

Proyecto 3

Bootcamp Online

Frontend Developer



GeeksHubs
academy__

Introducción

Ladies and gentlemen...start your engines!!! Ha llegado el turno de poner a prueba los conocimientos que hemos adquirido durante estas primeras semanas. Se nos encomienda emular el juego del Mastermind con JS.

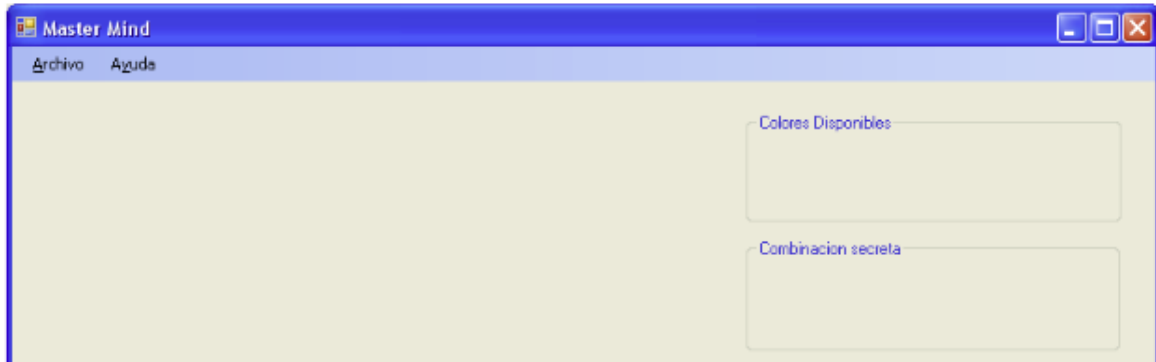


El objetivo es implementar una aplicación que permita jugar al juego del MasterMind. Se juega en un tablero con fichas blancas y negras pequeñas y de otros colores. Uno de los jugadores escoge un número de fichas de colores, 4 en el juego original, y pone un código secreto (combinación de colores) oculto al otro jugador. Éste, tomando fichas de colores del mismo conjunto, aventura una posibilidad contestada con negras (fichas de color bien colocadas) o blancas (fichas de color con el color correcto, pero mal colocadas).

El juego termina al averiguarse la combinación (es decir, se consigue una combinación con cuatro negras), o bien se agota el tablero (depende del tamaño, aunque generalmente son 10 combinaciones).

Requisitos funcionales

RF1 - SITUACIÓN INICIAL



En un panel superior derecho, incluiremos los colores disponibles. Estos serán variables y dependerá del nivel que elija el jugador (por defecto, serán 4). Se deberá implementar un único procedimiento (crear_colores) que cree en tiempo de ejecución tantos elementos, llamados “bolacolors”, como sean necesarios. El color de cada uno de estos componentes se obtendrá de un vector de Colores, previamente creado e inicializado.

En un panel inferior derecho, mostraremos la solución del juego (inicialmente no será visible el panel). Se deberá implementar un único procedimiento (crear_solucion) que cree en tiempo de ejecución tantos elementos, llamados “bolasolucion”, como sean necesarios.

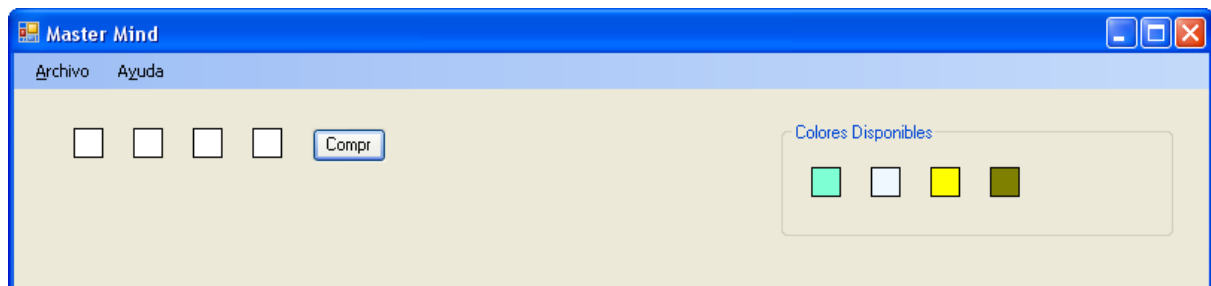
El color de cada uno de estos componentes se obtendrá aleatoriamente del vector de Colores arriba citado. Los colores de la solución se podrán repetir. Sería aconsejable guardar la solución en un vector.

RF2 - INICIANDO EL JUEGO

El menú Archivo del juego, tendrá únicamente 2 opciones: Salir y Nuevo Juego. Con este último se comenzará la primera partida, y se podrá acabar la partida actual.

Cada nueva partida se inicia mostrando los componentes vacíos para que el jugador pueda poner su combinación. Se deberá implementar un único procedimiento (crear_linea_bola) que cree en tiempo de ejecución tantos elementos, llamados “bola”, como sean necesarios. El número de estos componentes será una constante con valor 4. Junto a estos componentes, crearemos en el mismo procedimiento un botón que permita comprobar cuántos colores hemos acertado.

Es necesario también crear un procedimiento (borrar_componentes) que permita borrar todos los componentes creados durante el juego, cuando se pulsa Nuevo Juego.



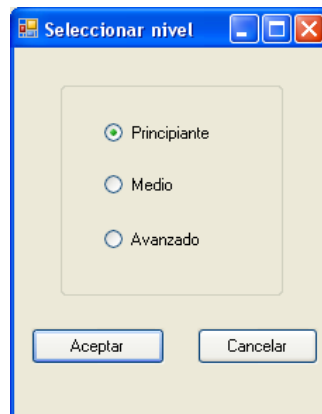
Finalmente, se deberán crear sendos formularios Como Jugar y Acerca De (donde deberá aparecer el nombre del alumno), dentro de un menú Ayuda.

RF3 - ELIGIENDO EL NIVEL DE JUEGO

Al pulsar un nuevo juego debemos pedir al usuario que nos indique el nivel al que desea jugar, si es que no lo ha indicado anteriormente.

Añadiremos también una opción al menú Archivo, denominada Nivel, que nos permitirá seleccionar el mismo.

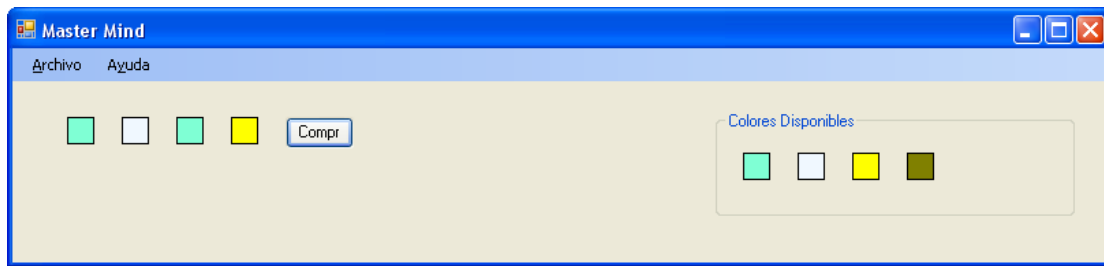
- Si el usuario selecciona Principiante, se jugarán con las opciones por defecto del juego: 4 colores posibles y 10 intentos. Si elige Medio, tendrá 8 intentos y 5 colores donde poder elegir y si escoge Avanzado tendrá 6 intentos y 6 colores a combinar.
- Si el usuario cancela al seleccionar el nivel, jugará con las opciones de Principiante.
- Si el usuario pulsa el botón Aceptar se iniciará un nuevo juego con las opciones seleccionadas.



La partida puede ser de humano vs humano o humano vs cpu, en este último caso esta seleccionará la combinación de colores oculta y validará cada tirada. Crea una vista de selección de jugadores a tal efecto.

RF4 - PONIENDO COLORES A NUESTROS COMPONENTES

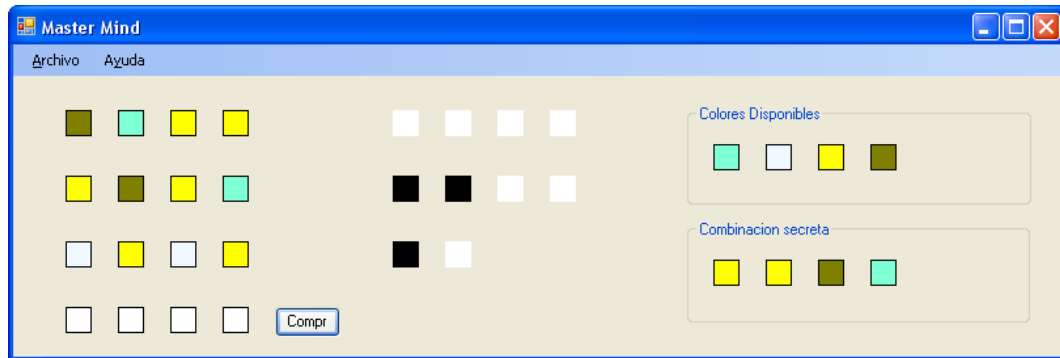
Con los colores disponibles debemos ir probando combinaciones hasta intentar dar con la solución. Para poder indicar que colores queremos incluir en nuestra combinación realizaremos un procedimiento cuando ganamos clic con el ratón en cada uno de los componentes creados, de la siguiente forma:



- Al pulsar el botón izquierdo del ratón iremos poniendo en el componente los colores disponibles de derecha a izquierda. Es decir, la primera vez que haga clic izquierdo y, siguiendo el ejemplo de la imagen de arriba, pondré el color azul claro, con el siguiente clic izquierdo el gris, después el amarillo y finalmente el verde. Si continúa haciendo clic, volveré a la situación inicial, mostrando en el componente el color azul claro.
- Al pulsar el botón derecho, el comportamiento es similar al de arriba, pero de derecha a izquierda. El primero será el verde y, sucesivamente, amarillo, gris y azul.
- Se pueden alternar las pulsaciones de derecha e izquierda.
- Cada clic es independiente del componente. Una vez que empiezo con el primer color de los disponibles, independientemente de si pincho en el mismo componente o en otro distinto, seguiré con el siguiente color de los disponibles. No por pinchar en un componente en blanco, empiezo por el primer color, sino que sigo por donde iba según la secuencia de colores disponibles.

RF5 - COMPROBANDO NUESTROS ACIERTOS

Cuando pulsamos el botón de comprobar debemos comprobar cuantos colores están encontrados y en su sitio (lo marcaremos con bolas negras) y cuantos colores están, pero no en su sitio (los marcaremos con bolas blancas).



- Comprobando las negras: debemos comprobar cuantos colores están correctamente encontrados y crearemos a la derecha del botón tantos elementos con color negro como colores hayamos encontrado, llamados “bolaresul”.
- Comprobando las blancas: debemos comprobar cuantos colores están, pero no se encuentran en su sitio. Debemos eliminar de esta comprobación los colores encontrados en su sitio (negras) y si un color está, pero mal situado, en más de una posición solo podemos marcarlo como una vez. Crearemos a la derecha de las negras tantos controles, blancos, como colores fuera de su sitio encontremos, llamados “bolaresul”.

Tras mostrar las bolas negras y blancas, debemos comprobar dos cosas:

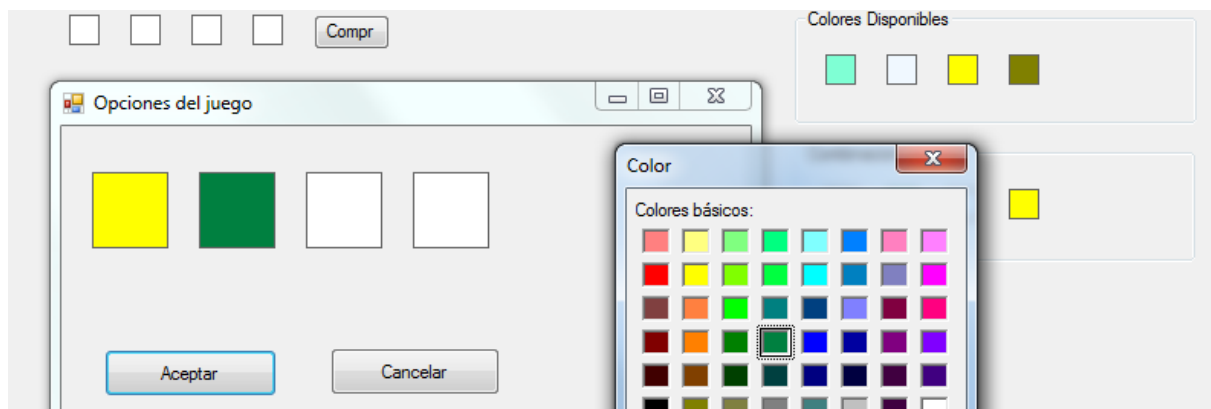
- Hemos ganado, mostrando un mensaje al usuario.
- Hemos gastado todos nuestros intentos, mostrando un mensaje al usuario y mostrándole la combinación secreta para que compruebe el resultado. El número de intentos dependerá del nivel elegido, siendo por defecto de 10. No podremos pulsar de nuevo Comprobar.

Si no hemos ganado ni perdido, seguiremos jugando. Nos crearemos otra fila de componentes vacíos bajo la anterior para que el usuario pueda elegir los colores

de nuevo. Además, reutilizaremos el mismo botón de comprobar para comprobar cada nuevo intento.

RF6 - ELIGIENDO LOS COLORES DISPONIBLES

Añadiremos una opción de menú, llamada Opciones, donde se nos abrirá una ventana con tantos elementos como colores tengamos dependiendo del nivel del juego. Esta opción estará inhabilitada hasta jugar al menos una vez. Al pinchar en cada elementos (inicialmente en blanco) se abrirá un panel de colores para seleccionar el color que deseemos. Al aceptar pasaremos los colores seleccionados al panel de colores de la ventana principal e iniciaremos un nuevo juego con los nuevos colores, que se utilizarán en las próximas partidas.



Descripción

Teniendo en cuenta los requisitos, llegamos a la conclusión de que este sería el mínimo de vistas convenientes:

- Vista Home/ Bienvenida / Press start button.
- Vista selección de nivel.
- Vista Selección de jugadores.
- Vista Panel con el tablero de juego y la información del turno en juego.
- About.

- Vista ganador.

Obligatorio

Para el desarrollo del frontend se utilizará HTML5 en combinación con JS (haciendo uso de la nomenclatura ES6) y CSS3 / Layout.

El proyecto se subirá a un repositorio público de GitHub y se valorará la existencia de ramas así será un requisito indispensable la inclusión de un Readme completo. Su subida a producción (Github Pages) será un requisito imprescindible).

Mejoras opcionales

Para llegar al 10:

- Buena explicación del mismo en readme/s.
- Uso de ilustraciones y animaciones (GIF, svg, etc).
- Uso de programación orientada a objetos.
- Tener varias ramas en GIT.
- Código bien comentado.
- Código ordenado e indentado.

Dudas

Para cualquier duda puedes dirigirte al Slack de Geekshubs. En el grupo formativo al que perteneces.