicate the look and feel of an analog display and analog controls on a graphical user interface. In order for this style to work, the user interface needs to resemble its real-world counterpart as much as possible. A good example of a skeuomorphic interface would be a calendar application, whose windows take on the appearance of pages being flipped, or a text photography app making the sound of a shutter when a picture is taken. Generally for an interface to be considered skeuomorphic, it needs to have most of the following elements:

- Clearly visible and realistically shaded buttons and controls, with an obvious distinction between their active and inactive states,
- Sliders and switches,
- Scrolling wheels or dials
- Tactile sounds (clicks, button presses, etc.), to clearly signal when an action is performed



"PRODUCING HIS CREDENTIALS."

After some time, when they observed that I made no more demands for meat, there appeared before me a person of high rank from his imperial majesty. His excellency, having mounted on the small of my right leg, advanced forwards up to my face, with about a dozen of his retinue: and, producing his credentials under the signet-royal,[10] which he ap-

Figure 37, Apple's iOS 6 Books application. Note the books being placed on bookshelves

An example in WinAMP

WinAMP is a media application which was designed in the late 90s and has endured to this day. It was first released in 1997 and later acquired by the company *Radionomy* in 2014. Its latest stable release was version 5.8, which was released in September of last year.

WinAMP is an application which is aimed mostly at music enthusiasts and audiophiles. As such, most of its users have a rather extensive knowledge and background of music and software that is often used with it. It provides a vast breadth of options and selections which in turn allow the user to control almost all of the parameters of the audio file they are listening to.

In order to mimic the appearance of an in depth control panel, the user interface was made to look a lot like one. Almost every option was clearly labeled. The interactive parts of the user interface would look like buttons and switches, and were designed and displayed in a very clear manner.

Due to the increased control that more in depth users desired, WinAMP would offer all of these functions, and more, to the user. As such it was hailed as an extraordinary in-depth tool for playing and editing audio. It seemed as though the user could easily control almost everything in the application, making all of its functions readily available. At the same time, more experienced users would not be left wanting for more, since they could smoothly transition from analogue to electronic machine, while also taking advantage of their already existing knowledge base. This would let them operate and understand the new applications much more efficiently.

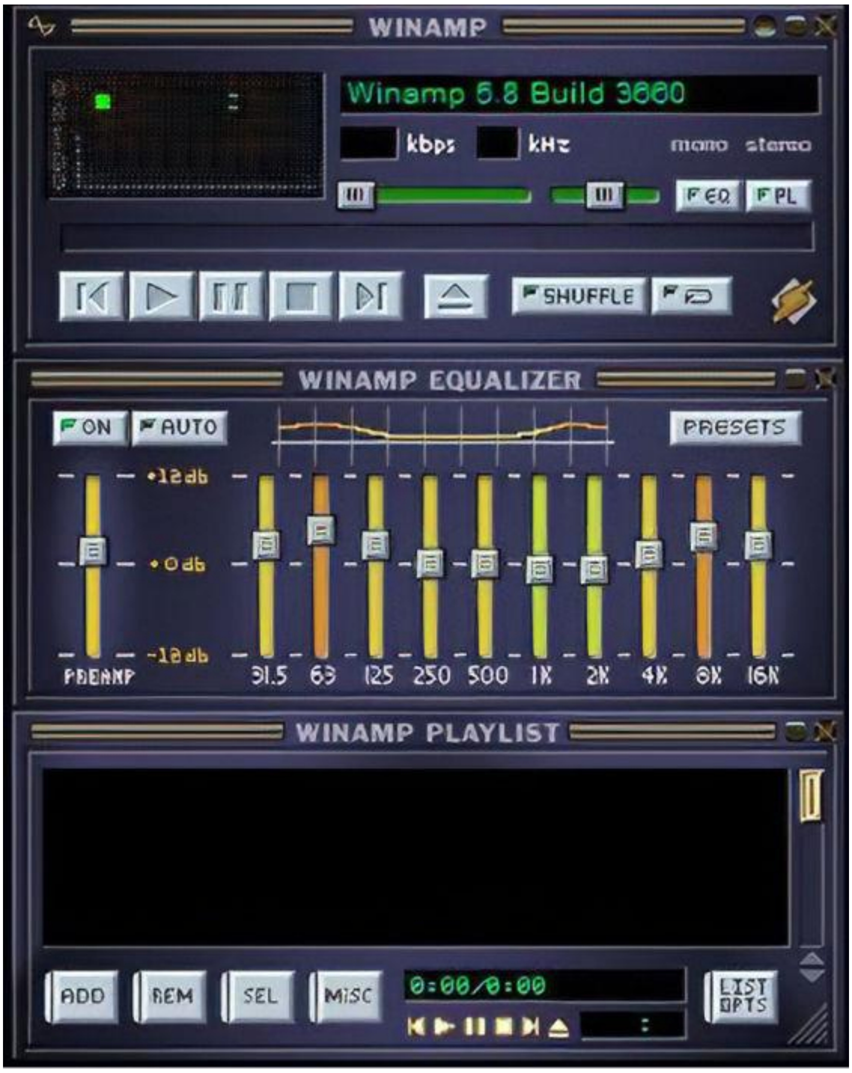


Figure 38, WinAMP's user interface. Note the skeuomorphic elements

Why did skeuomorphic design fall out of favor?

The main drawback of skeuomorphism is its constant and consistently high level of visual clutter. While at first it may be usable and even comforting to the average user, a completely new and inexperienced one would quickly feel out of their depth, akin to having to operate a complicated piece of machinery without any prior knowledge.

Computers and user interfaces are designed in order to make things easier and smoother for the user to operate, not more difficult. As such, when they are confronted with something so dense in detail and text, they will quickly resign themselves, and not operate the user interface as quickly and efficiently as they otherwise could.

The dense presentation of the skeuomorphic user interface also causes the user to experience increased cognitive load when using the interface. As useful as a plethora of knobs and switches and sliders and gauges may first appear, they turn into visual noise, which the user has to actively parse through in order to perform their tasks (Cooper, 1990). It’s well known that a cluttered work environment will result in inefficient work, while also increasing the likelihood of errors occurring. When a large amount of visual and aural static is pushed upon the operator, it also interferes with their ability of pattern recognition and visual association, causing them to feel “lost” when using any program. One need only ask a few people what it feels like to switch from a program that they are well versed in to a completely new one, and all the negatives that it brings with it.

Skeuomorphism today.

Skeuomorphism in this day and age is remembered fondly by many users. Even though it has outgrown its usefulness, it still appears in many forms, mostly as a design flourish or rather a callback to a more physical and tactile time. Skeuomorphic design in user interfaces has stayed in certain enthusiast programs and applications, much to the delight of many of their users. Certain technological devices which are inherently tactile, such as various smart devices such as watches still embrace it. Their form factors and applications make them perfect candidates. Video games and their user interfaces have benefitted the most from this tactile style, often using it as a way to express style or to make gameplay more immersive.

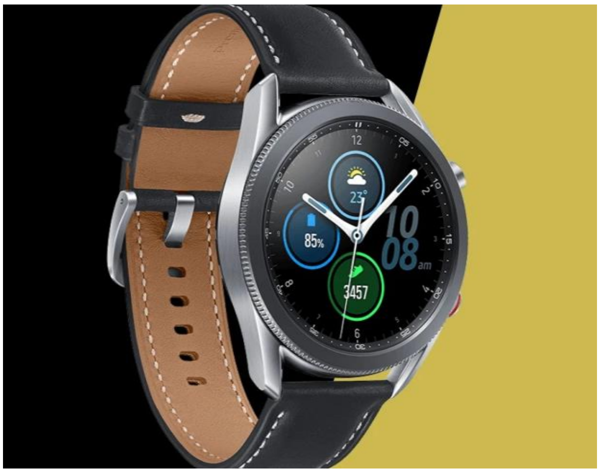


Figure 39, A Galaxy Watch smart watch. Skeuomorphic elements (hands) with applications beneath them

Finally, the term “Neumorphism” has appeared in the world of UI in the last few years. It aims to combine all the positives of classic skeuomorphism with a less cluttered and more modern and clean design style. Instead of having elements levitate on top of the running application’s desktop, Neumorphism attempts to make them “sink into” the interface itself. Due to most of the design principles behind it being similar to already known ones, many users have embraced it as a fun, albeit familiar design experiment.

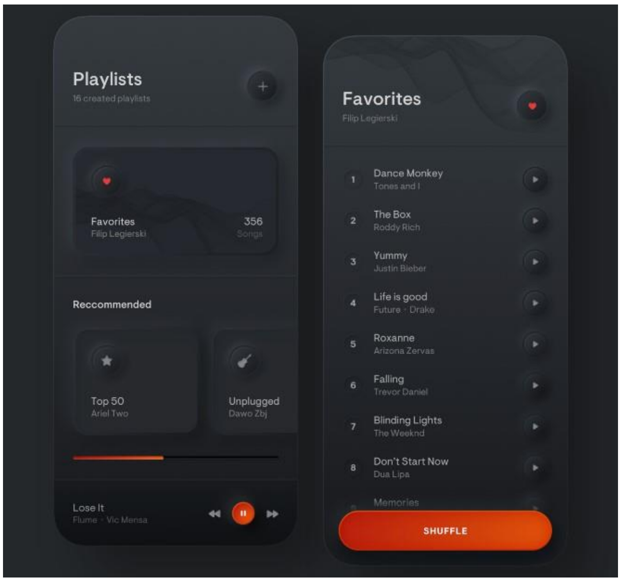


Figure 40, A modern Neumorphic UI. A combination of tactile looks with flat simplicity

Chapter 7 - Flat interfaces.

What is a flat style interface?

A flat style interface is a user interface characterized by a heavily simplified design language, which often appears "flat" or "two-dimensional". Their iconography is usually also flat, with extraneous detail being absent. Flat interfaces rely on a few colors, simple shapes and graphics in order to convey their message to the user. Anything that could be considered extraneous or unnecessary is usually removed, with only the most important elements left onscreen. Negative space is used in abundance, giving the interface a more relaxed and non-cluttered appearance. The flat interface has a very cohesive design to it, allowing the user experience to flow without many distractions (Spiliotopoulos, 2018).

Flat interfaces purposefully aim to reduce clutter and give the end user a more simplified and easy to read screen (Grinberg, 2013). Due to modern applications being so powerful and multi-layered, a first-time user can often find themselves easily lost in an application that they know nothing about and make numerous mistakes by extension. As such, a solution would need to be reached.

The solution in question would focus on the look and feel of the interface. In sharp contrast to the skeuomorphic look of the previous generation of user interfaces, this new generation would instead focus on providing a very simple and clean look that focused on text, simplified icons and a minimalist aesthetic for colors (Moran, 2017). As time went on, stricter and much more streamlined design rules would be imposed that would in turn dictate the evolution of these interfaces.

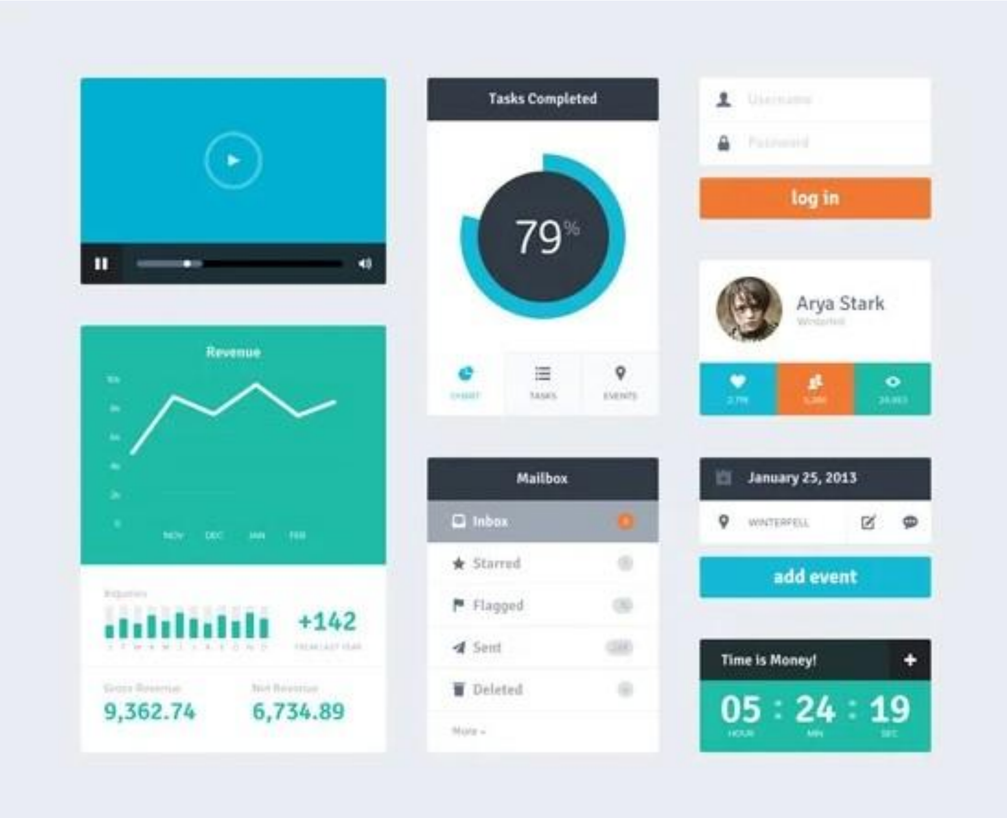


Figure 41, A modern flat style interface

Responsive design for user interfaces.

A very prevalent problem of user interfaces in the later parts of the 00s and early parts of the 2010s was a lack of responsiveness. Very often a user would try to open an application on their smartphone only to be greeted with a miniscule and hard – to – read desktop version on their screen. Similarly, desktop users would be greeted with enormous and unsightly black bars on parts of their screen, which were simply not used by the running app in question. The issue would be even worse when using a device with an unorthodox or uncommon aspect ratio, leading to multiple problems with usability.

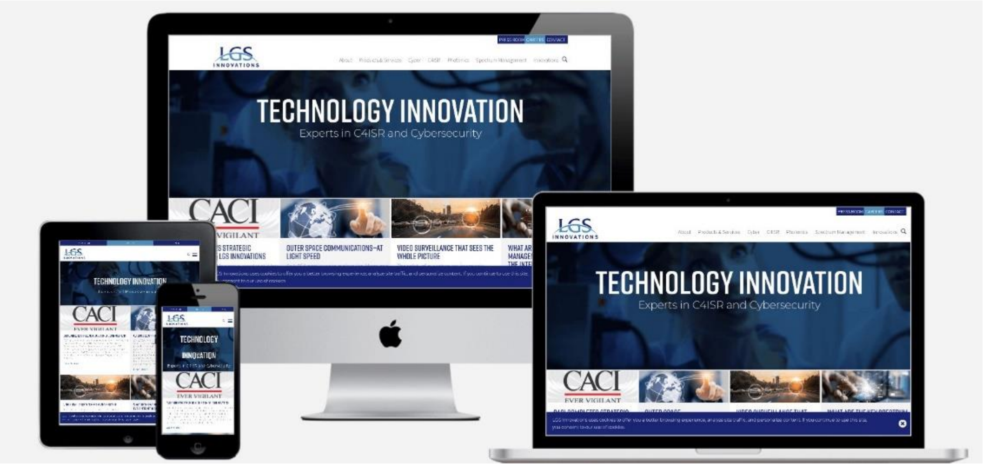


Figure 42, A webpage with a responsive web design layout. Note how the content smoothly scales from device to device.

The term “responsive design” is used to denominate an application which can easily and simply scale across multiple devices with multiple aspect ratios. **Its name suggests that design should respond to the user’s behavior and environment based on screen size, platform and orientation.**

Modern users almost always have more than one electronic device and switch between these devices very often. These devices run the same applications between them, and as such, must be able to scale and change the layout of the application in real time without problem from device to device. This shift from device to device must not compromise the application or website’s design, and should keep the hierarchy of information in the same order.

Modern, flat interfaces allow for moving of elements in a much easier way compared to older, more three – dimensional ones. Due to flat design being purposefully very minimal (as we’ll see below), elements have a similar visual weight, and as such can be moved around without creating confusion or compromising the aesthetic result of a page or application. This reorganizing of the elements is done through a complex system of grids via a CSS style sheet.

Responsive design has now been adopted by almost every application and website across most machines, and has allowed both older and newer, modern interfaces to be fully compatible for current and future devices. A very simple but extremely important element of modern design, which also allows for future proofing and compatibility between many different devices.

Flat interfaces.

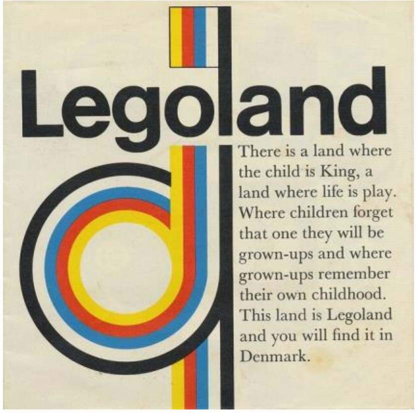
Flat design is a recently developed design trend through which most icons and elements are represented in a very simplistic and stylized manner. Flat design has a very specific set of aims, which are as follows:

- **clear visual hierarchy**
- **readability and legibility**
- **easy adjustability in terms of adaptive and responsive design**
- **effective legibility in various screen formats, screen sizes as well as more modern unorthodox displays.**

The origins of flat design.

Flat design was not solely invented for the modern smart device. Its roots hail all the way back to the early 20th century, more specifically to the “International” or “Swiss” style of typography (Terror, 2009). Much like today, the aim of that specific style of typography back in those days was to favor simplicity, legibility and objectivity.

Flat design of the 1950s and 1960s favored simple forms, bold and strict fonts and a geometrically oriented visual hierarchy of various forms and illustrations. These elements of graphic design would be imported into these user interfaces, making them clear, clean, and usable.



There is a land where the child is King, a land where life is play. Where children forget that one they will be grown-ups and where grown-ups remember their own childhood. This land is Legoland and you will find it in Denmark.

Figure 43, A flat style poster from 1960. Note how color, typography and legibility play a role in the design

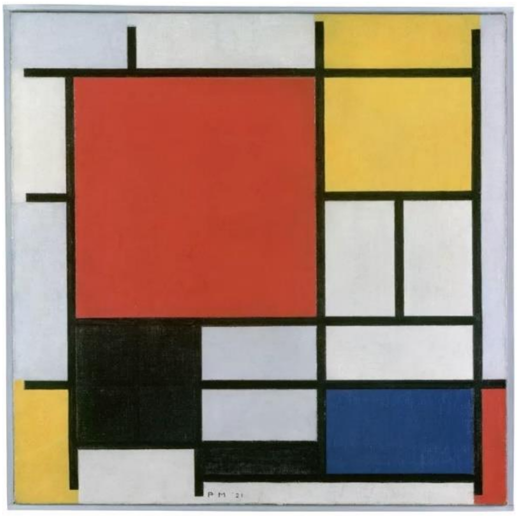


Figure 44, Piet Mondrian's Composition with Large Red Plane, Yellow, Black, Grey and Blue, 1921. Made in the style of De Stijl, was one of the earliest examples of flat forms and bold colors

Modern flat design in user interfaces eschews the previously used and detail oriented skeuomorphic style in order to introduce a much more usable user interface. Designers of the flat interface deliberately went for a style that lacks shading, depth as well as detailed and realistically drawn icons. Instead of replicating objects in the real world, flat design uses exclusively low detail icons, often monochrome (Turner, 2014). These icons can then easily be scaled according to the size and resolution of the display, thus offering maximum user experience on many different devices. It also heavily favors the use of bold yet muted colors. These colors are often used as a way to differentiate icons from backgrounds and allow for the user’s eyes to organically focus on the interactive parts of the UI.

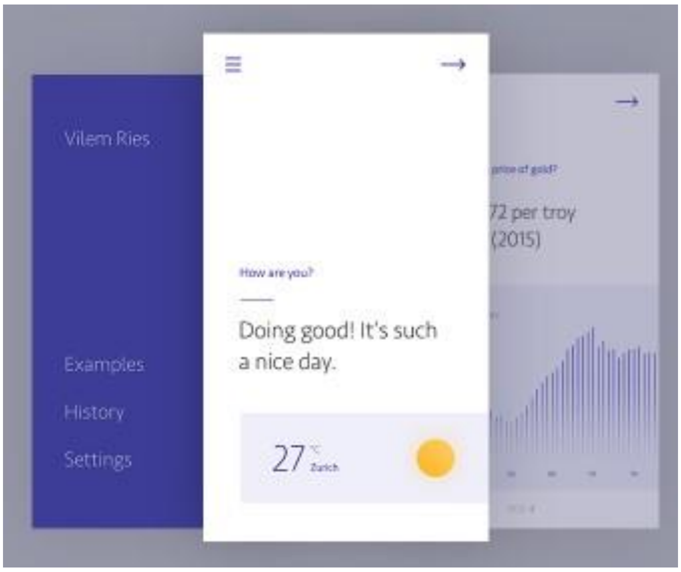
Flat user interfaces make heavy use of negative space. In stark contrast to the user interfaces of the early 2000s which also tended to suffer from the almost “horror vacui” level of introducing more and more elements to the screen. With a modern and flat user interface, instead of filling the screen with more, anything that is not currently required for the user to interact with is culled or hidden, leaving a large area of negative space for the user’s gaze to rest upon, allowing them to avoid visual fatigue.

Flat user interfaces also often include full size photographs. Modern high resolution displays, combined with larger screen sizes allow for large, colorful images to be fully displayed. These pictures have become increasingly common and used, especially with the advent of social media and their continuous posting. As such, a minimal and flat user interface aims to be as subtle and discreet as possible. Invisible when not needed and completely usable when needed.

Elements of flat design

Minimalism

When it comes to flat design, minimalism is maybe it’s most recognizable and distinguishable feature. Everything is purposefully simplified, especially the iconography. Icons are usually one solid color and lack almost all detail. They are usually designed as outlines or filled with a dark or light color. This makes them easily visible and identifiable in different screens, lighting conditions, as well as from a distance (Greaney, 2006).



Doing good! It's such a nice day.

Figure 45, a minimalist weather application. It only shows the bare minimum that's needed to the user

Call-to-action button

The “call-to action” button or CTA for short, is an interactive element of the user interface that is used to perform certain tasks for the corresponding application. The CTA button is often bright and distinguishable when compared to the rest of the application. It’s often large and has a strong color, thus allowing the user to find it easily on each page of the application.

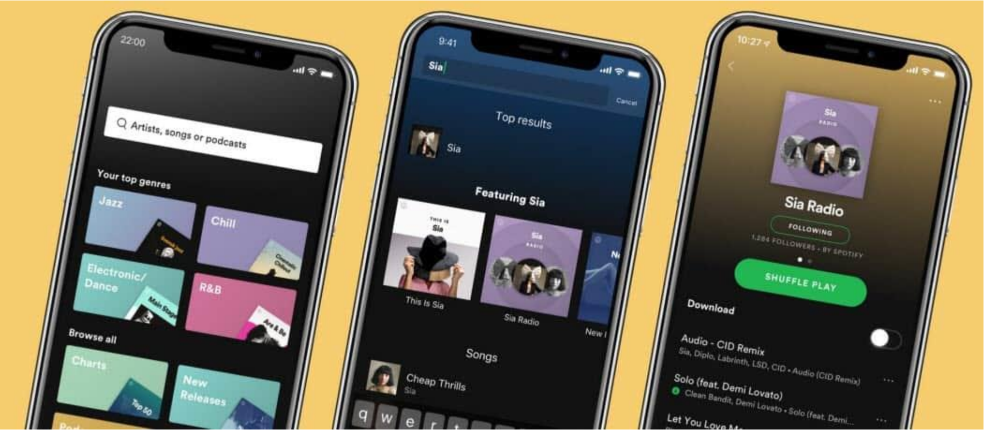
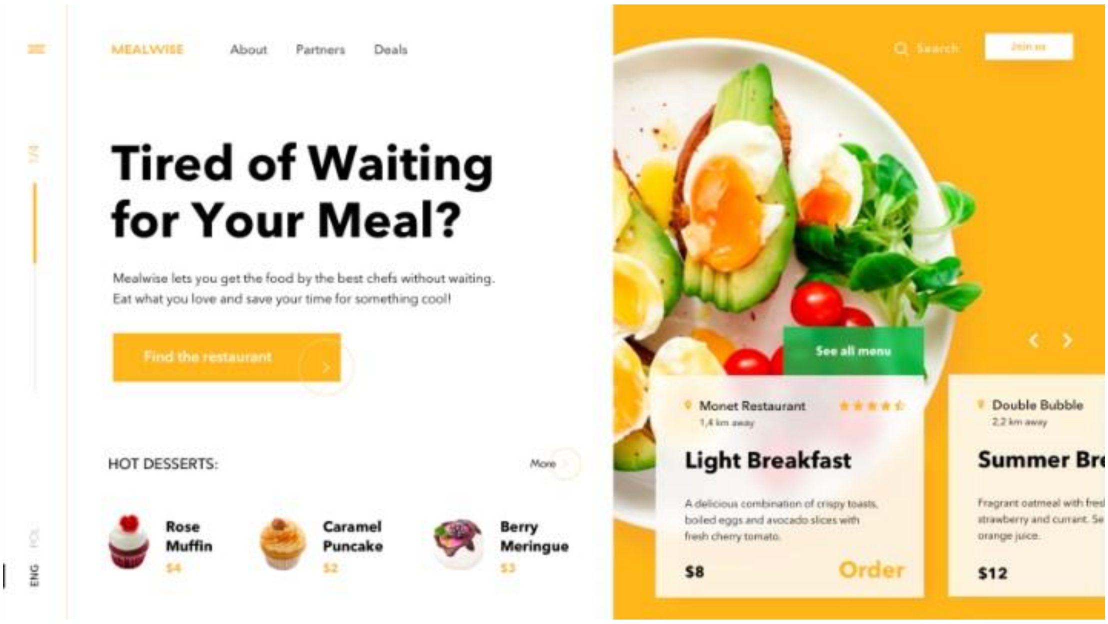


Figure 46, Spotify's CTA button is bold and green, instantly directing the user's focus toward it.

Clear visual hierarchy

In order to combat one of the most pervasive and difficult issues of interface design -that is, the visual hierarchy of elements that can and cannot be interacted with- modern user interfaces establish a very clear and very concrete visual hierarchy using various elements in order to aid the user.



Mealwise lets you get the food by the best chefs without waiting. Eat what you love and save your time for something cool!

A delicious combination of crispy toasts, boiled eggs and avocado slices with fresh cherry tomato.

Fragrant oatmeal with fres strawberry and currant. Se orange juice.

Figure 47, An example of visual hierarchy. Every image and element has its own weight and adds to the presentation without being noticeable

Typography

In regards to their typography, modern user interfaces have a deliberate and very clear simplicity to them. Most of the time, only a single font is used, with variations in thickness and size being the determining factor for hierarchy. The fonts that are more readily chosen, are ones that are sans- serif, since their simplicity and clarity shows up better on a monitor, as well as mobile devices. The spacing and kerning between the characters is standardized and clear, allowing for each option and selection to be viewed very clearly by the user.

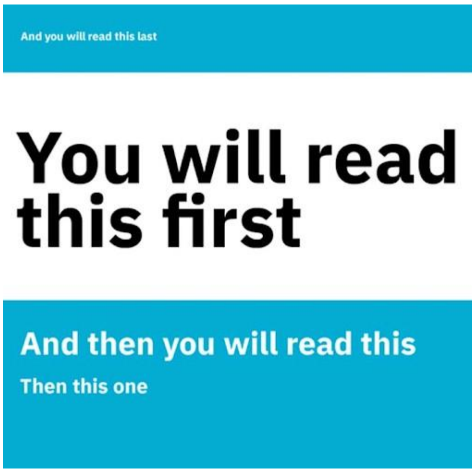


Figure 48, an example of visual hierarchy. Every image and element has its own weight and adds to the presentation without being noticeable

Colors

One light color, one dark color and one accent color is the rule of modern interface design.

When it comes to the color of the typography present in the user interface, the color is either a stark or off - white, or dark grey. Most user interfaces these days have both a dark mode and a light mode. The darker typeface is used when the application is in light mode, and the white typeface is used during the dark mode setting. Color is used sparingly, mostly to attract attention to what is interactive, and is usually only one. Bright enough to attract attention, but not too bright, causing the user's eyes to strain.

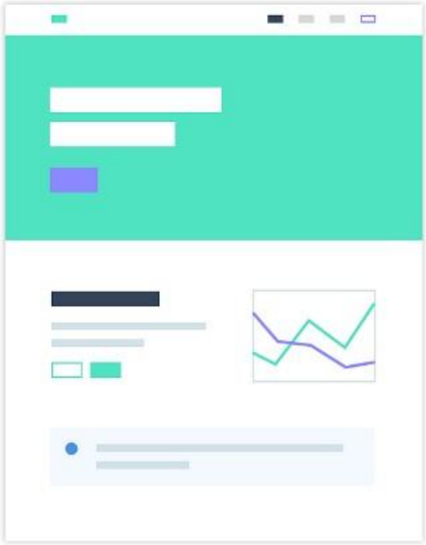


Figure 49, Few colors being used correctly in a modern user interface

The main criticism of modern flat interfaces.

Modern flat interfaces are not immune to criticism. They have been described as dull, lifeless, sterile and even obtuse. Their nature is very limiting, and meaningful changes are very difficult to implement without breaking one of their foundational rules. Flat interfaces also tend to be rather sparse in regards to delivering information, limiting the amount of text on screen on purpose in order to keep the cognitive load on the user low.

Finally, the biggest problem with flat interface designs is, surprisingly**, a lack of signifiers of interactivity**. Namely, the homogenous flat design doesn't always indicate clearly which elements are clickable and which are not. This was not a problem with past user interfaces, since their shading and prominent iconography clearly indicated which was which.

Most of these problems would be remedied with Flat 2.0, also known as "Material Design".

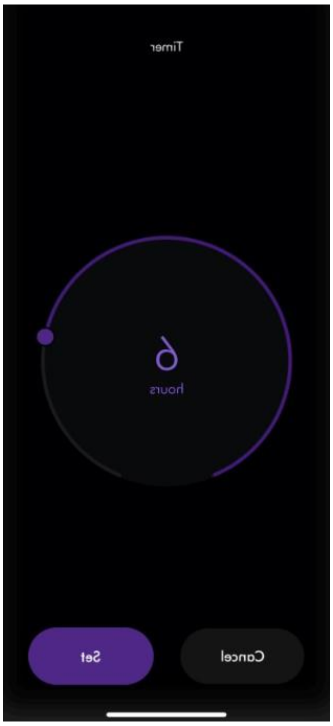


Figure 50, an example of an oversimplified user interface. The surfaces that can be interacted with are only known to users already familiar with this UI.

Chapter 8 - Material Design.

What is “Material Design”?

Material design, quantum paper or “material” is a design system introduced by the company Google. It was first introduced in the 2nd half of the 2010s as its own answer to the user interfaces at the time. In an effort to streamline and remove extraneous and overcomplicated details and design flourishes, Material design adhered to strict guidelines.

Material Design aimed to use the real world and its texture, its light and shadow play in order to create a clear and cohesive user interface. As per google, “material surfaces reimagine the mediums of paper and ink”.

In conjunction with the texture and shadow in the user interface, bold colors and graphics are also used in order to create a hierarchy in the user interface.

Fonts are sans serif and bold. This allows them to scale well across a variety of devices and aspect ratios. The interface always looks sharp and clear, allowing the user to easily parse its functions. Subtle shading is used in order to elevate parts of the user interface and make clear what is interactive and what’s not.

Finally, motion and animation play an equally important role. Not just as a way to make what is shown and interacted with more evident, but also as a way to add expressiveness to an otherwise flat interface.

A quick background.

Material design was first announced in 2013, though its background and the people behind its development are difficult to pinpoint. Material design hoped to create a beautiful and functional user interface across all the devices that used android. An arguably herculean task, since most devices of the time had vastly different designs and specifications (Ron, 2014). A unified look and user interface that leveraged both touch and physical buttons while at the same time looking good and remaining functional for the foreseeable future was not easy.

The introduction of material design coincided with the introduction of many new smartphones that would use “stock android”. A form of the android software that lacked many embellishments and features added by manufacturers. Devices would be purposefully built to run as smoothly and efficiently as possible without additional software load.

The platform would first be rolled out in 2014, and is still in use to this day. It would go through many iterations with each one focusing on different parts of the user experience. It is currently on its 3rd iteration, launched in 2021, dubbed “Material You”

The principles and innovations of Material Design.

According to Google, Material Design is guided by print-based design elements. It uses typography, grids, space, scale, color and imagery in order to create a visual hierarchy and improve the user experience (Google,2022) **While clean, flat and modern, a lot of the foundations of Material Design are very clearly skeuomorphic.** Through the usage of subtle shading and physics – based animation, a balance is struck between a clear minimalistic style and a playful and expressive motion which further reinforces the visual hierarchy of the on-screen elements.

Quantum Paper.

The main revolution of Material Design comes in the form of digital paper. It aims to utilize the inherent physical properties of paper in order to provide a clear and usable interface to navigate through.

In order to make the user interface feel more tactile, each surface is treated like a sheet of paper. Instead of having the interface be completely flat, a subtle shading is used in order to elevate elements which are interactive to the forefront. This gives immediate visual clarity as to what is and isn’t a part of the foreground as well as minimizing user error by giving visual cues as to which part of the screen is and isn’t currently active. Elements are also darkened and lightened dynamically in order to allow the user to easily identify as to whether their state is active or inactive.

Fig. 51, the user interface as a three-dimensional sheet of paper (quantum paper).

Color and contrast.

Color and contrast play a very important part in the Material Design guidelines. In order to ensure the clarity of the UI, strict rules are enforced in order to avoid low contrast situations.

Colors are usually limited to just a few, with various lighter and darker variants being used when necessary. Usually a strong primary color is placed upon the top part of the application's window. Each application uses a color which is quite different from each other, thus making each application feel different and distinct from each other. The bright colors are also easy to identify, especially in comparison to photos or text, making pattern recognition more possible. The user will come to identify applications based on their color. YouTube is always red in color whereas Spotify is always green.

The ever present "floating action button" either has the same color as the rest of the interface, or is its complementary color. Even though it may look different, it still needs to adhere to the aforementioned color palette.

The surfaces themselves are designed in order to provide maximum contrast and legibility. They are often in an off-white color, in order to avoid serious eye strain for the user when they use their device for an extended period of time.

Always a night mode or dark mode is also offered. While the header and accent color is kept consistent, the surfaces are darkened, with lighter text placed upon them. Eye fatigue is greatly reduced when looking at dimmer screens, and text often appears much clearer. A positive benefit of reduced brightness and darker interfaces is reduced battery consumption. Most displays today are self-illuminating, causing darker images to consume less power than brighter ones. Dark modes also serve to mitigate blue light exposure, which causes lack of sleep and tiredness for users. Since they have less of the blue light which is needed in order to create white, users can use their phone for longer without tiring their eyes too much.

Fig. 52, software buttons, titles, headers and footers with good, usable contrast between them.

Animation.

One of the most important parts of material design is animation and movement. Animation serves not only to make the interface more fun and expressive to use, but also to emphasize the way in which elements transition from one state to another.

The term “child element” refers to an element which comes from, or is expanded from another element. An example would be tapping or clicking on the picture of a contact in an app, and a window expanding with their info and details. This is a child element.

In order to keep interfaces as simple as possible, there is much less information displayed on the screen, hiding anything which is not immediately needed. Therefore, child elements are liberally used. When tapping upon any part of the interface, a new tab is expanded, which offers us metadata on the part of the interface that we have just interacted with.

In Material Design, this process is handled through animation. Tapping on an element will make it expand or shift through the page in a natural manner, becoming a focal point for the user. Animations have set rules of course, in order to not break the flow and clarity of the user interface. Two – dimensional elements will always overlap for example, and never phase through each other.

Another very important rule of Material's animation process is the animation speed. Animation speed is not linear, but rather it accelerates and decelerates in a natural manner. This process is called "easing". Easing mimics the way natural forces function. Not linearly, but with a natural and inconsistent manner. Since the user's inputs are not always perfectly linear as well, this animation timing makes it easier to follow and allows for a more tactile feel.

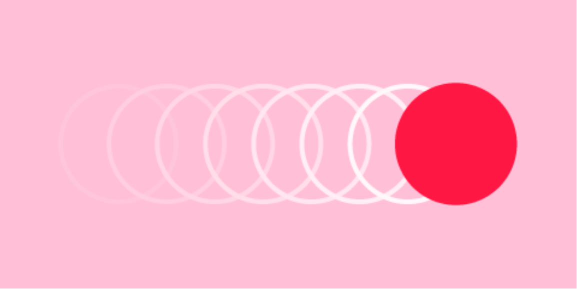


Fig. 53, a typical animation of material design. Unlike other contemporaries, it does not move in a steady manner, but rather accelerates and decelerates organically.

Gestures

While gestures in modern design interfaces are not a product of Material Design, this newer approach allows for users to easily understand which elements of the screen are dynamic and movable and which are not.

Universally accepted gestures such as pinching in and out for zoom and multiple finger scrolling were popularized by Apple in the mid-00s. Although they are now part of any navigation- based ecosystem, their application was not always met positively, since the user had to already know which parts of the screen were dynamic, and also what certain gestures would do before making them. It was a case of having to learn beforehand, not allowing the user to learn organically by doing.

Gestures in Material are heavily telegraphed through animation, allowing the user to understand what is going to happen before it actually happens. Most elements on screen have some sort of movement, and their interactivity is highlighted through subtle shading. Further evolving the metaphor of the user interface as paper, these elements and their tabs often conceal other buttons and controls beneath them, which move in and out of frame with a sliding motion.

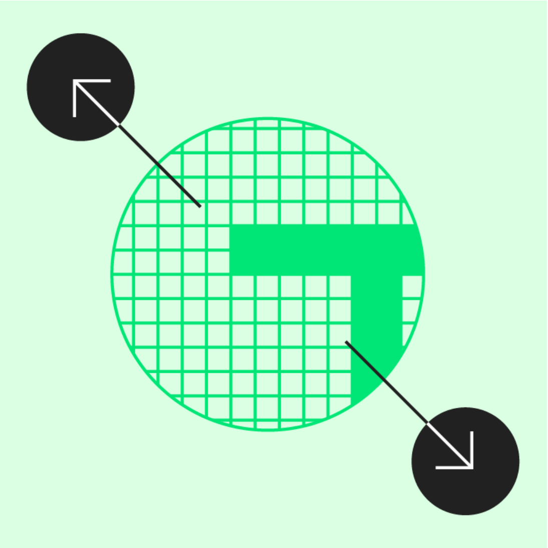


Fig. 54, the ubiquitous now two-finger "zoom in" pinch gesture.

User feedback.

One of the many problems of modern interfaces, especially touch interfaces, is that navigation, selection and operation can often prove problematic when moving from application to application and program to program. One of Material's many innovations is safeguards which allow users to avoid a situation in which they are unsure of how to proceed.

This is done through immediate feedback to the user. If they don't know how to proceed, parts of the screen will be highlighted, allowing them to instantly view the controls. If something can be interacted with, touching it will make it animate. If part of the screen is darker than the others, then that element is non – interactive. If the user has reached the end of a list, whether on the x or y axis, scrolling will make the interface rebound, showing them that there is nothing further to interact with.

Finally, when a state changes or an error occurs, a message is immediately show to the user. This message has one or two different selections to choose from. This allows for immediate reversal of the action, or acknowledgement of the state change. This lets the user always feel in control, and never as if the application is “getting away from them”.

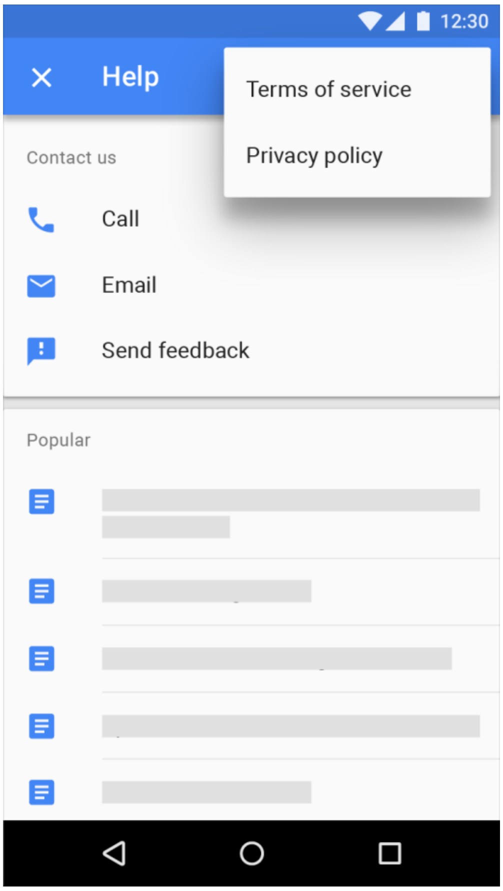


Fig. 55, the user interface focuses the user’s attention on one of the many functions, thus guiding them organically to their target. The large “X” on the top left also allows them to immediately step back, should they want to undo their command.

Material Design is not a revolution, but rather a refinement of existing technology.

Material Design and modern interfaces in general should not be seen as revolutionary necessarily, but rather as a refinement of existing technologies developed through years of iteration. It isn’t much of a stretch to assume that Material takes the better parts of user interfaces of the past and streamlines them into an easy to learn and access package, that could be operated by most users.

Instead of staunchly moving forward, Material takes into account animation from when interfaces were less simplistic, skeuomorphism in order to convey material qualities and depth, and the simpler, flat look of modern interfaces in order to avoid clutter and visual noise.

Modern user interfaces are the product of this refinement process and multiple years of iteration and experimentation. Their design is deceptively simple, and only when one goes through multiple years of user interface design, do they understand just how intricate and developed this process is. It is no simple task to make electronic devices react the way we expect them to do, yet every single day, through many hours of using our smartphone and computer, we do just that.

Conclusions

1. User interfaces are not something new, but have been in development for many years, and the most recent ones represent years of evolution and iteration over multiple decades.

2. User interfaces have developed often as a result of experimentation, instead of a deliberate design process.

3. User interfaces in the beginning would develop mostly in order to improve work and working efficiency.

4. Icons and the Desktop metaphor would allow more people to gain access and knowledge of computers and computing.

5. Design aesthetics would begin to play a larger role when more people would gain access to computers.

6. As time went on, user interfaces would evolve organically, with each generational leap being ushered in to address any issues of the previous design generation.

7. Technology evolved, evolving user interfaces along with them. Their evolution was often a direct response to the increased capabilities of the machines.

8. User interfaces would shift from not displaying enough information, to having too much information, to purposefully having very little.

9. The more powerful and complicated machines became, the simpler and more streamlined their interfaces became.

10. User interfaces may look wildly different today than they did 20 years ago, but the process of their design was always heavily iterative.

11. Even though unconventional technologies may develop in the future, user interfaces will still have to set usability, clarity and user experience at their forefront.

References

- Atkin, A. (2005). *Peirce On The Index and Indexical Reference*. Transactions of The Charles S. Peirce Society. 41 (1), 161–188
- Clement, C. (2002). *The History of the Xerox Alto*
- Cooper, G. (1990). *Cognitive load theory as an aid for instructional design*, Australasian Journal of Educational *Technology*
- Evans, C. (2013). "A Eulogy for Skeuomorphism". *Motherboard*.
- Ferdinand de Saussure (1959). *Course in General Linguistics*. *New York: McGraw-Hill.*
- Gatsou,C . Politis, A. Zevgolis, D.(2012*). The importance of mobile interface icons on user interaction "IJCSA International Journal of Computer Science and Applicati ons, Vol.9, No. 3, pp. 92 – 107,2012. (ISSN 0972-9038)*
- Google, (2022). *Material Design, Introduction, goals, principles* m1. material.io
- Greaney, P. (2006). Less is More: American Short Story Minimalism in Ernest Hemingway, Raymond Carver and Frederick Barthelme (phd thesis). *The Open University.*
- Grinberg, Y. (2013) iOS 7, Windows 8, and flat design: In defense of skeuomorphism, *VentureBeat*
- Hornor, T. (2015) *Why The Flat Design is Hurting Usability* vandelaydesign .com
- Janusheske, J. (2012). Thesis: Mimesis to Skeuomorph?
- Lee, T. (2011). Yes, Google "Stole" From Apple, And That's A Good Thing, *Forbes*
- Maxfield, C. (2011). How it was: Paper tapes and punched cards. *EE Times*.
- McJones, P. (2014). *Xerox Alto Source Code - The roots of the modern personal computer.* Software Gems: The Computer History Museum Historical Source Code Series. Computer History Museum.
- Moran, K. (2017). Flat UI Elements Attract Less Attention and Cause Uncertainty *Nielsen Norman Group (nngroup.com)*
- Müller-Prove, Matthias (1999) "The Interface of Kai Krause's Software", *online seminar*
- Norman, D. (1999) *The Design of Everyday Things,* Doubleday, N.Y.
- Nöth, W. (1990). *Handbook of Semiotics*. Indiana University Press

- Pavlus, J. (2013). Why Jony Ive is flattening iOS 7, *Fast Company*
- Pidyar, S. (2022). How UI design in 2022 will highlight Neumorphism, better *Experience Design – blog*
- Pouzin, L. (2000). The Origin of the Shell *multicians. org*
- Ron, A. (2014). The history of Android: the endless iteration of Google's mobile OS, *Ars Technica*
- Rounds,D. (2016). *A short history of Computer User Interface Design*, getfeedback.com
- Sharp, H.; Rogers, Y.; Preece, J.(2007*). Interaction Design: Beyond Human–Computer Interaction (2nd ed.)*. *John Wiley & Sons*
- Smith, D. (1988) *Fumbling the Future: how Xerox invented, then ignored, the first personal computer* New York, W. Morrow.
- Smith, N. (2021). Interfaces should be about feeling, *UX Collective, uxdesign.cc*
- Souppouris, A. (1990). Tracing iOS 7's influences: Apple remixes almost everyone in the industry*. The Verge*.
- Spiliotopoulos, K.; Rigou, M.; Sirmakessis, S. (2018). A Comparative Study of Skeuomorphic and Flat Design from a UX Perspective, *Multimodal Technol. Interact*
- Terror, D. (2009). Lessons From Swiss Style Graphic Design, *Smashing*
- The Teletype Corporation. (2021). The Teletype Story, *Hassell Street Press*
- Thompson, C. (2012). Clive Thompson on Analog Designs in the Digital Age. *Wired*.
- Turner, A. (2014) The history of flat design: How efficiency and minimalism turned the digital world flat, *The Next Web*
- Voorhies, D.; Scandura, J. M. (1977). Determination of memory load in information processing. *New York Academic Press*
- Wadlow, T. (1981) *The Xerox Alto Computer*, *Byte*
- Warren, T. (2019). *Microsoft unveils Windows Terminal, a new command line app for Windows. The Verge.*