

La Interfaz Hipertextual

1- Hipertexto e Interfaz

Durante el siglo XX, se produjo una revolución socio-cultural, acentuada por la revolución tecnológica que vivía la humanidad. El ser humano, como entidad, comienza a despojarse de lo material, comienza a formar una mente que se sostendría por la abstracción. La creación de la cámara fotográfica produce el nacimiento de la pintura abstracta, del surrealismo, del cubismo, del futurismo, e impulsa miles de corrientes vanguardistas. La posibilidad de grabar sonido incita a la creación del sonomontaje, que se aísla de sí mismo para abstraerse poéticamente. La mente del ser humano, incluso, deja de ser un sistema material. Ya no se habla de “cerebro”, ahora se habla de “pensamientos”. Se analizan los pensamientos, ya sea como postulaba Freud o como lo estimulaba Kandinsky. Lo tangible comienza a “aburrir” a la sociedad, en palabras de Maldonado, *el ser humano se ve atraído por lo fantasmático*.

Ahora bien, la pintura había mutado, la forma de producir música había cambiado, la idea de “pensamiento” era distinta, la ciencia, incluso, se *movía* diferente. Pero la literatura seguía estancada, como hace cientos de años. En todo ese movimiento vanguardista, hubo escritores que trataron de cambiar la forma en la que se contaban las historias. Como Borges con *El Aleph*, o Cortázar con *Final del Juego*. Los resultados fueron impactantes, sin lugar a dudas, pero no era lo que ellos buscaban, aún faltaban unas décadas antes de que la tecnología se encargara de crear el *soporte* necesario.

No le atribuyo al origen de internet como el punto de tensión exacto en el cual esto se da vuelta, sino al momento preciso en que *organismos* comenzaron a hacerse cargo de la gran masa global que es internet. A partir de ahí, estudiamos los *síntos web*. Hay que destacar lo importante de todo esto, ¿qué se estudia con exactitud? Básicamente, internet soporta dos sistemas complementarios: el *hipertexto* y la *interfaz*. Dos terminologías que han dado mucho para debatir en los últimos años, y aun así hay mucha confusión sobre las mismas. No son definiciones actuales, no obstante, tienen una aplicación más presente ahora que en el momento que fueron dichas.

Tenemos que discriminar ambos términos. No se trata de lo mismo, ni se trata de una diferenciación de un *todo*. Son dos cosas totalmente diferentes, complementarias, pero diferentes. Por un lado, el concepto de *hipertexto*. A mediados del siglo XX era un término muy popular, aquella característica que se buscaba de un *texto* como discurso. Borges define, en su cuento *El libro de arena*, que *un conjunto de espacios forman el hipervolumen*. Eso es, sino más, la definición más clara de lo que es un hipertexto, sin embargo, es solo el punto de partida de una definición. Un libro es un soporte lineal de una historia, la cual es contada con las limitaciones del soporte. A eso nos referimos cuando hablamos de un texto. En cambio, el hipertexto (el cual está soportado en los medios informáticos) no cuenta las cosas de manera lineal. No existen las jerarquías (al

menos no las que posee un libro), se establece un recorrido multilineal. Por lo tanto, la diferencia fundamental está en la abstracción. Un libro puede abstraer un texto, pero nunca un hipertexto. La informática es la encargada, como soporte, de abstraer un hipertexto y darle una utilidad. La historia es solo lo que se cuenta, no cambia el contenido, solo la forma de contarlo. Esto quiere decir que podríamos contar *La continuidad de los parques* de la forma lineal en la que la cuenta Cortázar o de una forma hipertextual. Siendo esta última a la que llamamos actualmente un *sitio web*.

Un relato lineal es clásico y ha sido siempre sostenido en soportes materiales, como un libro. En ese caso nosotros mismos somos la herramienta que nos permite acceder a lo que ese texto nos está contando. Pero ¿qué ocurre en un sistema intangible? ¿cómo accedemos a la información?. Era necesario plantearse eso al momento de proyectar un sistema informático. Nace, entonces, el concepto de *interfaz*. Rigurosamente hablando, el término interfaz es mucho más antiguo. Gui Bonsiepe lo menciona al analizar lo que ocurriría entre la interacción de usuario-objeto. Menciona tres componentes: el usuario, el objeto y la acción. Este sistema triangular admite que esos tres componentes están en constante relación unos con otros, ahora la pregunta es ¿cómo se sostienen?. Por inercia. Se produce un espacio vacío de articulación entre esos componentes. Esto es a lo que Bonsiepe denomina *interfaz*.

Podemos crear un hipertexto, pero no podemos cambiarlo. Sería pretender cambiar su formato multilineal, ya no sería un hipertexto. Es como si tratáramos de cambiar el formato lineal de un texto (eso se hizo cuando se creó el hipertexto, pero dejó de ser un texto). Por lo tanto, lo que se analizará y se creará, o innovará, es la interfaz que permite acceder a esa información, que brinda datos al usuario a través de una experiencia. Eso es lo que me dispongo a comentar en estas líneas.

2- Estudiar una Interfaz

Como buenos seres humanos que somos, el ser conservadores nos condiciona. Nos gusta ser conservadores, nos gusta lo constante, lo invariable. Desde Newton, y posteriormente la revolución industrial, se estableció que para estudiar ciencia hay que sentarse y leer, hay que resolver problemas, hay que observar, pero principalmente hay que formar la mirada científica. Hay que seguir procesos, quien los sigue entiende, quien entiende tiene la posibilidad de *cambiar al mundo*. Esto es válido para cualquier ciencia de la humanidad, ya sea *dura* o *blanda*.

Quien estudia biología, lee sobre los procesos fisiológicos y anatómicos de la vida, experimenta con sistemas orgánicos; quien estudia derechos lee las leyes, las analiza, las critica. De esta manera quien estudia álgebra solo logrará entender la lógica del sistema haciendo ejercicios; como quien estudia química solo entenderá la materia por experimentar con la misma. Con el arte siempre pasó algo distinto a ese modelo. El arte no puede estudiarse sino se elabora un *ojo crítico* primero. Criticar obras de *otros* nos da herramientas para aplicarlas a nuestras obras, eso se fusiona con nuestros conceptos, muta, se instala, se perturba, genera y contamina. Y

en ese ciclo está el estudio.

Una interfaz no es más que un elemento de producción artística. Es errado pensar que estudiar interfaz implica hacer interfaz. Eso solo mejoraría nuestra producción. El estudio de una interfaz implica elaborar una mirada crítica, aplicarla y reutilizarla. Implica descomponer cada partícula que la compone y analizarla con cuidado por separado y en colectivo.

3- Sistemas de Pre-Composición

Idea

No todo en el mundo informático es programar. Hay teorías interesantes que convierten a la informática en herramientas de aprendizaje, incluso varios estudiosos aseguran que la informática es, en esencia, un objeto artístico. Queda constatado que esa es también mi opinión. Ahora bien, ¿qué es lo que separa un sitio web comercial de uno experimental? ¿una aplicación estándar de una artística? Un simple planteo que, por desgracia, muy poca gente se hace a menudo: ¿qué quiero hacer?.

Una idea, en cuestión, no es, como muchos piensan, una *estrella fugaz* que pasa en algún momento de *inspiración*. Nuestra mente, por desgracia, nunca funcionó ni funcionará así. Existen puntos de quiebre, o núcleos disparadores, que muchas veces se dan en momentos inoportunos. Nuestra mente es hipertextual y es perfectamente capaz de combinar conceptos que parecen alejados en una unidad. Aún así, es necesario tomar ciertas medidas.

Si tenemos una “idea” lo primero es transformarla verdaderamente en una idea. Para eso es necesario un serio análisis. No importa de donde venga ese punto de quiebre, puede ser extraído de una película, una serie televisiva, un viejo cuento, un viaje al campo o hasta una teoría científica (como la experiencia me otorgó al leer el experimento del gato de Schrödinger). Sin embargo, lo primero que hay que hacer con esa idea es conceptualizarla, tratar de abstraerla en sí misma lo más posible para que solo quede su esencia. Recién conseguido eso, nos dedicaremos a simplificar la idea con el método de John Maeda: *Reducir y Organizar*. Buscaremos eso principalmente, reducir el concepto hasta que pueda ser muy sencillo de entender y de trabajar, y luego, organizarlo para que pueda ser interpretado sin problemas y utilizado en la web. De esta manera, estaremos trabajando con una idea sólida, la cual ya está sostenida por un proceso, un proceso de *autor*.

Investigación

Las ideas se establecen, pero se ven en la necesidad de no ser solo ideas. Los conceptos requieren materializarse (aunque suene un poco extraño hablar de materialidad en un sistema que es netamente intangible). Se necesita realizar un profundo análisis de datos, una gran investigación, antes de ponerse a

producir. Esto ayuda a darle forma a la idea y a solucionar problemas.

Cuando hablamos de “investigar” no solo es conseguir datos (como leer *Arquitectura para la WWW* de Morville y Rosenfeld) sino advertir, identificar y descomponer los componentes de un sitio web que compartan algún aspecto de nuestro concepto a trabajar. Por lo tanto, por conveniencia de lectura y explicación, se dividirá la investigación en tres etapas:

1. **Selección:** Una vez que accedemos a la información, deberemos realizar una selección pertinente. De nada sirve tener todos los datos desordenados y sin clasificar, y tampoco sirve tener toda la información disponible. Lo importante es tener la información más eficiente seleccionada. El reducir los datos es importante, no solo por una ganancia de tiempo, sino también por un aprendizaje más óptimo.
2. **Análisis:** No podemos simplemente tener la información disponible, hay que analizarla de alguna manera. Las posibilidades son infinitas, pero lo más recomendable es realizar un mapa de constantes y variables entre sitios. De esta manera tendremos una información mucho más valiosa que los datos mismos, tendremos un sistema sintético donde claramente se plantean que contenidos varían y cuales se mantienen constantes. Una variable es problemática en muchos casos, pero hay que pensar que están presentes y la razón es muy simple: las variables tienen tanto peso como las constantes. Si todo el hipertexto mantuviera siempre una constante, si no hubiese un bloque de ruptura, el usuario que recorre ese hipertexto optaría por dejar de recorrerlo.
3. **Conclusión:** Finalmente, teniendo una clara visión de los datos y habiéndolos analizado correctamente, emprendremos la tarea de tomar lo importante para nuestra interfaz. Las constantes por algo son constantes, y las variables varían por alguna razón. Si no mantengo en mi sitio constante lo que en sitios de temáticas similares si se mantiene, debo pensar muy seriamente como contrarrestar ese peso.

Entorno

Al igual que ocurre cuando uno realiza una animación, es necesario plantearse los objetivos comunicacionales que se tendrán como constante en todo el relato. Una interfaz requiere, incluso, un planteo más riguroso. Y es así como uno recurre a la rotoscopía cuando requiere producir una sensación de inmersión en el usuario, o a un stopmotion cuando necesita modificar la percepción. De la misma forma se recurre a diferentes “técnicas” o, mejor dicho, tecnologías en el soporte web. A esto es a lo que llamamos *entorno*. El concepto de entorno es suficientemente amplio como para escribir un ensayo completo dedicado a eso. Sin embargo, no es el objetivo de este escrito.

A continuación se presentan cinco categorías distintas que se desprenden del concepto de entorno. Desde un

punto de vista podríamos pensar que es una clasificación muy escolar, pero resulta muy útil y fácil de entender:

- **Entorno Estático:** Es un entorno de trabajo clásico, donde se estructura a través de cajas (generalmente en HTML, pudiéndose darle estilos con CSS). La estructura es de carga rápida, de lectura sencilla, casi sin interactividad, ideal para sitios de noticias o algunas galerías de imágenes. De sencillo manejo para el programador, propone un tiempo de carga ruda, donde las imágenes van apareciendo como una *persiana horizontal*, lo que produce un sentido más lento.
- **Entorno Dinámico:** Consiste en tomar la estructura del estático, pero se perfila a otra clase de llegada. A través de algunos lenguajes de programación (como Javascript) se le otorga a los elementos cierta dinámica, no son fijos, pueden ser animados, pueden intervenir, muy utilizado en sitios de autor, experimentales y la gran mayoría de las galerías. El tiempo es rudo, pero fluido, pueden realizarse ciertas *trampas* para que se genere un entorno incluso mas interesante en cuanto tiempo. Produce una sensación de inmersión.
- **Entorno de Objeto:** Posiblemente el tipo de entorno más famoso de internet, donde el sitio está creado a partir de la inclusión de un objeto (generalmente objetos de Flash, últimamente hay muchos de JAVA y, por supuesto, el objeto Canvas). Con la posibilidad de añadir una barra de carga, la información puede cargarse antes de ser mostrada. De esta manera la sensación de tiempo es muy neutral y equilibrada. Tiene el problema de que suelen requerirse filtros que no vienen incluidos en el navegador (excepto en Canvas, ahí su potencia). Muy utilizado en sitios de autor, de películas, comerciales y en experimentales.
- **Entorno orientado al Cliente:** Es el entorno dedicado a algún lenguaje que esté orientado al cliente (comúnmente utilizando JavaScript). Llamamos *cliente* al navegador con el cual se esté mostrando nuestro sitio web. Permite una gran fluidez del tiempo, con actualización del sitio en tiempo real. Eso otorga al usuario un tiempo calmado, nada rudo y con mucho soporte. Requiere una enorme cantidad de conocimientos en múltiples lenguajes, por lo que suele perfilarse para trabajar en grupos de personas especializadas y no de manera individual.
- **Entorno orientado al Servidor:** Es el entorno dedicado a algún lenguaje que esté orientado al servidor (comúnmente el lenguaje de PHP o un entorno AJAX). Se trata del entorno básico de las ya conocidas *redes sociales*, y es el motor principal de la web 2.0. Produce una sensación de tiempo frenético y pendiente, al ser actualizado permanentemente por usuarios ajenos a la *caja negra*. Tiene un grave problema en cuanto a planteo de estructuras, mapa de información y construcción del sitio, puesto que la conducta humana lo hace impredecible. Igual que un entorno orientado al cliente, es de construcción muy compleja y por lo general se requiere un grupo de trabajo.

Sistema

Este aspecto cierra la explicación y es bastante sencillo de explicar, el problema radica en llevarlo a cabo. Hay que ser muy detallista. Teniendo un concepto potenciado por una dedicada investigación, se define un entorno de trabajo lo cual nos otorga herramientas. Luego de esto hay que pensar en que interfaces se pondrán en juego, como se van a utilizar, como se van a relacionar con imágenes, textos y objetos, que tipografía se va a utilizar, que relación tendrá con la paleta cromática, que relación tendrá con el contenido, que clase de interactividad estará disponible, y podría mencionar varios aspectos más. En esto radica el sistema, en integrar, en relacionar y en definir cada elemento. Solo después de eso, nuestra mente estará ocupada en programación.

4- Cuadro Compositivo

La Grilla

Al momento de componer no podemos ubicar cada elemento en un lugar aleatorio, ni pensar que se van a sostener en el lugar que *quede mejor*. Es importante trazar lo que en programación web se denomina una *grilla*. Solucionar la grilla implica que se solucione la localización de cada bloque de texto, imagen y objeto. ¿Qué es la grilla? Es un conjunto de líneas imaginarias, que se forman por la distancia de márgenes asignados a cada estructura. Otorgan un sentido al sistema del sitio y, sobre todo, unidad a base de constantes. Un ejemplo muy básico sería mantener una caja marginada al centro de la pantalla durante todo el recorrido por el hipertexto. De esta forma, todo el contenido forma una unidad misma mientras se navega, se genera una constante. La concepción básica para su armado es la proporción que desee mantenerse, aún si se buscara expresar caos o desorden. La línea de inicio de margen mantendrá la misma proporción que la de cierre, el mismo caso ocurre para la línea superior y la línea inferior. Si se deseara que la línea de fin no tuviera la proporción con la línea de principio, todo el sistema perdería peso y entraría la necesidad de compensarlo con otro cambio en la grilla. Una solución podría ser cambiar la proporción de la línea superior hasta llegar a un resultado equilibrado. Cada elemento que se incluya altera la grilla, por lo tanto la grilla tiene que estar preparada para mutar, siempre es bueno ir de lo particular a lo general, tratando de tener simples grillas formadas para poder ir cambiándolas en el caso correspondiente.

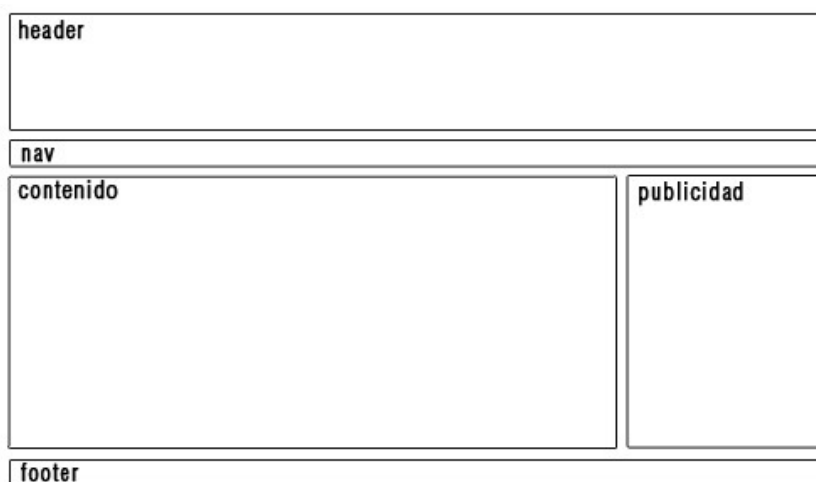
¿Todo el sitio tiene que tener la misma grilla? No es necesario, pero en principio sí. ¿Cuando la grilla puede cambiarse? Cuando se produzca una variable. Como se menciona anteriormente, las variables son tan importantes como las constantes, producen un quiebre, llaman la atención, llaman a la concentración en un elemento. Por lo tanto es posible “romper” la grilla general para acentuar esos efectos o bien para que se adapte. No obstante, se requiere previamente una buena grilla general sobre la cual trabajar. Además, hay

que tener en cuenta que la transición de esa ruptura no debe ser ruda, debe ser minuciosa y que no produzca ninguna clase de perturbación.

El Lay-Out

Se trata de un concepto antiguo en los sitios web, que quedó olvidado por la idea de grilla. No obstante, me interesa retomar la definición y separarlo de la grilla, siendo este otro elemento imprescindible para la construcción de un hipertexto. Previamente comenté que una grilla es una construcción imaginaria, construcción que ubica elementos de una forma estable, equilibrada. El *lay-out* es la grilla, pero en concreto. ¿Qué quiere decir? Que en un lay-out se toma la grilla previamente trazada y se comienzan a ubicar bloques para ver si funciona como distribución de contenido. Si no funciona, se vuelve un paso atrás, se modifica la grilla y se vuelven a trazar bloques de información. Es importante que se dibujen realmente, pues eso ayuda a entender mejor la distribución.

Una de las cosas más interesantes que nos permite un lay-out, y que no posee la grilla, es la posibilidad de agregar un nombre al bloque. De esta forma, al terminar el lay-out, tendremos una herramienta para utilizar al momento de diseñar el sitio, donde ya tendremos solucionada la proporción entre elementos, la ubicación, la disposición, el tamaño y la unidad.



Ejemplo de Lay-Out clásico.

La Interfaz

Es cierto que ya he mencionado previamente a la interfaz como un componente general de análisis y producción, siendo que, lo que en este ensayo se contempla, es como llegar a esa gran interfaz. También se establece a la interfaz como parte del sistema. Sin embargo, la interfaz que aquí se aborda es un aspecto más modular, más particular, que en su conjunto forma la enorme masa de interfaz colectiva hipertextual.

No solo vamos a pensar cada componente como conjunto de un sistema, sino como un universo propio dentro de su sistema individual. Quizás el título correcto de esta sección debería ser “*La interfaz individual*”, pero eso le quita atractivo.

Anteriormente se dio la definición de interfaz como aquel espacio de articulación. La interfaz hipertextual es aquel espacio que nos articula con un contenido hipertextual. La interfaz individual, en su conjunto, forma a la interfaz hipertextual. Por lo tanto, solucionar la interfaz individualmente nos soluciona parcialmente la problemática del hipertexto.

Lo fundamental es fragmentar los elementos y pensarlos como partes de un todo. Es importante que el puntero del *mouse* sea interesante en un sistema que no usa la metáfora del ordenador (como en interfaces para películas), pero es más importante ver que pasa cuando esa interfaz se acerca a otra interfaz, como una botonera de navegación superior. En este punto se establece una relación entre dos componentes intangibles, que a la vez son espacios vacíos donde se articula la acción usuario-contenido. Enigmático, en cierto punto, complejo, desde otra perspectiva, y muy sencillo a la vez. En el momento en que dos interfaces hacen *colisión*, entra en juego una tercer interfaz que contiene a las dos. Es un error muy frecuente que grandes diseñadores de interfaces produzcan sistemas que, individualmente, funcionan muy bien, pero poseen tanta presencia que compiten unos con otros al momento de *chocarse*. La solución es quitarles presencia a las interfaces desde lo individual, para evitar conflictos desde lo colectivo.

La interactividad

Anteriormente mencioné que darle un tratamiento minucioso a cada interfaz individual nos soluciona el problema del hipertexto de manera parcial. Puedo afirmar que, sin ánimos de dar una estadística precisa, el 50% del trabajo está en las interfaces individuales. El porcentaje restante se encuentra en la interactividad. El usuario acciona sobre una interfaz, el usuario está interactuando. Se produce una suerte de accionar recíproco, el cual *retroalimenta*, en un ciclo, al sistema. Si ese aspecto no está para nada cuidado, las interfaces no se sostienen. No solo porque el usuario terminará sin entender que efectos produce su presencia, sino también por la competencia entre las interfaces individuales (no olvidemos que las interfaces interactúan entre sí).

Tomemos el ejemplo de la unidad mínima de interactividad en un ordenador: el *click*. Un simple click puede hacer desde nada, pasando por intercambios de páginas, hasta complejas acciones de algoritmos. Si cuando hago click no pasa *nada*, tengo un problema. Si no va a pasar nada, la constante puede mantenerse, pero es imprescindible que esta se rompa (con un cambio de grilla, por ejemplo). ¿Cómo el usuario advierte que hizo click? ¿qué hace específicamente ese click? ¿hace siempre lo mismo? ¿cómo el usuario puede percibir que debe hacer click en un sitio y en otro no?. Si estas problemáticas no se contemplan, la interfaz caerá por su propio peso.

5- Beta-Tester

Los proyectistas y los programadores de interfaz son seres humanos. Saben que funciona y que no funciona, saben lo que es bello y lo que no. Sabemos, por el solo hecho de tener un razonamiento formado en visuales de cientos de años, lo que es estéticamente funcional. No obstante, una mente entrenada también es traicionera. Por lo tanto, en los últimos años, se ha iniciado un sistema de *testeo*. Proveniente del inglés *test*, que significa *prueba* o *probar*, *testeo* es una versión españolizada de la palabra que tiene el mismo significado. Esta práctica ya se realizaba hace varias décadas en la prueba de objetos industriales. Lo interesante es lo que comienza a plantearse con la masa global.

Normalmente un grupo de personas para un *testeo* cobraría por sus servicios, y nada garantizaba que sus opiniones realmente fuesen las de la mayoría. En la industria cinematográfica se le encontró una forma muy sencilla de solucionar el problema. Regalar entradas para ver el pre-estreno de una película, repartiendo tarjetas de opinión al respecto de la misma. Esto, en una escala mucho mayor, es lo que hace un *testeo* informático. Y es una escala mayor, porque se realiza de manera global.

Luego de analizar un poco a la sociedad virtual que posee el mundo en la actualidad, me doy cuenta que las personas (no solo las más fieles a la informática, sino también de otras disciplinas) sienten una atracción muy rara por formar parte de un grupo de *testeo*.

En lo que respecta a la informática en sí, suele hacerse una primer versión denominada *Alpha*. La versión alpha de una interfaz, normalmente es de prueba entre los programadores que la diseñaron u otros programadores. Corrigen errores técnicos, pero se requiere una mirada más suave todavía. Si esas interfaces son dirigidas a un público, es importante que el público vea previamente esos sistemas. Es por eso que se usa una versión *Beta*. Principalmente cuando se diseña una interfaz que ya es popular o que fácilmente puede moverse entre los usuarios, se da a conocer una versión beta. Los usuarios prueban posibles errores por el solo hecho de navegar, dejan miles de opiniones, discuten entre ellos y son globales. Un vil truco de manipulación, pero efectivo a la hora de probar interfaces.