



ugr

**Universidad de Granada**  
Departamento de Ciencias de la Computación  
e Inteligencia Artificial



---

## Asignatura: Programación Web

---

**Curso 2014/15**

**Curso: 3º Grado**

**Grupo de prácticas: 2**

---

### Práctica: 2

---

**Datos de los Alumnos: (ordenar alfabéticamente por apellidos).**

**Apellidos:** Andrades Molina

**Nombre:** Christian

**D.N.I.:** 75897720-C

**e-mail ugr:** [christianam@correo.ugr.es](mailto:christianam@correo.ugr.es)



**Apellidos:** Rodríguez Lupiáñez

**Nombre:** Celia

**D.N.I.:** 74737258-T

**e-mail ugr:** [cerl@correo.ugr.es](mailto:cerl@correo.ugr.es)



---

## Especificación de requisitos de software


Proyecto: Tablero de dibujo  
Revisión 1.0





Mayo

---

## Ficha del documento

Fecha	Revisión	Autores	Verificado dep. calidad.
30/05/2015	1.0	Christian Andrades Molina Celia Rodríguez Lupiáñez	

Documento validado por las partes en fecha: 30/05/2015

Por el cliente	Por la empresa suministradora
	
Fdo. D. Waldo Fajaldo Contreras	Fdo. D. Christian Andrades Molina

# Contenido

<b>FICHA DEL DOCUMENTO</b>	<b>3</b>
<b>CONTENIDO</b>	<b>4</b>
<b>1 INTRODUCCIÓN</b>	<b>6</b>
1.1 Propósito	6
1.2 Alcance	6
1.3 Personal involucrado	6
1.4 Definiciones, acrónimos y abreviaturas	7
1.5 Referencias	8
1.6 Resumen	9
<b>2 DESCRIPCIÓN GENERAL</b>	<b>9</b>
2.1 Perspectiva del producto	9
Funcionalidad del producto	9
2.2 Características de los usuarios	10
2.3 Restricciones	10
2.4 Suposiciones y dependencias	10
2.5 Evolución previsible del sistema	10
<b>3 REQUISITOS ESPECÍFICOS</b>	<b>11</b>
<b>3.1 Requisitos comunes de los interfaces</b>	<b>11</b>
3.1.1 Interfaces de usuario	11
3.1.2 Interfaces de hardware	12
3.1.3 Interfaces de software	12
3.1.4 Interfaces de comunicación	12
<b>3.2 Requisitos funcionales</b>	<b>13</b>
3.2.1 Requisito funcional 1	13
3.2.2 Requisito funcional 2	14
3.2.3 Requisito funcional 3	15
3.2.4 Requisito funcional 4	16
3.2.5 Requisito funcional 5	17
3.2.6 Requisito funcional 6	18
<b>3.3 Requisitos no funcionales</b>	<b>19</b>
3.3.1 Requisitos de rendimiento	19
3.3.2 Seguridad	20
3.3.3 Fiabilidad	20

---

3.3.4	Disponibilidad	20
3.3.5	Mantenibilidad	21
3.3.6	Portabilidad	21
<b>3.4</b>	<b>Otros requisitos</b>	<b>21</b>
<b>4</b>	<b>APÉNDICES</b>	<b>22</b>
<b>5</b>	<b>INTRODUCCIÓN DE LA SOLUCIÓN</b>	<b>22</b>
<b>6</b>	<b>DESCRIPCIÓN DE LA SOLUCIÓN</b>	<b>22</b>
<b>6.1</b>	<b>Interfaz(HTML)</b>	<b>23</b>
<b>6.2</b>	<b>Interfaz (CSS)</b>	<b>24</b>
6.2.1	Estilos del contenido:	24
6.2.2	Estilo del tablero:	24
6.2.3	Estilo de la paleta:	25
<b>6.3</b>	<b>Funcionalidad (JAVASCRIPT)</b>	<b>26</b>
6.3.1	Asignar paleta	26
6.3.2	Cambiar color	27
6.3.3	Dibujar tabla	27
6.3.4	Paint	27
6.3.5	Pintar	28
<b>7</b>	<b>CONCLUSIÓN</b>	<b>28</b>
<b>8</b>	<b>BIBLIOGRAFIA</b>	<b>29</b>

---

# 1 Introducción

El presente documento referido a la especificación de requerimientos de software (SRS) contiene la información necesaria para que a los desarrolladores del software y externos le ayuden a analizar y entender los requisitos y necesidades que el cliente reclama (en este caso, el profesor de la asignatura). Por lo tanto, constará de una descripción detallada de la aplicación a desarrollar, en este caso, un tablero de dibujo con varias funciones. A partir del documento obtendremos un detalle extenso sobre la interfaz del usuario, funciones a implementar, restricciones y otros aspectos a tener en cuenta.

Este documento está estructurado en base al estándar IEEE Std 830-1998.

## 1.1 Propósito

El propósito del documento es establecer las bases de acuerdo entre las personas implicadas en el software en el cual se basa esta especificación. Se presentará de manera formal la SRS con el fin de llevar a cabo su discusión y aceptación. Su función principal será ayudar a los usuarios directos de este sistema (en primera instancia) y en segunda, al grupo de desarrolladores de la aplicación.

En definitiva, la directiva del documento es explicar las características de las que debe disponer el programa, para tener una idea estructurada de forma que al crear el programa este funcione correctamente.

## 1.2 Alcance

El software a desarrollar se identificará como: "Tablero de dibujo". La aplicación brindará las funciones principales de un software con un uso sencillo e intuitivo, con el objetivo de ser accesible para dibujar a cualquier usuario sin importar su edad. Permitirá:

- Pintar / borrar en el tablero dispuesto con el color seleccionado tantas veces como el usuario desee.
- Una amplia gama de colores a elegir
- Emitir información sobre el estado del pincel y el color seleccionado

## 1.3 Personal involucrado

Nombre	Christian Andrades Molina
Rol	Programador, diseñador y gestor
Categoría profesional	Junior
Responsabilidades	Programar los módulos del sistema, diseñar la interfaz y realizar la documentación.
Información de contacto	<a href="mailto:christianam@correo.ugr.es">christianam@correo.ugr.es</a>
Aprobación	

---

Nombre	Celia Rodríguez Lupiáñez
Rol	Programador, diseñador y gestor
Categoría profesional	Junior
Responsabilidades	Programar los módulos del sistema, diseñar la interfaz y realizar la documentación.
Información de contacto	<a href="mailto:cerl@correo.ugr.es">cerl@correo.ugr.es</a>
Aprobación	

## 1.4 Definiciones, acrónimos y abreviaturas

- **SRS:** es una parte importante del proceso de requisitos del ciclo de vida de software y se usa en el diseño, aplicación, supervisión, comprobación, aprobación y pruebas como está descrito en IEEE Std. 1074-1997.
  - **Tablero de dibujo:** aplicación informática que permite llevar a cabo dibujos con mayor o menor habilidad.
  - **Interfaz:** conexión física y funcional entre dos sistemas o dispositivos de cualquier tipo dando una comunicación entre distintos niveles.
  - **IEEE:** es una asociación mundial de técnicos e ingenieros dedicada a la estandarización y el desarrollo en áreas técnicas. Con cerca de 425 000 miembros y voluntarios en 160 países, es la mayor asociación internacional sin ánimo de lucro formada por profesionales de las nuevas tecnologías, como ingenieros eléctricos, ingenieros en electrónica, científicos de la computación, ingenieros en computación, matemáticos aplicados, ingenieros en biomedicina, ingenieros en telecomunicación, ingenieros en mecatrónica, etc.
  - **Software:** equipamiento lógico o soporte lógico de un sistema informático, que comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos que son llamados hardware.
  - **Sistema operativo:** es un programa o conjunto de programas de un sistema informático que gestiona los recursos de hardware y provee servicios a los programas de aplicación, ejecutándose en modo privilegiado respecto de los restantes.
-

- **Javascript:** es un lenguaje de programación interpretado, dialecto del estándar ECMA Script. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.
- **HTML:** es un estándar que sirve de referencia para la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, videos, entre otros.
- **CSS:** es un lenguaje usado para definir y crear la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El World Wide Web Consortium (W3C) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.
- **Ratón:** es un dispositivo apuntador utilizado para facilitar el manejo de un entorno gráfico en una computadora (ordenador). Generalmente está fabricado en plástico, y se utiliza con una de las manos.
- **Nube:** la computación en la nube, conocido también como servicios en la nube, informática en la nube, nube de cómputo o nube de conceptos, (del inglés *cloudcomputing*), es un paradigma que permite ofrecer servicios de computación a través de Internet.
- **Navegador web:** Un navegador o navegador web, o browser, es un software que permite el acceso a Internet, interpretando la información de distintos tipos de archivos y sitios web para que éstos puedan ser visualizados.
- **Monitor:** El monitor de computadora (en Hispanoamérica) o pantalla de ordenador (en España) es el principal dispositivo de salida (interfaz), que muestra datos o información al usuario.
- **Bug:** un error de software, comúnmente conocido como *bug* («bicho»), es un error o fallo en un programa de computador o sistema de *software* que desencadena un resultado indeseado. Los programas que ayudan a la detección y eliminación de errores de programación de *software* son denominados depuradores (*debuggers*).

## 1.5 Referencias

Referencia	Título	Ruta	Fecha	Autor
NA	NA	NA	NA	NA



## 1.6 Resumen

El documento consistirá a partir de esta sección en tres apartados bien identificados.

Una primera sección dedicada a la introducción del mismo, destacando los objetivos que tiene el SRS y una visión general de nuestra aplicación

La segunda sección del documento consistirá en una descripción general del software, con la finalidad de abarcar las funciones más importantes que la aplicación debe acometer, así como sus características y posibles limitaciones que podría tener.

Por último, una tercer apartado centrado en los requerimientos específicos, mostrando paso a paso los requerimientos que el usuario desea obtener con el producto final.

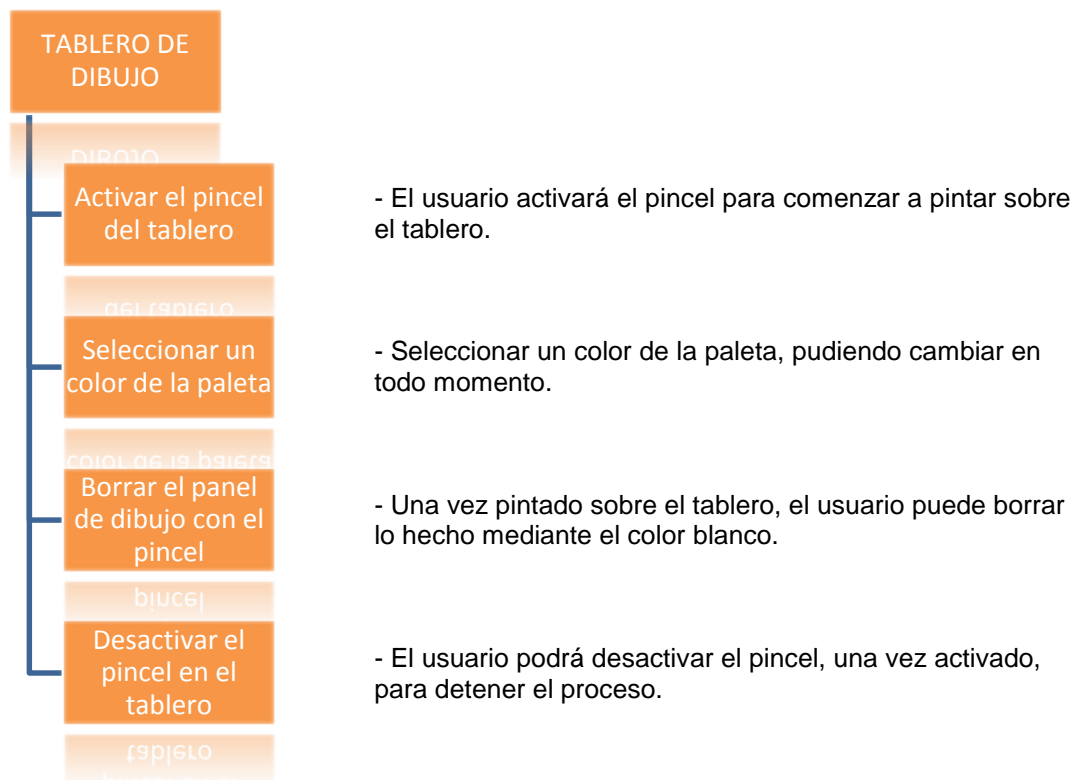
## 2 Descripción general

### 2.1 Perspectiva del producto

La aplicación a desarrollar es un producto independiente, soportado por cualquier sistema operativo, sin necesidad de acceso a internet y como requisito el acceso a un navegador web cualquiera. La interacción de los usuarios con el software se llevará a cabo mediante la interfaz diseñada.

### Funcionalidad del producto

Entre las funciones generales que debe realizar el sistema podemos encontrar:



## 2.2 Características de los usuarios

Tipo de usuario	Usuario de la aplicación
Formación	No se requiere ninguna formación específica
Habilidades	Habilidad básicas o profesionales de dibujo
Actividades	Realizar dibujos

Tipo de usuario	Técnico del sistema
Formación	Técnico en informática
Habilidades	Manejo avanzado de programación web y mantenimiento
Actividades	Mantenimiento y actualización de la aplicación

## 2.3 Restricciones

La aplicación estará desarrollada en Javascript para las funcionalidades y HTML / CSS para el diseño de la interfaz. Está soportada para todos los sistemas operativos y la mayoría de navegadores existentes (salvo excepciones). Se requiere el uso de un ratón laser o una tableta de dibujo, para una mayor precisión. La selección de colores se limita a los proporcionados por la aplicación (10) y el área de dibujo, un tablero de 50x50 celdas de 10x10 puntos.

## 2.4 Suposiciones y dependencias

El acceso desde un navegador no soportado podría dar problemas.

## 2.5 Evolución previsible del sistema

En caso de que la aplicación se actualizara, podrían llevarse a cabo las siguientes mejoras:

- Implementar un mayor número de colores a disposición del usuario.
- Nuevo tablero de dibujo con mayor extensión y sin división de cuadrícula.
- Diferentes métodos de dibujo, con pinceles de tamaño y tipo variados.
- Posibilidad de dibujar formas geométricas con modelos.
- Mejora visual de la interfaz.
- Implementación de un sistema en línea con posibilidad de compartir el tablero con otros usuarios.
- Añadir el acceso a redes sociales para compartir tus creaciones.
- Activar un modo pantalla completa.
- Permitir el acceso a un modo “tableta”, habilitando la interfaz para usar este periférico externo.
- Guardar tus creaciones en la nube o en el propio sistema local.

## 3 Requisitos específicos

### 3.1 Requisitos comunes de los interfaces

No consta en la aplicación.

#### 3.1.1 Interfaces de usuario

Las interfaces de usuario abarcan todo lo relacionado con la pantalla, ventanas que el usuario visualizará para realizar operaciones. Estas acciones se llevarán a cabo mediante ratón o tableta de dibujo.

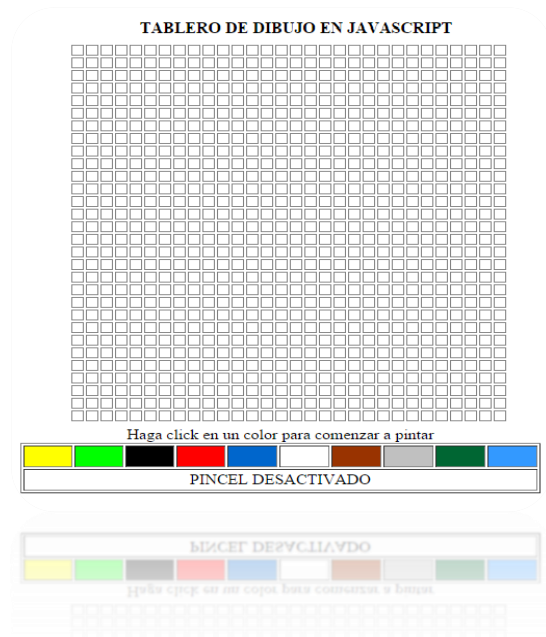
Estas interfaces por lo tanto ayudarán al usuario a llevar a cabo los procesos que dispone el sistema. En nuestro caso, dispondremos de una única interfaz compuesta de:

- Tablero de dibujo dividido en celdas de dibujo.
- Mensaje de ayuda al usuario.
- Menú selector de colores.
- Cuadro de aviso sobre el estado del pincel.
- Título de la aplicación.
- Pincel (ratón).

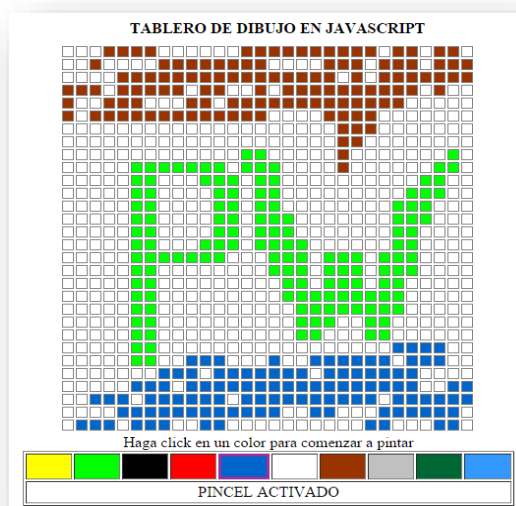
Mediante capturas de pantalla, mostraremos la interfaz de la que dispone la aplicación y sus variaciones según las acciones:

#### 1. Pantalla principal y única.

Podemos ver la interfaz completa de la aplicación, con su tablero de dibujo y el selector de colores.



2. Interfaz de la aplicación tras realizar varias acciones tales como pintar en el tablero de dibujo, seleccionar en la paleta el color azul (remarcado) y el estado del pincel activado.



### 3.1.2 Interfaces de hardware

- **Monitor:** mostrará la interfaz del software al usuario, reuniendo todas las funciones disponibles en la pantalla. Se recomienda una resolución mayor de 1024x768 para un correcto funcionamiento.
- **Ratón:** la aplicación requiere del uso de un ratón para que el usuario pueda llevar a cabo todas las acciones, seleccionando colores o pintando.
- **Impresora (opcional):** no incluye una opción específica para imprimir únicamente el resultado obtenido en el tablero.
- **Tableta de dibujo (opcional):** permitirá mayor precisión en el dibujo.

### 3.1.3 Interfaces de software

- **Navegador web:** es un software que permite el acceso a Internet, interpretando la información de distintos tipos de archivos y sitios web para que éstos puedan ser visualizados. Su función principal es visualizar la interfaz de la aplicación.

### 3.1.4 Interfaces de comunicación

No consta en la aplicación

## 3.2 Requisitos funcionales

### 3.2.1 Requisito funcional 1

Número de requisito	RF1
Nombre de requisito	Permitir dibujar en el tablero de dibujo
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Tablero de dibujo
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

#### INTRODUCCIÓN

La aplicación debe permitir dibujar en el tablero dedicado a ello mediante el pincel proporcionado (en este caso, el cursor del ratón).

#### SECUENCIA DE OPERACIONES

Mediante el ratón, haremos click para activar el pincel y arrastraremos el mismo por el tablero de dibujo para pintar (habiendo seleccionado previamente el color).

#### RESPUESTA A SITUACIONES ANORMALES

Si el usuario no selecciona un color o no activa el pincel, no dibujará sobre el tablero.

#### ENTRADAS

Interacción del usuario con el periférico apropiado para la aplicación (ratón).

#### SALIDAS

Se mostrará en pantalla el resultado de dibujar en el tablero de dibujo según la habilidad del usuario, colores seleccionados, etc.

#### RELACIÓN ENTRE ENTRADAS Y SALIDAS

La interacción del usuario en el tablero de dibujo será la salida mostrada en este proceso.

#### PARAMETROS

Para poder dibujar, tendremos que haber seleccionado un color previamente que será el que recibirá.

#### REQUISITOS LÓGICOS

No constan.

### 3.2.2 Requisito funcional 2

Número de requisito	RF2
Nombre de requisito	Permitir borrar en el tablero de dibujo
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Tablero de dibujo
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

#### INTRODUCCION

La aplicación debe permitir borrar lo hecho en el tablero dedicado a ello mediante el pincel proporcionado (en este caso, el cursor del ratón).

#### SECUENCIA DE OPERACIONES

Mediante el ratón, haremos click para activar el pincel y arrastraremos el mismo por el tablero de dibujo para borrar (habiendo seleccionado previamente el color blanco).

#### RESPUESTA A SITUACIONES ANORMALES

Si el usuario no selecciona el color blanco o no activa el pincel, no borrará sobre el tablero.

#### ENTRADAS

Interacción del usuario con el periférico apropiado para la aplicación (ratón).

#### SALIDAS

Se mostrará en pantalla el resultado de borrar en el tablero de dibujo habiendo seleccionado el color blanco y por lo tanto, eliminando las casillas del tablero pintadas.

#### RELACIÓN ENTRE ENTRADAS Y SALIDAS

La interacción del usuario en el tablero de dibujo será la salida mostrada en este proceso.

**PARAMETROS**

Para poder borrar, necesitaremos haber seleccionado el color blanco previamente.

**REQUISITOS LÓGICOS**

No constan.

**3.2.3 Requisito funcional 3**

Número de requisito	RF3
Nombre de requisito	Activar el pincel de dibujo
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Dibujo en el tablero
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

**INTRODUCCIÓN**

La aplicación debe permitir activar el pincel para poder dibujar en el tablero de dibujo.

**SECUENCIA DE OPERACIONES**

Mediante el ratón, haremos click para activar el pincel.

**RESPUESTA A SITUACIONES ANORMALES**

Si el usuario no activa el pincel, no podrá dibujar.

**ENTRADAS**

Interacción del usuario con el periférico apropiado para la aplicación (ratón).

**SALIDAS**

Se mostrará en pantalla el estado del pincel tras ser activado en la parte inferior de la paleta de colores.

**RELACIÓN ENTRE ENTRADAS Y SALIDAS**

La interacción del usuario con el ratón permitirá dibujar y mostrar el pincel activado en el estado.

**PARAMETROS**

Para poder activarlo, necesitaremos conocer el estado del pincel y el evento accionado.

**REQUISITOS LÓGICOS**

No constan.

**3.2.4 Requisito funcional 4**

Número de requisito	RF4
Nombre de requisito	Desactivar el pincel de dibujo
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Dibujo en el tablero
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

**INTRODUCCIÓN**

La aplicación debe permitir desactivar el pincel para parar de dibujar en el tablero.

**SECUENCIA DE OPERACIONES**

Mediante el ratón, haremos click para desactivar el pincel.

**RESPUESTA A SITUACIONES ANORMALES**

Si el usuario no desactiva el pincel, el pincel estará habilitado para dibujar en el tablero.

**ENTRADAS**

Interacción del usuario con el periférico apropiado para la aplicación (ratón).

**SALIDAS**

Se mostrará en pantalla el estado del pincel tras ser desactivado en la parte inferior de la paleta de colores.

**RELACIÓN ENTRE ENTRADAS Y SALIDAS**

La interacción del usuario con el ratón deshabilitará el pincel y mostrara el pincel desactivado en el estado.



**PARAMETROS**

Para poder desactivarlo, necesitaremos conocer el estado del pincel y el evento accionado.

**REQUISITOS LÓGICOS**

No constan.

**3.2.5 Requisito funcional 5**

Número de requisito	RF5
Nombre de requisito	Seleccionar un color de la paleta
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Paleta de colores
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

**INTRODUCCIÓN**

La aplicación debe permitir seleccionar de la paleta el color que queremos entre los 10 disponibles.

**SECUENCIA DE OPERACIONES**

Mediante el ratón, haremos click para seleccionar el color en la paleta disponible en la interfaz.

**RESPUESTA A SITUACIONES ANORMALES**

Si no selecciona un color de la paleta, el usuario no podrá dibujar en el tablero de dibujo.

**ENTRADAS**

Interacción del usuario con el periférico apropiado para la aplicación (ratón).

**SALIDAS**

Se mostrará en pantalla el resultado de seleccionar sobre la paleta el color. Una vez hecho, la casilla del color estará remarcada.

**RELACIÓN ENTRE ENTRADAS Y SALIDAS**

La interacción del usuario con la paleta será la salida mostrada por el proceso, proporcionando un color al pincel.

**PARAMETROS**

Para poder seleccionarlo, necesitaremos conocer el evento accionado.

**REQUISITOS LÓGICOS**

No constan.

**3.2.6 Requisito funcional 6**

Número de requisito	RF6
Nombre de requisito	Emitir estado del pincel
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Menú de estado
Prioridad del requisito	<input type="checkbox"/> Alta/Esencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

**INTRODUCCIÓN**

La aplicación debe permitir mostrar el estado del pincel tras activarlo o desactivarlo.

**SECUENCIA DE OPERACIONES**

Tras hacer click sobre el ratón, el ratón se activará o desactivará según el estado anterior del mismo. Una vez hecho, el panel de estado cambiará mostrando al usuario el modo en el que se encuentra el pincel.

**RESPUESTA A SITUACIONES ANORMALES**

Si el usuario no activa o desactiva el pincel, el estado no cambiará (por defecto, desactivado).

**ENTRADAS**

Interacción del usuario con el periférico apropiado para la aplicación (ratón).

**SALIDAS**

Se mostrará en pantalla el resultado de activar o desactivar el pincel en el estado del pincel.

**RELACIÓN ENTRE ENTRADAS Y SALIDAS**

La interacción con el dispositivo externo (ratón), dará lugar al cambio de estado.

**PARAMETROS**

Para poder emitirlo, necesitaremos conocer el estado del pincel

**REQUISITOS LÓGICOS**

No constan.

### 3.3 Requisitos no funcionales

#### 3.3.1 Requisitos de rendimiento

Número de requisito	RR1
Nombre de requisito	Tiempo de respuesta al usuario
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Interacción del usuario
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

La aplicación deberá responder a las acciones del usuario sobre la interfaz en tiempo real, sin retardo en la interacción para una experiencia con el software correcta.

Número de requisito	RR2
Nombre de requisito	Número de usuarios simultáneos
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Interacción del usuario
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

El sistema solo permitirá el acceso a la aplicación de forma local de un único usuario.

Número de requisito	RR3
Nombre de requisito	Numero de terminales a manejar
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Interacción del usuario
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

La aplicación correrá sobre un único terminal donde se llevará a cabo la interacción entre usuario y sistema.

### 3.3.2 Seguridad

Número de requisito	RS1
Nombre de requisito	Numero de terminales a manejar
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Interacción del usuario
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

La aplicación correrá a partir de un único terminal.

### 3.3.3 Fiabilidad

Número de requisito	RF1
Nombre de requisito	Ausencia de bugs
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Interacción del usuario
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

El sistema estará desarrollado y revisado con el fin de posibilitar la ausencia completa de bugs (errores de software).

### 3.3.4 Disponibilidad

Número de requisito	RD1
Nombre de requisito	Tiempo de disponibilidad del software
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Interacción del usuario
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

El sistema ha sido desarrollado tomando en cuenta las necesidades, requerimientos y reglas del cliente. Por lo tanto, la aplicación se encontrará disponible el 100% del tiempo en el ordenador local.

### 3.3.5 Mantenibilidad

Número de requisito	RM1
Nombre de requisito	Generación de reportes de errores
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Interacción del usuario
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Generación cada cierto tiempo de un reporte de errores producidos en la interacción con el usuario

### 3.3.6 Portabilidad

Número de requisito	RP1
Nombre de requisito	Uso de lenguajes de programación portables
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Interacción del usuario
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Los lenguajes de programación usados permiten portar la aplicación a la mayoría de sistemas disponibles actualmente

## 3.4 Otros requisitos

Número de requisito	RO1
Nombre de requisito	Elementos en regla
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Interacción del usuario
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

La aplicación deberá estar en regla con todos los artículos legales afectados en este caso.

## 4 Apéndices

N/A

## 5 Introducción de la solución

Tras llevar a cabo el proceso de especificación de requisitos de software requerido para extraer los elementos esenciales del proyecto a realizar, nos centraremos en la descripción de la solución implementada para conseguir las necesidades del cliente (práctica).

La aplicación que debemos desarrollar, como hemos comentado anteriormente, se identificará como: "Tablero de dibujo". Esta aplicación brindará las funciones principales de un software con un uso sencillo e intuitivo, con el objetivo de ser accesible para dibujar a cualquier usuario sin importar su edad.

La solución se describirá en dos apartados: interfaz y funcionalidad. En la interfaz describiremos los pasos realizados para conseguir un apartado visual cercano al ejemplo proporcionado por el cliente, mediante los lenguajes HTML y CSS. Una vez explicado, nos centraremos en las funcionalidades a implementar mediante el lenguaje de programación web, Javascript.

## 6 Descripción de la solución

La práctica propuesta está compuesta de tres archivos: index02.html, estilo02.css y el código javascript que se denominará ejercicio02.js. Los dos primeros ficheros estarán dedicados al estilo de la aplicación y el último a la funcionalidad de la misma.



## 6.1 Interfaz (HTML)

Mediante el ejemplo proporcionado por la práctica y las directrices a seguir de formato debemos conseguir un aspecto visual similar al siguiente:

Para ello empezaremos definiendo el archivo HTML que nos permitirá definir la estructura base de la página: tablero de dibujo, título de la aplicación, paleta de colores y mensajes descriptivos y de estado.

### - Cabecera del archivo HTML:

En el encontramos en primer lugar la marca **DOCTYPE**, que se refiere al tipo de documento visualizado además de ser necesaria para la comunicación entre navegador y servidor. A continuación la etiqueta **title**, que nos permite establecer el título de la propia ventana del navegador, útil para posibles búsquedas en los buscadores de internet. Se denominará como *"PRÁCTICA 2 – TABLERO DE DIBUJO"*.

Con la etiqueta **meta charset** en HTML especificaremos la codificación de caracteres utilizada en el documento. Por último las referencias a los archivos asociados al documento HTML, *estilos.css* (para el estilo) y *ejercicio02.js* (para la funcionalidad), y el lenguaje utilizado.

### - Contenido del archivo HTML:

El contenido de la página (localizado por las etiquetas **<body></body>** estará englobado por un único div (contenido), que nos permitirá centrar la página posteriormente.

Definiremos en primer lugar con la etiqueta **<p>** el título del contenido (*"TABLERO DE DIBUJO CON JAVASCRIPT"*) y un nuevo **<div>** para cumplir con la restricción de la práctica de utilizar los métodos de creación de nodos del DOM especificándose únicamente en el fichero .html con esta etiqueta y su identificador.

Tras ello, especificaremos mediante las etiquetas **<table>** y **<caption>**, por un lado la tabla que debemos construir para crear nuestra paleta de colores y por otro el título de la misma.

Construiremos dos filas con **<tr>**. La primera estará dividida por tantos **<td>** como sean necesarios para añadir los 10 colores pedidos y la segunda incluirá el estado del pincel (usaremos **colspan** para unificar horizontalmente las celdas).

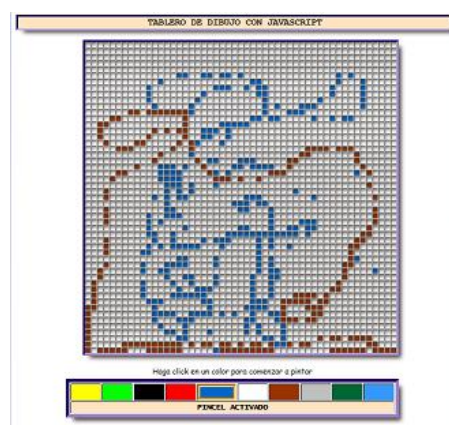
---

## 6.2 Interfaz (CSS)

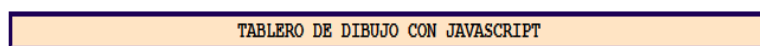
Una vez descrito el código de HTML, aplicaremos el estilo a cada elemento para conseguir el aspecto visual esperado. Empezaremos describiendo el estilo general del contenido.

### 6.2.1 Estilos del contenido:

En este apartado definiremos el estilo del contenido y el título superior del tablero. A la estructura general del contenido de la aplicación se le aplicará una serie de etiquetas de estilo para centrarlo y definir su tamaño con respecto a la ventana del navegador

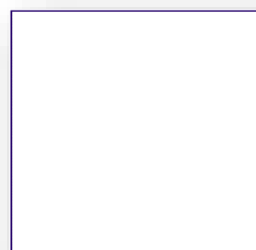


El título por encima del tablero de dibujo llevará un rectángulo de fondo de un determinado color (definido con **border-style**, **border-width** y **border-color**) y sombra con **box-shadow**. Se definirá la fuente, tamaño y estilo del título.



### 6.2.2 Estilo del tablero:

Estilo de la tabla que contiene el tablero. Definiremos la posición del mismo, su margen con respecto al contenido y el **padding** a cero. Añadiremos un borde a la tabla mediante las etiquetas **border-color**, **border-style** y **border-width** y una sombra con **box-shadow** a una determinada distancia y color.





Utilizamos **float** y **clear** para alinear las celdas del tablero. **Float** será útil para alinear un elemento (la tabla), haciendo que las celdas de alrededor del mismo se agrupen entorno a dicho elemento. Junto a **float**, usaremos **clear** para que elementos que tienen cerca un elemento flotante (mediante **float**) dejen de alinearse junto a él, consiguiendo que las celdas estén agrupadas correctamente y sin espacios en blanco en la tabla.



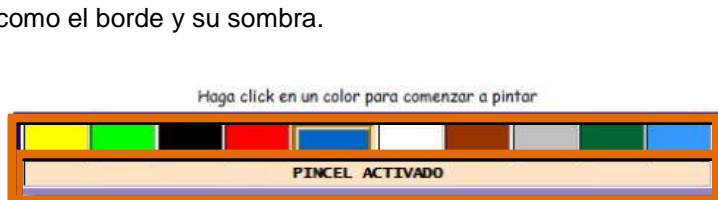
### 6.2.3 Estilo de la paleta:

Una vez definido el estilo del título y el tablero, únicamente nos queda definir el aspecto del menú de selección de colores. Primero de todo, deberemos escoger los 10 colores que formarán nuestra paleta, de los cuales dos de ellos son imprescindibles (negro y blanco).

Por lo tanto, para cada **td** identificado con un nombre de clase "color'X'", tendremos que especificar con **background-color**, el color de cada uno.



Tenemos los colores y necesitamos darle forma a la tabla que los engloba. Para ello especificamos su tamaño, la posición y añadimos elementos secundarios como el borde y su sombra.



Los colores y el formato de la tabla están implementados. Necesitamos definir el estilo de cada sección que cubrirá los colores, evitando que se produzca un descontrol en el tamaño de los mismos mediante **width y height**. El penúltimo paso será especificar la sección del estado del pincel a la cual le hemos añadido un color de fondo, alinear el estado en el centro de la subsección, el tipo de letra y la fuente de la misma.



Por último, aplicaremos estilo al recuadro que marcará la elección escogida por el usuario, remarcando lo máximo posible su borde y color para hacerlo fácilmente distinguible de cara al usuario. Para ello, hemos seleccionado un tipo de borde "Groove", un tamaño de 8px y un color naranja.



## 6.3 Funcionalidad (JAVASCRIPT)

En este apartado procedemos a explicar las funciones de JavaScript usadas para dar funcionamiento al tablero de dibujo, y por tanto que se cumplan (la mayor parte) los requisitos funcionales enunciados en la descripción del problema.

Comenzamos indicando todas las variables necesarias para la implementación de las funcionalidades, destacando los arrays usados para almacenar los colores o las variables creadas que guardan elementos u obteniendo de HTML. Para tener acceso a los colores, los recorremos y guardamos en un array(vector).

### 6.3.1 Asignar paleta

Función encargada de seleccionar el color con el que se desea pintar en la cuadrícula. Para ello se recorre la lista de colores y asignamos cada color al evento "mousedown", basado en la pulsación de un botón del ratón sobre un documento, enlace o form.

### 6.3.2 Cambiar color

Se encarga de cambiar el color seleccionado por un nuevo color. Para ello recorreremos los colores y comprobamos si el color seleccionado es un color disponible en el array. Si es así, será seleccionado y se le añadirá una marca que lo distinga de los otros ("seleccionado").

### 6.3.3 Dibujar tabla

Se crea la cuadrícula del tablero mediante las opciones **tr** y **td**, con un tamaño de 50x50 y asignándole un identificador a cada uno de las celdas del tablero. Para la creación de la tabla se mezclan dos bucles en los que se crean las celdas con las opciones **tr** y **td**.

La asignación del estilo se realiza en el archivo ".css" al cual hacemos referencia asignando un id a cada una de las secciones **div** (el estilo de los **td** vendrá descrito en el propio fichero *javascript*). Esta identificación nos servirá para pintar solamente en las celdas sobre las que pasa el ratón. Para finalizar la creación de la tabla se añade al **td** los nodos de **tr**, a la tabla la lista de **tr** y al tablero la tabla completa.)

### 6.3.4 Paint

Comprobamos el estado del pincel y actuamos en función de sus dos posibles estados:

- Activo: para identificar este estado cambiamos el tipo de puntero y mostramos en el panel de texto que el pincel se encuentra activo. A continuación se recorre la cuadrícula y se va pintando con el color seleccionado activando el evento *mouseover* mediante el método *addEventListener()*. Como añadido, implementamos un cambio de aspecto al puntero cuando el pincel está activo mediante *body.style.cursor*.
  - Desactivado: para identificar este estado establecemos el puntero por defecto del ratón y mostramos en el panel de texto que el pincel se encuentra desactivado. Recorremos la cuadrícula y desactivamos la función de *pintar()*; mediante el método *removeEventListener()*. Como añadido, implementamos un cambio de aspecto al puntero cuando el pincel está desactivado mediante *body.style.cursor* a su forma original.
-

### 6.3.5 Pintar

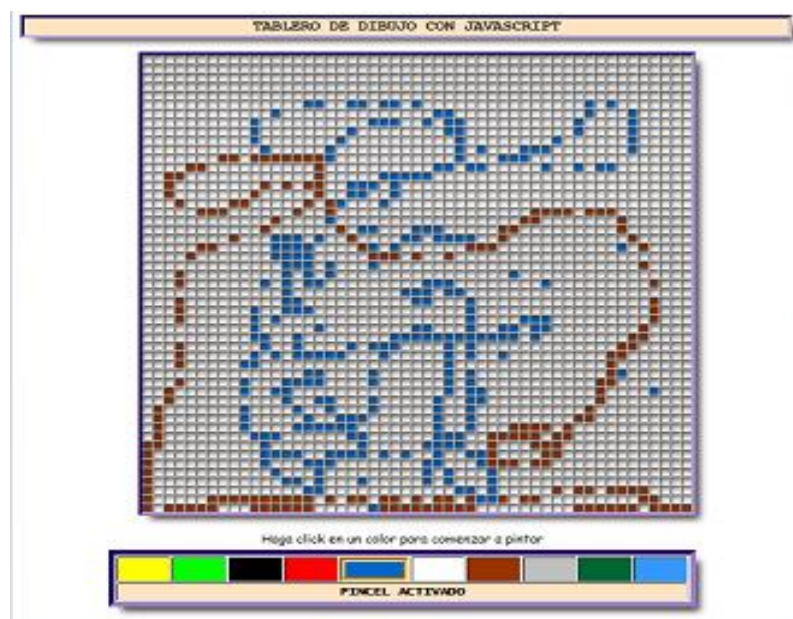
En esta función permite, a partir del color seleccionado, pintar en la cuadrícula mediante la llamada al método *addEventListener()* en la función *Paint()*.

Tras la implementación de los métodos se realizan las llamadas necesarias a los métodos para su correcta ejecución. En primer lugar se muestra una ventana emergente en la que se especifican los pasos a seguir para la utilización de la aplicación. Y a continuación la sucesión de llamadas a los métodos creados: *dibujarTabla*, *asignarPaleta*, *tablero.addEventListenerpaint*.

Por último, *window.onload* nos permitirá iniciar la función cuando la ventana se haya cargado completamente.

## 7 Conclusión

En definitiva, la práctica ha abarcado el diseño de una aplicación interactiva con el usuario basado en un tablero de dibujo sencillo e intuitivo. Es un proyecto que nos permite obtener conocimientos sobre el lenguaje Javascript y mejorar lo aprendido sobre HTML y CSS en el diseño y estructura de la aplicación web. Sin duda es un programa básico de dibujo al cual podría añadirse numerosas funciones con respecto a las solicitadas y un estilo visual más atractivo.



## 8 Bibliografía

- [www.w3schools.com/](http://www.w3schools.com/)
  - [www.prado.ugr.es/](http://www.prado.ugr.es/)
  - [www.librosweb.es/](http://www.librosweb.es/)
  - [www.codecademy.com](http://www.codecademy.com)
  - [http://es.wikipedia.org/wiki/Especificaci%C3%B3n\\_de\\_requisitos\\_de\\_software](http://es.wikipedia.org/wiki/Especificaci%C3%B3n_de_requisitos_de_software)
  - <http://www.infor.uva.es/~descuder/proyectos/ipo/requi.htm>
  - <http://www.javascript.info/>
  - [http://www.tutorialspoint.com/javascript/javascript\\_events.htm](http://www.tutorialspoint.com/javascript/javascript_events.htm)
-