

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACIÓN

Titulación:

INGENIERO EN INFORMÁTICA

Título del proyecto:

Determinación de la percepción del número de colores en el diseño de páginas web

Ekaitz Hernández Troyas
Carlos Sáenz Gamasa
Pamplona, 28-6-2012

Tabla de contenido

1. Objetivos	5
2. Antecedentes	6
2.1 Sistemas de color	6
2.2. Percepción y uso de colores en páginas web	6
2.3 Creando puntos de atención en las páginas web	10
2.4. Los colores en los monitores	11
3. Aplicación	13
4. Descripción general de la aplicación.....	14
5. Funcionamiento de la aplicación.....	23
5.1. Lenguajes de programación utilizados	23
5.2.1. HTML	23
5.2.2. JavaScript.....	23
5.2.3. AJAX.....	24
5.3.4. JSON.....	24
5.3.5. PHP	25
5.3.6. MySQL	25
5.3.7. CSS	25
5.2. Arquitectura	27
5.3. Aplicación.....	28
5.3.1 Instrucciones.....	28
5.3.2. Formulario	28
5.3.3. Test.....	30
5.3.4. Feedback.php.....	42
5.4 Base de datos	44
5.4.1 Requisitos	45
5.4.2. Diseño lógico	45
5.4.3. Diseño físico.....	47
6. Resultados	52
6.1. Número de colores importantes.....	52
6.2. Número de colores importantes separados por sexo	53
6.3. Área de la imagen ocupada por los colores elegidos.....	55
6.4. ¿Existen diferencias entre las imágenes seleccionadas respecto del número de colores elegidos?	58

6.5. ¿Existen diferencias entre las imágenes seleccionadas respecto de la fracción de área de la imagen ocupada por los colores seleccionados?	62
6.6. ¿Se escogen preferentemente aquellos colores que representan un porcentaje mayor del área?.....	64
6.7. Conclusión de los resultados	76
7. Conclusiones.....	78
8. Bibliografía	79
Anexo.....	80
1. Código	80
1.2. Instrucciones.....	80
1.2.1 instrucciones.html	80
1.3. formulario.....	82
1.3.1. formulario.html.....	82
1.4. Test.....	84
1.4.1. index.html.....	84
1.4.2. paleta.js.....	87
1.4.3. procesoSelectorDeImagenes.php.....	94
1.4.4. procesoBarra.php.....	95
1.5. Feedback.....	96
1.5.1. feedback.php	96
1.5.2. iniciarFeedback.php.....	99
1.5.3. jsondecode.php	99
1.5.4. cargarImagenFeedback.php	100
1.5.5. tratamientoFeedbacUsuarioFila.php.....	100
1.5.6. tratamientoFeedbackRestosUsusarios.php.....	102
2. colores.json	103
2.1 Fichero colores.json.....	103
2.1 Programa Java para la creación del fichero colores.json.....	105
3. Imágenes testeadas	107
3.1. Imágenes testeadas por los usuarios.....	107
3.2. URL de las imágenes almacenadas en el servidor	108

1. Objetivos

El objetivo del presente trabajo fin de carrera es la determinación mediante un experimento psicofísico, del número de colores que los usuarios de Internet perciben cuando juzgan el diseño de una página o sitio web.

Para la consecución del este objetivo general se han planteado las siguientes tareas y objetivos concretos:

- Seleccionar las páginas web que se utilizarán como estímulo en el experimento psicofísico.
- Desarrollar una aplicación web accesible a través de Internet para la gestión automatizada del experimento psicofísico con capacidad para recopilar, de forma automatizada, los resultados obtenidos de la participación de cada uno de los usuarios.
- Publicar la página web y recabar un número estadísticamente significativo de valoraciones de distintos usuarios.
- Analizar los datos obtenidos de acuerdo con el objetivo del proyecto.

2. Antecedentes

2.1 Sistemas de color

Existen dos sistemas de color, el sustractivo y el aditivo, que se utilizan según convenga al soporte donde se aplican -no es lo mismo un monitor que una revista-.

El sistema sustractivo mezcla tintes, pinturas, colorantes, etc. Así, para conseguir los distintos pigmentos, este sistema parte de los colores primarios, el cian, el magenta y el amarillo, y los combina hasta dar origen al resto. Los colores secundarios, el verde, el rojo y el violeta, se obtienen de las diferentes parejas de primarios y el negro, de la suma de todos en proporciones equitativas.

En el sistema aditivo, en cambio, los colores surgen por la emisión directa de luz. Así, para representar el espectro, se toman el rojo, el verde y el azul como fuentes primarias y, a partir de las mismas, se obtiene el cian, el magenta y el amarillo. A diferencia de la fórmula anterior, aquí la combinación equitativa de todas las fuentes genera el blanco. Este sistema es el que suelen emplear los monitores.

2.2. Percepción y uso de colores en páginas web

El uso de los distintos colores y la forma en la que éstos son percibidos es un aspecto clave en el diseño de páginas web. Tanto es así que la elección de los mismos puede beneficiar el sitio o imposibilitar su navegación. Las malas decisiones en cuanto a tonalidades, proporciones y posiciones, no sólo restan armonía al diseño creando estrés en el usuario, sino que pueden llegar a alterar la percepción de los elementos de la propia página web. Cualquier error cromático puede conllevar, por ejemplo, la confusión de los enlaces con el fondo de la página y tener como resultado una usabilidad nula.

Percepción de los colores

Los colores son percibidos de forma diferente según edad, sexo, estudios o nacionalidad. Así, las personas mayores se decantan por los tonos claros pero intensos, las mujeres, por los de menor intensidad, quienes cuentan con estudios especializados en diseño, moda o arquitectura, por los tonos pastel y los españoles y los argentinos, por los colores fríos.

Además, algunos observadores pueden presentar deficiencias en la distinción de los colores. Las más severas imposibilitan totalmente la discriminación, ocasionando en sus afectados una percepción monocolor. Las más leves, entre las que se encuentra la deuteranopia, que es la más habitual, permiten cierta diferenciación, tal como muestra la imagen. Así ve la bandera del arcoíris una persona aquejada por esta disfunción.

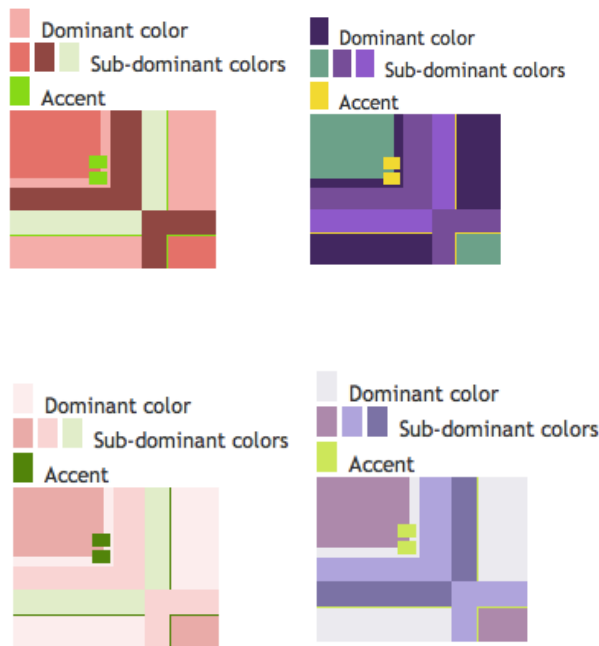


Uso de los colores en páginas web:

Los colores no se perciben de manera aislada: se ven afectados por los elementos que los rodean y de los que forman parte. De ahí que, cuando dos colores figuran juntos, se aprecie una mezcla de ambos y que acertar en el uso de los mismos sea fundamental. La elección de colores y su disposición determinan de forma decisiva la percepción de la página, aunque, como se ha visto, ésta no depende exclusivamente del diseño, sino también de la subjetividad del usuario.

Se suele distinguir entre colores dominantes, los que ocupan mayor proporción, y subdominantes, los que radican en áreas más pequeñas. Normalmente, estos colores comparten matices parecidos, aunque pueden llegar a contrastar por su disparidad en saturación, tono e intensidad. En terminología inglesa, se conocen como *accent colors* y

destacan porque tienen el poder de atraer la atención del usuario, aunque sólo ocupen una pequeña parte del total de la página web.



El uso de tres colores principales es el más recomendado en las guías de diseño de sitios web y, si se echa un vistazo a la red, también es el que más se encuentra. Los expertos aconsejan además que ese esquema tricolor responda a un buen balance entre el blanco, el negro y los colores de la marca. No seguir las recomendaciones básicas suele generar gran inconsistencia, por lo que resulta difícil encontrar diseños que se arriesguen a innovar en ese sentido, sobre todo, en las grandes corporaciones.

Ejemplos:

<http://www.movistar.es/>: azul, blanco y negro.

<https://www.bancosantander.es>: rojo, blanco y negro.

<http://www.acciona.es/>: rojo, blanco y negro.

<http://www.elcorteingles.es/>: verde, blanco y negro.

<http://www.ryanair.com/es>: amarillo, azul y blanco.

<http://www.aerlingus.com/>: verde, azul y blanco.

Casi todos los sitios web profesionales cumplen las indicaciones recogidas en las guías de diseño. Por norma, los colores de la marca se suelen emplear en las barras de

navegación, los botones y los links, sin embargo, apenas tienen protagonismo en la página web. El resto de colores elegidos en cada caso se suelen usar para el texto y el fondo, facilitándole, de este modo, la lectura de la página al usuario.

Los colores saturados son los más utilizados para los enlaces, los botones, las alertas y las zonas a las que se pretende dirigir la atención, sin embargo, el texto principal se coloca sobre el fondo más claro de la página para permitir la focalización del contenido. Por la misma razón, los expertos desaconsejan el uso de muchos colores.

Los colores con poca saturación suelen emplearse en los fondos de las web, las cabeceras, los fondos de los menús y el resto de fondos en general. Este tipo de tonalidades son recurrentes en áreas amplias, así que, suelen establecerse como colores dominantes. Del mismo modo, el cuerpo principal de las páginas web tiende a constituir la parte menos saturada de las mismas para no impedir la correcta legibilidad del texto.

Los colores dominantes se contraponen a las zonas saturadas a las que se quiere dirigir la atención, facilitando así la navegación por la web del usuario, quien suele otorgar mayor importancia a las áreas resaltadas por los tonos saturados a pesar de que éstas suelen ocupar menos espacio en la página.

2.3 Creando puntos de atención en las páginas web

Uno de los objetos del diseño web es conseguir que ciertas áreas de la página destaquen más que otras, constituyendo lo que se referirá como “dominio”. El énfasis que se le da a determinadas partes de la web a través del uso de colores, las formas de los elementos, los distintos tamaños de los mismos o su distribución, pretende dirigir la mirada del usuario hacia las zonas deseadas y contribuir a crear el área de atención que todo buen diseño debe tener.

En las páginas web, suele haber tres zonas de dominio principales:

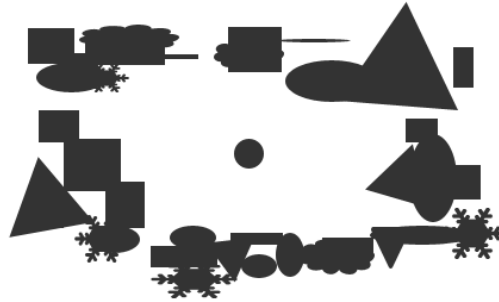
- Dominante: es la parte que más atención visual tiene.
- Sub-dominante: es aquella constituida por el elemento o los elementos secundarios de la composición.
- Subordinados: son los elementos que menos peso visual tienen en la estructura de la página web.

Cuando la página web no cuenta con un dominio, todos los objetos pueden tener el mismo nivel de atención y el usuario tiene que esforzarse para encontrar lo que busca. De ahí que sea necesario crear zonas de dominio en la página que faciliten la navegación al usuario y mejoren la usabilidad de la página.

Cómo se crean áreas de dominio en una página web:

Un área de atención se puede crear aumentando el tamaño de los elementos que se quieren resaltar o aplicándoles un color más saturado que al resto. El uso del espacio en blanco (entendido como área vacía, no como zona de ese color) también puede contribuir a tal fin, pues consigue destacar elementos concretos de la página. Así, un sitio web cuya densidad de elementos confunde al usuario, puede recurrir a los espacios en blanco para corregir ese problema. Ludwig Mies van der Rohe es un exponente claro de la efectividad de este concepto aplicado a la arquitectura.

La siguiente figura demuestra cómo un elemento que ocupa un área pequeña de la página es capaz de captar la atención gracias al recurso del espacio en blanco que le aísla de los demás.

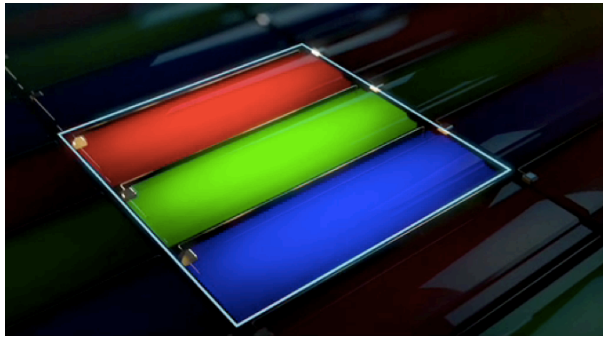


2.4. Los colores en los monitores

Los colores en los monitores se originan mediante síntesis aditiva. En concreto, los monitores utilizan el modelo RGB (Red, Green & Blue), cuyas siglas hacen referencia al rojo, al verde y al azul, que se reproducen con diferentes intensidades para crear el resto de variaciones del espectro.

Cada píxel del monitor emite un color mediante el modelo RGB y está compuesto por tres bytes, que a su vez pueden generar de 0 a 255 variaciones de colores cada uno. Esta numeración se suele expresar en base 16, también conocida como hexadecimal, donde 255 equivale a FF en base 16. Al tener tres bytes, el número de combinaciones posibles según este sistema sería 16.581.375 colores.

La siguiente imagen muestra cómo se vería un píxel si se ampliase lo suficiente una pantalla. Examinándolo de cerca, se podrían llegar a apreciar las tres luces que generan las diferentes intensidades de rojo, verde y azul.



Los monitores disponen de diferentes tecnologías para la reproducción de colores: CRT, LCD, plasma o TFT. Estas tecnologías se diferencian entre sí en términos de luminancia, contraste, definición y frecuencia de refresco, lo que provoca algunas variaciones en la emisión y ocasiona que los colores percibidos por un mismo usuario sean interpretados de forma distinta en función del monitor donde realice el test. Este proyecto no ha podido estudiar la cuestión en laboratorio cerrado por la falta de una muestra representativa para el análisis.

3. Aplicación

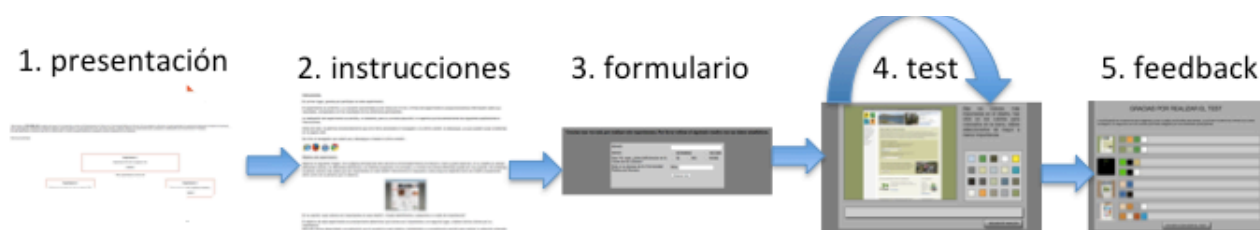
Como se explica en el apartado de objetivos, se quiere estudiar la preferencia subjetiva de los colores en páginas web por parte de los usuarios. Para ello se han elegido cuarenta páginas y se han obtenido sus correspondientes imágenes en formato GIF. Conviene recordar que estas imágenes no mantienen la “profundidad” original de la página, pues se reducen a 8bits, es decir, a 256 colores, pero también que dicha disminución apenas afecta a la percepción del usuario.

Cada usuario analizará cuatro imágenes, que serán distribuidas entre los participantes de manera que todas se observen el mismo número de veces. Para el análisis, el sistema mostrará una paleta con los 20 colores con más área que se situaran de forma aleatoria sin que puedan resultar previsibles en caso de que se muestre la misma imagen una segunda vez. La paleta estará formada por los colores que ocupen más área en la página, entendiendo por área la frecuencia con la que aparece una tonalidad determinada.

El usuario tendrá que elegir los colores que considera más característicos del diseño y ordenarlos en función de su importancia. Para ello, se ha creado una página web que posibilita la obtención inmediata del resultado del usuario y envía un *feedback* al usuario informándole de las respuestas que ha dado y las que han aportado los otros usuarios. Con esto, se pretende aumentar el interés de los participantes en la realización del test y disminuir así el número de usuarios que abandona el experimento por la mitad.

4. Descripción general de la aplicación

Lo primero que se muestra es la presentación, que dirige al usuario a las instrucciones de uso y al consiguiente formulario. Una vez que se haya completado el mismo, se puede proceder a la realización del análisis de las cuatro imágenes y, al finalizar, se obtiene el *feedback* correspondiente. En esta pantalla, el usuario tiene la opción de volver otra vez la prueba, si así lo desea, será redirigido a la pantalla del formulario. El siguiente diagrama de flujo refleja las diferentes partes de la aplicación, que son explicadas más detalladamente a continuación.



Partes de la aplicación

Presentación: esta página expone los diferentes experimentos del Departamento de Física de la Universidad Pública de Navarra que se pueden realizar. Actualmente cuenta con tres opciones. La del recuadro más grande corresponde al “Experimento 1: importancia del color en páginas web”.

Bienvenidos a **COLORLAB**, la página del grupo de investigación en óptica del Departamento de Física de la Universidad Pública de Navarra. En esta página te ofrecemos el poder participar en experiencias dedicadas al estudio de la percepción del color. Con tu participación contribuirás al estudio de algunos aspectos relativos a la percepción del color, de forma sencilla y cómoda. Realizar los experimentos que se proponen a continuación no te llevará más de 15 minutos. En cada experimento te daremos unas breves instrucciones y al final te proporcionaremos un pequeño resumen comparado de tus respuestas.

Gracias por participar.



The screenshot shows a web interface for selecting experiments. At the top, there is a box for 'Experimento 1: Importancia del color en paginas web.' with an 'empezar' button. Below it, the text 'Otros experimentos en esta web:' is centered. Underneath, there are two more boxes: 'Experimento 2: Preferencias de diseño de color en páginas Web.' and 'Experimento 3: Proporciones de color y equilibrio cromático.', each with an 'empezar' button.

Instrucciones: aquí, a través de diferentes capturas de imagen, se explica en qué consiste la página y la dinámica de uso de ésta. Si el usuario quiere, podrá descargarse las últimas versiones de los navegadores para solucionar los problemas de compatibilidad que pudiera tener con la página.

El siguiente texto es lo que le aparecerá al usuario:

Instrucciones.

En primer lugar, gracias por participar en este experimento.

El experimento es anónimo y su duración aproximada de 15 minutos. Al final del experimento le proporcionaremos información sobre sus resultados, comparados con los resultados de los anteriores participantes.

La realización del experimento es sencilla, no obstante, para su correcta ejecución, le rogamos que lea atentamente las siguientes explicaciones e instrucciones.

Antes de todo, le pedimos encarecidamente que sino tiene actualizado el navegador a la ultima versión, la descargue, ya que pueden surgir problemas en la pagina web.

De click al navegador que usted usa y descargue e instale la ultima versión:



Objetivo del experimento:

Observe la siguiente imagen. Es la página principal del sitio web de la Universidad Pública de Navarra. Como puede observar, en su diseño se utilizan diferentes colores, en diferentes elementos y en distinta proporción. En principio, el número de colores diferentes puede ser muy grande. Sin embargo, ¿Cuántos colores cree usted que son importantes en este diseño? Obviamente la respuesta a esta pregunta depende tanto del diseño propiamente dicho como de la persona que lo observa.



En su opinión ¿qué colores son importantes en este diseño?, ¿Puede identificarlos y asignarles un orden de importancia?

El objetivo de este experimento es precisamente determinar qué colores son importantes y en segundo lugar, ordenar dichos colores por su importancia.

Para ello hemos desarrollado una aplicación que le ayudará en este objetivo, facilitándole un procedimiento sencillo para realizar la selección ordenada de colores según su importancia.

Desarrollo del experimento.

Tras pedirle algunos datos estadísticos (sexo, edad, etc) comenzaremos el experimento.

Cuando comience el experimento, el programa le mostrará una pantalla similar a ésta:

•



The image shows a screenshot of the Livescribe Pulse website. The website has a dark blue header with the 'pulse' logo and the tagline 'NEVER MISS A WORD'. Below the header, there are several promotional banners for 'FREE SHIPPING', 'Macworld', and 'Popular Science'. At the bottom of the website, there is a navigation menu with links like 'Home', 'Pulse', 'Livescribe', 'Dot Paper', 'Photo', and 'Movies'. To the right of the website screenshot, there is a color selection palette consisting of a 4x5 grid of colored squares. The colors include various shades of gray, blue, red, and orange. Below the color palette, there is a large, empty rectangular box. At the bottom right of the interface, there is a button labeled 'SIGUIENTE IMAGEN'.

Elije los colores más importantes en el diseño, haz click en los colores para colocarlos en la barra, debes seleccionarlos de mayor a menor importancia

En ella se muestra a la izquierda la imagen de una página web. A la derecha se muestra una paleta formada por 20 colores. En la parte inferior hay una barra horizontal donde usted debe colocar los colores según su propio criterio.

Para colocar un color en la barra de resultado, simplemente pulse con el ratón sobre uno de los colores de la paleta (por ejemplo sobre el cuadrado rojo, como se muestra en el ejemplo). Verá que el color desaparece de la paleta y aparece en la barra de resultados.



Para elegir un segundo color, pulse con el ratón sobre él en la paleta. El color se colocará a continuación del anterior en la barra de resultados.

Puede repetir el proceso tantas veces quiera, traspasando colores de la paleta a la barra de resultados.

Elige los colores más importantes en el diseño, haz click en los colores para colocarlos en la barra, debes seleccionarlos de mayor a menor importancia.

SIGUIENTE IMAGEN

Elige los colores más importantes en el diseño, haz click en los colores para colocarlos en la barra, debes seleccionarlos de mayor a menor importancia.

SIGUIENTE IMAGEN

Si quiere rectificar una elección, para deshacer la selección, pulse sobre el color en la barra de resultados y el color volverá a la paleta, al mismo lugar donde estaba antes. Solamente podrá deshacer el

último color añadido, pero repetir este proceso tantas veces como desee.



Recuerde:

Debe elegir solamente aquellos colores que le parezcan importantes en el diseño.

Debe colocar los colores en la barra de resultados por su orden de importancia.

Cuando esté satisfecho con su elección, pulse sobre el botón situado a la derecha de la barra de selección.

El programa le mostrará un total de cuatro imágenes para su evaluación. Por favor intente evaluarlas todas. La validez de los resultados obtenidos depende de su colaboración.

Una vez realizadas las cuatro series, el programa le mostrará los resultados de su evaluación comparándolos con los resultados promedios obtenidos por todos los usuarios que han realizado el experimento hasta la fecha.

Si ha comprendido la mecánica del experimento, haga pulse

[CLICK PARA EMPEZAR EL TEST](#)

Formulario: se le pide al usuario que proporcione los datos relativos a su edad y sexo y que especifique si tiene alguna deficiencia en la distinción de los colores. El último campo sólo deben rellenarlo los alumnos de la Universidad Pública de Navarra. Esta información se utiliza posteriormente en el análisis de los resultados.

Gracias una vez más por realizar este experimento. Por favor rellene el siguiente cuadro con sus datos estadísticos.

EDAD:	<input type="text"/>
SEXO:	<input type="radio"/> HOMBRE <input type="radio"/> MUJER
Que Vd. sepa, ¿tiene deficiencias en la visión de los colores?	<input type="radio"/> SI <input type="radio"/> NO <input type="radio"/> NOSE
Solo si es alumno de la Universidad Publica de Navarra	NIA: <input type="text"/>
<input type="button" value="Empezar test"/>	

Test: en esta parte, el usuario procede a la realización del experimento. Se le presentan un total de cuatro imágenes correspondientes a las capturas de diferentes páginas web. Al lado de las mismas, se dispone una paleta con veinte colores entre los que el usuario deberá elegir los que considere más importantes y significativos para el diseño, que ordenará según su importancia. Sus respuestas se colocan en una barra, donde queda reflejada la selección resultante. Cuando el usuario termina la elección de los colores pertinentes, debe dar clic al botón siguiente, momento en el que toda esa información se guarda en la base de datos.



Elige los colores más importantes en el diseño, haz click en los colores para colocarlos en la barra, debes seleccionarlos de mayor a menor importancia

Feedback: en esta parte, se le muestran al usuario las imágenes y junto a cada una de ellas, dos barras. En la primera figuran los colores que él ha elegido y en la segunda, los colores promedio elegidos por los participantes anteriores.



5. Funcionamiento de la aplicación

5.1. Lenguajes de programación utilizados

Este experimento se ideó dando prioridad a dos necesidades principales: la usabilidad y la velocidad de codificación y eficiencia de ejecución de la página. Tales requisitos decantaron la materialización del proyecto por el uso de HTML, JavaScript, AJAX, CSS, JSON, MySQL, PHP, Adobe Muse, pues esta serie de tecnologías proporcionan más interactividad a la página y facilitan la labor de codificación, aumentando así la eficacia y la eficiencia.

5.2.1. HTML

HTML (*Hyper Text Markup Language*) es un lenguaje de marcado. Sirve para estructurar el sitio web e indica de qué elementos está compuesta una página determinada. Con esta tecnología también se puede realizar la apariencia de la página (colores, maquetación, etc.), aunque no se recomienda, pues para esto existe una tecnología más eficaz, las CSS (*Cascade Sheet Style*).

5.2.2. JavaScript

JavaScript es un lenguaje de programación interpretado. Se utiliza en la parte del cliente y consigue aumentar el dinamismo de las páginas. Al ser ejecutado en el cliente, deja libre el servidor para otros procesos.

Usos de JavaScript:

- Detectar acciones del usuario y realizar acciones en consecuencia: cambiar el diseño de la página web, por ejemplo.
- Realizar el control de formularios: avisa al usuario de forma dinámica cuando introduce un dato incorrecto.
- Almacenar datos del usuario de forma momentánea, antes de enviarlos al servidor.

5.2.3. AJAX

AJAX (*Asynchronous JavaScript And XML*) se define como un conjunto de técnicas de desarrollo web que utiliza el lenguaje de programación JavaScript. Con AJAX podemos enviar datos asíncronamente al servidor sin interferir en el uso de la aplicación por parte del usuario. La sigla *X* es de XML (*eXtensible Markup Language*), un lenguaje de marcado estándar para el intercambio de información estructurada. Sin embargo y a pesar de que este lenguaje de marcado se usa muy habitualmente en AJAX, para este proyecto, se ha optado por utilizar un lenguaje de intercambio de datos diferente, JSON.

5.3.4. JSON

JSON (*JavaScript Object Notation*) es un lenguaje de marcado para el intercambio de información estructurada descriptivo, como también lo es XML, lo que significa que ambos describen los fragmentos de texto pero no cómo tienen que ser representados. Para explicar la propia estructura del texto, JSON utiliza etiquetas, una forma ligera de transmitir los datos a través de la web, ya que permite transferir la información y posteriormente representar los datos según convenga.

Concretamente JSON hereda la notación del lenguaje JavaScript, su notación está muy relacionada con la orientación objetos. La forma de acceder a los elementos en JSON es muy parecida a la del atributo de un objeto en Java. Además, también es posible definir funciones dentro de un fichero JSON y acceder a éste como se hace a un método en Java.

Ventajas de JSON frente a XML:

- Al usar una notación orientada a objetos, JSON resulta mucho más sencillo.
- Escribir el analizador sintáctico es más fácil. Esto ha contribuido a que el lenguaje se haya popularizado en diferentes entornos de programación.
- La rapidez de acceso es superior.

En definitiva, la gran sencillez de utilización del código generado por el analizador sintáctico y su rapidez de acceso frente a XML convierten a JSON en un candidato óptimo para los fines específicos de este proyecto.

5.3.5. PHP

PHP (*Personal Home Page*) es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Permite introducir pequeños fragmentos de código en el propio código HTML, que el navegador web interpretará para ejecutar las acciones que correspondan.

Se pueden utilizar todas las características que se suelen emplear en un lenguaje de programación al uso: variables, bucles, controles de flujo, consultas a base de datos, etc. Esto hará que se pueda obtener una web más dinámica.

En esta página web, el uso de PHP se ha restringido a la interacción con el servidor y a la parte de representación de la página Feedback. Esto se ha dispuesto así, porque gran parte del código restante se ha realizado en JavaScript en vez de PHP.

5.3.6. MySQL

MySQL (*My Structured Query Language*) es un sistema SGBD (*Sistema Gestor de Bases de Datos*). Un SGBD sirve de interfaz entre las aplicaciones y la base de datos, proveyendo una serie de funciones para insertar información, consultar datos, administrar permisos, etc.

Se ha utilizado el SGBD MySQL porque su uso es generalizado y existe abundante documentación en Internet que permite resolver cualquier duda. Además, es gratuito y se apoya en el sistema de consultas SQL, que es un estándar de facto usado en las consultas a bases de datos.

5.3.7. CSS

CSS (*Cascading Style Sheets*) define la presentación gráfica de HTML, creando una independencia entre la estructura del contenido y su forma de representarlo. HTML se usa para describir la estructura de los elementos de la página web y CSS, para la

representación de estos elementos. Coloquialmente se puede decir que CSS es el lenguaje de maquetación de HTML.

Las ventajas de CSS son que cambiar el diseño de una web resulta mucho más rápido y sencillo, ya que todo el código relacionado con el diseño de la página se centraliza en un fichero para que se pueda aplicar a las páginas que se quiera. También provee funciones sencillas para crear dinamismo en la página: posibilitando, por ejemplo, que el usuario pueda cambiar los colores de la página a través de la realización de determinadas acciones.

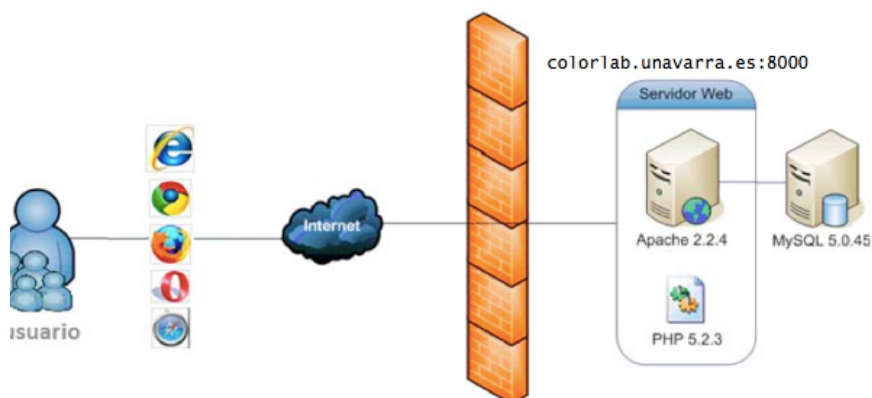
4.3.8. Adobe Muse

Adobe Muse es una aplicación que se ha utilizado para diseñar la maquetación de la página web. A través de una interfaz WYSIWYG (*What You See Is What You Get*), se puede concebir fácilmente la web sin ahondar demasiado en la parte del diseño.

Adobe Muse crea un fichero *.muse*, pero éste se puede exportar a HTML para que genere los ficheros CSS y JavaScript necesarios si se quiere usar como una aplicación web.

5.2. Arquitectura

La arquitectura se basa en un modelo cliente-servidor. La arquitectura cliente-servidor es una propuesta donde el servidor actúa como proveedor de contenidos y el cliente como demandante de los mismos. Este modelo cuenta con la ventaja de centralizar la mayor parte de su información en un servidor, haciendo que el mantenimiento sea más sencillo y económico. La siguiente imagen figura representa el modelo cliente-servidor.



De la figura anterior, se interpreta que se puede disponer de un servidor de páginas web Apache que trabaje con PHP para realizar las consultas al sistema de gestión de ficheros MySQL. Almacenar todos los códigos en el servidor es posible gracias a un protocolo de transferencia de ficheros FTP (*File Transfer Protocol*), sistema que se emplea para transferir códigos de páginas web desde cualquier lugar.

Para transferir los ficheros mediante FTP, se necesita un cliente FTP. Uno de los más recurridos en estos casos y el que se ha utilizado en el proyecto es FileZilla. Desde el exterior, la conexión al servidor FTP se realiza a través de la ip 130.206.158.228 y del puerto 2100, registrando como usuario *colorlab*. Dentro del campus de la Universidad Pública de Navarra, se puede acceder directamente desde la url *colorlab.unavarra.es* y el puerto 2100.

El servidor web utilizado en este experimento de la Universidad Pública de Navarra es Apache y se encarga de proveer páginas http. Funciona bajo UNIX, BSD, GNU/Linux, Microsoft Windows y Mac OS X. Para realizar el test, los clientes pueden conectarse a este servidor desde sus casas utilizando la dirección URL “colorlab.unavarra:8000”.

5.3. Aplicación

El código de la aplicación se puede organizar en cuatro apartados:

- Instrucciones
- Formulario
- Test
- Feedback

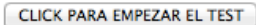
5.3.1 Instrucciones

Esta parte es bastante sencilla. Se trata de un texto que explica al usuario el funcionamiento del test. Posee también una serie de iconos que enlazan a las últimas versiones de los navegadores web, porque se recomienda la descarga de los mismos para que la página funcione en óptimas condiciones. Es interesante reseñar además que esta parte no cuenta con ningún código especial, sólo HTML.

- De click al navegador que usted usa y descargue e instale la última versión:



Al final de las instrucciones, se halla el botón que da inicio al test.

- Si ha comprendido la mecánica del experimento, pulse el siguiente botón 

Este botón funciona mediante un evento *onclick()* de JavaScript, hacer clic implica ser dirigido a la página formulario.html.

5.3.2. Formulario

Una vez que el usuario ha leído las instrucciones, es redirigido a esta página para responder a las siguientes cuestiones:

- EDAD: indicar su edad.
- SEXO: elegir entre HOMBRE o MUJER.
- Que Vd. sepa, ¿tiene deficiencias en la visión de los colores?: contestar SÍ, NO O NOSE.
- NIA (*Número de Identificación del Alumno*): facilitar su número de identificación como alumno de la Universidad Pública de Navarra si corresponde.

EDAD:

SEXO: HOMBRE MUJER

Que Vd. sepa, ¿tiene deficiencias en la visión de los colores? SI NO NOSE

Solo si es alumno de la Universidad Publica de Navarra NIA:

Implementación:

Esta parte de la aplicación se realiza mediante el fichero formulario.html, compuesto por un cuestionario construido mediante la etiqueta `<form>` de HTML. Cuando el usuario hace clic en el botón “Empezar test”, activa el evento `validateForm()` de JavaScript, que valida los datos introducidos en el formulario. Si éstos datos son correctos, se procede a dar de alta al usuario en la base de datos mediante el método *action* a “`alta.php`”, de lo contrario, se informa al mismo sobre el error encontrado en el formulario.

Función JavaScript validateForm():

Esta función controla el formulario a través de cuatro estructuras condicionales. El primer condicional comprueba si el campo de edad está o no vacío, el segundo,

confirma, a través de una expresión regular, si el valor introducido es numérico, el tercero, constata si hay algún *checkbox* relacionado con el sexo marcado y el último, examina lo mismo, pero en el campo referido a si se presenta o no alguna deficiencia en la diferenciación de colores. El NIA no es obligatorio, por lo que, no se controla. Si alguna de las condiciones no se cumple, se informa al usuario del motivo por el que no ha podido finalizar el proceso y se le informa de qué hacer para completarlo correctamente.

Una vez que el formulario ha sido verificado y se ha rellenado correctamente, se llama al fichero *alta.php* para proceder al registro del usuario en la base de datos y se dirige al mismo a la página principal del test, en este caso: *index.html*.

Pasos de ejecución del fichero *alta.php*:

1. Se obtienen los datos del formulario:

```
$edad=$_GET['edad'];
$sexo=$_GET['sex'];
$nia=$_GET['nia'];
$enf=$_GET['def'];
```

2. Se establece la conexión con la base de datos:

```
$con=mysql_connect("localhost","ekaitz","ekaitz");
mysql_select_db("ekaitzBD",$con);
```

3. Se insertan los datos en la base de datos y se asigna un identificador al usuario que se va autoincrementando.

```
$sql="INSERT INTO usuarios (edad,sexo,enfermedad,nia,fecha) VALUES
('$edad','$sexo','$enf','$nia','$date')";

$_SESSION["idautoincrementado"]=mysql_insert_id($con);
```

4. Se dirige al usuario a *index.html*.

```
Header("Location: index.html");
```

5.3.3. Test

Ésta es la parte en la que se le muestran al usuario las imágenes. Cuenta con un fichero *index.html*, para el diseño de la parte del test, y con una página en JavaScript (*paleta.js*),

para todo el proceso de interacción con el usuario y los cambios de la apariencia de index.html.

Elemento div:

El lenguaje HTML dispone de un componente llamado *div* que sirve para definir secciones y elementos en la página. Mediante un identificador asignado al componente *div*, se pueden cambiar las propiedades de dichas secciones y los elementos de forma versátil y eficiente.

Las propiedades que se pueden modificar son de todo tipo: visibilidad, color, borde del elemento, posición en la página web, tamaño, etc.

La barra y la paleta de la página poseen cada una 20 elementos construidos con etiquetas *div*, a los que se accede a través de su correspondiente identificador y a los que se les pueden cambiar sus propiedades según convenga.

Por ejemplo, en la siguiente línea de código, se puede ver un elemento *div*, que tiene asignado un identificador *n15*. Esta línea pertenece a un elemento de la paleta de colores.

```
<div class="colelem" id="n15" onclick=elegirColor(this.id)><!-- simple frame --></div>
```

Función load():

Index.html funciona de la siguiente manera: al abrir la página en el navegador, se ejecuta la función *load()*, alojada en el fichero paleta.js, y ésta se ocupa de cargar la imagen correspondiente y su paleta de colores asociada, pero al hacerlo, dispone los colores aleatoriamente, evitando así que una imagen reproduzca siempre la misma paleta de colores y en el mismo orden.

Al iniciar la función *load()*, se ejecuta *tlms=fecha.getTime()*, que devuelve el tiempo transcurrido desde 1970. Esto se lleva a cabo en ese momento y, cuando el usuario pasa a la siguiente imagen, se ejecuta otra vez, obteniendo de este modo el tiempo que le ha costado al usuario analizar y elegir los colores de la imagen.

Después de lo anterior, se pasa al proceso de obtención de la siguiente imagen y se procede a cargar la misma en la página web.

Elección de la imagen a cargar:

Debido a que cada imagen se quiere analizar igual número de veces y a que, cada 40 imágenes, se quiere cambiar el orden de reproducción de las mismas, se crea una lista que marca qué imagen se ha evaluado y se establece una distribución uniforme aleatoria de forma que no pueda resultar previsible qué imagen se va a mostrar.

Esta lista se encuentra en la base de datos con el nombre *evaluaciones*. Cada fila posee cuatro elementos. Las cuádruplas son números aleatorios entre 1 y 40. De este modo, cada 10 filas, se generan 40 números, mediante una distribución aleatoria uniforme, por lo que, nunca se va a repetir un elemento. Cada imagen será mostrada igual número de veces y de forma aleatoria, con lo que, se garantiza que el usuario nunca podrá saber qué imagen va a ser mostrada.

Estos números generados de 1 a 40 serán apuntadores a un fichero en el que se encuentran los nombres de las imágenes.

10 primeras filas de la tabla evaluaciones:

idEvaluación	imagen1	imagen2	imagen3	imagen4
1	31	28	26	12
2	39	18	7	17
3	25	21	33	1
4	34	3	14	6
5	19	30	36	5
6	13	8	11	16
7	40	37	2	29
8	32	9	35	27
9	20	24	38	4
10	15	10	22	23

Según el identificador del usuario que realiza el test, se accede a una fila u a otra de la tabla *evaluaciones* y se muestran las cuatro imágenes correspondientes. El acceso a la fila de la tabla *evaluaciones* se realiza a través de una llamada AJAX asíncrona al fichero PHP “selectorDeImagenes.php” tras acceder a la cuádrupla correspondiente, se almacena en un *array* denominado *trozeado[4elementos]*. Posteriormente, se obtiene el fichero *colores.json*, también mediante una llamada AJAX asíncrona.

(El fichero JSON, que contiene los datos de todas las imágenes, se incluye en el apartado 2 del anexo, en el que también se muestra un pequeño programa Java, creado para generar el *colores.json*)

El fichero json almacena la siguiente información de cada imagen:

id: identificador de la imagen de 1 a 40.

url: registro de los nombres de las imágenes que dan acceso al fichero en el servidor.

rgbp: conjunto de los 20 colores que más área tienen en la imagen. En cada caso, se almacena la información acerca de su cantidad de rojo, verde y azul, mediante el intervalo unidad. El 1 equivale a decir que el color ocupa el 100% de la imagen y el 0, a que no ocupa nada.

r: proporción de rojo de 0 a 255.

g: proporción de verde de 0 a 255.

b: proporción de azul de 0 a 255.

p: intervalo unidad.

También con el fichero JSON y según la imagen que se carga, se accede al resto de la información y se crea la paleta de colores en la página web.

Bucle para la creación de la paleta de colores:

Mediante toda la información de la imagen extraída del fichero JSON, se ponen los colores de la misma en la paleta de forma aleatoria. El siguiente bucle es la forma en la que se realiza.

```
for(i=0; i<=19; i++){
```

```

var aleatorio=aleatorio1_20(19,0);
v[i]=aleatorio;
distribuirTabla(i,aleatorio,the_object);
}

```

La función *aleatorio1_20* (apartado 1.4.2. del anexo) genera números aleatorios del 0 al 19 sin repetición, esto significa que, a diferencia de los generadores al uso, los números generados no se pueden repetir.

La función *distribuirTabla(i,aleatorio,the_object)* (apartado 1.4.2. del anexo) recibe un número *i*, un número aleatorio y la estructura JSON. La variable *aleatorio* es el índice que apunta al color en la estructura JSON y el color que se va a poner en la paleta, *i* es la posición donde se va a colocar el color y *the_object* es la estructura JSON.

Proceso de selección de los colores:

Una vez cargada la imagen, el usuario elige los colores dando clic en los mismos. Al seleccionar un color, éste desaparece de la paleta y se sitúa en el primer hueco libre que encuentre en la barra, empezando a buscar por la parte izquierda. Si se vuelve a hacer clic en alguno de los colores situados en la barra, éste desaparecerá de la barra para volver a formar parte de la paleta.

El sistema funciona gracias a las siguientes estructuras de datos:

huecosBarra[posición en barra]=0

Sirve para averiguar en qué posición no hay un color asignado en la barra.

idBPos[identificadores div de barra]=posición

Asocia los identificadores de los *div* de la barra y su posición respectiva en la barra.

barra[0]=identificador div de barra

Es la inversa de la anterior, asocia posición con identificador *div* en la barra.

arrayDeIdsDePaleta[posición]=identificador div paleta

Asocia posición en la paleta con identificadores *div* de la paleta.

idBidP[id div de barra]=id div de paleta

Relaciona un identificador *div* de la barra asociado con un identificador *div* de la paleta, según la posición en que se encuentre el color en la barra.

V[indentificador div de paleta]=apuntador a color en el fichero JSON

Asocia identificadores *div* de la paleta con apuntadores a los colores en el fichero JSON.

quitadosEnPaleta[identificadores id de paleta]

Controla qué colores se han seleccionado en la paleta.

Todas las estructuras de datos anteriores facilitan la codificación y usan los *divs* que podemos ver en la captura de pantalla siguiente. Se cuenta con 20 identificadores únicos para los 20 colores de la paleta y con otros 20 identificadores para cada hueco de la barra donde se pueden situar los colores.



The screenshot shows a forum page with a color palette on the right and a navigation bar at the bottom. The palette contains 20 colored squares, each with a label 'n15' through 'n34'. The navigation bar has labels 'n40 n41 n42 ...' and 'n59'. A button labeled 'SIGUIENTE IMAGEN' is located at the bottom right of the palette area.

Elige los colores más importantes en el diseño, haz click en los colores para colocarlos en la barra, debes seleccionarlos de mayor a menor importancia

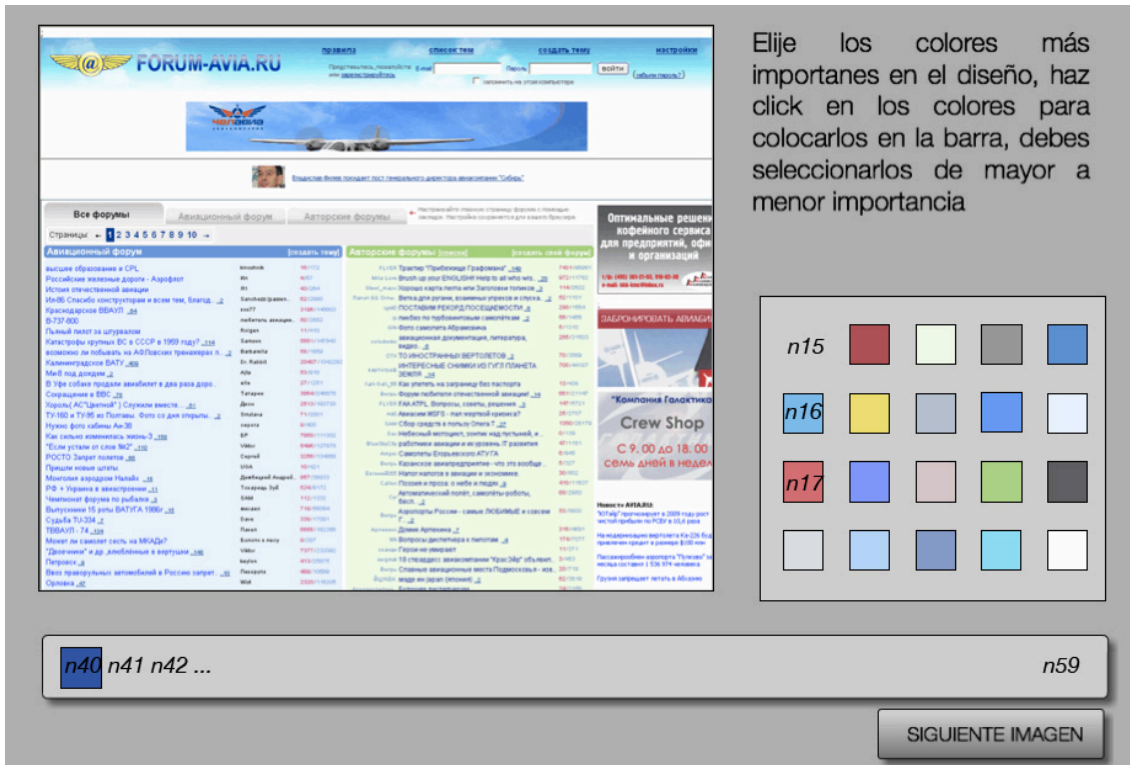
Para realizar esta parte de la página, se implementan dos funciones `elegirColor()` y `quitarColor()`. La primera se activa cuando el usuario hace clic en el color que está

situado en la paleta y la segunda, cuando lo haga en un color que esté situado en la barra.

Función elegirColor():

Tomando la captura anteriormente explicada como referente, es decir, partiendo de una paleta que dispone de todos los colores, si el usuario hace clic en el color azul asociado al *div* con identificador *n15*, se ejecutan las siguientes acciones:

1. Primero, se comprueba si ese hueco ha sido seleccionado anteriormente. Esto evita que se active la función si ya se ha dado clic en ese color, ya que sino el evento seguiría habilitado para ese hueco.
2. Se ejecuta la función *ndEsta()* que sirve para buscar el primer hueco disponible en la barra para situar el color.
3. A través de la posición de la barra, se accede al identificador *div* de la barra (*idBarra=barra[posición];*), que, en este caso concreto, será el *id n40*.
4. Como ya se tiene el identificador de la barra, se asocia el identificador de la paleta con el identificador de la barra (*idBidP[idBarra]=idPaleta*) para que, posteriormente, la función *quitarColor()* sepa a dónde volver en caso de que necesite retornar un color de la barra a su lugar original en la paleta. Los identificadores quedan asociados de la siguiente forma *idBidP[n40]=n15*.
5. Finalmente se quita el color de la paleta y se sitúa en la barra. Todo esto se realiza mediante los identificadores *div* y el DOM de JavaScript.



Elije los colores más importantes en el diseño, haz click en los colores para colocarlos en la barra, debes seleccionarlos de mayor a menor importancia

Función quitarColor():

Para ilustrar esta parte, se va a suponer que el usuario ha elegido un color como en la captura anterior y que el color está situado en el *div* con identificador *n40*.

Cuando el usuario hace clic en el color azul, sucede lo siguiente:

1. Se comprueba mediante el *array huecosBarra* que es verdad que efectivamente hay un color en el lugar seleccionado por el usuario (de este modo, no se ejecuta la función en caso de que no haya ningún color)
2. Se coloca el color en la paleta mediante la estructura `idBidP[idBarra]`, como previamente se había asociado el identificador de la barra con el identificador de la paleta, resulta muy sencillo hacerlo mediante una sentencia de DOM de JavaScript, que en este caso sería:

```
document.getElementById(idBidP[n40]).style.backgroundColor=document.getElementById(n40).style.backgroundColor;
```

3. Se quita el color de la barra:

```
document.getElementById(idBarra).style.borderColor = "transparent";
```

4. Se indica que se ha vuelto a poner el color en la paleta mediante quitadosEnPaleta[idBidP[idBarra]]=0. El 0 demuestra que el color está situado en la paleta.

5. Por último, se vuelven a cambiar las propiedades del *div* con identificador *n15*, transformando el transparente en color azul.

Elije los colores más importantes en el diseño, haz click en los colores para colocarlos en la barra, debes seleccionarlos de mayor a menor importancia

Función siguiente():

Una vez que el usuario esté satisfecho con los colores seleccionados, debe hacer clic en el botón que está situado en la parte inferior derecha de la imagen para que se ejecute la función siguiente() y se lleven a cabo las correspondientes acciones:

1. Primero, mediante el *array huecosBarra*, se controla que el usuario no ha dejado huecos libres en la barra, es decir, el usuario debe ubicar los colores sin dejar ningún hueco disponible entre ellos y empezando por la primera posición de la parte izquierda. De lo contrario, será avisado mediante un mensaje.
2. Si el usuario está analizando la primera, la segunda o la tercera imagen, será avisado de que tiene que esperar a que se cargue la imagen siguiente, pero si está en la cuarta, el aviso le anunciará que espere a que se cargue la página Feedback.

Espera a que la imagen se cargue...

Cargando el feedback, espere un momento...

3. Se inicializan todos los elementos de *arrayDeColores* a 20. Este es el *array* donde se almacenan los resultados de selección de un usuario para una determinada imagen.

```
for(i=0;i<=19;i++){  
    arrayColoresElegidos[i]=20;  
}
```

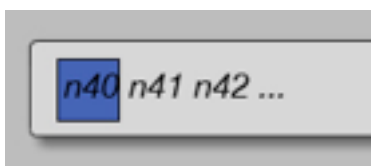
4. Se rellena un *array* con apuntadores al fichero *colores.json* que están en la barra.

```
for(i=0;i<=coloresElegidos-1;i++){  
  
    barraApuntador[i]=v[idBidP[barra[i].id]];  
  
}
```

En bucle anterior, se le proporcionan al *array v* los identificadores de la paleta que están asociados a la barra para obtener el color del fichero *colores.json*.

Ejemplo de funcionamiento del bucle:

Se tiene seleccionado un solo color. Se accede entonces a `barra[0].id`, que proporciona el identificador *div n40*. Posteriormente, se obtiene, mediante el identificador *n40*, su correspondiente identificador en la paleta a través de `idBidP[n40]`, que apunta a *n15*. Finalmente, se llega a `v[n15]`, que apunta al índice en JSON: un número entero entre 1 y 40 que corresponde a algún color en el fichero *colores.json*.



5. Se construye el *array arrayDeColoresElegidos*. Este *array* de 20 posiciones almacena, de menor a mayor, los colores elegidos por el usuario. El índice indica la posición en el fichero JSON y el contenido, en qué posición se elige el color.

Ejemplo de *arrayDeColoresElegidos*:

Posición relativa en el fichero json	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Apuntador al json	20	20	4	5	7	20	20	20	1	20	6	20	3	20	20	20	20	2	20	8

La tabla anteriormente expuesta indica que el color en posición 8, en el fichero JSON, ha sido elegido por el usuario en primera posición.

6. Se ejecuta por segunda vez la función *getTime()*, que ya había arrancado al iniciar la función *load()*. Después, se resta lo que han devuelto ambas funciones y se obtiene el tiempo que le ha costado al usuario seleccionar los colores de la imagen.


```

fecha2=new Date();
t2ms=fecha2.getTime();
tFin=t2ms-t1ms;

```

7. Se llama al fichero procesoBarra.php y se le pasaran los cuatro parámetros siguientes:

- arrayColoresElegidos.
- El tiempo que le ha costado al usuario analizar la imagen y seleccionar los colores (tFin).
- El identificador de la imagen que ha analizado (idImagen).
- numColoresEnBarra.

```

xmlhttp2.open("POST","procesoBarra.php?q="+arrayColoresElegidos+"&tiempo="+tFin+"&idImagen="+idImagen+"&numColorEnBarra="+numColoresEnBarra,true);

```

8. Ejecución de procesoBarra.php. Este fichero recibe los datos enviados desde paleta.js en la llamada del punto 7. Con estos datos, se actualiza la tabla Feedback de la bases de datos y se registra la selección del usuario en la tabla testImagen de la base de datos.

Las dos sentencias siguientes son las que registran los datos en testImagen y en Feedback respectivamente:

```

$sql="INSERT
INTOtestImagen(id,idImagen,numColoresElegidos,tiempo,color1,color2,color3,color4,color5,color6,color7,color8,color9,color10,color11,color12,color13,color14,color15,color16,color17,color18,color19,color20)VALUES('$id','$idImagen','$numColoresEnBarra','$tiempo','$cad[0]','$cad[1]','$cad[2]','$cad[3]','$cad[4]','$cad[5]','$cad[6]','$cad[7]','$cad[8]','$cad[9]','$cad[10]','$cad[11]','$cad[12]','$cad[13]','$cad[14]','$cad[15]','$cad[16]','$cad[17]','$cad[18]','$cad[19]');

```

```

$sql="UPDATE feedback SET color1='$cad[0]'+color1, color2='$cad[1]'+color2, color3='$cad[2]'+color3, color4='$cad[3]'+color4, color5='$cad[4]'+color5, color6='$cad[5]'+color6,color7='$cad[6]'+color7,color8='$cad[7]'+color8, color9='$cad[8]'+color9, color10='$cad[9]'+color10, color11='$cad[10]'+color11,color12='$cad[11]'+color12,color13='$cad[12]'+color13,color14='$cad[13]'+color14,color15='$cad[14]'+color15,color16='$cad[15]'+

```

```
color16,color17='$cad[16]'+color17,color18='$cad[17]'+color18,color19='$cad[18]'+color19,color20='$cad[19]'+color20 WHERE url='$idImagen'";
```

9. Una vez que se acaba de ejecutar el fichero proceso.php, el paleta.js recibe la respuesta a través de la sentencia `correcto=xmlhttp2.responseText;`.

En esa misma función, se usa un *if* para comprobar si el usuario se encuentra en la última imagen del test o en alguna de las tres primeras. Si está en la última imagen, el fichero se llamará feedback.php, si no, se llamará a la función *load()* e implicará la consecuente carga de la nueva imagen y de su respectiva paleta de colores.

```
if(contadorDeImágenes==4){
    window.location="feedback.php"
    contadorDeImágenes=0;
}
else{
    load();
}
```

5.3.4. Feedback.php

Como se explica en la descripción general de la aplicación, en el Feedback se le facilitan al usuario todas las imágenes y, junto a cada una de ellas, dos barras. La primera, muestra los colores que él ha elegido y la segunda, los colores promedio elegidos por los participantes anteriores.

En esta sección, sólo se emplea la tecnología PHP, ya que lo único que se necesita para crear la página HTML es obtener información del servidor, exactamente de la base de datos y del fichero JSON. No hay ninguna interacción con el usuario, así que se puede prescindir del uso de JavaScript en esta ocasión, a diferencia de las anteriores.

Aunque `feedback.php` es el PHP principal, existen más ficheros para modular la página. Se podría implementar todo en el fichero `feedback.php`, pero dificultaría la legibilidad y la capacidad de corregir errores. Desde la página `feedback.php`, se llama a los diferentes subprogramas PHP.

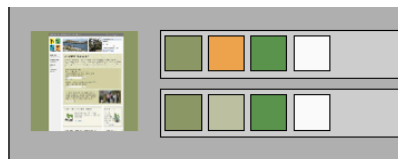
Los pasos para crear la página PHP son los siguientes:

1. Dentro del fichero *feedback*, en la primera línea de código, se llama a `iniciarFeedback.php` (ver apartado 1.5.2. del anexo). Aquí se conecta a la base de datos y se sacan los índices de las imágenes que acaba de testear el usuario.
2. Después, se realiza una llamada al fichero `json_decode.php` (ver apartado 1.5.3. del anexo).

El lector puede preguntarse por qué se usa `json_decode.php` para generar una librería propia si se dispone de una implementada en PHP por defecto. La razón es que el servidor usa una versión antigua de PHP y no tiene implementadas algunas funciones modernas. La versión exacta de PHP en el servidor es 4.1.22. y se necesita la versión 5.2.0 o alguna superior para usar `json_decode`, aunque, a día de hoy, ya está disponible la 5.3.0.

`Json_decode.php` es una función que realiza el análisis sintáctico del código JSON para facilitar el acceso a la estructura JSON a través de una sintaxis tipo orientado a objetos.

3. Posteriormente, se ejecuta todo el `feedback.php` y, según las partes que se quieran generar, se llama a los diferentes códigos PHP.



- Para la imagen testeada, se llama a cargarImagenFeedback.php.
- Para los colores seleccionados por el usuario, se llama a tratamientoFeedbackUsuarioFila.php.
- Para los colores promedio seleccionados por los usuarios, se llama a tratamientoFeedbackRestoUsuarios.php.

Las funciones anteriores se llaman cuatro veces, una vez por cada una de las imágenes que se tienen que mostrar.

4. cargarImagenFeedback.php (ver apartado 1.5.4. del anexo)
Aquí llamamos a *colores.json* y mostramos la imagen correspondiente mediante una *echo* de PHP.
5. tratamientoFeedbackUsuarioFila.php (ver apartado 1.5.5. del anexo)
Este proceso es igual que el apartado anterior, sólo que aquí se muestran los colores promedios seleccionados por los usuarios que han hecho el test hasta ese momento.
6. Finalmente, la parte inferior de la página dispone de un botón para que el usuario pueda volver al hacer test si lo desea. En caso de que así sea, simplemente con hacer clic, será redirigido a formulario.html mediante `onClick="location.href = 'formulario.html'"`.

5.4 Base de datos

La aplicación tiene implementada una Base de Datos mediante MySQL para almacenar la información relativa a usuarios, resultados de los análisis y características de las imágenes.

Se ha dividido la explicación de la base de datos en dos partes: el diseño lógico, es decir, el diseño conceptual de la base de datos, y el físico, que responde a cómo se ha implementado el diseño en el sistema gestor de base de datos MySQL.

Para acceder a la base de datos, se tiene que utilizar la URL: <http://colorlab.unavarra.es:8000/phpMyAdmin/>. Una vez que se cargue esta URL, se pedirá un nombre de usuario y una contraseña. Por cuestiones de seguridad, estos datos no se proporcionan aquí, pero, si en el futuro necesita hacer algún cambio, diríjase por medio del correo electrónico al servicio de sistemas de la Universidad Pública de Navarra a través de sistemas@unavarra.es. Ahí podrá resolver cualquier duda relativa al servidor MySQL.

Desde la dirección <http://colorlab.unavarra.es:8000/phpMyAdmin/>, se gestiona la base de datos mediante phpMyAdmin. Ésta es una interfaz web basada en PHP que facilita la interacción con el sistema MySQL.

5.4.1 Requisitos

La aplicación necesita una base de datos capaz de almacenar la información recabada sobre los usuarios (edad, sexo, enfermedad, número de identificación del alumno y fecha), los resultados de los test realizados y el promedio de esos mismos resultados.

5.4.2. Diseño lógico

Antes de realizar la implementación física de la base de datos, es necesario concebir el diseño lógico de la misma. Para ello, mediante el modelo entidad-relación, se han creado las siguientes entidades:

USUARIOS: almacena los datos relacionados con cada usuario que hace el test.

Campos:

- Id (*primary key*): identificador único por cada persona que hace el test; será de tipo entero.
- Edad: edad de cada sujeto que hace el test; será de tipo entero.
- Sexo: hombre o mujer; será de tipo *CHAR*.
- Enfermedad: sí, no o nose.
- NIA: número de identificación de alumno si el usuario es miembro de la Universidad Pública de Navarra.

- Fecha: hora, día, mes y año en la que el usuario se registra; será de tipo fecha.

TESTIMAGEN: almacena los datos obtenidos de las selecciones que hace cada usuario en cada imagen del test.

Campos:

- Id (*primary key*): identificador de registro asociado con los identificadores de la tabla usuarios.
- idImagen (*primary key*): identificador de la imagen; es el mismo que almacena las imágenes en el servidor.
- numColoresElegidos: número de colores seleccionados para una imagen por un usuario concreto.
- tiempo: tiempo que el usuario tarda en analizar la imagen.
- Color[1..20 registros]: posición seleccionada por el usuario para cada color; el 1, por ejemplo, indica que ese color ha sido seleccionado como el primero.

FEEDBACK: almacena el orden promedio de los colores elegidos para cada imagen por cada uno de los usuarios.

Campos:

- URL (*primary key*): nombre de la imagen testeada; es lo mismo que idImagen en testImagen.
- Color[1..20 registros]: es parecido al Color[1..20] de la tabla campos, sólo que aquí representa el orden promedio de los colores más importantes en cada imagen según la selección de todos los usuarios. El color que tenga el número menor será el que de media se haya seleccionado como más importante.

EVALUACIONES: almacena índices y, por cada índice, una cuádrupla de 4 elementos.

Campos:

- **idEvaluacion** (*primary key*): es un identificador que se asigna al usuario, el mismo que hay en testImagen y usuarios.
- **Imagen1**: almacena un entero entre 1 y 40 que será un apuntador al fichero *colores.json*. Este índice marca cuál será la primera imagen en testear el usuario con identificador igual a idEvaluación.
- **Imagen2**: almacena un entero entre 1 y 40 que será un apuntador al fichero *colores.json*. Este índice marca cuál será la segunda imagen a analizar por el usuario con identificador igual a idEvaluación.
- **Imagen3**: almacena un entero entre 1 y 40 que será un apuntador al fichero *colores.json*. Este índice marca cuál será la tercera imagen en estudiar el usuario con identificador igual a idEvaluación.
- **Imagen4**: almacena un entero entre 1 y 40 que será un apuntador al fichero *colores.json*. Este índice marca cuál será la cuarta imagen a examinar por el usuario con identificador igual a idEvaluación.

5.4.3. Diseño físico

Tabla de USUARIOS:

	Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/>	id	int(20)			No		auto_increment
<input type="checkbox"/>	edad	int(10)			No	0	
<input type="checkbox"/>	sexo	char(1)	latin1_swedish_ci		No		
<input type="checkbox"/>	enfermedad	varchar(4)	latin1_swedish_ci		No		
<input type="checkbox"/>	nia	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/>	fecha	datetime			No	0000-00-00 00:00:00	

Ejemplo de registros de USUARIOS:

id	edad	sexo	enfermedad	nia	fecha
32	21	m	no	71495	2012-01-10 16:19:47
33	45	m	no		2012-01-10 16:21:05
34	23	h	no	64680	2012-01-10 17:19:29
35	18	h	nose	71522	2012-01-10 18:58:10
36	18	h	no	71577	2012-01-10 19:33:02
37	18	h	no	mambrilla.71715	2012-01-10 19:35:06
38	18	h	no	71715	2012-01-10 19:35:45

La tabla anterior muestra el identificador de la persona que hace el test, su edad, sexo (“m” si es mujer o “h” si es hombre), si tiene alguna deficiencia en la distinción de colores, el NIA para los alumnos de la UPNA y, por último, la fecha y la hora en la que se ha dado de alta el usuario.

Tabla TESTIMAGEN:

	Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
<input type="checkbox"/>	<u>id</u>	int(20)			No	0
<input type="checkbox"/>	<u>idImagen</u>	varchar(50)	latin1_swedish_ci		No	
<input type="checkbox"/>	numColoresElegidos	int(2)			No	0
<input type="checkbox"/>	tiempo	int(10)			No	0
<input type="checkbox"/>	color1	int(2)			No	0
<input type="checkbox"/>	color2	int(2)			No	0
<input type="checkbox"/>	color3	int(2)			No	0
<input type="checkbox"/>	color4	int(2)			No	0
<input type="checkbox"/>	color5	int(2)			No	0
<input type="checkbox"/>	color6	int(2)			No	0
<input type="checkbox"/>	color7	int(2)			No	0
<input type="checkbox"/>	color8	int(2)			No	0
<input type="checkbox"/>	color9	int(2)			No	0
<input type="checkbox"/>	color10	int(2)			No	0
<input type="checkbox"/>	color11	int(2)			No	0
<input type="checkbox"/>	color12	int(2)			No	0
<input type="checkbox"/>	color13	int(2)			No	0
<input type="checkbox"/>	color14	int(2)			No	0
<input type="checkbox"/>	color15	int(2)			No	0
<input type="checkbox"/>	color16	int(2)			No	0
<input type="checkbox"/>	color17	int(2)			No	0
<input type="checkbox"/>	color18	int(2)			No	0
<input type="checkbox"/>	color19	int(2)			No	0
<input type="checkbox"/>	color20	int(2)			No	0

Ejemplo de registros para la tabla TESTIMAGEN:

id	idImagen	numColoresElegidos	tiempo	color1	color2	color3	color4	color5	color6	color7	color8	color9	color10	color11	color12	color13	color14	color15	color16	color17	color18	color19	color20
1	asandtech_com_11_small.GIF	4	22030	20	1	20	2	20	20	3	20	20	20	20	4	20	20	20	20	20	20	20	20
1	cyberpowerpc_com_11_small.GIF	4	24339	2	20	20	1	20	20	20	20	20	4	3	20	20	20	20	20	20	20	20	20
1	trevvy_com_11_small.GIF	4	26366	20	20	20	1	20	20	2	3	20	20	20	20	20	20	20	20	20	4	20	20
1	3839_com_11_small.GIF	4	16383	20	20	2	20	1	20	3	20	4	20	20	20	20	20	20	20	20	20	20	20
2	twcenter_net_11_small.GIF	7	85436	20	2	20	20	6	4	1	5	20	7	20	20	20	20	3	20	20	20	20	20
3	hangarnet_com_br_11_small.GIF	4	49508	2	20	20	4	20	20	20	1	20	20	20	20	20	20	20	20	20	20	3	20
2	devexpress_com_11_small.GIF	8	64250	20	4	3	20	1	2	20	8	5	7	20	20	20	20	20	20	20	6	20	20
3	rockettheme_com_11_small.GIF	4	26160	3	20	2	1	20	20	20	20	20	20	20	4	20	20	20	20	20	20	20	20
2	rmarsh_com_11_small.GIF	6	29533	1	20	2	20	20	3	20	20	4	20	20	20	20	20	20	20	5	20	20	6
3	line6_com_11_small.GIF	6	26055	20	6	20	3	20	20	20	20	2	1	20	20	20	20	20	4	5	20	20	20
3	rockwerchter_be_11_small.GIF	5	15296	20	5	3	1	4	20	20	20	2	20	20	20	20	20	20	20	20	20	20	20
2	baybul_com_11_small.GIF	7	38095	20	20	1	20	3	4	5	6	7	20	2	20	20	20	20	20	20	20	8	20

Tabla FEEDBACK:

	Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
<input type="checkbox"/>	<u>url</u>	varchar(50)	latin1_swedish_ci		No	
<input type="checkbox"/>	color1	int(2)			No	0
<input type="checkbox"/>	color2	int(2)			No	0
<input type="checkbox"/>	color3	int(2)			No	0
<input type="checkbox"/>	color4	int(2)			No	0
<input type="checkbox"/>	color5	int(2)			No	0
<input type="checkbox"/>	color6	int(2)			No	0
<input type="checkbox"/>	color7	int(2)			No	0
<input type="checkbox"/>	color8	int(2)			No	0
<input type="checkbox"/>	color9	int(2)			No	0
<input type="checkbox"/>	color10	int(2)			No	0
<input type="checkbox"/>	color11	int(2)			No	0
<input type="checkbox"/>	color12	int(2)			No	0
<input type="checkbox"/>	color13	int(2)			No	0
<input type="checkbox"/>	color14	int(2)			No	0
<input type="checkbox"/>	color15	int(2)			No	0
<input type="checkbox"/>	color16	int(2)			No	0
<input type="checkbox"/>	color17	int(2)			No	0
<input type="checkbox"/>	color18	int(2)			No	0
<input type="checkbox"/>	color19	int(2)			No	0
<input type="checkbox"/>	color20	int(2)			No	0

Ejemplo de registros FEEDBACK:

url	color1	color2	color3	color4	color5	color6	color7	color8	color9	color10	color11	color12	color13	color14	color15	color16	color17	color18	color19	color20
rockettheme_com_I1_small.GIF	203	290	248	43	187	300	324	364	278	378	313	354	205	363	359	377	371	373	354	351
axcess_co_za_I1_small.GIF	275	234	310	360	310	360	322	299	241	130	241	360	291	360	306	360	283	314	270	334
phoenix_de_I1_small.GIF	210	254	253	220	176	267	217	295	291	278	275	311	274	197	318	306	271	284	187	217
devexpress_com_I1_small.GIF	121	278	250	278	288	209	339	305	123	256	280	314	339	349	247	341	351	303	330	317
baybul_com_I1_small.GIF	291	284	190	292	168	68	314	307	358	193	349	372	282	380	357	378	365	374	334	355
skyglobe_ru_I1_small.GIF	213	270	182	202	206	264	273	197	242	252	189	297	204	299	261	282	298	248	267	300

En la tabla anterior, se ve que, para la imagen babybul_comI1_small.GIF, el color6 es el que ha sido valorado, en promedio, como más importante.

Tabla de EVALUACIONES:

	Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
<input type="checkbox"/>	<u>idEvaluacion</u>	int(20)			No	0
<input type="checkbox"/>	imagen1	int(11)			No	0
<input type="checkbox"/>	imagen2	int(11)			No	0
<input type="checkbox"/>	imagen3	int(11)			No	0
<input type="checkbox"/>	imagen4	int(11)			No	0

Ejemplo de registros de EVALUACIONES:

idEvaluacion	imagen1	imagen2	imagen3	imagen4
1	31	28	26	12
2	39	18	7	17
3	25	21	33	1
4	34	3	14	6
5	19	30	36	5
6	13	8	11	16

La tabla anterior muestra que el usuario al que se le asigna determinado identificador, por ejemplo el número 4, tiene que evaluar unas imágenes concretas, en caso del 4, la 34, la 3, la 14 y la 6, que corresponden a punteros a las imágenes del fichero *colores.json*.

6. Resultados

6.1. Número de colores importantes

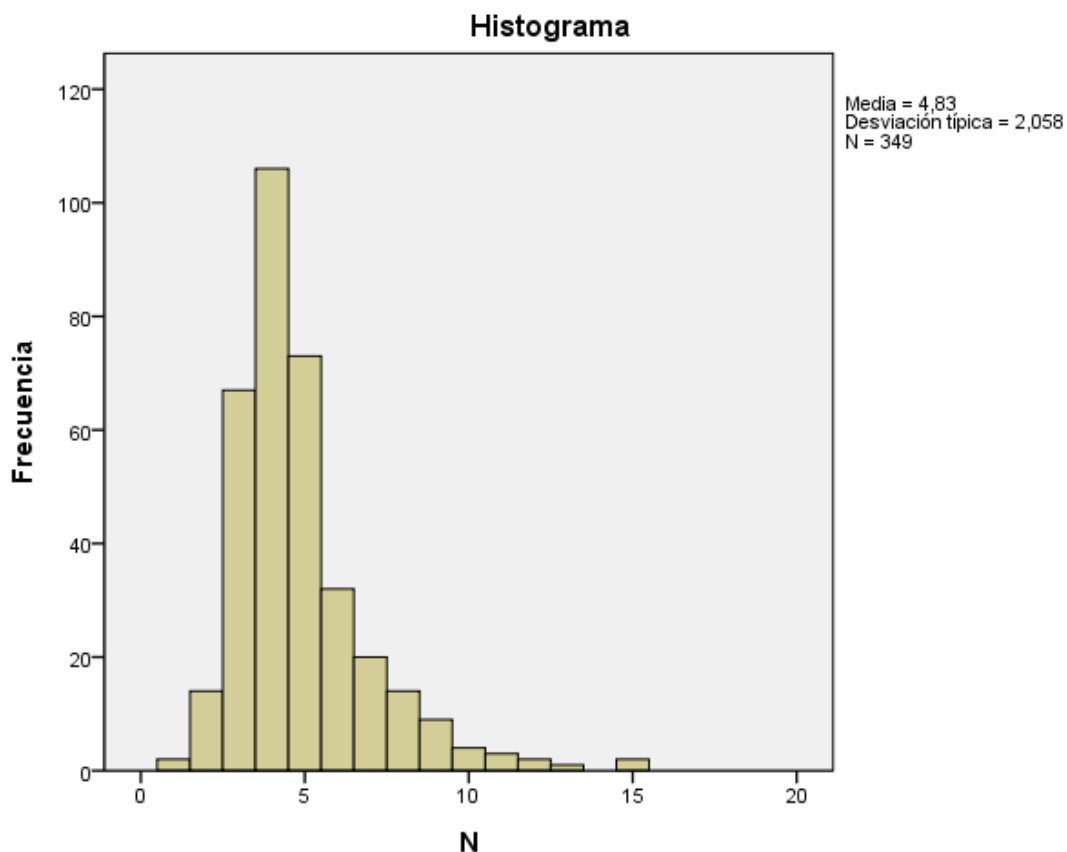
En primer lugar se ha analizado cuantos colores los usuarios consideran importantes en una página según los datos obtenidos en la prueba.

La muestra que se ha obtenido es de 349 personas, de los cuales 80 son mujeres y 269 hombres. Únicamente se ha aceptado en el estudio estadístico usuarios que hayan seleccionado un número de colores menor a 19, que hallan introducido edad, sexo y que hayan declarado que no tienen deficiencias en la visión de los colores.

En la siguiente tabla se muestran diferentes clase de estadísticos descriptivos sin separar el sexo en la muestra. Se ve que la media es 4,83 con un error típico de 0,110.

	N	Rango	Mínimo	Máximo	Media		Desv. típ.	Varianza
	Estadístico	Estadístico	Estadístico	Estadístico	Estadístico	Error típico	Estadístico	Estadístico
N	349	14	1	15	4,83	,110	2,058	4,235
N válido (según lista)	349							

A continuación se ve en formato histograma los resultados de la muestra, se observa que la mayoría ha considerado entre 3, 4 y 5 el número de colores importantes.



6.2. Número de colores importantes separados por sexo

Estadísticos descriptivos^a

	N	Rango	Mínimo	Máximo	Media		Desv. típ.	Varianza
	Estadístico	Estadístico	Estadístico	Estadístico	Estadístico	Error típico	Estadístico	Estadístico
N	80	14	1	15	5,04	,243	2,173	4,720
N válido (según lista)	80							

a. sexo = 1

Estadísticos descriptivos^a

	N	Rango	Mínimo	Máximo	Media		Desv. típ.	Varianza
	Estadístico	Estadístico	Estadístico	Estadístico	Estadístico	Error típico	Estadístico	Estadístico
N	269	14	1	15	4,77	,123	2,022	4,090
N válido (según lista)	269							

Estadísticos descriptivos^a

	N	Rango	Mínimo	Máximo	Media		Desv. típ.	Varianza
	Estadístico	Estadístico	Estadístico	Estadístico	Estadístico	Error típico	Estadístico	Estadístico
N	80	14	1	15	5,04	,243	2,173	4,720
N válido (según lista)	80							

a. sexo = 2

Con sexo=2 se refiere a los hombres y con sexo=1 a las mujeres.

Como se puede observar las mujeres han elegido ligeramente de media mas colores. Para comprobar que las diferencias entre ambos grupos no se debe simplemente al azar vamos a realizar la prueba de t de student.

La prueba de t de student asume igualdad de varianzas, por ello se va realizar la prueba de Levene para ver si existe homoestacidad.

Prueba de muestras independientes

	Prueba de Levene para la igualdad de varianzas		Prueba T para la igualdad de medias							
	F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Error típ. de la diferencia	95% Intervalo de confianza para la diferencia		
								Inferior	Superior	
N Se han asumido varianzas iguales	,102	,749	1,037	347	,301	,272	,262	-,244	,787	
No se han asumido varianzas iguales			,997	122,568	,321	,272	,272	-,268	,811	

En la tabla anterior se observa que la significación en la prueba de Levene , ésta hace que la hipótesis nula no se rechace, es decir, no existen diferencias significativas entre las varianzas, ya que para que sea rechazada la igualdad de varianzas, el nivel de

significación tendrá que ser inferior 0,05. Por ello se escoge la prueba de t de student para igualdad de varianzas.

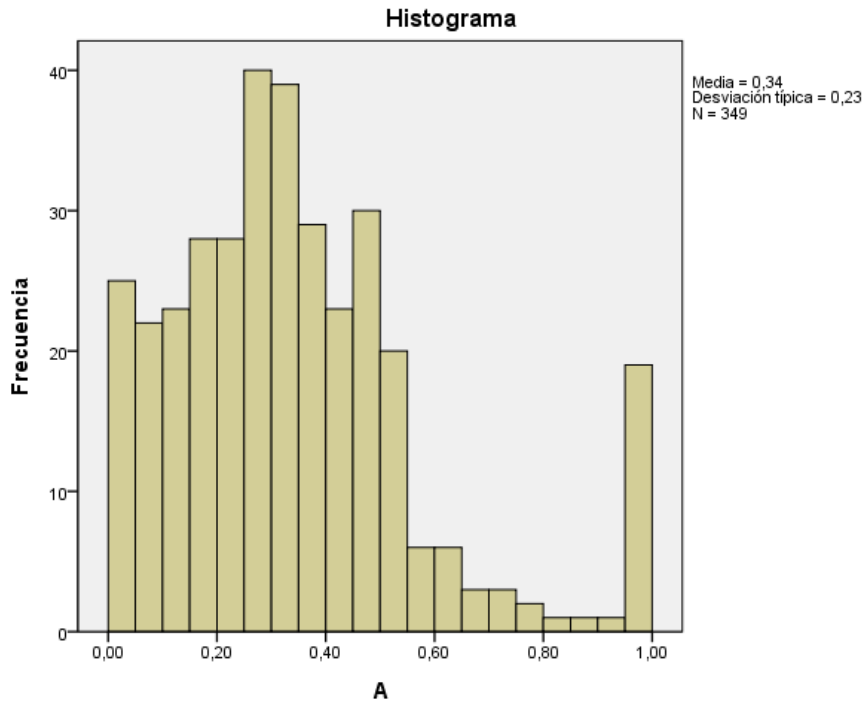
El valor t en la prueba de t de student sale 1,037 y da una significación bilateral de 0.301 (probabilidad de que la hipótesis nula sea falsa), debido a que la significación bilateral es mayor que 0.05 se acepta la hipótesis de que las medias son distintas debido al azar y no debido a que son poblaciones de estudio diferentes.

6.3. Área de la imagen ocupada por los colores elegidos

Estadísticos descriptivos								
	N	Rango	Mínimo	Máximo	Media		Desv. típ.	Varianza
	Estadístico	Estadístico	Estadístico	Estadístico	Estadístico	Error típico	Estadístico	Estadístico
A	349	,99	,00	,99	,3422	,01231	,22997	,053
N válido (según lista)	349							

Entre los 349 usuarios analizados la media de área ocupada por los colores ha sido de 0,3422 siendo 1 si la media hubiese ocupado toda la imagen y 0 si no hubiera elegido de media ningún color. El error típico es 0,01231.

En la siguiente figura se observa el histograma referente al área relativa escogida por los usuarios. En el eje de abscisas se muestra el número de usuarios que han escogido el área relativa.



A continuación se muestra la media de área relativa escogida separada por sexos.

Mujeres

	N	Rango	Mínimo	Máximo	Media		Desv. típ.	Varianza
	Estadístico	Estadístico	Estadístico	Estadístico	Estadístico	Error típico	Estadístico	Estadístico
A	80	,98	,00	,99	,3701	,02784	,24896	,062
N válido (según lista)	80							

a. sexo = 1

Hombres

	N	Rango	Mínimo	Máximo	Media		Desv. típ.	Varianza
	Estadístico	Estadístico	Estadístico	Estadístico	Estadístico	Error típico	Estadístico	Estadístico
A	269	,99	,00	,99	,3339	,01365	,22384	,050
N válido (según lista)	269							

a. sexo = 2

La media del área relativa de las mujeres es de 0,3701 con un error típico de 0,02784; los hombres tienen una media de 0,3339 y un error típico de 0,01365.

A partir de esta diferencia entre mujeres y hombres se plantea si esa diferencia es debido al azar o a poblaciones distintas. Se puede utilizar la prueba de t de Student para ver si esas diferencias de medias son relevantes. También se aplica previamente la prueba de homocedasticidad de Levene para saber si se puede aplicar la prueba de t de Student para igualdad de varianzas.

Estadísticos descriptivos^a

	N	Rango	Mínimo	Máximo	Media		Desv. típ.	Varianza
	Estadístico	Estadístico	Estadístico	Estadístico	Estadístico	Error típico	Estadístico	Estadístico
A	269	,99	,00	,99	,3339	,01365	,22384	,050
N válido (según lista)	269							

a. sexo = 2

Prueba de muestras independientes

	Prueba de Levene para la igualdad de varianzas		Prueba T para la igualdad de medias						
	F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Error típ. de la diferencia	95% Intervalo de confianza para la diferencia	
								Inferior	Superior
A Se han asumido varianzas iguales	,384	,536	1,238	347	,216	,03624	,02926	-,02132	,09379
No se han asumido varianzas iguales			1,169	119,512	,245	,03624	,03100	-,02515	,09762

En la tabla anterior hay una significación de 0,536, por lo tanto tomaremos la hipótesis nula, es decir, que las varianzas son iguales para luego realizar la prueba de t de student. La prueba de t student da un valor p de 0,216. Se ha supuesto un valor α de 0,05, al ser p mayor que α se acepta la hipótesis nula, es decir, la diferencia de medias es despreciable.

6.4. ¿Existen diferencias entre las imágenes seleccionadas respecto del número de colores elegidos?

Esta pregunta plantea si la cantidad de colores elegidos es diferente según la imagen expuesta en la prueba o si la imagen expuesta no influye en el número de colores elegidos.

A fin de responder a la anterior hipótesis en la siguiente tabla se muestra los datos recabados de las pruebas realizadas por los usuarios. Se tiene el número de valoraciones realizadas por cada imagen y la media de colores elegidos por cada imagen con su respectiva desviación típica.

Imagen	Numero valoraciones	Media	Desviación típica	Error típico	Intervalo de confianza para la media al 95%		Mínimo	Máximo
					Límite inferior	Límite superior		
1	10	5,70	1,947	,616	4,31	7,09	4	11
2	6	4,33	,816	,333	3,48	5,19	3	5
3	12	4,17	1,528	,441	3,20	5,14	2	8
4	5	3,60	,548	,245	2,92	4,28	3	4
5	8	4,38	1,996	,706	2,71	6,04	1	8
6	10	3,70	,675	,213	3,22	4,18	3	5
7	8	4,25	1,282	,453	3,18	5,32	3	7
8	9	4,78	2,438	,813	2,90	6,65	3	11
9	12	5,67	1,969	,569	4,42	6,92	3	9
10	8	4,00	,926	,327	3,23	4,77	3	5
11	10	4,90	,994	,314	4,19	5,61	3	6
12	11	6,00	2,366	,714	4,41	7,59	4	12
13	9	5,11	2,028	,676	3,55	6,67	3	8
14	11	5,55	2,252	,679	4,03	7,06	2	9
15	8	3,88	1,246	,441	2,83	4,92	2	6

16	4	5,25	2,630	1,315	1,07	9,43	3	9
17	12	4,25	1,357	,392	3,39	5,11	3	7
18	9	5,44	2,007	,669	3,90	6,99	3	10
19	6	4,17	1,941	,792	2,13	6,20	2	7
20	9	4,33	2,062	,687	2,75	5,92	2	8
21	13	4,69	1,653	,458	3,69	5,69	3	8
22	5	3,60	,894	,400	2,49	4,71	3	5
23	5	5,20	2,168	,970	2,51	7,89	4	9
24	9	4,56	2,297	,766	2,79	6,32	1	8
25	9	5,22	1,986	,662	3,70	6,75	3	9
26	12	5,08	2,678	,773	3,38	6,79	3	12
27	9	3,11	,782	,261	2,51	3,71	2	4
28	5	6,80	1,483	,663	4,96	8,64	5	9
29	8	4,88	1,246	,441	3,83	5,92	3	7
30	9	4,78	1,394	,465	3,71	5,85	4	8
31	11	4,36	1,629	,491	3,27	5,46	2	8
32	10	6,20	2,974	,940	4,07	8,33	4	13
33	7	5,29	2,498	,944	2,98	7,60	3	10
34	8	6,38	4,274	1,511	2,80	9,95	2	15
35	9	5,56	3,779	1,260	2,65	8,46	3	15
36	6	5,00	3,098	1,265	1,75	8,25	3	11
37	9	4,56	2,068	,689	2,97	6,15	2	9
38	8	4,50	1,309	,463	3,41	5,59	2	6
39	11	5,27	1,954	,589	3,96	6,59	2	8
40	9	4,00	1,118	,373	3,14	4,86	3	6

En la tabla anterior se ve que la media mínima es de 3,11 y la media máxima es de 6,80.

Para ver si estas diferencias entre la medias son significativas se ha aplicado **ANOVA** (*ANalysis Of Variance*). Debido a que hay mas de dos grupos a comparar no se ha aplicado la prueba de t de Student, en cambio ANOVA puede analizar los datos cuando hay mas de dos grupos.

Prueba de homogeneidad de varianzas

numColoresElegidos

Estadístico de			
Levene	gl1	gl2	Sig.
1,636	39	309	,013

El test Levene dice que las varianzas son distintas. ANOVA compara la variabilidad entre las medias de cada grupo y la variabilidad dentro de los grupos para poder concluir si esas diferencias entre las medias son significativas.

En la siguiente tabla se tienen los resultados de ANOVA y nos da una significación de 0,113, luego se puede aceptar que la diferencia de las medias no son significativas.

ANOVA

numColoresElegidos

	Suma de cuadrados	gl	Media cuadrática	F	Sig.
Inter-grupos	208,582	39	5,348	1,306	,113
Intra-grupos	1265,102	309	4,094		
Total	1473,685	348			

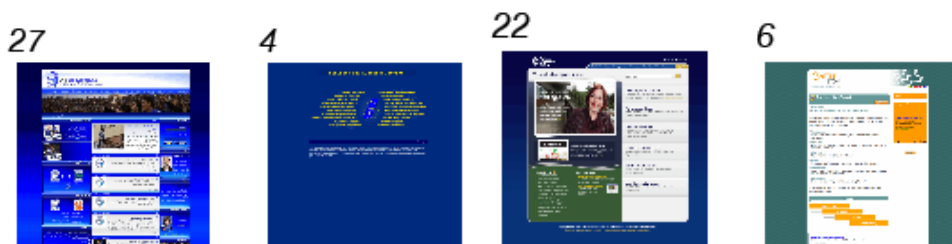
La siguiente tabla muestra el número de colores medios elegidos por imagen, la tabla esta ordenada por las medias en orden ascendente.

id_imagen	Numero valoraciones	Subconjunto para alfa = 0.05
		1
27	9	3,11
4	5	3,60
	5	3,60
6	10	3,70
15	8	3,88
10	8	4,00
40	9	4,00
3	12	4,17
19	6	4,17
7	8	4,25
17	12	4,25
2	6	4,33
20	9	4,33
31	11	4,36
5	8	4,38
38	8	4,50

24	9	4,56
37	9	4,56
21	13	4,69
8	9	4,78
30	9	4,78
29	8	4,88
11	10	4,90
36	6	5,00
26	12	5,08
13	9	5,11
23	5	5,20
25	9	5,22
16	4	5,25
39	11	5,27
33	7	5,29
18	9	5,44
14	11	5,55
35	9	5,56
9	12	5,67
1	10	5,70
12	11	6,00
32	10	6,20
34	8	6,38
28	5	6,80

Se observa en la tabla anterior que hay 7 imágenes que tienen una media de colores elegidos por debajo de 4,00 y 4 imágenes con una media superior a 6,00.

Imágenes por debajo de 4,00:



15



10



40



Imágenes por encima de 6,00:

12



32



34



28



Con estos resultados el análisis cualitativo que se puede llegar a inferir es que cuanto se usan muchos colores en una pagina web y que estos colores sean en general complementarios entre si y sus áreas relativas sean parecidas, los usuarios tienden a seleccionar mas colores en este tipo de imagen. Este tipo de diseño corresponde a las imágenes 12, 32 y 28.

6.5. ¿Existen diferencias entre las imágenes seleccionadas respecto de la fracción de área de la imagen ocupada por los colores seleccionados?

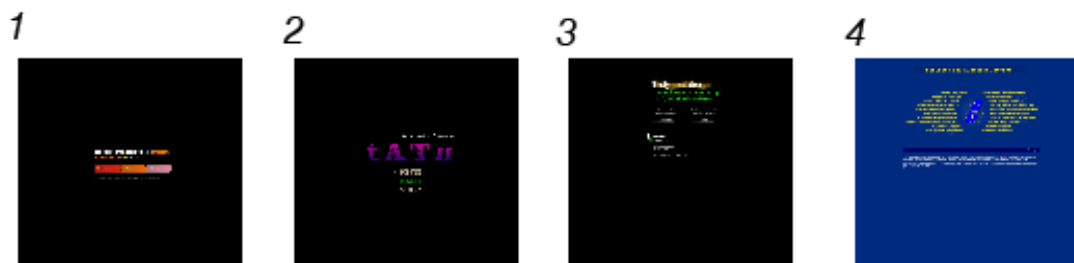
En la siguiente tabla se muestra la media de área que ocupan los colores seleccionados. La máxima media que se puede obtener es 1.

	Numero valoraci ones	Media	Desviación típica	Error típico	Intervalo de confianza para la media al 95%		Mínimo	Máximo
					Límite inferior	Límite superior		
					1	10		
2	6	,3308	,50203	,20495	-,1960	,8577	,00	,98
3	12	,6507	,47547	,13726	,3485	,9528	,00	,98
4	5	,5736	,51663	,23104	-,0679	1,2151	,01	,95
5	8	,2907	,32822	,11604	,0163	,5651	,00	,91
6	10	,4019	,25168	,07959	,2219	,5820	,08	,89
7	8	,2581	,22420	,07927	,0706	,4455	,04	,49
8	9	,3348	,23926	,07975	,1509	,5187	,05	,58
9	12	,3324	,23018	,06645	,1862	,4787	,09	,79
10	8	,3227	,19257	,06808	,1617	,4837	,06	,71
11	10	,2377	,17087	,05403	,1154	,3599	,03	,49
12	11	,2894	,17327	,05224	,1730	,4058	,07	,57
13	9	,2981	,08423	,02808	,2333	,3628	,21	,45
14	11	,4121	,16788	,05062	,2993	,5249	,13	,69
15	8	,3307	,12091	,04275	,2296	,4318	,19	,48
16	4	,3437	,10308	,05154	,1797	,5078	,23	,47
17	12	,3274	,14620	,04220	,2345	,4203	,15	,66
18	9	,4344	,16768	,05589	,3056	,5633	,09	,71
19	6	,3075	,11488	,04690	,1869	,4281	,12	,45
20	9	,2448	,14398	,04799	,1342	,3555	,03	,51
21	13	,4231	,12184	,03379	,3495	,4968	,25	,69
22	5	,3368	,04991	,02232	,2748	,3988	,25	,38
23	5	,1737	,10973	,04907	,0374	,3099	,04	,33
24	9	,3815	,22801	,07600	,2063	,5568	,01	,70
25	9	,2946	,08733	,02911	,2274	,3617	,15	,44
26	12	,2415	,14867	,04292	,1470	,3359	,06	,58
27	9	,2798	,10793	,03598	,1968	,3628	,16	,46
28	5	,3462	,06567	,02937	,2647	,4277	,26	,42
29	8	,3603	,16439	,05812	,2229	,4978	,15	,58
30	9	,3836	,06883	,02294	,3307	,4365	,32	,50
31	11	,2642	,11656	,03515	,1859	,3425	,10	,46
32	10	,3635	,19637	,06210	,2230	,5040	,14	,75
33	7	,3112	,13450	,05084	,1868	,4356	,13	,55
34	8	,3246	,26175	,09254	,1058	,5435	,09	,81
35	9	,3084	,19546	,06515	,1581	,4586	,10	,65
36	6	,3626	,11507	,04698	,2419	,4834	,24	,56
37	9	,2489	,13338	,04446	,1464	,3514	,09	,45
38	8	,2972	,10225	,03615	,2117	,3827	,16	,44
39	11	,3168	,12304	,03710	,2341	,3994	,15	,52
40	9	,2359	,15943	,05314	,1133	,3584	,09	,50

Ahora se aplica Levene para ver si se observa diferencia entre las varianzas. Nos da que no son iguales ya que la significación es 0,000.

Para inferir si se puede considerar que hay medias diferentes se ha aplicado ANOVA, el resultado es que hay medias diferentes. Debido a que ANOVA no nos proporciona que medias son diferentes, es decir, no sabemos si hay 1 media diferente o 10. Se ha aplicado un análisis post hoc o a posteriori. Para ello se ha aplicado la prueba de Tukey-b, se observa mediante esta prueba que solo las imágenes 23 y 3 se pueden considerar estadísticamente diferentes. Debido a que tenemos 40 imágenes podemos concluir que las imágenes según los usuarios son iguales.

Haciendo un análisis cualitativo se observa que las imágenes con mas área ocupada por los colores seleccionados son claramente diferentes a las demás imágenes en este caso son las imágenes 1, 2, 3 y 4. Son imágenes que la mayoría del área esta formada por un color determinado y se han utilizado muy pocos colores para el diseño de la página.



Respecto a las imágenes con menos área ocupada, no se ha podido observar un diferencia clara respecto a las demás.

6.6. ¿Se escogen preferentemente aquellos colores que representan un porcentaje mayor del área?

En la tabla de la siguiente página los colores están ordenados por área ocupada, la tabla expresa si los colores que mas área ocupan han sido los que mas han sido elegidos.

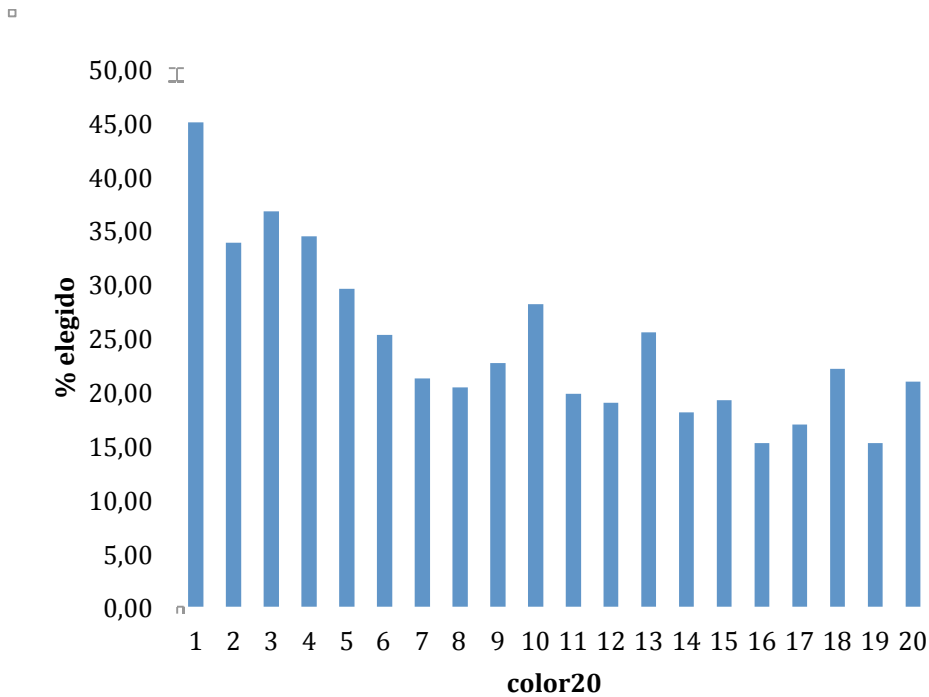
Por ejemplo, cuando el color 1 ha sido seleccionado 45,0 %, esto representa que el color con mas área en cada imagen ha sido seleccionado como el mas importante un 45,0% de las veces.

Hay una tendencia a seleccionar los colores que mas área ocupan en la imagen, no siendo, sin embargo, una diferencia muy notable. A partir de la posición 6 la tendencia no es muy descendente en el número de veces que se ha seleccionado los colores en esas posiciones y el porcentaje elegido se mantiene. Podemos decir que los colores con menos área influyen bastante más si lo comparamos con su área ocupada en la imagen.

También hay que decir que esta tabla puede llevar a engaño, por ejemplo, el color 1 de una imagen puede tener menos área que el color 2 de otra imagen.

Color	Veces elegido	Veces no elegido	Total	% elegido
1	157	192	349	45,0
2	118	231	349	33,8
3	128	221	349	36,7
4	120	229	349	34,4
5	103	246	349	29,5
6	88	261	349	25,2
7	74	275	349	21,2
8	71	278	349	20,3
9	79	270	349	22,6
10	98	251	349	28,1
11	69	280	349	19,8
12	66	283	349	18,9
13	89	260	349	25,5
14	63	286	349	18,1
15	67	282	349	19,2
16	53	296	349	15,2
17	59	290	349	16,9
18	77	272	349	22,1
19	53	296	349	15,2
20	73	276	349	20,9

Representación gráfica de la tabla anterior:



Podemos contar, para cada color, cuántas veces ha sido elegido en cada uno de los puestos. Estas 20 tablas son las siguientes:

		color1			Porcentaje acumulado
		Frecuencia	Porcentaje	Porcentaje válido	
Válidos	1	46	13,2	29,3	29,3
	2	32	9,2	20,4	49,7
	3	28	8,0	17,8	67,5
	4	22	6,3	14,0	81,5
	5	13	3,7	8,3	89,8
	6	5	1,4	3,2	93,0
	7	4	1,1	2,5	95,5
	8	4	1,1	2,5	98,1
	10	2	,6	1,3	99,4
	15	1	,3	,6	100,0
	Total	157	45,0	100,0	
Perdidos	Sistema	192	55,0		
Total		349	100,0		

color2

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	30	8,6	25,4	25,4
	2	27	7,7	22,9	48,3
	3	28	8,0	23,7	72,0
	4	14	4,0	11,9	83,9
	5	11	3,2	9,3	93,2
	6	6	1,7	5,1	98,3
	8	1	,3	,8	99,2
	14	1	,3	,8	100,0
	Total	118	33,8	100,0	
	Perdidos	Sistema	231	66,2	
Total		349	100,0		

color3

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	35	10,0	27,3	27,3
	2	25	7,2	19,5	46,9
	3	31	8,9	24,2	71,1
	4	8	2,3	6,3	77,3
	5	11	3,2	8,6	85,9
	6	8	2,3	6,3	92,2
	7	5	1,4	3,9	96,1
	8	2	,6	1,6	97,7
	9	1	,3	,8	98,4
	11	1	,3	,8	99,2
	14	1	,3	,8	100,0
	Total	128	36,7	100,0	
	Perdidos	Sistema	221	63,3	
Total		349	100,0		

color4

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	34	9,7	28,3	28,3
	2	29	8,3	24,2	52,5
	3	27	7,7	22,5	75,0
	4	11	3,2	9,2	84,2
	5	7	2,0	5,8	90,0
	6	6	1,7	5,0	95,0
	7	2	,6	1,7	96,7
	8	2	,6	1,7	98,3
	9	1	,3	,8	99,2
	13	1	,3	,8	100,0
	Total	120	34,4	100,0	
Perdidos	Sistema	229	65,6		
Total		349	100,0		

color5

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	15	4,3	14,6	14,6
	2	20	5,7	19,4	34,0
	3	29	8,3	28,2	62,1
	4	22	6,3	21,4	83,5
	5	11	3,2	10,7	94,2
	6	2	,6	1,9	96,1
	7	1	,3	1,0	97,1
	9	2	,6	1,9	99,0
	10	1	,3	1,0	100,0
		Total	103	29,5	100,0
Perdidos	Sistema	246	70,5		
Total		349	100,0		

color6

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado	
Válidos	1	16	4,6	18,2	18,2	
	2	17	4,9	19,3	37,5	
	3	25	7,2	28,4	65,9	
	4	13	3,7	14,8	80,7	
	5	5	1,4	5,7	86,4	
	6	7	2,0	8,0	94,3	
	7	3	,9	3,4	97,7	
	10	1	,3	1,1	98,9	
	12	1	,3	1,1	100,0	
	Total		88	25,2	100,0	
	Perdidos	Sistema	261	74,8		
Total		349	100,0			

color7

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	21	6,0	28,4	28,4
	2	16	4,6	21,6	50,0
	3	14	4,0	18,9	68,9
	4	9	2,6	12,2	81,1
	5	6	1,7	8,1	89,2
	6	3	,9	4,1	93,2
	7	2	,6	2,7	95,9
	8	2	,6	2,7	98,6
	9	1	,3	1,4	100,0
	Total		74	21,2	100,0
Perdidos	Sistema	275	78,8		
Total		349	100,0		

color8

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	21	6,0	29,6	29,6
	2	15	4,3	21,1	50,7
	3	7	2,0	9,9	60,6
	4	12	3,4	16,9	77,5
	5	9	2,6	12,7	90,1
	6	3	,9	4,2	94,4
	7	1	,3	1,4	95,8
	8	2	,6	2,8	98,6
	12	1	,3	1,4	100,0
	Total	71	20,3	100,0	
Perdidos	Sistema	278	79,7		
Total		349	100,0		

color9

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	10	2,9	12,7	12,7
	2	22	6,3	27,8	40,5
	3	16	4,6	20,3	60,8
	4	17	4,9	21,5	82,3
	5	4	1,1	5,1	87,3
	6	6	1,7	7,6	94,9
	7	1	,3	1,3	96,2
	8	1	,3	1,3	97,5
	9	2	,6	2,5	100,0
	Total	79	22,6	100,0	
Perdidos	Sistema	270	77,4		
Total		349	100,0		

color10

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	24	6,9	24,5	24,5
	2	23	6,6	23,5	48,0
	3	15	4,3	15,3	63,3
	4	12	3,4	12,2	75,5
	5	10	2,9	10,2	85,7
	6	6	1,7	6,1	91,8
	7	2	,6	2,0	93,9
	8	1	,3	1,0	94,9
	9	1	,3	1,0	95,9
	10	1	,3	1,0	96,9
	11	1	,3	1,0	98,0
	12	1	,3	1,0	99,0
	13	1	,3	1,0	100,0
	Total	98	28,1	100,0	
Perdidos	Sistema	251	71,9		
Total		349	100,0		

color11

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	18	5,2	26,1	26,1
	2	13	3,7	18,8	44,9
	3	8	2,3	11,6	56,5
	4	5	1,4	7,2	63,8
	5	8	2,3	11,6	75,4
	6	6	1,7	8,7	84,1
	7	4	1,1	5,8	89,9
	8	1	,3	1,4	91,3
	9	1	,3	1,4	92,8
	10	2	,6	2,9	95,7
	11	3	,9	4,3	100,0
	Total	69	19,8	100,0	
Perdidos	Sistema	280	80,2		
Total		349	100,0		

color12

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	4	1,1	6,1	6,1
	2	10	2,9	15,2	21,2
	3	22	6,3	33,3	54,5
	4	17	4,9	25,8	80,3
	5	4	1,1	6,1	86,4
	6	4	1,1	6,1	92,4
	7	2	,6	3,0	95,5
	8	1	,3	1,5	97,0
	11	1	,3	1,5	98,5
	12	1	,3	1,5	100,0
	Total	66	18,9	100,0	
Perdidos	Sistema	283	81,1		
Total		349	100,0		

color13

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	9	2,6	10,1	10,1
	2	24	6,9	27,0	37,1
	3	12	3,4	13,5	50,6
	4	18	5,2	20,2	70,8
	5	14	4,0	15,7	86,5
	6	4	1,1	4,5	91,0
	7	3	,9	3,4	94,4
	8	2	,6	2,2	96,6
	9	2	,6	2,2	98,9
	13	1	,3	1,1	100,0
	Total	89	25,5	100,0	
Perdidos	Sistema	260	74,5		
Total		349	100,0		

color14

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	13	3,7	20,6	20,6
	2	16	4,6	25,4	46,0
	3	10	2,9	15,9	61,9
	4	14	4,0	22,2	84,1
	5	4	1,1	6,3	90,5
	6	2	,6	3,2	93,7
	7	2	,6	3,2	96,8
	8	2	,6	3,2	100,0
	Total	63	18,1	100,0	
Perdidos	Sistema	286	81,9		
Total		349	100,0		

color15

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	5	1,4	7,5	7,5
	2	11	3,2	16,4	23,9
	3	8	2,3	11,9	35,8
	4	13	3,7	19,4	55,2
	5	13	3,7	19,4	74,6
	6	6	1,7	9,0	83,6
	7	7	2,0	10,4	94,0
	8	2	,6	3,0	97,0
	9	1	,3	1,5	98,5
	11	1	,3	1,5	100,0
	Total	67	19,2	100,0	
Perdidos	Sistema	282	80,8		
Total		349	100,0		

color16

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	14	4,0	26,4	26,4
	2	7	2,0	13,2	39,6
	3	9	2,6	17,0	56,6
	4	7	2,0	13,2	69,8
	5	4	1,1	7,5	77,4
	6	4	1,1	7,5	84,9
	7	4	1,1	7,5	92,5
	8	1	,3	1,9	94,3
	9	1	,3	1,9	96,2
	10	1	,3	1,9	98,1
	11	1	,3	1,9	100,0
	Total	53	15,2	100,0	
Perdidos	Sistema	296	84,8		
Total		349	100,0		

color17

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	10	2,9	16,9	16,9
	2	14	4,0	23,7	40,7
	3	8	2,3	13,6	54,2
	4	9	2,6	15,3	69,5
	5	6	1,7	10,2	79,7
	6	3	,9	5,1	84,7
	7	3	,9	5,1	89,8
	8	1	,3	1,7	91,5
	10	3	,9	5,1	96,6
	11	1	,3	1,7	98,3
	12	1	,3	1,7	100,0
	Total	59	16,9	100,0	
Perdidos	Sistema	290	83,1		
Total		349	100,0		

color18

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	12	3,4	15,6	15,6
	2	7	2,0	9,1	24,7
	3	10	2,9	13,0	37,7
	4	21	6,0	27,3	64,9
	5	8	2,3	10,4	75,3
	6	5	1,4	6,5	81,8
	7	5	1,4	6,5	88,3
	8	3	,9	3,9	92,2
	9	3	,9	3,9	96,1
	10	2	,6	2,6	98,7
	11	1	,3	1,3	100,0
	Total	77	22,1	100,0	
Perdidos	Sistema	272	77,9		
Total		349	100,0		

color19

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	6	1,7	11,3	11,3
	2	6	1,7	11,3	22,6
	3	5	1,4	9,4	32,1
	4	12	3,4	22,6	54,7
	5	8	2,3	15,1	69,8
	6	4	1,1	7,5	77,4
	7	5	1,4	9,4	86,8
	8	2	,6	3,8	90,6
	9	2	,6	3,8	94,3
	10	2	,6	3,8	98,1
	12	1	,3	1,9	100,0
		Total	53	15,2	100,0
Perdidos	Sistema	296	84,8		
Total		349	100,0		

color20

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	1	4	1,1	5,5	5,5
	2	11	3,2	15,1	20,5
	3	20	5,7	27,4	47,9
	4	12	3,4	16,4	64,4
	5	7	2,0	9,6	74,0
	6	3	,9	4,1	78,1
	7	3	,9	4,1	82,2
	8	6	1,7	8,2	90,4
	9	5	1,4	6,8	97,3
	10	1	,3	1,4	98,6
	15	1	,3	1,4	100,0
	Total	73	20,9	100,0	
Perdidos	Sistema	276	79,1		
	Total	349	100,0		

6.7. Conclusión de los resultados

Según los análisis anteriormente realizados, no se puede llegar a conclusiones categóricas. Se puede ver que hay una tendencia a elegir colores que ocupan una área mayor en la imagen, pero también se observa que colores que no ocupan una gran área en la imagen tienen mucha influencia en el usuario de lo que podría pensarse.

En el apartado 5.4. se ha podido ver que las imágenes con un mayor número de colores seleccionados son imágenes que tienen muchos colores, estos colores son muy diferentes entre sí y sus áreas relativas son parecidas, pero hay que dejar claro que según el análisis de ANOVA nos da que esas diferencias de número de colores elegidos no son significativas.

En el apartado 5.5. se puede observar que las cuatro imágenes con mayor fracción de área de la imagen ocupada por los colores seleccionados, son imágenes que tienen

pocos colores y hay un color claramente predominante respecto a los demás y ese color es el fondo del diseño de la pagina web.

Hay que señalar que ha podido haber errores de percepción del usuario. Hay imágenes que tienen muchos tonos de un color, cuando el usuario selecciona un color en el paleta pueden suceder errores de elección, pudiendo elegir colores en la paleta que verdaderamente no corresponden a los colores que el usuario pretendía elegir.

7. Conclusiones

Como conclusión general y final de este proyecto puede afirmarse que se han alcanzado los distintos objetivos establecidos en su comienzo.

Se ha estudiado, mediante un experimento psicofísico, el número de colores que los usuarios perciben cuando juzgan el diseño de una página web. Para ello se han alcanzado varios objetivos parciales, tal y como se estableció en el plan de trabajo:

- Se han seleccionado 40 páginas web con diseños gráficos variados, procurando elegir páginas con un número distinto de colores importantes en su diseño.
- Se ha elaborado un programa automatizado para la realización del experimento psicofísico basado en una aplicación web accesible mediante internet. La aplicación gestiona todos los aspectos del experimento y construye las bases de datos necesarias para el posterior análisis de los mismos.
- Se ha obtenido un número significativo de valoraciones de distintos usuarios. El número de participantes, 349, ha sido lo suficientemente elevado para permitir un análisis diferenciado según el sexo de los mismos.
- Se han analizado estadísticamente las respuestas de los usuarios y se han obtenido algunas conclusiones de carácter general. Aunque se han obtenido algunas tendencias generales, tal y como se explica en el apartado anterior, hay que decir que no han podido identificarse patrones o preferencias claras en los usuarios. Esto puede ser debido simplemente a que no existe un claro patrón de preferencia asociado al número de colores importantes en el diseño de una página web o bien a que el experimento o los estímulos seleccionados no han permitido un muestreo efectivo de dicha preferencia.

8. Bibliografía

How to Use Color to Successfully Brand Your Website

<http://uxmovement.com/content/how-to-use-color-to-brand-your-website/>

Color Theory, The Color Wheel And Color Schemes | Van SEO Design

<http://www.vanseodesign.com/web-design/color-theory/>

Color blindness - Wikipedia, the free encyclopedia

http://en.wikipedia.org/wiki/Color_blindness

Color Harmonies: complementary, analogous, triadic color schemes:

<http://www.tigercolor.com/color-lab/color-theory/color-harmonies.htm>

Dominance: Creating Focal Points In Your Design | Van SEO Design:

<http://www.vanseodesign.com/web-design/dominance/>

Whitespace: Less Is More In Web Design | Van SEO Design

<http://www.vanseodesign.com/web-design/whitespace/>

W3Schools Online Web Tutorials

<http://www.w3schools.com/>

PHP: Manual de PHP

<http://php.net/manual/es/index.php>

Foros del Web, comunidad para aprender web

<http://www.forosdelweb.com/>

Stack Overflow

<http://stackoverflow.com/>

t de student

<http://biplot.usal.es/DOCTORADO/3CICLO/BIENIO-06-08/tStudent.pdf>

ANOVA de un factor

http://www.uclm.es/profesorado/mdsalvador/58109/teoria/anova_un_factor-lectura.pdf

The reability of a sample mean

<http://www.ppsw.rug.nl/~bve/stat/module5/1betrouwbaarheid.html>

Análisis de la varianza

<http://www.upcomillas.es/personal/peter/analisisdevarianza/ANOVAIntroduccion.pdf>

Error típico de la media

<http://www.upcomillas.es/personal/peter/estadisticabasica/ErrorTipico.pdf>

Anexo

1. Código

En este anexo, se han recopilado los principales códigos utilizados para la página. Algunos ficheros, como el de *colores.json* o las dos hojas *css*, no se han incluido enteros para no alargar demasiado esta memoria. De todos modos, si se quiere, se puede consultar el código íntegro en el CD adjunto.

1.2. Instrucciones

1.2.1 instrucciones.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<metahttp-equiv="Content-Type"content="text/html; charset=UTF-8">
<!-- TemplateBeginEditable name="doctitle" -->
<title>Instrucciones de la pagina</title>
<!-- TemplateEndEditable -->
<!-- TemplateBeginEditable name="head" -->
<!-- TemplateEndEditable -->
<styletype="text/css">
-->
</style></head>
<body>
<divclass="container">
  <divclass="content">
    <p><u>Instrucciones.</u></p>
    <p>En primer lugar, gracias por participar en este experimento.</p>
    <p>El experimento es anónimo y su duración aproximada es de menos de 10 min. Al final del experimento le proporcionaremos información sobre sus resultados, comparados con los resultados de los anteriores participantes.</p>
    <p>La realización del experimento es sencilla, no obstante, para su correcta ejecución, le rogamos que lea atentamente las siguientes explicaciones e instrucciones.</p>
    <p>Antes de todo, le pedimos encarecidamente que sino tiene actualizado el navegador a la ultima versión, la descargue, ya que pueden surgir problemas en la pagina web.</p>
    <p>De click al navegador que usted usa y descargue e instale la ultima versión: </p>
    <p><a href="http://windows.microsoft.com/es-ES/internet-explorer/downloads/ie-8"><imgsrc="image/navegadores/explorer.png"alt="explorer"height="50"width="50"/></a>
    <a href="http://www.mozilla.org/es-ES/firefox/fx"><imgsrc="image/navegadores/firefox.png"alt="firefox"height="50"width="50"/></a>
    <a href="http://www.apple.com/es/safari/download/"><imgsrc="image/navegadores/safari.gif"alt="safari"height="50"width="50"/></a>
    <a href="https://www.google.com/chrome?platform=win&hl=es"><imgsrc="image/navegadores/chrome.jpg"alt="chrome"height="50"width="50"/></a>
    </p>
    <p><u>Objetivo del experimento:</u></p>
    <p>Observe la siguiente imagen. Es la página principal del sitio web de la Universidad Pública de Navarra. Como puede observar, en su diseño se utilizan diferentes colores, en diferentes elementos y en distinta proporción. En principio, el número de colores diferentes puede ser muy grande. Sin embargo, ¿Cuántos colores cree usted que son importantes en este diseño? Obviamente la respuesta a esta pregunta depende tanto
```


del diseño propiamente dicho como de la persona que lo observa. </p>

<u><imgsrc="image/instrucciones/ejemplo.gif"alt=""width="248"height="192"></u></p>

<p>En su opinión ¿qué colores son importantes en este diseño?, ¿Puede identificarlos y asignarles un orden de importancia?</p>

<p>El objetivo de este experimento es precisamente determinar qué colores son importantes y en segundo lugar, ordenar dichos colores por su importancia.

Para ello hemos desarrollado una aplicación que le ayudará en este objetivo, facilitándole un procedimiento sencillo para realizar la selección ordenada de colores según su importancia.</p>

<u><p>Desarrollo del experimento.</u></p>

<p>Tras pedirle algunos datos estadísticos (sexo, edad, etc) comenzaremos el experimento.</p>

<p>Cuando comience el experimento, el programa le mostrará una pantalla similar a ésta:

<u><imgsrc="image/instrucciones/vacio.png"alt=""width="461"height="342"align="baseline"></u></p>

<p>En ella se muestra a la izquierda la imagen de una página web. A la derecha se muestra una paleta formada por 20 colores. En la parte inferior hay una barra horizontal donde usted debe colocar los colores según su propio criterio.</p>

<p>Para colocar un color en la barra de resultado, simplemente pulse con el ratón sobre uno de los colores de la paleta (por ejemplo sobre el cuadrado rojo, como se muestra en el ejemplo). Verá que el color desaparece de la paleta y aparece en la barra de resultados.</p>

<u><imgsrc="image/instrucciones/1color.png"alt=""width="461"height="337"align="baseline"></u></p>

<p>Para elegir un segundo color, pulse con el ratón sobre él en la paleta. El color se colocará a continuación del anterior en la barra de resultados. Puede repetir el proceso tantas veces quiera, traspasando colores de la paleta a la barra de resultados.

<u><imgsrc="image/instrucciones/2colores.png"alt=""width="473"height="343"></u></p>

<p><brclear="all">

Si quiere rectificar una elección, para deshacer la selección, pulse sobre el color en la barra de resultados y el color volverá a la paleta, al mismo lugar donde estaba antes. Solamente podrá deshacer el último color añadido, pero repetir este proceso tantas veces como desee.</p>

<u><imgsrc="image/instrucciones/cambiarColor.png"alt=""width="456"height="353"hspace="12"align="bottom"></u></p>

<palign="center"> </p>

<p> </p>

<p>Recuerde:

 Debe elegir solamente aquellos colores que le parezcan importantes en el diseño.

Debe colocar los colores en la barra de resultados por su orden de importancia.</p>

<p>Cuando esté satisfecho con su elección, pulse sobre el botón (SIGUIENTE IMAGEN) situado en la parte derecha de la imagen</p>

<u><imgsrc="image/instrucciones/cambiarColor.png"alt=""width="477"height="337">

El programa le mostrará un total de cuatro imágenes para su evaluación. Por favor intente evaluarlas todas. La validez de los resultados obtenidos depende de su colaboración.</p>

<p>Una vez realizadas las cuatro series, el programa le mostrará los resultados de su evaluación comparándolos con los resultados promedios obtenidos por todos los usuarios que han realizado el experimento hasta la fecha.</p>

<p>Si ha comprendido la mecánica del experimento, pulse el siguiente botón

<inputtype="button"value="CLICK PARA EMPEZAR EL TEST"onClick="window.location='formulario.html' "></p>

<p> </p>

<p> </p>

</div>

<!-- end .container --></div>

</body>

</html>

1.4. Test

1.4.1. index.html

```
<!DOCTYPE html>
<html>
<head>

  <metahttp-equiv="Content-type"content="text/html;charset=UTF-8"/>
  <metaname="generator"content="0.8.726.95"/>
  <title>Home</title>
  <!-- CSS -->
  <linkrel="stylesheet" type="text/css"href="css/site_global.css"/>
  <linkrel="stylesheet" type="text/css"href="css/index.css"/>
  <!-- Other scripts -->
  <scripttype="text/javascript">
    document.documentElement.className = 'js';
  </script>
  <scriptsrc="scripts/llamadasAJAX.js" type="text/javascript"></script>
  <scriptsrc="scripts/paleta.js" type="text/javascript"></script>
</head>
<bodyonload="load()">

  <divclass="clearfix" id="page"><!-- column -->
    <divclass="position_content" id="page_position_content">
      <divclass="clearfix colelem" id="n1"><!-- group -->
        <divclass="grpelem" id="n2"></div>
        <divclass="clearfix grpelem" id="n3"><!-- column -->
          <divclass="clearfix colelem" id="n4"><!-- group -->
            <divclass="grpelem" id="n5"><!-- rasterized frame -->
              <imgid="n5_img"src="image/index_n5.png"alt="Elige los colores más importantes en el diseño, haz click en los colores para colocarlos en la barra, debes seleccionarlos de mayor a menor importancia"width="308"height="168"/>
            </div>
            <divclass="clearfix grpelem" id="n9"><!-- content -->
              <p>&nbsp;</p>
            </div>
          </div>
          <divclass="clearfix colelem" id="n13"><!-- group -->
            <divclass="clearfix grpelem" id="n14"><!-- column -->
              <divclass="colelem" id="n15"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="colelem" id="n16"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="colelem" id="n17"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="colelem" id="n18"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="clearfix grpelem" id="n19"><!-- column -->
              <divclass="colelem" id="n20"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="colelem" id="n21"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="colelem" id="n22"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="colelem" id="n23"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="clearfix grpelem" id="n24"><!-- column -->
              <divclass="colelem" id="n25"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="colelem" id="n26"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="colelem" id="n27"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="colelem" id="n28"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="clearfix grpelem" id="n29"><!-- column -->
              <divclass="colelem" id="n30"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
            <divclass="colelem" id="n31"onclick=elegirColor(this.id)><!-- simple frame --
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</body>
</html>
```

```

</div>
<divclass="colelem" id="n32" onclick=elegirColor(this.id)><!-- simple frame --
</div>
<divclass="colelem" id="n33" onclick=elegirColor(this.id)><!-- simple frame --
</div>
</div>
<divclass="clearfix grpelem" id="n34"><!-- column -->
<divclass="colelem" id="n35" onclick=elegirColor(this.id)><!-- simple frame --
</div>
<divclass="colelem" id="n36" onclick=elegirColor(this.id)><!-- simple frame --
</div>
<divclass="colelem" id="n37" onclick=elegirColor(this.id)><!-- simple frame --
</div>
<divclass="colelem" id="n38" onclick=elegirColor(this.id)><!-- simple frame --
</div>
</div>
</div>
</div>
</div>
<divclass="colelem" id="n39"><!-- group -->
<divclass="_n39 f3s_top"></div>
<divclass="_n39 f3s_mid clearfix">
<divclass="grpelem" id="n40" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n41" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n42" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n43" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n44" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n45" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n46" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n47" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n48" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n49" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n50" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n51" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n52" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n53" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n54" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n55" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n56" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n57" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n58" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
<divclass="grpelem" id="n59" name="barra" onclick=quitarColor(this.id)><!-- simple
frame --></div>
</div>
<divclass="_n39 f3s_bot"></div>
</div>
<divclass="colelem" id="n60"><!-- group -->
<divclass="_n60 f3s_top clearfix" id="siguiente" onclick="siguiente(this.id)">
<divclass="grpelem" id="n61"><!-- rasterized frame -->
<imgid="n61_img" src="image/index_n61.png" alt="SIGUIENTE
IMAGEN" width="160" height="15" />
</div>
</div>
</div>
</div>
</div>
<!-- JS includes -->
<scriptsrc="http://musecdn.businesscatalyst.com/scripts/1.0/jquery-
1.6.4.min.js" type="text/javascript"></script>
<scripttype="text/javascript">
window.jQuery || document.write('\x3Cscript src="scripts/1.0/jquery-1.6.4.min.js"

```

```
type="text/javascript">\x3C/script>');  
</script>  
<scriptsrc="scripts/1.0/sprydomutils.js" type="text/javascript"></script>  
<scriptsrc="scripts/1.0/museutils.js" type="text/javascript"></script>  
<!-- Other scripts -->  
<scripttype="text/javascript">  
    Muse.Uutils.addSelectorFn('body',  
Muse.Uutils.transformMarkupToFixBrowserProblemsPreInit);/* body */  
Muse.Uutils.addSelectorFn('a.nonblock', Muse.Uutils.addHyperlinkAnchor); /* a.nonblock */  
Muse.Uutils.addSelectorFn('body', Muse.Uutils.showWidgetsWhenReady);/* body */  
Muse.Uutils.addSelectorFn('body', Muse.Uutils.transformMarkupToFixBrowserProblems);/* body  
*/  
  
</script>  
</body>  
</html>
```

1.4.2. paleta.js

```
var t1;
var t2;
var tFin;
var idImagen;
var v=new Array(); //vector que apunta al array de json el 0 apunta al color 1 de ese
array a partir del id de la paleta
var u=new Array();//vector e eleccion de colores del usuario

var arrayDeIdsPaleta=new Array();
var huecosBarra=new Array();
var huecoBarraLLeno=new Boolean();
var huecoBarraLLeno="false";
var posicionArray;
var quitadosEnPaleta=new Array(); //Este array es para controlar que colores en la
paleta se han quitado si esta a uno esta quitado sino esta 1 (sirve para desactivar la
funcion javascript de la paleta para ese div)
//Representa si hay algun color puesto en la barra para asi saber donde poner un nuevo
color clickeado en la paleta

var coloresElegidos=0;
var idBidP=new Array(); //Array que asocia un id de la barra inferior a un id de la
paleta n7-> n20
var barra=new Array(); //Array que asocia posicion en barra con su respectivo id 0-
>n7
var idBPos=new Array(); //Array que asocia id de la barra con posicion n7->0
var generados =new Array(); //numeros generados a insertar en la barra
var contadorDeImagenes=0; // Contador de imagenes cargadas (al final seran las que hay
en la BBDD)
var numImagenesATestear=8;
var numColoresEnBarra=0;

idBPos["n40"]=0;
idBPos["n41"]=1;
idBPos["n42"]=2;
idBPos["n43"]=3;
idBPos["n44"]=4;

idBPos["n45"]=5;
idBPos["n46"]=6;
idBPos["n47"]=7;
idBPos["n48"]=8;
idBPos["n49"]=9;

idBPos["n50"]=10;
idBPos["n51"]=11;
idBPos["n52"]=12;
idBPos["n53"]=13;
idBPos["n54"]=14;

idBPos["n55"]=15;
idBPos["n56"]=16;
idBPos["n57"]=17;
idBPos["n58"]=18;
idBPos["n59"]=19;

barra[0]="n40";
barra[1]="n41";
barra[2]="n42";
barra[3]="n43";
barra[4]="n44";

barra[5]="n45";
barra[6]="n46";
barra[7]="n47";
barra[8]="n48";
barra[9]="n49";

barra[10]="n50";
```

```

barra[11]="n51";
barra[12]="n52";
barra[13]="n53";
barra[14]="n54";

barra[15]="n55";
barra[16]="n56";
barra[17]="n57";
barra[18]="n58";
barra[19]="n59";

prueba=0;
arrayDeIdsPaleta[0]="n15"; arrayDeIdsPaleta[1]="n16"; arrayDeIdsPaleta[2]="n17";
arrayDeIdsPaleta[3]="n18";
arrayDeIdsPaleta[4]="n20"; arrayDeIdsPaleta[5]="n21"; arrayDeIdsPaleta[6]="n22";
arrayDeIdsPaleta[7]="n23";
arrayDeIdsPaleta[8]="n25"; arrayDeIdsPaleta[9]="n26"; arrayDeIdsPaleta[10]="n27";
arrayDeIdsPaleta[11]="n28";
arrayDeIdsPaleta[12]="n30"; arrayDeIdsPaleta[13]="n31"; arrayDeIdsPaleta[14]="n32";
arrayDeIdsPaleta[15]="n33";
arrayDeIdsPaleta[16]="n35"; arrayDeIdsPaleta[17]="n36"; arrayDeIdsPaleta[18]="n37";
arrayDeIdsPaleta[19]="n38";
var trozeado=new Array();
function load(){ //Esta funcion se va encargar de cargar la primera imagen del test con
su respectiva paleta y arrancar el primer t1
    var xmlhttp;
    var xmlhttp3;

    var respuesta;
    var i;
    var correcto=0;

    fechal=new Date(); //Creacion objeto fecha

    document.getElementById('n2').style.backgroundImage="url('image/espere.png')";
    t1ms=fechal.getTime(); //devuelve los milisegundos desde 1970
    if(window.XMLHttpRequest)
        { // code for IE7+, Firefox, Chrome, Opera, Safari

            xmlhttp3=new XMLHttpRequest();
        }
    else
        { // code for IE6, IE5
            xmlhttp3=new ActiveXObject("Microsoft.XMLHTTP");
        }

    xmlhttp3.onreadystatechange=function(){
        if(xmlhttp3.readyState==4 && xmlhttp3.status==200){ //200=RESPUESTA OK,
readyState=4-> solicitud finalizada y respuesta esta preparada

            respuesta=xmlhttp3.responseText;
            trozeado=respuesta.split(",");
            if(window.XMLHttpRequest){ // code for IE7+,
Firefox, Chrome, Opera, Safari

                xmlhttp=new XMLHttpRequest();
            }
            else{ // code for IE6, IE5

                xmlhttp=new
ActiveXObject("Microsoft.XMLHTTP");
            }

            xmlhttp.onreadystatechange=function() //manejad
or de las respuestas del servidor
            {
                if(xmlhttp.readyState==4 &&
xmlhttp.status==200) //200=RESPUESTA OK, readyState=4-> solicitud finalizada y
respuesta esta preparada
                {
                    var json_data =
                    var the_object =eval("(" +
json_data + ")");

```



```

tas[trozeado[contadorDeImagenes]-1].url;
                                idImagen=the_object.colores.pale
                                var
cargaImagen="url('image/"+the_object.colores.paletas[trozeado[contadorDeImagenes]-
1].url+"')";
                                document.getElementById('n2').st
                                reseteandoVariables();
                                /*bucle que coje elementos de la
                                estos numeros aleatorios de 19
                                */
                                for(i=0;i<=19;i++){
                                    var
                                v[i]=aleatorio;
                                distribuirTabla(i,aleato
                                }
                                activo=1;
                                }
                                }
                                xmlhttp.open("POST","colores.json",true);//
                                xmlhttp.send();
                                }
                                }
                                xmlhttp3.open("POST","procesoSelectorImagenes.php",true);
                                xmlhttp3.send();
                                }
                                //Esta funcion vacia generados y quita los colores de la barra y deja huecosBarra todos
                                los elementos en 0.
                                function reseteandoVariables(){
                                    generados=[];
                                    coloresElegidos=0;
                                    numColoresEnBarra=0;
                                    for(i=0;i<=19;i++){
                                        document.getElementById(barra[i]).style.backgroundColor="transparent";
                                        document.getElementById(barra[i]).style.border="0px";
                                        document.getElementById(arrayDeIdsPaleta[i]).style.border="1px solid";
                                        huecosBarra[i]=0;
                                        quitadosEnPaleta[i]=0;
                                    }
                                }
                                //Esta funcion se encarga de insertar un elemento de la tabla object
                                function distribuirTabla(i,aleatorio,the_object){
                                    var red=the_object.colores.paletas[trozeado[contadorDeImagenes]-
                                1].rgbp[aleatorio].r;
                                    var
                                green=the_object.colores.paletas[trozeado[contadorDeImagenes]-1].rgbp[aleatorio].g;
                                    var
                                blue=the_object.colores.paletas[trozeado[contadorDeImagenes]-1].rgbp[aleatorio].b;
                                    v[arrayDeIdsPaleta[i]]=aleatorio;
                                    quitadosEnPaleta[arrayDeIdsPaleta[i]]=0;
                                    document.getElementById(arrayDeIdsPaleta[i]).style.backgroundCol
                                or="rgb("+ red+" "+green+" "+blue +")";
                                }
                                /*
                                Genera numeros aleatorios que no se hayan generado anteriormetne

```

para ello utiliza generados para recordar cuales se han generado.

```
*/
function aleatorio1_20(superior,inferior){

    var numPosibilidades;
    var igual=new Boolean();
    var numPosibilidades,aleat;
    var salida;

    var igual="true";

    while(igual=="true"){

        numPosibilidades = superior - inferior
        aleat = Math.random()* numPosibilidades
        aleat = Math.round(aleat)
        igual=buscar(aleat);
        if(igual=="false"){
            generados.push(aleat);
        }
    }

    salida= parseInt(inferior)+ aleat ;
    return salida;

}

function buscar(obj){
    var esta=new Boolean();
    esta="false"; //ponemos en false ya que solo tenemos que mantenernos generando
aleatorios solo si esta en el array de generados
    //alert("buscar elemento "+obj+" en "+generados);
    for(var i=0; i<generados.length; i++){
        if(generados[i]== obj){
            esta="true";
            return esta;
        }
    }

    return esta;
}

function dndEsta(){ //esta funcion no se podra entrar si hueco esta lleno
    var esta=new Boolean();
    esta="false"; //ponemos en false

    for(var i=0; i<=huecosBarra.length; i++){
        if(huecosBarra[i]==0){//Si encuentro un hueco en huecos barra lo pongo ocupado
            esta="true";
            posicionArray=i;
            huecosBarra[i]=1;//Ocupado el hueco
            coloresElegidos=coloresElegidos+1; // contador del nº de colores ya
            escogidos esto servira para hacer un control al leer la bara en la funcion siguiente()

            return i;
        }
    }
    huecoBarraLleno="true";
}

function elegirColor(idPaleta){

    if(quitadosEnPaleta[idPaleta]==0){
        var primeravez=0;
        var posicion;
        var idBarra;
        //Donde hay hueco en la barra

        posicion=dndEsta();
        anterior=posicion;
    }
}
```

```

//mediante quitadosEnPaleta[idPaleta]=1 indicamos que este color en esta
seccion div se ha seleccionado
quitadosEnPaleta[idPaleta]=1;

if(posicion+1>numColoresEnBarra){
    numColoresEnBarra=posicion+1;
}

//Conseguimos el id Barra del hueco que hay en la barra
//alert("posicion a poner en la barra "+posicion);
idBarra=barra[posicion];
//Asociamos el id de la barra donde lo vamos a poner al id de la
paleta que el usuario ha escogido.
idBidP[idBarra]=idPaleta;

//Cogemos el background color de la paleta y lo insertamos en la barra y
luego barromos el color de la paleta
document.getElementById(idBarra).style.backgroundColor=document.getEleme
ntById(idPaleta).style.backgroundColor;
document.getElementById(idBarra).style.border="1px solid";

//cambiando paleta
document.getElementById(idPaleta).style.backgroundColor="transparent";
document.getElementById(idPaleta).style.borderColor="transparent";
}
}

function quitarColor(idBarra){
    //idBarra esta asociado a un idPaleta idBidP[idBarra]->apunta a idPaleta
    if(huecosBarra[idBPos[idBarra]]==1){ //Este if es para que cuando ya no haya
color en la barra no se pueda modificar la paleta con el color de fondo de la barra
        huecosBarra[idBPos[idBarra]]=0
        document.getElementById(idBidP[idBarra]).style.backgroundColor=document.
getElementById(idBarra).style.backgroundColor;

        document.getElementById(idBidP[idBarra]).style.border='1px solid';
        document.getElementById(idBarra).style.backgroundColor="transparent";
        document.getElementById(idBarra).style.borderColor="transparent";

        quitadosEnPaleta[idBidP[idBarra]]=0; // Esto significa que el color se
a restaurado en la paleta

    }
}

function siguiente(idSiguiente){

    var i;
    var arrayColoresElegidos=new Array();
    var barraApuntador=new Array();
    var ejecutar=new Boolean();
    var xmlhttp2;
    //aquí accedo a todo la barra, despues hare un array con (codigo
imagen, array de style.backgroundColor estos son mediante rgb y el tiempo.).
    var barra=document.getElementsByName("barra");

    ejecutar="true";

    i=0;

    while(i<=coloresElegidos-1){

        if(huecosBarra[i]==0 && huecosBarra[i+1]==1){
            alert("No puede haber un hueco al principio de la barra
izquierda, ni tampoco huecos entre los colores");
            ejecutar="false";
        }
        i++
    }
    if(huecosBarra[0]==0 && huecosBarra[1]==0){

```

```

        alert("No puede haber un hueco al principio de la barra
izquierda, ni tampoco huecos entre los colores");
        ejecutar="false";
    }
    if(ejecutar=="true"){
        contadorDeImagenes=contadorDeImagenes+1; //incrementamos num
esto es para cargar la siguiente imagen del susuario
        if(contadorDeImagenes<4){

            document.getElementById('n2').style.backgroundImage="url
('image/espere.png');
        }
        else{
            document.getElementById('n2').style.backgroundImage="url
('image/cargandoFeedback.png');
        }

        //Dejamos el arrayDeColoresElegidos a 20 ya que el cero
significara que el usuario cree que ese color no predomina en la imagen.
        for(i=0;i<=19;i++){
            arrayColoresElegidos[i]=20;
        }

        for(i=0;i<=coloresElegidos-1;i++){
            //Aqui lo que hacemos es recorrer v buscando el primer
elemento que coincida con v[j], que quede claro que v[j]= son elementos que apuntan al
array d rgbs del json

            barraApuntador[i]=v[idBidP[barra[i].id]]; //Este array
representa cada elemento de la barra y a donde y a que color apunta en el JSON

        }

        //alert("coloresElegidos "+coloresElegidos);
        for(i=0;i<=19;i++){
            //alert("i "+i);
            for(j=0;j<=coloresElegidos-1;j++){ //Esto recorre barra

                if(barraApuntador[j]==i){

                    arrayColoresElegidos[i]=j+1; //Los
indices son apuntadores a los colores del JSON y "j" representa la posicion escogida por
el usuario

                }

            }

        }

        //problema a solucionar el xmlhttp2 crea un hilo si el usuario
le da siguiente demasiado rapida entrara otra vez otro proceso a siguiente
//las variable ejecutar se encontra a true y otra vez se llamara
a proceso barra.php asi hasta contadorDeImagenes=4
// posible solcuion poner sincrono con false-> profesor dice que
hay problemas de cargar fotos y que no le funciona el boton
// no tiene sentido ya que load funciona antes de la funcion
siguiente y deberia cargar las fotos
        fecha2=new Date();
        t2ms=fecha2.getTime();
        tFin=t2ms-t1ms;

        if(window.XMLHttpRequest){ // code for IE7+, Firefox, Chrome,
Opera, Safari

            xmlhttp2=new XMLHttpRequest();

        }
        else{ // code for IE6, IE5
            xmlhttp2=new ActiveXObject("Microsoft.XMLHTTP");
        }
    }
}

```

```

del servidor          xmlhttp2.onreadystatechange=function(){// fucion de respuesta

                    if(xmlhttp2.readyState==4 && xmlhttp2.status==200){
                        var correcto=xmlhttp2.responseText;
                        //alert(correcto);

                    if(contadorDeImagenes==4){//Esta funcion hace el
control de la imagen en la que estamos si era la ultima imagen en hacer click enviamos
al usuario al feedback

                    window.location="feedback.php"
                    contadorDeImagenes=0;
                    }
                    else{
                    load(); //llamos a la funcion cargar
que cargara una nueva imagen y una nueva paleta
                    }
                    }
                    //alert("llamando a proceso barra "+arrayColoresElegidos);
                    //alert("numero de colores en barra "+numColoresEnBarra);
                    xmlhttp2.open("POST", "procesoBarra.php?q="+arrayColoresElegidos+"&tiempo="+tFin+"&idImagen="+idImagen+"&numColorEnBarra="+numColoresEnBarra,true);
//paso el arrayDeColos al fichero php y el id del usuario relacionado
                    xmlhttp2.send();
                    }
}

```

1.4.3. procesoSelectorDelimagenes.php

```
<?php
    session_start();

    $sid=$_SESSION['idautoincrementado'];

    $con=mysql_connect("localhost","ekaitz","ekaitz");
    mysql_select_db("ekaitzBD",$con);

    if(!$con){
        die('Could not connect: '.mysql_error());
    }

    $sql="SELECT imagen1,imagen2,imagen3,imagen4
        FROM evaluaciones WHERE idEvaluacion='$sid'";

    $result=mysql_query($sql);
    while($row=mysql_fetch_array($result))
    {
        echo$row[0];
        echo",";
        echo$row[1];
        echo",";
        echo$row[2];
        echo",";
        echo$row[3];
    }
    mysql_close($con);
?>
```

1.4.4. procesoBarra.php

```
<?php
    session_start();

    //obtenemos los datos enviados por ajax desde el fichero paleta.js
    $q=$_GET['q'];
    $idImagen=$_GET['idImagen'];
    $tiempo=$_GET['tiempo'];
    $id=$_SESSION['idautoincrementado'];
    $numColoresEnBarra=$_GET['numColorEnBarra'];

    $cad=explode(",",$q);

    //conexion a la base de datos
    $con=mysql_connect("localhost","ekaitz","ekaitz");
    mysql_select_db("ekaitzBD",$con);

    if(!$con){
        die('Could not connect: '.mysql_error());
    }

    echo$numColoresEnBarra;

    //registro de los datos de la seleccion del usuario
    $sql="INSERT INTO testImagen (id,idImagen,numColoresElegidos
    ,tiempo,color1,color2,color3,color4,
    color5,color6,color7,color8,color9,color10,color11,color12,color13,
    color14,color15,color16,color17,color18,color19,color20)
    VALUES
    ('$id','$idImagen','$numColoresEnBarra','$tiempo','$cad[0]','$cad[1]','$cad[2]','$cad[3]
    ','$cad[4]',
    '$cad[5]','$cad[6]','$cad[7]','$cad[8]','$cad[9]','$cad[10]','$cad[11]',
    '$cad[12]','$cad[13]','$cad[14]','$cad[15]','$cad[16]','$cad[17]','$cad[18]','$cad[1
    9]')";

    if(!mysql_query($sql,$con)){
        die('Error: '.mysql_error());
    }

    //actualizacion de la tabla feedback con los datos de la ultima seleccion el usuario
    $sql="UPDATE feedback
    SET color1='$cad[0]'+color1, color2='$cad[1]'+color2, color3='$cad[2]'+color3,
    color4='$cad[3]'+color4, color5='$cad[4]'+color5,
    color6='$cad[5]'+color6,color7='$cad[6]'+color7,color8='$cad[7]'+color8,
    color9='$cad[8]'+color9, color10='$cad[9]'+color10, color11='$cad[10]'+color11,
    color12='$cad[11]'+color12, color13='$cad[12]'+color13,
    color14='$cad[13]'+color14, color15='$cad[14]'+color15, color16='$cad[15]'+color16,
    color17='$cad[16]'+color17, color18='$cad[17]'+color18,
    color19='$cad[18]'+color19,color20='$cad[19]'+color20
    WHERE url='$idImagen'";

    if(!mysql_query($sql,$con)){
        die('Error: '.mysql_error());
    }
?>
```

1.5. Feedback

1.5.1. feedback.php

```
<?php include 'iniciarFeedback.php'; include('jsondecode.php');?>
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="Content-type" content="text/html; charset=UTF-8" />
  <meta name="generator" content="0.8.726.95" />
  <title>FeedBack</title>
  <!-- CSS -->
  <link rel="stylesheet" type="text/css" href="css/site_global.css" />
  <link rel="stylesheet" type="text/css" href="css/feedback.css" />
  <!-- Other scripts -->
  <script type="text/javascript">
    document.documentElement.className = 'js';
  </script>
</head>
<body>
  <div class="clearfix" id="page"><!-- column -->
    <div class="colelem" id="n1"><!-- rasterized frame -->
      
    </div>
    <div class="clearfix colelem" id="n17"><!-- content -->
      <p>&nbsp;</p>
    </div>
    <div class="clearfix colelem" id="n21"><!-- group -->
      <?php
        $apuntadorImagenJSON=$imagen1;
        $etiqueta=22;

        $numPaleta=0;

        include("cargarImagenFeedback.php");
        ?>
      <div class="clearfix grpelem" id="n23"><!-- column -->
        <div class="clearfix colelem" id="n24"><!-- group -->
          <?php
            $etiqueta=24;
            $apuntadorImagenJSON=$imagen1;

            include("tratamientoFeedbackUsuarioFila.php");
            ?>
          </div>
        <div class="clearfix colelem" id="n45"><!-- group -->
          <?php
            $etiqueta=45;
            $apuntadorImagenJSON=$imagen1;

            include("tratamientoFeedbackRestoUsuarios.php");
            ?>
          </div>
        </div>
      </div>
      <div class="clearfix colelem" id="n66"><!-- group -->
        <?php
          $apuntadorImagenJSON=$imagen2;
          $etiqueta=67;
          $numPaleta=0;
          include("cargarImagenFeedback.php");

          ?>
        <div class="clearfix grpelem" id="n68"><!-- column -->
          <div class="clearfix colelem" id="n69"><!-- group -->
            <?php
              $etiqueta=69;
```



```

        $apuntadorImagenJSON=$imagen2;
        include("tratamientoFeedbackUsuarioFila.php");
    ?>
</div>
<div class="clearfix colelem" id="n90"><!-- group -->
    <?php
        $etiqueta=90;
        $apuntadorImagenJSON=$imagen2;
        include("tratamientoFeedbackRestoUsuarios.php");
    ?>
</div>
</div>
</div>
<div class="clearfix colelem" id="n111"><!-- group -->
<?php
    $apuntadorImagenJSON=$imagen3;
    $etiqueta=112;
    $numPaleta=0;
    include("cargarImagenFeedback.php");

    ?>
<div class="clearfix grpelem" id="n113"><!-- column -->
    <div class="clearfix colelem" id="n114"><!-- group -->
        <?php
            $etiqueta=114;
            $apuntadorImagenJSON=$imagen3;
            include("tratamientoFeedbackUsuarioFila.php");
        ?>
    </div>
    <div class="clearfix colelem" id="n135"><!-- group -->
        <?php
            $etiqueta=135;
            $apuntadorImagenJSON=$imagen3;
            include("tratamientoFeedbackRestoUsuarios.php");
        ?>
    </div>
</div>
</div>
<div class="clearfix colelem" id="n156"><!-- group -->
<?php
    $apuntadorImagenJSON=$imagen4;
    $etiqueta=157;
    $numPaleta=0;
    include("cargarImagenFeedback.php");

    ?>
<div class="clearfix grpelem" id="n158"><!-- column -->
    <div class="clearfix colelem" id="n159"><!-- group -->
        <?php
            $etiqueta=159;
            $apuntadorImagenJSON=$imagen4;
            include("tratamientoFeedbackUsuarioFila.php");
        ?>
    </div>
    <div class="clearfix colelem" id="n180"><!-- group -->
        <?php
            $etiqueta=180;
            $apuntadorImagenJSON=$imagen4;
            include("tratamientoFeedbackRestoUsuarios.php");
        ?>
    </div>
</div>
</div>
<div class="clearfix colelem" id="n201"><!-- group -->
    <div class="_n201 f3s_top clearfix" onClick="location.href = 'formulario.html'" >
        <div class="grpelem" id="n202" ><!-- content -->
            <p>&nbsp;</p>
        </div>
        <div class="grpelem" id="n206"><!-- rasterized frame -->
            
        </div>
    </div>
</div>
</div>
</div>
<!-- JS includes -->
<script src="http://musecdn.businesscatalyst.com/scripts/1.0/jquery-1.6.4.min.js"

```

```

type="text/javascript"></script>
  <script type="text/javascript">
    window.jQuery || document.write('\x3Cscript src="scripts/1.0/jquery-1.6.4.min.js"
type="text/javascript">\x3C/script>');
</script>
  <script src="scripts/1.0/sprydomutils.js" type="text/javascript"></script>
  <script src="scripts/1.0/museutils.js" type="text/javascript"></script>
  <!-- Other scripts -->
  <script type="text/javascript">
    Muse.Uutils.addSelectorFn('body',
Muse.Uutils.transformMarkupToFixBrowserProblemsPreInit);/* body */
Muse.Uutils.addSelectorFn('a.nonblock', Muse.Uutils.addHyperlinkAnchor); /* a.nonblock */
Muse.Uutils.addSelectorFn('body', Muse.Uutils.showWidgetsWhenReady);/* body */
Muse.Uutils.addSelectorFn('body', Muse.Uutils.transformMarkupToFixBrowserProblems);/* body
*/

</script>
</body>
</html>

```

1.5.2. iniciarFeedback.php

```
<?php
    session_start();
    $con=mysql_connect("localhost","ekaitz","ekaitz");
    mysql_select_db("ekaitzBD",$con);

    $id=$_SESSION['idautoincrementado'];

    if(!$con){
        die('Could not connect: '.mysql_error());
    }

    $sql="SELECT imagen1,imagen2,imagen3,imagen4
        FROM evaluaciones WHERE idEvaluacion='$id'";

    $result=mysql_query($sql);

    while($row=mysql_fetch_array($result))
    {
        $imagen1=$row[0]-1;
        $imagen2=$row[1]-1;
        $imagen3=$row[2]-1;
        $imagen4=$row[3]-1;

    }

?>
```

1.5.3. jsondecode.php

```
<?php
function json_decode($json)
{
    $comment=false;
    $out='$x=';
    for($i=0;$i<strlen($json);$i++)
    {
        if(!$comment)
        {
            if(($json[$i]=='{'||($json[$i]=='['))
                $out.=' array(';
            elseif(($json[$i]=='}'||($json[$i]==']'))
                $out.=')';
            elseif($json[$i]==':')
                $out.='>';
            else
                $out.=$json[$i];
        }
        else
            $out.=$json[$i];
        if($json[$i]=='"'&&$json[($i-1)]!="\\")
            $comment=!$comment;
    }
    eval($out.'');
    return$x;
}

?>
```

1.5.4. cargarImagenFeedback.php

<?php

```
//capta el fichero json
$cadena=file("colores.json");

//pasa de array a string
$string=implode($cadena);

// parseamos el codigo json
$obj_json=json_decode($string,true);

//accedemos al nombre de la imagen mediante $apuntadorImagenJSON
$imagen=$obj_json["colores"]["paletas"][$apuntadorImagenJSON]["url"];

// accedemos un echo con para mostrar la imagen mediante la etiqueta <img> de html
echo"<div class=\"grpelem\" id=\"n\".$etiqueta.\"><img
src=\"image/redimensionadasFeedback/\".$imagen.\"></div>";

?>
```

1.5.5. tratamientoFeedbacUsuarioFila.php

<?php

```
$cadena=file("colores.json");
$string=implode($cadena);
$obj_json=json_decode($string,true);

//accedemos a la url de la imagen que nos corresponde mediante
$apuntadorImagenJSON
$imagen=$obj_json["colores"]["paletas"][$apuntadorImagenJSON]["url"];
$numPaleta=$apuntadorImagenJSON;

//consulta de los colores se lecciones para la imagen aacutal
$consultasql="SELECT
color1,color2,color3,color4,color5,color6,color7,color8,color9,color10,color11,color12,c
olor13,color14,color15,color16,color17,color18,color19,color20 FROM testImagen WHERE
idImagen=' $imagen' AND id=' $id'";
$coloreElegidos=mysql_query($consultasql);

//consulsta para saber cuantos colores selecciono el usuario para una imagen
en concreto
$consultasql="SELECT numColoresElegidos FROM testImagen WHERE
idImagen=' $imagen' AND id=' $id'";

$numColoresElegidos=mysql_query($consultasql);

$row=mysql_fetch_array($coloreElegidos);
$fila=mysql_fetch_array($numColoresElegidos);

$i=0; //apuntador en el string que se mueve para buscar el minimo del array
row
$abuscar=1;//elemento a buscar por en row

// este bucle accede a los colores seleccionados por el usuario y los muestra
mediante un echo
while($abuscar<=$fila[0]){

    while($i<=19 ){

        if($row[$i]==$abuscar){//si elemento abuscar esta en row imprimimos
el color
```

```

$etiqueta=$etiqueta+1;
$r=$obj_json["colores"]["paletas"][$numPaleta]["rgbp"][$i]["r"];
$g=$obj_json["colores"]["paletas"][$numPaleta]["rgbp"][$i]["g"];
$b=$obj_json["colores"]["paletas"][$numPaleta]["rgbp"][$i]["b"];

echo"<div class=\"grpelem\" id=\"n.$etiqueta.\"
style=\"background-color:rgb(\".$r.\".\".$g.\".\".$b.\");border:1px solid black;\"><!-- simple
frame --></div>";

    }
    $i=$i+1;
}
$i=0;
$abuscador=$abuscador+1;
}
?>

```

1.5.6. tratamientoFeedbackRestosUsuarios.php

```
<?php
    $cadena=file("colores.json");

    $string=implode($cadena);
    $obj_json=json_decode($string,true);

    $url=$obj_json["colores"]["paletas"][$apuntadorImagenJSON]["url"];
    $imagen=$url;
    $numPaleta=$apuntadorImagenJSON;

    $consultasql="SELECT
color1,color2,color3,color4,color5,color6,color7,color8,color9,color10,color11,color12,c
olor13,color14,color15,color16,color17,color18,color19,color20 FROM feedback WHERE
url='$url'";
    $coloresElegidos=mysql_query($consultasql);
    $consultasql="SELECT numColoresElegidos FROM testImagen WHERE idImagen='$imagen'
AND id='$id'";

    $numColoresElegidos=mysql_query($consultasql);

    $row=mysql_fetch_array($coloresElegidos);
    $fila=mysql_fetch_array($numColoresElegidos);
    $i=0; //apuntador en el string que se mueve para buscar el minimo del array row
    $abuscar=1;//elemento que estamos buscando ira incrementando hasta encontrar el
primero que sera el minimo
    $buscados=1;
    $maximo=max($row);

    while($abuscar<=$maximo&&$buscados<=$fila[0]){//abucar apunta a max del
maximo del array y el numero encontrados es menor al maximo elegido por el usuario
seguimos en el bucle

        while($i<=19){
            if($row[$i]==$abuscar&&$buscados<=$fila[0]){//si elemento
abucar esta en row imprimimos el color

                $etiqueta=$etiqueta+1;
                $r=$obj_json["colores"]["paletas"][$numPaleta]["rgbp"][$i][
r"];
                $g=$obj_json["colores"]["paletas"][$numPaleta]["rgbp"][$i][
g"];
                $b=$obj_json["colores"]["paletas"][$numPaleta]["rgbp"][$i][
b"];

                echo"<div class=\"grpelem\" id=\"n\".$etiqueta.\"\"
style=\"background-color:rgb(\".$r.\"\",\".$g.\"\",\".$b.\");border:1px solid black;\">><!-- simple
frame --></div>";

                $buscados=$buscados+1;

            }
            $i=$i+1;
        }
        $i=0;
        $abuscar=$abuscar+1;
    }

?>
```

2. colores.json

2.1 Fichero colores.json

Sólo se incluyen los datos de un color porque insertar los de todos ocuparía demasiado en el proyecto. De todas formas, el fichero íntegro se encuentra disponible en su totalidad en el CD adjunto.

```
{
  "colores":{
    "paletas":[
      {
        "id":1,
        "url":"rockwerchter_be_I1_small.GIF",
        "rgbp":[
          {
            "r":0,
            "g":0,
            "b":0,
            "p":0.9752955219
          },
          {
            "r":251,
            "g":251,
            "b":250,
            "p":0.0033988205
          },
          {
            "r":230,
            "g":107,
            "b":8,
            "p":0.0025339563
          },
          {
            "r":16,
            "g":10,
            "b":7,
            "p":0.0024892770
          },
          {
            "r":206,
            "g":135,
            "b":151,
            "p":0.0022307751
          },
          {
            "r":198,
            "g":91,
            "b":10,
            "p":0.0022243924
          },
          {
            "r":204,
            "g":37,
            "b":21,
            "p":0.0019180198
          },
          {
            "r":43,
            "g":41,
            "b":40,
            "p":0.0018382353
          },
          {
            "r":178,
            "g":35,
            "b":18,
            "p":0.0016595180
          }
        ]
      }
    ]
  }
}
```

```

    {
      "r":171,
      "g":120,
      "b":133,
      "p":0.0015350541
    },
    {
      "r":71,
      "g":70,
      "b":66,
      "p":0.0009542228
    },
    {
      "r":117,
      "g":48,
      "b":5,
      "p":0.0007914624
    },
    {
      "r":225,
      "g":214,
      "b":212,
      "p":0.0007786969
    },
    {
      "r":105,
      "g":95,
      "b":97,
      "p":0.0006127451
    },
    {
      "r":163,
      "g":164,
      "b":164,
      "p":0.0004914726
    },
    {
      "r":68,
      "g":34,
      "b":1,
      "p":0.0004755157
    },
    {
      "r":230,
      "g":175,
      "b":162,
      "p":0.0003223295
    },
    {
      "r":79,
      "g":13,
      "b":8,
      "p":0.0001851001
    },
    {
      "r":227,
      "g":135,
      "b":95,
      "p":0.0001340380
    },
    {
      "r":26,
      "g":148,
      "b":74,
      "p":0.0001308466
    }
  ],
},

```


2.1 Programa Java para la creación del fichero colores.json

```
import java.io.*;

public class ponFormato {

    static FileWriter fichero;
    static PrintWriter pw;
    static String spliteado[];
    static int j;

    public static void main(String[] args)
    {
        try{
            fichero = new FileWriter(
=new PrintWriter(fichero);

            FileInputStream fstream
=new FileInputStream("/Users/ekaitzhernandez/Documents/workspace/formateando/bin/datosImagen.txt");
            DataInputStream in = new DataInputStream(fstream);
            BufferedReader br
=new BufferedReader(new InputStreamReader(in));
            String strLine;
            strLine=br.readLine();
            for(int i =1;i<=40;i++){
                strLine=br.readLine();
                tratamiento(strLine,i);
                j=1;
                while(j<=20) {
                    // Print the content on the console
                    strLine = br.readLine();
                    System.out.println(strLine);
                    spliteado = strLine.split(";");
                    pw.println("{");
                    pw.println("\r\":"+spliteado[1]+",");
                    pw.println("\g\":"+spliteado[2]+",");
                    pw.println("\b\":"+spliteado[3]+",");
                    pw.println("\p\":"+spliteado[4]);
                    if(j!=20){
                        pw.println(",");
                    }
                    else{
                        pw.println("}");
                    }
                    j++;
                }
                strLine=br.readLine();
                pw.println("}");// el cierre del rgbp
                pw.println(",");//el cierre del id
            }
            pw.println("}");//el cierre del id sin "," es el ultimo

            //Close the input stream
            in.close();
            pw.close();
        }

        catch(Exception e){//Catch exception if any
            System.err.println("Error: "+ e.getMessage());
        }
    }

    public static void tratamiento(String linea,int i) throws IOException{
        pw.println("{");
        pw.println("\id\":"+i+",");
        pw.println("\url\":"+linea+"\",");
        pw.println("\rgbp\":"+linea+"\",");

        //Close the output stream
    }
}
```

```
}  
}
```

Para ejecutar el programa anterior, debemos crear un fichero llamado DatosImagen.txt con la información que queremos representar mediante JSON.

El fichero DatosImagen.txt debe tener el siguiente formato:

[
aquí debe ir un salto de línea vacío]

```
rockwerchter_be_I1_small.GIF //aquí debe ir el nombre de la imagen  
1;0;0;0;0.9752955219; //aquí debe ir el primer color con mas proporción de color indic  
2;251;251;250;0.0033988205; // el formato de cada una de estas líneas es el siguiente  
id;red;Green;blue;porcentaje del color en cuestión.  
3;230;107;8;0.0025339563;  
4;16;10;7;0.0024892770;  
5;206;135;151;0.0022307751;  
6;198;91;10;0.0022243924;  
7;204;37;21;0.0019180198;  
8;43;41;40;0.0018382353;  
9;178;35;18;0.0016595180;  
10;171;120;133;0.0015350541;  
11;71;70;66;0.0009542228;  
12;117;48;5;0.0007914624;  
13;225;214;212;0.0007786969;  
14;105;95;97;0.0006127451;  
15;163;164;164;0.0004914726;  
16;68;34;1;0.0004755157;  
17;230;175;162;0.0003223295;  
18;79;13;8;0.0001851001;  
19;227;135;95;0.0001340380;  
20;26;148;74;0.0001308466;
```

[
aquí debe ir un salto de línea vacío]

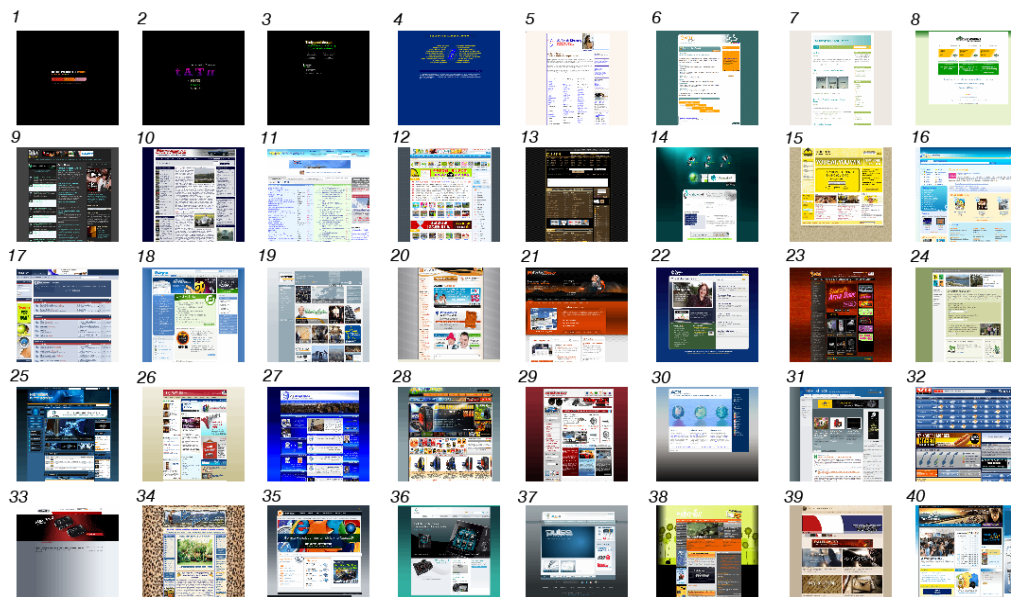
```
tatu_ru_I1_small.GIF  
1;0;0;0;0.9736328125;  
2;92;91;90;0.0041232639;  
3;11;15;10;0.0030828738;  
4;67;5;113;0.0026041667;  
5;218;217;213;0.0020839716;  
6;189;1;156;0.0019020629;  
7;155;2;143;0.0015414369;  
8;103;3;124;0.0014552696;  
9;201;191;166;0.0013659109;  
10;130;3;132;0.0013084661;  
11;30;1;42;0.0010180504;  
12;162;156;141;0.0008872038;  
13;48;5;95;0.0008520987;  
14;1;126;1;0.0008138021;  
15;124;119;108;0.0007499745;  
16;39;38;35;0.0006989124;  
17;69;67;60;0.0005935968;  
18;3;82;4;0.0005170037;  
19;70;1;73;0.0004787071;  
20;99;1;88;0.0002904156
```

Los datos aparecerán con el nombre del fichero salida.txt. Para comprobar que la salida está codificada correctamente según el estándar JSON, podemos usar <http://jsonlint.com/> y validarla.

3. Imágenes testeadas

3.1. Imágenes testeadas por los usuarios

Las siguientes imágenes han sido testeadas por los usuarios. El orden en el que figuran corresponde al del fichero *colores.json*. Las imágenes en su resolución original se encuentran disponibles en el CD adjunto.



3.2. URL de las imágenes almacenadas en el servidor

1. rockwerchter_be_I1_small.GIF
2. tatu_ru_I1_small.GIF
3. greensock_com_I1_small.GIF
4. eeb_org_I1_small.GIF
5. al-articledirectory_com_I1_small.GIF
6. ganttproject_biz_I1_small.GIF
7. rmarsh_com_I1_small.GIF
8. net-publics_de_I1_small.GIF
9. guitarsite_com_I1_small.GIF
10. ultimatecarpage_com_I1_small.GIF
11. forumavia_ru_I1_small.GIF
12. 3839_com_I1_small.GIF
13. gangqinpu_com_I1_small.GIF
14. fullshare_net_I1_small.GIF
15. cpk_com_I1_small.GIF
16. skyglobe_ru_I1_small.GIF
17. baybul_com_I1_small.GIF
18. devexpress_com_I1_small.GIF
19. phoenix_de_I1_small.GIF
20. axxess_co_za_I1_small.GIF
21. rockettheme_com_I1_small.GIF
22. pcc_edu_I1_small.GIF
23. spencersonline_com_I1_small.GIF
24. geocaching_com_I1_small.GIF
25. hangarnet_com_br_I1_small.GIF
26. trevvy_com_I1_small.GIF
27. fcesteghlal_ir_I1_small.GIF
28. cyberpowerpc_com_I1_small.GIF
29. vividracing_com_I1_small.GIF
30. acninc_com_I1_small.GIF
31. anandtech_com_I1_small.GIF
32. pent_no_I1_small.GIF
33. line6_com_I1_small.GIF
34. kwathabeng_co_za_I1_small.GIF
35. futuremark_com_I1_small.GIF
36. logicool_co_jp_I1_small.GIF
37. livescribe_com_I1_small.GIF
38. todopsp_com_I1_small.GIF
39. twcenter_net_I1_small.GIF
40. amgentourofcalifornia_com_I1_small.GIF

